

VF2000 串口控制协议

版本号	更改日期	更新内容
V0.0.1	2017.05.31	P3. BYT5 增加控制字节
V0.0.2	2017.09.27	P15. 增加输出卡水平和垂直偏移值的读取; P10. 增加输出卡校正数据保存命令;
V0.0.3	2017.10.18	P9. 更改“保存所有窗口数据”命令; P6. 增加“ADC 自动校正”命令; P6. 增加“烧录 EDID”命令; P13.增加返回“当前显示模式”数据; P18.读取 Overlay 及窗口显示数据增加“显示模式参数”; P19.读取 Win 数据增加“显示模式参数”;
V0.0.4	2017.10.24	P20.增加“调用模式并返回模式相关数据”命令; P11.增加“保存 LED 构建数据+所有窗口命令”命令; P23.增加“读取设备总数据”命令;
V0.0.5	2017.10.26	P21.“调用模式并返回模式相关数据”命令返回增加“LED 构建数据”字段;
V0.0.6	2017.10.27	P27.增加“设备数据备份”命令; P27.“读取设备总数据”命令返回增加“LED 水平和垂直数目”字段;
V0.0.7	2017.10.28	P12.增加“导入输出卡+LED 构建数据”命令; P14.增加“导入窗口数据”命令; P15.增加“数据加载”命令; P29.“设备数据备份”去除所有模式,必须一个模式一个模式备份;
		P9. 执行输出卡协议并保存窗口数据 P25. P29.
V0.0.9	2017.11.29	“读取设备总数据”返回增加“输入卡 OSD 字符数据”; 增加“读取软件版本号”命令;
V0.1.0	2017.12.04	增加“设置输出卡应用模式”命令
V0.1.1	2017.12.13	增加“设备 LAN 设置”命令 增加“读取 LAN 设置”命令

V0.1.2	2017.12.14	增加“ 执行输出卡协议并保存窗口数据 NEW ”命令
V0.1.3	2017.12.18	增加“ 系统初始化 ”命令
V0.1.4	2018.01.21	“ 调用模式并返回模式相关数据 ”命令返回增加“ 边框数据 ”字段; “ 读取设备总数据 ”命令返回增加“ 边框数据 ”字段;
V0.1.5	2018.06.25	增加“ 测试输出卡命令 (AA) ”命令

VF2000 软件版本 v0.0.0 及以后更高版本已开放部分 RS232 串口/LAN/USB 命令 (v0.1.5)

一、RS232 串行通讯协议

波特率：115200

无奇偶校验

8 位数据位

1 位停止位

二、RS232 命令格式

每条命令由 n 个字节的 data 串组成，定义为：

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYTn
0	1	2	3	4	5	6	7	8				

BYT0: 设备类型号，对于 **VF2000**，**BYT0**=0x20;

BYT1: 受控设备的编号，范围从 00~FE 共 255 个编号，0xFF 表示所有设备受控;

BYT2: 命令总长度 n+1 高位;

BYT3: 命令总长度 n+1 低位;

BYT4: 命令目标：0-输入卡/1-输出卡/2-底板;

BYT5: 控制字节

0x01-表示受控设备在收到该命令并完成操作后返回命令执行结果;

0x00-表示受控设备不返任何数据;

0xFE-表示读取命令;

0xFF-表示测试命令，受控设备返回该命令；

BYT6~BYTn-1: 命令参数；

BYTn: 前 **BYT0~BYTn-1** 个字节数据的 异或 或者 设备类型号(0x20) 校验值 **ChkSum**；

*建议使用前 **BYT0~BYTn-1** 个字节数据的 异或 作为校验值

命令返回：

详见(六)命令处理结果返回值详解；

三、输入卡控制命令

BYT0 = 0x20: VF2000 设备类型号；

BYT1 = 设备编号: 0x00~0xFE；

BYT2 = 0x00: 20 个字节长度命令高位；

BYT3 = 0x14: 20 个字节长度命令低位；

BYT4 = 0x00: 表示操作输入卡；

BYT5 = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果；

= 0x00: 表示受控设备不返任何数据；

= 0xFE: 表示读取命令；

= 0xFF: 表示测试命令，受控设备返回该命令；

BYTE6~BYTE18: 13 个字节的输入卡命令；

BYTE6 = 0x86: 固定字节；

BYTE7 = 0x00~0x0F: 输入卡号；

BYTE8 = cmd: 命令字节，详见下面 **BYTE5~BYTE17** 详解；

BYTE9~BYT17: 命令参数；

BYTE18 = Chksum, **BYTE5~BYTE16** 异或运算或者 0x86；

BYT19 = Checksum/0x20: **BYT0~BYT18** 校验码；

BYTE6~BYTE18 详解：

1、切换输入卡输入信号源（01）

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8

86	XX	01	XX	Chksu m
----	----	----	----	------------

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=01**: "切换输入卡输入信号源"命令;
- 4) **BYT9=00**: 切换至 SDI 通道;
 =01: 切换至 DVI 通道;
 =02: 切换至 VGA 通道;
 =03: 切换至 VIDEO 通道;
- 5) **BYT10 到 BYT17** 无实际意义, 置为 0;
- *该命令只在切换状态下有效

2、进入 PIP 准备状态或退出 PIP 准备状态/PIP 状态 (02)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	02	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=02**: "进入 PIP 准备状态或退出 PIP 准备状态/PIP 状态"命令
- 4) **BYT9=00**: 退出 PIP 准备状态/PIP 状态, 回到切换状态, 仅在 PIP 准备状态/PIP 状态有效;
 =01: 进入 PIP 准备状态, 仅在切换状态有效;
- 5) **BYT10 到 BYT17** 无实际意义, 置为 0;

3、进入 PIP 状态并切换 PIP 输入信号源 (03)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	03	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节

- 2) **BYT7=0~15**: 输入卡号;
 - 3) **BYT8=03**: "进入 PIP 状态并切换 PIP 输入信号源"命令;
 - 4) **BYT9=00**: 切换至 SDI 通道;
 - =01: 切换至 DVI 通道;
 - =02: 切换至 VGA 通道;
 - =03: 切换至 VIDEO 通道;
 - 5) **BYT10 到 BYT17** 无实际意义, 置为 0;
- *该命令只在 PIP 准备状态/PIP 状态有效

4、进入 TEXT 准备状态或退出 TEXT 准备/TEXT 状态 (04)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	04	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
 - 3) **BYT8=04**: "进入 TEXT 准备状态或退出 TEXT 准备状态/TEXT 状态"命令
 - 4) **BYT9=00**: 退出 TEXT 准备/TEXT 状态, 回到切换状态, 仅在 TEXT 准备/TEXT 状态有效;
 - =01: 进入 TEXT 准备状态, 仅在切换状态有效;
 - 5) **BYT10 到 BYT17** 无实际意义, 置为 0;

5、进入 TEXT 状态并切换 TEXT 输入信号源 (05)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	05	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
 - 3) **BYT8=05**: "进入 TEXT 状态并切换 TEXT 输入信号源"命令;
 - 4) **BYT9=00**: 切换至 SDI 通道;

- =01: 切换至 DVI 通道;
- =02: 切换至 VGA 通道;
- =03: 切换至 VIDEO 通道;

5) **BYT10** 到 **BYT17** 无实际意义, 置为 0;

*该命令只在 TEXT 准备状态/TEXT 状态有效

6、设置 PIP 主信号源大小位置 (06)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8
86	XX	06	XX	Chksu m

说明 1) **BYT6=86**: 固定定节

- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=06**: "设置 PIP 主信号源大小位置"命令;
- 4) **BYT9**=水平起始低 8 位;
- 5) **BYT10**=水平起始高 8 位;
- 6) **BYT11**=宽度低 8 位;
- 7) **BYT12**=宽度高 8 位;
- 8) **BYT13**=垂直起始低 8 位;
- 9) **BYT14**=垂直起始高 8 位;
- 10) **BYT15**=高度低 8 位;
- 11) **BYT16**=高度高 8 位;
- 12) **BYT17** 无实际意义, 置为 0;

*请对设置的值作范围限制:

*水平起始+图像宽度<输出分辨率最大宽度;

*垂直超始+图像高度<输出分辨率最大高度;

7、设置 PIP 子信号源大小位置 (07)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8

86	XX	07	XX	Chksu m
----	----	----	----	------------

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=06**: "设置 PIP 子信号源大小位置"命令;
- 4) **BYT9**=水平起始低 8 位;
- 5) **BYT10**=水平起始高 8 位;
- 6) **BYT11**=宽度低 8 位;
- 7) **BYT12**=宽度高 8 位;
- 8) **BYT13**=垂直起始低 8 位;
- 9) **BYT14**=垂直起始高 8 位;
- 10) **BYT15**=高度低 8 位;
- 11) **BYT16**=高度高 8 位;
- 12) **BYT17** 无实际意义, 置为 0;

*请对设置的值作范围限制:

*水平起始+图像宽度<输出分辨率最大宽度;

*垂直超始+图像高度<输出分辨率最大高度;

8、设置亮度 (08)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	08	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=08**: "设置亮度"命令;
- 4) **BYT9=0~100**: 亮度值;
- 5) **BYT10** 到 **BYT17** 无实际意义, 置为 0;

9、设置对比度 (09)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	09	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=09**: "设置对比度"命令;
- 4) **BYT9=0~100**: 对比度值;
- 5) **BYT10** 到 **BYT17** 无实际意义, 置为 0;

10、设置颜色 (0A)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	0A	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=0A**: "设置颜色"命令;
- 4) **BYT9=0~100**: 颜色值;
- 5) **BYT10** 到 **BYT17** 无实际意义, 置为 0;

11、设置 TEXT 属性 (0B)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	0B	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=0B**: "设置 TEXT 属性"命令;
- 4) **BYT9=0**, 小于阈值抠像;
=1, 大于阈值抠像;

- 5) **BYT10**=0~255, 字幕阈值红;
- 6) **BYT11**=0~255, 字幕阈值绿;
- 7) **BYT12**=0~255, 字幕阈值蓝;
- 8) **BYT13** 到 **BYT17** 无实际意义, 置为 0;

12、VGA 自动校正 (0C)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8
86	XX	0C	XX	Chksu m

- 说明 1) **BYT6**=86: 固定定节
- 2) **BYT7**=0~15: 输入卡号;
 - 3) **BYT8**=0C: "VGA 自动校正"命令;
 - 4) **BYT9** 到 **BYT17** 无实际意义, 置为 0;
- *该命令只在切换状态且 VGA 信号源有效;

13、输入卡初始化 (0D)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8
86	XX	0D	XX	Chksu m

- 说明 1) **BYT6**=86: 固定定节
- 2) **BYT7**=0~15: 输入卡号;
 - 3) **BYT8**=0D: "输入卡初始化"命令;
 - 4) **BYT9** 到 **BYT17** 无实际意义, 置为 0;

14、设置 OSD 字符 (0E)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T17	8
86	XX	0E	XX	Chksu

				m
--	--	--	--	---

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=0E**: "设置 OSD 字符"命令;
- 4) **BYT9=00**: SDI 通道字符;
 =01: DVI 通道字符;
 =02: VGA 通道字符;
 =03: VIDEO 通道字符;
- 5) **BYT10~ BYT17**=第 1~8 个英文字符 ASCII 码;

15、设置 OSD 字符开关 (0F)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	0F	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=0F**: "设置 OSD 字符开关"命令;
- 4) **BYT9=00**: OSD 字符不显示;
 =01: OSD 字符显示;
- 5) **BYT10~ BYT17** 无实际意义, 置为 0;

16、ADC 自动校正 (E0)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	E0	XX	Chksu m

- 说明 1) **BYT6=86**: 固定定节
- 2) **BYT7=0~15**: 输入卡号;
- 3) **BYT8=E0**: "ADC 自动校正"命令;

5) **BYT9~BYT17** 无实际意义，置为 0；

*该命令仅供工厂调试使用；

17、烧录 EDID (E1)

BYT 6	BYT 7	BYT 8	BYT9~BY T17	BYT1 8
86	XX	E1	XX	Chksu m

说明 1) **BYT6=86**: 固定定节

2) **BYT7=0~15**: 输入卡号；

3) **BYT8=E1**: ” 烧录 EDID”命令；

4) **BYT9=00**: VGA EDID；

=01: OSD 字符显示；

5) **BYT10~BYT17** 无实际意义，置为 0；

四、输出卡控制命令

BYT0 = 0x20: VF2000 设备类型号；

BYT1 = 设备编号: 0x00~0xFE；

BYT2 = 0x05: 1480 个字节长度命令高位；

BYT3 = 0xC8: 1480 个字节长度命令低位；

BYT4 = 0x01: 表示操作输出卡；

BYT5 = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果；

= 0x00: 表示受控设备不返任何数据；

= 0xFE: 表示读取命令；

= 0xFF: 表示测试命令，受控设备返回该命令；

BYTE6~BYTE1478: 1473 个字节的输出卡命令；

BYTE6 = : 固定字节；

BYTE7 = : 固定字节；

BYTE8 = : 固定字节；

BYTE9~BYT1478: 固定字节；

BYT1479 = Checksum/0x20: **BYT0~BYT1478** 校验码;

*****未完待续*****

五、系统卡控制命令

BYT0 = 0x20: VF2000 设备类型号;

BYT1 = 设备编号: 0x01~0xFF;

BYT2 = 0x00: 43 个字节长度命令高位;

BYT3 = 0x2B: 43 个字节长度命令低位;

BYT4 = 0x02: 表示操作底板;

BYT5 = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

= 0xFE: 表示读取命令;

= 0xFF: 表示测试命令, 受控设备返回该命令;

BYTE6~BYTE41: 36 个字节的系统卡命令;

BYTE6 = 0x01: 固定字节;

BYTE7 = 0x01: 系统卡号, 只有一块系统卡 0x01;

BYTE8 = cmd: 命令字节, 详见下面 **BYTE6~BYTE41** 详解;

BYTE9~BYT40: 命令参数;

BYTE41 = Chksum, **BYTE6~BYTE41** 异或运算或者 0x01;

BYT42 = Checksum/0x20: **BYT0~BYT41** 校验码;

BYTE6~BYTE41 详解:

1、切换矩阵 (00)

BYT	BYT	BYT	BYT9~BY	BYT4
6	7	8	T40	1
01	01	00	XX	Chksu

				m
--	--	--	--	---

说明 1) **BYT6=01**, 固定字节;

2) **BYT7=01**, 系统卡号, 只有一块系统卡 0x01;

3) **BYT8=01**, 切换矩阵命令;

4) **BYT9=00~0F**, Out1 第 1 个输入口选择的输入卡 In1~In16;

5) **BYT10=00~0F**, Out1 第 2 个输入口选择的输入卡 In1~In16;

6) **BYT11=00~0F**, Out1 第 3 个输入口选择的输入卡 In1~In16;

7) **BYT12=00~0F**, Out1 第 4 个输入口选择的输入卡 In1~In16;

8) **BYT13=00~0F**, Out2 第 1 个输入口选择的输入卡 In1~In16;

.....

11) **BYT16=00~0F**, Out2 第 4 个输入口选择的输入卡 In1~In16;

12) **BYT17=00~0F**, Out3 第 1 个输入口选择的输入卡 In1~In16;

.....

35) **BYT40=00~0F**, Out8 第 4 个输入口选择的输入卡 In1~In16;

36) **BYT41=Chksum**, **BYTE6~BYTE40** 异或运算或者 0x01;

* **BYTE9~BYTE40** 初始值为 0x00;

*****未完待续*****

六、窗口保存及控制命令

BYT0 = 0x20: VF2000 设备类型号;

BYT1 = 设备编号: 0x00~0xFE;

BYT2 = 0x08: 2048 个字节长度命令高位;

BYT3 = 0x00: 2048 个字节长度命令低位;

BYT4 = 0x03: 表示窗口保存及控制;

BYT5 = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

= 0xFE: 表示读取命令;

= 0xFF: 表示测试命令, 受控设备返回该命令;

BYTE6~BYTE2046: 协议数据;

BYT2047 = Checksum/0x20: **BYT0~BYT2046** 校验码;

BYTE6~BYTE2046 详解:

1、保存 LED 构建数据 (00)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	00	XX	Chksu m

说明 1) **BYT6=AA**, 固定字节;

2) **BYT7=01**, 固定字节;

3) **BYT8=00**, 保存 LED 构建数据命令;

4) **BYT9=LED** 水平方向屏数 (HxV);

5) **BYT10=LED** 垂直方向屏数 (HxV);

6) **BYT11=LED** 水平总宽度低 8 位;

7) **BYT12=LED** 水平总宽度高 8 位;

8) **BYT13=LED** 水平总高度低 8 位;

9) **BYT14=LED** 水平总高度高 8 位;

10) **BYT15=LED1** 水平起始低 8 位;

11) **BYT16= LED1** 水平起始高 8 位;

12) **BYT17= LED1** 宽度低 8 位;

13) **BYT18= LED1** 宽度高 8 位;

14) **BYT19= LED1** 垂直起始低 8 位;

15) **BYT20= LED1** 垂直起始高 8 位;

16) **BYT21= LED1** 高度低 8 位;

17) **BYT22= LED1** 高度高 8 位;

18) **BYTE23~BYTE30= LED2** 尺寸数据;

19) **BYTE31~BYTE38= LED3** 尺寸数据;

.....

20) **BYTE519~BYTE526= LED64** 尺寸数据;

21) **BYTE527~BYTE2045:** 保留, 填充 0;

22) **BYT2046**= Checksum/0xAA: **BYT6~BYT2045** 校验码;

2、执行输出卡协议并保存窗口数据 (01)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	01	XX	Chksu m

说明 1) **BYT6**=AA, 固定字节;

2) **BYT7**=01, 固定字节;

3) **BYT8**=01, 执行输出卡协议并保存窗口数据命令;

4) **BYT9~BYT1677**: 输出卡协议, 详见输出卡协议文档;

5) **BYT1678**=窗口 ID;

6) **BYT1679**=窗口是否显示;

7) **BYT1680**=窗口信号源;

8) **BYT1681**=保留;

9) **BYT1682~BYT1697**=该窗口在 LED1~LED16 中占用的 CH;

10) **BYT1698**=虚拟窗口水平起始低 8 位;

11) **BYT1699**= 虚拟窗口水平起始高 8 位;

12) **BYT1700**= 虚拟窗口宽度低 8 位;

13) **BYT1701**=虚拟窗口宽度高 8 位;

14) **BYT1702**=虚拟窗口垂直起始低 8 位;

15) **BYT1703**= 虚拟窗口垂直起始高 8 位;

16) **BYT1704**= 虚拟窗口高度低 8 位;

17) **BYT1705**=虚拟窗口高度高 8 位;

26) **BYT1706~BYT1769**=窗口 1~窗口 64 的 Overlay;

28) **BYT1770~BYTE2045**: 保留, 填充 0;

29) **BYT2046**= Checksum/0xAA: **BYT6~BYT2045** 校验码;

3、窗口数据初始化 (02)

BYT	BYT	BYT	BYT9~BYT1	BYT17
-----	-----	-----	-----------	-------

6	7	8	788	89
AA	01	02	XX	Chksu m

说明 1) **BYT6=AA**, 固定字节;

2) **BYT7=01**, 固定字节;

3) **BYT8=02**, 窗口数据初始化命令;

77) **BYTE9~BYTE2045**: 保留, 填充 0;

28) **BYT2046=Checksum/0xAA**: **BYT6~BYT2045** 校验码;

4、保存所有窗口数据 (03)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	00	XX	Chksu m

说明 1) **BYT6=AA**, 固定字节;

2) **BYT7=01**, 固定字节;

3) **BYT8=03**, 保存所有窗口数据命令;

4) **BYT9=00~0F**, 显示模式;

5) **BYT10~BYT21** 共 12 个字节为窗口 1 数据;

.) **BYT10**=窗口 1 ID;

.) **BYT11**=窗口 1 是否显示;

.) **BYT12**=窗口 1 信号源;

.) **BYT13**=窗口 1 叠加层次;

.) **BYT14**=窗口 1 水平起始低 8 位;

.) **BYT15**=窗口 1 水平起始高 8 位;

.) **BYT16**=窗口 1 宽度低 8 位;

.) **BYT17**=窗口 1 宽度高 8 位;

.) **BYT18**=窗口 1 垂直起始低 8 位;

.) **BYT19**=窗口 1 垂直起始高 8 位;

.) **BYT20**=窗口 1 高度高 8 位;

- .) **BYT21**=窗口 1 高度低 8 位;
- 6) **BYT22~BYT33**=窗口 2 数据;
- 7) **BYT33~BYT44**=窗口 3 数据;
-
- 20) **BYT766~BYT777**=窗口 64 数据;
- 21) **BYT778~BYTE2045**: 保留, 填充 0;
- 22) **BYT2046**= Checksum/0xAA: **BYT6~BYT2045** 校验码;

5、保存输出卡校正数据 (04)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	04	XX	Chksu m

- 说明 1) **BYT6**=AA, 固定字节;
- 2) **BYT7**=01, 固定字节;
 - 3) **BYT8**=04, 保存输出卡校正数据命令;
 - 4) **BYT9~BYT10** 共 2 个字节为 Out1 校正数据;
 - .) **BYT9**=水平校正;
 - .) **BYT10**=垂直校正;
 - 5) **BYTE11~BYTE12**=Out2 校正数据;
 - 6) **BYTE13~BYTE14**=Out3 校正数据;
 -
 - 20) **BYT39~BYT40**=Out16 校正;
 - 21) **BYT41~BYTE2045**: 保留, 填充 0;
 - 22) **BYT2046**= Checksum/0xAA: **BYT6~BYT2045** 校验码;

6、清除窗口 (05)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	05	XX	Chksu

				m
--	--	--	--	---

- 说明 1) **BYT6=AA**, 固定字节;
- 2) **BYT7=01**, 固定字节;
- 3) **BYT8=05**, 清除窗口命令;
- 4) **BYT9~BYT1677**: 输出卡协议, 详见输出卡协议文档;
- 5) **BYT1678~BYT2045**: 保留, 填充 0;
- 6) **BYT2046=Checksum/0xAA: BYT6~BYT2045** 校验码;

7、切换显示模式 (06)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	06	XX	Chksu m

- 说明 1) **BYT6=AA**, 固定字节;
- 2) **BYT7=01**, 固定字节;
- 3) **BYT8=05**, 切换显示模式命令;
- 4) **BYT9~BYT1677**: 输出卡协议, 详见输出卡协议文档;
- 5) **BYT1678~BYT2045**: 保留, 填充 0;
- 6) **BYT2046=Checksum/0xAA: BYT6~BYT2045** 校验码;

8、初始化 (07)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	07	XX	Chksu m

- 说明 1) **BYT6=AA**, 固定字节;
- 2) **BYT7=01**, 固定字节;
- 3) **BYT8=05**, 初始化命令;
- 4) **BYT9~BYT1677**: 输出卡协议, 详见输出卡协议文档;
- 5) **BYT1678~BYT2045**: 保留, 填充 0;

6) **BYT2046**= Checksum/0xAA: **BYT6~BYT2045** 校验码;

9、保存 LED 构建数据+所有窗口命令 (08)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	08	XX	Chksu m

说明 1) **BYT6**=AA, 固定字节;

2) **BYT7**=01, 固定字节;

3) **BYT8**=08, 保存 LED 构建数据+所有窗口命令命令;

4) **BYT9**=00~0F, 显示模式;

BYT10~BYT141 共 132 个字节为 LED 构建数据;

BYT10~BYT13 共 4 个字节为 LED 总尺寸数据;

.) **BYT10**=LED 水平总宽度低 8 位;

.) **BYT11**=LED 水平总宽度高 8 位;

.) **BYT12**=LED 水平总高度低 8 位;

.) **BYT13**=LED 水平总高度高 8 位;

BYT14~BYT21 共 8 个字节为 LED1 尺寸数据;

.) **BYT14**=LED1 水平起始低 8 位;

.) **BYT15**= LED1 水平起始高 8 位;

.) **BYT16**= LED1 宽度低 8 位;

.) **BYT17**= LED1 宽度高 8 位;

.) **BYT18**= LED1 垂直起始低 8 位;

.) **BYT19**= LED1 垂直起始高 8 位;

.) **BYT20**= LED1 高度低 8 位;

.) **BYT21**= LED1 高度高 8 位;

BYT22~BYT29 共 8 个字节为 LED2 尺寸数据;

BYT30~BYT37 共 8 个字节为 LED3 尺寸数据;

.....

BYT134~BYT141 共 8 个字节为 LED16 尺寸数据;

BYT142~BYT21 共 12 个字节为窗口数据；

BYT142~BYT153 共 12 个字节为窗口 1 数据；

- .) **BYT142**=窗口 1 ID；
- .) **BYT143**=窗口 1 是否显示；
- .) **BYT144**=窗口 1 信号源；
- .) **BYT145**=窗口 1 叠加层次；
- .) **BYT146**=窗口 1 水平起始低 8 位；
- .) **BYT147**=窗口 1 水平起始高 8 位；
- .) **BYT148**=窗口 1 宽度低 8 位；
- .) **BYT149**=窗口 1 宽度高 8 位；
- .) **BYT150**=窗口 1 垂直起始低 8 位；
- .) **BYT151**=窗口 1 垂直起始高 8 位；
- .) **BYT152**=窗口 1 高度高 8 位；
- .) **BYT153**=窗口 1 高度低 8 位；

BYT154~BYT165 共 12 个字节为窗口 2 数据；

BYT166~BYT176 共 12 个字节为窗口 3 数据；

.....

BYT898~BYT909 共 12 个字节为窗口 64 数据；

BYT910~BYT2045: 保留，填充 0；

BYT2046= Checksum/0xAA: **BYT6~BYT2045** 校验码；

*该命令可以在改变 PC 窗口整体大小时使用；

10、导入输出卡+LED 构建数据（从控制端加载数据至设备）（09）

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	09	XX	Chksu m

说明 1) **BYT6**=AA，固定字节；

- 2) **BYT7=01**, 固定字节;
- 3) **BYT8=09**, **导入输出卡+LED 构建数据**命令;
- 4) **BYT9=00~0F**, 显示模式;

BYT10~ BYT609 共 600 个字节, 为输出卡的数据;

BYT10~ BYT84 共 75 个字节, 为 Out-1 的数据;

- .) **BYT10 = 0~7**: 输出卡号;
- .) **BYT11= 0**: 该输入卡无插卡;
 = 1: 该输入卡有插卡;
- .) **BYT12=00**: 应用模式, 暂时保留;
- .) **BYT13=0~15**: 显示模式;
- .) **BYT14=0~100**: 亮度;
- .) **BYT15=BIT3~BIT0**, IMG4~IMG1 通道开启/关闭, BIT=0, IMG 关闭, BIT=1, IMG 开启;
- .) **BYT16=0~15**: IMG1 对应的输入卡 In1~In16;
- .) **BYT17=0~15**: IMG2 对应的输入卡 In1~In16;
- .) **BYT18=0~15**: IMG3 对应的输入卡 In1~In16;
- .) **BYT19=0~15**: IMG4 对应的输入卡 In1~In16;
- .) **BYT20=0~23**: overlay 值;
- .) **BYT21= IMG1** 输入水平起始高 8 位;
- .) **BYT22= IMG1** 输入水平起始低 8 位;
- .) **BYT23= IMG1** 输入宽度高 8 位;
- .) **BYT24= IMG1** 输入宽度低 8 位;
- .) **BYT25= IMG1** 输入垂直起始高 8 位;
- .) **BYT26= IMG1** 输入垂直起始低 8 位;
- .) **BYT27= IMG1** 输入高度高 8 位;
- .) **BYT28= IMG1** 输入高度低 8 位;
- .) **BYT29= IMG1** 输出水平起始高 8 位;
- .) **BYT30= IMG1** 输出水平起始低 8 位;
- .) **BYT31= IMG1** 输出宽度高 8 位;
- .) **BYT32= IMG1** 输出宽度低 8 位;
- .) **BYT33= IMG1** 输出垂直起始高 8 位;

- .) **BYT34**= IMG1 输出垂直起始低 8 位;
- .) **BYT35**= IMG1 输出高度高 8 位;
- .) **BYT36**= IMG1 输出高度低 8 位;
- .) **BYT37~BYT52**=IMG2 输入输出大小位置数据;
- .) **BYT53~BYT68**=IMG3 输入输出大小位置数据;
- .) **BYT69~BYT84**=IMG4 输入输出大小位置数据;
- BYT85~BYT159** 共 75 个字节, 为 Out-2 的数据;
- BYT160~BYT234** 共 75 个字节, 为 Out-3 的数据;
-
- BYT535~BYT609** 共 75 个字节, 为 Out-8 的数据;

BYT610~ BYT741 共 132 个字节, 为 LED 构建数据;

- .) **BYT610**=LED 水平总宽度低 8 位;
- .) **BYT611**=LED 水平总宽度高 8 位;
- .) **BYT612**=LED 水平总高度低 8 位;
- .) **BYT613**=LED 水平总高度高 8 位;
- BYT614~ BYT621** 共 8 个字节, 为 LED1 的数据;
- .) **BYT614**=LED1 水平起始低 8 位;
- .) **BYT615**= LED1 水平起始高 8 位;
- .) **BYT616**= LED1 宽度低 8 位;
- .) **BYT617**= LED1 宽度高 8 位;
- .) **BYT618**= LED1 垂直起始低 8 位;
- .) **BYT619**= LED1 垂直起始高 8 位;
- .) **BYT620**= LED1 高度低 8 位;
- .) **BYT621**= LED1 高度高 8 位;
- .) **BYT622~ BYT629**= LED2 尺寸数据;
- .) **BYT630~ BYT637**= LED3 尺寸数据;
-
- .) **BYT734~BYT741**= LED16 尺寸数据;

BYT742~BYT2045: 保留, 填充 0;

BYT2046=Checksum/0xAA: BYT6~BYT2045 校验码;

*该命令在从控制端加载数据至设备时使用;

11、导入窗口数据（从控制端加载数据至设备）（0A）

BYT 6	BYT 7	BYT 8	BYT9~BYT1 788	BYT17 89
AA	01	0A	XX	Chksu m

说明 1) **BYT6=AA**, 固定字节;

2) **BYT7=01**, 固定字节;

3) **BYT8=0A**, **导入窗口数据**命令;

4) **BYT9=00~0F**, 显示模式;

BYT10~BYT1801 共 1792 个字节, 为窗口基本数据;

BYT10~BYT37 共 28 个字节, 为 Win1 的数据;

.) **BYT10=窗口 ID**;

.) **BYT11=窗口是否显示**;

.) **BYT12=窗口信号源**;

.) **BYT13=窗口叠加层次**;

.) **BYT14~BYT29=该窗口在 LED1~LED16 中占用的 CH**;

.) **BYT30= 虚拟窗口水平起始低 8 位**;

.) **BYT31= 虚拟窗口水平起始高 8 位**;

.) **BYT32= 虚拟窗口宽度低 8 位**;

.) **BYT33=虚拟窗口宽度高 8 位**;

.) **BYT34=虚拟窗口垂直起始低 8 位**;

.) **BYT35=虚拟窗口垂直起始高 8 位**;

.) **BYT36=虚拟窗口高度高 8 位**;

.) **BYT37=虚拟窗口高度低 8 位**;

BYT38~BYT65 共 28 个字节, 为 Win2 的数据;

BYT66~BYT93 共 28 个字节, 为 Win3 的数据;

.....

BYT1774~BYT1801 共 28 个字节，为 Win64 的数据；

BYT1802~BYT1865 共 64 个字节，为窗口 overlay 数据；

BYT1866~BYT1867 共 2 个字节，为 LED 水平和垂直数目；

.) **BYT1866**=LED 水平个数；

.) **BYT1867**=LED 垂直个数；

BYT1868~BYT2045: 保留，填充 0；

BYT2046=Checksum/0xAA: **BYT6~BYT2045** 校验码；

*该命令在从控制端加载数据至设备时使用；

12、数据加载（从系统控制卡加载数据至输出卡）(0B)

BYT 6	BYT 7	BYT 8	BYT9~BYT11 788	BYT17 89
AA	01	0B	XX	Chksu m

说明 1) **BYT6**=AA，固定字节；

2) **BYT7**=01，固定字节；

3) **BYT8**=0B，**数据加载**命令；

4) **BYT9**=BIT7~BIT0，表示加载模式 8~模式 1；

5) **BYT10**= BIT7~BIT0，表示加载模式 16~模式 9；

5) **BYT11**= BIT7~BIT0，表示加载输出卡 8~输出卡 1；

BYT10~BYT2045: 保留，填充 0；

BYT2046=Checksum/0xAA: **BYT6~BYT2045** 校验码；

*该命令在从系统控制卡加载数据至输出卡时使用，使用时，一般先使用“导入输出卡+LED 构建数据”和“导入窗口数据”把所有数据先加载进系统控制卡，再调用该指令；

13、设置输出卡应用模式 (0C)

BYT 6	BYT 7	BYT 8	BYT9~BY T10	BYT1 1
AA	01	0C	XX	Chksu m

- 说明 .) **BYT0** = 20, VF2000 设备类型编号;
- .) **BYT1** = 01, 设备编号;
- .) **BYT2** = 00, 13 个字节长度命令高位;
- .) **BYT3** = 0D, 13 个字节长度命令低位;
- .) **BYT4** = 03, 固定字节;
- .) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;
= 0x00: 表示受控设备不返任何数据;
- .) **BYT6** = AA, 固定字节;
- .) **BYT7** = 01, 固定字节;
- .) **BYT8** = 0C, 设置输出卡应用模式;
- .) **BYT9** = 0~7, 表示要设置的板卡 Out1~Out8;
= 0xff, 表示设置所有板卡;
- .) **BYT10** = 0, 设置板卡应用模式为 4 画面;
= 1, 设置板卡应用模式为 2+2 画面;
- .) **BYT11** =Checksum/0xAA: **BYT6~BYT10** 校验码;
- .) **BYT12** =Checksum/0x20: **BYT0~BYT11** 校验码;
- *该命令在更改应用模式时使用;

13、设备 LAN 设置 (0E)

BYT 6	BYT 7	BYT 8	BYT9~BY T10	BYT1 1
AA	01	0C	XX	Chksu m

- 说明 .) **BYT0** = 20, VF2000 设备类型编号;
- .) **BYT1** = 01, 设备编号;
- .) **BYT2** = 00, 30 个字节长度命令高位;

- .) **BYT3** = 1E, 30 个字节长度命令低位;
- .) **BYT4** = 03, 固定字节;
- .) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;
= 0x00: 表示受控设备不返任何数据;
- .) **BYT6** = AA, 固定字节;
- .) **BYT7** = 01, 固定字节;
- .) **BYT8** = 0E, 设备 LAN 设置;
- .) **BYT9** = 0, IP 地址设置不生效;
= 1, IP 地址设置生效;
- .) **BYT10** = 0~255, IP 地址 a.b.c.d 第一位 a;
- .) **BYT11** = 0~255, IP 地址 a.b.c.d 第二位 b;
- .) **BYT12** = 0~255, IP 地址 a.b.c.d 第三位 c;
- .) **BYT13** = 0~255, IP 地址 a.b.c.d 第四位 d;
- .) **BYT14** = 0, 子网掩码设置不生效;
= 1, 子网掩码设置生效;
- .) **BYT15** = 0~255, 子网掩码 a.b.c.d 第一位 a;
- .) **BYT16** = 0~255, 子网掩码 a.b.c.d 第二位 b;
- .) **BYT17** = 0~255, 子网掩码 a.b.c.d 第三位 c;
- .) **BYT18** = 0~255, 子网掩码 a.b.c.d 第四位 d;
- .) **BYT19** = 0, 网关设置不生效;
= 1, 网关设置生效;
- .) **BYT20** = 0~255, 网关 a.b.c.d 第一位 a;
- .) **BYT21** = 0~255, 网关 a.b.c.d 第二位 b;
- .) **BYT22** = 0~255, 网关 a.b.c.d 第三位 c;
- .) **BYT23** = 0~255, 网关 a.b.c.d 第四位 d;
- .) **BYT24** = 0, MAC 地址设置不生效;
= 1, MAC 地址设置生效;
- .) **BYT25** = 0x00~0xFF, MAC 地址 a-b-c-d-e-f 第四位 d;
- .) **BYT26** = 0x00~0xFF, MAC 地址 a-b-c-d-e-f 第四位 e;
- .) **BYT27** = 0x00~0xFF, MAC 地址 a-b-c-d-e-f 第四位 f;

.) **BYT28** =Checksum/0xAA: **BYT6~BYT27** 校验码;

.) **BYT29** =Checksum/0x20: **BYT0~BYT28** 校验码;

*该命令在设置 LAN 时使用，设置完成后，请重新启动机器，方能生效;

14、执行输出卡协议并保存窗口数据 **NEW** (0F)

BYT	BYT	BYT	BYT9~BYT1	BYT17
6	7	8	788	89
AA	01	01	XX	Chksu m

说明 .) **BYT0** = 20, VF2000 设备类型编号;

.) **BYT1** = 01, 设备编号;

.) **BYT2** = 06, 1772 个字节长度命令高位;

.) **BYT3** = EC, 1772 个字节长度命令低位;

.) **BYT4** = 03, 固定字节;

.) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

.) **BYT6** = AA, 固定字节;

.) **BYT7** = 01, 固定字节;

.) **BYT8** = 0F, 执行输出卡协议并保存窗口数据 **NEW**;

BYT9~BYT1677: 共 1669 个字节, 为输出卡协议, 详见输出卡协议文档;

BYT1678~BYT1705: 共 28 个字节, 为当前窗口数据;

.) **BYT1678** =窗口 ID;

.) **BYT1679** =窗口是否显示;

.) **BYT1680** =窗口信号源;

.) **BYT1681** =保留;

.) **BYT1682~BYT1697**

= 该窗口在 LED1~LED16 中占用的 CH (16BYTE);

.) **BYT1698** =虚拟窗口水平起始低 8 位;

.) **BYT1699** = 虚拟窗口水平起始高 8 位;

.) **BYT1700** = 虚拟窗口宽度低 8 位;

- .) **BYT1701** =虚拟窗口宽度高 8 位；
- .) **BYT1702** =虚拟窗口垂直起始低 8 位；
- .) **BYT1703** = 虚拟窗口垂直起始高 8 位；
- .) **BYT1704** = 虚拟窗口高度低 8 位；
- .) **BYT1705** =虚拟窗口高度高 8 位；

BYT1706~BYT1769: 共 64 个字节，为窗口 1~窗口 64 的 Overlay；

- .) **BYT1770** =Checksum/0xAA: **BYT6~BYT1769** 校验码；
- .) **BYT1771** =Checksum/0x20: **BYT0~BYT1770** 校验码；

*该命令在新建窗口/改变窗口位置大小/改变窗口叠加层次等操作时使用；

15、系统初始化 (10)

BYT 6	BYT 7	BYT 8	BYT9~BYT1 788	BYT17 89
AA	01	07	XX	Chksu m

- 说明 .) **BYT0** = 20, VF2000 设备类型编号；
- .) **BYT1** = 01, 设备编号；
 - .) **BYT2** = 00, 13 个字节长度命令高位；
 - .) **BYT3** = 0C, 13 个字节长度命令低位；
 - .) **BYT4** = 03, 固定字节；
 - .) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果；
= 0x00: 表示受控设备不返任何数据；
 - .) **BYT6** = AA, 固定字节；
 - .) **BYT7** = 01, 固定字节；
 - .) **BYT8** = 10, **系统初始化**；
 - .) **BYT9** = 0, 输出板卡初始化为 4 画面；
= 1, 输出板卡初始化为 2+2 画面；
 - .) **BYT10** =Checksum/0xAA: **BYT6~BYT9** 校验码；
 - .) **BYT11** =Checksum/0x20: **BYT0~BYT10** 校验码；

*该命令在初始化系统时使用；

16、测试输出卡命令 (AA)

BYT 6	BYT 7	BYT 8	BYT9~BY T10	BYT1 1
AA	01	0C	XX	Chksu m

说明 .) **BYT0** = 20, VF2000 设备类型编号;

.) **BYT1** = 01, 设备编号;

.) **BYT2** = 06, 1680 个字节长度命令高位;

.) **BYT3** = 90, 1680 个字节长度命令低位;

.) **BYT4** = 03, 固定字节;

.) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

.) **BYT6** = AA, 固定字节;

.) **BYT7** = 01, 固定字节;

.) **BYT8** = AA, 设置输出卡应用模式;

.) **BYT9~BYT1677**: 共 1669 个字节, 为输出卡协议, 详见输出卡协议文档;

.) **BYT1678** =Checksum/0xAA: **BYT6~BYT1677** 校验码;

.) **BYT1679** =Checksum/0x20: **BYT0~BYT1678** 校验码;

*该命令在测试输出卡时应用;

七、切换台控制命令

BYT0 = 0x20: VF2000 设备类型号;

BYT1 = 设备编号: 0x00~0xFE;

BYT2 = : n 个字节长度命令高位;

BYT3 = : n 个字节长度命令低位;

BYT4 = 0x04: 表示切换台控制命令;

BYT5 = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

BYT6~BYTn-2: 协议数据;

BYTn-1 = Checksum/0x20: BYT0~BYTn-2 校验码;

命令详解:

1、测试切换台输出卡基本命令 (AA)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T10	1
AA	01	0C	XX	Chksu m

说明 .) **BYT0** = 20, VF2000 设备类型编号;

.) **BYT1** = 01, 设备编号;

.) **BYT2** = 06, 1680 个字节长度命令高位;

.) **BYT3** = 90, 1680 个字节长度命令低位;

.) **BYT4** = 04, 固定字节;

.) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;

= 0x00: 表示受控设备不返任何数据;

.) **BYT6** = AA, 固定字节;

.) **BYT7** = 01, 固定字节;

.) **BYT8** = AA, **测试切换台输出卡基本命令;**

.) **BYT9~BYT1677**: 共 1669 个字节, 为输出卡协议, 详见输出卡协议文档;

.) **BYT1678** =Checksum/0xAA: **BYT6~BYT1677** 校验码;

.) **BYT1679** =Checksum/0x20: **BYT0~BYT1678** 校验码;

*该命令在测试输出卡时应用;

2、测试切换台输出卡预切换命令 (AB)

BYT	BYT	BYT	BYT9~BY	BYT1
6	7	8	T10	1
AA	01	0C	XX	Chksu m

说明 .) **BYT0** = 20, VF2000 设备类型编号;

- .) **BYT1** = 01, 设备编号;
 - .) **BYT2** = 06, 1680 个字节长度命令高位;
 - .) **BYT3** = 90, 1680 个字节长度命令低位;
 - .) **BYT4** = 04, 固定字节;
 - .) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;
= 0x00: 表示受控设备不返任何数据;
 - .) **BYT6** = AA, 固定字节;
 - .) **BYT7** = 01, 固定字节;
 - .) **BYT8** = AB, 测试切换台输出卡预切换命令;
 - .) **BYT9~BYT1677**: 共 1669 个字节, 为输出卡协议, 详见输出卡协议文档;
 - .) **BYT1678** =Checksum/0xAA: **BYT6~BYT1677** 校验码;
 - .) **BYT1679** =Checksum/0x20: **BYT0~BYT1678** 校验码;
- *该命令在测试输出卡时应用;

3、测试切换台输出卡换命令 (AC)

BYT 6	BYT 7	BYT 8	BYT9~BY T10	BYT1 1
AA	01	0C	XX	Chksu m

- 说明 .) **BYT0** = 20, VF2000 设备类型编号;
- .) **BYT1** = 01, 设备编号;
 - .) **BYT2** = 06, 1680 个字节长度命令高位;
 - .) **BYT3** = 90, 1680 个字节长度命令低位;
 - .) **BYT4** = 04, 固定字节;
 - .) **BYT5** = 0x01: 表示受控设备在收到该命令并完成操作后返回命令执行结果;
= 0x00: 表示受控设备不返任何数据;
 - .) **BYT6** = AA, 固定字节;
 - .) **BYT7** = 01, 固定字节;
 - .) **BYT8** = AC, 测试切换台输出卡换命令;
 - .) **BYT9~BYT1677**: 共 1669 个字节, 为输出卡协议, 详见输出卡协议文档;

.) **BYT1678** =Checksum/0xAA: **BYT6~BYT1677** 校验码;

.) **BYT1679** =Checksum/0x20: **BYT0~BYT1678** 校验码;

*该命令在测试输出卡时应用;

八、命令处理结果返回值详解

当 **BYT5=0x01** 时, 系统会返回一串数据表示命令处理结果;

每条命令处理结果由 8 个字节的字符串组成, 定义为:

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT
0	1	2	3	4	5	6	7

BYT0: 设备类型号, 对于 **VF2000**, **BYT0=0x20**;

BYT1: 受控设备的编号, 范围从 01~FF 共 255 个编号;

BYT2: 0xFF, 固定字节;

BYT3: 命令目标: 0-输入卡/1-输出卡/2-底板;

BYT4: 命令处理结果;

BYT5: 参数 1;

BYT6: 参数 2;

BYT7: 前 **BYT0~BYT6** 个字节数据的 异或 校验值 **ChkSum**;

BYTE4~BYTE6 详解:

1、命令处理成功 (00)

BYT	BYT	BYT
4	5	6
00	00	00

说明 1) **BYT4=00**, 命令处理成功;

2) **BYT5~BYT6**, 保留, 无实际意义;

*****未完待续*****

九、读取命令

BYT0 = 0x20: VF2000 设备类型号;

BYT1 = 设备编号: 0x01~0xFF;

BYT2 = 0x00: 16 个字节长度命令高位;

BYT3 = 0x10: 16 个字节长度命令低位;

BYT4 = 保留;

BYT5 = 0xFE: 表示读取命令;

BYT6~BYT14: 9 个字节的读取命令;

BYT6 = 0x01: 读取底板;

= 0x69: 读取输出卡;

= 0x86: 读取输入卡;

BYT7 = 0x01: 卡号, 暂时保留, 目前读取所有卡;

BYT8 = 0x00: 命令, 目前保留;

BYT9~BYT13: 参数, 详见下面 **BYT6~BYT14** 详解;;

BYT14 = Chksum, **BYTE6~BYTE13** 异或运算或者 **BYT6**;

BYT15 = Checksum/0x20: **BYT0~BYT15** 校验码;

返回:

BYT0 = 0x20: VF2000 设备类型号;

BYT1 = 设备编号: 0x01~0xFF;

BYT2 = n+1 个字节长度命令高位;

BYT3 = n+1 个字节长度命令低位;

BYT4 = 保留;

BYT5 = 0xFE: 表示读取命令;

BYT6~BYT_{n-1}: n-6 个字节的数据返回;

BYT6 = 0x01: 返回底板数据;

= 0x69: 返回输出卡数据;

= 0x86: 返回输入卡数据;

= 0xaa: 返回 Win 相关数据;

BYT7 = 0x01: 卡号, 暂时保留, 目前读取所有卡;

BYT8 = 0x00: 命令, 目前保留;

BYT9~ BYT_{n-1}: 返回的数据, 详见读取命令详解;

BYT_{n-1} = Chksum, **BYTE6~ BYT_{n-2}** 异或运算或者 **BYT6**;

BYT_n = Checksum/0x20: **BYT0~BYT_{n-1}** 校验码;

读取命令详解:

1、读取系统信息 (00)

BYT 6	BYT 7	BYT 8	BYT 9	BYT 10	BYT1 1	BYT 12	BYT 13	BYT 14
01	01	00	00	00	00	00	00	Chk

返回

BYT 0	BYT 1	BYT 2	BYT 3	BYT 4	BYT 5	BYT 6	BYT 7	BYT 8	BYT9~BY T18	BYT 19
20	01	00	13	02	FE	01	01	00	返回数据	Chk

- 1) **BYT0**=20, 固定字节;
- 2) **BYT1**=设备编号: 0x01~0xFF;
- 3) **BYT2**=0x00: **20** 个字节长度返回命令高位;
- 4) **BYT3**=0x14: **20** 个字节长度返回命令低位;
- 5) **BYT4**=00: 保留;
- 6) **BYT5**=FE: 表示读取命令;
- 7) **BYT6** = 0x01: 返回底板数据;
- 8) **BYT7** = 0x01: 卡号, 暂时保留, 目前读取所有卡;
- 9) **BYT8** = 0x00: 命令, 目前保留;
- 10) **BYT9**, 输入卡 In-16~In-9 是否插卡状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

11) **BYT10**, 输入卡 In-8~In-1 是否插卡状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

12) **BYT11**, 输出卡 Out-8~Out-1 是否插卡状态;

BIT7~BIT0: Out-8~Out-1 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

13) **BYT12**, 输入卡 In-16~In-9 主信号源 (PA) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

14) **BYT13**, 输入卡 In-8~In-1 主信号源 (PA) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

15) **BYT14**, 输入卡 In-16~In-9 子信号源 (PB) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

16) **BYT15**, 输入卡 In-8~In-1 子信号源 (PB) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

17) **BYT16**, 输入卡 In-16~In-9 系统 (BUSY) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示空闲, 对应比特位为 1 表示繁忙;

18) **BYT17**, 输入卡 In-8~In-1 系统 (BUSY) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示空闲, 对应比特位为 1 表示繁忙;

19) **BYT18=0~15**, 当前显示模式;

20) **BYT19=Chksum**, **BYTE6~BYTE18** 异或运算或者 0x01;

21) **BYT20=Chksum**, **BYTE0~BYTE19** 异或运算或者 0x20;

3、读取所有输出卡信息 (02)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
69	01	00	xx	xx	xx	xx	00	Chk

1) **BYT6** = 0x69: 读取输出卡数据;

2) **BYT7** = 0x01: 暂时保留;

3) **BYT8** = 0x00: 保留;

4) **BYT9**: 显示模式;

BIT7~BIT4: Out-1 显示模式;

BIT3~BIT0: Out-2 显示模式;

4) **BYT10**: 显示模式;

BIT7~BIT4: Out-3 显示模式;

BIT3~BIT0: Out-4 显示模式;

4) **BYT11**: 显示模式;

BIT7~BIT4: Out-5 显示模式;

BIT3~BIT0: Out-6 显示模式;

4) **BYT12**: 显示模式;

BIT7~BIT4: Out-7 显示模式;

BIT3~BIT0: Out-8 显示模式;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	69	01	00	返回数据	Chk

1) **BYT0**=20, 固定字节;

2) **BYT1**=设备编号: 0x01~0xFF;

3) **BYT2**=0x03: **811** 个字节长度返回命令高位;

4) **BYT3**=0x2B: **811** 个字节长度返回命令低位;

5) **BYT4**=00: 保留;

6) **BYT5**=FE: 表示读取命令;

7) **BYT6** = 0x69: 返回输入卡数据;

8) **BYT7** = 0x01: 卡号, 暂时保留, 目前读取所有卡;

9) **BYT8** = 0x00: 命令, 目前保留;

BYT9~ BYT108 共 100 个字节, 为 Out-1 的数据;

10) **BYT9** = 0~7: 输出卡号;

11) **BYT10** = 0: 该输入卡无插卡;

= 1: 该输入卡有插卡;

12) **BYT11**=00: 应用模式, 暂时保留;

13) **BYT12** =0~15: 显示模式;

- 14) **BYT13** =0~100: 亮度;
- 15) **BYT14** =BIT3~BIT0, IMG4~IMG1 通道开启/关闭, BIT=0, IMG 关闭, BIT=1, IMG 开启;
- 16) **BYT15** =0~15: IMG1 对应的输入卡 In1~In16;
- 17) **BYT16** =0~15: IMG2 对应的输入卡 In1~In16;
- 18) **BYT17** =0~15: IMG3 对应的输入卡 In1~In16;
- 19) **BYT18** =0~15: IMG4 对应的输入卡 In1~In16;
- 20) **BYT19** =0~23: overlay 值;
- 21) **BYT20**=IMG1 输入水平起始高 8 位;
- 22) **BYT21**=IMG1 输入水平起始低 8 位;
- 23) **BYT22**=IMG1 输入宽度高 8 位;
- 24) **BYT23**=IMG1 输入宽度低 8 位;
- 25) **BYT24**=IMG1 输入垂直起始高 8 位;
- 26) **BYT25**=IMG1 输入垂直起始低 8 位;
- 27) **BYT26**=IMG1 输入高度高 8 位;
- 28) **BYT27**=IMG1 输入高度低 8 位;
- 29) **BYT28**=IMG1 输出水平起始高 8 位;
- 30) **BYT29**=IMG1 输出水平起始低 8 位;
- 31) **BYT30**=IMG1 输出宽度高 8 位;
- 32) **BYT31**=IMG1 输出宽度低 8 位;
- 33) **BYT32**=IMG1 输出垂直起始高 8 位;
- 34) **BYT33**=IMG1 输出垂直起始低 8 位;
- 35) **BYT34**=IMG1 输出高度高 8 位;
- 36) **BYT35**=IMG1 输出高度低 8 位;
- 36) **BYT36~BYT51**=IMG2 输入输出大小位置数据;
- 52) **BYT52~BYT67**=IMG3 输入输出大小位置数据;
- 68) **BYT68~BYT83**=IMG4 输入输出大小位置数据;
- 69) **BYT84**=Out-1 水平偏移值;
- 70) **BYT85**=Out-1 水平垂直偏移值;
- 85) **BYT86~BYT108**=Out-1 保留数据;
- BYT109~ BYT208** 共 100 个字节, 为 Out-2 的数据;

BYT209~BYT308 共 100 个字节，为 Out-3 的数据；

BYT309~BYT408 共 100 个字节，为 Out-4 的数据；

BYT409~BYT508 共 100 个字节，为 Out-5 的数据；

BYT509~BYT608 共 100 个字节，为 Out-6 的数据；

BYT609~BYT708 共 100 个字节，为 Out-7 的数据；

BYT709~BYT808 共 100 个字节，为 Out-8 的数据；

810) **BYT809=Checksum**, **BYTE6~BYTE808** 异或运算或者 0x69;

811) **BYT810=Checksum**, **BYTE0~BYTE809** 异或运算或者 0x20;

4、读取 LED 构建数据 (00)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	00	xx	xx	xx	xx	00	Chk

1) **BYT6** = 0xaa: 读取输出卡数据;

2) **BYT7** = 0x01: 暂时保留;

3) **BYT8** = 0x00: 表示读取 LED 构建数据;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	69	01	00	返回数据	Chk

1) **BYT0**=20, 固定字节;

2) **BYT1**=保留;

3) **BYT2**=0x00: 145 个字节长度返回命令高位;

4) **BYT3**=0x91: 145 个字节长度返回命令低位;

5) **BYT4**=00: 保留;

6) **BYT5**=FE: 表示读取命令;

7) **BYT6** = 0xaa: 返回窗口数据;

8) **BYT7** = 0x01: 暂时保留, 目前读取所有卡;

9) **BYT8** = 0x00: 命令, 目前保留;

- 10) **BYT9**=LED 水平方向屏数 (HxV);
 - 11) **BYT10**=LED 垂直方向屏数 (HxV);
 - 12) **BYT11**=LED 水平总宽度低 8 位;
 - 13) **BYT12**=LED 水平总宽度高 8 位;
 - 14) **BYT13**=LED 水平总高度低 8 位;
 - 15) **BYT14**=LED 水平总高度高 8 位;
- BYT15~ BYT22** 共 8 个字节, 为 LED1 的数据;
- 16) **BYT15**=LED1 水平起始低 8 位;
 - 17) **BYT16**= LED1 水平起始高 8 位;
 - 18) **BYT17**= LED1 宽度低 8 位;
 - 19) **BYT18**= LED1 宽度高 8 位;
 - 20) **BYT19**= LED1 垂直起始低 8 位;
 - 17) **BYT20**= LED1 垂直起始高 8 位;
 - 18) **BYT21**= LED1 高度低 8 位;
 - 19) **BYT22**= LED1 高度高 8 位;
 - 20) **BYTE23~BYTE30**= LED2 尺寸数据;
 - 21) **BYTE31~BYTE38**= LED3 尺寸数据;
 -
 - 22) **BYTE135~BYTE142**= LED16 尺寸数据;
 - 23) **BYT143**= Checksum/0xAA: **BYT6~BYT142** 校验码;
 - 24) **BYT144**=Chksum, **BYT0~BYT143** 异或运算或者 0x20;

5、读取 Overlay 及窗口显示数据 (01)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	01	xx	xx	xx	xx	00	Chk

- 1) **BYT6** = 0xaa: 读取输出卡数据;
- 2) **BYT7** = 0x01: 暂时保留;
- 3) **BYT8** = 0x01: 表示读取 **Overlay** 及窗口显示数据;

4) **BYT9 = 0~15**: 表示显示模式;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	4B	00	FE	69	01	00	返回数据	Chk

- 1) **BYT0**=20, 固定字节;
- 2) **BYT1**=设备编号: 0x01~0xFF;
- 3) **BYT2**=0x00: **139** 个字节长度返回命令高位;
- 4) **BYT3**=0x8B: **139** 个字节长度返回命令低位;
- 5) **BYT4**=00: 保留;
- 6) **BYT5**=FE: 表示读取命令;
- 7) **BYT6** = 0xaa: 返回输入卡数据;
- 8) **BYT7** = 0x01: 暂时保留, 目前读取所有卡;
- 9) **BYT8** = 0x01: 命令, 目前保留;

BYT9~ BYT72 共 64 个字节, 为 Overlay 的数据;

- 10) **BYT9**=WIN1 的 Overlay 数据;
- 11) **BYT10**= WIN2 的 Overlay 数据;
- 12) **BYT11**= WIN3 的 Overlay 数据;
- 20) **BYT12~BYT72**= Win4~Win64 的 Overlay 数据;
- 21) **BYT73~BYT136**=Win1~Win64 的显示状态数据;
- 22) **BYT137**=Chksum, **BYT6~BYT136** 异或运算或者 0xaa;
- 23) **BYT138**=Chksum, **BYT0~BYT137** 异或运算或者 0x20;

6、读取 Win 数据 (02)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	02	xx	xx	xx	xx	00	Chk

- 1) **BYT6** = 0xaa: 读取输出卡数据;
- 2) **BYT7** = 0x01: 暂时保留;

- 3) **BYT8 = 0x02**: 表示读取 Win 数据;
- 4) **BYT9 = 0x00**: 表示读取 Win1~Win32 数据;
= 0x01: 表示读取 Win33~Win64 数据;

5) **BYT10 = 0~15**: 表示显示模式;

仅需读取对应的 win 便可;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

- 1) **BYT0=20**, 固定字节;
- 2) **BYT1=设备编号**: 0x01~0xFF;
- 3) **BYT2=0x04**: **1163** 个字节长度返回命令高位;
- 4) **BYT3=0x8B**: **1163** 个字节长度返回命令低位;
- 5) **BYT4=00**: 保留;
- 6) **BYT5=FE**: 表示读取命令;
- 7) **BYT6 = 0xaa**: 返回窗口数据;
- 8) **BYT7 = 0x01**: 暂时保留, 目前读取所有卡;
- 9) **BYT8 = 0x02**: 命令, 目前保留;

BYT9~ BYT92 共 36 个字节, 为 Win1 的数据;

- 5) **BYT9=窗口 ID**;
- 6) **BYT10=窗口是否显示**;
- 7) **BYT11=窗口信号源**;
- 8) **BYT12=窗口叠加层次**;
- 9) **BYT13~BYT28=该窗口在 LED1~LED16 中占用的 CH**;
- 12) **BYT29= 虚拟窗口水平起始低 8 位**;
- 13) **BYT30= 虚拟窗口水平起始高 8 位**;
- 10) **BYT31= 虚拟窗口宽度低 8 位**;
- 11) **BYT32 =虚拟窗口宽度高 8 位**;
- 12) **BYT33 =虚拟窗口垂直起始低 8 位**;

13) **BYT34** =虚拟窗口垂直起始高 8 位;

14) **BYT35** =虚拟窗口高度高 8 位;

15) **BYT36**=虚拟窗口高度低 8 位;

12) **BYT37**=保留;

13) **BYT38**=保留;

16) **BYT39**=保留;

17) **BYT40**=保留;

18) **BYT41**=保留;

19) **BYT42**=保留;

20) **BYT43**=保留;

21) **BYT44**=保留;

BYT45~BYT80 共 36 个字节, 为 Win2 的数据;

BYT81~BYT116 共 36 个字节, 为 Win3 的数据;

.....

BYTE1125~BYTE1160 共 36 个字节, 为 Win32 的数据;

22) **BYT1161**=Chksum, **BYTE6~BYTE2695** 异或运算或者 0xaa;

23) **BYT1162**=Chksum, **BYTE0~BYTE2696** 异或运算或者 0x20;

6、调用模式并返回模式相关数据 (03)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	03	xx	xx	xx	xx	00	Chk

1) **BYT6** = 0xaa: 读取输出卡数据;

2) **BYT7** = 0x01: 暂时保留;

3) **BYT8** = 0x03: 表示调用模式并返回模式相关数据;

4) **BYT9** = 0~15: 表示显示模式;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

- 1) **BYT0**=20, 固定字节;
- 2) **BYT1**=设备编号: 0x01~0xFF;
- 3) **BYT2**=0x0A: **2631** 个字节长度返回命令高位;
- 4) **BYT3**=0x47: **2631** 个字节长度返回命令低位;
- 5) **BYT4**=00: 保留;
- 6) **BYT5**=FE: 表示读取命令;
- 7) **BYT6** = 0xaa: 返回窗口数据;
- 8) **BYT7** = 0x01: 暂时保留, 目前读取所有卡;
- 9) **BYT8** = 0x02: 命令, 目前保留;

BYT9~ BYT608 共 600 个字节, 为输出卡的数据;

BYT9~ BYT83 共 75 个字节, 为 Out-1 的数据;

- .) **BYT9**= 0~7: 输出卡号;
- .) **BYT10** = 0: 该输入卡无插卡;
= 1: 该输入卡有插卡;
- .) **BYT11** =00: 应用模式, 暂时保留;
- .) **BYT12** =0~15: 显示模式;
- .) **BYT13** =0~100: 亮度;
- .) **BYT14** =BIT3~BIT0, IMG4~IMG1 通道开启/关闭, BIT=0, IMG 关闭, BIT=1, IMG 开启;
- .) **BYT15** =0~15: IMG1 对应的输入卡 In1~In16;
- .) **BYT16** =0~15: IMG2 对应的输入卡 In1~In16;
- .) **BYT17** =0~15: IMG3 对应的输入卡 In1~In16;
- .) **BYT18** =0~15: IMG4 对应的输入卡 In1~In16;
- .) **BYT19** =0~23: overlay 值;
- .) **BYT20** = IMG1 输入水平起始高 8 位;
- .) **BYT21**= IMG1 输入水平起始低 8 位;
- .) **BYT22**= IMG1 输入宽度高 8 位;
- .) **BYT23**= IMG1 输入宽度低 8 位;
- .) **BYT24**= IMG1 输入垂直起始高 8 位;
- .) **BYT25**= IMG1 输入垂直起始低 8 位;

- .) **BYT26**= IMG1 输入高度高 8 位;
- .) **BYT27**= IMG1 输入高度低 8 位;
- .) **BYT28**= IMG1 输出水平起始高 8 位;
- .) **BYT29**= IMG1 输出水平起始低 8 位;
- .) **BYT30**= IMG1 输出宽度高 8 位;
- .) **BYT31**= IMG1 输出宽度低 8 位;
- .) **BYT32**= IMG1 输出垂直起始高 8 位;
- .) **BYT33**= IMG1 输出垂直起始低 8 位;
- .) **BYT34**= IMG1 输出高度高 8 位;
- .) **BYT35**= IMG1 输出高度低 8 位;
- .) **BYT36~BYT51**=IMG2 输入输出大小位置数据;
- .) **BYT52~BYT67**=IMG3 输入输出大小位置数据;
- .) **BYT68~BYT83**=IMG4 输入输出大小位置数据;
- BYT84~BYT158** 共 75 个字节, 为 Out-2 的数据;
- BYT159~BYT233** 共 75 个字节, 为 Out-3 的数据;
-
- BYT534~BYT608** 共 75 个字节, 为 Out-8 的数据;

BYT609~ BYT740 共 132 个字节, 为 LED 构建数据;

- .) **BYT609**=LED 水平总宽度低 8 位;
- .) **BYT610**=LED 水平总宽度高 8 位;
- .) **BYT611**=LED 水平总高度低 8 位;
- .) **BYT612**=LED 水平总高度高 8 位;
- BYT613~ BYT620** 共 8 个字节, 为 LED1 的数据;
- .) **BYT621**=LED1 水平起始低 8 位;
- .) **BYT622**= LED1 水平起始高 8 位;
- .) **BYT623**= LED1 宽度低 8 位;
- .) **BYT624**= LED1 宽度高 8 位;
- .) **BYT625**= LED1 垂直起始低 8 位;
- .) **BYT626**= LED1 垂直起始高 8 位;

- .) **BYT627**= LED1 高度低 8 位;
- .) **BYT628**= LED1 高度高 8 位;
- .) **BYT629~BYT636**= LED2 尺寸数据;
- .) **BYT637~BYT644**= LED3 尺寸数据;
-
- .) **BYT733~BYT740**= LED16 尺寸数据;

BYT741~ BYT2532 共 1792 个字节, 为窗口基本数据;

BYT741~ BYT768 共 28 个字节, 为 Win1 的数据;

- .) **BYT741**=窗口 ID;
- .) **BYT742**=窗口是否显示;
- .) **BYT743**=窗口信号源;
- .) **BYT744**=窗口叠加层次;
- .) **BYT745~BYT760**=该窗口在 LED1~LED16 中占用的 CH;
- .) **BYT761**= 虚拟窗口水平起始低 8 位;
- .) **BYT762**= 虚拟窗口水平起始高 8 位;
- .) **BYT763**= 虚拟窗口宽度低 8 位;
- .) **BYT764**=虚拟窗口宽度高 8 位;
- .) **BYT765**=虚拟窗口垂直起始低 8 位;
- .) **BYT766**=虚拟窗口垂直起始高 8 位;
- .) **BYT767**=虚拟窗口高度高 8 位;
- .) **BYT768**=虚拟窗口高度低 8 位;

BYT769~BYT796 共 28 个字节, 为 Win2 的数据;

BYT797~BYT824 共 28 个字节, 为 Win3 的数据;

.....

BYT2505~BYT2532 共 28 个字节, 为 Win64 的数据;

BYT2533~BYT2596=窗口 1~窗口 64 的 Overlay;

BYT2597~BYT2628=Out1~Out8 的边框数据;

.) **BYT2597**, Out1 CH1 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT2598**, Out1 CH2 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT2599**, Out1 CH3 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT2600**, Out1 CH4 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

BYT2601~BYT2604 共 4 个字节, 为 Out2 边框数据;

BYT2605~BYT2608 共 4 个字节, 为 Out3 边框数据;

.....

BYT2625~BYT2628 共 4 个字节, 为 Out8 边框数据;

BYT2629=Chksum, **BYT6~BYT2628** 异或运算或者 0xaa;

BYT2630=Chksum, BYT0~BYT2629 异或运算或者 0x20;

*该命令可在调用模式时使用，调用并会读会调用的模式数据；

6、读取设备总数据 (04)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	04	xx	xx	xx	xx	00	Chk

- 1) **BYT6 = 0xaa**: 读取输出卡数据;
- 2) **BYT7 = 0x01**: 暂时保留;
- 3) **BYT8 = 0x04**: 表示**读取设备总数据**;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

- 1) **BYT0=20**, 固定字节;
- 2) **BYT1=设备编号**: 0x01~0xFF;
- 3) **BYT2=0x0E**: **3619** 个字节长度返回命令高位;
- 4) **BYT3=0x23**: **3619** 个字节长度返回命令低位;
- 5) **BYT4=00**: 保留;
- 6) **BYT5=FE**: 表示读取命令;
- 7) **BYT6 = 0xaa**: 返回窗口数据;
- 8) **BYT7 = 0x01**: 暂时保留, 目前读取所有卡;
- 9) **BYT8 = 0x02**: 命令, 目前保留;

BYT9~ BYT18 共 9 个字节, 为系统控制卡的数据;

.) **BYT9**, 输入卡 In-16~In-9 是否插卡状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

.) **BYT10**, 输入卡 In-8~In-1 是否插卡状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

.) **BYT11**, 输出卡 Out-8~Out-1 是否插卡状态;

BIT7~BIT0: Out-8~Out-1 状态, 对应比特位为 0 表示无插卡, 对应比特位为 1 表示有插卡;

.) **BYT12**, 输入卡 In-16~In-9 主信号源 (PA) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

.) **BYT13**, 输入卡 In-8~In-1 主信号源 (PA) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

.) **BYT14**, 输入卡 In-16~In-9 子信号源 (PB) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

.) **BYT15**, 输入卡 In-8~In-1 子信号源 (PB) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示无信号, 对应比特位为 1 表示有信号;

.) **BYT16**, 输入卡 In-16~In-9 系统 (BUSY) 状态;

BIT7~BIT0: In-16~In-9 状态, 对应比特位为 0 表示空闲, 对应比特位为 1 表示繁忙;

.) **BYT17**, 输入卡 In-8~In-1 系统 (BUSY) 状态;

BIT7~BIT0: In-8~In-1 状态, 对应比特位为 0 表示空闲, 对应比特位为 1 表示繁忙;

.) **BYT18=0~15**, 当前显示模式;

BYT19~BYT466 共 448 个字节, 为输入卡的数据;

BYT19~BYT46 共 28 个字节, 为 In-1 的数据;

.) **BYT19=0~15**: 输入卡号;

.) **BYT20=0**: 该输入卡无插卡;

= 1: 该输入卡有插卡;

.) **BYT21=00**: 主信号源为 SDI;

=01: 主信号源为 DVI;

=02: 主信号源为 VGA;

=03: 主信号源为 VIDEO;

.) **BYT22=00**: 子信号源为 SDI;

=01: 子号源为 DVI;

=02: 子号源为 VGA;

=03: 子号源为 VIDEO;

.) **BYT23=00**: 切换状态;

=01: PIP 准备状态;

=02: PIP 状态;

=03: TEXT 准备状态;

=03: TEXT 状态;

.) **BYT24**=0~100: 亮度;

.) **BYT25**=0~100: 对比度;

.) **BYT26**=0~100: 颜色;

.) **BYT27**=00: <字幕阈值;

=01: >字幕阈值;

.) **BYT28**=0~248, 步进 8: 字幕阈值红;

.) **BYT29**=0~252, 步进 4: 字幕阈值绿;

.) **BYT30**=0~248, 步进 8: 字幕阈值蓝;

.) **BYT31**=主信号源水平起始低 8 位;

.) **BYT32**=主信号源水平起始高 8 位;

.) **BYT33**=主信号源宽度低 8 位;

.) **BYT34**=主信号源宽度高 8 位;

.) **BYT35**=主信号源垂直起始低 8 位;

.) **BYT36**=主信号源垂直起始高 8 位;

.) **BYT37**=主信号源高度低 8 位;

.) **BYT38**=主信号源高度高 8 位;

.) **BYT39**=子信号源水平起始低 8 位;

.) **BYT40**=子信号源水平起始高 8 位;

.) **BYT41**=子信号源宽度低 8 位;

.) **BYT42**=子信号源宽度高 8 位;

.) **BYT43**=子信号源垂直起始低 8 位;

.) **BYT44**=子信号源垂直起始高 8 位;

.) **BYT45**=子信号源高度低 8 位;

.) **BYT46**=子信号源高度高 8 位;

BYT47~BYT74 共 28 个字节, 为 In-2 的数据;

BYT75~BYT102 共 28 个字节, 为 In-3 的数据;

.....

BYT439~BYT466 共 28 个字节，为 In-16 的数据；

BYT467~BYT1066 共 600 个字节，为输出卡的数据；

BYT467~BYT541 共 75 个字节，为 Out-1 的数据；

- .) **BYT467** = 0~7: 输出卡号；
- .) **BYT468** = 0: 该输入卡无插卡；
= 1: 该输入卡有插卡；
- .) **BYT469** =00: 应用模式，暂时保留；
- .) **BYT470** =0~15: 显示模式；
- .) **BYT471** =0~100: 亮度；
- .) **BYT472** =BIT3~BIT0, IMG4~IMG1 通道开启/关闭, BIT=0, IMG 关闭, BIT=1, IMG 开启；
- .) **BYT473** =0~15: IMG1 对应的输入卡 In1~In16；
- .) **BYT474** =0~15: IMG2 对应的输入卡 In1~In16；
- .) **BYT475** =0~15: IMG3 对应的输入卡 In1~In16；
- .) **BYT476** =0~15: IMG4 对应的输入卡 In1~In16；
- .) **BYT477** =0~23: overlay 值；
- .) **BYT478** = IMG1 输入水平起始低 8 位；
- .) **BYT479** = IMG1 输入水平起始高 8 位；
- .) **BYT480** = IMG1 输入宽度低 8 位；
- .) **BYT481** = IMG1 输入宽度高 8 位；
- .) **BYT482** = IMG1 输入垂直起始低 8 位；
- .) **BYT483** = IMG1 输入垂直起始高 8 位；
- .) **BYT484** = IMG1 输入高度低 8 位；
- .) **BYT485** = IMG1 输入高度高 8 位；
- .) **BYT486** = IMG1 输出水平起始低 8 位；
- .) **BYT487** = IMG1 输出水平起始高 8 位；
- .) **BYT488** = IMG1 输出宽度低 8 位；
- .) **BYT489** = IMG1 输出宽度高 8 位；
- .) **BYT490** = IMG1 输出垂直起始低 8 位；
- .) **BYT491** = IMG1 输出垂直起始高 8 位；

- .) **BYT492** = IMG1 输出高度低 8 位;
- .) **BYT493** = IMG1 输出高度高 8 位;
- .) **BYT494~BYT509**=IMG2 输入输出大小位置数据;
- .) **BYT510~BYT525**=IMG3 输入输出大小位置数据;
- .) **BYT526~BYT541**=IMG4 输入输出大小位置数据;
- BYT542~BYT616** 共 75 个字节, 为 Out-2 的数据;
- BYT617~BYT691** 共 75 个字节, 为 Out-3 的数据;
-
- BYT992~BYT1066** 共 75 个字节, 为 Out-8 的数据;

BYT1067~ BYT1198 共 132 个字节, 为 LED 构建数据;

- .) **BYT1067**=LED 水平总宽度低 8 位;
- .) **BYT1068**=LED 水平总宽度高 8 位;
- .) **BYT1069**=LED 水平总高度低 8 位;
- .) **BYT1070**=LED 水平总高度高 8 位;
- BYT1071~BYT1078** 共 8 个字节, 为 LED1 的数据;
- .) **BYT1071**=LED1 水平起始低 8 位;
- .) **BYT1072**= LED1 水平起始高 8 位;
- .) **BYT1073**= LED1 宽度低 8 位;
- .) **BYT1074**= LED1 宽度高 8 位;
- .) **BYT1075**= LED1 垂直起始低 8 位;
- .) **BYT1076**= LED1 垂直起始高 8 位;
- .) **BYT1077**= LED1 高度低 8 位;
- .) **BYT1078**= LED1 高度高 8 位;
- .) **BYT1079~BYT1086**= LED2 尺寸数据;
- .) **BYT1087~BYT1094**= LED3 尺寸数据;
-
- .) **BYT1191~BYT1198**= LED16 尺寸数据;

BYT1199~ BYT2990 共 1792 个字节, 为窗口基本数据;

BYT1199~BYT1226 共 28 个字节，为 Win1 的数据；

- .) **BYT1199**=窗口 ID；
- .) **BYT1200**=窗口是否显示；
- .) **BYT1201**=窗口信号源；
- .) **BYT1202**=窗口叠加层次；
- .) **BYT1203~BYT1218**=该窗口在 LED1~LED16 中占用的 CH；
- .) **BYT1219**= 虚拟窗口水平起始低 8 位；
- .) **BYT1220**= 虚拟窗口水平起始高 8 位；
- .) **BYT1221**= 虚拟窗口宽度低 8 位；
- .) **BYT1222** =虚拟窗口宽度高 8 位；
- .) **BYT1223** =虚拟窗口垂直起始低 8 位；
- .) **BYT1224** =虚拟窗口垂直起始高 8 位；
- .) **BYT1225** =虚拟窗口高度高 8 位；
- .) **BYT1226**=虚拟窗口高度低 8 位；

BYT1227~BYT1254 共 28 个字节，为 Win2 的数据；

BYT1255~BYT1282 共 28 个字节，为 Win3 的数据；

.....

BYT2963~BYT2990 共 28 个字节，为 Win64 的数据；

BYT2991~BYT3054=窗口 1~窗口 64 的 Overlay；

BYT3055~BYT3056 共 2 个字节，为 LED 水平和垂直数目；

- .) **BYT3055**=LED 水平个数；
- .) **BYT3056**=LED 垂直个数；

BYT3057~BYT3584 共 528 个字节，为输入卡 OSD 字符数据；

BYT3057~BYT3087 共 33 个字节，为 In1 OSD 字符数据；

- .) **BYT3057**=0，OSD 字符关闭；
=1，OSD 字符开启；
- .) **BYT3058~BYT3065**=SDI 信号源 OSD 字符 ASCII；

.) **BYT3066~BYT3073**=DVI 信号源 OSD 字符 ASCII;

.) **BYT3074~BYT3081**=VGA 信号源 OSD 字符 ASCII;

.) **BYT3082~BYT3089**=VIDEO 信号源 OSD 字符 ASCII;

BYT3090~BYT3122 共 33 个字节, 为 In2 OSD 字符数据;

.....

BYT3552~BYT3584 共 33 个字节, 为 In16 OSD 字符数据;

BYT3585~BYT3616 共 32 个字节, 为 Out1~Out8 的边框数据;

.) **BYT3585**, Out1 CH1 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT3586**, Out1 CH2 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT3587**, Out1 CH3 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

.) **BYT3588**, Out1 CH4 边框数据;

BIT7~BIT4: 保留, 无实际意义

BIT3: =0, CH 下边边框关闭; =1, CH 下边边框开启;

BIT2: =0, CH 上边边框关闭; =1, CH 上边边框开启;

BIT1: =0, CH 右边边框关闭; =1, CH 右边边框开启;

BIT0: =0, CH 左边边框关闭; =1, CH 左边边框开启;

BYT3589~BYT3592 共 4 个字节, 为 Out2 边框数据;

BYT3593~BYT3596 共 4 个字节, 为 Out3 边框数据;

.....

BYT3613~BYT3616 共 4 个字节, 为 Out8 边框数据;

BYT3617=Chksum, BYT6~BYT3616 异或运算或者 0xaa;

BYT3618=Chksum, BYTE0~BYT3617 异或运算或者 0x20;

*该命令可在开启软件进行连接时调用;

6、设备数据备份（系统控制卡->控制设备）（05）

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	05	xx	xx	xx	xx	00	Chk

- 1) **BYT6** = 0xaa: 读取输出卡数据;
- 2) **BYT7** = 0x01: 暂时保留;
- 3) **BYT8** = 0x05: 表示**设备数据备份**;
- 3) **BYT9** = 0~15: 表示对应的显示模式;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

- 1) **BYT0**=20, 固定字节;
- 2) **BYT1**=设备编号: 0x01~0xFF;
- 3) **BYT2**=0x09: **2537** 个字节长度返回命令高位;
- 4) **BYT3**=0xE9: **2537** 个字节长度返回命令低位;
- 5) **BYT4**=00: 保留;

- 6) **BYT5=FE**: 表示读取命令;
- 7) **BYT6 = 0xaa**: 返回窗口数据;
- 8) **BYT7 = 0x01**: 暂时保留, 目前读取所有卡;
- 9) **BYT8 = 0x02**: 命令, 目前保留;

BYT9~ BYT608 共 600 个字节, 为输出卡的数据;

BYT9~ BYT83 共 75 个字节, 为 Out-1 的数据;

- .) **BYT9=0~7**: 输出卡号;
- .) **BYT10= 0**: 该输入卡无插卡;
= 1: 该输入卡有插卡;
- .) **BYT11=00**: 应用模式, 暂时保留;
- .) **BYT12=0~15**: 显示模式;
- .) **BYT13=0~100**: 亮度;
- .) **BYT14=BIT3~BIT0**, IMG4~IMG1 通道开启/关闭, BIT=0, IMG 关闭, BIT=1, IMG 开启;
- .) **BYT15=0~15**: IMG1 对应的输入卡 In1~In16;
- .) **BYT16=0~15**: IMG2 对应的输入卡 In1~In16;
- .) **BYT17=0~15**: IMG3 对应的输入卡 In1~In16;
- .) **BYT18=0~15**: IMG4 对应的输入卡 In1~In16;
- .) **BYT19=0~23**: overlay 值;
- .) **BYT20= IMG1** 输入水平起始高 8 位;
- .) **BYT21= IMG1** 输入水平起始低 8 位;
- .) **BYT22= IMG1** 输入宽度高 8 位;
- .) **BYT23= IMG1** 输入宽度低 8 位;
- .) **BYT24= IMG1** 输入垂直起始高 8 位;
- .) **BYT25= IMG1** 输入垂直起始低 8 位;
- .) **BYT26= IMG1** 输入高度高 8 位;
- .) **BYT27= IMG1** 输入高度低 8 位;
- .) **BYT28= IMG1** 输出水平起始高 8 位;
- .) **BYT29= IMG1** 输出水平起始低 8 位;
- .) **BYT30= IMG1** 输出宽度高 8 位;

- .) **BYT31**= IMG1 输出宽度低 8 位;
- .) **BYT32**= IMG1 输出垂直起始高 8 位;
- .) **BYT33**= IMG1 输出垂直起始低 8 位;
- .) **BYT34**= IMG1 输出高度高 8 位;
- .) **BYT35**= IMG1 输出高度低 8 位;
- .) **BYT36~ BYT51**=IMG2 输入输出大小位置数据;
- .) **BYT52~BYT67**=IMG3 输入输出大小位置数据;
- .) **BYT68~BYT83**=IMG4 输入输出大小位置数据;
- BYT84~BYT158** 共 75 个字节, 为 Out-2 的数据;
- BYT159~BYT233** 共 75 个字节, 为 Out-3 的数据;
-
- BYT534~BYT608** 共 75 个字节, 为 Out-8 的数据;

BYT609~ BYT740 共 132 个字节, 为 LED 构建数据;

- .) **BYT609**=LED 水平总宽度低 8 位;
- .) **BYT610**=LED 水平总宽度高 8 位;
- .) **BYT611**=LED 水平总高度低 8 位;
- .) **BYT612**=LED 水平总高度高 8 位;
- BYT613~BYT620** 共 8 个字节, 为 LED1 的数据;
- .) **BYT613**=LED1 水平起始低 8 位;
- .) **BYT614**= LED1 水平起始高 8 位;
- .) **BYT615**= LED1 宽度低 8 位;
- .) **BYT616**= LED1 宽度高 8 位;
- .) **BYT617**= LED1 垂直起始低 8 位;
- .) **BYT618**= LED1 垂直起始高 8 位;
- .) **BYT619**= LED1 高度低 8 位;
- .) **BYT620**= LED1 高度高 8 位;
- .) **BYT621~ BYT628**= LED2 尺寸数据;
- .) **BYT629~ BYT636**= LED3 尺寸数据;
-

.) **BYT733~BYT740**= LED16 尺寸数据;

BYT741~ BYT2532 共 1792 个字节, 为窗口基本数据;

BYT741~ BYT768 共 28 个字节, 为 Win1 的数据;

.) **BYT741**=窗口 ID;

.) **BYT742**=窗口是否显示;

.) **BYT743**=窗口信号源;

.) **BYT744**=窗口叠加层次;

.) **BYT745~BYT760**=该窗口在 LED1~LED16 中占用的 CH;

.) **BYT761**= 虚拟窗口水平起始低 8 位;

.) **BYT762**= 虚拟窗口水平起始高 8 位;

.) **BYT763**= 虚拟窗口宽度低 8 位;

.) **BYT764**=虚拟窗口宽度高 8 位;

.) **BYT765**=虚拟窗口垂直起始低 8 位;

.) **BYT766**=虚拟窗口垂直起始高 8 位;

.) **BYT767**=虚拟窗口高度高 8 位;

.) **BYT778**=虚拟窗口高度低 8 位;

BYT769~BYT796 共 28 个字节, 为 Win2 的数据;

BYT797~BYT824 共 28 个字节, 为 Win3 的数据;

.....

BYT2505~BYT2532 共 28 个字节, 为 Win64 的数据;

BYT2533~BYT1896 共 64 个字节, 为窗口 Overlay 数据;

BYT2597~BYT2598 共 2 个字节, 为 LED 水平和垂直数目;

.) **BYT2597**=LED 水平个数;

.) **BYT2598**=LED 垂直个数;

BYT2599=Chksum, **BYT6~BYT2598** 异或运算或者 0xaa;

BYT2600=Chksum, **BYTE0~BYT2599** 异或运算或者 0x20;

*该命令可在模式导出时调用；

7、读取软件版本号（06）

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	06	xx	xx	xx	xx	00	Chk

- 1) **BYT6** = 0xaa: 固定参数；
- 2) **BYT7** = 0x01: 固定参数；
- 3) **BYT8** = 0x05: 表示**读取软件版本号**；

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

- 1) **BYT0**=20, 固定字节；
- 2) **BYT1**=设备编号: 0x01~0xFF；
- 3) **BYT2**=0x00: **78** 个字节长度返回命令高位；
- 4) **BYT3**=0x4E: **78** 个字节长度返回命令低位；
- 5) **BYT4**=00: 保留；
- 6) **BYT5**=FE: 表示读取命令；
- 7) **BYT6** = 0xaa: 返回窗口数据；
- 8) **BYT7** = 0x01: 暂时保留, 目前读取所有卡；
- 9) **BYT8** = 0x02: 命令, 目前保留；

BYT9~BYT11 共 3 个字节, 为控制卡的版本号；

- .) **BYT9**=版本号第一位 Va.b.c 中的 a, 返回 0xFF 表示通信错误；
- .) **BYT10**=版本号第二位 Va.b.c 中的 b, 返回 0xFF 表示通信错误；
- .) **BYT11**=版本号第三位 Va.b.c 中的 c, 返回 0xFF 表示通信错误；

BYT12~BYT59 共 48 个字节, 为输入卡的版本号；

BYT12~BYT14 共 3 个字节, 为 In1 的版本号；

- .) **BYT12**=版本号第一位 Va.b.c 中的 a, 返回 0xFF 表示通信错误；

.) **BYT13**=版本号第二位 Va.b.c 中的 b, 返回 0xFF 表示通信错误;

.) **BYT14**=版本号第三位 Va.b.c 中的 c, 返回 0xFF 表示通信错误;

BYT15~BYT17 共 3 个字节, 为 In2 的版本号;

.....

BYT57~BYT59 共 3 个字节, 为 In16 的版本号;

BYT60~BYT75 共 16 个字节, 为输出卡版本号;

BYT60~BYT61 共 2 个字节, 为 Out1 的版本号;

.) **BYT60**=版本号第一位 Va.b 中的 a, 返回 0xFF 表示通信错误;

.) **BYT61**=版本号第二位 Va.b 中的 b, 返回 0xFF 表示通信错误;

BYT62~BYT63 共 2 个字节, 为 Out2 的版本号;

.....

BYT74~BYT75 共 2 个字节, 为 Out8 的版本号;

BYT76=Chksum, **BYT6~BYT75** 异或运算或者 0xaa;

BYT77=Chksum, **BYT0~BYT76** 异或运算或者 0x20;

*该命令可在读取版本号或者测试各板卡通信时使用;

7、读取 LAN 设置 (07)

BYT	BYT	BYT	BYT	BYT	BYT1	BYT	BYT	BYT
6	7	8	9	10	1	12	13	14
aa	01	07	xx	xx	xx	xx	00	Chk

1) **BYT6** = 0xaa: 固定参数;

2) **BYT7** = 0x01: 固定参数;

3) **BYT8** = 0x05: 表示**读取 LAN 设置**;

返回

BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT	BYT9~BY	BYT
0	1	2	3	4	5	6	7	8	T27	19
20	01	00	10	00	FE	aa	01	00	返回数据	Chk

1) **BYT0** = 20, 固定字节;

- 2) **BYT1** = 设备编号: 0x01~0xFF;
- 3) **BYT2** = 0x00: 30 个字节长度返回命令高位;
- 4) **BYT3** = 0x1E: 30 个字节长度返回命令低位;
- 5) **BYT4** = 00: 保留;
- 6) **BYT5** = FE: 表示读取命令;
- 7) **BYT6** = 0xaa: 返回窗口数据;
- 8) **BYT7** = 0x01: 暂时保留, 目前读取所有卡;
- 9) **BYT8** = 0x02: 命令, 目前保留;

BYT9~BYT11 共 4 个字节, 为 IP 地址:

- .) **BYT9** = IP 地址 a.b.c.d, 第一位 a;
- .) **BYT10** = IP 地址 a.b.c.d, 第二位 b;
- .) **BYT11** = IP 地址 a.b.c.d, 第三位 c;
- .) **BYT12** = IP 地址 a.b.c.d, 第四位 d;

BYT13~BYT16 共 4 个字节, 为子网掩码:

- .) **BYT13** = 子网掩码 a.b.c.d, 第一位 a;
- .) **BYT14** = 子网掩码 a.b.c.d, 第二位 b;
- .) **BYT15** = 子网掩码 a.b.c.d, 第三位 c;
- .) **BYT16** = 子网掩码 a.b.c.d, 第四位 d;

BYT17~BYT20 共 4 个字节, 为网关:

- .) **BYT17** = 网关 a.b.c.d, 第一位 a;
- .) **BYT18** = 网关 a.b.c.d, 第二位 b;
- .) **BYT19** = 网关 a.b.c.d, 第三位 c;
- .) **BYT20** = 网关 a.b.c.d, 第四位 d;

BYT21~BYT26 共 6 个字节, 为 MAC 地址:

- .) **BYT21** = MAC 地址 a-b-c-d-e-f, 第一位 a;
- .) **BYT22** = MAC 地址 a-b-c-d-e-f, 第二位 b;
- .) **BYT23** = MAC 地址 a-b-c-d-e-f, 第三位 c;
- .) **BYT24** = MAC 地址 a-b-c-d-e-f, 第四位 d;
- .) **BYT25** = MAC 地址 a-b-c-d-e-f, 第五位 e;

.) **BYT26** = MAC 地址 a-b-c-d-e-f, 第六位 f;

BYT27~BYT27 共 1 个字节, 为端口号;

BYT28=Chksum, **BYT6~BYT27** 异或运算或者 0xaa;

BYT29=Chksum, **BYT0~BYT28** 异或运算或者 0x20;

*该命令可在读取 LAN 通信使用;

*****未完待续*****

四、软件设计

- 1、软件初始须首先配置 COM 口;
- 2、接着选择被控设备的设备编号 (可通过设备面板 **Info** 按键读取或者通过设备面板 **Setup** 按键设置);
- 3、测试 COM 口通讯是否正常 (发 1 条**读设备状态**命令看是否准确返回);
- 3、接着读取设备基本配置和当前状态;
- 4、可定时读取设备基本配置和当前状态, 以此判断设备已执行完成软件发送的操作命令;
- 5、为确保设备正确接收到命令, 并执行完相应操作, 部分命令有返回设置选项, 即 **BYT2** 最高位置 1, 建议使用该选项。