

## 定长锁定节目命令序列

该定制项目需求如下：有两个节目，第一个节目为需要定长锁定的节目，第二个节目为平时显示的节目。在制作节目时，第一个节目需要设置播放时段开始时间为 00:00:00, 结束时间也是 00:00:00，第二个节目采用默认设置。

下面以 5E3 为例，具体分析该命令序列的组成和修改。

### 命令序列

当需要锁定节目 1 显示 10 秒钟时，需向控制卡发送如下命令序列（十六进制格式）：

1. 不需要返回值时的锁定节目命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 54 03 01 00 00 00 00 84 96 00 00 00 00 00 00 00 0F 00 15 0F D5 00 A2 0E 0A 00 00 01 50 30 30 30 E8 03 00 00 5A

2. 不需要返回值时的解除节目锁定命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 54 03 01 00 00 00 00 84 96 00 00 00 00 00 00 00 0F 00 18 9F 48 00 A2 0E 0A 00 00 00 50 30 30 30 E8 03 00 00 5A

3. 需要返回值时的锁定节目命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 54 03 01 00 00 00 00 84 96 00 00 00 00 00 00 00 0F 00 D4 0F 14 01 A2 0E 0A 00 00 01 50 30 30 30 E8 03 00 00 5A

4. 需要返回值时的锁定节目命令正确执行后的返回序列：

A5 A5 A5 A5 A5 A5 A5 A5 00 80 01 00 05 00 54 03 01 00 00 80 00 02 96 34 8A 8A 8A 8A 8A 8A 09 00 0A 78 89 00 A0 00 00 00 00 00 00 00 00 00 5A

5. 需要返回值时的解除节目锁定命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 54 03 01 00 00 00 00 84 96 00 00 00 00 00 00 00 0F 00 D9 9F 89 01 A2 0E 0A 00 00 00 50 30 30 30 E8 03 00 00 5A

6. 需要返回值时的解除节目锁定命令正确执行后的返回序列：

与序列 4 相同

7. BX-5A0 需要返回值命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 51 01 00 00 01 00 00 84 96 12 76 00 00 00 20 00 0B 00 57 25 FA 01 A2 0E 06 00 00 01 50 30 30 30 5A

8. BX-5QL 不需要返回值命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 56 13 01 00 00 00 00 84 96 00 00 00 00 00 00 00 0F 00 15 0F C7 00 A2 0E 0A 00 00 01 50 30 30 30 E8 03 00 00 5A

9. BX-5Q2+需要返回值命令序列：

A5 A5 A5 A5 A5 A5 A5 A5 FE FF 00 80 05 01 56 12 00 00 01 00 00 84 96 47 76 00 00 00 20 00 0B 00 57 25 BA 01 A2 0E 06 00 00 01 50 30 30 30 5A

10. BX-5Q2+需要返回值时的锁定节目命令正确执行后的返回序列：

A5 A5 A5 A5 A5 A5 A5 A5 00 80 FE FF 05 01 56 12 00 00 01 80 00 84 96 FB 00 8A 8A 8A AA 8A 09 00 07 E8 E5 01 A0 00 00 00 00 00 00 00 00 00 5A

11. BX-5E1 需要返回值时的锁定节目命令序列，如果不需要返回值把红色 01 改为 00 即可：

A5 A5 A5 A5 A5 A5 A5 A5 01 00 00 80 05 00 54 01 00 00 01 00 00 84 96 52 76 00 00 00 20 00  
0B 00 96 E5 BE 01 A2 0E 06 00 00 01 50 30 30 31 5A

12.BX-5E1 需要返回值时的锁定节目命令正确执行后的返回序列:

A5 A5 A5 A5 A5 A5 A5 A5 00 80 01 00 05 00 54 01 00 00 01 80 00 02 96 62 00 8A 8A 8A AA  
8A 09 00 0A 78 77 00 A0 00 00 00 00 00 00 00 5A

13.BX-6Q1 需要返回值时的锁定节目命令序列, 如果不需要返回值, 把红色 01 改成 00 即可:

A5 A5 A5 A5 A5 A5 A5 A5 FE FF 00 80 07 01 66 01 00 00 01 00 84 00 96 87 76 00 00 00 20 00  
10 00 7F 54 00 19 01 A2 0E 06 00 00 01 50 30 30 30 00 FF FF FF FF 5A

14.BX-6Q1 需要返回值时的锁定节目命令正确执行后的返回序列:

A5 A5 A5 A5 A5 A5 A5 00 80 FE FF 07 01 66 01 00 00 01 80 84 00 96 16 00 8A 8A 8A AA 8A  
09 00 07 E8 00 29 01 A0 00 00 00 00 00 00 00 5A 5A

## 锁定时长修改

以第一组命令序列为例, 紫色字符(50 30 30 30)表示第一个节目号, 修改其他节目号, 修改最后一个 30 为 31(第二个节目)、32(第三个节目)等其他即可。红色字符(E8 03 00 00)表示锁定时长(小端模式), 转换为十进制为 1000, 由于该参数以 10 毫秒为单位, 在本例中表示锁定时长为 10 秒钟。当该参数改变时, 由于绿色字符(15 0F)为蓝色字符(D5)之后(不包括蓝色字符自身及结束符 5A)字符序列的 CRC16 校验值, 其也需做相应改变, 本协议采用的 [CRC16 算法](#) 详见附录。当绿色字符修改后, 蓝色字符也需相应修改, 其为从帧头 A5 后第一个字节(本例中为 01)到绿色字节结尾(本例中为 0F)异或运算的结果。

**注意:** 5Q、6Q 需要把 00 80 05 00 换成 00 80 05 01。

**注意:** 通讯方式是串口把 54 03 01 换成 54 03 00, GPRS 需要换成 54 03 02, 网口不改。

## 控制器类型修改

以第一组命令序列为例, 橙色字符(54 03)代表控制器类型为 5E3, 当橙色字符修改后, 蓝色字符也需相应修改, 其为从帧头 A5 后第一个字节(本例中为 01)到绿色字节结尾(本例中为 0F)异或运算的结果。

各种控制器型号如下:

BX-5A 系列控制型号 0x0051

BX-5AT	0x0051
BX-5A0	0x0151
BX-5A1	0x0251
BX-5A2	0x0351
BX-5A3	0x0451
BX-5A4	0x0551
BX_5A1_WIFI	0x0651
BX_5A2_WIFI	0x0751
BX_5A4_WIFI	0x0851
BX_5A	0x0951
BX_5A2_RF	0x1351
BX_5A4_RF	0x1551

BX-5AT_WIFI	0x1651
BX-5AL	0x1851
BX_5AT_3G	0x1751
BX_5A2_3G	0x1951

AX-AT	----	0x2051
AX-A0	----	0x2151

## BX-5M 系列控制型号 0x0052

BX-5M1	----	0x0052
BX-5M1X	----	0x0152
BX-5M2	----	0x0252
BX-5M3	----	0x0352
BX-5M4	----	0x0452
BX-5MT	----	0x0552

AX-MT	----	0x2052
AX-M1	----	0x2152
AX-M2	----	0x2252
AX-M3	----	0x2352
AX-M4	----	0x2452

## BX-5E 系列控制器型号 0x0054

BX-5E1	----	0x0154
BX-5E2	----	0x0254
BX-5E3	----	0x0354

## BX-5U 系列控制器型号 0x0055

BX-5UT	----	0x0055
BX-5U0	----	0x0155
BX-5U1	----	0x0255
BX-5U2	----	0x0355
BX-5U3	----	0x0455
BX-5U4	----	0x0555
BX-5U5	----	0x0655
BX-5U	----	0x0755
BX-5UL	----	0x0855

DX-UL	----	0x1055
DX-UT	----	0x1155
DX-U0	----	0x1255

AX-UL	----	0x2055
AX-UT	----	0x2155
AX-U0	----	0x2255
AX-U1	----	0x2355
AX-U2	----	0x2455
AX-U3	----	0x2555
AX-U4	----	0x2655

## BX-5Q 系列控制器型号 0x0056

BX-5Q0	0x0056
--------	--------

BX-5Q1	0x0156
BX-5Q2	0x0256
BX-5Q0+	0x1056
BX-5Q1+	0x1156
BX-5Q2+	0x1256
BX_5QL	0x1356
BX-5QS 系列控制器型号	0x0057
BX-5QS1	0x0157
BX-5QS2	0x0257
BX-5QS	0x0357
BX-5QS1+	0x1157
BX-5QS2+	0x1257
BX-5QS+	0x1357
BX-6Q 系列控制器型号	0x0066
BX-6Q0	0x0066
BX-6Q1	0x0166
BX-6Q2	0x0266
BX-6Q3	0x0366
BX-2QL	0x0466

## 返回值验证

检测返回序列以 A5 开头，最后一个字符为 5A，其前面 8 个字符与序列 4 中绿色标识部分一致，就认为该命令正确执行了。若无返回值或与上述标准不一致则认为该命令没有正确执行，重新发送命令即可。

## 数据转义

前面讲述了通讯内容，为正确识别帧头和帧尾，通讯前需对上述通讯内容（除去帧头的 8 个 A5 和帧尾的 1 个 5A）进行转义，具体如下：

封包过程：

- a) 遇到 0xA5，则将之转义为 0xA6, 0x02，如遇到 0xA6，则将之转义为 0xA6, 0x01
- b) 遇到 0x5A，则将之转义为 0x5B, 0x02，如遇到 0x5B，则将之转义为 0x5B, 0x01

解包过程：

- a) 如果遇到连续两个字节为 0xA6, 0x02, 则反转义为 0xA5
  - b) 如果遇到连续两个字节为 0xA6, 0x01, 则反转义为 0xA6
- 对 0x5A 的解包同 0xA5

## CRC16 校验算法

For the calculation of the CRC-16 the following polynomial is used:

---


$$X^{16} + X^{15} + X^2 + 1 = (x + 1) * (X^{15} + x + 1).$$

For this polynomial efficient calculation via a table is possible. Below the algorithm is given in C:

```
#define CRC(crc,byte) (((crc) >> 8) ^ tabel[((crc) ^ (unsigned int) (byte)) & 0xFF])
unsigned short tabel[256] = {
    0X0000, 0XC0C1, 0XC181, 0X0140, 0XC301, 0X03C0, 0X0280, 0XC241,
    0XC601, 0X06C0, 0X0780, 0XC741, 0X0500, 0XC5C1, 0XC481, 0X0440,
    0XCC01, 0X0CC0, 0X0D80, 0XCD41, 0X0F00, 0XCFC1, 0XCE81, 0X0E40,
    0X0A00, 0XCAC1, 0XCB81, 0X0B40, 0XC901, 0X09C0, 0X0880, 0XC841,
    0XD801, 0X18C0, 0X1980, 0XD941, 0X1B00, 0XDBC1, 0XDA81, 0X1A40,
    0X1E00, 0XDEC1, 0XDF81, 0X1F40, 0XDD01, 0X1DC0, 0X1C80, 0XDC41,
    0X1400, 0XD4C1, 0XD581, 0X1540, 0XD701, 0X17C0, 0X1680, 0XD641,
    0XD201, 0X12C0, 0X1380, 0XD341, 0X1100, 0XD1C1, 0XD081, 0X1040,
    0XF001, 0X30C0, 0X3180, 0XF141, 0X3300, 0XF3C1, 0XF281, 0X3240,
    0X3600, 0XF6C1, 0XF781, 0X3740, 0XF501, 0X35C0, 0X03480, 0XF441,
    0X3C00, 0XFCC1, 0XFD81, 0X3D40, 0XFF01, 0X3FC0, 0X3E80, 0XFE41,
    0XFA01, 0X3AC0, 0X3B80, 0XFB41, 0X3900, 0XF9C1, 0XF881, 0X03840,
    0X2800, 0XE8C1, 0XE981, 0X2940, 0XEB01, 0X2BC0, 0X2A80, 0XEA41,
    0XEE01, 0X2EC0, 0X2F80, 0XEF41, 0X2D00, 0XEDC1, 0XEC81, 0X2C40,
    0XE401, 0X24C0, 0X2580, 0XE541, 0X2700, 0XE7C1, 0XE681, 0X2640,
    0X2200, 0XE2C1, 0XE381, 0X2340, 0XE101, 0X21C0, 0X2080, 0XE041,
    0XA001, 0X60C0, 0X6180, 0XA141, 0X6300, 0XA3C1, 0XA281, 0X6240,
    0X6600, 0XA6C1, 0XA781, 0X6740, 0XA501, 0X65C0, 0X6480, 0XA441,
    0X6C00, 0XACC1, 0XAD81, 0X6D40, 0XAF01, 0X6FC0, 0X6E80, 0XAE41,
    0XAA01, 0X6AC0, 0X6B80, 0XAB41, 0X6900, 0XA9C1, 0XA881, 0X6840,
    0X7800, 0XB8C1, 0XB981, 0X7940, 0XBB01, 0X7BC0, 0X7A80, 0XBA41,
    0XBE01, 0X7EC0, 0X7F80, 0XBF41, 0X7D00, 0XBDC1, 0XBC81, 0X7C40,
    0XB401, 0X74C0, 0X7580, 0XB541, 0X7700, 0XB7C1, 0XB681, 0X7640,
    0X7200, 0XB2C1, 0XB381, 0X7340, 0XB101, 0X71C0, 0X7080, 0XB041,
    0X5000, 0X90C1, 0X9181, 0X5140, 0X9301, 0X53C0, 0X5280, 0X9241,
    0X9601, 0X56C0, 0X5780, 0X9741, 0X5500, 0X95C1, 0X9481, 0X5440,
    0X9C01, 0X5CC0, 0X5D80, 0X9D41, 0X5F00, 0X9FC1, 0X9E81, 0X5E40,
    0X5A00, 0X9AC1, 0X9B81, 0X5B40, 0X9901, 0X59C0, 0X5880, 0X9841,
    0X8801, 0X48C0, 0X4980, 0X8941, 0X4B00, 0X8BC1, 0X8A81, 0X4A40,
    0X4E00, 0X8EC1, 0X8F81, 0X4F40, 0X8D01, 0X4DC0, 0X4C80, 0X8C41,
    0X4400, 0X84C1, 0X8581, 0X4540, 0X8701, 0X47C0, 0X4680, 0X8641,
    0X8201, 0X42C0, 0X4380, 0X8341, 0X4100, 0X81C1, 0X8081, 0X4040
};

unsigned short CalcCRC(char *data; int size){
    int i;
    unsigned short crc = 0;
    for (i = 0; i < size; i++) {
        crc = CRC(crc, data[i]);
    }
    return crc;
}
```