bx_sdk_dual.dll附录文档

附录一 枚举参数

屏型 E_ScreenColor_G56

```
public enum E_ScreenColor_G56
{
    eSCREEN_COLOR_SINGLE = 1,    //单基色
    eSCREEN_COLOR_DOUBLE,       //双基色
    eSCREEN_COLOR_THREE,        //七彩色
    eSCREEN_COLOR_FULLCOLOR,    //全彩色
}
```

屏型 E_DoubleColorPixel_G56

```
public enum E_DoubleColorPixel_G56 : int
{
    eDOUBLE_COLOR_PIXTYPE_1 = 1, //双基色1：G+R
    eDOUBLE_COLOR_PIXTYPE_2,    //双基色2：R+G
}
```

单多行 E_arrMode

```
public enum E_arrMode : int
{
    eSINGLELINE,    //单行
    eMULTILINE,     //多行
}
```

日期格式 E_DateStyle

```
public enum E_DateStyle : int
{
    eYYYY_MM_DD_MINUS,    //YYYY-MM-DD
    eYYYY_MM_DD_VIRGURE, //YYYY/MM/DD
    eDD_MM_YYYY_MINUS,    //DD-MM-YYYY
    eDD_MM_YYYY_VIRGURE, //DD/MM/YYYY
    eMM_DD_MINUS,         //MM-DD
    eMM_DD_VIRGURE,       //MM/DD
    eMM_DD_CHS,           //MM月DD日
    eYYYY_MM_DD_CHS,      //YYYY年MM月DD日
}
```

时间格式 E_TimeStyle

```
public enum E_TimeStyle : int
{
    eHH_MM_SS_COLON,  //HH:MM:SS
    eHH_MM_SS_CHS,    //HH时MM分SS秒
    eHH_MM_COLON,     //HH:MM
    eHH_MM_CHS,       //HH时MM分
    eAM_HH_MM,        //AM HH:MM
    eHH_MM_AM,        //HH:MM AM
};
```

星期格式 E_WeekStyle

```
public enum E_WeekStyle : int
{
    eMonday = 1,    //Monday
    eMon,          //Mon.
    eMonday_CHS,  //星期一
}
```

颜色取值 E_Color_G56

```
public enum E_Color_G56
{
    eBLACK,     //黑色
    eRED,       //红色
    eGREEN,     //绿色
    eYELLOW,    //黄色
    eBLUE,      //蓝色
    eMAGENTA,   //品红/洋红
    eCYAN,      //青色
    eWHITE,     //白色
}  //5代时间区只支持四种颜色
```

表盘样式 E_ClockStyle

```
public enum E_ClockStyle
{
    eLINE,     //线形
    eSQUARE,   //方形
    eCIRCLE,   //圆形
};//表盘样式
```

图文区文字方向---暂不支持 E_txtDirection

```
public enum E_txtDirection
{
    pNORMAL,       //正常
    pROTATERIGHT,  //向右旋转
    pMIRROR,       //镜像
    pROTATELEFT,   //向左旋转
};//图文区文字方向---暂不支持
```

附录二 显示特效

```
0x00 -随机显示
0x01 -静止显示
0x02 -快速打出
0x03 -向左移动
0x04 -向左连移
0x05 -向上移动
0x06 -向上连移
0x07 -闪烁
0x08 -飘雪
0x09 -冒泡
0x0a -中间移出
0x0b -左右移入
0x0c -左右交叉移入
0x0d -上下交叉移入
0x0e -画卷闭合
0x0f -画卷打开
0x10 -向左拉伸
0x11 -向右拉伸
0x12 -向上拉伸
0x13 -向下拉伸
0x14 -向左镭射
0x15 -向右镭射
0x16 -向上镭射
0x17 -向下镭射
0x18 -左右交叉拉幕
0x19 -上下交叉拉幕
0x1a -分散左拉
0x1b -水平百页
0x1c -垂直百页
0x1d -向左拉幕
0x1e -向右拉幕
0x1f -向上拉幕
0x20 -向下拉幕
0x21 -左右闭合
0x22 -左右对开
0x23 -上下闭合
0x24 -上下对开
0x25 -向右移动
0x26 -向右连移
0x27 -向下移动
0x28 -向下连移
```

附录三 结构体参数详细说明

ConfigFile

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct ConfigFile
    {
        public byte FileType;
        public byte[] ControllerName;
        ushort Address;
```

```
        public byte Baudrate;
        ushort ScreenWidth;
        ushort ScreenHeight;
        public byte Color;
        public byte MirrorMode;
        public byte OEPol;
        public byte DAPol;
        public byte RowOrder;
        public byte FreqPar;
        public byte OEWidth;
        public byte OEAngle;
        public byte FaultProcessMode;
        public byte CommTimeoutValue;
        public byte RunningMode;
        public byte LoggingMode;
        public byte GrayFlag;
        public byte CascadeMode;
        public byte Default_brightness;
        public byte HUBConfig;
        public byte Language;
        public byte Backup;
        ushort CRC16;
    }
```

参数	说明
FileType	文件类型
ControllerName	控制器名称
Address	控制器地址, 出厂默认地址为 0x0001(0x0000 地址将保留), 0xfffe 为广播地址
Baudrate	串口波特率 0x00 –保持原有波特率不变 0x01 –强制设置为 9600 0x02 –强制设置为 57600
ScreenWidth	显示屏宽度
ScreenHeight	显示屏高度
Color	显示屏颜色定义 Bit0 表示红， bit1 表示绿， bit2 表示蓝， 对于每一个 Bit， 0 表示灭， 1 表示亮
MirrorMode	0x00 –无镜向 0x01 –镜向
OEPol	OE极性 0x00–OE低有效 0x01–OE高有效
DAPol	数据极性 0x00 –数据低有效 0x01 –数据高有效
RowOrder	行序模式， 该值范围为 0-31 0-15 代表正序 0 代表从第 0 行开始顺序扫描 1 代表从第 1 行开始顺序扫描..... 16-31 代表逆序 0 代表从第 0 行开始逆序扫描 1 代表从第 1 行开始逆序扫描
FreqPar	CLK 分频倍数 注意： 针对于 AX 系列， 为后级分频 数值为 0~15， 共 16 个等级
OEWidth	OE 宽度
OEAngle	OE 提前角
FaultProcessMode	控制器的错误处理模式 0x00 –自动处理 0x01 –手动处理(此模式仅供调试人员使用)
CommTimeoutValue	通讯超时设置（单位秒） 建议值: 串口– 2S TCP/IP – 6S GPRS – 30S

参数	说明
RunningMode	控制器运行模式， 具体定义如下： 0x00 –正常模式 0x01 –调试模式
LoggingMode	日志记录模式 0x00 –无日志 0x01 –只对控制器错误及对错误进行的错误进行记录 0x02 –对控制器的所有操作进行记录， 包括： 控制器接收的各条指令、发生的错误及错误处理
GrayFlag	灰度标志(仅 5Q 卡时有该字节) 0x00–无灰度 0x01–灰度
CascadeMode	级联模式： (仅 5Q 卡时有该字节) 0x00–非级联模式 0x01–级联模式
Default_brightness	AX 系列控制器专用， 表示上电时， 默认的亮度等级值。 根据不同的屏幕类型有所不同
HUBConfig	HUB 板设置(仅 6E 控制器支持) 0x00–HUB512 默认项 0x01–HUB256
Language	控制器多语言显示区。 0x00 ----简体中文显示。 0x01 ----非中文显示， 控制器显示图形加英文字符。 其他值保留
Backup	备用字节
CRC16	整个文件的 CRC16 校验

ConfigFile_G6

```
   [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct ConfigFile_G6
    {
        public byte FileType;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        public byte[] ControllerName;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 48)]
        byte[] ScreenAddress;
        ushort Address;
        public byte Baudrate;
        ushort ScreenWidth;
        ushort ScreenHeight;
        public byte Color;
        public byte modeofdisp;
        public byte TipLanguage;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 5)]
        public byte[] Reserved;
```

```
        public byte FaultProcessMode;
        public byte CommTimeoutValue;
        public byte RunningMode;
        public byte LoggingMode;
        public byte DevideScreenMode;
        public byte Reserved2;
        public byte Default_brightness;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 5)]
        public byte[] Backup;
        public ushort CRC16;
    };
```

参数	说明
FileType	文件类型
ControllerName	控制器名称
ScreenAddress	屏幕安装地址限制为 24个字节长度
Address	控制器地址
Baudrate	串口波特率 0x00 –保持原有波特率不变 0x01 –强制设置为 9600 0x02 –强制设置为 57600
ScreenWidth	显示屏宽度
ScreenHeight	显示屏高度
Color	显示屏颜色定义 Bit0 表示红， bit1 表示绿， bit2 表示蓝， 对于每一个 Bit， 0 表示灭， 1 表示亮
modeofdisp	6Q 系列显示模式： 0为888, 1为565，对其余控制卡该字节为0
TipLanguage	0 表示上位机软件是中文版，底层固件在显示提示信息时需调用内置的中文提示信息 1 表示上位机软件是英文版，底层固件在显示提示信息时需调用内置的英文提示信息 255 表示上位机软件是其他语言版，底层固件在显示提示信息时需调用自定义提示信息
Reserved	5个备用字节
FaultProcessMode	控制器的错误处理模式 0x00 –自动处理 0x01 –手动处理(此模式仅供调试人员使用)
CommTimeoutValue	通讯超时设置（单位秒） 建议值: 串口– 2S TCP/IP – 6S GPRS – 30S
RunningMode	控制器运行模式， 具体定义如下: 0x00 –正常模式 0x01 –调试模式
LoggingMode	日志记录模式 0x00 –无日志 0x01 –只对控制器错误及对错误进行的错误进行记录 0x02 –对控制器的所有操作进行记录， 包括： 控制器接收的各条指令、发生的错误及错误处理
DevideScreenMode	针对 6Q2 卡的分屏模式 对其余的卡为保留字节 0
Default_brightness	AX 系列控制器专用，表示上电时，默 认的亮度等级值。其余的控制卡该字 节为保留字 0

参数	说明
Backup	备用字节
CRC16	整个文件的 CRC16 校验

Ping_data

```
    [StructLayout(LayoutKind.Sequential, Pack = 1)]
    public struct Ping_data
    {
        public ushort ControllerType;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
        public byte[] FirmwareVersion;
        public byte ScreenParaStatus;
        public ushort uAddress;
        public byte Baudrate;
        public ushort ScreenWidth;
        public ushort ScreenHeight;
        public byte Color;
        public byte CurrentBrigtness;
        public byte CurrentOnOffStatus;
        public ushort ScanConfNumber;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 9)]
        public byte[] reversed;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 20)]
        public byte[] ipAdder;
    }
```

参数	说明
ControllerType	控制器类型 小端存储低位在前高位在后， 比如 0x254 反着取，低位表示系列，高位编号 [0x54, 0x02] 【系列，编号】
FirmwareVersion	固件版本号
ScreenParaStatus	控制器参数文件状态 0x00 –控制器中没有参数配置文件，以下返回的是控制器的默认参数。此时， PC 软件应提示用户必须先加载屏参。 0x01 –控制器中有参数配置文件
uAddress	控制器地址，屏号
Baudrate	波特率
ScreenWidth	屏宽
ScreenHeight	屏高
Color	显示屏颜色定义 1单色屏 3双色屏 7三色屏 255全彩
CurrentBrigtness	当前亮度值 整数1-16
CurrentOnOffStatus	控制器开关机状态 0 关机 1开机
ScanConfNumber	扫描配置编号
reversed	保留位
ipAdder	控制器ip地址

heartbeatData

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 1)]
    public struct heartbeatData
    {
        public string password;
        public string ip;
        public string subNetMask;
        public string gate;
        public short port;
        public string mac;
        public string netID;
    }
```

参数	说明
password	密码
ip	控制器IP地址
subNetMask	子网掩码
gate	网关
port	端口
mac	MAC地址
netID	控制器网络ID

NetSearchCmdRet

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct NetSearchCmdRet
    {
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 6)]
        public byte[] Mac;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] IP;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] SubNetMask;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] Gate;
        public ushort Port;
        public byte IPMode;
        public byte IPStatus;
        public byte ServerMode;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] ServerIPAddress;
        public ushort ServerPort;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
        public byte[] ServerAccessPassword;
        public ushort HeartBeatInterval;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 12)]
        public byte[] CustomID;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        public byte[] BarCode;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] ControllerType;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
        public byte[] FirmwareVersion;
        public byte ScreenParaStatus;
        public ushort Address;
        public byte Baudrate;
        public ushort ScreenWidth;
        public ushort ScreenHeight;
        public byte Color;
        public byte BrightnessAdjMode;
        public byte CurrentBrigtness;
        public byte TimingOnOff;
```

```
        public byte CurrentOnOffStatus;
        public ushort ScanConfNumber;
        public byte RowsPerChanel;
        public byte GrayFlag;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] UnitWidth;
        public byte modeofdisp;
        public byte NetTranMode;
        public byte PackageMode;
        public byte BarcodeFlag;
        public ushort ProgramNumber;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] CurrentProgram;
        public byte ScreenLockStatus;
        public byte ProgramLockStatus;
        public byte RunningMode;
        public byte RTCStatus;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] RTCYear;
        public byte RTCMonth;
        public byte RTCDate;
        public byte RTCHour;
        public byte RTCMinute;
        public byte RTCSecond;
        public byte RTCWeek;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        public byte[] Temperature1;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        public byte[] Temperature2;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] Humidity;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] Noise;
        public byte Reserved;
        public byte LogoFlag;
        public ushort PowerOnDelay;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] WindSpeed;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] WindDirction;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] PM2_5;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] PM10;
        public ushort ExtendParaLen;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        public byte[] ControllerName;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 44)]
        public byte[] ScreenLocation;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] NameLocalationCRC32;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] PM100;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        public byte[] AtmosphericPressure ;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] illumination;
    }
```

参数	说明
Mac	Mac 地址
IP	控制器 IP 地址
SubNetMask	子网掩码
Gate	网关
Port	端口
IPMode	1表示 DHCP 2表示手动设置
IPStatus	0表示 IP 设置失败 1表示 IP 设置成功
ServerMode	Bit[0]表示服务器模式是否使能：1 –使能，0 –禁止 Bit[1]表示服务器模式：1 –web 模式，0 –普通模式
ServerIPAddress	服务器 IP 地址
ServerPort	服务器端口号
ServerAccessPassword	服务器访问密码
HeartBeatInterval	默认20S， 心跳时间间隔（单位：秒）
CustomID	用户自定义 ID，作为网络 ID 的前半部分，便于用户识别其控制卡
BarCode	条形码，作为网络 ID 的后半部分，用以实现网络 ID 的唯一性
ControllerType	其中低位字节表示设备系列，而高位字节表示设备编号，例如 BX - 6Q2 应表示为[0x66, 0x02]，其它型号依此类推。
FirmwareVersion	Firmware 版本号
ScreenParaStatus	控制器参数文件状态 0x00 –控制器中没有参数配置文件，以下返回的是控制器的默认参数。此时，PC软件应提示用户必须先加载屏参。 0x01 –控制器中有参数配置文件
Address	控制器地址控制器出厂默认地址为 0x0001(0x0000 地址将保留) 控制除了对发送给自身地址的数据包进行处理外，还需对广播数据包进行处理
Baudrate	波特率 0x00 –保持原有波特率不变 0x01 –强制设置为 9600 0x02 –强制设置为 57600
ScreenWidth	显示屏宽度
ScreenHeight	显示屏高度
Color	对于无灰度系统，单色时返回 1，双色时返回 3，三色时返回 7；对于有灰度系统，返回 255

参数	说明
BrightnessAdjMode	调亮模式 0x00 –手动调亮 0x01 –定时调亮 0x02 –自动调亮
CurrentBrigtness	当前亮度值
TimingOnOff	Bit0 –定时开关机状态， 0 表示无定时开关机， 1 表示有定时开关机
CurrentOnOffStatus	开关机状态
ScanConfNumber	扫描配置编号
RowsPerChanel	一路数据带几行
GrayFlag	对于无灰度系统，返回 0；对于有灰度系
UnitWidth	最小单元宽度
modeofdisp	6Q 显示模式 : 0 为 888, 1 为 565， 其余卡为 0
NetTranMode	当该字节为 0 时，网口通讯使用老的模式，即 UDP 和 TCP 均根据下面的PackageMode 字节确定包长，并且 UDP通讯时，将大包分为小包，每发送一小包做一下延时 当该字节不为 0 时，网口通讯使用新的模式，即 UDP 的包长等于 UDPPackageMode * 8KBYTE，且不再分为小包，将整包数据丢给协议栈 TCP 的包长等于 PackageMode * 16KBYTE
PackageMode	包模式。 0 小包模式，分包 600 byte。 1 大包模式，分包 16K byte
BarcodeFlag	是否设置了条码 ID如果设置了，该字节第 0 位为 1，否则为0
ProgramNumber	控制器上已有节目个数
CurrentProgram	当前节目名
ScreenLockStatus	Bit0 –是否屏幕锁定 1b’0 –无屏幕锁定，1b’1 –屏幕锁定
ProgramLockStatus	Bit0 –是否节目锁定 1b’0 –无节目锁定，1’b1 –节目锁定
RunningMode	控制器运行模式
RTCStatus	RTC 状态 0x00 – RTC 异常 0x01 – RTC 正常
RTCYear	年

参数	说明
RTCMonth	月
RTCDate	日
RTCHour	小时
RTCMinute	分钟
RTCSecond	秒
RTCWeek	星期，范围为 1~7，7 表示周日
Temperature1	温度传感器当前值
Temperature2	温度传感器当前值
Humidity	湿度传感器当前值
Noise	噪声传感器当前值(除以 10 为当前值) 针对 BX - ZS(485) 0xffff 时无效
Reserved	保留字节
LogoFlag	0：表示未设置 Logo 节目 1：表示设置了 Logo 节目
PowerOnDelay	0：未设置开机延时 1：开机延时时长
WindSpeed	风速(除以 10 为当前值) 0xfffff 时无效
WindDirction	风向(当前值) 0xfffff 时无效
PM2_5	PM2.5 值(当前值) 0xfffff 时无效
PM10	PM10 值(当前值) 0xfffff 时无效
ExtendParaLen	扩展参数长度
ControllerName	控制器名称 限制为 16 个字节长度(全是 0x00 表示屏参丢失，参数无效，上位机空白显示)
ScreenLocation	屏幕安装地址 限制为 44 个字节长度(全是 0x00 表示屏参丢失，参数无效，上位机空白显示)
NameLocalationCRC32	控制器和屏幕安装地址共 60 个字节的CRC32 校验值，该值是为了便于上位机区分此处 64 个字节是表示控制器名称还是用来表示控制器名称和屏幕安装地址，进而采取不同的处理策略 为了保持兼容，下位机不对该值进行验证

参数	说明
PM100	PM100(当前值) 0xfffff 时无效
AtmosphericPressure	大气压力值（KPa) 0xfffff 时无效
illumination	光照强度(当前值) 0xffffffff 时无效

NetSearchCmdRet_Web

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct NetSearchCmdRet_Web
    {
        byte CmdGroup;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 6)]
        byte Mac;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        byte IP;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        byte SubNetMask;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        byte Gate;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        byte Port;
        byte IPMode;
        byte IPStatus;
        byte ServerMode;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        byte ServerIPAddress;
        ushort ServerPort;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
        byte ServerAccessPassword;
        public ushort HeartBeatInterval;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 12)]
        byte CustomID;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 128)]
        byte WebUserID;
        public int GroupNum;
        byte DomainFlag;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 128)]
        byte DomainName;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 128)]
        byte WebControllerName;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        byte BarCode;
        ushort ControllerType;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 8)]
        byte FirmwareVersion;
        byte ScreenParaStatus;
        ushort Address;
```

```
        byte Baudrate;
        ushort ScreenWidth;
        ushort ScreenHeight;
        byte Color;
        byte BrightnessAdjMode;
        byte CurrentBrigtness;
        byte TimingOnOff;
        byte CurrentOnOffStatus;
        ushort ScanConfNumber;
        byte RowsPerChanel;
        byte GrayFlag;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        byte UnitWidth;
        byte modeofdisp;
        byte NetTranMode;
        byte PackageMode;
        byte BarcodeFlag;
        ushort ProgramNumber;
        int CurrentProgram;
        byte ScreenLockStatus;
        byte ProgramLockStatus;
        byte RunningMode;
        byte RTCStatus;
        ushort RTCYear;
        byte RTCMonth;
        byte RTCDate;
        byte RTCHour;
        byte RTCMinute;
        byte RTCSecond;
        byte RTCWeek;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        byte Temperature1;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        byte Temperature2;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        byte Humidity;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 2)]
        byte Noise;
        byte Reserved;
        byte LogoFlag;
        ushort PowerOnDelay;
        ushort WindSpeed;
        ushort WindDirction;
        ushort PM2_5;
        ushort PM10;

        ushort ExtendParaLen;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        byte ControllerName;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 44)]
        byte ScreenLocation;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        byte NameLocalationCRC32;
    };
```

```
C#
```

参数	说明
CmdGroup	0xA4 命令组
Mac	Mac 地址
IP	控制器 IP 地址
SubNetMask	子网掩码
Gate	网关
Port	端口
IPMode	1表示 DHCP 2表示手动设置
IPStatus	0表示 IP 设置失败 1表示 IP 设置成功
ServerMode	Bit[0]表示服务器模式是否使能：1 –使能，0 –禁止 Bit[1]表示服务器模式：1 –web 模式，0 –普通模式
ServerIPAddress	服务器 IP 地址
ServerPort	服务器端口号
ServerAccessPassword	服务器访问密码
HeartBeatInterval	默认20S， 心跳时间间隔（单位：秒）
CustomID	用户自定义 ID，作为网络 ID 的前半部分，便于用户识别其控制卡
WebUserID	WEB 平台用户 id
GroupNum	屏幕组号
DomainFlag	域名标志 0 - 无域名，1—域名
DomainName	域名名称 当 DomainFlag 为 1 时下发
WebControllerName	LED00001 WEB 平台控制器名称
BarCode	条形码，作为网络 ID 的后半部分，用以实现网络 ID 的唯一性
ControllerType	其中低位字节表示设备系列，而高位字节表示设备编号，例如 BX - 6Q2 应表示为[0x66, 0x02]，其它型号依此类推。
FirmwareVersion	Firmware 版本号
ScreenParaStatus	控制器参数文件状态 0x00 –控制器中没有参数配置文件，以下返回的是控制器的默认参数。此时，PC软件应提示用户必须先加载屏参。 0x01 –控制器中有参数配置文件
Address	控制器地址控制器出厂默认地址为 0x0001(0x0000 地址将保留) 控制除了对发送给自身地址的数据包进行处理外，还需对广播数据包进行处理

参数	说明
Baudrate	波特率 0x00 –保持原有波特率不变 0x01 –强制设置为 9600 0x02 –强制设置为 57600
ScreenWidth	显示屏宽度
ScreenHeight	显示屏高度
Color	对于无灰度系统，单色时返回 1，双色时返回 3，三色时返回 7；对于有灰度系统，返回 255
BrightnessAdjMode	调亮模式 0x00 –手动调亮 0x01 –定时调亮 0x02 –自动调亮
CurrentBrigtness	当前亮度值
TimingOnOff	Bit0 –定时开关机状态， 0 表示无定时开关机， 1 表示有定时开关机
CurrentOnOffStatus	开关机状态
ScanConfNumber	扫描配置编号
RowsPerChanel	一路数据带几行
GrayFlag	对于无灰度系统，返回 0；对于有灰度系
UnitWidth	最小单元宽度
modeofdisp	6Q 显示模式 : 0 为 888, 1 为 565， 其余卡为 0
NetTranMode	当该字节为 0 时，网口通讯使用老的模式，即 UDP 和 TCP 均根据下面的PackageMode 字节确定包长，并且 UDP通讯时，将大包分为小包，每发送一小包做一下延时 当该字节不为 0 时，网口通讯使用新的模式，即 UDP 的包长等于 UDPPackageMode * 8KBYTE，且不再分为小包，将整包数据丢给协议栈 TCP 的包长等于 PackageMode * 16KBYTE
PackageMode	包模式。 0 小包模式，分包 600 byte。 1 大包模式，分包 16K byte
BarcodeFlag	是否设置了条码 ID如果设置了，该字节第 0 位为 1，否则为0
ProgramNumber	控制器上已有节目个数
CurrentProgram	当前节目名
ScreenLockStatus	Bit0 –是否屏幕锁定 1b’0 –无屏幕锁定，1b’1 –屏幕锁定

参数	说明
ProgramLockStatus	Bit0 –是否节目锁定 1b'0 –无节目锁定，1'b1 –节目锁定
RunningMode	控制器运行模式
RTCStatus	RTC 状态 0x00 – RTC 异常 0x01 – RTC 正常
RTCYear	年
RTCMonth	月
RTCDate	日
RTCHour	小时
RTCMinute	分钟
RTCSecond	秒
RTCWeek	星期，范围为 1~7，7 表示周日
Temperature1	温度传感器当前值
Temperature2	温度传感器当前值
Humidity	湿度传感器当前值
Noise	噪声传感器当前值(除以 10 为当前值) 针对 BX - ZS(485) 0xffff 时无效
Reserved	保留字节
LogoFlag	0：表示未设置 Logo 节目 1：表示设置了 Logo 节目
PowerOnDelay	0：未设置开机延时 1：开机延时时长
WindSpeed	风速(除以 10 为当前值) 0xfffff 时无效
WindDirction	风向(当前值) 0xfffff 时无效
PM2_5	PM2.5 值(当前值) 0xfffff 时无效
PM10	PM10 值(当前值) 0xfffff 时无效
ExtendParaLen	扩展参数长度

参数	说明
ControllerName	控制器名称 限制为 16 个字节长度(全是 0x00 表示屏参丢失，参数无效，上位机空白显示)
ScreenLocation	屏幕安装地址 限制为 44 个字节长度(全是 0x00 表示屏参丢失，参数无效，上位机空白显示)
NameLocalationCRC32	控制器和屏幕安装地址共 60 个字节的CRC32 校验值，该值是为了便于上位机区分此处 64 个字节是表示控制器名称还是用来表示控制器名称和屏幕安装地址，进而采取不同的处理策略 为了保持兼容，下位机不对该值进行验证

TimingOnOff

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct TimingOnOff
    {
        public byte onHour;
        public byte onMinute;
        public byte offHour;
        public byte offMinute;
    }
```

参数	说明
onHour	开机小时
onMinute	开机分钟
offHour	关机小时
offMinute	关机分钟

Brightness

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct Brightness
    {
        public byte BrightnessMode;
        public byte HalfHourValue0;
        public byte HalfHourValue1;
        public byte HalfHourValue2;
        public byte HalfHourValue3;
        public byte HalfHourValue4;
        public byte HalfHourValue5;
        public byte HalfHourValue6;
        public byte HalfHourValue7;
        public byte HalfHourValue8;
```

```
        public byte HalfHourValue9;
        public byte HalfHourValue10;
        public byte HalfHourValue11;
        public byte HalfHourValue12;
        public byte HalfHourValue13;
        public byte HalfHourValue14;
        public byte HalfHourValue15;
        public byte HalfHourValue16;
        public byte HalfHourValue17;
        public byte HalfHourValue18;
        public byte HalfHourValue19;
        public byte HalfHourValue20;
        public byte HalfHourValue21;
        public byte HalfHourValue22;
        public byte HalfHourValue23;
        public byte HalfHourValue24;
        public byte HalfHourValue25;
        public byte HalfHourValue26;
        public byte HalfHourValue27;
        public byte HalfHourValue28;
        public byte HalfHourValue29;
        public byte HalfHourValue30;
        public byte HalfHourValue31;
        public byte HalfHourValue32;
        public byte HalfHourValue33;
        public byte HalfHourValue34;
        public byte HalfHourValue35;
        public byte HalfHourValue36;
        public byte HalfHourValue37;
        public byte HalfHourValue38;
        public byte HalfHourValue39;
        public byte HalfHourValue40;
        public byte HalfHourValue41;
        public byte HalfHourValue42;
        public byte HalfHourValue43;
        public byte HalfHourValue44;
        public byte HalfHourValue45;
        public byte HalfHourValue46;
        public byte HalfHourValue47;
    }
```

参数	说明
BrightnessMode	0x00 –手动调亮 0x01 –定时调亮 注:以下的亮度值表，在定时调亮和手 动调亮时控制器才需处理。但在协议上 不论什么模式，此表都需要发送给控制器
HalfHourValue0	00:00 – 00:29 的亮度值， 0x00 – 0x0f
HalfHourValue1	00:30-0:59的亮度值， 0x00 – 0x0f
.....	.....
HalfHourValue47	23:30-23:59的亮度值， 0x00 – 0x0f

ControllerStatus_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct ControllerStatus_G56
    {
        public byte onoffStatus;
        public byte timingOnOff;
        public byte brightnessAdjMode;
        public byte brightness;
        public short programeNumber;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] currentProgram;
        public byte screenLockStatus;
        public byte programLockStatus;
        public byte runningMode;
        public byte RTCStatus;
        public short RTCYear;
        public byte RTCMonth;
        public byte RTCDate;
        public byte RTCHour;
        public byte RTCMinute;
        public byte RTCSecond;
        public byte RTCWeek;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        public byte[] temperature1;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 3)]
        public byte[] temperature2;
        public short humidity;
        public short noise;
        public byte switchStatus;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 12)]
        public byte[] CustomID;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 16)]
        public byte[] BarCode;
    }
```

参数	说明
onoffStatus	开关机状态 Bit 0 –开机/关机， 0 表示关机， 1 表示开机
timingOnOff	定时开关机状态 0 表示无定时开关机， 1 表示有定时开关机
brightnessAdjMode	亮度模式 0x00 –手动调亮 0x01 –定时调亮 0x02 –自动调亮
brightness	当前亮度值
programeNumber	控制器上已有节目个数
currentProgram	当前节目名
screenLockStatus	是否屏幕锁定，0 –无屏幕锁定， 1 –屏幕锁定
programLockStatus	是否节目锁定， 0 –无节目锁定，1 –节目锁定
runningMode	控制器运行模式
RTCStatus	RTC 状态0x00 – RTC 异常 0x01 – RTC 正常
RTCYear	年
RTCMonth	月
RTCDate	日
RTCHour	时
RTCMinute	分
RTCSecond	秒
RTCWeek	星期 1--7
temperature1	温度1传感器当前值
temperature2	温度2传感器当前值
humidity	湿度传感器当前值
noise	噪声传感器当前值
switchStatus	测试按钮状态 0 –打开 1 –闭合
CustomID	用户自定义 ID，作为网络 ID 的前半部分，便于用户识别其控制卡
BarCode	条形码，作为网络 ID 的后半部分，用以实现网络 ID 的唯一性

TimingReset

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct TimingReset
    {
        public byte rstMode;
        uint RstInterval;
        public byte rstHour1;
        public byte rstMin1;
        public byte rstHour2;
        public byte rstMin2;
        public byte rstHour3;
        public byte rstMin3;
    }
```

参数	说明
rstMode	复位模式 0x00 –取消定时复位功能 0x01 –周期复位， 此时 RstInterval 字段有效 0x02 –只在指定时间复位
RstInterval	复位周期， 单位： 分钟; 如此字段为 0， 不进行复位操作
rstHour1	小时 0Xff–表示此组无效， 下同
rstMin1	分钟
rstHour2	小时
rstMin2	分钟
rstHour3	小时
rstMin3	分钟

BattleTime

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct BattleTime
    {
        public short BattleRTCYear;
        public byte BattleRTCMonth;
        public byte BattleRTCDate;
        public byte BattleRTCHour;
        public byte BattleRTCMinute;
        public byte BattleRTCSecond;
        public byte BattleRTCWeek;
    }
```

参数	说明
BattleRTCYear	年
BattleRTCMonth	月
BattleRTCDate	日
BattleRTCHour	时
BattleRTCMinute	分
BattleRTCSecond	秒
BattleRTCWeek	星期

EQprogrampTime_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQprogrampTime_G56
    {
        public byte StartHour;
        public byte StartMinute;
        public byte StartSecond;
        public byte EndHour;
        public byte EndMinute;
        public byte EndSecond;
    };
```

参数	说明
StartHour	开始 时
StartMinute	开始 分
StartSecond	开始 秒
EndHour	结束 时
EndMinute	结束 分
EndSecond	结束 秒

public struct EQprogramppGrp_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQprogramppGrp_G56
    {
        public byte playTimeGrpNum;
        public EQprogrampTime_G56 timeGrp0;
        public EQprogrampTime_G56 timeGrp1;
        public EQprogrampTime_G56 timeGrp2;
        public EQprogrampTime_G56 timeGrp3;
        public EQprogrampTime_G56 timeGrp4;
        public EQprogrampTime_G56 timeGrp5;
        public EQprogrampTime_G56 timeGrp6;
        public EQprogrampTime_G56 timeGrp7;
    };
```

参数	说明
playTimeGrpNum	播放时间有效组数 0 没有播放时段全天播放 最大值8
timeGrp0	第一组播放时段EQprogrampTime_G56
timeGrp1	第二组播放时段
timeGrp2	第三组播放时段
timeGrp3	第四组播放时段
timeGrp4	播放时段
timeGrp5	播放时段
timeGrp6	播放时段
timeGrp7	播放时段

EQprogramHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQprogramHeader
    {
        public byte FileType;
        public uint ProgramID;
        public byte ProgramStyle;
        public byte ProgramPriority;
        public byte ProgramPlayTimes;
        public ushort ProgramTimeSpan;
        public byte ProgramWeek;
        public ushort ProgramLifeSpan_sy;
        public byte ProgramLifeSpan_sm;
        public byte ProgramLifeSpan_sd;
        public ushort ProgramLifeSpan_ey;
        public byte ProgramLifeSpan_em;
        public byte ProgramLifeSpan_ed;
    }
```

参数	说明
FileType	文件类型默认：0x00LOGO文件:0x08 扫描配置文件:0x02 日志文件:0x06 字库文件:0x05 提示信息库文件: 0x07
ProgramID	节目ID
ProgramStyle	节目类型 Bit0 –全局节目标志位 Bit1 –动态节目标志位 Bit2 –屏保节目标志位
ProgramPriority	节目等级，带播放时段的节目优先级为 1，不带播放时段的节目优先级为 0
ProgramPlayTimes	节目重播放次数
ProgramTimeSpan	播放的方式 0x0000 –顺序播放 其它–播放的时间长度（单位-秒）
ProgramWeek	节目星期属性 Bit0 – 1 表示一周中的每一天都播放 Bit0 为 0 时，需判断 bit1-bit7 的来决定 每天播放，bit1-bit7 依次表示周一到周 日
ProgramLifeSpan_sy	0xffff–年月日均无效，可以无限期播放 0xfffe-年无效 其它–播放起始年份，范围为 0x1900 ~ 0x2099，即 1900 年到 2099 年
ProgramLifeSpan_sm	起始月份 0xfe--月无效
ProgramLifeSpan_sd	起始日 0xfe—日无效
ProgramLifeSpan_ey	结束年
ProgramLifeSpan_em	结束日
ProgramLifeSpan_ed	结束天

EQscreenframeHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQscreenframeHeader
    {
        public byte FrameDispFlag;
        public byte FrameDispStyle;
        public byte FrameDispSpeed;
        public byte FrameMoveStep;
        public byte FrameWidth;
        public ushort FrameBackup;
    }
```

参数	说明
FrameDispFlag	边框是否显示 0x00 –不显示 0x01 –显示 注：如边框不显示，则不发送以 下数据
FrameDispStype	边框显示方式： 0x00 –闪烁 0x01 –顺时针转动 0x02 –逆时针转动 0x03 –闪烁加顺时针转动 0x04 –闪烁加逆时针转动 0x05 –红绿交替闪烁 0x06 –红绿交替转动 0x07 –静止打出
FrameDispSpeed	边框显示速度
FrameMoveStep	边框移动步长，单位为点，此参 数范围为 1~16
FrameWidth	边框组元宽度
FrameBackup	保留字

EQareaframeHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQareaframeHeader
    {
        public byte AreaFFlag;
        public byte AreaFDispStyle;
        public byte AreaFDispSpeed;
        public byte AreaFMoveStep;
        public byte AreaFWidth;
        public ushort AreaFBackup;
    }
```

参数	说明
AreaFFlag	区域边框标志位 0无边框 1有边框
AreaFDispStyle	边框显示方式： 0x00 –闪烁 0x01 –顺时针转动 0x02 –逆时针转动 0x03 –闪烁加顺时针转动 0x04 –闪烁加逆时针转动 0x05 –红绿交替闪烁 0x06 –红绿交替转动 0x07 –静止打出
AreaFDispSpeed	边框显示速度
AreaFMoveStep	边框移动步长 该值取值范围：1~8；
AreaFWidth	边框组元长度
AreaFBackup	边框组元宽度

EQareaHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQareaHeader
    {
        public byte AreaType;
        public ushort AreaX;
        public ushort AreaY;
        public ushort AreaWidth;
        public ushort AreaHeight;
    }
```

参数	说明
AreaType	区域类型 图文字幕:0x00 字库区域:0x01 时间区:0x02 温度区：0x03 湿度区：0x04 噪声区：0x05 透明文本：0x06 霓虹区：0x08 战斗时间：0x09
AreaX	区域左上角横坐标(Top Left)，单 位 Pixel
AreaY	区域左上角纵坐标(Top Left)，单 位 Pixel
AreaWidth	区域宽度，单位 Pixel
AreaHeight	区域高度，单位 Pixel

EQpageHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQpageHeader
    {
        public byte PageStyle;
        public byte DisplayMode;
        public byte ClearMode;
        public byte Speed;
        public ushort StayTime;
        public byte RepeatTime;
        public ushort ValidLen;
        public E_arrMode arrMode;
        public ushort fontSize;
        public uint color;
        public byte fontBold;
        public byte fontItalic;
        public E_txtDirection tdirection;
        public ushort txtSpace;
        public byte Valign;
        public byte Halign;
    }
```

参数	说明
PageStyle	数据页类型，固定=0
DisplayMode	显示方式
ClearMode	退出方式/清屏方式，固定为0
Speed	速度等级1-64，1最快
StayTime	停留时间，单位为 10ms
RepeatTime	重复次数，固定=1
ValidLen	有效宽度，此字段只在左移右移方式下有效，默认等于区域宽度
arrMode	排列方式--单行多行E_arrMode
fontSize	字体大小
color	字体颜色 E_Color_G56 此通过此枚举值可以直接配置七彩色，如果大于枚举范围使用RGB888模式
fontBold	是否为粗体 0否 1是
fontItalic	是否为斜体 0否 1是
tdirection	文字方向E_txtDirection，无效
txtSpace	文字间隔，无效
Valign	横向对齐方式（0系统自适应、1左对齐、2居中、3右对齐）
Halign	纵向对齐方式（0系统自适应、1上对齐、2居中、3下对齐）

EQprogram

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQprogram
    {
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] fileName;
        public byte fileType;
        public uint fileLen;
        public IntPtr fileAddre;
        public uint fileCRC32;
    }
```

参数	说明
fileName	节目参数文件名
fileType	文件类型
fileLen	参数文件长度
fileAddre	文件所在的缓存地址
fileCRC32	文件CRC32校验码

getPageData【】

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct getPageData
    {
        ushort allPageNub;
        uint pageLen;
        public byte[] fileAddre;
    }
```

参数	说明
allPageNub	
pageLen	
fileAddre	

EQunitHeader【】

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQunitHeader
    {
        ushort UnitX;
        ushort UnitY;
        public byte UnitType;
        public byte Align;
        public byte UnitColor;
        public byte UnitMode;
    }
```

参数	说明
UnitX	
UnitY	
UnitType	
Align	
UnitColor	
UnitMode	

EQtimeAreaData_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQtimeAreaData_G56
    {
        public E_arrMode linestyle;
        public uint color;
        public string fontName;
        public ushort fontSize;
        public byte fontBold;
        public byte fontItalic;
        public byte fontUnderline;
        public byte fontAlign;
        public byte date_enable;
        public E_DateStyle datestyle;
        public byte time_enable;
        public E_TimeStyle timestyle;
        public byte week_enable;
        public E_WeekStyle weekstyle;
    }
```

参数	说明
linestyle	排列方式E_arrMode
color	字体颜色
fontName	字体名字
fontSize	字体大小
fontBold	是否为粗体 0否 1是
fontItalic	斜体0否 1是
fontUnderline	字体加下划线0否 1是
fontAlign	对齐方式 0居左 1居中 2居右
date_enable	是否添加日期 0否 1是
datestyle	日期格式E_DateStyle
time_enable	是否添加时间 0否 1是
timestyle	时间格式E_TimeStyle
week_enable	是否添加星期 0否 1是
weekstyle	星期格式E_WeekStyle

EQAnalogClockHeader_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQAnalogClockHeader_G56
    {
        public ushort OrignPointX;
        public ushort OrignPointY;
        public byte UnitMode;
        public byte HourHandWidth;
        public byte HourHandLen;
        public uint HourHandColor;
        public byte MinHandWidth;
        public byte MinHandLen;
        public uint MinHandColor;
        public byte SecHandWidth;
        public byte SecHandLen;
        public uint SecHandColor;
    }
```

参数	说明
OrignPointX	圆点横坐标
OrignPointY	圆点纵坐标
UnitMode	表针模式
HourHandWidth	时针宽度
HourHandLen	时针长度
HourHandColor	时针颜色
MinHandWidth	分针宽度
MinHandLen	分针长度
MinHandColor	分针颜色
SecHandWidth	秒针宽度
SecHandLen	秒针长度
SecHandColor	秒针颜色

EQprogramHeader_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQprogramHeader_G6
    {
        public byte FileType;
        public uint ProgramID;
        public byte ProgramStyle;
        public byte ProgramPriority;
        public byte ProgramPlayTimes;
        public ushort ProgramTimeSpan;
        public byte SpecialFlag;
        public byte CommExtendParaLen;
        public ushort ScheduNum;
        public ushort LoopValue;
        public byte Intergrate;
        public byte TimeAttributeNum;
        public ushort TimeAttributeOOffset;
        public byte ProgramWeek;
        public ushort ProgramLifeSpan_sy;
        public byte ProgramLifeSpan_sm;
        public byte ProgramLifeSpan_sd;
        public ushort ProgramLifeSpan_ey;
        public byte ProgramLifeSpan_em;
        public byte ProgramLifeSpan_ed;
        public byte PlayPeriodGrpNum;
    }
```

参数	说明
FileType	文件类型默认：0x00LOGO文件:0x08 扫描配置文件:0x02 日志文件:0x06 字库文件:0x05 提示信息库文件: 0x07
ProgramID	节目ID
ProgramStyle	节目类型 Bit0 –全局节目标志位 Bit1 –动态节目标志位 Bit2 –屏保节目标志位
ProgramPriority	节目等级，带播放时段的节目优先级为 1，不带播放时段的节目优先级为 0
ProgramPlayTimes	节目重播放次数
ProgramTimeSpan	播放的方式
SpecialFlag	特殊节目标
CommExtendParaLen	扩展参数长度，默认为0x00
ScheduNum	节目调度
LoopValue	调度规则循环次数
Intergrate	调度相关
TimeAttributeNum	时间属性组数
TimeAttribute0Offset	第一组时间属性偏移量--目前只支持一组
ProgramWeek	节目星期属性
ProgramLifeSpan_sy	开始年
ProgramLifeSpan_sm	开始月
ProgramLifeSpan_sd	开始日
ProgramLifeSpan_ey	结束年
ProgramLifeSpan_em	结束日
ProgramLifeSpan_ed	结束天
PlayPeriodGrpNum	播放时段的组数

EQscreenframeHeader_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQscreenframeHeader_G6
    {
        public byte FrameDispStype;
        public byte FrameDispSpeed;
        public byte FrameMoveStep;
        public byte FrameUnitLength;
        public byte FrameUnitWidth;
        public byte FrameDirectDispBit;
    }
```

参数	说明
FrameDispStype	边框是否显示 0x00 –不显示 0x01 –显示
FrameDispSpeed	边框显示速度
FrameMoveStep	边框移动步长，单位为点，此参 数范围为 1~16
FrameUnitLength	边框组元长度
FrameUnitWidth	边框组元宽度
FrameDirectDispBit	上下左右边框显示标志位 该字节低 4 位(高位 Bit3 到低位 Bit0)分别表示上下左右边框是否 显示 0 –显示 1 –不显示 (目前只支持 6QX-M 卡)

EQSound_6G

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQSound_6G
    {
        public byte SoundFlag;
        public byte SoundPerson;
        public byte SoundVolum;
        public byte SoundSpeed;
        public byte SoundDataMode;
        public int SoundReplayTimes;
        public int SoundReplayDelay;
        public byte SoundReservedParaLen;
        public byte Soundnumdeal;
        public byte Soundlanguages;
        public byte Soundwordstyle;
        public int SoundDataLen;
        public IntPtr SoundData;
    }
```

参数	说明
SoundFlag	是否使能语音播放;0 表示不使能语音; 1 表示播放下文中 SoundData 部分内容
SoundPerson	发音人 该值范围是 0 - 5，共 6 种选择 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundVolum	音量 该值范围是 0~10，共 11 种，0表示静音 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundSpeed	语速 该值范围是 0~10，共 11 种 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundDataMode	SoundData 的编码格式： 该值意义如下：0x00 GB2312; 0x01 GBK; 0x02 BIG5; 0x03 UNICODE 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送
SoundReplayTimes	重播次数, 该值为 0，表示播放 1 次该值为 1，表示播放 2 次 该值为 0xffffffff，表示播放无限次只有 SoundFlag（是否使能语播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReplayDelay	重播时间间隔 该值表示两次播放语音的时间间隔，单位为 10ms 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReservedParaLen	语音参数保留参数长度,固定=3
Soundnumdeal	0：自动判断 1：数字作号码处理 2：数字作数值处理 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数
Soundlanguages	0：自动判断语种 1：阿拉伯数字、度量单位、特殊符号等合成为中文 2：阿拉伯数字、度量单位、特殊符号等合成为英文 只有当 SoundFlag 为 1 且 SoundReservedParaLen不为 0才发送此参数（目前只支持中英文）
Soundwordstyle	0：自动判断发音方式 1：字母发音方式 2：单词发音方式; 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数
SoundDataLen	语音数据长度; 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送

参数	说明
SoundData	语音数据 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送

ClockColor_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 1)]
    public struct ClockColor_G56
    {
        public uint Color369;
        public uint ColorDot;
        public uint ColorBG;
    }
```

参数	说明
Color369	369点颜色
ColorDot	点颜色
ColorBG	表盘外圈颜色 模式没有圈则此颜色无效

EQareaHeader_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 1)]
    public struct EQareaHeader_G6
    {
        public byte AreaType;
        public ushort AreaX;
        public ushort AreaY;
        public ushort AreaWidth;
        public ushort AreaHeight;
        public byte BackGroundFlag;
        public byte Transparency;
        public byte AreaEqual;
        public EQSound_6G stSoundData;
    }
```

参数	说明
AreaType	区域类型 图文字幕:0x00 字库区域:0x01 时间区:0x02 温度区：0x03 湿度区：0x04 噪声区：0x05 透明文本：0x06 霓虹区：0x08 战斗时间：0x09
AreaX	区域左上角横坐标(Top Left)，单 位 Pixel
AreaY	区域左上角纵坐标(Top Left)，单 位 Pixel
AreaWidth	区域宽度，单位 Pixel
AreaHeight	区域高度，单位 Pixel
BackGroundFlag	是否有背景，目前不支持，固定给0
Transparency	透明度，目前不支持，固定给101
AreaEqual	前景、背景区域大小是否相同，目前不支持，固定给0
stSoundData	语音内容EQSound_6G

EQPicAreaSoundHeader_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQPicAreaSoundHeader_G6
    {
        public byte SoundPerson;
        public byte SoundVolum;
        public byte SoundSpeed;
        public byte SoundDataMode;
        public uint SoundReplayTimes;
        public uint SoundReplayDelay;
        public byte SoundReservedParaLen;
        public byte Soundnumdeal;
        public byte Soundlanguages;
        public byte Soundwordstyle;
    }//图文分区播放语音
```

参数	说明
SoundPerson	发音人 该值范围是 0 - 5，共 6 种选择 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundVolum	音量 该值范围是 0~10，共 11 种，0表示静音 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundSpeed	语速 该值范围是 0~10，共 11 种 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundDataMode	SoundData 的编码格式： 该值意义如下：0x00 GB2312; 0x01 GBK; 0x02 BIG5; 0x03 UNICODE 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送
SoundReplayTimes	重播次数, 该值为 0，表示播放 1 次该值为 1，表示播放 2 次 该值为 0xffffffff，表示播放无限次只有 SoundFlag（是否使能语播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReplayDelay	重播时间间隔 该值表示两次播放语音的时间间隔，单位为 10ms 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReservedParaLen	语音参数保留参数长度,固定=3
Soundnumdeal	0：自动判断 1：数字作号码处理 2：数字作数值处理 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数
Soundlanguages	0：自动判断语种 1：阿拉伯数字、度量单位、特殊符号等合成为中文 2：阿拉伯数字、度量单位、特殊符号等合成为英文 只有当 SoundFlag 为 1 且 SoundReservedParaLen不为 0才发送此参数（目前只支持中英文）
Soundwordstyle	0：自动判断发音方式 1：字母发音方式 2：单词发音方式； 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数

EQTimeAreaBattle_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQTimeAreaBattle_G6
    {
        public ushort BattleStartYear;
        public byte BattleStartMonth;
        public byte BattleStartDate;
        public byte BattleStartHour;
        public byte BattleStartMinute;
        public byte BattleStartSecond;
        public byte BattleStartWeek;
        public byte StartUpMode;
    }
```

参数	说明
BattleStartYear	起始年份（BCD格式，下同）
BattleStartMonth	起始月份
BattleStartDate	起始日期
BattleStartHour	起始小时
BattleStartMinute	起始分钟
BattleStartSecond	起始秒钟
BattleStartWeek	起始星期值
StartUpMode	启动模式

EQpageHeader_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQpageHeader_G6
    {
        public byte PageStyle;
        public byte DisplayMode;
        public byte ClearMode;
        public byte Speed;
        public ushort StayTime;
        public byte RepeatTime;
        public ushort ValidLen;
        public byte CartoonFrameRate;
        public byte BackNotValidFlag;
        public E_arrMode arrMode;
        public ushort fontSize;
        public uint color;
        public byte fontBold;
        public byte fontItalic;
```

```
        public E_txtDirection tdirection;
        public ushort txtSpace;
        public byte Valign;
        public byte Halign;
    }
```

参数	说明
PageStyle	数据页类型，固定=0
DisplayMode	显示方式
ClearMode	退出方式/清屏方式，固定为0
Speed	速度等级1-64，1最快
StayTime	停留时间，单位为 10ms
RepeatTime	重复次数，固定=1
ValidLen	有效宽度，此字段只在左移右移方式下有效，默认等于区域宽度
CartoonFrameRate	特技为动画方式时，该值代表其帧率，固定=0
BackNotValidFlag	背景无效标志，固定=0
arrMode	排列方式--单行多行E_arrMode
fontSize	字体大小
color	字体颜色 E_Color_G56 此通过此枚举值可以直接配置七彩色，如果大于枚举范围使用RGB888模式
fontBold	是否为粗体 0否 1是
fontItalic	是否为斜体 0否 1是
tdirection	文字方向E_txtDirection，无效
txtSpace	文字间隔，无效
Valign	横向对齐方式（0系统自适应、1左对齐、2居中、3右对齐）
Halign	纵向对齐方式（0系统自适应、1上对齐、2居中、3下对齐）

EQprogram_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 1)]
    public struct EQprogram_G6
    {
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] fileName;
        public byte fileType;
        public uint fileLen;
        public IntPtr fileAddre;
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
```

```
        public byte[] dfileName;
        public byte dfileType;
        public uint dfileLen;
        public IntPtr dfileAddre;
    }
```

参数	说明
fileName	节目参数文件名
fileType	文件类型
fileLen	参数文件长度
fileAddre	文件所在的缓存地址
dfileName	节目数据文件名
dfileType	节目数据文件类型
dfileLen	数据文件长度
dfileAddre	数据文件缓存地址

GetDirBlock_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct GetDirBlock_G56
    {
        public byte fileType;
        public ushort fileNumber;
        public IntPtr dataAddre;
    }
```

参数	说明
fileType	要获取的文件类型
fileNumber	返回有多少个文件
dataAddre	返回文件列表地址

FileAttribute_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct FileAttribute_G56
    {
        [MarshalAs(UnmanagedType.ByValArray, SizeConst = 4)]
        public byte[] fileName;
        public byte fileType;
        public int fileLen;
        public int fileCRC;
    }
```

参数	说明
fileName	文件名
fileType	文件类型
fileLen	文件长度
fileCRC	文件CRC校验

EQdynamicHeader

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQdynamicHeader
    {
        public byte RunMode;
        ushort Timeout;
        public byte ImmePlay;
        public byte AreaType;
        ushort AreaX;
        ushort AreaY;
        ushort AreaWidth;
        ushort AreaHeight;
    }
```

参数	说明
RunMode	动态区运行模式 0— 动态区数据循环显示。 1— 动态区数据显示完成后静止显 示最后一页数据。 2— 动态区数据循环显示，超过设 定时间后数据仍未更新时不再 显示 3— 动态区数据循环显示，超过设 定时间后数据仍未更新时显示 Logo 信息,Logo 信息即为动 态区域的最后一页信息 4— 动态区数据顺序显示，显示完 最后一页后就不再显示
Timeout	动态区数据超时时间，单位为秒
ImmePlay	是否立即播放 该字节为 0 时，该动态区域与异 步节目一起播放 该字节为 1 时，异步节目停止播 放，仅播放该动态区域 注意： 当该字节为 0 时，RelateAllPro 到 RelateProSerialN-1 的参数才有 效，否则无效 当该参数为 1 时，由于不与异步 节目同时播放，为控制该动态区 域能及时结束，可选择 RunMode 参数为 2 或 4，当然也可通过删除 该区域来实现
AreaType	区域类型0x10
AreaX	区域左上角横坐标(Top Left)，单 位 Pixel
AreaY	区域左上角纵坐标(Top Left)，单 位 Pixel
AreaWidth	区域宽度，单位 Pixel
AreaHeight	区域高度，单位 Pixel

EQSoundDepend_6G

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQSoundDepend_6G
    {
        public byte VoiceID;
        public EQSound_6G stSound;
    }
```

参数	说明
VoiceID	语音队列中每个语音的 ID，从 0 开始
stSound	语言数据EQSound_6G

FileCRC16_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct FileCRC16_G56
    {
        IntPtr fileAddre;
        ushort fileLen;
        ushort fileCRC16;
    }
```

参数	说明
fileAddre	文件地址指針
fileLen	文件长度
fileCRC16	文件CRC16校验

###

FileCRC32_G56

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct FileCRC32_G56
    {
        IntPtr fileAddre;
        ushort fileLen;
    }
```

参数	说明
fileAddre	文件地址指針
fileLen	文件长度
fileCRC32	文件CRC32校验

DynamicAreaParams

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct DynamicAreaParams
    {
        public byte uAreaId;
        public bxdualsdk.EQareaHeader_G6 oAreaHeader_G6;
        public bxdualsdk.EQpageHeader_G6 stPageHeader;
        public IntPtr fontName;
        public IntPtr strAreaTxtContent;
    }
```

参数	说明
uAreaId	区域ID
oAreaHeader_G6	区域属性
stPageHeader	显示数据属性EQareaHeader_G6
fontName	字体名称EQpageHeader_G6
strAreaTxtContent	显示内容数据

BxAreaFrmae_Dynamic_G6

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct BxAreaFrmae_Dynamic_G6
    {
    public byte AreaFFlag;       // 1 0x00 区域边框标志位;
    public EQscreenframeHeader_G6 oAreaFrame;
    public byte[] pStrFramePathFile;
};
```

参数	说明
AreaFFlag	区域边框标志位 0无边框 1有边框
oAreaFrame	边框属性EQscreenframeHeader_G6
pStrFramePathFile	边框图片文件

BXSound_6G

- 注意：这个语音结构体BXSound_6G仅在动态区时使用；图文分区播放语音请使用：EQPicAreaSoundHeader_G6

```
[StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
public struct BXSound_6G
{
    public byte SoundFlag;
    public byte SoundPerson;
    public byte SoundVolum;
    public byte SoundSpeed;
    public byte SoundDataMode;
```

```
    public int SoundReplayTimes;
    public int SoundReplayDelay;
    public byte SoundReservedParaLen;
    public byte Soundnumdeal;
    public byte Soundlanguages;
    public byte Soundwordstyle;
    public int SoundDataLen;
    public IntPtr SoundData;
};
```

参数	说明
SoundFlag	是否使能语音播放;0 表示不使能语音; 1 表示播放下文中 SoundData 部分内容
SoundPerson	发音人 该值范围是 0 - 5，共 6 种选择 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0

参数	说明
SoundVolum	音量 该值范围是 0~10，共 11 种，0表示静音 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundSpeed	语速 该值范围是 0~10，共 11 种 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 5
SoundDataMode	SoundData 的编码格式： 该值意义如下：0x00 GB2312; 0x01 GBK; 0x02 BIG5; 0x03 UNICODE 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送
SoundReplayTimes	重播次数, 该值为 0，表示播放 1 次该值为 1，表示播放 2 次 该值为 0xffffffff，表示播放无限次只有 SoundFlag（是否使能语播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReplayDelay	重播时间间隔 该值表示两次播放语音的时间间隔，单位为 10ms 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送该值默认为 0
SoundReservedParaLen	语音参数保留参数长度,固定=3
Soundnumdeal	0：自动判断 1：数字作号码处理 2：数字作数值处理 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数
Soundlanguages	0：自动判断语种 1：阿拉伯数字、度量单位、特殊符号等合成为中文 2：阿拉伯数字、度量单位、特殊符号等合成为英文 只有当 SoundFlag 为 1 且 SoundReservedParaLen不为 0才发送此参数（目前只支持中英文）
Soundwordstyle	0：自动判断发音方式 1：字母发音方式 2：单词发音方式； 只有当 SoundFlag 为 1 且SoundReservedParaLen不为 0才发送此参数
SoundDataLen	语音数据长度; 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送
SoundData	语音数据 只有 SoundFlag（是否使能语音播放）为 1 时才发送该字节，否则不发送

DynamicAreaBaseInfo_5G

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct DynamicAreaBaseInfo_5G
    {
        public byte nType;
        public byte DisplayMode;
        public byte ClearMode;
        public byte Speed;
        public ushort StayTime;
        public byte RepeatTime;
        public EQfontData oFont;
        public IntPtr fontName;
        public IntPtr strAreaTxtContent;
        public IntPtr filePath;
    }
```

参数	说明
nType	nType=1:文本； nType=2:图片；
DisplayMode	显示方式
ClearMode	退出方式/清屏方式，固定为0
Speed	速度等级1-64，1最快
StayTime	停留时间，单位为 10ms
RepeatTime	重复次数，固定为1
oFont	字体属性EQfontData
fontName	字体名称
strAreaTxtContent	显示文本
filePath	图片路径

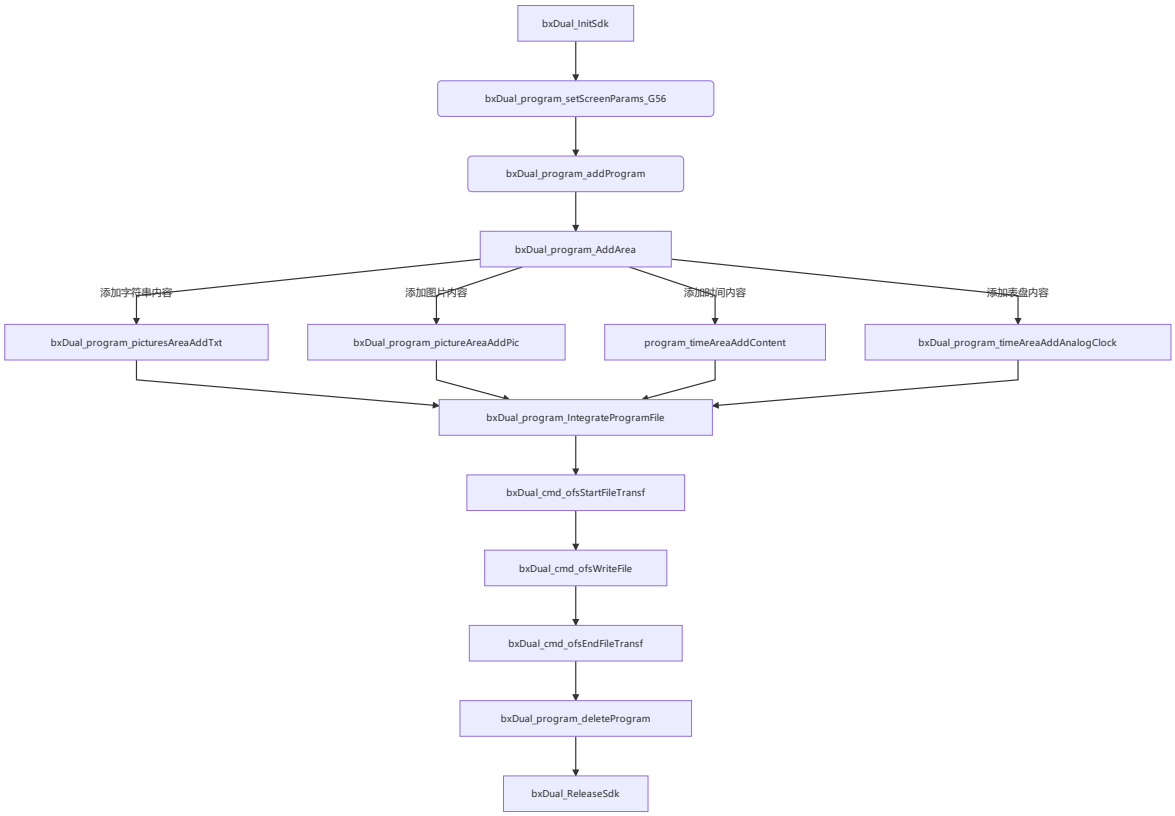
EQfontData

```
    [StructLayoutAttribute(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack =
1)]
    public struct EQfontData
    {
        public E_arrMode arrMode;
        public ushort fontSize;
        public uint color;
        public byte fontBold;
        public byte fontItalic;
        public E_txtDirection tdirection;
        public ushort txtSpace;
        public byte Halign;
        public byte Valign;
    }
```

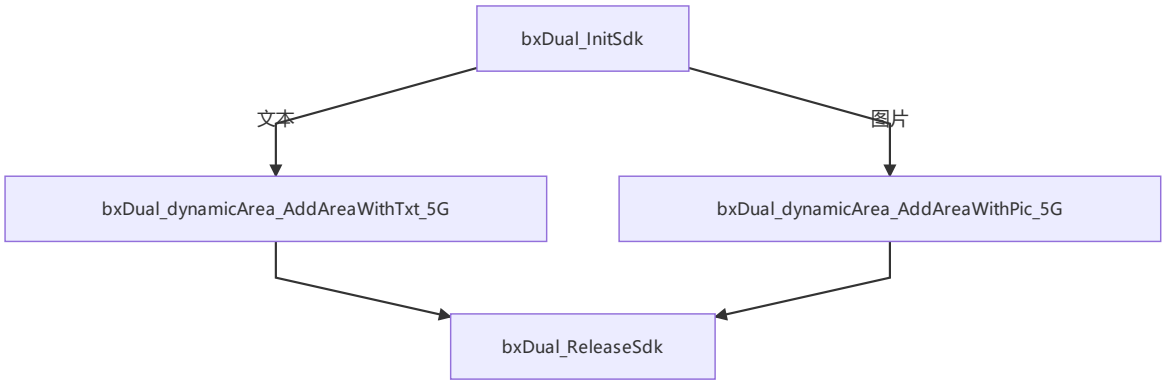
参数	说明
arrMode	排列方式--单行多行 E_arrMode
fontSize	字体大小
color	字体颜色 E_Color_G56 此通过此枚举值可以直接配置七彩色，如果大于枚举范围使用RGB888模式
fontBold	是否为粗体 0否 1是
fontItalic	是否为斜体 0否 1是
tdirection	文字方向E_txtDirection，无效
txtSpace	文字间隔，无效
Halign	横向对齐方式（0系统自适应、1左对齐、2居中、3右对齐）
Valign	纵向对齐方式（0系统自适应、1上对齐、2居中、3下对齐）

附录四 发送流程

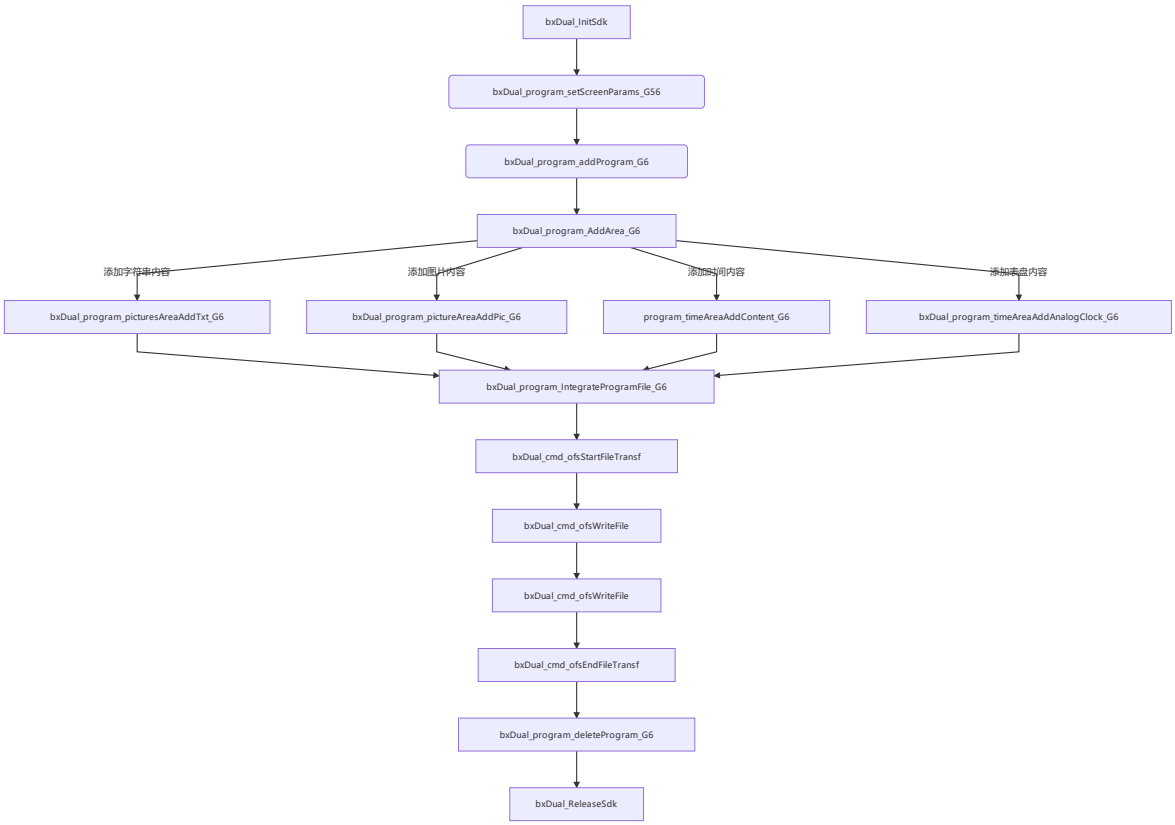
BX-5系列控制卡发送节目



BX-5系列控制卡发送动态区(BX-5E)



BX-6系列控制卡发送节目



BX-6系列控制卡发送动态区(6E 6EX)