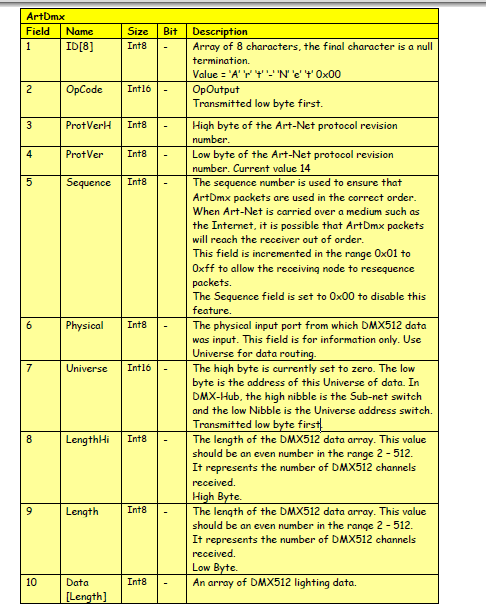
**DMX协议说明**

**Art-NET Dmx packet definition**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 名字 | 大小 | Bit | 描述 |
| 1 | ID[8] | Int8 | -8 | 8个字符数组，最后一个字符是一个空的终止。  Value = ‘A’ ‘r’ ‘t’ ‘-‘ ‘N’ ‘e’ ‘t’ 0x00 |
| 2 | OpCode | Int16 | -2 | OpOutput 先传输低字节 |
| 3 | ProtVerHi | Int8 | -1 | Art-Net协议版本号高字节 |
| 4 | ProtVerLo | Int8 | -1 | Art-Net协议版本号低字节，当前值为14 |
| 5 | Sequence | Int8 | -1 | 序列号用来确保ArtDmx数据包以正确的顺序使用。当Art-net是通过一个介质诸如因特网，可能的ArtDmx数据包在接收时出现乱序。  此字段递增取值范围0x01至0xFF，以允许接收节点重新排序数据包。  序列号设置为0来禁止这功能 |
| 6 | Physical | Int8 | -1 | 从DMX512数据输入的物理输入端口。此字段仅供参考。使用全集来数据路由。 |
| 7 | SubUni | Int8 | - | 目的地数据包的15位端口地址的低字节 |
| 8 | Net | Int8 | - | 目的地数据包的15位端口地址的高7位字节 |
| 9 | LengthHi | Int8 | - | DMX512数据数组的长度。这个值范围需要是2-512.  表示数据包编码的DMX512通道数目。注：产品转换Art-Net到DMX512可以选择总发送512个通道。  高字节 |
| 10 | Length | Int8 | - | 上述内容低字节 |
| 11 | Data[Length] | Int8 | - | DMX512的灯光数据的变长数组(18-529) |



（1）数据包总数：530 端口6454

（2）0到17是包头 ,byte[14] 是通道号（0-255）//byte[13]

开头识别字符：Art-Net （0-8）byte //（0-7）

（3）byte从18到529是有效数据;

**初始化代码：**

public Socket dmxSo;

dmxSo = new Socket(AddressFamily.InterNetwork, SocketType.Dgram, ProtocolType.Udp);

EndPoint ipep = new IPEndPoint(IPAddress.Any, port);

dmxSo.Bind(ipep);

EndPoint epip = new IPEndPoint(IPAddress.Any, 0);

dmxSo.BeginReceiveFrom(asyncRevDmxBuf, 0, asyncRevDmxBuf.Length, SocketFlags.None, ref epip, new AsyncCallback(OnDmxRecvCallback), "udp");//开始新的异步接

/// <summary>

/// 异步接收主处理过程

/// </summary>

/// <param name="state"></param>

private void OnDmxRecvCallback(IAsyncResult iar)

{

EndPoint ep = new IPEndPoint(IPAddress.Any, 0);

if (hjSocket != null)

{

int Datalen = 0;

try

{

Datalen = dmxSo.EndReceiveFrom(iar, ref ep);//终结异步接收

}

catch (Exception ex)

{

this.closeDmx();

return;

}

if (Datalen > 0)

{

try

{

IPEndPoint ipp = (IPEndPoint)ep;

this.processBuf(asyncRevDmxBuf, ipp.Address.ToString(), ipp.Port.ToString());//将得到的数据包放到getBuf中处理

EndPoint epip = new IPEndPoint(IPAddress.Any, 0);

dmxSo.BeginReceiveFrom(asyncRevDmxBuf, 0, asyncRevDmxBuf.Length, SocketFlags.None, ref epip, new AsyncCallback(OnDmxRecvCallback), "udp");//开始新的异步接

}

catch (Exception ex)

{

ex = ex;

}

}

}

}

void closeDmx()

{

try

{

dmxUdp = false;

if (dmxSo != null) { dmxSo.Close(); }

}

catch { }

}

**发送DMX代码：**

/// 播放记录的Dmx数据

/// </summary>

/// <param name="fileName"></param>

public bool StartDmxPlay(string fileName)

{

if (!File.Exists(clsGloba.AppPath + "\\Dmx\\" + fileName)) return false;

if (mBolplayDmx)

{

StopDmxPlay();

}

mBolplayDmx = true;

rs = new BinaryReader(new FileStream(clsGloba.AppPath + "\\Dmx\\" + fileName, FileMode.Open), System.Text.ASCIIEncoding.Default);

//\_tDmxplay.Change(0,200);

//if(rs.ReadToEnd == string.Empty)

thrPlay = new Thread(new ThreadStart(proBufToDmx));

thrPlay.Priority = ThreadPriority.Normal;

thrPlay.Start();

return true;

}

**//播放线程**

private void proBufToDmx()

{

try

{

byte biaozhun = rs.ReadByte(); // 读取是什么协议0，1

byte[] buf = null;

//byte[] bufTime = null;

int bLen = 0;

switch (biaozhun)

{

case 0:

bLen = 528;

break;

case 1:

bLen = 530;

break;

}

while (mBolplayDmx)

{

try

{

if (clsGloba.isHDL\_ArtNet == true)

{

#region

bLen = 530;

ticks = System.DateTime.Now;

// bufTime = rs.ReadBytes(2);

int timers1 = rs.ReadByte();

int timers2 = rs.ReadByte();

int allTime = timers1 \* 256 + timers2;

//rs.ReadInt32();

buf = rs.ReadBytes(530);

if (buf.Length == 0) break; //

while (true)

{

Application.DoEvents();

if (!mBolplayDmx) break;

ts = System.DateTime.Now - ticks; //现在时间 减去 读取前时间

oldspanTick = ts.Milliseconds; //相隔的毫秒数

clsGloba.DMX\_number = oldspanTick;

if (oldspanTick > allTime || oldspanTick == allTime)

{

for (int y = 0; y < clsGloba.dmxIP\_list.Count; y++)

{

clsGloba.myhjCls.SendDmxBufToRemote(buf, clsGloba.dmxIP\_list[y].ToString(), port);

}

break;

}

}

#endregion

}

else

{

#region

bLen = 528;

ticks = System.DateTime.Now;

int timers = rs.ReadByte();

buf = rs.ReadBytes(bLen);

if (buf.Length == 0) break;

while (true)

{

Application.DoEvents();

if (!mBolplayDmx) break;

ts = System.DateTime.Now - ticks;

oldspanTick = ts.Milliseconds;

if (oldspanTick > timers || oldspanTick == timers)

{

for (int y = 0; y < clsGloba.dmxIP\_list.Count; y++)

{

clsGloba.myhjCls.SendDmxBufToRemote(buf, clsGloba.dmxIP\_list[y].ToString(), port);

}

break;

}

}

#endregion

}

/\*采用休眠

ticks = System.DateTime.Now;

byte[] buf=rs.ReadBytes(DmxDataLen-1);

timers=rs.ReadByte();

if(buf.Length == 0)

{break;}

clsGloba.myhjCls.SendBufToRemote(buf,ip,port);

Thread.Sleep (timers);

\*/

}

catch

{

break;

}

}

StopDmxPlay();

}

catch { StopDmxPlay(); }

}

/// <summary>

/// 有目的地的发送数据给设备

/// </summary>

/// <param name="buf"></param>

/// <param name="ip"></param>

public void SendDmxBufToRemote(byte[] buf, string ip, string port)

{

if (dmxSo != null)

{

try

{

EndPoint ep = new IPEndPoint(IPAddress.Parse(ip), int.Parse(port));

dmxSo.BeginSendTo(buf, 0, buf.Length, SocketFlags.None, ep, new AsyncCallback(this.OnEndSendCallback), "Ok");

}

catch (Exception ex) { ex = ex; }

}

else

{

this.SendBufToRemote(buf, ip, port);

}

}

**接收DMX代码**

**//开始准备接收启动**

/// <summary>

/// 开始录制Dmx数据

/// </summary>

public bool StartDmxRecord(string fileName, int chns, byte biaozhun)

{

dmxLastBuf = new byte[DmxDataLen];

dBiaozhun = biaozhun;

DmxSavePath = clsGloba.AppPath + "\\Dmx\\" + fileName;

try

{

//if(File.Exists(DmxSavePath)){File.Delete(DmxSavePath);}

if (ws != null) ws.Close();

ws = new BinaryWriter(new FileStream(DmxSavePath, FileMode.OpenOrCreate, FileAccess.ReadWrite), System.Text.ASCIIEncoding.Default);

ws.Write(biaozhun);

}

catch (Exception ex) { MessageBox.Show("File is the visit, please try again later!" + ex.Message); if (ws != null) { ws = null; };return false; }

dmxChnNum = chns;

// startTick = 00;//2010-06-25

DmxRecordCount = 0;

AvDmxSpeed = 0;

DmxRecordStart = true;

return true;

//dmxLastBuf=null;

}

**//接收数据包的处理**

void processBuf(byte[] buf, string ip, string port)

{

string mb5 = "Art-Net";

if (System.Text.ASCIIEncoding.Default.GetString(buf, 0, 7) == mb5)

{ clsGloba.CDmx.ArtNetBufFromDmx(buf); return; }

}

/// 处理ARTDmx数据内容

/// </summary>

/// <param name="buf"></param>

public void ArtNetBufFromDmx(byte[] buf1)

{

if (!DmxRecordStart) return;

if (dmxChnNum > 0 && buf1[14] != dmxChnNum) return;

byte[] bufmm = null;

ticks = System.DateTime.Now; //现在时刻

int old1 = 0; int old2 = 0;

if (DmxRecordCount == 0)

{

oldspanTick = 0; //写入时间等于0

}

else

{

ts = ticks - startTick;

oldspanTick = ts.Milliseconds;

// if (oldspanTick > 200) oldspanTick = 200;

}

old1 = oldspanTick / 256;

old2 = oldspanTick % 256;

startTick = ticks;

ws.Write(Convert.ToByte(old1));

ws.Write(Convert.ToByte(old2));

ws.Write(buf1, 0, 530);

DmxRecordCount++;

}