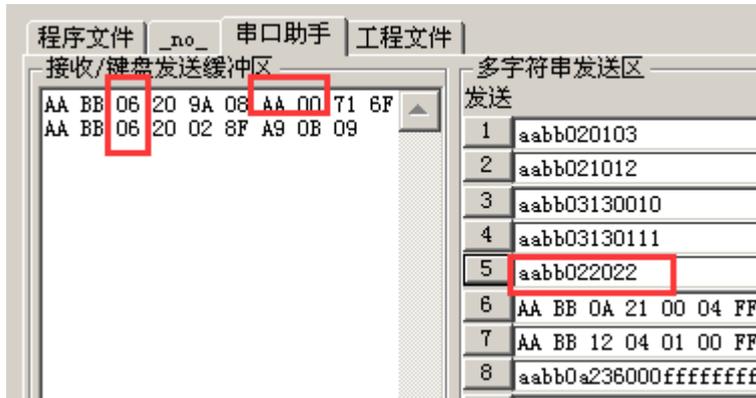


关于 UID 卡号包含 AA 字符的处理方法

实例：



发送读卡号指令： AABB022022,

第一张卡返回：AABB06209A08AA00716F,

第二张卡返回：AABB0620028FA90B09。

解释：

第一个是卡号为 9A08AA71, 所以上位机收到的是 AA 后面插入了 00, 但是前面的长度字节还是 06, 没有加 1, 虽然后面实际数据包多了 1 个 00。

第二个卡号是 028FA90B, 没有带 AA。

在 YHY502 和 YHY522, YHY521 等 RFID 模块的串口通信协议里面有这样的规定：
主机到 模块的传输协议：

主机发送的数据帧格式

Header	Length	Command	Data	CSUM
2 Byte	1 Byte	1 Byte	N Bytes	1 Byte

说明：

1. **Header:** 数据帧头表示一个数据帧的开始, 这里固定为 0xAABB。
2. **Length:** 长度字包含了从命令字到校验字的数据长度。
3. **Command:** 命令字表示要执行的命令的代码。
4. **Data:** 数据域包含了各个命令代码需要携带的参数, 数据域可以为空。
5. **CSUM:** 校验字, 其结果为从长度字到数据域最后一个字节逐个字节异或的结果。

$$CSUM = Length \oplus Command \oplus Data[0] \oplus Data[1] \cdots \oplus Data[n-1]$$

注意：如果在数据域内或者校验字出现“AA”，为了和数据帧头区分开，需要在发送的命令里面的数据域内的“AA”后面插入“00”，**但是长度字不需要改变**，也就是说，不需要因为插入了一个“00”字节而把长度字加1，RFID模块在收到这样的数据时会自动判断并剔除无效的“00”。

Example:

AA BB 0A 21 00 04 FF FF FF FF FF FF 2F

AA BB: Header

0A: Length

21: Command

00 04 FF FF FF FF FF FF: Data

2F: CSUM = 00 ⊕ 04 ⊕ FF ⊕ FF ⊕ FF ⊕ FF ⊕ FF ⊕ FF

发送示例代码 Code example:

```
void UartSend(unsigned char *cSendBuffer)
{
    unsigned char i;
    unsigned char cChecksum;
    TXC = 1;
    g_bReceCommandOk = 0;
    UDR = 0xAA;
    while (!TXC);
    TXC = 1;
    UDR = 0xBB;    // send header AABB
    while (!TXC);
    cChecksum = 0;
    for (i=0; i<cSendBuffer[0]; i++)
    {
        cChecksum ^= cSendBuffer[i];
        TXC = 1;
```

```

UDR = cSendBuffer[i];

while (!TXC);

if (cSendBuffer[i] == 0xAA) // 如果数据域里面存在 AA 字节

//if there is a "0xAA" in the data field but not the command header,

//add a "0x00" follow the "0xAA", CL (command length) will not changed.

{

    TXC = 1;

    UDR = 0; // 在 AA 后面插入 00

    while (!TXC);

}

}

TXC = 1;

UDR = cCheckSum; // 发送校验字节

while (!TXC);

TXC = 1;

}

```

RFID 模块到主机传输协议

模块往主机发送的数据帧格式

Header	Length	Status	Data	CSUM
2 Byte	1 Byte	1 Byte	N Bytes	1 Byte

说明：

1. **Header:** 数据帧头表示一个数据帧的开始，这里固定为 0xAABB。
2. **Length:** 长度字包含了从命令字到校验字的数据长度。
3. **Status:** 状态字，表示指令执行结果状态，如果执行成功则返回命令字的原码，否则返回反码。比如，命令字是 0x19，如果执行成功，那么返回的状态字是 0x19，否则就是反码 0xe6。
4. **Data:** 包含了指令执行后的结果内容，可以为空。
5. **CSUM:** 校验字，其结果为从长度字到数据域最后一个字节逐字节异或的结果。

$$\text{CSUM} = \text{Length} \oplus \text{Command} \oplus \text{Response}[0] \oplus \text{Response}[1] \oplus \dots \oplus \text{Response}[n-1]$$

注意：如果在接收的数据域内出现“AA”，为了和数据帧头区分开，在收到这样的数据时会发现 AA 后面多了 2 个 00，那么这时候在接收代码里面把 00 去掉。

串口接收示例代码 Code example:

```
// This flag is set on USART Receiver buffer overflow
#define RXEN UCSRB.4
// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
    static unsigned char i;
    static unsigned char cReceivedData;
    static unsigned char cChecksum;
    char status;
    status=UCSRA;
    cReceivedData=UDR;

    if ((status & (FRAMING_ERROR | PARITY_ERROR | DATA_OVERRUN))==0)
    {
        if (g_bReceAA)
        {
            g_bReceAA = 0;
            if (0 != cReceivedData) // 第二个字节不是 00，说明是帧头 AABB
            {
                g_cReceNum = 0; // 数据长度置零
            }
        }
        else
        {
            if (0xAA == cReceivedData) // 收到 AA 字节
            {
                g_bReceAA = 1; // AA 标志位置 1
            }
            g_cReceBuf[g_cReceNum++] = cReceivedData; // 开始接收数据域

            if (g_cReceNum > g_cReceBuf[0]) // 接收完成
            {
                cChecksum = 0;
                for (i=0; i <= g_cReceBuf[0]; i++) // 计算异或校验
                {
                    cChecksum ^= g_cReceBuf[i];
                }
            }
        }
    }
}
```

```

    }
    if (0 == cChecksum)          // 比对结果正确
    {
        g_bReceCommandOk = 1;   // 命令成功标志位置 1
        RXEN = 0;
    }
    g_bReceAA = 0;
    g_cReceNum = 0;
}
if (g_cReceNum >= sizeof(g_cReceBuf))
{
    g_cReceNum = 0;
    g_bReceAA = 0;
}
}
}
}
//-----

```

如有疑问请及时和我们取得联系。

非常感谢您使用我们的产品！欢迎您给我们多提意见和建议，我们将不断的改进我们的产品和服务！同时，我们也保留改进和更新产品的权利。

谢谢！

订货信息	
生产商	北京易火眼科技有限公司
订货型号	YHY521X, YHY522, YHY502CTG, YHY502CG
产品名称	RFID 读写模块
订货电话:	010-80128328
邮箱:	info@ehuoyan.com
QQ	销售专员: 381785816 , 技术支持: 1443536237
网址:	http://www.ehuoyan.com/

Since 2008, 专注RFID技术

