

---

# LDAS200-H 智能采集终端 产品说明书

天津朗思世纪科技发展有限公司

---

# 目 录

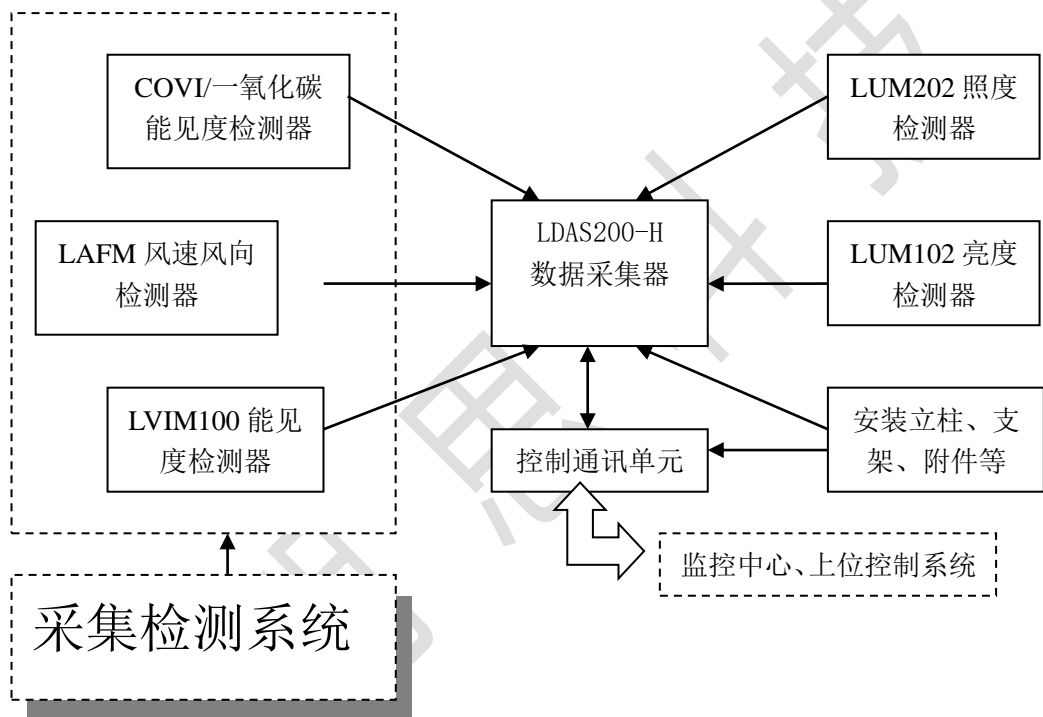
|                           |    |
|---------------------------|----|
| 一. LDAS200-H 产品介绍.....    | 3  |
| 1.1 系统组成.....             | 3  |
| 1.2 系统特点.....             | 3  |
| 1.3 LDAS200-H 智能监测终端..... | 4  |
| 二. 智能采集终端操作指南.....        | 5  |
| 2.1 智能采集终端结构说明.....       | 5  |
| 2.2 指示灯说明.....            | 5  |
| 2.3 智能采集终端接线说明.....       | 6  |
| 2.4 接线接口说明.....           | 6  |
| 三. 上位机介绍.....             | 7  |
| 3.1 连接线缆.....             | 7  |
| 3.2 设置.....               | 7  |
| 四. 云平台介绍.....             | 8  |
| 五. 通讯协议.....              | 10 |
| 5.1 命令：F5.....            | 10 |
| 5.2 返回：F5.....            | 10 |
| 5.3 各气象要素的量程和单位如下：.....   | 10 |
| 5.4 GPRS 主动上传.....        | 10 |
| 5.5 参数设置协议.....           | 11 |
| 六. CRC 计算.....            | 11 |

# 一. LDAS200-H 产品介绍

LDAS200-H 智能监测终端系统是一种高智能、免维护、一体化的自动监测系统。能够精确、实时的将隧道 COVI 一氧化碳能见度检测器、LAFM 风速检测器、LUM102 亮度检测器、LUM202 照度检测器的数据信息及仪器状态发布到监测中心。广泛适用于高速公路、公路、桥梁、机场、农林、水电和港口等各个领域，能在各种气候环境条件下，提供数据监测服务。LDAS200-H 系统采用模块化设计，可根据不同使用要求自由组合，并实现组网运行。

## 1.1 系统组成

LDAS200-H 智能监测终端系统包含传感器、数据采集器和控制通讯单元等，如下图所示：



LDAS200-H 智能监测终端系统组成示意

## 1.2 系统特点

- ◆ 智能监测终端控制器，采用 COTEX-A3 高性能低功耗 32 位 ARM 处理器；
- ◆ 2.2 寸真彩液晶屏显示，读数准确直观，操作简单易用；
- ◆ 可配置监测 COVI、LAFM、LUM102、LUM202；
- ◆ RS\_485/GPRS 接口通讯（选择其一）与监控中心连接；
- ◆ 电源供电采用模块化设计，满足超宽电压输入，供电稳定、可靠，符合工业级产品技术要求；
- ◆ EMC 应用，供电更稳定，低纹波，低噪声，增加数据采集的精确度和准确度；
- ◆ 内部集成防雷设计，提升了系统在野外工作的可靠性、安全性，可在极其恶劣的环境下稳定工作；

- ◆ 配备计算机端监测分析软件，操作简单，分析能力强；
- ◆ 安装简易灵活；

### 1.3 LDAS200-H 智能监测终端

LDAS200-H 型采集器作为监测系统的核心，具有实时性能佳，可靠性强，工作稳定，采集精度高、处理速度快、数据传输方便以及多种连接方式等优点。

软件平台的开发基于 windows 平台，采用面向对象的编程工具和分析工具，软件产品成熟而先进，具有良好的扩充性。操作界面友好、安全可靠、实时性高的特点，支持有线和无线通信，完成数据采集、处理、通信、存储和报文上传等功能。软件采用面向对象的设计思想，软件界面人性化设计，用户可自定义显示模式，满足专业人士操作习惯。

核心数据采集器采用嵌入式系统设计，能直观方便的查看相关仪器数据及状态，路面状况及方便现场调试，同时通过多种本地和远程通信方式，例如：本地通讯（RS\_232、RS\_485/422）、远程有线通讯（以太网）和远程无线通讯（GPRS、CDMA-1X,4G 等）。

实时和主控制室进行通信，实时发送当前气象数据，并可无线传输，供相关人员分析使用。LDAS200-H 型采集器采用 GPRS 无线传输，加大了数据传输的灵活性，灵活多样的接口设计实现了各要素的灵活安装与即插即用的要求，与远程业务中心数据交换可采用无线通讯，免去了现场布线的繁琐，减少投入成本。模块化的设计使得产品的维护检测工作变的更加容易。

#### 1.3.1 采集器技术指标

|          |                                   |
|----------|-----------------------------------|
| 处理器      | SAMSUNG S3C2416XH-40 32 位 ARM 处理器 |
| 数据存储     | 标准 32kbit                         |
| 防雷设计     | 内部集成防雷设计                          |
| 采样速率     | 最快 250ms                          |
| 传感器输入    | 1 路 RS_485                        |
| 上传周期     | 2 秒-10 分钟可调                       |
| 工作环境     | 温度：-50—+70℃ 湿度：0—100%RH（无水汽凝结）    |
| 保存环境     | 温度：-50—+150℃ 湿度：0—100%RH（无水汽凝结）   |
| 通讯接口     | 标准 RS_232/RS_485，以太网接口            |
| 无线通讯     | GPRS、CDMA 可选，天线外置，全天候，带吸盘         |
| 供电       | 9—28 VDC                          |
| 功耗       | MAX：<3W                           |
| 显示形式     | 4.5 寸触摸控制液晶屏                      |
| 外形尺寸(mm) | 200(L)*130(W)*80(H)               |
| 重量       | 1.25kg                            |

注：上述参数为标准型号参数，如有特殊要求，可定制。

## 二. 智能采集终端操作指南

### 2.1 智能采集终端结构说明

LDAS200-H 采集器是带彩色液晶显示，配合 COVI、风速仪、亮度照度仪使用的实时性采集模块，具有实时性能佳、可靠性强、工作稳定、处理速度快、数据传输方便等优点。

LDAS200-H 采集器采用嵌入式系统设计，液晶屏实时显示当前配套仪器的数据值和仪器运行状态，同时通过本地和远程通信方式将数据上传上位机或服务器。



图 3-1

### 2.2 指示灯说明

- 1.POWER 灯: 红色电源指示灯，上电后常亮
- 2.RUN 灯:绿色设备运行指示灯，当设备正常运行时，以 1S 频率闪烁
- 3.LINK 灯: 当网络连接好时 Link 为绿色，可用于判断设备是否和服务器建立通讯链路。

## 2.3 智能采集终端接线说明



## 2.4 接线接口说明

| 端子号         | 说明  | 备注   |
|-------------|---|--|
| RS485 通讯 A1 | RS485 通讯输入 1 的 A 线                                    | 波特率为 9600, 数据位为 8, 校验位为 1, 停止位为 None, 与设备端连接, 读取设备端数据; <b>注意不可接入电源线</b>        |
| RS485 通讯 B1 | RS485 通讯输入 1 的 B 线                                    |  |
| RS485 通讯 A2 | RS485 通讯输入 2 的 A 线                                    | 波特率为 19200, 数据位为 8, 校验位为 1, 停止位为 None, 与上位机软件连接, 通过上位机软件进行设置; <b>注意不可接入电源线</b> |
| RS485 通讯 B2 | RS485 通讯输入 2 的 B 线                                    |  |
| 电源输入        | 9-24VDC DC 插座和接线端子输入任选一种; 接线端子 (DC+/DC-): 3.08mm2P 端子 | DC 插座为内正外负; DC- 接直流负极, DC+ 接入直流正极  |
| GPRS 天线接口   | 外接天线  |  |

## 三. 上位机介绍

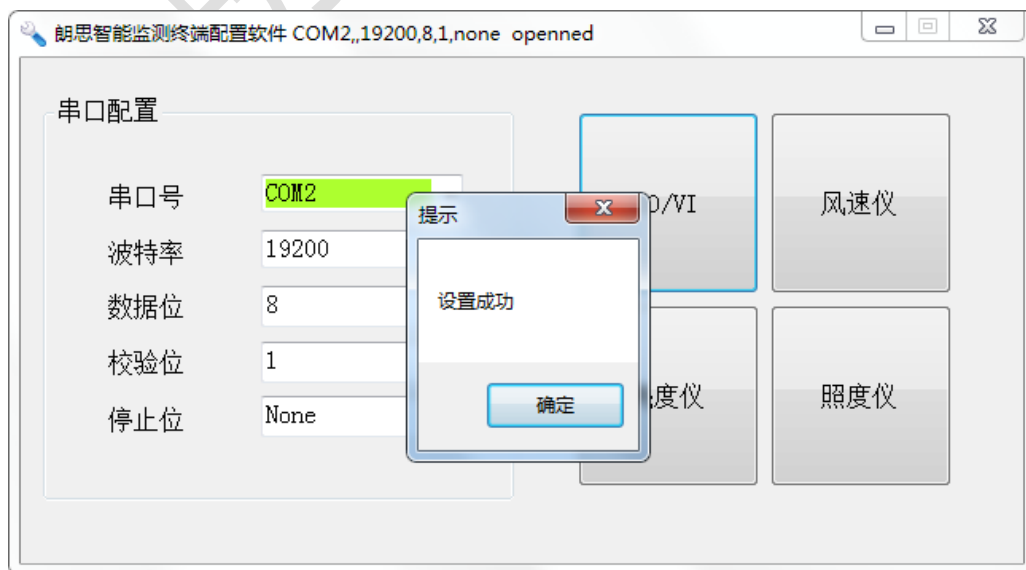
### 3.1 连接线缆

将电脑用 RS485 通讯线与设备 RS485 通讯 A2、B2 连接好之后，选择并双击“监测终端配置软件.exe”软件，将软件打开后根据所连接串口进行串口配置；串口号：调试电脑所使用的的串口号，波特率选择 19200，数据位 8，校验位 1，停止位 None



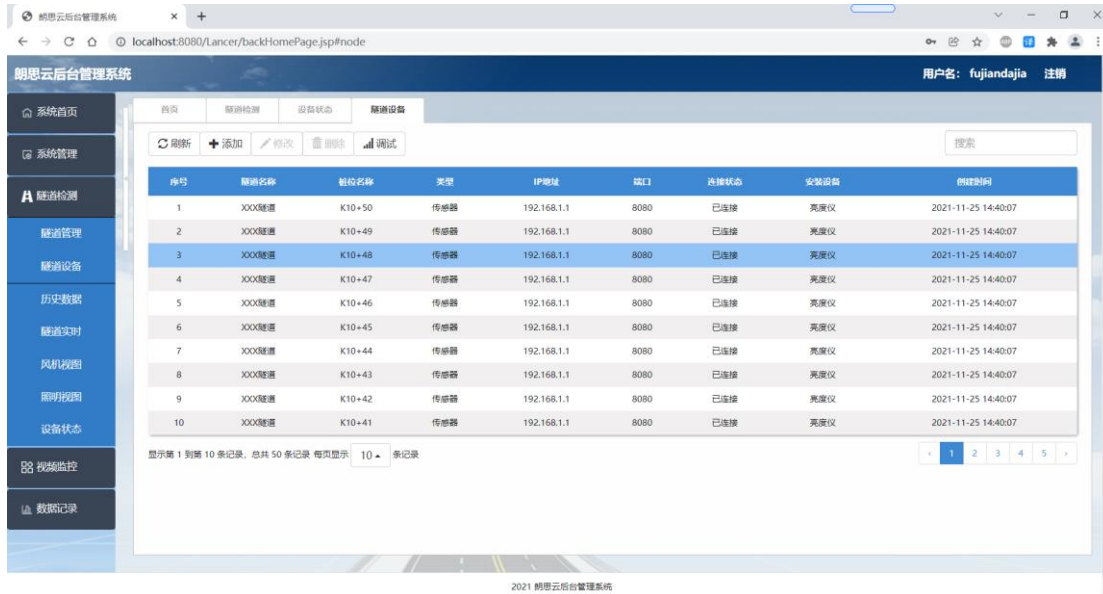
### 3.2 设置

根据所使用设备点击右侧选择 CO/VI、风速仪、亮度仪、照度仪按键，对设备进行配置。配置完成后软件提示：设置成功。设备显示窗口完成相应设置。

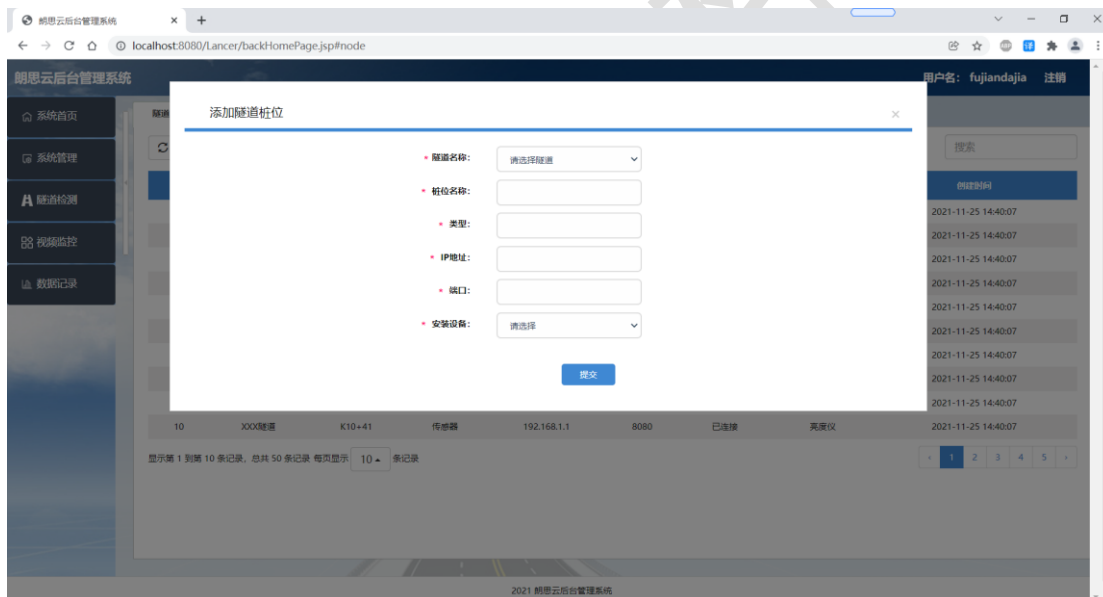


## 四. 云平台介绍

浏览器输入 <http://39.105.34.43:8888/Lancer> 登录，登录成功之后点击隧道检测下隧道设备标签，打开隧道设备页面，如下



1. 点击添加按钮，根据设备信息添加设备



2. 填入设备的信息，点击添加，完成设备添加。添加完成之后，就可以在设备状态页面，查看设备的具体状态。



---

朗思科技

## 五. 通讯协议

### 5.1 命令：F5

|    |    |    |    |    |    |     |    |    |    |    |    |    |    |    |       |       |    |    |
|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|----|-------|-------|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06  | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15    | 16    | 17 | 18 |
| AA | 55 | 05 | 00 | 0A | F5 | 00  | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00    | 00    | 00 | 5D |
| 帧头 | 帧尾 | 类别 | 备用 | 长度 | 命令 | 数据位 |    |    |    |    |    |    |    |    | CRC高位 | CRC低位 | 帧尾 |    |

表 5-1（表中数据第二行为 16 进制表示）

### 5.2 返回：F5

|    |    |    |    |    |    |       |    |    |    |    |    |    |    |    |       |       |    |    |
|----|----|----|----|----|----|-------|----|----|----|----|----|----|----|----|-------|-------|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06~31 |    |    |    |    |    |    |    |    | 96    | 97    | 98 |    |
| AA | 55 | 05 | 00 | 1A | F5 | 00    | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00    | 00    | 00 | 5D |
| 帧头 | 帧尾 | 类别 | 备用 | 长度 | 命令 | 数据位   |    |    |    |    |    |    |    |    | CRC高位 | CRC低位 | 帧尾 |    |

表 5-2（表中数据第二行为 16 进制表示）

06~95 位为数据位

### 5.3 各气象要素的量程和单位如下：

| 序号 | 数据位<br>(06-95) | 测量变量 | 数据类型  | 长度<br>(byte) | 量程  |       |     |
|----|----------------|------|-------|--------------|-----|-------|-----|
|    |                |      |       |              | 最小值 | 最大值   | 单位  |
| 0  | 8-11           | 软件版本 | float | 4            |     |       |     |
| 1  | 12-15          | CO 值 | float | 4            | 0   | 300   |     |
| 2  | 16-19          | VI 值 | float | 4            | 0   |       |     |
| 3  | 20-23          | 亮度值  | float | 4            | 0   | 6500  |     |
| 4  | 24-27          | 照度值  | float | 4            | 0   | 20000 |     |
| 5  | 28-31          | 风速值  | float | 4            | -30 | 30    | M/S |

### 5.4 GPRS 主动上传

上传时间：2s/次

上传内容：01 03 14 xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx CRCH CRCL

|     |     |      |             |       |             |             |             |
|-----|-----|------|-------------|-------|-------------|-------------|-------------|
| 地址码 | 功能码 | 数据长度 | 寄存器<br>0x00 | ..... | 寄存器<br>0x09 | CRC 校<br>验高 | CRC 校<br>验低 |
| 01  | 03  | 14   | xx xx       |       | xx xx       | xx          | xx          |

|       |      |      |
|-------|------|------|
| 寄存器地址 | 数据定义 | 数据类型 |
|-------|------|------|

|            |                        |   |
|------------|------------------------|---|
| 0x00(16 位) | CO 值                   | Float(低 16 位在前 高 16 位在后)                            |
| 0x01(16 位) |                        |   |
| 0x02(16 位) | VI 值                   | Float(低 16 位在前 高 16 位在后)                            |
| 0x03(16 位) |                        |   |
| 0x04(16 位) | 风速<br>-30.00-30.00 m/s | Float(低 16 位在前 高 16 位在后)                            |
| 0x05(16 位) |                        |   |
| 0x06(16 位) | 亮度值<br>0-6500.0        | Float(低 16 位在前 高 16 位在后)                            |
| 0x07(16 位) |                        |   |
| 0x08(16 位) | 照度值<br>0-20000.0       | Float(低 16 位在前 高 16 位在后)                            |
| 0x09(16 位) |                        |   |
| 0x0A(16 位) | 设备状态码                  | Integer<br>00 正常<br>01 通讯故障<br>02 模拟量故障<br>03 传感器故障 |

## 5.5 参数设置协议

命令 F7

|    |    |    |    |    |    |     |    |    |    |    |    |    |    |    |    |       |       |    |
|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|----|----|-------|-------|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06  | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16    | 17    | 18 |
| AA | 55 | 05 | 00 | 0A | F7 | 00  | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00    | 00    | 5D |
| 帧头 | 帧尾 | 类别 | 备用 | 长度 | 命令 | 数据位 |    |    |    |    |    |    |    |    |    | CRC高位 | CRC低位 | 帧尾 |

## 5.6 命令举例

实时上传命令

发送: AA 55 05 00 0A F5 00 00 00 00 00 00 00 00 00 00 60 63 5D

返回: AA 55 05 00 5A F5 0A 01 01 0D 04 02 40 2C CC CD 43 18 19 9A 41 8E 66 66 42 2D 99 9A 00 00 00 00 00 00 00 00 44 62 79 9A 45 9C 40 00 41 FE 66 66 00 00 00 00 00 00 00 00 40 80 00 00 00 00 00 00 00 00 00 00 00 00 00 00 CA 01 5D

## 六. CRC 计算

CRC 计算依据以下规则:

Norm: CRC-CCITT

Polynomial:  $1021h = x^{16} + x^{12} + x^5 + 1$

Start value: FFFFh

C 语言举例如下:

```
/******
```

```
Function: 16 bit CRC-CCITT calculation
```

---

-----  
Call: calc\_crc(unsigned short crc\_buff, unsigned char input)  
-----

Response: Newly calculated 16 bit CRC checksum  
-----

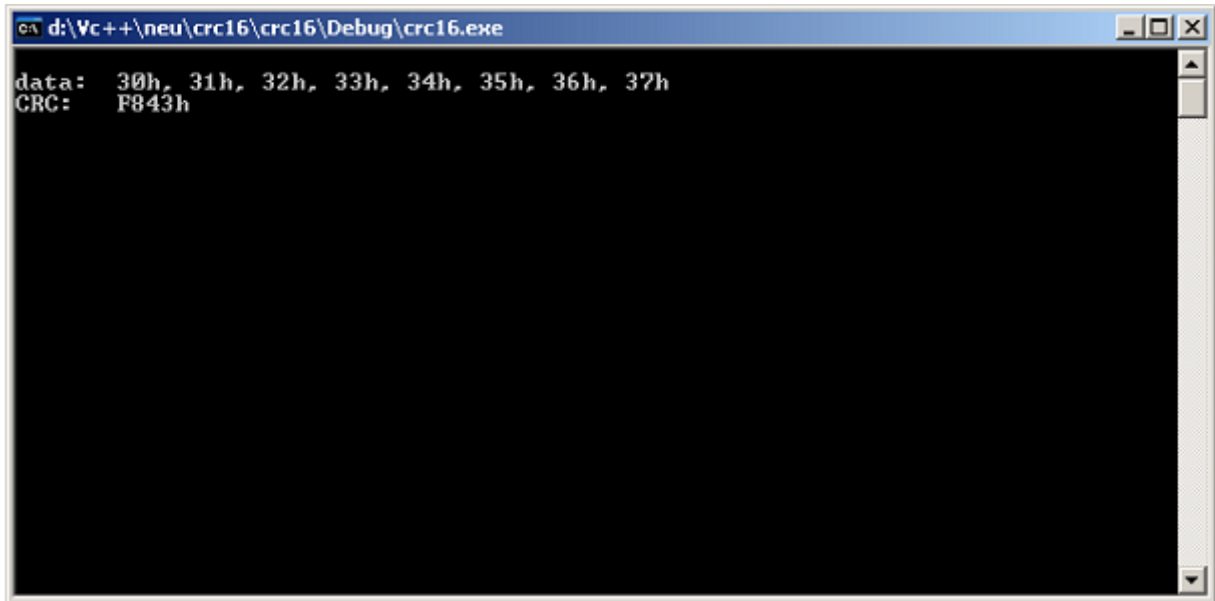
Description: Calculates the checksum for 'input' in accordance with the CRC polynomial  $x^{16} + x^{12} + x^5 + 1$ .

'crc\_buff' is the previously calculated checksum. This must be set to 0xFFFF at the beginning of a test sequence.

-----/

```
unsigned short calc_crc(unsigned short crc_buff, unsigned char input)
{
    unsigned char i;
    unsigned short x16; // we'll use this to hold the XOR mask
    for (i=0; i<8; i++)
    {
        // XOR current D0 and next input bit to determine x16 value
        if( (crc_buff & 0x0001) ^ (input & 0x01) )
            x16 = 0x8408;
        else
            x16 = 0x0000;
        // shift crc buffer
        crc_buff = crc_buff >> 1;
        // XOR in the x16 value
        crc_buff ^= x16;
        // shift input for next iteration
        input = input >> 1;
    }
    return(crc_buff);
}
// ***** MAIN *****
void main(void)
{
    // example: CRC for 8 Bytes
    unsigned char values[8] =
{0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37};
    // initialise startvalue FFFFh
    unsigned short crc = 0xFFFF;
    // calculation
    for(int n = 0; n < 8; n++)
    {
        crc = calc_crc(crc, values[n]);
    }
    // output
```

```
printf("\ndata: 30h, 31h, 32h, 33h, 34h, 35h, 36h, 37h");  
printf("\nCRC: %04Xh\n", crc);  
}
```



```
d:\Vc++\neu\crc16\crc16\Debug\crc16.exe  
data: 30h, 31h, 32h, 33h, 34h, 35h, 36h, 37h  
CRC: F843h
```

C#.net 举例:

此函数返回 CRC 验证结果一个数组，高字节在前，低字节在后。

```
public static byte[] CRC (byte[] data)  
{  
    byte CRC16Lo = 0xFF;  
    byte CRC16Hi = 0xFF;  
    byte CL = 0x08;  
    byte CH = 0x84;  
    byte saveHi = 0;  
    byte saveLo = 0;  
    byte[] returnData = new byte[2];  
    for (int i = 0; i <= data.Length - 1; i++)  
    {  
        CRC16Lo = (byte)(CRC16Lo ^ data[i]);  
        for (int j = 0; j <= 7; j++)  
        {  
            saveHi = CRC16Hi;
```

---

```
    saveLo = CRC16Lo;
    CRC16Hi = (byte)(CRC16Hi >> 1);
    CRC16Lo = (byte)(CRC16Lo >> 1);
    if ((saveHi & 0x01) == 0x01)
    {
        CRC16Lo = (byte)(CRC16Lo | 0x80);
    }
    if ((saveLo & 0x01) == 0x01)
    {
        CRC16Hi = (byte)(CRC16Hi ^ CH);
        CRC16Lo = (byte)(CRC16Lo ^ CL);
    }
}
}
returnData[0] = CRC16Hi;
returnData[1] = CRC16Lo;
return returnData;
}
```