

数字变送器模块 使用说明书

目录

| | |
|--------------------|----|
| 第一章 概述 | 1 |
| 1.1 技术规格..... | 1 |
| 第二章 安装配线..... | 2 |
| 2.1 注意事项 | 2 |
| 2.2 传感器连接..... | 2 |
| 2.3 接线示意图..... | 3 |
| 第三章 串口通讯..... | 5 |
| 3.1 自定义协议 | 5 |
| 3.2 MODBUS制约 | 9 |
| 3.3 指示灯含义 | 12 |
| 第四章 在线升级程序..... | 13 |

第一章 概述

WD200 系列是天津市丽景微电子有限公司开发研制的数字信号处理的变送器模块，可支持1~8路称重传感器接入，其中WD200-4可支持最多四路称重传感器接入，WD200-8可支持最多八路称重传感器接入，具有RS485或RS232通讯接口，标准MODBUS-RTU和自定义两种通讯协议。

1.1 技术规格

- 电源：DC12~24V
- 传感器激励电压：DC5V
- 工作温度：-10℃~+40℃
- 最大湿度：90%R.H不可结露
- 功耗：最大10W
- 精度：0.02%，满足C3等级称重精度要求
- 物理尺寸：93.3mm×87.6mm(WD200-4)
123mm×88mm(WD200-8)

第二章 安装配线

2.1 注意事项

- 请注意电源正负极性，不要接反。
- 不要用力拖拉导线，防止脱落。
- 注意防潮防水，防剧烈震动。

2.2 传感器连接

传感器连接端子各端口分配为：

| 端口 | EXC+ | EXC- | SIG+ | SIG- |
|-------|-------|-------|------|------|
| 四线制接线 | 激励电源正 | 激励电源负 | 信号正 | 信号负 |

注意事项：

- 由于传感器输出信号是对电子噪声比较敏感的模拟信号，因此传感器接线应采用屏蔽电缆，而且与其它电缆分开铺设，尤其是要远离交流电源；
- 对于传输距离短且温度变化不大的场合或精度要求不高的场合可以选择四线制传感器；但是对于传输距离远或精度要求高的应选择六线制传感器；
- 对于多传感器并联的应用，要保证各传感器的灵敏度(mV/V)一致。

2.3 接线示意图

| | | | | | | |
|----|-------------|------|--|------|------|---|
| 23 | GND | | | 第一路 | EXC+ | 1 |
| 24 | 485-B | | | | SIG+ | 2 |
| 25 | 485-A | | | | SIG- | 3 |
| 26 | 电源正(12-24V) | | | | EXC- | 4 |
| 27 | 电源负(0V) | | | 第二路 | EXC+ | 5 |
| | | SIG+ | | | 6 | |
| | | SIG- | | | 7 | |
| | | EXC- | | | 8 | |
| | | 第三路 | | EXC+ | 9 | |
| 17 | 常开 | | | SIG+ | 10 | |
| 18 | 公共端 | | | SIG- | 11 | |
| 19 | 常闭 | | | EXC- | 12 | |
| 20 | 常开 | 第四路 | | EXC+ | 13 | |
| 21 | 公共端 | | | SIG+ | 14 | |
| 22 | 常闭 | | | SIG- | 15 | |
| | | | | EXC- | 16 | |

图二：WD200-4

| | | | | | | |
|----|-------------|-------|-----|------|------|----|
| 39 | GND | | | 第一路 | EXC+ | 1 |
| 40 | 485-B | | | | SIG+ | 2 |
| 41 | 485-A | | | | SIG- | 3 |
| 42 | 电源正(12-24V) | | | | EXC- | 4 |
| 43 | 电源负(0V) | | | 第二路 | EXC+ | 5 |
| | | | | | SIG+ | 6 |
| | | | | | SIG- | 7 |
| | | | | | EXC- | 8 |
| 33 | 常开 | 开出第一路 | | 第三路 | EXC+ | 9 |
| 34 | 公共端 | | | | SIG+ | 10 |
| 35 | 常闭 | | | | SIG- | 11 |
| | | EXC- | | | 12 | |
| 36 | 常开 | 开出第二路 | | 第四路 | EXC+ | 13 |
| 37 | 公共端 | | | | SIG+ | 14 |
| 38 | 常闭 | | | | SIG- | 15 |
| | | EXC- | | | 16 | |
| 25 | EXC- | 第七路 | 第五路 | EXC+ | 17 | |
| 26 | SIG- | | | SIG+ | 18 | |
| 27 | SIG+ | | | SIG- | 19 | |
| 28 | EXC+ | | | EXC- | 20 | |
| 29 | EXC- | 第八路 | 第六路 | EXC+ | 21 | |
| 30 | SIG- | | | SIG+ | 22 | |
| 31 | SIG+ | | | SIG- | 23 | |
| 32 | EXC+ | | | EXC- | 24 | |

图二：WD200-8

第三章 串口通讯

可以通过串行接口RS485或RS232(需订货时说明)实现与上位机的通讯, 串口协议为自定义协议和MODBUS协议两种。

3.1 自定义协议

1. 规约工作方式: 命令方式
2. 数据格式: 8位数据位、1位停止位、无校验(出厂默认)
3. 波特率: 9600(出厂默认)
4. 编码标准: 十六进制
5. 超过一字节的数据小端对齐(即16位与32位数据都是低字节在前)

主机发送帧格式

| 地址 | 指令码 | 数据长度 | 数据 | CRC校验码 |
|-----|-----|------|-----|--------|
| 1字节 | 1字节 | 2字节 | N字节 | 2字节 |

注: 地址=设备地址

数据=数据缓存地址 (2字节)+数据长度 (2字节)

数据长度: 占用2字节, 表示后续数据的长度(不包括CRC)

CRC校验码: 占用2字节

从机回复帧格式

| 起始符 | 地址 | 指令 | 数据长度 | 数据 | CRC校验码 |
|-----|-----|-----|------|-----|--------|
| 55H | 1字节 | 1字节 | 2字节 | N字节 | 2字节 |

注: 地址=设备地址

数据长度: 占用2字节, 表示后续数据的长度(不包括CRC)

CRC校验码: 占用2字节

指令:

| 指令码 | 说明 | 备注 |
|------|-------------|--|
| 0x02 | 读设备MID | MID是设备唯一标识, 出厂时设置, 8位16进制数 |
| 0x03 | 读设备MADR | 读通讯地址, 目的地址使用广播地址, 参数为MID |
| 0x04 | 写设备MADR | 写通讯地址, 目的地址可以使用广播地址 |
| 0x3C | 读数据 | 读取内存变量 |
| 0x3D | 写数据 | 修改内存变量 |
| 0x3E | 清零 | 设零点修正数值为当前原始重量值 |
| 0x41 | 保存数据变更 | 将需要保存的内存数据写入Flash |
| 0x50 | 设置新校准点 | 以给定重量生成新校准点, 原有校准数据作废 |
| 0x51 | 插入新校准点 | 以给定重量生成新校准点, 插入校准数据表中 |
| 0x54 | 分度值设置 | |
| 0x55 | 开机自动清零/零点跟踪 | 开机清零范围: 1-10分别为满量程的1%-10%; 零点跟踪范围: 1-10分别为所设分度值的1-10倍 |
| 0x56 | 继电器控制 | |

| | | |
|------|------|---------------------------|
| 0x3f | 規約切换 | 0: 自定义协议; 1: MODBUS-RTU协议 |
| 0xfc | 软重启 | 执行该指令后设备延时20s后重启 |

内存变量地址表:

| 地址 | 说明 | 备 注 |
|---|---------|-----------------------|
| 0x0A | AD | 传感器实时采集AD值 |
| 0x0C | 重量 | 重量 = 原始重量 - 零点修正值 |
| 0x0D | 净重 | 净重 = 重量 - 扣重 |
| 0x0B | 原始重量 | 由AD直接计算得到的重量值 |
| 0x0F | 零点修正值 | 对零点的小范围数据修正, 通过清零操作设置 |
| 0x11 | 扣重 | |
| 0x12 | 单重 | 用于计算数量, 未使用 |
| 0x13 | 最大量程 | |
| 0x16 | 分度值 | |
| 0x15 | 零点AD值 | 校准数据, 用于计算原始重量 |
| 0x17 | 校准数据重量值 | 校准数据, 共10个32位数 |
| 0x21 | 校准数据AD值 | 校准数据, 共10个32位数 |
| 第二路到第八路对应地址在上路的基础上依次增加0x21(即第二路的AD值为0x2B) | | |

命令举例:

1. 读取设备ID号 0x02

上位机读取地址为0x01终端的ID号(0xFFFFFFFF)

上位机发: 0x01 0x02 0x00 0x00 crc

终端返回: 0x55 0x01 0x02 0x04 0x00 0xFF 0xFF 0xFF 0xFF crc

说明: 目前设备终端的ID号全为0xFFFFFFFF

2. 读取设备通讯地址 0x03 (通过广播方式)

上位机获取ID为0xFFFFFFFF终端的地址(0x01)

上位机发: 0x00 0x03 0x04 0x00 0xFF 0xFF 0xFF 0xFF crc

终端返回: 0x55 0x01 0x03 0x01 0x00 0x01 crc

3. 修改设备通讯地址 0x04

上位机将ID为0xFFFFFFFF通讯地址为0x01的终端的地址改为0x02

上位机发: 0x01 0x04 0x01 0x00 0x02 crc

终端返回: 0x55 0x02 0x04 0x01 0x00 0x02 crc

上位机发出的帧也可以使用广播地址

上位机发: 0x00 0x04 0x05 0x00 0xFF 0xFF 0xFF 0xFF 0x02 crc

4. 读数据 0x3C

上位机读取地址为0x01终端的重量值(变量地址0x0C), 重量值为0x10000

上位机发: 0x01 0x3C 0x04 0x00 0x0C 0x00 0x01 0x00 crc

终端返回: 0x55 0x01 0x3C 0x04 0x00 0x00 0x00 0x01 0x00 crc

注: 其中数据部分前两字节是起始地址, 后两字节是数据长度(字), 两部分

组成一个段。一个帧可以包含若干个这样的数据段，返回帧会按顺序填充数据。

5. 写数据 0x3D

上位机将地址为0x01终端的扣重值（变量地址0x11）改为0x10000

上位机发：0x01 0x3D 0x08 0x00 0x11 0x00 0x01 0x00 0x00 0x00 0x01 0x00 crc

终端返回：0x55 0x01 0x3D 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

注：其中数据部分前两字节是起始地址，后续两字节是数据长度(字)，再后面跟随要写入的数据，三部分组成一个段。一个帧可以包含若干个这样的数据段。

6. 清零 0x3E

上位机对地址为0x01终端清零，终端返回清零后的零点修正值

上位机发：0x01 0x3E 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

终端返回：0x55 0x01 0x3E 0x04 0x00 0x00 0x01 0x00 0x00 0x00 0x00 0x00

7. 保存数据变量 0x41

上位机发：0x01 0x41 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

终端返回：0x55 0x01 0x41 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

注：保存至Flash的数据包括

- 1)通讯地址
- 2)零点修正值
- 3)扣重
- 4)单重
- 5)最大量程
- 6)分度值
- 7)零点AD值
- 8)校准数据重量值
- 9)校准数据AD值
- 10)开机清零/零点跟踪设置

8. 重新设置校准点 0x50

上位机对地址为0x01终端校准零点

上位机发：0x01 0x50 0x04 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

终端返回：0x55 0x01 0x50 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

上位机对地址为0x01终端校准重量1000g(0x186A0)

上位机发：0x01 0x50 0x04 0x00 0xA0 0x86 0x01 0x00 0x00 0x00 0x00 0x00

终端返回：0x55 0x01 0x50 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00

注：重设校准点会将原有的校准数据全部作废(不包括零点AD值)，即校准后校准数据数组中只有给定重量值对应的一组校准数据。给定重量值为0时，只校准零点AD值。

9. 插入新校准点 0x51

上位机对地址为0x01终端校准重量1000g (0x186A0)

上位机发: 0x01 0x51 0x04 0x00 0xA0 0x86 0x01 0x00 crc

终端返回: 0x55 0x01 0x51 0x00 0x00 crc

注: 1.插入校准点的指令格式与0x50指令完全相同, 区别在于不会清除原有校准数据。

2.如果原有校准数据不满8组, 则将新校准数据直接插入适当的位置; 已满8组, 则选择AD值最接近的一组, 将其替换。

3.给定重量值为0时, 只校准零点AD值。

10. 设置最大量程

上位机对地址为0x01终端校准重量5000g (0x7A120)

上位机发: 0x01 0x3D 0x08 0x00 0x13 0x00 0x01 0x00 0x20 0xA1 0x07 0x00 crc

终端返回: 0x55 0x01 0x3D 0x00 0x00 crc

说明: 0x13 0x00地址首地址; 0x01 0x00寄存器数量;

0x20 0xA1 0x07 0x00 寄存器值

11. 分度值设置

上位机发: 0x01 0x54 0x01 0x00 xx(0x01、0x2、0x5、0x0A、0x14、0x32) crc

终端返回: 0x55 0x01 0x54 0x00 0x00 crc

说明: 分度值: xx可设置为0x01、0x2、0x5、0x0A、0x14、0x32, 分别为1、2、5、10、20、50单位为克

12. 继电器控制

上位机发: 0x01 0x56 0x01 0x00 xx(0x00、0x1、0x2、0x03) crc

终端返回: 0x55 0x01 0x56 0x01 0x00 xx(0x00、0x1、0x2、0x03) crc

说明:

分度值: xx可设置为0x00、0x1、0x2、0x03, 0x00: 继电器1、2处于复位状态;

0x01: 继电器1处于动作状态、继电器2处于复位状态;

0x02: 继电器2处于动作状态、继电器1处于复位状态;

0x03: 继电器1、2处于动作状态。

13. 开机自动清零/零点跟踪使能设置

上位机发: 0x01 0x55 0x02 0x00 xx(开机自动清零) xx (零点跟踪) crc

终端返回: 0x55 0x01 0x55 0x02 0x00 xx(开机自动清零) xx (零点跟踪) crc

说明:

开机清零: 0x00表示开机清零功能关闭,

0x01-0x0A表示开机清零功能打开(清零范围为满量程的百分之1至百分之10)

零点跟踪: 开机2s时间内, 重量值在零点跟踪范围内, 重量值归零。

0x00表示零点跟踪功能关闭, 0x01-0x0A表示零点功能打开(零点跟踪范围为所设分度值的1-10倍)

如果开机清零/零点跟踪值不在0x00-0x0A内则不生效, 返回原来设置值。(可用于

读取设备中保存值)

14. 规约切换

上位机发：0x01 0x3f 0x01 0x00 0x01 crc 终端返回：无

15. 读版本号

上位机发：0x01 0x01 0x00 0x00 CRC

终端返回：0x55 0x01 0x01 0x04 0x00 xx xx xx xx crc

16. 修改波特率

上位机对地址为0x01终端校准波特率19200(0x4B00)

上位机发：0x01 0x0C 0x04 0x00 0x00 0x4B 0x00 0x00 0x3C 0x55

终端返回：因为波特率改变所以回复数据无意义。

17. 修改校验位

上位机发送修改偶校验的报文(0：无校验位；1：偶校验；2：奇校验)

上位机发：0x01 0x0D 0x01 0x00 0x01 0x8B 0x6C

终端返回：0x55 0x01 0x0D 0x01 0x00 0x01 0x8B 0x6C

18. 远程重启

上位机对地址为0x01终端进行远程重启

上位机发：0x01 0xfc 0x00 0x00 0xC1 0xE8

3.2 MODBUS规约

WD200系列Modbus协议为RTU通讯模式。

数据格式(出厂设置)：波特率：9600

数据位：8

停止位：1

校验位：无校验

功能码：0x03：保持寄存器(可读可写) 0x04：输入寄存器(只读)

0x05：写单个线圈(只写) 0x06：写单个功能寄存器(只写)

0x10：连续写多个功能寄存器(只写)

Modbus通讯参数列表

表一：输入寄存器

| 地址编码 | 数据域 | 备注 |
|------|-----------|----|
| 0 | AD值高16位 | |
| 1 | AD值低16位 | |
| 2 | 毛重高16位 | |
| 3 | 毛重低16位 | |
| 4 | 净重高16位 | |
| 5 | 净重低16位 | |
| 6 | 软件版本号高16位 | |
| 7 | 软件版本号低16位 | |
| 8 | 硬件ID高16位 | |
| 9 | 硬件ID低16位 | |

表二：保持寄存器

| 地址编码 | 数据域 | 备注 |
|------|--------------------------|-----|
| 10 | 最大量程高16位 | 连续写 |
| 11 | 最大量程低16位 | |
| 12 | 分度值，在0/1/2/5/10/20/50中取值 | |
| 13 | 保留，未使用 | |
| 14 | 零点追踪，取值范围0-10 | |
| 15 | 开机清零，取值范围0-10 | |

表三：功能寄存器

| 地址编码 | 数据域 | 备注 |
|------|-------------------------------|-----|
| 16 | 校准零点，只能写入0xAA55 | |
| 17 | 清零，只能写入0xAA55 | |
| 18 | 校准点重量值高16位 | 连续写 |
| 19 | 校准点重量值低16位，写入后执行校准操作 | |
| 20 | 插入校准点重量值高16位 | 连续写 |
| 21 | 插入校准点重量值低16位，写入后执行校准操作 | |
| 22 | PWM校准点PWM值，数值1表示0x1fff，以此类推 | 连续写 |
| 23 | PWM校准点电流值，单位0.001mA，写入后执行校准操作 | |
| 24 | ModBus协议通讯地址 | |
| 25 | 串口奇偶校验，（0：无01：奇 02：偶） | |
| 26 | 波特率高16位 | 连续写 |
| 27 | 波特率低16位，写入后更改波特率 | |
| 28 | 扣重，只能写入0xAA55 | |
| 29 | 保存数据，只能写入0xAA55 | |
| 30 | 恢复部分数据的出厂设置，只能写入0xAA55 | |
| 31 | 切换通讯协议（0：自定义协议；1：MODBUS协议） | |
| 32 | 设备重启指令，只能写入0xAA55 | |

注意：读取2~8路时只需更改通讯地址即可。

表四：写单个线圈

| 地址编码 | 数据域 | 备注 |
|------|----------------------------------|----|
| 0 | 继电器1，写入0xFF00继电器动作，写入0x0000继电器复位 | |
| 1 | 继电器2，写入0xFF00继电器动作，写入0x0000继电器复位 | |

命令举例

1. 功能码0X04读输入寄存器（读取净重值）

| 请求 | | 回复 | |
|---------|--------------|---------|----------------|
| 数据(hex) | 说明 | 数据(hex) | 说明 |
| 01 | 地址 | 01 | 地址 |
| 04 | 功能码 | 04 | 功能码 |
| 00 | 输入寄存器起始地址高字节 | 04 | 字节数量 |
| 04 | 输入寄存器起始地址低字节 | 00 | 寄存器(0X04)数据高字节 |
| 00 | 读取寄存器数量高字节 | 01 | 寄存器(0X04)数据低字节 |
| 02 | 读取寄存器数量低字节 | 66 | 寄存器(0X05)数据高字节 |

| | | | |
|----|-------|----|----------------|
| 30 | CRC校验 | FC | 寄存器(0X05)数据低字节 |
| 0A | | 81 | CRC校验 |
| | | A5 | |

2. 功能码0X03读保持寄存器（读分度值）

| 请求 | | 回复 | |
|---------|--------------|---------|----------------|
| 数据(hex) | 说明 | 数据(hex) | 说明 |
| 01 | 地址 | 01 | 地址 |
| 03 | 功能码 | 03 | 功能码 |
| 00 | 保持寄存器起始地址高字节 | 02 | 字节数量 |
| 0C | 保持寄存器起始地址低字节 | 00 | 寄存器(0X0C)数据高字节 |
| 00 | 写保持寄存器数量高字节 | 02 | 寄存器(0X0C)数据低字节 |
| 01 | 写保持寄存器数量低字节 | 39 | CRC校验 |
| 44 | CRC校验 | 85 | |
| 09 | | | |

3. 功能码0X06写保持寄存器(写分度值)

| 请求 | | 回复 | |
|---------|--------------|---------|----------------|
| 数据(hex) | 说明 | 数据(hex) | 说明 |
| 01 | 地址 | 01 | 地址 |
| 06 | 功能码 | 06 | 功能码 |
| 00 | 保持寄存器起始地址高字节 | 00 | 保持寄存器起始地址高字节 |
| 0C | 保持寄存器起始地址低字节 | 0C | 保持寄存器起始地址低字节 |
| 00 | 写保持寄存器数据高字节 | 00 | 寄存器(0X0C)数据高字节 |
| 02 | 写保持寄存器数据低字节 | 02 | 寄存器(0X0C)数据低字节 |
| C8 | CRC校验 | C8 | CRC校验 |
| 08 | | 08 | |

4. 功能码0X10连续写寄存器(写最大量程)

| 请求 | | 回复 | |
|---------|----------------|---------|--------------|
| 数据(hex) | 说明 | 数据(hex) | 说明 |
| 01 | 地址 | 01 | 地址 |
| 10 | 功能码 | 10 | 功能码 |
| 00 | 寄存器起始地址高字节 | 00 | 保持寄存器起始地址高字节 |
| 0A | 寄存器起始地址低字节 | 0A | 保持寄存器起始地址低字节 |
| 00 | 写寄存器数量高字节 | 00 | 写寄存器数量高字节 |
| 02 | 写寄存器数量低字节 | 02 | 写寄存器数量低字节 |
| 04 | 字节数量 | 61 | CRC校验 |
| 00 | 寄存器(0X0A)数据高字节 | CA | |
| 4C | 寄存器(0X0A)数据低字节 | | |
| 4B | 寄存器(0X0B)数据高字节 | | |
| 40 | 寄存器(0X0B)数据低字节 | | |
| 85 | CRC校验 | | |
| 07 | | | |

| | | | |
|----|-------|--|--|
| 85 | CRC校验 | | |
| 07 | | | |
| | | | |

5. 功能码0X05写单个线圈(继电器1动作)

| 请求 | | 回复 | |
|---------|-----------|---------|-----------|
| 数据(hex) | 说明 | 数据(hex) | 说明 |
| 01 | 地址 | 01 | 地址 |
| 05 | 功能码 | 05 | 功能码 |
| 00 | 寄存器地址高字节 | 00 | 寄存器地址高字节 |
| 00 | 寄存器地址低字节 | 00 | 寄存器地址低字节 |
| FF | 写寄存器数值高字节 | FF | 写寄存器数量高字节 |
| 00 | 写寄存器数值低字节 | 00 | 写寄存器数量低字节 |
| 8C | CRC校验 | 8C | CRC校验 |
| 3A | | 3A | |

3.3 指示灯含义

绿灯：运行指示


白灯：接收到正确报文(Modbus协议)

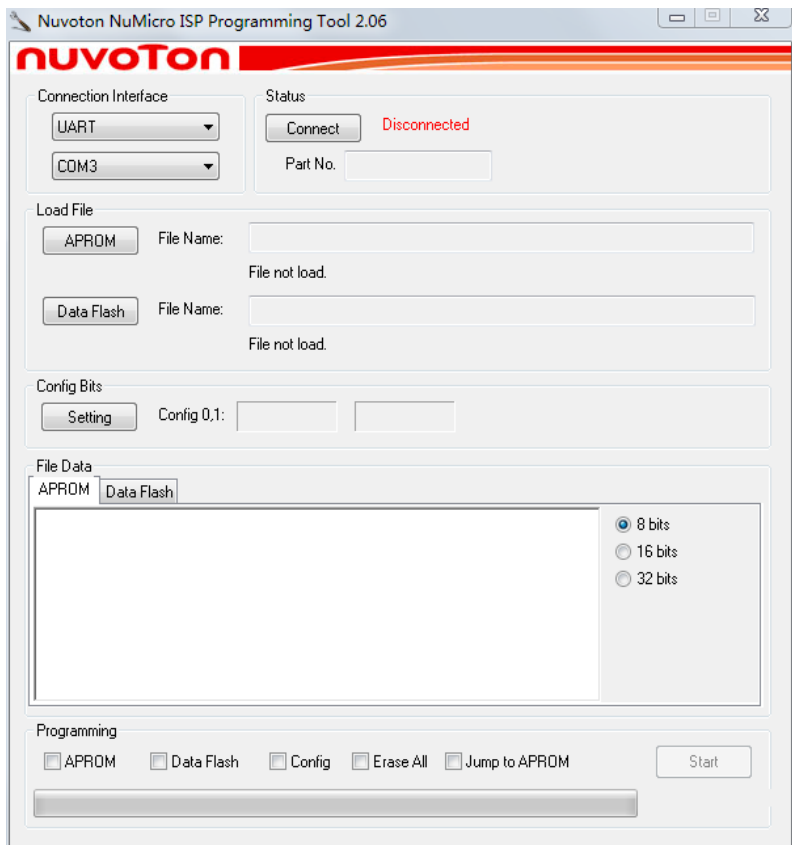
蓝灯：接收到正确报文(自定义协议)

红灯：采样数据不稳定

黄灯：标定指示

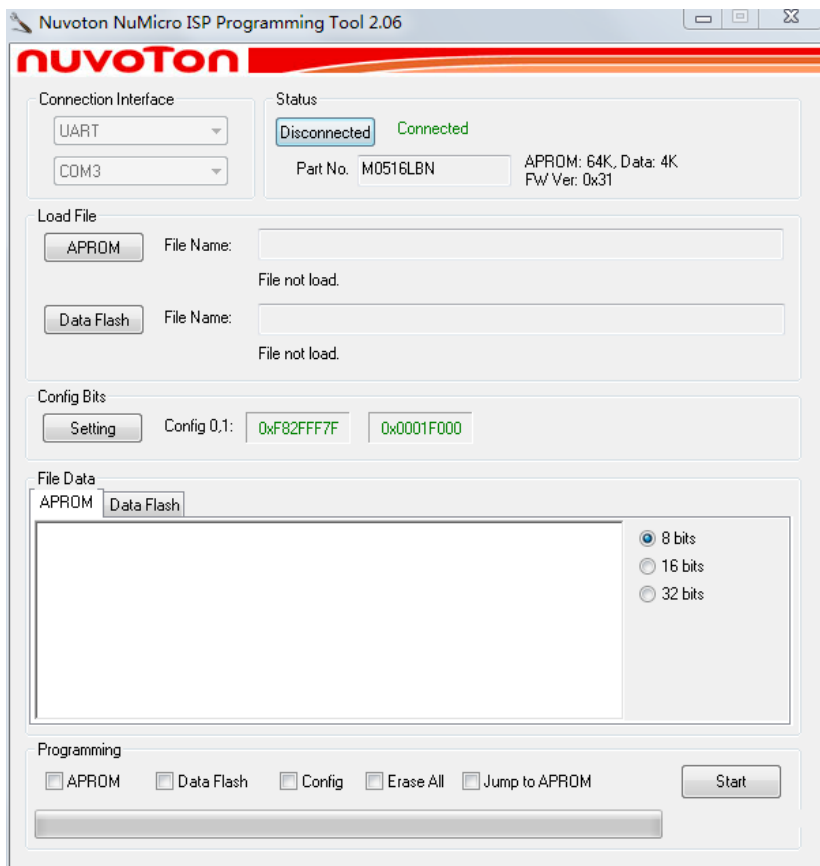
第四章 在线升级程序

1. 点击快捷图标，打开Nuvoton官方工具，如下图所示：

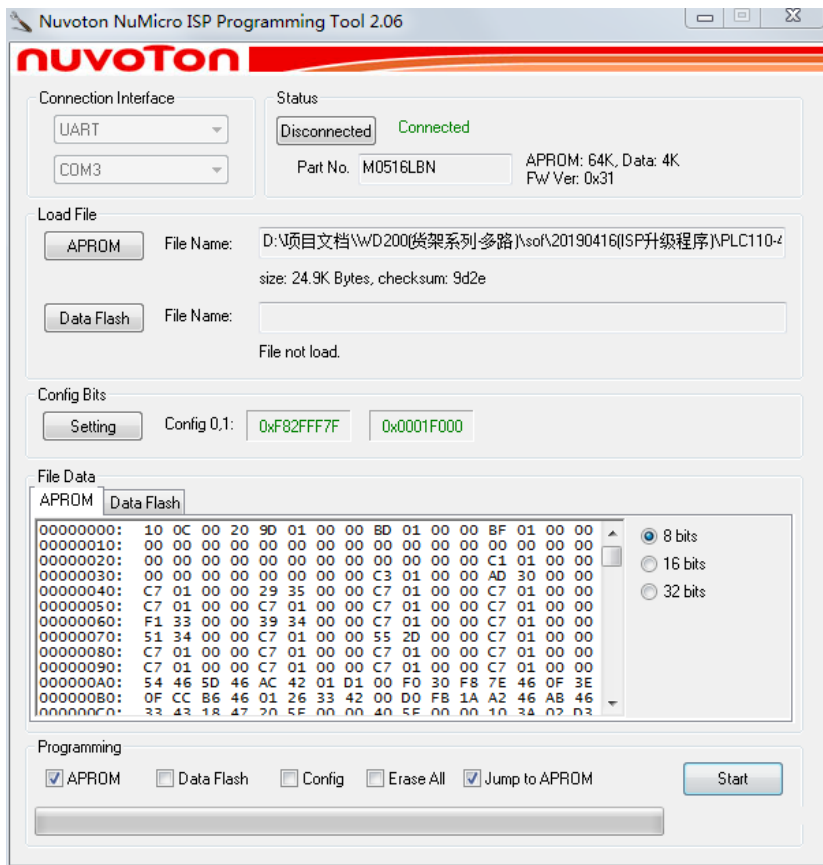


注：选择Connection Interface下的UART和对应串口→Connect

2. 板卡断电重启会出现下图所示:

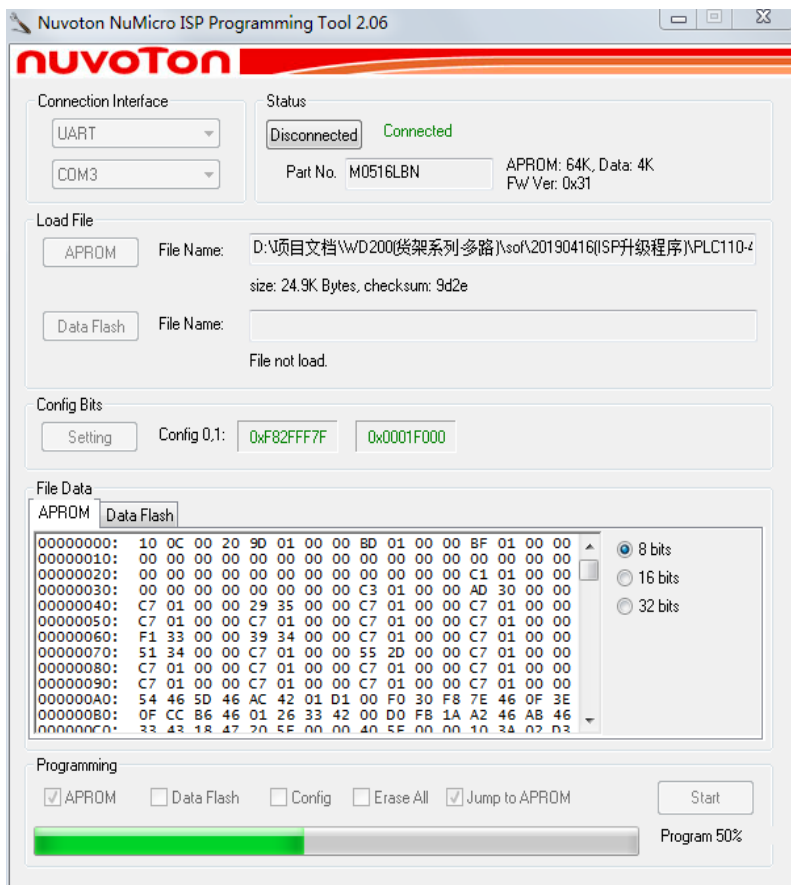


注: Status变更为Connected

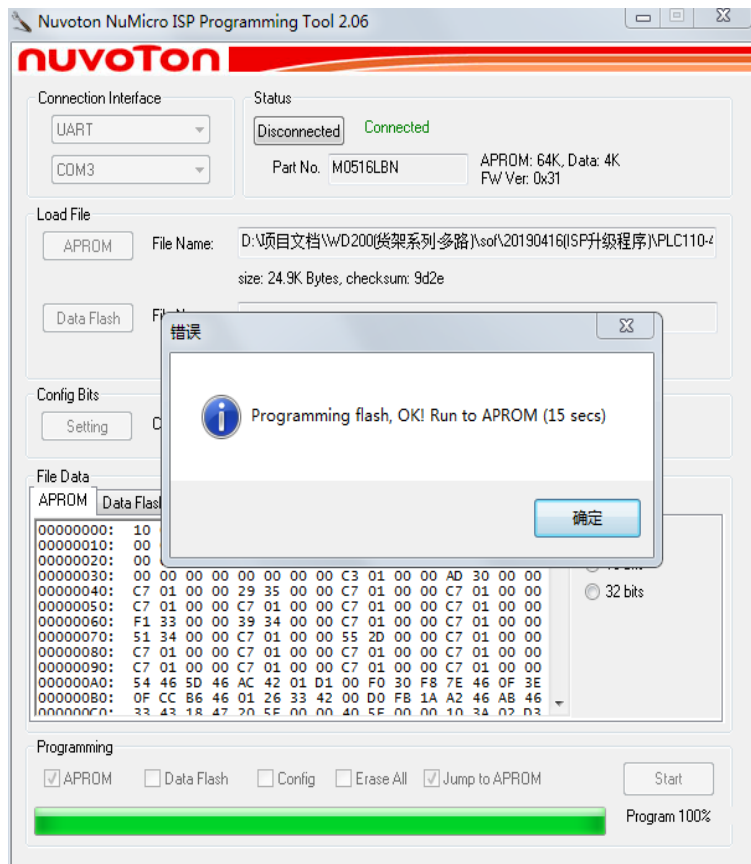


注：选择Load File→ARROM→File Name(后缀为.bin文件)→Programming→ARROM→Jump to ARROM→Start

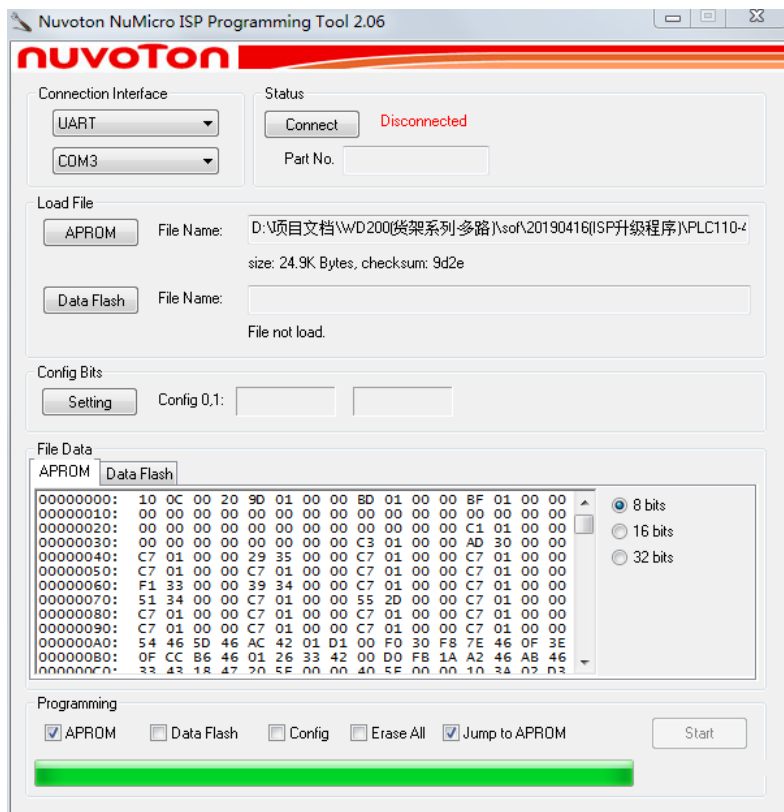
3. 开始下载程序



4. 程序升级完成，点击确定



5. 程序升级完成后会自动断开与板卡的连接



天津市丽景微电子有限公司

**地址：天津市华苑产业区环外海泰南道28号
海泰国际产业基地C座7门302室**

邮编：300384

免费服务电话：4000-022-007

电话：022-83719630/83719631

网址：<http://www.lascaux.com.cn>

邮箱：mail@lascaux.com.cn