

苏州迅鹏

技术规范

WPR70C/50B-系列无纸记录仪与 PC 机

Modbus 通讯协议

版本号		A
NOTES		
技术规范		
编写人		
审核人		
审批人		

编写时间	2015 年 6 月 2 日
编写人	赵克安

版本记录 |

[illegible]

版本记录 II

[illegible]

适用范围

本协议适用于 WPR70C/WPR50B 系列无纸记录仪与 PC 机之间的通讯。

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 4 页	共 45 页

目 录

1 Modbus 协议.....	6
1.1 传输模式.....	6
1.2 消息帧.....	6
1.3 功能代码.....	7
1.4 数据校验.....	9
2 Modbus 层次结构.....	10
3 命令层.....	10
3.1 命令定义.....	10
3.2 消息帧格式.....	10
3.3 Modbus 编址.....	11
3.3.1 地址格式.....	11
3.3.2 数据类型.....	11
3.3.3 变量类型和偏移量.....	11
3.4 命令解释.....	37
3.4.1 01H 读取一组开关量.....	37
3.4.2 03H 读一个或多个寄存器.....	37
3.4.3 05H 写一个开关量.....	38
3.4.4 06H 写一个寄存器.....	38
3.4.5 0FH 写多个开关量.....	39
3.4.6 10H 写多个寄存器.....	40
附录 A 异常代码.....	41
附录 B CRC 检测.....	41
1. CRC 校验.....	41
2. 函数实现.....	42
附录 C 通讯格式.....	45

1 Modbus 协议

1.1 传输模式

Modbus 标准网络两种传输模式：ASCII 或RTU。本协议中使用RTU模式，不支持ASCII模式。

RTU模式

地址	功能代码	数据数	数据 1	...	数据 n	CRC 高字	CRC 低字
----	------	-----	------	-----	------	--------	--------

当控制器设为在 Modbus 网络上以 RTU（远程终端单元）模式通信，在消息中的每个 8Bit 字节包含两个 4Bit 的十六进制字符。这种方式的主要优点是：在同样的波特率下，可比 ASCII 方式传送更多的数据。

代码系统

- 8 位二进制，十六进制数 0...9, A...F
- 消息中的每个 8 位域都是一个两个十六进制字符组成

每个字节的位：

- 1 个起始位
- 8 个数据位，最小的有效位先发送
- 1 个奇偶校验位，无校验则无
- 1 个停止位（有校验时），2 个 Bit（无校验时）

错误检测域

- CRC(循环冗长检测)

1.2 消息帧

传输中，传输设备以将 Modbus 消息转为有起点和终点的帧，这就允许接收的设备在消息起始处开始工作，读地址分配信息，判断哪一个设备被选中（广播方式则传给所有设备），判知何时信息已完成。部分的消息也能侦测到并且错误能设置为返回结果。

使用 RTU 模式，消息发送至少要以 3.5 个字符时间的停顿间隔开始。在网络波特率下多样的字符时间，这是最容易实现的(如下图的 T1-T2-T3-T4 所示)。传输的第一个域是设备地址。可以使用

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 6 页	共 45 页

的传输字符是十六进制的 0...9,A...F。网络设备不断侦测网络总线，包括停顿间隔时间内。当第一个域（地址域）接收到，每个设备都进行解码以判断是否发往自己的。在最后一个传输字符之后，一个至少 3.5 个字符时间的停顿标定了消息的结束。一个新的消息可在此停顿后开始。

整个消息帧必须作为一连续的流传输。如果在帧完成之前有超过 1.5 个字符时间的停顿时间，接收设备将刷新不完整的消息并假定下一字节是一个新消息的地址域。同样地，如果一个新消息在小于 3.5 个字符时间内接着前个消息开始，接收的设备将认为它是前一消息的延续。这将导致一个错误，因为在最后的 CRC 域的值不可能是正确的。一典型的消息帧如下所示：

起始位	设备地址	功能代码	数据	CRC 校验	结束符
T1-T2-T3-T4	8Bit	8Bit	n 个 8Bit	16Bit	T1-T2-T3-T4

图 RTU 消息帧

3、地址域

消息帧的地址域包含 8Bit（RTU），单个设备的地址范围是 6...254。主设备通过将要联络的从设备的地址放入消息中的地址域来选通从设备。当从设备发送回应消息时，它把自己的地址放入回应的地址域中，以便主设备知道是哪一个设备作出回应。

4、字符的连续传输

RTU字符帧位的序列是：

有奇偶校验

起始位	1	2	3	4	5	6	7	8	奇偶位	停止位
-----	---	---	---	---	---	---	---	---	-----	-----

无奇偶校验

起始位	1	2	3	4	5	6	7	8	停止位	停止位
-----	---	---	---	---	---	---	---	---	-----	-----

位顺序（RTU）

1.3 功能代码

下表是Modbus支持的功能码：

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 7 页	共 45 页

功能码	名称	作用
01	读取线圈状态	取得一组逻辑线圈的当前状态（ON/OFF）
02	读取输入状态	取得一组开关输入的当前状态（ON/OFF）
03	读取保持寄存器	在一个或多个保持寄存器中取得当前的二进制值
04	读取输入寄存器	在一个或多个输入寄存器中取得当前的二进制值
05	强置单线圈	强置一个逻辑线圈的通断状态
06	预置单寄存器	把具体二进制装入一个保持寄存器
07	读取异常状态	取得8个内部线圈的通断状态，这8个线圈的地址由控制器决定
08	回送诊断校验	把诊断校验报文送从机，以对通信处理进行评鉴
09	编程（只用于484）	使主机模拟编程器作用，修改PC从机逻辑
10	控询（只用于484）	可使主机与一台正在执行长程序任务从机通信，探询该从机是否已完成其操作任务，仅在含有功能码9的报文发送后，本功能码才发送
11	读取事件计数	可使主机发出单询问，并随即判定操作是否成功，尤其是该命令或其他应答产生通信错误时
12	读取通信事件记录	可是主机检索每台从机的ModBus事务处理通信事件记录。如果某项事务处理完成，记录会给出有关错误
13	编程（184/384 484 584）	可使主机模拟编程器功能修改PC从机逻辑
14	探询（184/384 484 584）	可使主机与正在执行任务的从机通信，定期控询该从机是否已完成其程序操作，仅在含有功能13的报文发送后，本功能码才得发送
15	强置多线圈	强置一串连续逻辑线圈的通断
16	预置多寄存器	把具体的二进制值装入一串连续的保持寄存器
17	报告从机标识	可使主机判断编址从机的类型及该从机运行指示灯的状态
18	（884和MICRO 84）	可使主机模拟编程功能，修改PC状态逻辑

19	重置通信链路	发生非可修改错误后，是从机复位于已知状态，可重置顺序字节
20	读取通用参数（584L）	显示扩展存储器文件中的数据信息
21	写入通用参数（584L）	把通用参数写入扩展存储文件，或修改之
22~64	保留作扩展功能备用	
65~72	保留以备用户功能所用	留作用户功能的扩展编码
73~119	非法功能	
120~127	保留	留作内部作用
128~255	保留	用于异常应答

下表是ModBus各功能码对应的数据类型。

代码	功能	数据类型
01	读	位
02	读	位
03	读	整型、字符型、状态字、浮点型
04	读	整型、状态字、浮点型
05	写	位
06	写	整型、字符型、状态字、浮点型
08	N/A	重复“回路反馈”信息
15	写	位
16	写	整型、字符型、状态字、浮点型
17	读	字符型

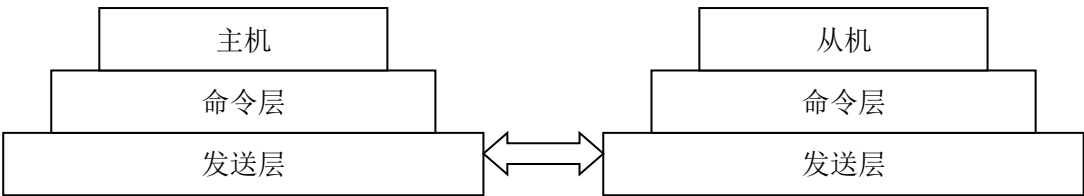
1.4 数据校验

标准的 Modbus 串行网络采用两种错误检测方法。奇偶校验对每个字符都可用，帧检测（LRC 或 CRC）应用于整个消息。它们都是在消息发送前由主设备产生的，从设备在接收过程中检测每个字符和整个消息帧。

用户要给主设备配置一预先定义的超时时间间隔，这个时间间隔要足够长，以使任何从设备都能作为正常反应。如果从设备测到一传输错误，消息将不会接收，也不会向主设备作出回应。这样超时事件将触发主设备来处理错误。发往不存在的从设备的地址也会产生超时。

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 9 页	共 45 页

2 Modbus 层次结构



Modbus 的层次结构

采用主从方式进行通讯

发送层：负责数据传输，目前分为两种方式（RS232/485 和 Ethernet）

命令层：发送和接收数据报

主机/从机：负责数据内容的填充

3 命令层

3.1 命令定义

支持的功能如下

功能代码	功能	解释	适用仪表
01H	读一组开关量	读一组开关量的当前状态	WPR50B、WPR70C、WPR80A
03H	读一个或多个寄存器	读一个或一组寄存器的值	WPR50B、WPR70C、WPR80A
05H	写一个开关量	写一个开关量的状态	WPR50B、WPR70C
06H	写一个寄存器	写一个寄存器的值	WPR50B、WPR70C
0FH	写多个开关量	写一组开关量的状态	WPR50B、WPR70C
10H	写多个寄存器	写一组寄存器的值	WPR50B、WPR70C

3.2 消息帧格式

除非特别指明，凡是大于 1 个字节的数据（如 int 型或 short 型），均为高字节在前，低字节在后。帧的最大长度需限制在 256 字节。

除了广播命令，任何主机对从机的访问均需从机回应。

CRC 校验出错无需返回。

主机发送格式

名称	设备地址	功能代码	数据区	CRC
字节	1	1	N	2

主机发送格式

名称	设备地址	功能代码	数据区	CRC
字节	1	1	N	2

3.3 Modbus 编址

定义 Modbus 的地址格式，数据类型和变量类型等。本协议把所有的变量分成两类。组态变量和实时变量。组态变量是仪表组态设置相关的量，如组态常数；实时变量是仪表运行过程的实时量，如 AI 的测量值，AO 的输出值等。

3.3.1 地址格式

Modbus 的地址用两个字节表示，地址编码从 1 开始，小于 0x5000 的地址为实时变量地址，大于等于 0x5000 的地址为组态变量地址。

3.3.2 数据类型

类型	说明
Boolean（开关量）	1 Bit Boolean
Integer（寄存器）	16 Bits Integer
Floating	32 Bits IEEE Floating Point

大于 2 字节的数据均用 Integer 类型的数据组合而成

浮点数的字节顺序不遵循高字节在前的规定，而是按仪表[浮点格式]项的设置排列，格式和数据排列关系如下：

浮点格式	字节排列（以 48.81667 为例，其 16 进制值为 (0x42434445)）	按 3210 格式解析，左侧 16 进制数相对应的浮点数
------	--	------------------------------

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 11 页	共 45 页

0123	0x45, 0x44, 0x43, 0x42	3. 140204e+3
1032	0x44, 0x45, 0x42, 0x43	7. 890353e+2
2301	0x43, 0x42, 0x45, 0x44	1. 942706e+2
3210	0x42, 0x43, 0x44, 0x45	4. 881667e+1

3.3.3 变量类型和偏移量

3.3.3.1 实时变量

实时变量一览表：

序号	变量	起始地址	读写属性	数据类型	功能代码
1	AI 工程量实时值	0x0001	只读	浮点数	0x03
2	AI 报警状态（6 种）	0x0100	只读	开关量	0x01
3	DI 输入	0x0300	只读	开关量	0x01
4	FI 输入	0x0400	只读	浮点数	0x03
5	FI 报警状态（6 种）	0x0500	只读	开关量	0x01
6	A0 的工程量实时值	0x0600	只读	浮点数	0x03
7	DO 输出	0x0700	只读	开关量	0x01
8	PWM 输出	0x0800	只读	开关量	0x01
9	VA 工程量实时值	0x0900	只读	浮点数	0x03
10	VA 报警状态（6 种）	0x0a00	只读	开关量	0x01
11	VD 的状态	0x0b00	只读	开关量	0x01
12	ACC 的工程量	0x0c00	只读	浮点数	0x03
13	FLOW 的工程量	0x0d00	只读	浮点数	0x03
14	FLOW 报警状态（6 种）	0x0e00	只读	开关量	0x01
15	TIM 的状态	0x0F00	只读	开关量	0x01
16	PID 回路 SV 的工程量	0x1000	读写	浮点数	0x03, 0x10
17	PID 回路 MV 的工程量	0x1100	读写	浮点数	0x03, 0x10
18	PID 回路 DH 的状态	0x1200	只读	开关量	0x01
19	PID 回路的状态	0x1300	只读	寄存器	0x03
20	PID 回路的 AM	0x1320	只写	开关量	0x05, 0x0F
21	PID 回路的 LR	0x1340	只写	开关量	0x05, 0x0F
22	PID 回路的 Tune	0x1360	只写	开关量	0x05
23	PID 回路的 PID 参数（3 个）	0x1400	读写	寄存器	0x03, 0x06, 0x10
24	ONOFF 回路的状态	0x1600	只读	寄存器	0x03
25	ONOFF 回路的 AM	0x1620	只写	开关量	0x05, 0x0F
26	ONOFF 回路的 LR	0x1640	只写	开关量	0x05, 0x0F
27	ONOFF 回路 DH 的状态	0x1660	只读	开关量	0x01

28	ONOFF 回路 SV 的工程量	0x1700	读写	浮点数	0x03, 0x10
29	ONOFF 回路 MV 的工程量	0x1720	读写	开关量	0x01, 0x05, 0x0F
30	PROG 回路的状态	0x3200	只读	寄存器	0x03
31	PROG 回路的工艺号	0x3220	读写	寄存器	0x03, 0x06, 0x10
32	PROG 回路的 Run/Hold	0x3300	只写	开关量	0x05, 0x0F
33	PROG 回路的 Reset	0x3320	只写	开关量	0x05, 0x0F
34	PROG 回路的 FastRun	0x3340	只写	开关量	0x05, 0x0F
35	PROG 回路的 DO	0x3400	只读	开关量	0x01
36	PROG 回路的 Seg	0x3500	读写	寄存器	0x03, 0x06, 0x10
37	PROG 回路的段总时间	0x3520	只读	寄存器	0x03
38	PROG 回路已运行时间	0x3540	只读	寄存器	0x03
39	PROG 回路 PV 工程量	0x3600	只读	浮点数	0x03
40	PROG 回路 OUT 工程量	0x3620	只读	浮点数	0x03
41	CONB	0x3700	读写	开关量	0x01, 0x05, 0x0F
42	CONI	0x3800	读写	寄存器	0x03, 0x06, 0x10
43	CONF	0x3900	读写	浮点数	0x03, 0x10

实时变量详细地址分布：

AI：变量类型 0

地址	数据类型	功能代码	变量	说明	访问权限
0x0001	Floating	03	AI01	模拟量输入	只读
0x0003	Floating	03	AI02	模拟量输入	只读
0x0005	Floating	03	AI03	模拟量输入	只读
0x0007	Floating	03	AI04	模拟量输入	只读
0x0009	Floating	03	AI05	模拟量输入	只读
0x000B	Floating	03	AI06	模拟量输入	只读
0x000D	Floating	03	AI07	模拟量输入	只读
0x000F	Floating	03	AI08	模拟量输入	只读
0x0011	Floating	03	AI09	模拟量输入	只读
0x0013	Floating	03	AI10	模拟量输入	只读
0x0015	Floating	03	AI11	模拟量输入	只读
0x0017	Floating	03	AI12	模拟量输入	只读
.....
0x003D	Floating	03	AI31	模拟量输入	只读
0x003F	Floating	03	AI32	模拟量输入	只读

WPR50B 最大支持 AI01~AI16

WPR70C 最大支持 AI01~AI24

WPR80A 最大支持 AI01~AI32

AI 的报警：变量类型 1~2

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 13 页	共 45 页

地址	数据类型	功能代码	变量	说明	访问权限	通道
0x0100	Boolean	01	AI01.HH	上上限报警	只读	AI01
0x0101	Boolean	01	AI01.Hi	上限报警	只读	
0x0102	Boolean	01	AI01.Lo	下限报警	只读	
0x0103	Boolean	01	AI01.LL	下下限报警	只读	
0x0104	Boolean	01	AI01.RH	上升速率报警	只读	
0x0105	Boolean	01	AI01.RL	下降速率报警	只读	
0x0106	Boolean	01	AI02.HH	上上限报警	只读	AI02
0x0107	Boolean	01	AI02.Hi	上限报警	只读	
0x0108	Boolean	01	AI02.Lo	下限报警	只读	
0x0109	Boolean	01	AI02.LL	下下限报警	只读	
0x010A	Boolean	01	AI02.RH	上升速率报警	只读	
0x010B	Boolean	01	AI02.RL	下降速率报警	只读	
0x010C	Boolean	01	AI03.HH	上上限报警	只读	AI03
0x010D	Boolean	01	AI03.Hi	上限报警	只读	
0x010E	Boolean	01	AI03.Lo	下限报警	只读	
0x010F	Boolean	01	AI03.LL	下下限报警	只读	
0x0110	Boolean	01	AI03.RH	上升速率报警	只读	
0x0111	Boolean	01	AI03.RL	下降速率报警	只读	
0x0112	Boolean	01	AI04.HH	上上限报警	只读	AI04
0x0113	Boolean	01	AI04.Hi	上限报警	只读	
0x0114	Boolean	01	AI04.Lo	下限报警	只读	
0x0115	Boolean	01	AI04.LL	下下限报警	只读	
0x0116	Boolean	01	AI04.RH	上升速率报警	只读	
0x0117	Boolean	01	AI04.RL	下降速率报警	只读	
0x0118	Boolean	01	AI05.HH	上上限报警	只读	AI05
0x0119	Boolean	01	AI05.Hi	上限报警	只读	
0x011A	Boolean	01	AI05.Lo	下限报警	只读	
0x011B	Boolean	01	AI05.LL	下下限报警	只读	
0x011C	Boolean	01	AI05.RH	上升速率报警	只读	
0x011D	Boolean	01	AI05.RL	下降速率报警	只读	
0x011E	Boolean	01	AI06.HH	上上限报警	只读	AI06
0x011F	Boolean	01	AI06.Hi	上限报警	只读	
0x0120	Boolean	01	AI06.Lo	下限报警	只读	
0x0121	Boolean	01	AI06.LL	下下限报警	只读	
0x0122	Boolean	01	AI06.RH	上升速率报警	只读	
0x0123	Boolean	01	AI06.RL	下降速率报警	只读	
0x0124	Boolean	01	AI07.HH	上上限报警	只读	AI07
0x0125	Boolean	01	AI07.Hi	上限报警	只读	

0x0126	Boolean	01	AI07.Lo	下限报警	只读	
0x0127	Boolean	01	AI07.LL	下下限报警	只读	
0x0128	Boolean	01	AI07.RH	上升速率报警	只读	
0x0129	Boolean	01	AI07.RL	下降速率报警	只读	
0x012A	Boolean	01	AI08.HH	上上限报警	只读	AI08
0x012B	Boolean	01	AI08.Hi	上限报警	只读	
0x012C	Boolean	01	AI08.Lo	下限报警	只读	
0x012D	Boolean	01	AI08.LL	下下限报警	只读	
0x012E	Boolean	01	AI08.RH	上升速率报警	只读	
0x012F	Boolean	01	AI08.RL	下降速率报警	只读	
0x0130	Boolean	01	AI09.HH	上上限报警	只读	AI09
0x0131	Boolean	01	AI09.Hi	上限报警	只读	
0x0132	Boolean	01	AI09.Lo	下限报警	只读	
0x0133	Boolean	01	AI09.LL	下下限报警	只读	
0x0134	Boolean	01	AI09.RH	上升速率报警	只读	
0x0135	Boolean	01	AI09.RL	下降速率报警	只读	
0x0136	Boolean	01	AI10.HH	上上限报警	只读	AI10
0x0137	Boolean	01	AI10.Hi	上限报警	只读	
0x0138	Boolean	01	AI10.Lo	下限报警	只读	
0x0139	Boolean	01	AI10.LL	下下限报警	只读	
0x013A	Boolean	01	AI10.RH	上升速率报警	只读	
0x013B	Boolean	01	AI10.RL	下降速率报警	只读	
0x013C	Boolean	01	AI11.HH	上上限报警	只读	AI11
0x013D	Boolean	01	AI11.Hi	上限报警	只读	
0x013E	Boolean	01	AI11.Lo	下限报警	只读	
0x013F	Boolean	01	AI11.LL	下下限报警	只读	
0x0140	Boolean	01	AI11.RH	上升速率报警	只读	
0x0141	Boolean	01	AI11.RL	下降速率报警	只读	
0x0142	Boolean	01	AI12.HH	上上限报警	只读	AI12
0x0143	Boolean	01	AI12.Hi	上限报警	只读	
0x0144	Boolean	01	AI12.Lo	下限报警	只读	
0x0145	Boolean	01	AI12.LL	下下限报警	只读	
0x0146	Boolean	01	AI12.RH	上升速率报警	只读	
0x0147	Boolean	01	AI12.RL	下降速率报警	只读	
.....
.....	
.....	
.....	
.....	

.....	
0x01BA	Boolean	01	AI32.HH	上上限报警	只读	AI32
0x01BB	Boolean	01	AI32.Hi	上限报警	只读	
0x01BC	Boolean	01	AI32.Lo	下限报警	只读	
0x01BD	Boolean	01	AI32.LL	下下限报警	只读	
0x01BE	Boolean	01	AI32.RH	上升速率报警	只读	
0x01BF	Boolean	01	AI32.RL	下降速率报警	只读	

WPR50B 支持 AI01～AI12

WPR70C 支持 AI01～AI16

WPR80A 支持 AI01～AI32

DI：变量类型为 3

地址	数据类型	功能代码	变量	说明	访问权限
0x0300	Boolean	01	DI01	开关量输入	只读
0x0301	Boolean	01	DI02	开关量输入	只读
0x0302	Boolean	01	DI03	开关量输入	只读

WPR50B 最大支持 DI01～DI02

WPR70C 最大支持 DI01～DI03

FI：变量类型为 4

地址	数据类型	功能代码	变量	说明	访问权限
0x0400	Floating	03	FI01	频率量输入	只读
0x0402	Floating	03	FI02	频率量输入	只读
0x0404	Floating	03	FI03	频率量输入	只读

WPR50B 最大支持 FI01～FI02

WPR70C 最大支持 FI01～FI03

FI 的报警：变量类型为 5

地址	数据类型	功能代码	变量	说明	访问权限	通道
0x0500	Boolean	01	FI01.HH	上上限报警	只读	FI01
0x0501	Boolean	01	FI01.Hi	上限报警	只读	
0x0502	Boolean	01	FI01.Lo	下限报警	只读	
0x0503	Boolean	01	FI01.LL	下下限报警	只读	
0x0504	Boolean	01	FI01.RH	上升速率报警	只读	
0x0505	Boolean	01	FI01.RL	下降速率报警	只读	
0x0506	Boolean	01	FI02.HH	上上限报警	只读	FI02
0x0507	Boolean	01	FI02.Hi	上限报警	只读	
0x0508	Boolean	01	FI02.Lo	下限报警	只读	
0x0509	Boolean	01	FI02.LL	下下限报警	只读	
0x050A	Boolean	01	FI02.RH	上升速率报警	只读	

0x050B	Boolean	01	FI02.RL	下降速率报警	只读	FI03
0x0506	Boolean	01	FI03.HH	上上限报警	只读	
0x0507	Boolean	01	FI03.Hi	上限报警	只读	
0x0508	Boolean	01	FI03.Lo	下限报警	只读	
0x0509	Boolean	01	FI03.LL	下下限报警	只读	
0x050A	Boolean	01	FI03.RH	上升速率报警	只读	
0x050B	Boolean	01	FI03.RL	下降速率报警	只读	

WPR50B 最大支持 FI01~FI02

WPR70C 最大支持 FI01~FI03

AO：变量类型为 6

地址	数据类型	功能代码	变量	说明	访问权限
0x0600	Floating	03	AO01	模拟量输出	只读
0x0602	Floating	03	AO02	模拟量输出	只读
0x0604	Floating	03	AO03	模拟量输出	只读
0x0606	Floating	03	AO04	模拟量输出	只读
0x0608	Floating	03	AO05	模拟量输出	只读
0x060A	Floating	03	AO06	模拟量输出	只读

WPR50B 最大支持 AO01~AO04

WPR70C 最大支持 AO01~AO06

DO：变量类型为 7

地址	数据类型	功能代码	变量	说明	访问权限
0x0700	Boolean	01	DO01	开关量输出	只读
0x0701	Boolean	01	DO02	开关量输出	只读
0x0702	Boolean	01	DO03	开关量输出	只读
0x0703	Boolean	01	DO04	开关量输出	只读
0x0704	Boolean	01	DO05	开关量输出	只读
0x0705	Boolean	01	DO06	开关量输出	只读
0x0706	Boolean	01	DO07	开关量输出	只读
0x0707	Boolean	01	DO08	开关量输出	只读
0x0708	Boolean	01	DO09	开关量输出	只读
0x0709	Boolean	01	DO10	开关量输出	只读
0x070A	Boolean	01	DO11	开关量输出	只读
0x070B	Boolean	01	DO12	开关量输出	只读
.....
0x071F	Boolean	01	DO32	开关量输出	只读

WPR50B 最大支持 DO01~DO12

WPR70C 最大支持 DO01~DO16

WPR80A 最大支持 DO01~DO32

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 17 页	共 45 页

PWM: 变量类型为 8

地址	数据类型	功能代码	变量	说明	访问权限
0x0800	Boolean	01	PWM01	时间比例输出	只读
0x0802	Boolean	01	PWM02	时间比例输出	只读
0x0804	Boolean	01	PWM03	时间比例输出	只读
0x0806	Boolean	01	PWM04	时间比例输出	只读
0x0808	Boolean	01	PWM05	时间比例输出	只读
0x080A	Boolean	01	PWM06	时间比例输出	只读
0x080C	Boolean	01	PWM07	时间比例输出	只读
0x080E	Boolean	01	PWM08	时间比例输出	只读
0x0810	Boolean	01	PWM09	时间比例输出	只读
0x0812	Boolean	01	PWM10	时间比例输出	只读
0x0814	Boolean	01	PWM11	时间比例输出	只读
0x0816	Boolean	01	PWM12	时间比例输出	只读

WPR50B 最大支持 PWM01~PWM04

WPR70C 最大支持 PWM01~PWM08

VA: 变量类型为 9

地址	数据类型	功能代码	变量	说明	访问权限
0x0900	Floating	03	VA01	虚拟模拟量	只读
0x0902	Floating	03	VA02	虚拟模拟量	只读
0x0904	Floating	03	VA03	虚拟模拟量	只读
0x0906	Floating	03	VA04	虚拟模拟量	只读
0x0908	Floating	03	VA05	虚拟模拟量	只读
0x090A	Floating	03	VA06	虚拟模拟量	只读
0x090C	Floating	03	VA07	虚拟模拟量	只读
0x090E	Floating	03	VA08	虚拟模拟量	只读
0x0910	Floating	03	VA09	虚拟模拟量	只读
0x0912	Floating	03	VA10	虚拟模拟量	只读
0x0914	Floating	03	VA11	虚拟模拟量	只读
0x0916	Floating	03	VA12	虚拟模拟量	只读
0x0918	Floating	03	VA13	虚拟模拟量	只读
0x091A	Floating	03	VA14	虚拟模拟量	只读
0x091C	Floating	03	VA15	虚拟模拟量	只读
0x091E	Floating	03	VA16	虚拟模拟量	只读

WPR50B 最大支持 VA01~VA16

WPR70C 最大支持 VA01~VA16

VA 的报警: 变量类型为 10

地址	数据类型	功能代码	变量	说明	访问权限	通道
----	------	------	----	----	------	----

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 18 页	共 45 页

0x0A00	Boolean	01	VA01.HH	上上限报警	只读	VA01
0x0A01	Boolean	01	VA01.Hi	上限报警	只读	
0x0A02	Boolean	01	VA01.Lo	下限报警	只读	
0x0A03	Boolean	01	VA01.LL	下下限报警	只读	
0x0A04	Boolean	01	VA01.RH	上升速率报警	只读	
0x0A05	Boolean	01	VA01.RL	下降速率报警	只读	
0x0A06	Boolean	01	VA02.HH	上上限报警	只读	VA02
0x0A07	Boolean	01	VA02.Hi	上限报警	只读	
0x0A08	Boolean	01	VA02.Lo	下限报警	只读	
0x0A09	Boolean	01	VA02.LL	下下限报警	只读	
0x0A0A	Boolean	01	VA02.RH	上升速率报警	只读	
0x0A0B	Boolean	01	VA02.RL	下降速率报警	只读	
0x0A0C	Boolean	01	VA03.HH	上上限报警	只读	VA03
0x0A0D	Boolean	01	VA03.Hi	上限报警	只读	
0x0A0E	Boolean	01	VA03.Lo	下限报警	只读	
0x0A0F	Boolean	01	VA03.LL	下下限报警	只读	
0x0A10	Boolean	01	VA03.RH	上升速率报警	只读	
0x0A11	Boolean	01	VA03.RL	下降速率报警	只读	
0x0A12	Boolean	01	VA04.HH	上上限报警	只读	VA04
0x0A13	Boolean	01	VA04.Hi	上限报警	只读	
0x0A14	Boolean	01	VA04.Lo	下限报警	只读	
0x0A15	Boolean	01	VA04.LL	下下限报警	只读	
0x0A16	Boolean	01	VA04.RH	上升速率报警	只读	
0x0A17	Boolean	01	VA04.RL	下降速率报警	只读	
0x0A18	Boolean	01	VA05.HH	上上限报警	只读	VA05
0x0A19	Boolean	01	VA05.Hi	上限报警	只读	
0x0A1A	Boolean	01	VA05.Lo	下限报警	只读	
0x0A1B	Boolean	01	VA05.LL	下下限报警	只读	
0x0A1C	Boolean	01	VA05.RH	上升速率报警	只读	
0x0A1D	Boolean	01	VA05.RL	下降速率报警	只读	
0x0A1E	Boolean	01	VA06.HH	上上限报警	只读	VA06
0x0A1F	Boolean	01	VA06.Hi	上限报警	只读	
0x0A20	Boolean	01	VA06.Lo	下限报警	只读	
0x0A21	Boolean	01	VA06.LL	下下限报警	只读	
0x0A22	Boolean	01	VA06.RH	上升速率报警	只读	
0x0A23	Boolean	01	VA06.RL	下降速率报警	只读	
0x0A24	Boolean	01	VA07.HH	上上限报警	只读	VA07
0x0A25	Boolean	01	VA07.Hi	上限报警	只读	
0x0A26	Boolean	01	VA07.Lo	下限报警	只读	

0x0A27	Boolean	01	VA07.LL	下下限报警	只读	
0x0A28	Boolean	01	VA07.RH	上升速率报警	只读	
0x0A29	Boolean	01	VA07.RL	下降速率报警	只读	
0x0A2A	Boolean	01	VA08.HH	上上限报警	只读	VA08
0x0A2B	Boolean	01	VA08.Hi	上限报警	只读	
0x0A2C	Boolean	01	VA08.Lo	下限报警	只读	
0x0A2D	Boolean	01	VA08.LL	下下限报警	只读	
0x0A2E	Boolean	01	VA08.RH	上升速率报警	只读	
0x0A2F	Boolean	01	VA08.RL	下降速率报警	只读	
0x0A00	Boolean	01	VA09.HH	上上限报警	只读	VA09
0x0A01	Boolean	01	VA09.Hi	上限报警	只读	
0x0A02	Boolean	01	VA09.Lo	下限报警	只读	
0x0A03	Boolean	01	VA09.LL	下下限报警	只读	
0x0A04	Boolean	01	VA09.RH	上升速率报警	只读	
0x0A05	Boolean	01	VA09.RL	下降速率报警	只读	VA10
0x0A06	Boolean	01	VA10.HH	上上限报警	只读	
0x0A07	Boolean	01	VA10.Hi	上限报警	只读	
0x0A08	Boolean	01	VA10.Lo	下限报警	只读	
0x0A09	Boolean	01	VA10.LL	下下限报警	只读	
0x0A0A	Boolean	01	VA10.RH	上升速率报警	只读	
0x0A0B	Boolean	01	VA10.RL	下降速率报警	只读	VA11
0x0A0C	Boolean	01	VA11.HH	上上限报警	只读	
0x0A0D	Boolean	01	VA11.Hi	上限报警	只读	
0x0A0E	Boolean	01	VA11.Lo	下限报警	只读	
0x0A0F	Boolean	01	VA11.LL	下下限报警	只读	
0x0A10	Boolean	01	VA11.RH	上升速率报警	只读	VA12
0x0A11	Boolean	01	VA11.RL	下降速率报警	只读	
0x0A12	Boolean	01	VA12.HH	上上限报警	只读	
0x0A13	Boolean	01	VA12.Hi	上限报警	只读	
0x0A14	Boolean	01	VA12.Lo	下限报警	只读	
0x0A15	Boolean	01	VA12.LL	下下限报警	只读	
0x0A16	Boolean	01	VA12.RH	上升速率报警	只读	VA13
0x0A17	Boolean	01	VA12.RL	下降速率报警	只读	
0x0A18	Boolean	01	VA13.HH	上上限报警	只读	
0x0A19	Boolean	01	VA13.Hi	上限报警	只读	
0x0A1A	Boolean	01	VA13.Lo	下限报警	只读	
0x0A1B	Boolean	01	VA13.LL	下下限报警	只读	
0x0A1C	Boolean	01	VA13.RH	上升速率报警	只读	
0x0A1D	Boolean	01	VA13.RL	下降速率报警	只读	

0x0A1E	Boolean	01	VA14.HH	上上限报警	只读	VA14
0x0A1F	Boolean	01	VA14.Hi	上限报警	只读	
0x0A20	Boolean	01	VA14.Lo	下限报警	只读	
0x0A21	Boolean	01	VA14.LL	下下限报警	只读	
0x0A22	Boolean	01	VA14.RH	上升速率报警	只读	
0x0A23	Boolean	01	VA14.RL	下降速率报警	只读	
0x0A24	Boolean	01	VA15.HH	上上限报警	只读	VA15
0x0A25	Boolean	01	VA15.Hi	上限报警	只读	
0x0A26	Boolean	01	VA15.Lo	下限报警	只读	
0x0A27	Boolean	01	VA15.LL	下下限报警	只读	
0x0A28	Boolean	01	VA15.RH	上升速率报警	只读	
0x0A29	Boolean	01	VA15.RL	下降速率报警	只读	
0x0A2A	Boolean	01	VA16.HH	上上限报警	只读	VA16
0x0A2B	Boolean	01	VA16.Hi	上限报警	只读	
0x0A2C	Boolean	01	VA16.Lo	下限报警	只读	
0x0A2D	Boolean	01	VA16.LL	下下限报警	只读	
0x0A2E	Boolean	01	VA16.RH	上升速率报警	只读	
0x0A2F	Boolean	01	VA16.RL	下降速率报警	只读	

WPR50B 最大支持 VA01~VA016

WPR70C 最大支持 VA01~VA16

VD: 变量类型为 11

地址	数据类型	功能代码	变量	说明	访问权限
0x0B00	Boolean	01	VD01	虚拟开关量	只读
0x0B01	Boolean	01	VD02	虚拟开关量	只读
0x0B02	Boolean	01	VD03	虚拟开关量	只读
0x0B03	Boolean	01	VD04	虚拟开关量	只读
0x0B04	Boolean	01	VD05	虚拟开关量	只读
0x0B05	Boolean	01	VD06	虚拟开关量	只读
0x0B06	Boolean	01	VD07	虚拟开关量	只读
0x0B07	Boolean	01	VD08	虚拟开关量	只读
0x0B08	Boolean	01	VD09	虚拟开关量	只读
0x0B09	Boolean	01	VD10	虚拟开关量	只读
0x0B0A	Boolean	01	VD11	虚拟开关量	只读
0x0B0B	Boolean	01	VD12	虚拟开关量	只读
0x0B0C	Boolean	01	VD13	虚拟开关量	只读
0x0B0D	Boolean	01	VD14	虚拟开关量	只读
0x0B0E	Boolean	01	VD15	虚拟开关量	只读
0x0B0F	Boolean	01	VD16	虚拟开关量	只读
0x0B10	Boolean	01	VD17	虚拟开关量	只读

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 21 页	共 45 页

0x0B11	Boolean	01	VD18	虚拟开关量	只读
0x0B12	Boolean	01	VD19	虚拟开关量	只读
0x0B13	Boolean	01	VD20	虚拟开关量	只读
0x0B14	Boolean	01	VD21	虚拟开关量	只读
0x0B15	Boolean	01	VD22	虚拟开关量	只读
0x0B16	Boolean	01	VD23	虚拟开关量	只读
0x0B17	Boolean	01	VD24	虚拟开关量	只读
0x0B18	Boolean	01	VD25	虚拟开关量	只读
0x0B19	Boolean	01	VD26	虚拟开关量	只读
0x0B1A	Boolean	01	VD27	虚拟开关量	只读
0x0B1B	Boolean	01	VD28	虚拟开关量	只读
0x0B1C	Boolean	01	VD29	虚拟开关量	只读
0x0B1D	Boolean	01	VD30	虚拟开关量	只读
0x0B1E	Boolean	01	VD31	虚拟开关量	只读
0x0B1F	Boolean	01	VD32	虚拟开关量	只读

WPR50B 最大支持 VD01~VD032

WPR70C 最大支持 VD01~VD032

AC：变量类型为 12

地址	数据类型	功能代码	变量	说明	访问权限
0x0C00	Floating	3	AC01	累积值	只读
0x0C02	Floating	3	AC02	累积值	只读
0x0C04	Floating	3	AC03	累积值	只读
0x0C06	Floating	3	AC04	累积值	只读
0x0C08	Floating	3	AC05	累积值	只读
0x0C0A	Floating	3	AC06	累积值	只读
0x0C0C	Floating	3	AC07	累积值	只读
0x0C0E	Floating	3	AC08	累积值	只读
0x0C10	Floating	3	AC09	累积值	只读
0x0C12	Floating	3	AC10	累积值	只读
0x0C14	Floating	3	AC11	累积值	只读
0x0C16	Floating	3	AC12	累积值	只读

WPR50B 最大支持 AC01~AC16

WPR70C 最大支持 AC01~AC16

WPR80A 最大支持 AC01~AC32

FLOW：变量类型为 13

地址	数据类型	功能代码	变量	说明	访问权限
0x0D00	Floating	3	FLOW01	流量值	只读
0x0D02	Floating	3	FLOW02	流量值	只读

0x0D04	Floating	3	FLOW03	流量值	只读
0x0D06	Floating	3	FLOW04	流量值	只读
0x0D08	Floating	3	FLOW05	流量值	只读
0x0D0A	Floating	3	FLOW06	流量值	只读
0x0D0C	Floating	3	FLOW07	流量值	只读
0x0D0E	Floating	3	FLOW08	流量值	只读
0x0D10	Floating	3	FLOW09	流量值	只读
0x0D12	Floating	3	FLOW10	流量值	只读
0x0D14	Floating	3	FLOW11	流量值	只读
0x0D16	Floating	3	FLOW12	流量值	只读
0x0D18	Floating	3	FLOW13	流量值	只读
0x0D1A	Floating	3	FLOW14	流量值	只读
0x0D1C	Floating	3	FLOW15	流量值	只读
0x0D1E	Floating	3	FLOW16	流量值	只读

WPR50B 最大支持 FLOW01~FLOW12

WPR70C 最大支持 FLOW01~FLOW16

FLOW 的报警：变量类型为 14(目前还不支持该类型)

地址	数据类型	功能代码	变量	说明	访问权限	通道
0x0E00	Boolean	01	FLOW01.HH	上上限报警	只读	FLOW01
0x0E01	Boolean	01	FLOW01.Hi	上限报警	只读	
0x0E02	Boolean	01	FLOW01.Lo	下限报警	只读	
0x0E03	Boolean	01	FLOW01.LL	下下限报警	只读	
0x0E04	Boolean	01	FLOW01.RH	上升速率报警	只读	
0x0E05	Boolean	01	FLOW01.RL	下降速率报警	只读	
0x0E06	Boolean	01	FLOW02.HH	上上限报警	只读	FLOW02
0x0E07	Boolean	01	FLOW02.Hi	上限报警	只读	
0x0E08	Boolean	01	FLOW02.Lo	下限报警	只读	
0x0E09	Boolean	01	FLOW02.LL	下下限报警	只读	
0x0E0A	Boolean	01	FLOW02.RH	上升速率报警	只读	
0x0E0B	Boolean	01	FLOW02.RL	下降速率报警	只读	
0x0E0C	Boolean	01	FLOW03.HH	上上限报警	只读	FLOW03
0x0E0D	Boolean	01	FLOW03.Hi	上限报警	只读	
0x0E0E	Boolean	01	FLOW03.Lo	下限报警	只读	
0x0E0F	Boolean	01	FLOW03.LL	下下限报警	只读	
0x0E10	Boolean	01	FLOW03.RH	上升速率报警	只读	
0x0E11	Boolean	01	FLOW03.RL	下降速率报警	只读	
0x0E12	Boolean	01	FLOW04.HH	上上限报警	只读	FLOW04
0x0E13	Boolean	01	FLOW04.Hi	上限报警	只读	

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 23 页	共 45 页

0x0E14	Boolean	01	FLOW04.Lo	下限报警	只读	
0x0E15	Boolean	01	FLOW04.LL	下下限报警	只读	
0x0E16	Boolean	01	FLOW04.RH	上升速率报警	只读	
0x0E17	Boolean	01	FLOW04.RL	下降速率报警	只读	
0x0E18	Boolean	01	FLOW05.HH	上上限报警	只读	FLOW05
0x0E19	Boolean	01	FLOW05.Hi	上限报警	只读	
0x0E1A	Boolean	01	FLOW05.Lo	下限报警	只读	
0x0E1B	Boolean	01	FLOW05.LL	下下限报警	只读	
0x0E1C	Boolean	01	FLOW05.RH	上升速率报警	只读	
0x0E1D	Boolean	01	FLOW05.RL	下降速率报警	只读	
0x0E1E	Boolean	01	FLOW06.HH	上上限报警	只读	FLOW06
0x0E1F	Boolean	01	FLOW06.Hi	上限报警	只读	
0x0E20	Boolean	01	FLOW06.Lo	下限报警	只读	
0x0E21	Boolean	01	FLOW06.LL	下下限报警	只读	
0x0E22	Boolean	01	FLOW06.RH	上升速率报警	只读	
0x0E23	Boolean	01	FLOW06.RL	下降速率报警	只读	
0x0E24	Boolean	01	FLOW07.HH	上上限报警	只读	FLOW07
0x0E25	Boolean	01	FLOW07.Hi	上限报警	只读	
0x0E26	Boolean	01	FLOW07.Lo	下限报警	只读	
0x0E27	Boolean	01	FLOW07.LL	下下限报警	只读	
0x0E28	Boolean	01	FLOW07.RH	上升速率报警	只读	
0x0E29	Boolean	01	FLOW07.RL	下降速率报警	只读	
0x0E2A	Boolean	01	FLOW08.HH	上上限报警	只读	FLOW08
0x0E2B	Boolean	01	FLOW08.Hi	上限报警	只读	
0x0E2C	Boolean	01	FLOW08.Lo	下限报警	只读	
0x0E2D	Boolean	01	FLOW08.LL	下下限报警	只读	
0x0E2E	Boolean	01	FLOW08.RH	上升速率报警	只读	
0x0E2F	Boolean	01	FLOW08.RL	下降速率报警	只读	
0x0E30	Boolean	01	FLOW09.HH	上上限报警	只读	FLOW09
0x0E31	Boolean	01	FLOW09.Hi	上限报警	只读	
0x0E32	Boolean	01	FLOW09.Lo	下限报警	只读	
0x0E33	Boolean	01	FLOW09.LL	下下限报警	只读	
0x0E34	Boolean	01	FLOW09.RH	上升速率报警	只读	
0x0E35	Boolean	01	FLOW09.RL	下降速率报警	只读	
0x0E36	Boolean	01	FLOW10.HH	上上限报警	只读	FLOW10
0x0E37	Boolean	01	FLOW10.Hi	上限报警	只读	
0x0E38	Boolean	01	FLOW10.Lo	下限报警	只读	
0x0E39	Boolean	01	FLOW10.LL	下下限报警	只读	
0x0E3A	Boolean	01	FLOW10.RH	上升速率报警	只读	

0x0E3B	Boolean	01	FLOW10.RL	下降速率报警	只读	FLOW11
0x0E3C	Boolean	01	FLOW11.HH	上上限报警	只读	
0x0E3D	Boolean	01	FLOW11.Hi	上限报警	只读	
0x0E3E	Boolean	01	FLOW11.Lo	下限报警	只读	
0x0E3F	Boolean	01	FLOW11.LL	下下限报警	只读	
0x0E40	Boolean	01	FLOW11.RH	上升速率报警	只读	
0x0E41	Boolean	01	FLOW11.RL	下降速率报警	只读	
0x0E42	Boolean	01	FLOW12.HH	上上限报警	只读	FLOW12
0x0E43	Boolean	01	FLOW12.Hi	上限报警	只读	
0x0E44	Boolean	01	FLOW12.Lo	下限报警	只读	
0x0E45	Boolean	01	FLOW12.LL	下下限报警	只读	
0x0E46	Boolean	01	FLOW12.RH	上升速率报警	只读	
0x0E47	Boolean	01	FLOW12.RL	下降速率报警	只读	
0x0E48	Boolean	01	FLOW13.HH	上上限报警	只读	FLOW13
0x0E49	Boolean	01	FLOW13.Hi	上限报警	只读	
0x0E4A	Boolean	01	FLOW13.Lo	下限报警	只读	
0x0E4B	Boolean	01	FLOW13.LL	下下限报警	只读	
0x0E4C	Boolean	01	FLOW13.RH	上升速率报警	只读	
0x0E4D	Boolean	01	FLOW13.RL	下降速率报警	只读	
0x0E4E	Boolean	01	FLOW14.HH	上上限报警	只读	FLOW14
0x0E4F	Boolean	01	FLOW14.Hi	上限报警	只读	
0x0E50	Boolean	01	FLOW14.Lo	下限报警	只读	
0x0E51	Boolean	01	FLOW14.LL	下下限报警	只读	
0x0E52	Boolean	01	FLOW14.RH	上升速率报警	只读	
0x0E53	Boolean	01	FLOW14.RL	下降速率报警	只读	
0x0E54	Boolean	01	FLOW15.HH	上上限报警	只读	FLOW15
0x0E55	Boolean	01	FLOW15.Hi	上限报警	只读	
0x0E56	Boolean	01	FLOW15.Lo	下限报警	只读	
0x0E57	Boolean	01	FLOW15.LL	下下限报警	只读	
0x0E58	Boolean	01	FLOW15.RH	上升速率报警	只读	
0x0E59	Boolean	01	FLOW15.RL	下降速率报警	只读	
0x0E5A	Boolean	01	FLOW16.HH	上上限报警	只读	FLOW16
0x0E5B	Boolean	01	FLOW16.Hi	上限报警	只读	
0x0E5C	Boolean	01	FLOW16.Lo	下限报警	只读	
0x0E5D	Boolean	01	FLOW16.LL	下下限报警	只读	
0x0E5E	Boolean	01	FLOW16.RH	上升速率报警	只读	
0x0E5F	Boolean	01	FLOW16.RL	下降速率报警	只读	

WPR50B 最大支持 FLOW01~FLOW12

WPR70C 最大支持 FLOW01~FLOW16

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 25 页	共 45 页

TIM: 变量类型为 15

地址	数据类型	功能代码	变量	说明	访问权限
0x0F00	Boolean	01	TIM01	定时器	只读
0x0F01	Boolean	01	TIM02	定时器	只读
0x0F02	Boolean	01	TIM03	定时器	只读
0x0F03	Boolean	01	TIM04	定时器	只读
0x0F04	Boolean	01	TIM05	定时器	只读
0x0F05	Boolean	01	TIM06	定时器	只读
0x0F06	Boolean	01	TIM07	定时器	只读
0x0F07	Boolean	01	TIM08	定时器	只读

WPR50B 最大支持 TIM01~TIM08

WPR70C 最大支持 TIM01~TIM08

PID 回路 SV: 变量类型为 16

地址	数据类型	功能代码	变量	说明	访问权限
0x1000	Floating	03, 16	SV01	回路设定值	读写
0x1002	Floating	03, 16	SV02	回路设定值	读写
0x1004	Floating	03, 16	SV03	回路设定值	读写
0x1006	Floating	03, 16	SV04	回路设定值	读写

WPR50B 最大支持 SV01~SV04

WPR70C 最大支持 SV01~SV04

PID 回路 MV: 变量类型为 17

地址	数据类型	功能代码	变量	说明	访问权限
0x1100	Floating	03, 16	MV01	回路输出	读写
0x1102	Floating	03, 16	MV02	回路输出	读写
0x1104	Floating	03, 16	MV03	回路输出	读写
0x1106	Floating	03, 16	MV04	回路输出	读写

WPR50B 最大支持 MV01~MV04

WPR70C 最大支持 MV01~MV04

PID 回路偏差报警: 变量类型为 18

地址	数据类型	功能代码	变量	说明	访问权限
0x1200	Boolean	01	DH01	0: 无报警, 1: 有报警	只读
0x1201	Boolean	01	DH02	0: 无报警, 1: 有报警	只读
0x1202	Boolean	01	DH03	0: 无报警, 1: 有报警	只读
0x1203	Boolean	01	DH04	0: 无报警, 1: 有报警	只读

WPR50B 最大支持 DH01~DH04

WPR70C 最大支持 DH01~DH04

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 26 页	共 45 页

PID 回路状态：变量类型为 19

地址	数据类型	功能代码	变量	说明	权限
0x1300	Integer	03	PID01.State	1 自动, 2 手动, 3 自整定, 4 MV 跟踪	只读
0x1301	Integer	03	PID02.State	1 自动, 2 手动, 3 自整定, 4 MV 跟踪	只读
0x1302	Integer	03	PID03.State	1 自动, 2 手动, 3 自整定, 4 MV 跟踪	只读
0x1303	Integer	03	PID04.State	1 自动, 2 手动, 3 自整定, 4 MV 跟踪	只读

PID 回路手自动：变量类型为 20

地址	数据类型	功能代码	变量	说明	权限
0x1320	Boolean	05,15	PID01.A/M	0: 自动, 1: 手动	只写
0x1321	Boolean	05,15	PID02.A/M	0: 自动, 1: 手动	只写
0x1322	Boolean	05,15	PID03.A/M	0: 自动, 1: 手动	只写
0x1323	Boolean	05,15	PID04.A/M	0: 自动, 1: 手动	只写

PID 回路内外给定：变量类型为 21

地址	数据类型	功能代码	变量	说明	权限
0x1340	Boolean	05,15	PID01.L/R	0: 内给定, 1: 外给定	只写
0x1341	Boolean	05,15	PID02.L/R	0: 内给定, 1: 外给定	只写
0x1342	Boolean	05,15	PID03.L/R	0: 内给定, 1: 外给定	只写
0x1343	Boolean	05,15	PID04.L/R	0: 内给定, 1: 外给定	只写

PID 整定：变量类型为 22

地址	数据类型	功能代码	变量	说明	权限
0x1360	Boolean	05	PID01.Tune	0: 停止自整定, 1: 开始自整定	只写
0x1361	Boolean	05	PID02.Tune	0: 停止自整定, 1: 开始自整定	只写
0x1362	Boolean	05	PID03.Tune	0: 停止自整定, 1: 开始自整定	只写
0x1363	Boolean	05	PID04.Tune	0: 停止自整定, 1: 开始自整定	只写

PID 回路参数：变量类型为 23

地址	数据类型	功能代码	变量	说明	权限
0x1400	Integer	03、06、16	PID01.P	1000 = 100%	读写
0x1401	Integer	03、06、16	PID01.I	1000 = 100.0s	读写
0x1402	Integer	03、06、16	PID01.D	1000 = 100.0s	读写
0x1403	Integer	03、06、16	PID02.P	1000 = 100%	读写
0x1404	Integer	03、06、16	PID02.I	1000 = 100.0s	读写
0x1405	Integer	03、06、16	PID02.D	1000 = 100.0s	读写

0x1406	Integer	03、06、16	PID03.P	1000 = 100%	读写
0x1407	Integer	03、06、16	PID03.I	1000 = 100.0s	读写
0x1408	Integer	03、06、16	PID03.D	1000 = 100.0s	读写
0x1409	Integer	03、06、16	PID04.P	1000 = 100%	读写
0x140A	Integer	03、06、16	PID04.I	1000 = 100.0s	读写
0x140B	Integer	03、06、16	PID04.D	1000 = 100.0s	读写

0x1500 为 PID 保留

ONOFF 回路状态：变量类型为 24

地址	数据类型	功能代码	变量	说明	权限
0x1600	Integer	03	ONOFF01.State	1: 自动, 2: 手动	只读
0x1601	Integer	03	ONOFF02.State	1: 自动, 2: 手动	只读
0x1602	Integer	03	ONOFF03.State	1: 自动, 2: 手动	只读
0x1603	Integer	03	ONOFF04.State	1: 自动, 2: 手动	只读
0x1604	Integer	03	ONOFF05.State	1: 自动, 2: 手动	只读
0x1605	Integer	03	ONOFF06.State	1: 自动, 2: 手动	只读

ONOFF 手自动：变量类型为 25

地址	数据类型	功能代码	变量	说明	权限
0x1620	Boolean	05,15	ONOFF01.A/M	0: 自动, 1: 手动	只写
0x1621	Boolean	05,15	ONOFF02. A/M	0: 自动, 1: 手动	只写
0x1622	Boolean	05,15	ONOFF03. A/M	0: 自动, 1: 手动	只写
0x1623	Boolean	05,15	ONOFF04. A/M	0: 自动, 1: 手动	只写
0x1624	Boolean	05,15	ONOFF05. A/M	0: 自动, 1: 手动	只写
0x1625	Boolean	05,15	ONOFF06. A/M	0: 自动, 1: 手动	只写

ONOFF 内外给定：变量类型为 26

地址	数据类型	功能代码	变量	说明	权限
0x1640	Boolean	05,15	ONOFF01.L/R	0: 内给定, 1: 外给定	只写
0x1641	Boolean	05,15	ONOFF02. L/R	0: 内给定, 1: 外给定	只写
0x1642	Boolean	05,15	ONOFF03. L/R	0: 内给定, 1: 外给定	只写
0x1643	Boolean	05,15	ONOFF04. L/R	0: 内给定, 1: 外给定	只写
0x1644	Boolean	05,15	ONOFF05. L/R	0: 内给定, 1: 外给定	只写
0x1645	Boolean	05,15	ONOFF06. L/R	0: 内给定, 1: 外给定	只写

ONOFF 偏差报警：变量类型为 27

地址	数据类型	功能代码	变量	说明	权限
0x1660	Boolean	01	ONOFF01.DH	0: 无报警, 1: 有报警	只读

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 28 页	共 45 页

0x1661	Boolean	01	ONOFF02. DH	0: 无报警, 1: 有报警	只读
0x1662	Boolean	01	ONOFF03. DH	0: 无报警, 1: 有报警	只读
0x1663	Boolean	01	ONOFF04. DH	0: 无报警, 1: 有报警	只读
0x1664	Boolean	01	ONOFF05. DH	0: 无报警, 1: 有报警	只读
0x1665	Boolean	01	ONOFF06. DH	0: 无报警, 1: 有报警	只读

ONOFF SV: 变量类型为 28

地址	数据类型	功能代码	变量	说明	权限
0x1700	Floating	03,16	ONOFF01.SV	设定值	读写
0x1702	Floating	03,16	ONOFF02. SV	设定值	读写
0x1704	Floating	03,16	ONOFF03. SV	设定值	读写
0x1706	Floating	03,16	ONOFF04. SV	设定值	读写
0x1708	Floating	03,16	ONOFF05. SV	设定值	读写
0x170A	Floating	03,16	ONOFF06. SV	设定值	读写

ONOFF MV: 变量类型为 29

地址	数据类型	功能代码	变量	说明	权限
0x1720	Boolean	01,05,15	ONOFF01.MV	0: OFF, 1: ON	读写
0x1721	Boolean	01,05,15	ONOFF02. MV	0: OFF, 1: ON	读写
0x1722	Boolean	01,05,15	ONOFF03. MV	0: OFF, 1: ON	读写
0x1723	Boolean	01,05,15	ONOFF04. MV	0: OFF, 1: ON	读写
0x1724	Boolean	01,05,15	ONOFF05. MV	0: OFF, 1: ON	读写
0x1725	Boolean	01,05,15	ONOFF06. MV	0: OFF, 1: ON	读写

//-----保留一些空间-----

PROG 回路状态: 变量类型为 30

地址	数据类型	功能代码	变量	说明	权限
0x3200	Integer	03	PROG01.State	1 复位, 2 运行, 3 保持, 4 快进, 5 等待, 6 结束	只读
0x3201	Integer	03	PROG02.State	1 复位, 2 运行, 3 保持, 4 快进, 5 等待, 6 结束	只读
0x3202	Integer	03	PROG03.State	1 复位, 2 运行, 3 保持, 4 快进, 5 等待, 6 结束	只读

PROG 工艺序号: 变量类型为 31

地址	数据类型	功能代码	变量	说明	权限
0x3220	Integer	03、06、16	PROG01.File	工艺序号 0, 1, ...	读写
0x3221	Integer	03、06、16	PROG02.File	工艺序号 0, 1, ...	读写
0x3222	Integer	03、06、16	PROG03.File	工艺序号 0, 1, ...	读写

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 29 页	共 45 页

		6			
--	--	---	--	--	--

PROG 保持/运行：变量类型为 32

地址	数据类型	功能代码	变量	说明	权限
0x3300	Boolean	05,15	PROG01.R/H	0：保持，1：运行	只写
0x3301	Boolean	05,15	PROG02.R/H	0：保持，1：运行	只写
0x3302	Boolean	05,15	PROG03.R/H	0：保持，1：运行	只写

PROG 复位：变量类型为 33

地址	数据类型	功能代码	变量	说明	权限
0x3320	Boolean	05,15	PROG01.Reset	0：无效，1：复位	只写
0x3321	Boolean	05,15	PROG02.Reset	0：无效，1：复位	只写
0x3322	Boolean	05,15	PROG03.Reset	0：无效，1：复位	只写

PROG 快进：变量类型为 34

地址	数据类型	功能代码	变量	说明	权限
0x3340	Boolean	05,15	PROG01.FastRun	0：无效，1：快进	只写
0x3341	Boolean	05,15	PROG02.FastRun	0：无效，1：快进	只写
0x3342	Boolean	05,15	PROG03.FastRun	0：无效，1：快进	只写

PROG 事件输出：变量类型为 35

地址	数据类型	功能代码	变量	说明	权限
0x3400	Boolean	01	PROG01.DO1	程序 1 的第 1 个事件输出	只读
0x3401	Boolean	01	PROG01.DO2	程序 1 的第 2 个事件输出	只读
0x3402	Boolean	01	PROG01.DO3	程序 1 的第 3 个事件输出	只读
0x3403	Boolean	01	PROG01.DO4	程序 1 的第 4 个事件输出	只读
0x3404	Boolean	01	PROG01.DO5	程序 1 的第 5 个事件输出	只读
0x3405	Boolean	01	PROG01.DO6	程序 1 的第 6 个事件输出	只读
0x3406	Boolean	01	PROG01.DO7	程序 1 的第 7 个事件输出	只读
0x3407	Boolean	01	PROG01.DO8	程序 1 的第 8 个事件输出	只读
0x3410	Boolean	01	PROG02.DO1	程序 2 的第 1 个事件输出	只读
0x3411	Boolean	01	PROG02.DO2	程序 2 的第 2 个事件输出	只读
0x3412	Boolean	01	PROG02.DO3	程序 2 的第 3 个事件输出	只读
0x3413	Boolean	01	PROG02.DO4	程序 2 的第 4 个事件输出	只读
0x3414	Boolean	01	PROG02.DO5	程序 2 的第 5 个事件输出	只读
0x3415	Boolean	01	PROG02.DO6	程序 2 的第 6 个事件输出	只读
0x3416	Boolean	01	PROG02.DO7	程序 2 的第 7 个事件输出	只读
0x3417	Boolean	01	PROG02.DO8	程序 2 的第 8 个事件输出	只读
0x3420	Boolean	01	PROG03.DO1	程序 3 的第 1 个事件输出	只读
0x3421	Boolean	01	PROG03.DO2	程序 3 的第 2 个事件输出	只读
0x3422	Boolean	01	PROG03.DO3	程序 3 的第 3 个事件输出	只读
0x3423	Boolean	01	PROG03.DO4	程序 3 的第 4 个事件输出	只读
0x3424	Boolean	01	PROG03.DO5	程序 3 的第 5 个事件输出	只读
0x3425	Boolean	01	PROG03.DO6	程序 3 的第 6 个事件输出	只读

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 30 页	共 45 页

0x3426	Boolean	01	PROG03.DO7	程序 3 的第 7 个事件输出	只读
0x3427	Boolean	01	PROG03.DO8	程序 3 的第 8 个事件输出	只读

PROG 段号：变量类型为 36

地址	数据类型	功能代码	变量	说明	权限
0x3500	Integer	03、06、16	PROG01.Seg	正常段号 1、2...	读写
0x3501	Integer	03、06、16	PROG02.Seg	正常段号 1、2...	读写
0x3502	Integer	03、06、16	PROG03.Seg	正常段号 1、2...	读写

PROG 段总时间：变量类型为 37

地址	数据类型	功能代码	变量	说明	权限
0x3520	Integer	03	PROG01.AllTicks	当前段的总时间（s）（高 16 位）	只读
0x3521	Integer	03		当前段的总时间（s）（低 16 位）	只读
0x3522	Integer	03	PROG02.AllTicks	当前段的总时间（s）（高 16 位）	只读
0x3523	Integer	03		当前段的总时间（s）（低 16 位）	只读
0x3524	Integer	03	PROG03.AllTicks	当前段的总时间（s）（高 16 位）	只读
0x3525	Integer	03		当前段的总时间（s）（低 16 位）	只读

PROG 段运行时间：变量类型为 38

地址	数据类型	功能代码	变量	说明	权限
0x3540	Integer	03	PROG01. CurTicks	当前段已运行时间（s）（高 16 位）	只读
0x3541	Integer	03		当前段已运行时间（s）（低 16 位）	只读
0x3542	Integer	03	PROG02. CurTicks	当前段已运行时间（s）（高 16 位）	只读
0x3543	Integer	03		当前段已运行时间（s）（低 16 位）	只读
0x3544	Integer	03	PROG03. CurTicks	当前段已运行时间（s）（高 16 位）	只读
0x3545	Integer	03		当前段已运行时间（s）（低 16 位）	只读

PROG PV：变量类型为 39

地址	数据类型	功能代码	变量	说明	权限
0x3600	Floating	03	PROG01.PV	程序测量值	只读
0x3602	Floating	03	PROG02.PV	程序测量值	只读
0x3604	Floating	03	PROG03.PV	程序测量值	只读

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 31 页	共 45 页

PROG MV: 变量类型为 40

地址	数据类型	功能代码	变量	说明	权限
0x3620	Floating	03	PROG01.OUT	程序输出值	只读
0x3622	Floating	03	PROG02.OUT	程序输出值	只读
0x3624	Floating	03	PROG03.OUT	程序输出值	只读

CONB: 变量类型为 41

地址	数据类型	功能代码	变量	说明	访问权限
0x3700	Boolean	01、05、15	CONB01	布尔常量	读写
0x3701	Boolean	01、05、15	CONB02	布尔常量	读写
0x3702	Boolean	01、05、15	CONB03	布尔常量	读写
0x3703	Boolean	01、05、15	CONB04	布尔常量	读写
0x3704	Boolean	01、05、15	CONB05	布尔常量	读写
0x3705	Boolean	01、05、15	CONB06	布尔常量	读写
0x3706	Boolean	01、05、15	CONB07	布尔常量	读写
0x3707	Boolean	01、05、15	CONB08	布尔常量	读写
0x3708	Boolean	01、05、15	CONB09	布尔常量	读写
0x3709	Boolean	01、05、15	CONB10	布尔常量	读写
0x370A	Boolean	01、05、15	CONB11	布尔常量	读写
0x370B	Boolean	01、05、15	CONB12	布尔常量	读写
0x370C	Boolean	01、05、15	CONB13	布尔常量	读写
0x370D	Boolean	01、05、15	CONB14	布尔常量	读写
0x370E	Boolean	01、05、15	CONB15	布尔常量	读写
0x370F	Boolean	01、05、15	CONB16	布尔常量	读写
0x3710	Boolean	01、05、15	CONB17	布尔常量	读写
0x3711	Boolean	01、05、15	CONB18	布尔常量	读写
0x3712	Boolean	01、05、15	CONB19	布尔常量	读写
0x3713	Boolean	01、05、15	CONB20	布尔常量	读写
0x3714	Boolean	01、05、15	CONB21	布尔常量	读写
0x3715	Boolean	01、05、15	CONB22	布尔常量	读写
0x3716	Boolean	01、05、15	CONB23	布尔常量	读写
0x3717	Boolean	01、05、15	CONB24	布尔常量	读写
.....
.....
0x373F	Boolean	01、05、15	CONB48	布尔常量	读写

WPR50B 最大支持 CONB01~CONB24

WPR70C 最大支持 CONB01~CONB48

CONI: 变量类型为 42

地址	数据类型	功能代码	变量	说明	访问权限
----	------	------	----	----	------

迅鹏	TITLE	文件版本	WPR50B/WPR70C	
	Modbus 通讯协议		A	第 32 页

0x3800	Integer	03、06、16	CONI01	整型常量	读写
0x3801	Integer	03、06、16	CONI02	整型常量	读写
0x3802	Integer	03、06、16	CONI03	整型常量	读写
0x3803	Integer	03、06、16	CONI04	整型常量	读写
0x3804	Integer	03、06、16	CONI05	整型常量	读写
0x3805	Integer	03、06、16	CONI06	整型常量	读写
0x3806	Integer	03、06、16	CONI07	整型常量	读写
0x3807	Integer	03、06、16	CONI08	整型常量	读写
0x3808	Integer	03、06、16	CONI09	整型常量	读写
0x3809	Integer	03、06、16	CONI10	整型常量	读写
0x380A	Integer	03、06、16	CONI11	整型常量	读写
0x380B	Integer	03、06、16	CONI12	整型常量	读写
0x380C	Integer	03、06、16	CONI13	整型常量	读写
0x380D	Integer	03、06、16	CONI14	整型常量	读写
0x380E	Integer	03、06、16	CONI15	整型常量	读写
0x380F	Integer	03、06、16	CONI16	整型常量	读写
0x3810	Integer	03、06、16	CONI17	整型常量	读写
0x3811	Integer	03、06、16	CONI18	整型常量	读写
0x3812	Integer	03、06、16	CONI19	整型常量	读写
0x3813	Integer	03、06、16	CONI20	整型常量	读写
0x3814	Integer	03、06、16	CONI21	整型常量	读写
0x3815	Integer	03、06、16	CONI22	整型常量	读写
0x3816	Integer	03、06、16	CONI23	整型常量	读写
0x3817	Integer	03、06、16	CONI24	整型常量	读写
.....
.....
0x382F	Integer	03、06、16	CONI48	整型常量	读写

WPR50B 最大支持 CONI01~CONI24

WPR70C 最大支持 CONI01~CONI48

CONF：变量类型为 43

地址	数据类型	功能代码	变量	说明	访问权限
0x3900	Floating	03、16	CONF01	浮点常量	读写
0x3902	Floating	03、16	CONF02	浮点常量	读写
0x3904	Floating	03、16	CONF03	浮点常量	读写
0x3906	Floating	03、16	CONF04	浮点常量	读写
0x3908	Floating	03、16	CONF05	浮点常量	读写
0x390A	Floating	03、16	CONF06	浮点常量	读写
0x390C	Floating	03、16	CONF07	浮点常量	读写

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 33 页	共 45 页

0x390E	Floating	03、16	CONF08	浮点常量	读写
0x3910	Floating	03、16	CONF09	浮点常量	读写
0x3912	Floating	03、16	CONF10	浮点常量	读写
0x3914	Floating	03、16	CONF11	浮点常量	读写
0x3916	Floating	03、16	CONF12	浮点常量	读写
0x3918	Floating	03、16	CONF13	浮点常量	读写
0x391A	Floating	03、16	CONF14	浮点常量	读写
0x391C	Floating	03、16	CONF15	浮点常量	读写
0x391E	Floating	03、16	CONF16	浮点常量	读写
0x3920	Floating	03、16	CONF17	浮点常量	读写
0x3922	Floating	03、16	CONF18	浮点常量	读写
0x3924	Floating	03、16	CONF19	浮点常量	读写
0x3926	Floating	03、16	CONF20	浮点常量	读写
0x3928	Floating	03、16	CONF21	浮点常量	读写
0x392A	Floating	03、16	CONF22	浮点常量	读写
0x392C	Floating	03、16	CONF23	浮点常量	读写
0x392E	Floating	03、16	CONF24	浮点常量	读写
.....
.....
0x395E	Floating	03、16	CONF48	浮点常量	读写

WPR50B 最大支持 CONF01~CONF24

WPR70C 最大支持 CONF01~CONF48

3.3.3.2 组态变量

CONI: 变量类型为 0

地址	数据类型	功能代码	变量	说明	访问权限
0x8000	Integer	03、06、16	CONI01	整型常量	读写
0x8001	Integer	03、06、16	CONI02	整型常量	读写
0x8002	Integer	03、06、16	CONI03	整型常量	读写
0x8003	Integer	03、06、16	CONI04	整型常量	读写
0x8004	Integer	03、06、16	CONI05	整型常量	读写
0x8005	Integer	03、06、16	CONI06	整型常量	读写
0x8006	Integer	03、06、16	CONI07	整型常量	读写
0x8007	Integer	03、06、16	CONI08	整型常量	读写
0x8008	Integer	03、06、16	CONI09	整型常量	读写
0x8009	Integer	03、06、16	CONI10	整型常量	读写
0x800A	Integer	03、06、16	CONI11	整型常量	读写

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 34 页	共 45 页

0x800B	Integer	03、06、16	CONI12	整型常量	读写
0x800C	Integer	03、06、16	CONI13	整型常量	读写
0x800D	Integer	03、06、16	CONI14	整型常量	读写
0x800E	Integer	03、06、16	CONI15	整型常量	读写
0x800F	Integer	03、06、16	CONI16	整型常量	读写
0x8010	Integer	03、06、16	CONI17	整型常量	读写
0x8011	Integer	03、06、16	CONI18	整型常量	读写
0x8012	Integer	03、06、16	CONI19	整型常量	读写
0x8013	Integer	03、06、16	CONI20	整型常量	读写
0x8014	Integer	03、06、16	CONI21	整型常量	读写
0x8015	Integer	03、06、16	CONI22	整型常量	读写
0x8016	Integer	03、06、16	CONI23	整型常量	读写
0x8017	Integer	03、06、16	CONI24	整型常量	读写
.....
.....
0x802F	Integer	03、06、16	CONI48	整型常量	读写

WPR50B 最大支持 CONI01~CONI24

WPR70C 最大支持 CONI01~CONI48

CONB：变量类型为 1

地址	数据类型	功能代码	变量	说明	访问权限
0x8100	Boolean	01、05、15	CONB01	布尔常量	读写
0x8101	Boolean	01、05、15	CONB02	布尔常量	读写
0x8102	Boolean	01、05、15	CONB03	布尔常量	读写
0x8103	Boolean	01、05、15	CONB04	布尔常量	读写
0x8104	Boolean	01、05、15	CONB05	布尔常量	读写
0x8105	Boolean	01、05、15	CONB06	布尔常量	读写
0x8106	Boolean	01、05、15	CONB07	布尔常量	读写
0x8107	Boolean	01、05、15	CONB08	布尔常量	读写
0x8108	Boolean	01、05、15	CONB09	布尔常量	读写
0x8109	Boolean	01、05、15	CONB10	布尔常量	读写
0x810A	Boolean	01、05、15	CONB11	布尔常量	读写
0x810B	Boolean	01、05、15	CONB12	布尔常量	读写
0x810C	Boolean	01、05、15	CONB13	布尔常量	读写
0x810D	Boolean	01、05、15	CONB14	布尔常量	读写
0x810E	Boolean	01、05、15	CONB15	布尔常量	读写
0x810F	Boolean	01、05、15	CONB16	布尔常量	读写
0x8110	Boolean	01、05、15	CONB17	布尔常量	读写
0x8111	Boolean	01、05、15	CONB18	布尔常量	读写

0x8112	Boolean	01、05、15	CONB19	布尔常量	读写
0x8113	Boolean	01、05、15	CONB20	布尔常量	读写
0x8114	Boolean	01、05、15	CONB21	布尔常量	读写
0x8115	Boolean	01、05、15	CONB22	布尔常量	读写
0x8116	Boolean	01、05、15	CONB23	布尔常量	读写
0x8117	Boolean	01、05、15	CONB24	布尔常量	读写
.....
.....
0x812F	Boolean	01、05、15	CONB48	布尔常量	读写

WPR50B 最大支持 CONB01~CONB24

WPR70C 最大支持 CONB01~CONB48

CONF: 变量类型为 2

地址	数据类型	功能代码	变量	说明	访问权限
0x8200	Floating	03、16	CONF01	浮点常量	读写
0x8202	Floating	03、16	CONF02	浮点常量	读写
0x8204	Floating	03、16	CONF03	浮点常量	读写
0x8206	Floating	03、16	CONF04	浮点常量	读写
0x8208	Floating	03、16	CONF05	浮点常量	读写
0x820A	Floating	03、16	CONF06	浮点常量	读写
0x820C	Floating	03、16	CONF07	浮点常量	读写
0x820E	Floating	03、16	CONF08	浮点常量	读写
0x8210	Floating	03、16	CONF09	浮点常量	读写
0x8212	Floating	03、16	CONF10	浮点常量	读写
0x8214	Floating	03、16	CONF11	浮点常量	读写
0x8216	Floating	03、16	CONF12	浮点常量	读写
0x8218	Floating	03、16	CONF13	浮点常量	读写
0x821A	Floating	03、16	CONF14	浮点常量	读写
0x821C	Floating	03、16	CONF15	浮点常量	读写
0x821E	Floating	03、16	CONF16	浮点常量	读写
0x8220	Floating	03、16	CONF17	浮点常量	读写
0x8222	Floating	03、16	CONF18	浮点常量	读写
0x8224	Floating	03、16	CONF19	浮点常量	读写
0x8226	Floating	03、16	CONF20	浮点常量	读写
0x8228	Floating	03、16	CONF21	浮点常量	读写
0x822A	Floating	03、16	CONF22	浮点常量	读写
0x822C	Floating	03、16	CONF23	浮点常量	读写
0x822E	Floating	03、16	CONF24	浮点常量	读写
.....

.....
0x825E	Floating	03、16	CONF48	浮点常量	读写

WPR50B 最大支持 CONF01~CONF24

WPR70C 最大支持 CONF01~CONF48

3.4 命令解释

3.4.1 01H 读取一组开关量

问询帧

名称	设备地址	0x01	起始地址	数据数量	CRC
字节	1	1	2	2	2

返回帧

名称	设备地址	0x01	返回字节数 N	返回数据	CRC
字节	1	1	1	N×1	2

特别说明

问询帧：数据数量表示从起始地址开始读多少个开关量，数据范围（1~200）。
返回帧：每一个字节表示 8 个开关量的值，每一位为 0 表示对应的开关断开，为 1 表示闭合。
如果询问的开关量不是 8 的整倍数，那么最后一个字节的高位部分无意义，置为 0。

范例

从起始地址 0x0700 读取 12 个开关量的状态。
01 07 00 00 0C —— 01 02 CD 0B
CD 表示开关量的 8~1，0B 表示开关量的 16~9。对于开关量 8~1 的状态依次为 ON-ON-OFF-OFF-ON-ON-OFF-ON。

3.4.2 03H 读一个或多个寄存器

问询帧

名称	设备地址	0x03	起始地址	数据数量	CRC
字节	1	1	2	2	2

返回帧

名称	设备地址	0x03	返回字节数 N	返回数据	CRC
----	------	------	---------	------	-----

字节	1	1	1	N×1	2
----	---	---	---	-----	---

特别说明

问询帧：数据数量表示读取寄存器的个数，数据范围（1～50）。
返回帧：每一个寄存器的高字节在前，低字节在后。（浮点数数据按浮点格式排列）

范例

从起始地址 0x0001 读取 2 个寄存器的值。
03 00 01 00 02 — 03 04 01 02 03 04
第 1 个寄存器的值为 0x0102，第 2 个寄存器的值为 0x0304。

3.4.3 05H 写一个开关量

问询帧

名称	设备地址	0x05	起始地址	数据值	CRC
字节	1	1	2	2	2

返回帧

名称	设备地址	0x05	起始地址	数据值	CRC
字节	1	1	2	2	2

特别说明

问询帧：此命令只能下置一个开关量的状态。数据值须为【0xFF00】或【0x0000】，【0xFF00】表示闭合，【0x0000】表示断开，其它数值非法。
返回帧：如果成功把计算机发送的命令原样返回，否则异常应答。

范例

将 1 写入起始地址 0x8100。
05 81 00 FF 00 — 05 81 00 FF 00
强制第 1 个开关量变成闭合状态。

3.4.4 06H 写一个寄存器

问询帧

名称	设备地址	0x06	起始地址	数据值	CRC
----	------	------	------	-----	-----

返回帧

字节	1	1	2	2	2
名称	设备地址	0x06	起始地址	数据值	CRC
字节	1	1	2	2	2

由于浮点型数据必须要 4 个字节，故浮点型数据不支持该命令。

特别说明

问询帧：此命令只能下置一个寄存器的值。每一个寄存器的高字节在前，低字节在后。

返回帧：如果成功把计算机发送的命令原样返回，否则异常应答。

范例

写 1 个寄存器，写入地址 0x8000。

06 80 00 01 02 — 06 80 00 01 02

将值 0x0102 写入地址为 0x8000 的寄存器。

3.4.5 0FH 写多个开关量

问询帧

名称	设备地址	0x0F	起始地址	数据数量 N	字节数	数据值	CRC
字节	1	1	2	2	1	(N+7) / 8	2

返回帧

名称	设备地址	0x0F	起始地址	数据数量 N	CRC
字节	1	1	2	2	2

特别说明

问询帧：此命令可写多个开关量的状态。数据数量的范围（1~200）。下置的数据每一个字节表示 8 个开关量的值，每一位为 0 表示对应的开关断开，为 1 表示闭合。如果下置的开关量不是 8 的整倍数，那么最后一个字节的高位部分无意义，置为 0。

返回帧：如果成功把计算机发送的命令除字节数和数据值外原样返回，否则异常应答。

范例

写入从起始地址 0x8100 开始的 10 个开关状态。

0F 81 00 00 0A 02 CD 03 — 0F 81 00 00 0A

CD 表示开关量的 8~1，03 表示开关量的 16~9。对于开关量 8~1 写入的状态依次为 ON-ON-OFF-OFF-ON-ON-OFF-ON。

3.4.6 10H 写多个寄存器

问询帧

名称	设备地址	0x10	起始地址	数据数量 N	字节数	数据值	CRC
字节	1	1	2	2	1	N×2	2

由于浮点型数据必须要 4 个字节，故写浮点型数据时，必须保证数据数量为偶数。

返回帧

名称	设备地址	0x10	起始地址	数据数量 N	CRC
字节	1	1	2	2	2

特别说明

问询帧：数据数量表示写入寄存器的个数，数据范围（1~100）。每一个寄存器用 2 个字节表示。高字节在前，低字节在后。

当写入数据表示浮点数时，由两个寄存器共同表示一个浮点数，该 4 字节数据的字节顺序遵循浮点格式的设定

返回帧：如果成功把计算机发送的命令除字节数和数据值外原样返回，否则异常应答。

范例

写入从起始地址 0x8000 开始的 2 个寄存器的值。

10 80 00 00 02 04 01 02 03 04 —— 10 80 00 00 02

第 1 个寄存器被写成 0x0102，第 2 个寄存器被写成 0x0304。

从起始地址 0x8200 开始写入一个浮点数。

10 82 00 00 02 04 42 48 00 00 —— 10 82 00 00 02

在地址 0x8200 写入值为 50.0 的浮点数（3210 模式）。

附录 A 异常代码

通讯异常的回应格式如下

名称	设备地址	功能代码	异常代码	CRC
字节	1	1	1	2

功能代码：0x80+主机下发的功能代码。如主机下发的功能代码是 01，则异常应答的功能代码就是 0x81；如主机下发的功能代码是 03，则异常应答的功能代码就是 0x83。以次类推。

异常代码：表示通讯出错的类型，具体请参见下表。

异常代码	名称	解 释
01H	功能代码非法	从机接收的是一个不能执行功能代码。
02H	数据地址非法	1. 数据的地址从机无法识别 2. 数据地址和数据数量合成的地址无效
03H	数据数值非法	1. 数据数量超出范围 2. 数据长度错误 3. 数据数值非法
04H	从机设备故障	从机执行主机请求时出现不可恢复的错误。
05H	确认	从机已接收请求处理数据，但需要较长的处理时间，为避免主机出现超时错误而发送该确认响应。主机以此再发送一个“查询程序完成”未决定从机是否已完成处理。
06H	从机设备忙	从机正忙于处理一个长时程序命令，请 求主机在从机空闲时发送信息。
07H	否定	从机不能执行查询要求的程序功能时，该代码使用十进制 13 或 14 代码，向主机返回一个“不成功的编程请求”信息。主机应请求诊断从机的错误信息。
08H	内存奇偶校验错误	从机读扩展内存中的数据时，发现有奇偶校验错误，主机按从机的要求重新发送数据请求。

附录 B CRC 检测

1. CRC 校验

使用 RTU 模式，消息包括了一基于 CRC 方法的错误检测域。CRC 域检测了整个消息的内容。

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 41 页	共 45 页

CRC 域是两个字节，包含一 16 位的二进制值。它由传输设备计算后加入到消息中。接收设备重新计算收到消息的 CRC，并与接收到的 CRC 域中的值比较，如果两值不同，则有误。

CRC 是先调入一值是全“1”的 16 位寄存器，然后调用一过程将消息中连续的 8 位字节各当前寄存器中的值进行处理。仅每个字符中的 8Bit 数据对 CRC 有效，起始位和停止位以及奇偶校验位均无效。

CRC 产生过程中，每个 8 位字符都单独和寄存器内容相或（OR），结果向最低有效位方向移动，最高有效位以 0 填充。LSB 被提取出来检测，如果 LSB 为 1，寄存器单独和预置的值或一下，如果 LSB 为 0，则不进行。整个过程要重复 8 次。在最后一位（第 8 位）完成后，下一个 8 位字节又单独和寄存器的当前值相或。最终寄存器中的值，是消息中所有的字节都执行之后的 CRC 值。

2. 函数实现

CRC 简单函数如下：

```
unsigned short CRC16(unsigned char *puchMsg, unsigned short usDataLen)
{
    unsigned char uchCRChi = 0xFF ; /* 高CRC字节初始化 */
    unsigned char uchCRCLo = 0xFF ; /* 低CRC 字节初始化 */
    unsigned uIndex ; /* CRC循环中的索引 */
    while (usDataLen-- /* 传输消息缓冲区 */)
    {
        uIndex = uchCRChi ^ *puchMsgg++ ; /* 计算CRC */
        uchCRChi = uchCRCLo ^ auchCRChi[uIndex] ;
        uchCRCLo = auchCRCLo[uIndex] ;
    }
    return (uchCRChi << 8 | uchCRCLo) ;
}

/* CRC 高位字节值表 */
static unsigned char auchCRChi[] = {
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
    0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1,
    0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1,
    0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40,
```

```
0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1,
0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1,
0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
0x80, 0x41, 0x00, 0xC1, 0x81, 0x40
} ;
```

/* CRC低位字节值表*/

```
static char auchCRCLo[] = {
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06,
    0x07, 0xC7, 0x05, 0xC5, 0xC4, 0x04, 0xCC, 0x0C, 0x0D, 0xCD,
    0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09,
    0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A,
    0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC, 0x14, 0xD4,
    0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3,
    0xF2, 0x32, 0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4,
    0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A,
    0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29,
    0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF, 0x2D, 0xED,
    0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60,
    0x61, 0xA1, 0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67,
    0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F,
    0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68,
    0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA, 0xBE, 0x7E,
    0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
    0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71,
    0x70, 0xB0, 0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92,
    0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C,
    0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B,
    0x99, 0x59, 0x58, 0x98, 0x88, 0x48, 0x49, 0x89, 0x4B, 0x8B,
```

0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42,
0x43, 0x83, 0x41, 0x81, 0x80, 0x40 };

迅鹏	TITLE Modbus 通讯协议	文件版本 A	WPR50B/WPR70C	
			第 44 页	共 45 页

附录 C 通讯格式

01H读一组开关量

设备地址	0x01	起始地址	数据数量	CRC
1	1	2	2	2

01H返回帧

设备地址	0x01	返回字节数 N	返回数据	CRC
1	1	1	N×1	2

03H读一个或多个寄存器

设备地址	0x03	起始地址	数据数量	CRC
1	1	2	2	2

03H返回帧

设备地址	0x03	返回字节数 N	返回数据	CRC
1	1	1	N×1	2

05H写一个开关量

设备地址	0x05	起始地址	数据值	CRC
1	1	2	2	2

05H返回帧

设备地址	0x05	起始地址	数据值	CRC
1	1	2	2	2

06H写一个寄存器

设备地址	0x06	起始地址	数据值	CRC
1	1	2	2	2

06H返回帧

设备地址	0x06	起始地址	数据值	CRC
1	1	2	2	2

0FH写多个开关量

设备地址	0x0F	起始地址	数据数量 N	字节数	数据值	CRC
1	1	2	2	1	(N+7) / 8	2

0FH返回帧

设备地址	0x0F	起始地址	数据数量 N	CRC
1	1	2	2	2

10H写多个寄存器

设备地址	0x10	起始地址	数据数量 N	字节数	数据值	CRC
1	1	2	2	1	N×2	2

10H返回帧

设备地址	0x10	起始地址	数据数量 N	CRC
1	1	2	2	2

异常应答

设备地址	功能代码	异常代码	CRC
1	1	1	2