

Modbusprotocol communication protocol

The register address and corresponding function code table are as follows:

Register address	name		Function	Function code 0X03	Function code 0X06	Function code 0X0A	Function code 0X0B	Function code 0X0C	Assignment range
0X0000	Remote	Mode	Remote operation mode 0:Local operation 1: Remote operation						0X0000~0X0001
0X0001	CH1	Vmax	CH1 Maximum voltage setting 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X0002	CH2	Vmax	CH2 Maximum voltage setting 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X0003	CH3	Vmax	CH3 Maximum voltage setting 0~6V						0X0000~0X1770
0X0004	CH1	Amax	CH1 Maximum current setting 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X0005	CH2	Amax	CH1 Maximum current setting 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X0006	CH3	Amax	CH3 Maximum current setting 0~3.1A						0X0000~0X0C1C
0X0007	CH1	Vset	CH1 setting voltage 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X0008	CH1	Aset	CH1 setting current 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X0009	CH2	Vset	CH2 setting voltage 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X000A	CH2	Aset	CH2 setting current 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X000B	CH3	Vset	CH3 setting voltage 0~6V						0X0000~0X1770
0X000C	CH3	Aset	CH3setting current 0~3.1A						0X0000~0X0C1C
0X000D	CH1	OVPset	CH1 overvoltage setting 0~32V / 0~62V						0X0000~0X7D00 / 0X0000~0XF230
0X000E	CH2	OVPset	CH2 over voltage setting 0~32V / 0~62V						0X0000~0X7D00 / 0X0000~0XF230
0X000F	CH3	OVPset	CH3 over voltage setting 0~7V						0X0000~0X1B58
0X0010	CH1	OutTimer	CH1 timing output time						0X0000~0X8CA0
0X0011	CH2	OutTimer	CH2 timing output time						0X0000~0X8CA0
0X0012	CH2	OutTimer	CH3 timing output time						0X0000~0X8CA0
0X0013	CH1	TIM On/OFF	CH1 Timing output state setting 0:close 1:open						0X0000~0X0001
0X0014	CH2	TIM On/OFF	CH2Timing output state setting 0:close 1:open						0X0000~0X0001
0X0015	CH3	TIM On/OFF	CH3Timing output state setting 0:close 1:open						0X0000~0X0001
0X0016		OutState	Power-on output status 0:output close						0X0000~0X0001

			1:Maintain the state before shutdown						
0X0017	OutParamState		Boot parameter status 0:Factory default 1:Pre-shutdown parameters are maintained						0X0000~0X0001
0X0018	KnobState		Code switch status 0:unlock 1:lock						0X0000~0X0001
0X0019	BuzzerState		Buzzer status 0:close 1:open						0X0000~0X0001
0X001A	MemoryGroup		storage group						0X0001~0X0004
0X001B	LockState		keyboard lock status 0:unlock 1:lock						0X0000~0X0001
0X001C	CombModeState		Combined state 0:independent 1:serial 2:parallel 3:tracking						0X0000~0X0003
0X001D	CombChunnelState		Combined channel 0:CH1+CH2 1:CH2+CH3 2:CH1+CH2+CH3						0X0000~0X0002
0X001E	CH1	On/Off	CH1Channel output state setting 0:close 1:output						0X0000~0X0001
0X001F	CH2	On/Off	CH2Channel output state setting 0:close 1:output						0X0000~0X0001
0X0020	CH3	On/Off	CH3Channel output state setting 0:close 1:output						0X0000~0X0001
0X0021	ALL	On/Off	Power total output state setting 0:close 1:output						0X0000~0X0001
0X0022	CH1	Vout	CH1 actual voltage 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X0023	CH1	Aout	CH1 actual current 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X0024	CH1	CV/CC	CH1 Constant voltage and constant current status flag 0:CV 1:CC						0X0000~0X0001
0X0025	CH2	Vout	CH2actual voltage 0~31V / 0~61V						0X0000~0X7918 / 0X0000~0XEE48
0X0026	CH2	Aout	CH2actual current 0~3.1A / 0~6.1A						0X0000~0X0C1C / 0X0000~0X17D4
0X0027	CH2	CV/CC	CH2Constant voltage and constant current status flag 0:CV 1:CC						0X0000~0X0001
0X0028	CH3	Vout	CH3 actual voltage 0~6V						0X0000~0X1770
0X0029	CH3	Aout	CH3 actual current 0~3.1A						0X0000~0X0C1C
0X002A	CH3	CV/CC	CH3Constant voltage and constant current status flag 0:CV 1:CC						0X0000~0X0001
0X002B	CAL	Year	Calibration Date: Calibration Year						0X0000~0X0802
0X002C	CAL	Month/Day	Calibration date: high byte is month, low						0X01~0X0C, 0X01~0X1F

		byte is day						
0X002D	Factory	Manufacturer: A total of 10 bytes of ASCII code						0X20~0X7E, 0X20~0X7E
0X0031								0X20~0X7E, 0X20~0X7E
0X0032	Model	Product model: A total of 12 bytes ASCII code						0X20~0X7E, 0X20~0X7E
0X0037								0X20~0X7E, 0X20~0X7E
0X0038	SN	Product serial number: a total of 14 bytes of ASCII code						0X20~0X7E, 0X20~0X7E
0X003E								0X20~0X7E, 0X20~0X7E
0X003F	Hardware Version	Hardware version:A total of 2 bytes ASCII code						0X20~0X7E, 0X20~0X7E
0X0040	Software Version	Software version: A total of 2 bytes ASCII code						0X20~0X7E, 0X20~0X7E
0X005B	USER ADDR 1	User storage point 1 0:recall 1:save						0X0000~0X0001
0X0064	USER ADDR 10	User storage point 10 0:recall 1:save						0X0000~0X0001
0X0065	CH1 V ADJ 1	CH1 Voltage calibration point 1						0X0000~0XFFFF
0X006E	CH1 V ADJ 10	CH1Voltage calibration point 10						0X0000~0XFFFF
0X00C9	CH2V ADJ 1	CH2Voltage calibration point 1						0X0000~0XFFFF
0X00D2	CH2V ADJ 10	CH2Voltage calibration point 10						0X0000~0XFFFF
0X012D	CH3V ADJ 1	CH3Voltage calibration point 1						0X0000~0XFFFF
0X0136	CH3V ADJ 10	CH3Voltage calibration point 10						0X0000~0XFFFF
0X0191	CH1 A ADJ 1	CH1current calibration point 1						0X0000~0XFFFF
0X019A	CH1 A ADJ 10	CH1current calibration point 10						0X0000~0XFFFF
0X01F5	CH2A ADJ 1	CH2current calibration point 1						0X0000~0XFFFF
0X01FE	CH1 A ADJ 10	CH2current calibration point 10						0X0000~0XFFFF
0X0259	CH3A ADJ 1	CH3current calibration point 1						0X0000~0XFFFF
0X0263	CH3A ADJ 10	CH3current calibration point 10						0X0000~0XFFFF

Note: All operations take effect in remote operation mode (register 0X0000 value is 0X0001) !

*Based on local operating mode after power cycle!

The function code table and interval time are as follows:

function code	Corresponding function	time between two operations
---------------	------------------------	-----------------------------

0X03	Read one or more registers consecutively	N * 5ms
0X06	Write a single register command	(register 0x0000~0x0021)10ms, (register 0x0065~0x02BC)300ms
0X0A	Write one or more register commands in succession	N * 5ms
0X0B	A single configuration corresponds to a calibration point and returns the calibration point value	10ms
0X0C	Save one or more calibration points in a row	N * 10ms

Note: N is the number of registers

The communication protocol format is as follows:

function code 0X03

PC send:8 Byte

address	Function code	start address high byte	start address low byte	end address high byte	end address low byte	CRC16 Check code high byte	CRC16 Check code low byte
0XFE	0X03						

power return: 8+N*2 Byte

address	Function code	start address high byte	start address low byte	end address high byte	end address low byte	return data high byte	return data low byte	return data N+1 high byte	return data N+1 low byte	CRC16Check code high byte	CRC16Check code low byte
0XFE	0X03										

*address range: (0X0000~0X0040)

Example 1: Read a single register status(0X0000)

PC send: FE 03 00 00 00 00 C5 51 (start address is 0X0000, end address is 0X0000, CRC16The verification result is 0X0551)

Power return: FE 03 00 00 00 00 01 03 3C (0X0001Indicates remote operation mode, 0X033C is CRC16 calibration result)

Example 2: Read Multiple Register Status(0X0001~0X0003)

0XFE	0X0A										
------	------	--	--	--	--	--	--	--	--	--	--

电源返回: 8+N*2 Byte

地址	功能码	起始地址 高字节	起始地址 低字节	结束地址 高字节	结束地址 低字节	写入数据 高字节	写入数据 低字节	写入数据 N+1 高字节	写入数据 N+1 低字节	CRC16 校验码 高字节	CRC16 校验码 低字节
0XFE	0X0A										

*地址范围: (0X0000~0X0021) (0X002B~0X0040)

举例一: 写单个寄存器状态 (0X0001)

PC 发送: FE 0A 00 01 00 0103 E8 7D 08 (开始地址为 0X0001, 结束地址为 0X0001, 写入数据为 0X03E8, CRC16 校验结果为 0X7D08)

电源返回: FE 0A 00 01 00 01 03 E8 7D 08 (开始地址为 0X0001, 结束地址为 0X0001, 写入数据为 0X03E8, CRC16 校验结果为 0X7D08)

举例二: 写多个寄存器状态 (0X0001~0X0003)

PC 发送: FE 0A 00 01 00 03 03 E8 07 D0 0B B8 EB 98 (开始地址为 0X0001, 结束地址为 0X0003, CRC16 校验结果为 0XEB98)

电源返回: FE 0A 00 01 00 03 03 E8 07 D0 0B B8 EB 98 (0X03E8 为寄存器 0X0001 值, 0X07D0 为寄存器 0X0002 值, 0X0BB8 为寄存器 0X0003 值)

功能码 0X0B

PC 发送: 8 Byte

地址	功能码	配置地址 高字节	配置地址 低字节	写入密码 高字节	写入密码 低字节	CRC16 校验码 高字节	CRC16 校验码 低字节
0XFE	0X0B			0X88	0X66		

电源返回: 8Byte

地址	功能码	配置地址 高字节	配置地址 低字节	返回数据 高字节	返回数据 低字节	CRC16 校验码 高字节	CRC16 校验码 低字节
0XFE	0X0B						

*地址范围: 0X0065~0X02BC 返回的数据为校准点参数值

举例一: 配置 CH1 电压第一个校准点 (0X0065)

PC 发送: FE 0B 00 65886631 46 (配置校准点地址为 0X0065, 写入密码为 0X8866, CRC16 校验结果为 0X3146)

电源返回: FE 0B 00 65 00 0A1C 20 (配置校准点地址为 0X0065, 校准电压为 0X000A, CRC16 校验结果为 0X1C20)

功能码 0X0C

PC 发送: 10 Byte

地址	功能码	起始地址 高字节	起始地址 低字节	结束地址 高字节	结束地址 低字节	写入密码 高字节	写入密码 低字节	CRC16 校验码 高字节	CRC16 校验码 低字节
0XFE	0X0C					0X88	0X66		

PC 发送: 10 Byte

地址	功能码	起始地址 高字节	起始地址 低字节	结束地址 高字节	结束地址 低字节	返回保存状态 高字节	返回保存状态 低字节	CRC16 校验码 高字节	CRC16 校验码 低字节
0XFE	0X0C								

*地址范围:0X0065~0X02BC 返回状态为 0X0001: 保存成功 返回状态为 0X0000: 保存失败

举例一: 保存 CH1 电压第一个校准点(0X0065)

PC 发送: FE 0B 00 65 00 6588 66 3E CE (保存校准点开始地址为 0X0065, 结束地址为 0X0065, 写入密码为 0X8866, CRC16 校验结果为 0X76BC)

电源返回: FE 0B 00 65 00 6500 01D4 E9(保存校准点开始地址为 0X0065, 结束地址为 0X0065, 0X0001 标示保存成功, CRC16 校验结果为 0XD4E9)

举例一: 保存 CH1 电压全部个校准点(0X0065~0X00C8)

PC 发送: FE 0B 00 65 00 C8 88 66 DF 5F (保存校准点开始地址为 0X0065, 结束地址为 0X00C8, 写入密码为 0X8866, CRC16 校验结果为 0XDF5F)

电源返回: FE 0B 00 65 00 C800 0135 78(保存校准点开始地址为 0X0065, 结束地址为 0X00C8, 0X0001 标示保存成功, CRC16 校验结果为 0X3578)

CRC16 计算代码:

```
const unsigned char CRCHTable[] =
```

```
{
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, 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, 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, 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
}
```

```
const unsigned char CRCLTable[] =
{
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  
};
```

```
unsigned int crc16(unsigned char *DData, unsigned char len)  
{  
    unsigned char CRCHi = 0xFF;  
    unsigned char CRCLo = 0xFF;  
    unsigned int wIndex;  
    unsigned int CRC_DData;  
    while(len--)  
    {  
        wIndex = CRCLo ^ *DData++;  
    }  
}
```

```
    CRCLo = CRChi ^ CRCHTaIbe[wIndex];  
    CRChi = CRCLTaIbe[wIndex];  
}  
CRC_DData = CRChi;  
CRC_DData<<= 8;  
CRC_DData |=CRCLo;  
    return CRC_DData;  
}
```