

易事特集团股份有限公司	文件编号		文件版本	V1.8
	文件密级	非密	生效日期	2020.09
	制定部门	软件部		

# EA990 G4 40-120K

## Modbus 通信协议

易事特集团股份有限公司	文件编号		文件版本	V1.8
	文件密级	非密	生效日期	2020.09
	制定部门	软件部		

序号	版本	修改内容	修改人	修改时间	备注
1	Ver 1.0	确定基本的电气量	Gaop	2018-11-14	
2	Ver 1.1	修改 02、04、03 功能码表	Gaop	2018-11-20	
3	Ver 1.2	增加遥信量，修改遥信量名称	Gaop	2019-01-23	
4	Ver 1.3	增加遥信量 249、432	Gaop	2019-03-28	
5	Ver 1.4	修改告警量通信异常、配置错误，修改告警名称	Gaop	2019-05-23	
6	Ver 1.5	添加遥信量	Gaop	2019-06-26	
7	Ver 1.6	添加遥信量	Gaop	2020-05-14	
8	Ver 1.7	添加遥信量	Lizs	2020-09-01	
9	Ver 1.8	补充适用范围和接口定义	Tanqp	2020-09-21	

# 目 录

目 录.....	
<b>一、协议相关说明.....</b>	<b>1</b>
1、协议适用范围.....	1
2、规范性引用文件.....	1
3、协议简介.....	1
4、接口方式.....	1
5、Modbus RTU 数据帧格式.....	3
6、响应信息分类.....	4
7、功能代码.....	5
<b>二、寄存器列表.....</b>	<b>6</b>
1、读输入寄存器（功能码 0x04）.....	6
2、读离散量（功能码 0x02）.....	9
3、读保持寄存器（功能码 0x03）.....	25
<b>三、通信内容.....</b>	<b>26</b>
1、读输入寄存器（功能码 0x04）.....	26
2、读离散量（功能码 0x02）.....	27
3、读取设备寄存器（功能码 0x03）.....	28
4、预置寄存器（功能码 0x06,0x10）.....	29
<b>附录A CRC 校验.....</b>	<b>30</b>
<b>附录B 高低位字节表.....</b>	<b>32</b>

## 一、协议相关说明

### 1、协议适用范围

本协议文档规范了设备 EA990 G4 40-120K UPS 通过 RS485 连接提供的 Modbus 接口需求。

### 2、规范性引用文件

1. RFC791, 互联网协议, Sep81 DARPA

2. MODBUS 协议参考指南 Rev J, MODICON, 1996 年 6 月, doc#PI\_MBUS\_300

MODBUS 是一项应用层报文传输协议, 用于在通过不同类型的总线或网络连接的设备之间的客户机/服务器通信。目前, 使用下列情况实现 MODBUS:

1) 以太网上的 TCP/IP。

2) 各种媒体 (有线: EIA/TIA-232-E、EIA/TIA-485-A; 光纤、无线等等) 上的异步串行传输。

### 3、协议简介

Modbus 协议是应用于控制器上的一种通用语言。通过该协议使控制器经由网络和其他 UPS 设备之间可以进行通信。本通信采用应答方式, 由主机发起请求 (发送遥测、遥信信息), 从机执行请求并且应答。从机需通过地址设置加以区分, 从机可设置的地址范围为 1~247。

### 4、接口方式

RS485 接口: 异步, 半双工

波特率: 默认 9600bps, 可设置为 1200bps、2400 bps、4800 bps、9600bps

数据长度: RTU 模式时为 8 位

奇偶校验位: 无奇偶校验

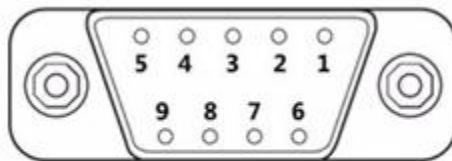
停止位: 1 位

接口说明: 从左往右依次为 GND、A 端、B 端



RS232 接口:

PC 接线 与 UPS 接线规则(采用 9Pin 头母座)



DB9母头 ( 孔 )

针脚编号	名称	功能
2	RXD (Received Data)	串口数据输入
3	TXD (Trasmitted Data)	串口数据输出
5	GND (Signal Ground)	信号地线

## 5、Modbus RTU 数据帧格式

本协议仅支持Modbus 通信RTU 模式。

控制器以RTU 模式在Modbus 总线上进行通讯时，信息中的每个字节按十六进制。RTU 模式中每个字节的格式为：

编码系统： 8 位二进制；

起始位： 1 位；

数据位： 8 位；

奇/偶校验： 奇校验或者偶校验时为1 位；无奇偶校验时该位为1 位停止位；

停止位： 1 位；

错误校验区：循环冗余校验(CRC)；

RTU 模式的请求帧格式为：

起始	设备地址	功能代码	寄存器起始地址	寄存器个数	CRC 低字节	CRC 高字节	结束
至少3.5 个字符空闲时间	1 byte	1 byte	2 bytes	2 bytes	1 byte	1 byte	至少3.5 个字符空闲时

其中 RTU 模式字符传输格式采用 11 位传输，其中数据位为 8 位，若无奇偶校验位，则 10 位传输

起始位	1	2	3	4	5	6	7	8	停止位（奇/偶校验位）	停止位

RTU 模式的响应帧格式为：

起始	设备地址	功能代码	数据	CRC 低字节	CRC 高字节	结束
至少3.5 个字符空闲时间	1 byte	1 byte	N bytes	1 byte	1 byte	至少3.5 个字符空闲时间

消息发送至少需要 3.5 个字符时间的停顿间隔开始。在最后一个传输字符之后，需要至少 3.5 字符时间的停顿来标定消息的结束。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转输。如果在帧完成之前两个字符间有超过 1.5 个字符空闲的停顿时间，认为帧错误，停止接收，并重新启动接收。也就是要保证两个帧间的间隔至少大于 3.5 个字符的时间，1.5 个字符时间和 3.5 个字符时间与具体的通信波特率有关，计算方法如下：如通信波特率为 9600，那么

$$1.5 \text{ 个字符间隔时间} = (1/9600) \times 11 \times 1.5 \times 1000 = 1.72 \text{ ms}$$

$$3.5 \text{ 个字符间隔时间} = (1/9600) \times 11 \times 3.5 \times 1000 = 4.01 \text{ ms}$$

【例如】请求1号机的数据，位置为：寄存器起始地址0002，寄存器个数为1个，寄存器内容为0x1222

请求帧信息：

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x01	0x03	0x00	0x02	0x00	0x01	0x25	0xCA
字节数	1	1	2		2		2	

响应帧信息：

	地址	功能码	返回数据字节数	数据内容		CRC 校验	
数据	0x01	0x03	0x02	0x12	0x22	0x34	0xFD
字节数	1	1	1	2		2	

## 6、响应信息分类

主机向从机设备发送查询并希望有一个正常响应，主机查询中有可能产生4种事件：

(1) 从机接收查询，无通讯错误，正常处理信息，则返回一个正常响应事件。

(2) 由于通讯出错，从机不能接收查询数据，因而不返回响应。此时，主机依靠处理程序判定为查询超时。

(3) 若从机接收查询，发现有CRC通讯错误，不返回响应，此时依靠主机处理程序判定为查询超时。

(4) 从机接收查询，无通讯错误，但无法处理(如读不存在的寄存器地址或错误的寄存器个数)时，向主机报告错误的性质。

向主机报告错误的响应信息有2个与正常响应不相同的区域：

**功能代码区：**正常响应时，从机的响应功能代码区，带原查询的功能代码。所有功能代码的MSB为0(其值低于80H)。不正常响应时，从机把功能代码的MSB置为1，使功能代码值大于80H，高于正常响应的值。这样，主机应用程序能识别不正常响应事件，能检查不正常代码的数据区。

**数据区：**正常响应中，数据区含有(按查询要求给出的)数据或统计值，在不正常响应中，数据区为一个不正常代码，它说明从机产生不正常响应的条件和原因。

不正常代码及含义如下表所示：

代码	名称	含义
0x01	不合法功能代码	从机接收的是一种不能执行功能代码。发出查询命令后，该代码指示无程序功能
0x02	不合法数据地址	接收的数据地址，是从机不允许的地址。
0x03	不合法数据	查询数据区的值是从机不允许的值。
0x04	从机设备故障	从机执行主机请求的动作时出现不可恢复的错误。
0x08	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。

【例如】\*\*\*

RTU 模式：（ASCII 模式类似）

命令信息：请求1号机的数据，位置为：寄存器起始地址0066，寄存器个数为2个

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x01	0x03	0x00	0x66	0x00	0x02	0x24	0x14

响应信息：1号机的响应帧，因为寄存器起始地址错误，因此返回信息为不合法的数据地址

	地址	功能码	数据内容	CRC 校验	
数据	0x01	0x83	0x02	0xC0	0xF1

## 7、功能代码

功能码	名称	作用
0x02	读离散量输入	读从机离散量输入寄存器中的二进制数据
0x03	读取保持寄存器	在一个或多个保持寄存器取得当前的二进制值
0x04	读取输入寄存器	在一个或多个输入寄存器取得当前的二进制值
0x06	写单个保持寄存器	写从机上的单个寄存器（可作为设置单个参数的功能码）
0x10	写多个保持寄存器	写从机上的多个寄存器（可作为设置多个参数的功能码）



## 二、寄存器列表

## 1、读输入寄存器（功能码 0x04）

地址		寄存器内容	长度 /格式	数据类型	单位	系数	备注
DEC	HEX						
0	0x0000	旁路 A 相电压	2Bytes	SHORT	V	0.1	
1	0x0001	旁路 B 相电压	2Bytes	SHORT	V	0.1	
2	0x0002	旁路 C 相电压	2Bytes	SHORT	V	0.1	
3	0x0003	旁路 A 相电流	2Bytes	SHORT	A	0.1	
4	0x0004	旁路 B 相电流	2Bytes	SHORT	A	0.1	
5	0x0005	旁路 C 相电流	2Bytes	SHORT	A	0.1	
6	0x0006	旁路 A 相频率	2Bytes	SHORT	Hz	0.1	
7	0x0007	旁路 B 相频率	2Bytes	SHORT	Hz	0.1	
8	0x0008	旁路 C 相频率	2Bytes	SHORT	Hz	0.1	
9	0x0009	预留	2Bytes	SHORT	/	1	
10	0x000A	预留	2Bytes	SHORT	/	1	
11	0x000B	预留	2Bytes	SHORT	/	1	
12	0x000C	输入 A 相电压	2Bytes	SHORT	V	0.1	
13	0x000D	输入 B 相电压	2Bytes	SHORT	V	0.1	
14	0x000E	输入 C 相电压	2Bytes	SHORT	V	0.1	
15	0x000F	输入 A 相电流	2Bytes	SHORT	A	0.1	
16	0x0010	输入 B 相电流	2Bytes	SHORT	A	0.1	
17	0x0011	输入 C 相电流	2Bytes	SHORT	A	0.1	
18	0x0012	输入 A 相频率	2Bytes	SHORT	Hz	0.1	
19	0x0013	输入 B 相频率	2Bytes	SHORT	Hz	0.1	
20	0x0014	输入 C 相频率	2Bytes	SHORT	Hz	0.1	
21	0x0015	预留	2Bytes	SHORT	/	1	
22	0x0016	预留	2Bytes	SHORT	/	1	
23	0x0017	预留	2Bytes	SHORT	/	1	
24	0x0018	输出 A 相电压	2Bytes	SHORT	V	0.1	
25	0x0019	输出 B 相电压	2Bytes	SHORT	V	0.1	
26	0x001A	输出 C 相电压	2Bytes	SHORT	V	0.1	
27	0x001B	输出 A 相电流	2Bytes	SHORT	A	0.1	

28	0x001C	输出 B 相电流	2Bytes	SHORT	A	0.1	
29	0x001D	输出 C 相电流	2Bytes	SHORT	A	0.1	
30	0x001E	输出 A 相频率	2Bytes	SHORT	Hz	0.1	
31	0x001F	输出 B 相频率	2Bytes	SHORT	Hz	0.1	
32	0x0020	输出 C 相频率	2Bytes	SHORT	Hz	0.1	
33	0x0021	预留	2Bytes	SHORT	/	1	
34	0x0022	预留	2Bytes	SHORT	/	1	
35	0x0023	预留	2Bytes	SHORT	/	1	
36	0x0024	输出 A 相视在功率	2Bytes	SHORT	kVA	0.1	
37	0x0025	输出 B 相视在功率	2Bytes	SHORT	kVA	0.1	
38	0x0026	输出 C 相视在功率	2Bytes	SHORT	kVA	0.1	
39	0x0027	输出 A 相有功功率	2Bytes	SHORT	kW	0.1	
40	0x0028	输出 B 相有功功率	2Bytes	SHORT	kW	0.1	
41	0x0029	输出 C 相有功功率	2Bytes	SHORT	kW	0.1	
42	0x002A	预留	2Bytes	SHORT	/	1	
43	0x002B	预留	2Bytes	SHORT	/	1	
44	0x002C	预留	2Bytes	SHORT	/	1	
45	0x002D	输出 A 相负载百分比	2Bytes	SHORT	%	1	
46	0x002E	输出 B 相负载百分比	2Bytes	SHORT	%	1	
47	0x002F	输出 C 相负载百分比	2Bytes	SHORT	%	1	
48	0x0030	预留	2Bytes	SHORT	/	1	
49	0x0031	电池电压	2Bytes	SHORT	V	0.1	
50	0x0032	预留	2Bytes	SHORT	/	1	
51	0x0033	电池电流	2Bytes	SHORT	A	0.1	
52	0x0034	预留	2Bytes	SHORT	/	1	

53	0x0035	电池温度	2Bytes	SHORT	℃	0.1	
54	0x0036	电池节数	2Bytes	SHORT	/	1	
55	0x0037	电池容量	2Bytes	SHORT	AH	1	
56	0x0038	电池剩余放电时间	2Bytes	SHORT	min	1	
57	0x0039	电池剩余容量	2Bytes	SHORT	%	1	
58	0x003A	预留	2Bytes	SHORT	/	1	
59	0x003B	预留	2Bytes	SHORT	/	1	
60	0x003C	预留	2Bytes	SHORT	/	1	
61	0x003D	额定容量	2Bytes	SHORT	kVA	1	
62	0x003E	预留	2Bytes	SHORT	/	1	
63	0x003F	额定输入电压	2Bytes	SHORT	V	1	
64	0x0040	额定输入频率	2Bytes	SHORT	Hz	1	
65	0x0041	额定输出电压	2Bytes	SHORT	V	1	
66	0x0042	额定输出频率	2Bytes	SHORT	Hz	1	
67	0x0043	能流线状态寄存器	2Bytes	SHORT	/	1	编号从低到高，每段占用 2bit， 00 表示静止， 01 表示正向流动(向右向下)， 02 表示反向流动(向左向上)
68	0x0044	预留	2Bytes	SHORT	/	1	
69	0x0045	预留	2Bytes	SHORT	/	1	
70	0x0046	机型识别寄存器	2Bytes	SHORT	/	1	0: 单单，1: 三单， 2: 三三
71	0x0047	供电方式	2Bytes	SHORT	/	1	0: 不供电； 1: 市电供电； 2: 电池供电； 3: 联合供电； 4: 旁路供电；

## 2、读离散量（功能码0x02）

地址		寄存器内容	长度/格式	备注
DEC	HEX			
208	0x00D0	系统板通信异常	1bit	208~335 为监控告警
209	0x00D1	整流通信异常	1bit	/
210	0x00D2	逆变通信异常	1bit	/
211	0x00D3	预留 211	1bit	/
212	0x00D4	预留 212	1bit	/
213	0x00D5	预留 213	1bit	/
214	0x00D6	预留 214	1bit	/
215	0x00D7	预留 215	1bit	/
216	0x00D8	预留 216	1bit	/
217	0x00D9	系统板配置错误	1bit	/
218	0x00DA	整流配置错误	1bit	/
219	0x00DB	逆变配置错误	1bit	/
220	0x00DC	预留 220	1bit	/
221	0x00DD	预留 221	1bit	/
222	0x00DE	预留 222	1bit	/
223	0x00DF	预留 223	1bit	/
224	0x00E0	预留 224	1bit	/
225	0x00E1	预留 225	1bit	/
226	0x00E2	紧急告警	1bit	/
227	0x00E3	次要告警	1bit	/
228	0x00E4	旁路供电	1bit	/
229	0x00E5	电池供电	1bit	/
230	0x00E6	电池低压(DOD)	1bit	/
231	0x00E7	电池低压(EOD)	1bit	/
232	0x00E8	旁路故障	1bit	/
233	0x00E9	风扇故障	1bit	/
234	0x00EA	电池接地故障	1bit	/
235	0x00EB	油机模式	1bit	/
236	0x00EC	电池开关断开	1bit	/

237	0x00ED	紧急停机	1bit	/
238	0x00EE	配电柜输出开关断开	1bit	/
239	0x00EF	配电柜维修开关闭合	1bit	/
240	0x00F0	配电柜旁路开关断开	1bit	/
241	0x00F1	交流防雷器断开	1bit	/
242	0x00F2	外部变压器过温	1bit	/
243	0x00F3	市电输入空开断开	1bit	/
244	0x00F4	机器输出开关断开	1bit	/
245	0x00F5	机器维修开关闭合	1bit	/
246	0x00F6	机器旁路开关断开	1bit	/
247	0x00F7	维修盖板打开	1bit	/
248	0x00F8	旁路反灌保护接触器断开	1bit	/
249	0x00F9	分时下电	1bit	/
250	0x00FA	从机通信异常	1bit	
251	0x00FB	预留 251	1bit	
252	0x00FC	预留 252	1bit	
253	0x00FD	预留 253	1bit	
254	0x00FE	BMS 通信异常	1bit	
255	0x00FF	电池柜数目不一致	1bit	
256	0x0100	电池柜内模组数目不一致	1bit	
257	0x0101	模组内电芯数目不一致	1bit	
258	0x0102	预留	1bit	
259	0x0103	预留	1bit	
260	0x0104	预留	1bit	
261	0x0105	预留	1bit	
262	0x0106	预留	1bit	
263	0x0107	预留	1bit	
264	0x0108	预留	1bit	
265	0x0109	预留	1bit	
266	0x010A	预留	1bit	
267	0x010B	预留	1bit	
268	0x010C	预留	1bit	
269	0x010D	预留	1bit	
270	0x010E	预留	1bit	

271	0x010F	预留	1bit	
272	0x0110	预留	1bit	
273	0x0111	预留	1bit	
274	0x0112	预留	1bit	
275	0x0113	预留	1bit	
276	0x0114	预留	1bit	
277	0x0115	预留	1bit	
278	0x0116	预留	1bit	
279	0x0117	预留	1bit	
280	0x0118	预留	1bit	
281	0x0119	预留	1bit	
282	0x011A	预留	1bit	
283	0x011B	预留	1bit	
284	0x011C	预留	1bit	
285	0x011D	预留	1bit	
286	0x011E	预留	1bit	
287	0x011F	预留	1bit	
288	0x0120	预留	1bit	
289	0x0121	预留	1bit	
290	0x0122	预留	1bit	
291	0x0123	预留	1bit	
292	0x0124	预留	1bit	
293	0x0125	预留	1bit	
294	0x0126	预留	1bit	
295	0x0127	预留	1bit	
296	0x0128	预留	1bit	
297	0x0129	预留	1bit	
298	0x012A	预留	1bit	
299	0x012B	预留	1bit	
300	0x012C	预留	1bit	
301	0x012D	预留	1bit	
302	0x012E	预留	1bit	
303	0x012F	预留	1bit	
304	0x0130	预留	1bit	

305	0x0131	预留	1bit	
306	0x0132	预留	1bit	
307	0x0133	预留	1bit	
308	0x0134	预留	1bit	
309	0x0135	预留	1bit	
310	0x0136	预留	1bit	
311	0x0137	预留	1bit	
312	0x0138	预留	1bit	
313	0x0139	预留	1bit	
314	0x013A	预留	1bit	
315	0x013B	预留	1bit	
316	0x013C	预留	1bit	
317	0x013D	预留	1bit	
318	0x013E	预留	1bit	
319	0x013F	预留	1bit	
320	0x0140	预留	1bit	
321	0x0141	预留	1bit	
322	0x0142	预留	1bit	
323	0x0143	预留	1bit	
324	0x0144	预留	1bit	
325	0x0145	预留	1bit	
326	0x0146	预留	1bit	
327	0x0147	预留	1bit	
328	0x0148	预留	1bit	
329	0x0149	预留	1bit	
330	0x014A	预留	1bit	
331	0x014B	预留	1bit	
332	0x014C	预留	1bit	
333	0x014D	预留	1bit	
334	0x014E	预留	1bit	
335	0x014F	预留	1bit	
336	0x0150	系统板与逆变模块 CAN 通讯异常	1bit	336~463 为旁路模块/系统板告警
337	0x0151	多个逆变器的地址相同	1bit	

338	0x0152	供电状态冲突	1bit	
339	0x0153	逆变模块不均流	1bit	
340	0x0154	输出过载 105%	1bit	
341	0x0155	输出过载 110%	1bit	
342	0x0156	输出过载 125%	1bit	
343	0x0157	输出过载 150%	1bit	
344	0x0158	旁路与逆变切换失败	1bit	
345	0x0159	锁相失败	1bit	
346	0x015A	系统自检失败	1bit	
347	0x015B	输出电压快速异常转旁路	1bit	
348	0x015C	负载冲击转旁路	1bit	
349	0x015D	输出过载告警	1bit	
350	0x015E	反复切换锁定旁路	1bit	
351	0x015F	反复切换锁定逆变	1bit	
352	0x0160	系统板之间的 CAN 通讯异常	1bit	
353	0x0161	系统过载 105%	1bit	
354	0x0162	系统过载 110%	1bit	
355	0x0163	系统过载 125%	1bit	
356	0x0164	系统过载 150%	1bit	
357	0x0165	并机均流异常	1bit	
358	0x0166	邻机请求转旁路	1bit	
359	0x0167	系统过载告警	1bit	
360	0x0168	输出相序接反	1bit	
361	0x0169	机架编号异常	1bit	
362	0x016A	并机线连接异常	1bit	
363	0x016B	模块未锁(SYS)	1bit	
364	0x016C	ECU 未就绪	1bit	
365	0x016D	功率模块数不满足负载量	1bit	
366	0x016E	频率超跟踪范围	1bit	
367	0x016F	输出采样异常	1bit	
368	0x0170	旁路相电压过压	1bit	
369	0x0171	旁路相电压欠压	1bit	
370	0x0172	旁路频率过高	1bit	
371	0x0173	旁路频率过低	1bit	



372	0x0174	旁路相序接反	1bit	
373	0x0175	旁路缺相	1bit	
374	0x0176	旁路相电压不平衡	1bit	
375	0x0177	旁路电压快检测异常	1bit	
376	0x0178	旁路过流	1bit	
377	0x0179	ECO 旁路电压过压	1bit	
378	0x017A	ECO 旁路电压欠压	1bit	
379	0x017B	ECO 旁路频率过频	1bit	
380	0x017C	ECO 旁路频率欠频	1bit	
381	0x017D	ECO 旁路快速低压	1bit	
382	0x017E	ECO 旁路相序反	1bit	
383	0x017F	ECO 旁路中线丢失	1bit	
384	0x0180	旁路 E2PROM 操作失败	1bit	
385	0x0181	旁路 DSP 与监控通讯失败	1bit	
386	0x0182	旁路 DSP 软件版本异常	1bit	
387	0x0183	旁路软件版本和硬件版本不匹配	1bit	
388	0x0184	旁路风扇故障	1bit	
389	0x0185	旁路 SCR 开路/保险故障	1bit	
390	0x0186	旁路 SCR 短路故障	1bit	
391	0x0187	紧急停机	1bit	
392	0x0188	停机按钮	1bit	
393	0x0189	旁路辅助电源故障	1bit	
394	0x018A	旁路反灌	1bit	
395	0x018B	旁路熔断器故障	1bit	
396	0x018C	旁路散热器过温	1bit	
397	0x018D	预留	1bit	
398	0x018E	预留	1bit	
399	0x018F	预留	1bit	
400	0x0190	输出电压快检测异常	1bit	
401	0x0191	输出电压过压	1bit	
402	0x0192	输出电压欠压	1bit	
403	0x0193	输出频率过高	1bit	
404	0x0194	输出频率过低	1bit	
405	0x0195	输出电压不平衡	1bit	

406	0x0196	逆变过载 105%	1bit	
407	0x0197	逆变过载 110%	1bit	
408	0x0198	逆变过载 125%	1bit	
409	0x0199	逆变过载 150%	1bit	
410	0x019A	旁路过载 125%	1bit	
411	0x019B	旁路过载 135%	1bit	
412	0x019C	旁路过载 150%	1bit	
413	0x019D	旁路过载 200%	1bit	
414	0x019E	逆变过载告警	1bit	
415	0x019F	旁路过载告警	1bit	
416	0x01A0	备电时间预告警	1bit	
417	0x01A1	剩余容量预告警	1bit	
418	0x01A2	电池维护提醒	1bit	
419	0x01A3	电池放电时间终止	1bit	
420	0x01A4	电池放电电压终止	1bit	
421	0x01A5	电池过温	1bit	
422	0x01A6	电池低温	1bit	
423	0x01A7	电池自检失败	1bit	
424	0x01A8	电池放电超时	1bit	
425	0x01A9	预留	1bit	
426	0x01AA	预留	1bit	
427	0x01AB	预留	1bit	
428	0x01AC	预留	1bit	
429	0x01AD	预留	1bit	
430	0x01AE	预留	1bit	
431	0x01AF	预留	1bit	
432	0x01B0	维修旁路空开闭合	1bit	
433	0x01B1	输出空开断开	1bit	
434	0x01B2	拨码处于测试状态	1bit	
435	0x01B3	保持在逆变输出	1bit	
436	0x01B4	保持在旁路输出	1bit	
437	0x01B5	手动旁路开启	1bit	
438	0x01B6	机架内两系统板间 SPI 通信异常	1bit	
439	0x01B7	并机线 1 异常	1bit	

440	0x01B8	并机线 2 异常	1bit	
441	0x01B9	机架间心跳异常	1bit	
442	0x01BA	并机线连接故障	1bit	
443	0x01BB	系统板和旁路 can 通信异常	1bit	
444	0x01BC	额定机架数量和当前机架数量不符	1bit	
445	0x01BD	额定模块数量和实际模块数量不符	1bit	
446	0x01BE	双系统板备份 IO 异常	1bit	
447	0x01BF	机架间旁路输入不一致	1bit	
448	0x01C0	EOD 关机 自启动但市电异常禁止开机	1bit	
449	0x01C1	冗余模块不足	1bit	
450	0x01C2	输出短路	1bit	
451	0x01C3	预留	1bit	
452	0x01C4	预留	1bit	
453	0x01C5	预留	1bit	
454	0x01C6	预留	1bit	
455	0x01C7	预留	1bit	
456	0x01C8	预留	1bit	
457	0x01C9	预留	1bit	
458	0x01CA	预留	1bit	
459	0x01CB	预留	1bit	
460	0x01CC	预留	1bit	
461	0x01CD	预留	1bit	
462	0x01CE	预留	1bit	
463	0x01CF	预留	1bit	
464	0x01D0	输入电压过压	1bit	464~591 为整流模块告警
465	0x01D1	输入电压欠压	1bit	
466	0x01D2	输入频率过频	1bit	
467	0x01D3	输入频率欠频	1bit	
468	0x01D4	输入相序接反	1bit	
469	0x01D5	输入缺相	1bit	
470	0x01D6	输入电压不平衡	1bit	

471	0x01D7	输入电压快检测异常	1bit	
472	0x01D8	输入电流过流	1bit	
473	0x01D9	输入电流不平衡	1bit	
474	0x01DA	输入零线未接	1bit	
475	0x01DB	输入保险故障	1bit	
476	0x01DC	输入限功率	1bit	
477	0x01DD	市电电池切换频繁	1bit	
478	0x01DE	输入过载	1bit	
479	0x01DF	预留	1bit	
480	0x01E0	电池未接	1bit	
481	0x01E1	电池过温	1bit	
482	0x01E2	电池自检失败	1bit	
483	0x01E3	电池过压	1bit	
484	0x01E4	电池 DOD 欠压	1bit	
485	0x01E5	电池 EOD 欠压	1bit	
486	0x01E6	电池过充	1bit	
487	0x01E7	电池低温	1bit	
488	0x01E8	电池硬件过压故障	1bit	
489	0x01E9	电池充电过流	1bit	
490	0x01EA	电池放电过流	1bit	
491	0x01EB	电池保险/充电器开关开路	1bit	
492	0x01EC	充电器开关短路	1bit	
493	0x01ED	电池放电超时	1bit	
494	0x01EE	电池反接	1bit	
495	0x01EF	电池缺 N 线故障	1bit	
496	0x01F0	正母线电压过压	1bit	
497	0x01F1	负母线电压过压	1bit	
498	0x01F2	正母线电压欠压	1bit	
499	0x01F3	负母线电压欠压	1bit	
500	0x01F4	正负母线电压不平衡	1bit	
501	0x01F5	母线硬件过压故障	1bit	
502	0x01F6	母线过压次数到	1bit	
503	0x01F7	母线电容寿命低于 1 年	1bit	
504	0x01F8	正母线瞬时欠压	1bit	

505	0x01F9	负母线瞬时欠压	1bit	
506	0x01FA	母线短路	1bit	
507	0x01FB	母线电压过压	1bit	
508	0x01FC	母线电压欠压	1bit	
509	0x01FD	母线升压异常	1bit	
510	0x01FE	母线电容故障预告警	1bit	
511	0x01FF	预留	1bit	
512	0x0200	输入 AC 软启失败	1bit	
513	0x0201	电池 DC 软启失败	1bit	
514	0x0202	BUS DC/DC 软启失败	1bit	
515	0x0203	输入锁相失败	1bit	
516	0x0204	市电电池切换频繁锁电池	1bit	
517	0x0205	整流软启次数到	1bit	
518	0x0206	整流器硬件逐波限流故障	1bit	
519	0x0207	整流器硬件过流故障	1bit	
520	0x0208	整流器硬件逐波限流告警	1bit	
521	0x0209	输入 PFC 软启动失败	1bit	
522	0x020A	整流器过流	1bit	
523	0x020B	整流器市电模式过流	1bit	
524	0x020C	整流器电池模式过流	1bit	
525	0x020D	锁相故障	1bit	
526	0x020E	母线软启次数到	1bit	
527	0x020F	母线硬件软启失败	1bit	
528	0x0210	整流 IGBT 模块过温	1bit	
529	0x0211	整流 E2PROM 读写失败	1bit	
530	0x0212	整流 DSP 与监控通讯失败	1bit	
531	0x0213	整流 DSP 与 CPLD 通讯失败	1bit	
532	0x0214	风机故障	1bit	
533	0x0215	整流 PowerOK 异常	1bit	
534	0x0216	风扇故障预告警	1bit	
535	0x0217	整流 CPLD 软件版本异常	1bit	
536	0x0218	整流 DSP 软件版本异常	1bit	
537	0x0219	整流软件版本和硬件版本不匹配	1bit	
538	0x021A	整流器辅助电源异常	1bit	

539	0x021B	整流逆变间 SPI 通信故障	1bit	
540	0x021C	驱动连接故障	1bit	
541	0x021D	整流接触器故障	1bit	
542	0x021E	电池接触器故障	1bit	
543	0x021F	紧急停机	1bit	
544	0x0220	充电器开关短路	1bit	
545	0x0221	充电器开关开路	1bit	
546	0x0222	充电器软启失败	1bit	
547	0x0223	充电器过压	1bit	
548	0x0224	充电器硬件过压故障	1bit	
549	0x0225	充电器欠压	1bit	
550	0x0226	充电器过流	1bit	
551	0x0227	充电器过温	1bit	
552	0x0228	充电器硬件逐波限流故障	1bit	
553	0x0229	充电器硬件逐波限流告警	1bit	
554	0x022A	充电器硬件过流故障	1bit	
555	0x022B	预留	1bit	
556	0x022C	预留	1bit	
557	0x022D	预留	1bit	
558	0x022E	预留	1bit	
559	0x022F	预留	1bit	
560	0x0230	平衡桥臂硬件过流故障	1bit	
561	0x0231	平衡桥臂硬件逐波限流故障	1bit	
562	0x0232	平衡桥臂硬件逐波限流告警	1bit	
563	0x0233	平衡桥臂快速过流	1bit	
564	0x0234	平衡桥臂过流	1bit	
565	0x0235	平衡桥臂过温	1bit	
566	0x0236	模块未锁(PFC)	1bit	
567	0x0237	拨码处于测试状态	1bit	
568	0x0238	锂电充电一级保护	1bit	
569	0x0239	锂电放电一级保护	1bit	
570	0x023A	锂电充电二级保护	1bit	
571	0x023B	锂电放电二级保护	1bit	
572	0x023C	锂电充电三级保护	1bit	

573	0x023D	锂电放电三级保护	1bit	
574	0x023E	锂电充电告警	1bit	
575	0x023F	锂电放电告警	1bit	
576	0x0240	预留	1bit	
577	0x0241	预留	1bit	
578	0x0242	预留	1bit	
579	0x0243	预留	1bit	
580	0x0244	预留	1bit	
581	0x0245	预留	1bit	
582	0x0246	预留	1bit	
583	0x0247	预留	1bit	
584	0x0248	预留	1bit	
585	0x0249	预留	1bit	
586	0x024A	预留	1bit	
587	0x024B	预留	1bit	
588	0x024C	预留	1bit	
589	0x024D	预留	1bit	
590	0x024E	预留	1bit	
591	0x024F	预留	1bit	
592	0x0250	母线短路	1bit	592~719 为逆变模块告警
593	0x0251	母线异常	1bit	/
594	0x0252	母线过压	1bit	/
595	0x0253	母线欠压	1bit	/
596	0x0254	母线电压不平衡	1bit	/
597	0x0255	预留	1bit	/
598	0x0256	预留	1bit	/
599	0x0257	预留	1bit	/
600	0x0258	预留	1bit	/
601	0x0259	预留	1bit	/
602	0x025A	预留	1bit	/
603	0x025B	预留	1bit	/
604	0x025C	预留	1bit	/
605	0x025D	预留	1bit	/
606	0x025E	预留	1bit	/

607	0x025F	预留	1bit	/
608	0x0260	逆变电压过压	1bit	/
609	0x0261	逆变电压欠压	1bit	/
610	0x0262	逆变电压不平衡	1bit	/
611	0x0263	直流分量过大(R S T)	1bit	/
612	0x0264	逆变模块过载 105%	1bit	/
613	0x0265	逆变模块过载 110%	1bit	/
614	0x0266	逆变模块过载 125%	1bit	/
615	0x0267	逆变模块过载 150%	1bit	/
616	0x0268	逆变输出短路	1bit	/
617	0x0269	逆变模块过载告警	1bit	/
618	0x026A	逆变过频	1bit	/
619	0x026B	逆变欠频	1bit	/
620	0x026C	预留	1bit	/
621	0x026D	预留	1bit	/
622	0x026E	预留	1bit	/
623	0x026F	预留	1bit	/
624	0x0270	旁路 SCR 开路故障	1bit	/
625	0x0271	旁路 SCR 短路故障	1bit	/
626	0x0272	旁路过载 125%	1bit	/
627	0x0273	旁路过载 135%	1bit	/
628	0x0274	旁路过载 150%	1bit	/
629	0x0275	旁路过载 200%	1bit	/
630	0x0276	旁路过载告警	1bit	/
631	0x0277	旁路过温	1bit	/
632	0x0278	旁路过压	1bit	/
633	0x0279	旁路欠压	1bit	/
634	0x027A	旁路过频	1bit	/
635	0x027B	旁路欠频	1bit	/
636	0x027C	旁路相序接反	1bit	/
637	0x027D	预留	1bit	/
638	0x027E	预留	1bit	/
639	0x027F	预留	1bit	/
640	0x0280	逆变软启失败	1bit	/



641	0x0281	锁相失败	1bit	
642	0x0282	旁路与逆变切换频繁	1bit	
643	0x0283	逆变软启次数到	1bit	
644	0x0284	并机均流异常	1bit	
645	0x0285	捕获故障	1bit	
646	0x0286	负载冲击	1bit	
647	0x0287	邻机请求转旁路	1bit	
648	0x0288	并机线异常	1bit	
649	0x0289	驱动连接故障	1bit	
650	0x028A	同步方波异常	1bit	
651	0x028B	逆变自检失败	1bit	
652	0x028C	预留	1bit	
653	0x028D	预留	1bit	
654	0x028E	预留	1bit	
655	0x028F	预留	1bit	
656	0x0290	逆变器散热器过温	1bit	
657	0x0291	逆变 E2PROM 操作失败	1bit	
658	0x0292	逆变 DSP 与监控通讯失败	1bit	
659	0x0293	逆变 DSP 与系统板通讯失败	1bit	
660	0x0294	逆变 DSP 与 CPLD 通讯失败	1bit	
661	0x0295	逆变硬件过流故障	1bit	
662	0x0296	逆变保险故障	1bit	
663	0x0297	紧急停机	1bit	
664	0x0298	逆变接触器故障	1bit	
665	0x0299	逆变硬件逐波限流故障	1bit	
666	0x029A	逆变硬件逐波限流告警	1bit	
667	0x029B	逆变电源异常	1bit	
668	0x029C	逆变 CPLD 软件版本异常	1bit	
669	0x029D	逆变 DSP 软件版本异常	1bit	
670	0x029E	逆变软件版本和硬件版本不匹配	1bit	
671	0x029F	零地故障	1bit	
672	0x02A0	逆变继电器开路故障	1bit	
673	0x02A1	逆变继电器短路故障	1bit	
674	0x02A2	逆变电容故障	1bit	

675	0x02A3	逆变风扇故障	1bit	
676	0x02A4	整流逆变间 SPI 通信故障	1bit	
677	0x02A5	逆变快速过流	1bit	
678	0x02A6	逆变 PowerOK 异常	1bit	
679	0x02A7	模块未锁(INV)	1bit	
680	0x02A8	逆变风扇故障预告警	1bit	
681	0x02A9	逆变电容故障预告警	1bit	
682	0x02AA	拨码处于测试状态	1bit	
683	0x02AB	保持在逆变输出	1bit	
684	0x02AC	保持在旁路输出	1bit	
685	0x02AD	预留	1bit	
686	0x02AE	预留	1bit	
687	0x02AF	预留	1bit	
688	0x02B0	输出过压	1bit	
689	0x02B1	输出欠压	1bit	
690	0x02B2	预留	1bit	
691	0x02B3	预留	1bit	
692	0x02B4	预留	1bit	
693	0x02B5	预留	1bit	
694	0x02B6	预留	1bit	
695	0x02B7	预留	1bit	
696	0x02B8	预留	1bit	
697	0x02B9	预留	1bit	
698	0x02BA	预留	1bit	
699	0x02BB	预留	1bit	
700	0x02BC	预留	1bit	
701	0x02BD	预留	1bit	
702	0x02BE	预留	1bit	
703	0x02BF	预留	1bit	
704	0x02C0	预留	1bit	
705	0x02C1	预留	1bit	
706	0x02C2	预留	1bit	
707	0x02C3	预留	1bit	
708	0x02C4	预留	1bit	

709	0x02C5	预留	1bit	
710	0x02C6	预留	1bit	
711	0x02C7	预留	1bit	
712	0x02C8	预留	1bit	
713	0x02C9	预留	1bit	
714	0x02CA	预留	1bit	
715	0x02CB	预留	1bit	
716	0x02CC	预留	1bit	
717	0x02CD	预留	1bit	
718	0x02CE	预留	1bit	
719	0x02CF	预留	1bit	

## 3、读保持寄存器（功能码 0x03）

地址		寄存器内容	数据长度 /格式	数据 类型	单位	系数	说明
DEC	HEX						
32768	0x8000	预留	2 bytes				
32769	0x8001	预留	2 bytes				
32770	0x8002	蜂鸣器消音	2 bytes	SHORT	/	1	支持读取/设置。 1: 蜂鸣器消音; 0: 蜂鸣器正常工作。
32771	0x8003	预留	2 bytes				
32772	0x8004	预留	2 bytes				
32773	0x8005	预留	2 bytes				
32774	0x8006	预留	2 bytes				
32775	0x8007	预留	2 bytes				
32776	0x8008	开关机	2 bytes	SHORT	/	1	支持设置。 1: 关机/关机断输出; 2: 关机转旁路; 3: 开机;
32777	0x8009	系统 Unix 时钟 _低十六位	2 bytes	SHORT	/	1	支持读取/设置。 32 位 Unix 时钟，两个寄存器必须同时设置。
32778	0x800A	系统 Unix 时钟 _高十六位	2 bytes	SHORT	/	1	只写，1: 按设置的自检时间开始自检。
32779	0x800B	电池测试 (按时间)	2Bytes	SHORT	/	1	只写，1: 按设置的自检结束电压开始自检。
32780	0x800C	电池测试 (按电压)	2Bytes	SHORT	/	1	0~23.0
32781	0x800D	电池自检时间	2Bytes	SHORT	/	1	1.60~1.90
32782	0x800E	电池自检结束电压	2Bytes	SHORT	/	1	

### 三、通信内容

#### 1、读输入寄存器（功能码0x04）

##### 【举例】

假设UPS 设备地址设置为0x18，查询寄存器起始地址值为0x0010，寄存器个数为2 个，即查询“R 相输出电流”和“S 相输出电流”的值；假设此时“R 相输出电流”的值为89.2A，“S 相输出电流”的值为88.9A，根据该值的系数为0.1，那么：

寄存器0x0010 的值为： $(892)_D = (037C)_H$

寄存器0x0011 的值为： $(889)_D = (0379)_H$

则返回数据的字节数为4 个，RTU 模式时，对数据查询的请求帧信息和响应帧信息为：

请求帧信息为：

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x18	0x04	0x00	0x10	0x00	0x02	0x72	0x07
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	返回数据字节数	数据内容		CRC 校验	
数据	0x18	0x04	0x04	0x037C	0x0379	0x73	0xCB
字节数	1	1	1	4		2	

## 2、读离散量（功能码0x02）

### 【举例】

假设UPS 设备地址设置为0x18，查询寄存器起始地址值为51，即0x0033，寄存器个数为1个，即查询“UPS 过载状态”；假设此时“UPS 已过载”；即该值为1。返回数据时，在该字节中由低位向高位排列，直至8个位为止。下一个字节中的8个输入位也是从低位到高位排列。若返回的输入位数不是8的倍数，则在最后的数据字节中的剩余位直至字节的最高位全部填零。

RTU 模式时，对状态查询的请求帧信息和响应帧信息为：

请求帧信息为：

	地址	功能码	起始地址		离散量个数		CRC 校验	
数据	0x18	0x02	0x00	0x33	0x00	0x01	0x4B	0xCC
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	返回数据字节数	数据内容	CRC 校验	
数据	0x18	0x02	0x01	0x01	0x67	0x14
字节数	1	1	1	1	2	

### 3、读取设备寄存器（功能码 0x03）

假设 UPS 设备地址设置为 0x18，请求读取寄存器 43-44 的内容：

寄存器 0x0043 的值为：(541)D = (021D)H

寄存器 0x0044 的值为：(309)D = (0135)H

则返回数据的字节数为 4 个，RTU 模式时，对数据查询的请求帧信息和响应帧信息为：

请求帧信息为：

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x18	0x03	0x00	0x43	0x00	0x02	0x37	0xD6
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	返回数据字节数	数据内容		CRC 校验	
数据	0x18	0x03	0x04	0x021D	0x0135	0x22	0xCB
字节数	1	1	1	4		2	

#### 4、预置寄存器（功能码 0x06,0x10）

##### 【举例】

假设UPS 设备地址设置为0x18，预置寄存器起始地址值为1，寄存器个数为1 个，即电池测试10S。寄存器内容被预置后返回正常响应；预置单个寄存器的请求帧信息和响应帧信息为：请求帧信息为：

	地址	功能码	寄存器起始地址		预置数据		CRC 校验	
数据	0x18	0x06	0x00	0x01	0xFF	0xFF	0xDB	0xB3
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	寄存器地址		预置成功的数据		CRC 校验	
数据	0x18	0x06	0x00	0x01	0xFF	0xFF	0xDB	0xB3
字节数	1	1	2		2		2	



## CRC 循环冗余校验

循环冗余校验 (CRC) 域为两个字节，包含一个二进制 16 位值。附加在报文后面的 CRC 的值由发送设备计算。接收设备在接收报文时重新计算 CRC 的值，并将计算结果于实际接收到的 CRC 值相比较。如果两个值不相等，则为错误。

CRC 的计算, 开始对一个 16 位寄存器预装全 1. 然后将报文中的连续的 8 位子节对其进行后续的计算。只有字符中的 8 个数据位参与生成 CRC 的运算，起始位，停止位和校验位不参与 CRC 计算。CRC 的生成过程中，每个 8-位字符与寄存器中的值异或。然后结果向最低有效位 (LSB) 方向移动(Shift) 1 位，而最高有效位 (MSB) 位置充零。然后提取并检查 LSB：如果 LSB 为 1，则寄存器中的值与一个固定的预置值异或；如果 LSB 为 0，则不进行异或操作。这个过程将重复直到执行完 8 次移位。完成最后一次（第 8 次）移位及相关操作后，下一个 8 位字节与寄存器的当前值异或，然后又同上面描述过的一样重复 8 次。当所有报文中子节都运算之后得到的寄存器中的最终值，就是 CRC。

### 生成 CRC 的过程为:

1. 将一个 16 位寄存器装入十六进制 FFFF (全 1). 将之称作 CRC 寄存器.
2. 将报文的第一个 8 位字节与 16 位 CRC 寄存器的低字节异或，结果置于 CRC 寄存器.
3. 将 CRC 寄存器右移 1 位 (向 LSB 方向)，MSB 充零. 提取并检测 LSB.
4. (如果 LSB 为 0): 重复步骤 3 (另一次移位).
- (如果 LSB 为 1): 对 CRC 寄存器异或多项式值 0xA001 (1010 0000 0000 0001).
5. 重复步骤 3 和 4，直到完成 8 次移位。当做完此操作后，将完成对 8 位字节的完整操作。
6. 对报文中的下一个字节重复步骤 2 到 5，继续此操作直至所有报文被处理完毕。
7. CRC 寄存器中的最终内容为 CRC 值.
8. 当放置 CRC 值于报文时，如下面描述的那样，高低字节必须交换。

将 CRC 放置于报文当 16 位 CRC (2 个 8 位字节) 在报文中传送时，低位字节首先发送，然后是高位字节。

**例:**执行 CRC 生成的 C 语言的函数在下面示出。所有的可能的 CRC 值都被预装在两个数组中，当计算报文内容时可以简单的索引即可。一个数组含有 16 位 CRC 域的所有 256 个可能的高位字节，另一个数组含有低位字节的值。这种索引访问 CRC 的方式提供了比对报文缓冲区的每个新字符都计算新的 CRC 更快的方法。

注意: 此函数内部执行高/低 CRC 字节的交换。此函数返回的是已经经过交换的 CRC 值。

也就是说，从该函数返回的 CRC 值可以直接放置于报文用于发送。

函数使用两个参数：

unsigned char \*puchMsg; 指向含有用于生成 CRC 的二进制数据报文缓冲区的指针

unsigned short usDataLen; 报文缓冲区的字节数。

CRC 生成函数

unsigned short CRC16 ( puchMsg, usDataLen ) /\* 函数以 unsigned short 类型返回 CRC \*/

unsigned char \*puchMsg ; /\* 用于计算 CRC 的报文 \*/

unsigned short usDataLen ; /\* 报文中的字节数 \*/

```
{
    unsigned char uchCRCHi = 0xFF ; /* CRC 的高字节初始化 */
    unsigned char uchCRCLo = 0xFF ; /* CRC 的低字节初始化 */
    unsigned uIndex ;          /* CRC 查询表索引 */
    while (usDataLen-- )      /* 完成整个报文缓冲区 */
    {
        uIndex = uchCRCLo ^ *puchMsgg++ ; /* 计算 CRC */
        uchCRCLo = uchCRCHi ^ auchCRCHi[uIndex] ;
        uchCRCHi = auchCRCLo[uIndex] ;
    }
    return (uchCRCHi << 8 | uchCRCLo) ;
}
```

## 附录B 高低位字节表

高字节表

/\* 高位字节的 CRC 值 \*/

```
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
};

```

### 低字节表

/\* 低位字节的 CRC 值 \*/

```

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  
};