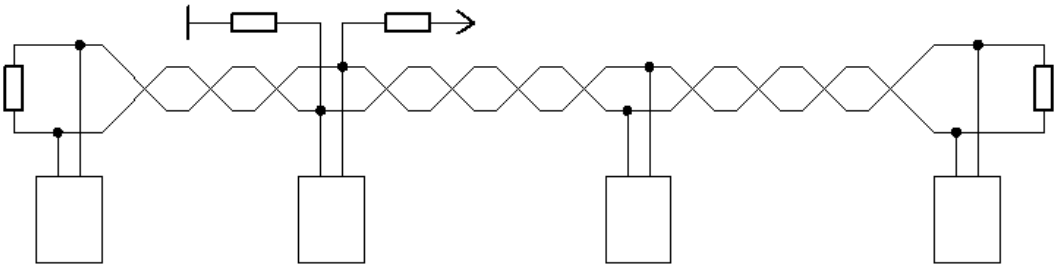
BKR MODBUS 通信协议



文档版本

日期	版本	注释
06-11-08	01	文件草案发布
07-03-20	02	新软件的功能(V2.1.x)
07-09-21	03	新的MODBUS硬件接口 固件版本V2.2.x及以上有效
08-11-18	04	增加附加的总线地址

目录

重要信息！

如果在手册的文字信息旁边出现了左边的标志，建议读者仔细阅读相应的信息，因为它可能对该设备的使用非常重要。

它包含了设备正确操作的安全建议或其它信息。

如果忽视这些信息，设备可能会无法正常使用，甚至损坏！

MODBUS 协议的附加文档可以从 www.modbus.org 找到

MODBUS 标准也可以在那里找到。

1 概述

BKR MODBUS 扩展提供了从设备中读取数值和修改设备设置的功能。

注意：每次请求 BKR 最多能发送 30 个数值。

本文件主要介绍通过 MODBUS 通信协议的传输。协议规定了进行数据传输和访问控制的方法，但并没有把用户限制在一个单一的物理传输系统。BKR 的 RS485 用在物理层上。总线电缆接口可以把一个或更多的 BKR 连接到一对电缆上，利用 ID 访问各个单元。

许多商用设备和 PLC，无论是总线的主机还是从机都可以用 MODBUS 协议。因为可以从不同的供应商获得多种 SCADA 解决方案，所以在现有的总线系统或在新建的系统中集成 BKR 不成问题。

2. MODBUS / RS485

符合以下两个基本的方面：

- RS485 传输使用串行数据传输。在一个总线配置中 RS485 可以互联多个设备，RS485 协议可以为高等级的 MODBUS 协议提供“服务”。
- MODBUS 协议使用优先串行数据传输层（RS485 既是如此）与几个总线设备通信。定义了访问从机的命令，地址结构和数据结构。

2.1 物理层 - RS485（在 EIA485/ISO8482 中定义）

RS485 支持基本的串行数据传输到更高等级的 MODBUS 协议层，正因如此它叫做总线系统的“物理层”。高层的协议使用低物理层当作数据传输的基本“服务”。

RS485 使用两条数据线进行串行传输，每条线通过传输设备提供的 0V 或者 5V 驱动。这两条数据线一般有不同的电压等级。一种情况(一条线 5V，另一条接地)表示逻辑“关”的状态。两条线交换他们的电压表示逻辑“开”的状态。这种不同的传输模式使 RS485 非常抗电磁干扰，所以它允许的传输距离达到 1000 米以上。

BKR 的数据传输比率可以选择 1200、2400、9600、19200 或者 38400 波特。奇偶校验可以选择奇(odd)、偶(even)和无校验(no parity)。所有总线设备需要使用相同的设置，标准设置为：9600 波特和偶校验(even parity)。

RS485 存在两种不同类型：

- 2 线 RS485：这种类型只使用从一个通道引出的两条数据线。这就意味着发送完请求后，总线的主机必须使它的发送端无效，保证数据线路的空闲，来应答设备。（半双工模式）
- 4 线 RS485：这种类型使用一条数据线（2 线）用在主机到从机方向，另一条（2 线）用在从机到主机方向。**BKR 不支持 4 线 RS485。**

虽然未明确提及，2 线和 4 线类型都需要另一条线连接：接地线。所以，2 线版本需要一条 3 线线缆，4 线版本需一条 5 线线缆。必须使用屏蔽线，但不要在接地连接线上使用屏蔽。它只需用在保护接地上来防止电磁干扰。

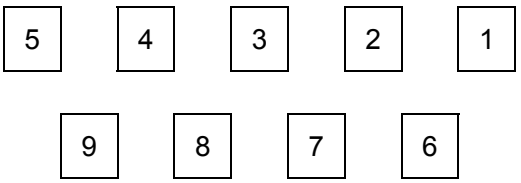
RS485 总线可以连接多台装置（高达 32 台）。为实现这个，总线所有设备的多个数据信号需要互联。两条数据线和共同接地点 GND。总线上的所有设备并联。避免使用分接头，这样可以避免来自在长距离传输时的传输误差，应该尽量将设备直接连接到主总线线路上。

一条包含所有设备的总线电缆称为“总线段”。几条总线段可以通过“中继器”互联。

2.1.1 连接

MODBUS 有两种不同的接口：

a) 通过 9 针 sub-d 连接



PIN1 +5V（这个电压输出仅对数据的线路偏压，**不提供给任何其他的外部电路**）

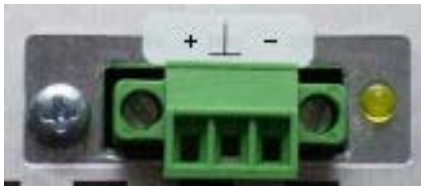
PIN2 偏压接地和所有总线设备的共同接地端

PIN5 D(B) - 数据信号 B

PIN9 D(A) - 数据信号 A

b) 通过 3 针接口连接

这种连接方式是通过 3 针接口，见右侧图片。使用这种 MODBUS，需要连接数据线 + 和 - 还有共同接地端（中间口）。



2.1.2 线路终端

一个非常重要的地方是总线线路的终端。工作总线系统必须把线路末端的回声屏蔽，防止数据信号的失真。在总线电缆的末端，要在两条线缆之间加一个电阻，其电阻值等于电缆阻抗。通常选用 120Ω，连接在数据线 D(+)和 D(-)各自总线末端之间。

有些设备，尤其是总线接头内置电阻。检查总线上所有设备的手册。如果它们内部电阻没有被禁用，会对总线有非常大的影响：必须把这些设备放置在总线的一个末端！如果总线只有两个末端，那就只能在此总线上使用两个带固定电阻的设备！

2.1.3 线路偏压

另一个比较重要的地方是线路偏压。如果没有设备参与传输，数据线将会出现左偏压。因为终端电阻需要相同的电压。这可能会因为外部影响产生虚假的数据信号。对于这样的线路偏压，一般是给数据线定义“关”状态。

两个阻值大概 500-600Ω 的电阻分别被连接在 D(+)与+5V 之间和 D(-)和 GND 之间。每

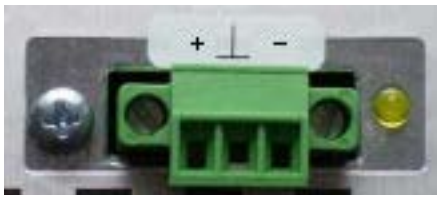
条总线的两个偏压电阻只需要一次，偏压电阻的位置并不重要。可以放在总线的任何地方，甚至在中间位置。请检查所有总线设备的手册，内部是否集成了电阻！

当设备采用 a)型（9 针）接口，总线接头的电压 5V 和 GND 有效，这两个电阻可以焊接在连接头外壳的内部。

不过，这些不适合 b)型（3 针）接口。**注意：不同的产品，都描述为 A=+和 B=-是不正确的，具体要视情况而定。**

2.1.4 通信指示灯

设备背面的黄灯用来指示主动传输，在设备主动与总线主机通信时闪烁。



通信指示灯对两种接口类型都有效。

2.2 MODBUS 协议

2.2.1 MODBUS 简介

MODBUS 协议将 RS485 作为一个基础物理层和为数据传输提供控制机制来使用。位于 OSI 层级模式的 2 层（连接层）用于数据传输系统。

2.2.2 串口数据格式和组帧

数据以固定帧传输，帧的分隔需要总线至少停止 3.5 个字来实现。所有的数据在协议数据单元（protocol data units, PDUs）中被编排，可以基于基础物理协议层在串行总线系统中传输。

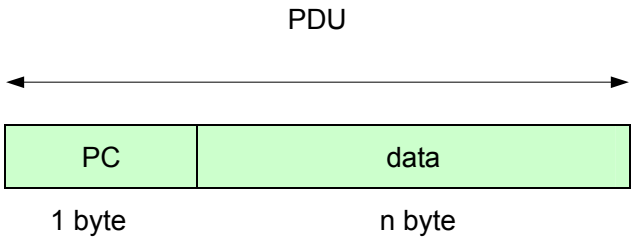


插图 1：PDU

PDU 由两部分组成：

- 功能码(FC)给出了一个命令，定义了从机单元需要做什么。
- 数据块由 FC 对应的数据组成。它的用途取决于 FC，可以包含纯数据，也可以是从机数据访问的寄存器地址。

PDU 定义了一个单独的数据单元，为了实现一个功能而与某个总线设备通信。传输不一致，取决于使用的物理层。

为了能控制传输，PDU 扩展了附加数据块。对于 RS485，扩展的结果表现在应用数据单元(application data unit, ADU)。

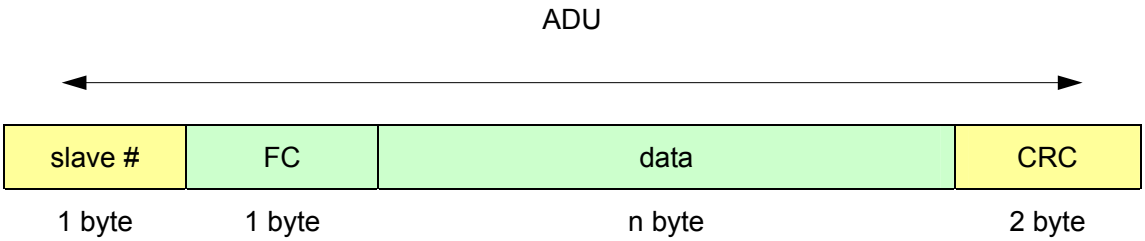


插图 2：ADU

应用数据单元，实际上通过 RS485 进行串行通信，包括两个附加数据块：

- 首先指定数据块的目标，“从机编码”（=从机地址）
- 通过 CRC16 误差校正码，传输更有保证。

2.2.3 串行通信模式

协议为帧数据目录定义了两种不同的编码，**BKR 使用 RTU 模式！ASCII 模式不被支持，在此提及只为了阐述的完整性。**

远程终端装置(remote terminal unit, RTU)

这种传输模式下，所有 8 位数据字包括两个 4 位十六进制数，它们作为一个完整的字节进行传输；接近最大的传说密度。通过所有的数据字，可以传输以下信息：

- 1 起始位
- 8 数据位，“最低位”在前
- 1 校验位（如果有此设置）
- 1 停止位为奇偶校验 / 2 校验没有补偿缺失的校验位。

美国信息交换标准码 (American Standard Code for Information Interchange, ASCII)

ASCII 模式下，8 位数据字中的两个 4 字节与在 ASCII 码表述中是分离传输的。一个包含 5B(HEX)的数据字节将被分成两部分，每部分按照一个字节进行独立传输。结果会通过 35H(=ASCII 码 5)和 42H(=ASCII 码 B)的两个数据字节传输。此数据码考虑到了兼容性的因素，而且便于传输线路上进行调试，但明显的降低了传输速率。

2.2.4 功能码

前面已经提到，数据包中包含规定了从总线主机到总线从机的命令的功能码。如果从机执行命令，将会在回复确认命令时使用相同的功能码回答。功能码的有效范围从 1 到 127，只有其中的一部分实际使用。更详细的信息请查阅 MODBUS 技术规范。如果命令是从机不能执行的，会回复异常（=错误码）。异常包的功能码是标准命令的功能码，当故障发生后在某一方面被改变：从机设置最高位用来向主机发送错误情况。数据包目录有更详细的说明。

BKR 支持 03H(读取保持寄存器)、04H(读取输入寄存器)和 06H(写入单独寄存器)。

2.2.5 异常代码

如果主机发生的命令从机不能执行，从机以异常码回复。在 MODBUS 规则里有全部的代码。主机软件会自动处理大多异常情况，所以在此就不在列出。如果需要对 MODBUS 主机堆栈进行编程，这种情况下才需要错误码的全部列表。

2.2.6 主-从机协议

主-从机协议用来通信。仅用于让总线主机开始数据传输。主机发送命令，传输数据格式及相应的功能码(=命令)给从机，然后开始执行数据交换。

- 单点传送模式在 MODBUS 系统的通信中广泛使用，在主机数据包中一个单独从机的地址用从机编号表示。有效地址从 1 到 247。从机执行命令并回答确认数据包返回主机。
- 不是在任何情况下主机都会收到它请求的回答：在多点模式下，所有从机在总线上的地址是并行的。它们执行同样的命令，但都不会响应。主机使用 0 作为从机编号开始多点传送。

2.2.7 BKR MODBUS 设置

如果设备支持 MODBUS，在设备的"setup"菜单有一个附加的有效入口。进入菜单，可以选择以下项目：

- Address(地址)：设备的从机地址(slave ID)。有效范围：1-247。
- Baud rate(波特率)：在此选择波特率，有效范围：1200-38400 波特。
- Parity(奇偶性)：选择奇偶校验为奇、偶或无(数据位/奇偶校验/停止位)
- 对于所有总线设备，波特率和奇偶性设置要一致；每一个设备的地址必需是唯一的。

2.2.8 地址空间

BKR 的数据是有序编排的，凭地址访问。每一个地址访问一个数据字。一个数据字通

常是 16 位。

BKR 的地址和功能码没有差异。这是一个巨大的有效的地址空间，可以访问每个地址的数据，可以使用任何有效的功能码。不过，数据只有通过正确的方式解析才有意义。

数据有以下几种方式：

- real：这是 32 位浮点数，符合 IEEE 标准 754。
- uint16：这是无符号 16 整数值
- uint32，sint32：这是无符号/有符合 32 位整数值。

数据按照 16 位宽字编排，长数据目录需要读取一组连续的地址。这种情况下，表中给出了基准地址。通过基准地址 12 读取 real，一个数据需要读取地址 12 和 13 的两个 16 位字。这两个数值需要连接起来才能得到预期的 32 位。大多 SCADA 软件包或者 PLC 可以完成此工作。

不同类型的地址：

MODBUS 地址通常以 0 开始，最大至 65535。可以被所有功能码使用。PLC 不能正确处理 0，因此在地址加 1，所以它们的地址(MODBUS 地址+1)总是从 1 开始。

一些 SCADA 工具通过增加偏移来决定功能码，用来在给出的地址中访问设备。有时也在 MODBUS 地址加 1。例如，地址 40001 为“使用功能码 03H 读取 MODBUS 地址 0”，30012 为“使用功能码 04H 读取地址 11”。请参照软件手册找寻正确的地址。

下表给出了上面提及到的 MODBUS 地址。

2.2.9 测量值

在 2 数据字的间隔中，测量值从地址 0 开始有效。

如果计算有效谐波的电流或电压太小，在基准地址（基点）读取为 0.0%。这表明，电流或者电压的高次谐波也会无效。

所有数值可以通过功能码 03H 和 04H 访问。**涉及到对称配电系统的视在功率 S-sum、有功功率 P-sum、无功功率 Q-sum、缺少的无功功率△Q 和功率因数(P/S)数值。**

地址	数值	字	类型	单位
0	功率	2	REAL	Hz
2	线电压	2	REAL	V
4	相电压	2	REAL	V
6	L1 相电流	2	REAL	A
18	视在功率 S-sum	2	REAL	VA
26	有功功率 P-sum	2	REAL	W
34	无功功率 Q-sum	2	REAL	var
42	缺少的无功功率△Q	2	REAL	var
50	功率因数(P/S)	2	REAL	-
58	THDU	2	REAL	%
60	基波电压	2	REAL	%
62	2 次谐波电压	2	REAL	%
64	3 次谐波电压	2	REAL	%
…				
122	32 次谐波电流	2	REAL	%
124	THDI	2	REAL	%
130	基波电流	2	REAL	%
132	2 次谐波电流	2	REAL	%
134	3 次谐波电流	2	REAL	%
…				
192	32 次谐波电流	2	REAL	%
322	环境温度	2	REAL	°C

2.2.10 工作计数器

工作计数器/累加器安排在一个专门的通道。对于防止它们的精度下降很有必要。每个计数器由两部分组成：

1. Float 型基础计数器是简单的累积/整合。如果此计数器到达 1000000.0，扩展计数器将加 1，然后基础计数器继续无进位计数。

2. Long 型扩展计数器，MW / Mvar 计数部分高达 $(2^{32}-1)\cdot 10^{6}$。

为了获取真实工作值，需要扩展计数器的值乘以 1000000 再增加基础计数器的值。这样保证了 Float 型基础计数器的精度在可以接受的范围，所以对于大的计数值来说不会有工作丢失。

所有此类值可以通过功能码 03H 和 04H 访问。

地址	数值	字	类型	单位
1792	WQ 感性–扩展计数器	2	UINT32	MVarh
1794	WQ 感性–基础计数器	2	REAL	Varh
1796	WQ 容性–扩展计数器	2	UINT32	MVarh
1798	WQ 容性–基础计数器	2	REAL	Varh
1800	WP 输入–扩展计数器	2	UINT32	MWh
1802	WP 输入–基础计数器	2	REAL	Wh
1804	WP 输出–扩展计数器	2	UINT32	MWh
1806	WP 输出–基础计数器	2	REAL	Wh

2.2.11 参数设置

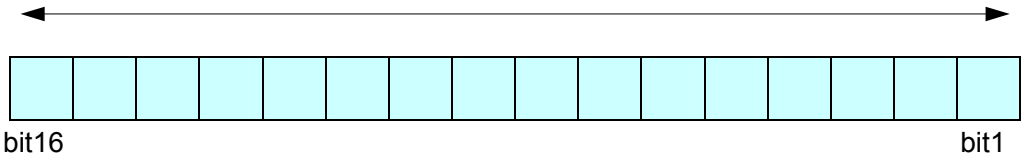
用户进行参数设置，以不同的数据类型进行存储。基础地址和数据类型见下表。

所有这些参数可以通过功能代码 03H、04H 和 06H 访问。

地址	数值	字	类型	单位
512	PT 变比 x10	1	UINT16	-
513	CT 变比 x10	1	UINT16	-
515	相位校正	1	UINT16	°
517	额定电压 L-L	2	UINT32	V
519	最大允许电压	1	UINT16	%
520	最小允许电压	1	UINT16	%
522	开关延迟时间 x10	1	UINT16	s
523	目标 cos ϕ 1 (0..100..200 = i0.00..1.00..c0.00)	1	UINT16	-
524	目标 cos ϕ 2 (0..100..200 = i0.00..1.00..c0.00)	1	UINT16	-
525	控制灵敏度	1	UINT16	%
526	投切次数平衡允许误差	1	UINT16	%
533	温度报警门限 1 x 10	1	UINT16	°C
534	温度报警门限 2 x 10	1	UINT16	°C
535	滞后温度 x 10	1	UINT16	°C
536	THDU 报警门限 x 10	1	UINT16	%
537	THDI 报警门限 x 10	1	UINT16	%
539	步长警告门限	1	UINT16	%
540	投切次数警告门限	2	UINT32	-
542	有功功率 P 报警门限	2	UINT32	W
544	无功功率 Q 报警门限	2	UINT32	var
546	最低功率因数报警 (0...100...200 = i0.00...1.00...c0.00)	1	UINT16	-
547	最高功率因数报警 (0...100...200 = i0.00...1.00...c0.00)	1	UINT16	-
548	功率因数报警延迟时间	1	UINT16	s
555	步交换投切时间延迟 x10	1	UINT16	s
556	快速控制延迟时间	1	UIN16	per
557	最大步长(快速控制)	1	UINT16	var
559	平均无功(快速控制)	1	UINT16	var

在地址 514 所有用户参数被收集，没有下面的数值。在这一点所有用户参数采用二进制编码。每一位代表菜单"Measurement"各自的"Control"中的调整。UINT16 数值代码如下所示。

用户参数

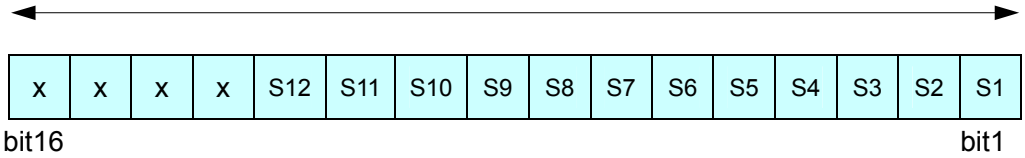


• Bit 1	频率同步	00 = 自动	
• Bit 2		01 = 固定 50Hz	11 = 固定 60Hz
• Bit 3	连接类型	1 = L-L	0 = L-N
• Bit 4	步识别	1 = Off	0 = On
• Bit 5	投切次数平衡	1 = Yes	0 = No
• Bit 6	测试模式	1 = Yes	0 = No
• Bit 7	保留		
• Bit 8	步切换	1 = Yes	0 = No
• Bit 9	控制	00 = 自动	
• Bit 10		01 = LIFO	10 = Combined filter
• Bit 11	I < Limit; 步冻结	1 = Yes	0 = No
• Bit 12	快速控制；同步脉冲	1 = Yes	0 = No
• Bit 13 - 16	保留		

2.2.12 步状态

所有步的信息储存在数据库中。在不同的数据类型中，上面提到的信息都有效。下面提到的存储器的位分配适用于输出：

输出



S1 - S12：1 到 12 步的输出

基准地址和数据类型如下表所示。

所有这些数据可以通过功能码 03H 和 04H 访问。

地址	数值	字	类型	单位
768	步快速 (1 = fast)	1	UINT16	-
769	步固定 (1 = fix)	1	UINT16	-
770	步固定投入/切除 (1 = on)	1	UINT16	-
772	步故障 (1 = defective)	1	UINT16	-
1280	工作状态 (1 = on)	1	UINT16	-

其它步信息的所有更远的基准地址和数据类型如下表所述，步长的数值建立在额定电压的基础上。

地址	数值	字	类型	单位
773	步长 L-第 1 步值	2	SINT32	var
779	步长 L-第 2 步值	2	SINT32	var
773+(6*(n-1))	步长 L-第 n 步值	2	SINT32	var
...	...			
839	步长 L-第 12 步值	2	SINT32	var
845	步长 F-第 1 步值	2	SINT32	var
851	步长 F-第 2 步值	2	SINT32	var
845+((6*(n-1))	步长 F-第 n 步值	2	SINT32	var
...	...			
911	步长 F-第 12 步值	2	SINT32	var
917	第 1 步投切时间	2	SINT32	-
919	第 2 步投切时间	2	SINT32	-
...	...			
939	第 12 步投切时间	2	SINT32	-
941	第 1 步放电时间×10	1	UINT16	s
942	第 2 步放电时间×10	1	UINT16	s
...	...			
952	第 12 步放电时间×10	1	UINT16	s

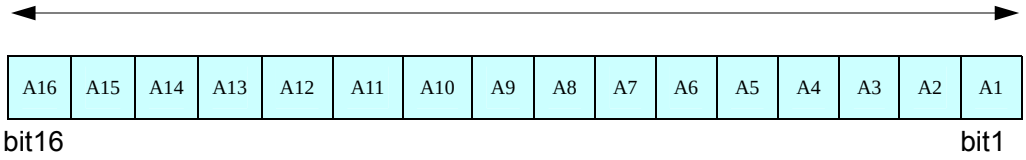
2.2.13 设备状态

下面提到的寄存器包含报警信息、消息和数字输出状态。报警分配可以见下面的位屏蔽。如果位为 1，报警激活。

所有值可以通过功能码 03H 和 04H 访问。

A 地址	值	字	类型	单位
1536	报警状态	1	UINT16	-

输出



A1: 过/欠电压

A2: 过电流

A3: 无电流

A4: 温度 1 过高

A5: 温度 2 过高

A6: 电压谐波

A7: 电流谐波

A8: 步容量递减

A9: 步缺陷

A10: 数字输入报警

A11: 系统错误

A12: 绝对有功过高

A13: 绝对无功过高

A14: 输出有效

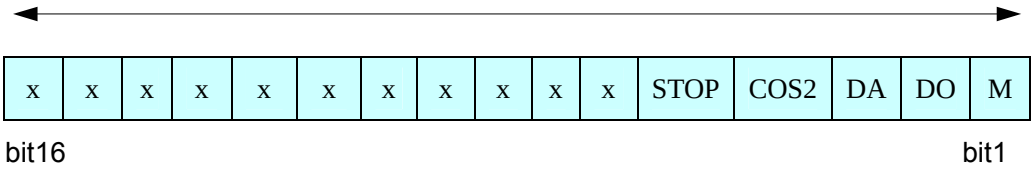
A15: 规则

A16: 未达到 cos ϕ

报警分配可以见下述位掩码。如果位为 1，输出或消息激活。

A 地址	值	字	类型	单位
1537	状态信息	1	UINT16	-

输出



M: 报警继电器

DO: 数字输出

COS2: 目标功率因数 cos ϕ 2 激活

STOP: 控制器停止，步关闭

HALT: 步冻结

通过 MODBUS 发生的所有设置会被立即使用，但须记住这些信息仅储存在工作存储器，停电后会消失。如果想长久储存，需要把数据储存到非易失性存储器中。

2.2.14 存储器设置

要把设置长久保存在非易失性存储器（EPROM）需用到下表。

所有这些值可以通过功能码 03H、04H 和 06H 访问。

A 地址	值	字	类型	单位
4096	保存参数数据到 EPROM	1	UINT16	-

如果要写“29864”到上述地址，参数数据将被长久储存到 EPROM。正确接收之后，在同一寄存器里以“1”表示确认。

由于闪存的使用寿命限制，此行为不能长久或者太经常进行。

设备存储器上的上面所述数值之外的其他数值，可能包含设备重要的安装数据。在不能确认时不要进行任何地址的写入操作。

3 问题解答

如果总线连接不能正确工作，请检查以下几点：

1、如果完全没有通信，错误一定发生于 BKR 表与 PC 之间！

可以由以下情况引起：

- 检查 BKR 表的波特率、校验位和地址调整，可能造成结构的改变。
- 可能总线 A 和 B 接反，必要时更改。
- 检查 RS485/RS232 转换器的调整，使用转换器的数据列表。
- 可能端口已经被其他设备使用，必要时停止这种多重预定。
- 检查端子和偏压电阻，必要时校正。

2、总线连接电缆是否有问题？所有插头是否正确？必要时置换。

3、RS485 的引进分配是否正确？必要时修改。

4、总线线路的屏蔽罩不能与总线的地相连。但屏蔽罩要连接保护性接地。必要时更正。

5、如果可以通信，但与用户的软件之间有问题，请按以下几点检查：

- 检查软件的总线地址、校验和波特率调整。
- 检查数据格式。