

西门子 1200 与 HV500 ModbusRTU 通讯配置文档

禾望电气

目录

1. 通讯平台搭建.....	3
2. 变频器通讯配置.....	4
2.1 变频器 Modbus 功能码	4
2.2 变频器 Modbus 寻址	4
2.3 通讯参数配置	5
2.4 频率读写实现方式	5
3. 硬件组态步骤.....	6
3.1 添加通讯模块	6
3.2 通讯端口配置	6
4. 软件编程.....	8
5. 通讯状态监控.....	19
5.1 PLC 通讯监控	19
5.2 变频器通讯监控	19
6. 注意事项.....	20

摘要：本文档基于西门子 S7-1200 系列 PLC 作为控制器，介绍西门子 CPU 1214 与 HV500 变频器在博途（TIA V15）中组态 ModbusRTU（简称 RS485）通讯总线的步骤，文档可作为自动化系统工程师和技术支持工程师在系统集成及产品应用的指导手册。

1. 通讯平台搭建

1) 软件：

- TIA Portal（版本 V15）
- HV500 变频器调试软件 hopeInsight（版本 004.008.000）

2) 硬件：

- CPU 1214C DC/DC/DC（版本 V4.2）
- CM 1241（RS422/485）（版本 V2.2）
- HV500 变频器（V107 版本）

3) 通讯参数设置：

设备名称	站地址	波特率	奇偶校验	数据位	停止位
CPU 1214C		9.6 kbps	无	8 位/字符	1
HV500	2	9.6 kbps	无	8 位/字符	1

表 1.1 通讯参数设置

4) RS485 通讯接线

西门子 CM 1241 通讯模块的接口为九针连接器（插孔式），CM 1241 通讯模块与 HV500 变频器的连接如图 1.2 所示。

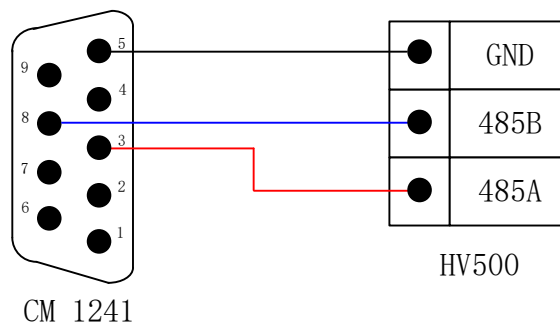


图 1.2 RS485 通讯接线

说明：3 号针脚—RS485 信号 B（+）；8 号针脚—RS485 信号 A（-）；5 号针脚—接屏蔽等电位点。

2. 变频器通讯配置

2.1 变频器 Modbus 功能码

HV500 变频器支持的 Modbus 功能码如表 2.1 所示。

功能码	含义
0x03H	读取多个 16 位寄存器的内容
0x06H	修改单个 16 位寄存器的内容，变频器掉电后，修改的值不保存
0x10H	修改多个 16 位寄存器的内容，变频器掉电后，修改的值不保存
0x17H	读取和修改多个 16 位寄存器的内容，变频器掉电后，修改的值不保存
0x47H	修改单个 16 位寄存器的内容，变频器掉电后，修改的值保存
0x6DH	修改多个 16 位寄存器的内容，变频器掉电后，修改的值保存
0x64H	读取多个 32 位寄存器的内容
0x65H	修改单个 32 位寄存器的内容，变频器掉电后，修改的值不保存
0x66H	修改多个 32 位寄存器的内容，变频器掉电后，修改的值不保存
0x67H	读写多个 32 位寄存器的内容，变频器掉电后，修改的值不保存
0x6CH	修改单个 32 位寄存器的内容，变频器掉电后，修改的值保存
0x6EH	修改多个 32 位寄存器的内容，变频器掉电后，修改的值保存

表 2.1 变频器功能码

2.2 变频器 Modbus 寻址

变频器的参数功能码需映射为 Modbus 的读写寄存器，其映射规则如下：

参数功能码由功能码组号和功能码 ID 号组成，如：参数功能码【S-25.01】的功能码组号是 25，功能码 ID 号是 01。功能码组号 = Modbus 寄存器地址高字节；功能码 ID 号 = Modbus 寄存器地址低字节。例如：求参数功能码【S-25.01】的 Modbus 寄存器地址，功能码组号 25 转换为十六进制为高字节是 19H；功能码 ID 号 01 转化成十六进制为低字节 01H；则 Modbus 寄存器地址为 1901H。

2.3 通讯参数配置

RS485 通讯参数主要配置从站地址【S-17.01】、通讯波特率【S-17.02】、通讯格式【S-17.03】、通讯超时时间【S-17.04】，本例以波特率 = 9.6kbps，奇偶校验 = 无，数据位 = 8 位/字符，停止位 = 1 为例，如图 2.2 所示。

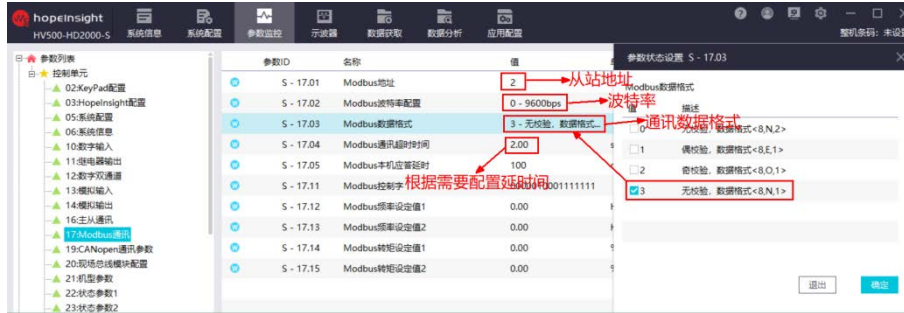


图 2.2 RS485 通讯配置

2.4 频率读写实现方式

变频器的频率设定值、输出频率、电机转速均为 32 位寄存器数据，需要 64H、65H、6CH、6EH 功能码才能更改寄存器的值。但西门子 PLC 不支持该功能码读写数据，只有把上述参数数据的值写入到 16 位寄存器，PLC 才能正常读写数据。

1) 频率设定值配置

频率设定值参数组【S-17.12】、【S-17.13】为 32 位寄存器数据，需 PLC 把给定频率写入到 16 位寄存器参数【S-76.151】，然后把参数【S-76.151】关联到频率给定参数【S-27.11】，如图 2.3 所示。

S - 27.09	主给定频率通道1设置	99 - 源选择模式
S - 27.11	主给定频率通道源选择1	S - 76.151 → 频率给定配置
S - 27.13	主给定频率通道切换源选择	0
S - 76.151	用户自定义变量1	1000 → 频率给定值10Hz -32768 32767

图 2.3 频率设定值配置

2) 读取输出频率配置

输出频率参数【S-22.36】为 32 为寄存器数据，需使用乘法自由功能块把输出频率【S-22.36】转换成 16 位寄存器数据，乘法自由功能块输出【S-76.23】是百分比数，在 PLC 程序里需要进行线性转换得到输出频率实际值。读取输出频率配置如图 2.4 所示。

W	S - 75.01	自由功能块使能	1 - 使能		
W	S - 75.06	执行周期组[0]	1	0	500
W	S - 76.21	运算“乘法”模块[1]输入1源选择	S - 22.36	→ 输出频率	
W	S - 76.22	运算“乘法”模块[1]输入2源选择	100%		
	S - 76.23	运算“乘法”模块[1]输出	20.00	→ 转换成16位寄存器	
W	S - 76.24	运算“乘法”模块[1]执行周期选择	0 - 执行周期组[0]		

图 2.4 读取输出频率配置

变频器参数配置完成后，需进行固化参数操作，在【系统配置】界面选择【参数固化】，点击【设置】按钮进行参数固化。

3. 硬件组态步骤

3.1 添加通讯模块

打开【设备视图】，添加 CPU 1214C，在【硬件目录】里找到【通讯模块】→【点对点】→【CM 1241 (RS422/485)】，拖拽此模块至 CPU 左侧即可，如图 3.1 所示。

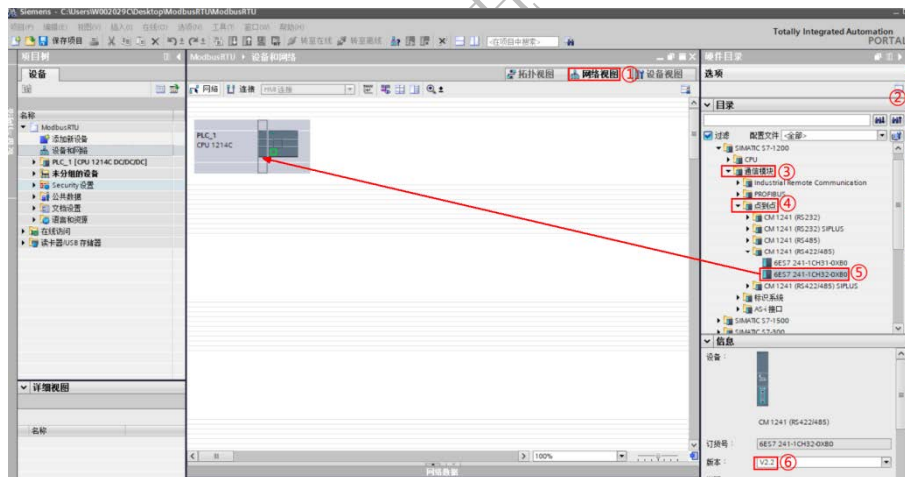


图 3.2 添加 RS485 通讯模块

注意：固件版本>=V2.1 的 CM 1241 RS422/485 模块，才支持新版本 Modbus RTU 指令集。

3.2 通讯端口配置

- 1) 在【设备视图】中用鼠标选中 CM1241 (RS422/485) 模块，在【属性】→【端口组态】中配置此模块硬件接口参数，本例以波特率 = 9.6kbps，奇偶校验 = 无，数据位 = 8 位/字符，停止位 = 1 为例，如图 3.2 所示。PLC 侧通讯端口参数配置须

与 HV500 的通讯参数配置一致，否则 PLC 与 HV500 变频器的 RS485 通讯不能正常建立。

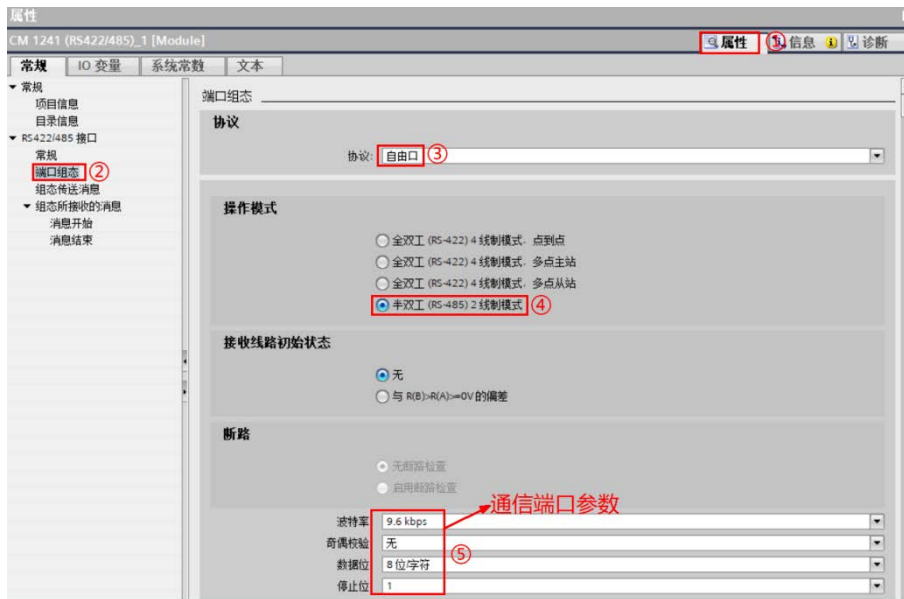


图 3.2 通讯端口参数配置

- 2) 激活 CPU 的【系统和时钟存储器功能】，如图 3.3 所示。（本例程中使用了 CPU 首次扫描位来实现 Modbus 的初始化，使用 CPU 的时钟信号来控制发送频率，该方法供参考。）

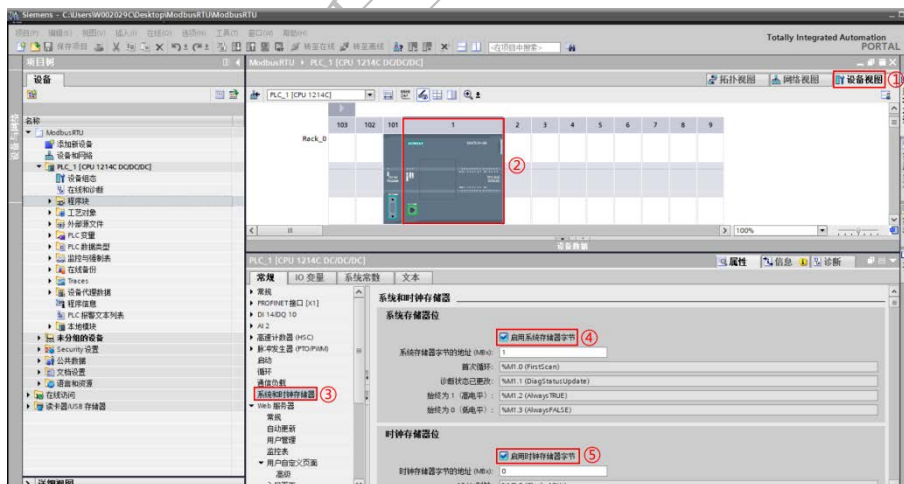


图 3.3 激活系统和时钟存储器功能

4. 软件编程

西门子 PLC 作为 ModbusRTU 主站，编程需要调用【Modbus_Comm_Load】指令和【Modbus_Master】指令。【Modbus_Comm_Load】指令通过 ModbusRTU 协议对通讯模块进行组态；【Modbus_Master】指令可通过由【Modbus_Comm_Load】指令组态的端口作为 Modbus 主站进行通讯，【Modbus_Comm_Load】指令的 MB_DB 参数必须连接到【Modbus_Master】指令的（静态）MB_DB 参数。当在程序中添加【Modbus_Comm_Load】和【Modbus_Master】指令时，将自动分配背景数据块。

- 1) 添加一个新 FB 块，将其命名【ModbusMaster 通讯程序】。在该 FB 块中以多重背景方式调用【Modbus_Comm_Load】和【Modbus_Master】指令，在【指令】目录下【通讯】→【通讯处理器】→【Modbus (RTU)】下，如图 4.1 所示。

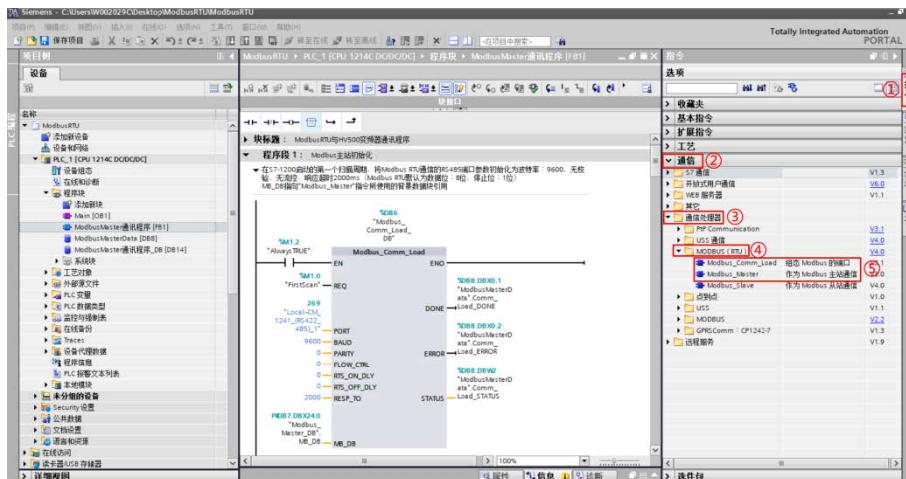


图 4.1 ModbusRTU 主站指令

- 2) 对【Modbus_Comm_Load】指令进行参数设置，该指令主要参数如表 4.2 所示。

参数和类型	数据类型	说明
EN	IN	Bool 注：Modbus RTU、Modbus_Comm_Load 指令使用 RDREC 和 WRREC 指令初始化 PTP 模块。但 RDREC/WRREC 指令异步运行，这意味着需要几次扫描才能完成指令运行。因此，您必须保持 Modbus_Comm_Load 指令的 EN 参数为真，直到 RDREC/WRREC 指令运行结束。
REQ	IN	Bool 通过由低到高的（上升沿）信号启动操作。 (仅版本 2.0)
PORT	IN	Port 安装并组态 CM 或 CB 通信设备之后，端口标识符将出现在 PORT 功能框连接的参数助手下拉列表中。分配的 CM 或 CB 端口值为设备配置属性“硬件标识符”。端口符号名称在 PLC 变量表的“系统常量”(System constants) 选项卡中分配。
BAUD	IN	UDInt 波特率选择： 300、600、1200、2400、4800、9600、19200、38400、57600、76800、115200，其它所有值均无效

参数和类型		数据类型	说明
PARITY	IN	UInt	奇偶校验选择： <ul style="list-style-type: none"> 0 - 无 1 - 奇校验 2 - 偶校验
FLOW_CTRL ¹	IN	UInt	流控制选择： <ul style="list-style-type: none"> 0 - (默认) 无流控制 1 - RTS 始终为 ON 的硬件流控制 (不适用于 RS485 端口) 2 - 带 RTS 切换的硬件流控制
RTS_ON_DLY ¹	IN	UInt	RTS 接通延时选择： <ul style="list-style-type: none"> 0 - (默认) 从 RTS 激活一直到传送消息的第一个字符之前无延时 1 到 65535 - 从 RTS 激活一直到传送消息的第一个字符之前以毫秒表示的延时 (不适用于 RS485 端口)。不管 FLOW_CTRL 选择为何, 都将应用 RTS 延时。
RTS_OFF_DLY ¹	IN	UInt	RTS 关断延时选择： <ul style="list-style-type: none"> 0 - (默认) 从传送最后一个字符一直到 RTS 转入非活动状态之前无延时 1 到 65535 - 从传送最后一个字符一直到 RTS 转入非活动状态之前以毫秒表示的延时 (不适用于 RS485 端口)。不管 FLOW_CTRL 选择为何, 都将应用 RTS 延时。
RESP_TO ¹	IN	UInt	响应超时： Modbus_Master 允许用于从站响应的的时间 (以毫秒为单位)。如果从站在此时间段内未响应, Modbus_Master 将重试请求, 或者在发送指定次数的重试请求后终止请求并提示错误。 5 ms 到 65535 ms (默认值 = 1000 ms)。
MB_DB	IN	Variant	对 Modbus_Master 或 Modbus_Slave 指令所使用的背景数据块的引用。在用户的程序中放置 Modbus_Master 或 Modbus_Slave 后, 该 DB 标识符将出现在 MB_DB 功能框连接的参数助手下拉列表中。
DONE	OUT	Bool	上一请求已完成且没有出错后, DONE 位将保持为 TRUE 一个扫描周期时间。(仅版本 2.0)
ERROR	OUT	Bool	上一请求因错误而终止后, ERROR 位将保持为 TRUE 一个扫描周期时间。STATUS 参数中的错误代码值仅在 ERROR = TRUE 的一个扫描周期内有效。
STATUS	OUT	Word	执行条件代码

表 4.2 Modbus_Comm_Load 指令参数说明

- 3) 首先要为【Modbus_Comm_Load】指令指定端口, 即该指令是针对哪个点对点模块进行参数化的。在硬件配置中, 每个硬件均有一个硬件标识符, 通过拖拽的方式, 将 Modbus 主站接口的硬件标识符拖至【Modbus_Comm_Load】指令的【Port】接口参数处, 如图 4.3 所示。

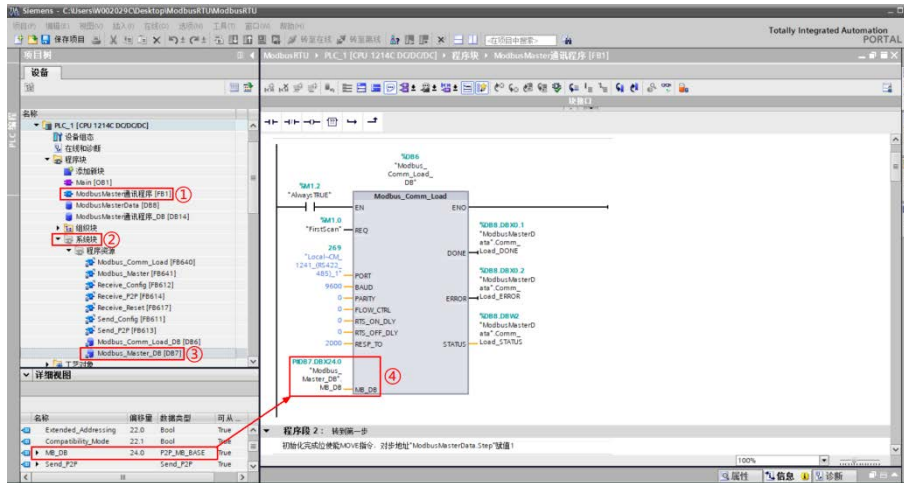


图 4.3 为 Modbus_Comm_Load 指令指定端口

- 4) 接下来，定义端口的工作模式。本示例中，点对点模块的工作模式为 RS485，所以需要将【Modbus_Comm_Load】背景数据中静态变量的【MODE】参数赋值为 4，赋值既可以通过【Move】指令来完成，也可直接修改该静态变量的默认值来实现，本实例使用后一种方法，如图 4.4 所示。

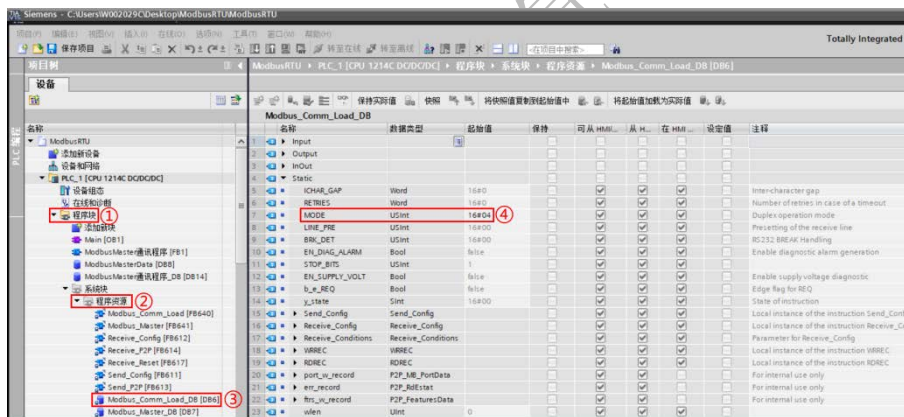


图 4.4 定义 ModbusRTU 主站端口工作模式为 RS485

- 5) 接下来，通过对【Modbus_Comm_Load】指令的【MD_DB】参数赋值，将【Modbus_Comm_Load】指令与【Modbus_Master】指令进行关联，即将【Modbus_Master】指令的背景 DB 块中静态变量【MB_DB】赋值给【Modbus_Comm_Load】指令的【MD_DB】，可以通过拖拽的方式来实现，拖拽路径如图 4.5 所示。

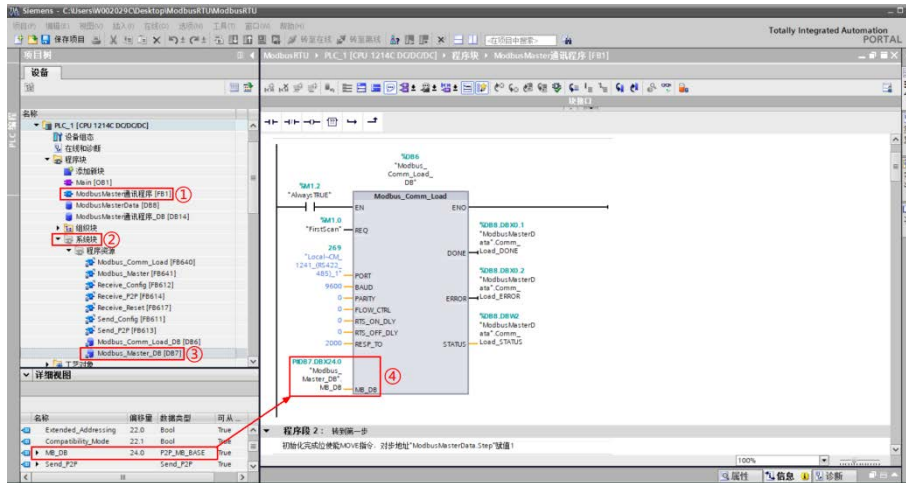


图 4.5 对 Modbus_Comm_Load 指令的 MB_DB 赋值

除以上操作外，对于【Modbus_Comm_Load】指令的 REQ 参数，本实例使用 PLC 的首个扫描位来完成。其它参数如波特率，奇偶校验等，请根据实际使用情况对这些参数进行赋值，所以上参数使用缺省设置即可。

6) 对【Modbus_Master】指令进行参数设置，该指令主要参数如表 4.6 所示。

参数和类型	数据类型	说明
REQ	IN	Bool 0 = 无请求 1 = 请求将数据传送到 Modbus 从站
MB_ADDR	IN	V1.0: USInt V2.0: UInt Modbus RTU 站地址： 标准寻址范围（1 到 247） 扩展寻址范围（1 到 65535） 值 0 被保留用于将消息广播到所有 Modbus 从站。只有 Modbus 功能代码 05、06、15 和 16 是可用于广播的功能代码。
MODE	IN	USInt 模式选择：指定请求类型（读、写或诊断）。请参见下面的 Modbus 功能表了解详细信息。
DATA_ADDR	IN	UDInt 从站中的起始地址：指定要在 Modbus 从站中访问的数据的起始地址。请参见下面的 Modbus 功能表了解有效地址信息。
DATA_LEN	IN	UInt 数据长度：指定此请求中要访问的位数或字数。请参见下面的 Modbus 功能表了解有效长度信息。
DATA_PTR	IN_OUT	Variant 数据指针：指向要写入或读取的数据的 M 或 DB 地址（未经优化的 DB 类型）。
DONE	OUT	Bool 上一请求已完成且没有出错后，DONE 位将保持为 TRUE 一个扫描周期时间。
BUSY	OUT	Bool • 0 - 无 Modbus_Master 操作正在进行 • 1 - Modbus_Master 操作正在进行
ERROR	OUT	Bool 上一请求因错误而终止后，ERROR 位将保持为 TRUE 一个扫描周期时间。STATUS 参数中的错误代码值仅在 ERROR = TRUE 的一个扫描周期内有效。
STATUS	OUT	Word 执行条件代码

表 4.6 Modbus_Master 指令参数说明

7) 由于 Modbus 指令读取或写入的数据区必须为指针寻址，所以必须是有绝对地址的区域方可访问，而 S7-1200 创建的 DB 块缺省为优化的 DB 块，变量没有绝对地址，无法直接使用。本例中我们创建的 DB 数据块【ModbusMasterData】，并在该 DB

块内创建了编写 ModbusRTU 通讯程序用的变量。然后右键该 DB 块选择【属性】，在其弹窗中的【属性】中将【优化的块访问】前的勾去掉，重新编译该 DB 块，就可看到每个变量都有偏移地址了，如图 4.7 所示。

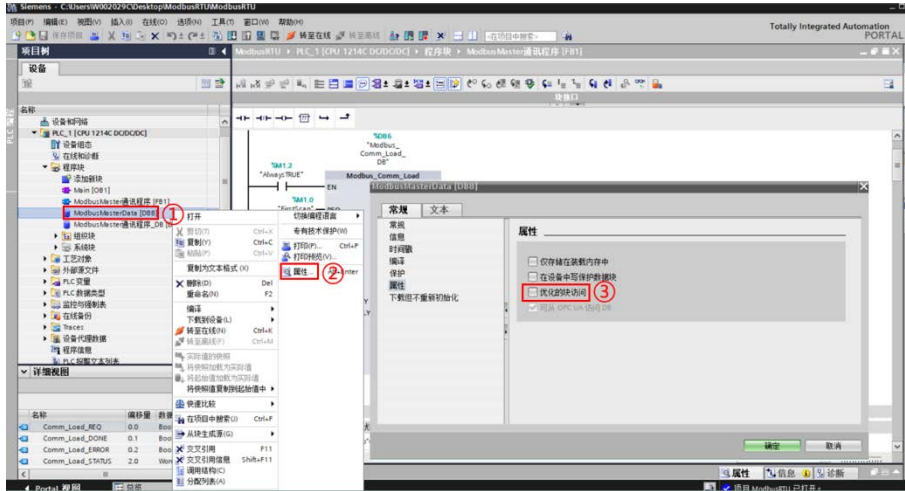


图 4.7 设置数据块非优化访问

8) 在设置调用【Modbus_Master】指令时，参数【MODE】和【DATA_ADDR】结合使用可指定在实际 Modbus 帧中使用的功能代码。表 4.8 显示了【MODE】参数、Modbus 功能代码和【DATA_ADDR】中 Modbus 地址范围之间的关系。

MODE	DATA_ADDR (Modbus 地址)	DATA_LEN (数据长度)	Modbus 功能 代码	运行和数据
0	1 到 9999	每个请求的位数 1 到 2000/1992 ¹	01	读取输出位: 0 到 9998
0	1000 到 19999 ¹	每个请求的位数 1 到 2000/1992 ¹	02	读取输入位: 0 到 9998
0	4000 到 49999 ¹	每个请求的字数 1 到 125/124 ¹	03	读取保持寄存器: 0 到 9998
	4000 到 465535 ⁰¹	1 到 125/124 ¹		0 到 65534
0	3000 到 39999 ¹	每个请求的字数 1 到 125/124 ¹	04	读取输入字: 0 到 9998
1	1 到 9999	每个请求的位数 1	05	写入一个输出位: 0 到 9998

MODE	DATA_ADDR (Modbus 地址)			DATA_LEN (数据长度)			Modbus 功能 代码	运行和数据
1				每个请求 1 个字			06	写入一个保持寄存器:
	4000	到	49999	1				0 到 9998
	4000	到	465535	1				0 到 65524
1				每个请求的位数			15	写入多个输出位:
	1	到	9999	2	到	1968/1960 ¹		0 到 9998
1				每个请求的字数			16	写入多个保持寄存器:
	4000	到	49999	2	到	123/122		0 到 9998
	4000	到	465534	2	到	123/122 ¹		0 到 65534
2 ²				每个请求的位数			15	写入一个或多个输出位:
	1	到	9999	1	到	1968/1960 ¹		0 到 9998
2 ²				每个请求的字数			16	写入一个或多个保持寄存器:
	4000	到	49999	1	到	123		0 到 9998
	4000	到	465535	1	到	122 ¹		0 到 65534
11	此功能将忽略 Modbus_Master 的 DATA_ADDR 和 DATA_LEN 操作数。						11	读取从站通信的状态字和事件计数器。状态字表示“忙”(0-不忙, 0xFFFF-忙)。事件计数器随着帧的每次成功处理而递增。
80				每个请求 1 个字			08	使用数据诊断代码 0x0000 检查从站状态(回送测试-从站返回请求的回应)
-				1			-	
81				每个请求 1 个字			08	利用数据诊断代码 0x000A 重新设置从站事件计数器
-				1			-	
104 ³				每个请求的字数			04	读取输入字
	0	到	65535	1	到	125/124 ¹	0 到 65535	
3 到 10, 12 到 79, 82 到 103, 105 到 255	-			-				保留

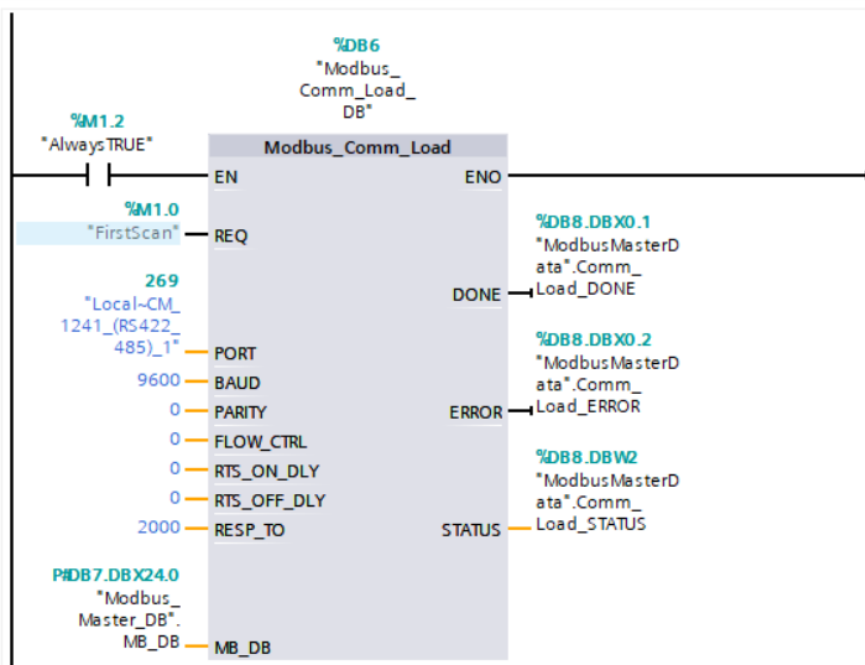
表 4.8 Modbus 功能码表

- 9) 当 Modbus RTU 网络中存在多个 Modbus RTU 从站或一个 Modbus RTU 从站同时需要多个作业, 例如需要读和写或者读多个区域等, 则需要调用多个【Modbus_Master】指令, 【Modbus_Master】指令之间需要采用轮询方式调用, 并且使用相同背景

数据块。因PLC读写HV500变频器的参数地址不连续，需要调用4个【Modbus_Master】指令进行轮询才能完成读写操作，轮询程序如图4.9所示。

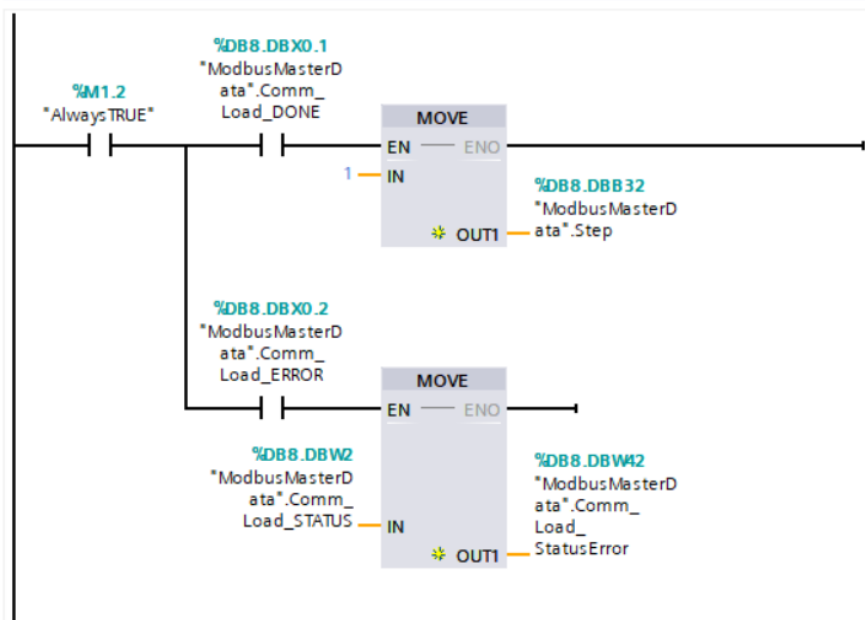
程序段 1：Modbus主站初始化

在S7-1200启动的第一个扫描周期，将Modbus RTU通信的RS485端口参数初始化为波特率：9600，无校验，无流控，响应超时2000ms（Modbus RTU默认为数据位：8位，停止位：1位）MB_DB指向“Modbus_Master”指令所使用的背景数据块引用



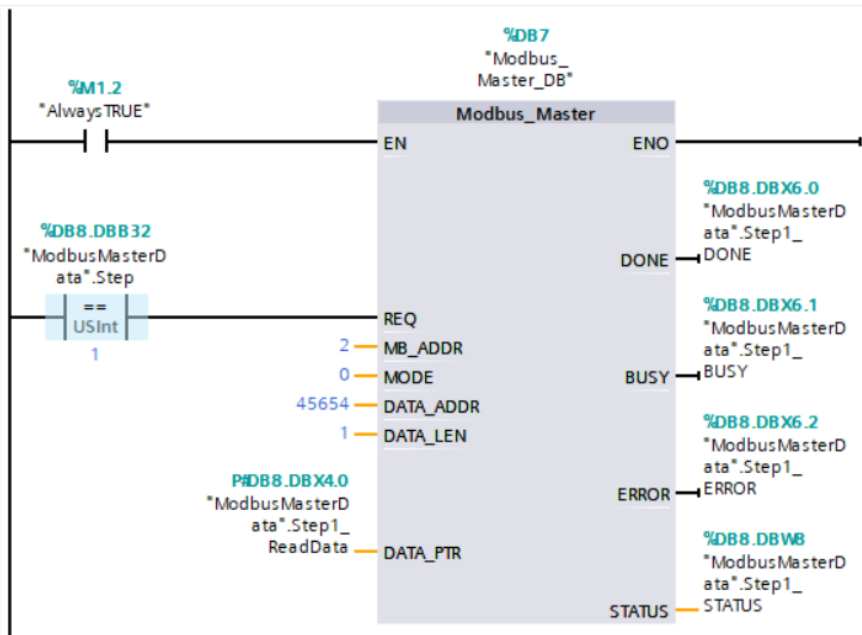
程序段 2：转到第一步

初始化完成位使能MOVE指令，对步地址“ModbusMasterData.Step”赋值1



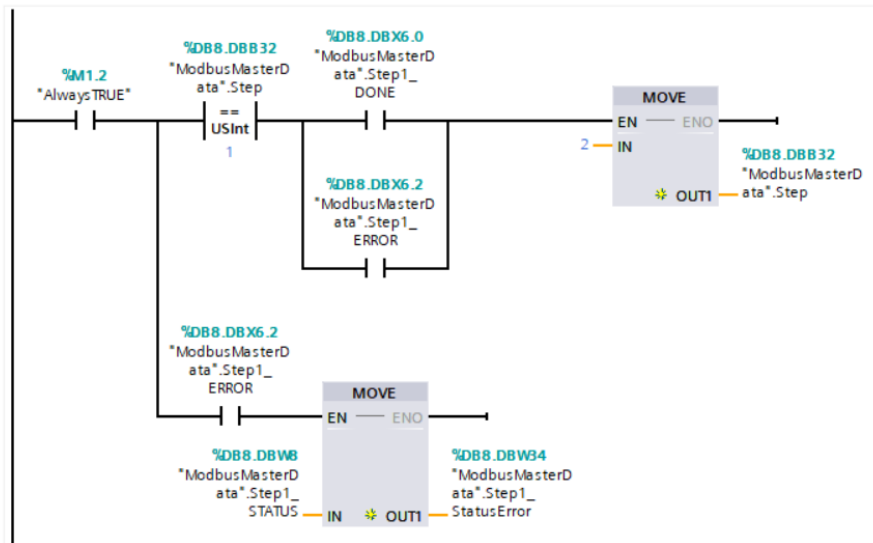
▼ 程序段 3： 第一步读取变频器状态字状态【S-22.21】

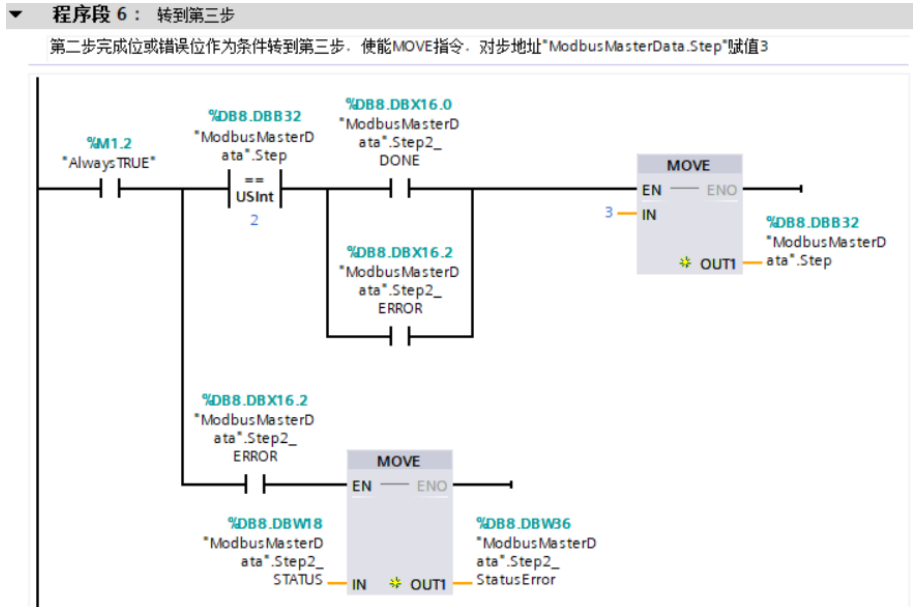
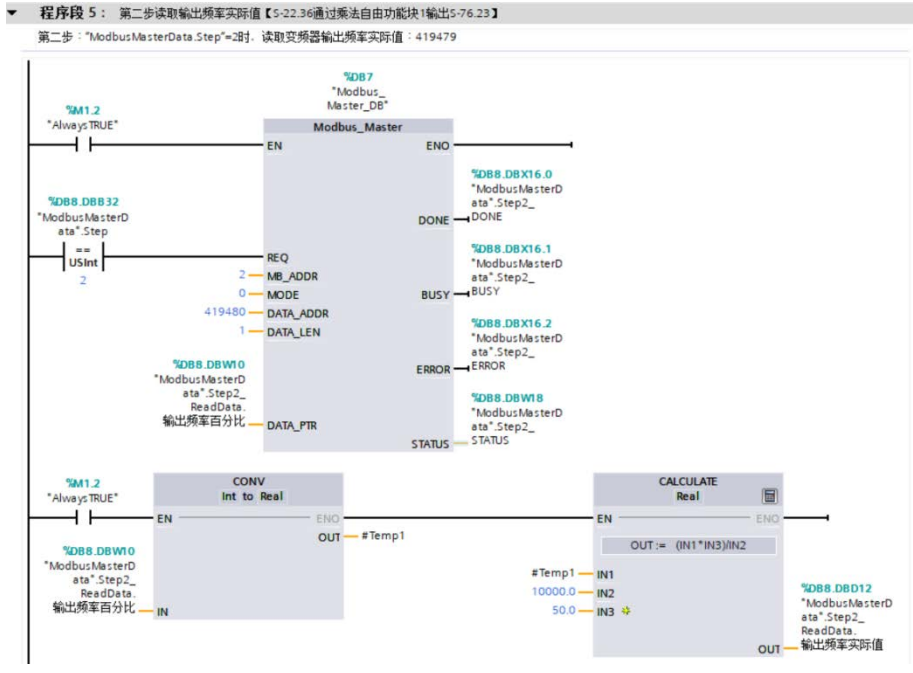
第一步：“ModbusMasterData.Step”=1时，读取变频器状态字状态：45653



▼ 程序段 4： 转到第二步

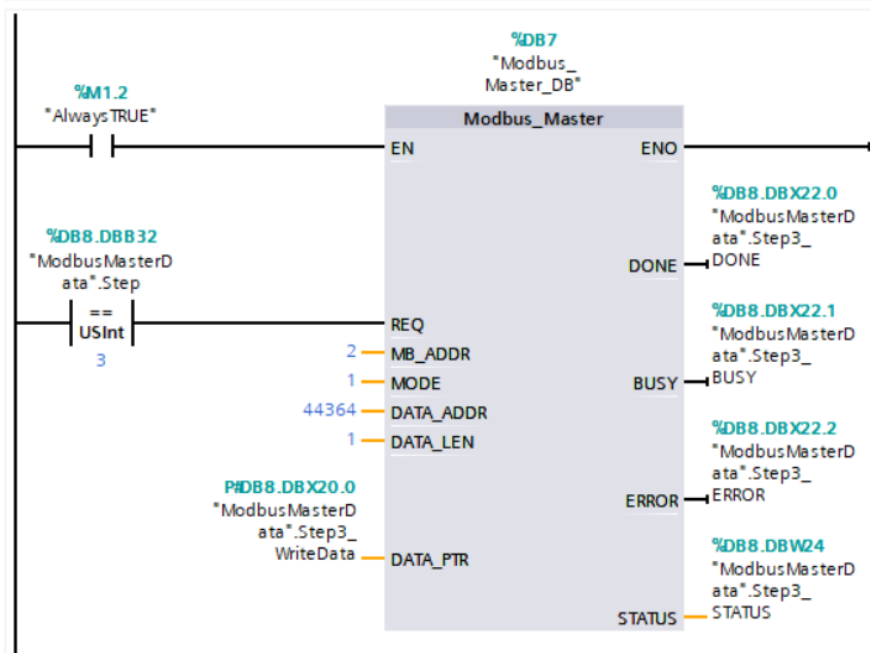
第一步完成位或错误位作为条件转到第二步，使能MOVE指令，对步地址“ModbusMasterData.Step”赋值2





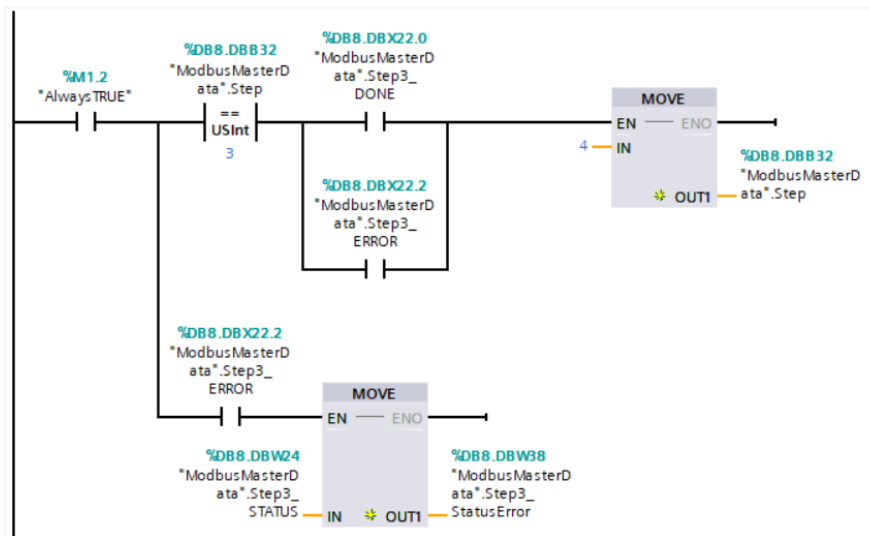
▼ 程序段 7：第三步写变频器控制字【S-17.11】

第三步：“ModbusMasterData.Step”=3时，写变频器控制字：44363



▼ 程序段 8：转到第四步

第三步完成位或错误位作为条件转到第四步，使能MOVE指令，对步地址“ModbusMasterData.Step”赋值4



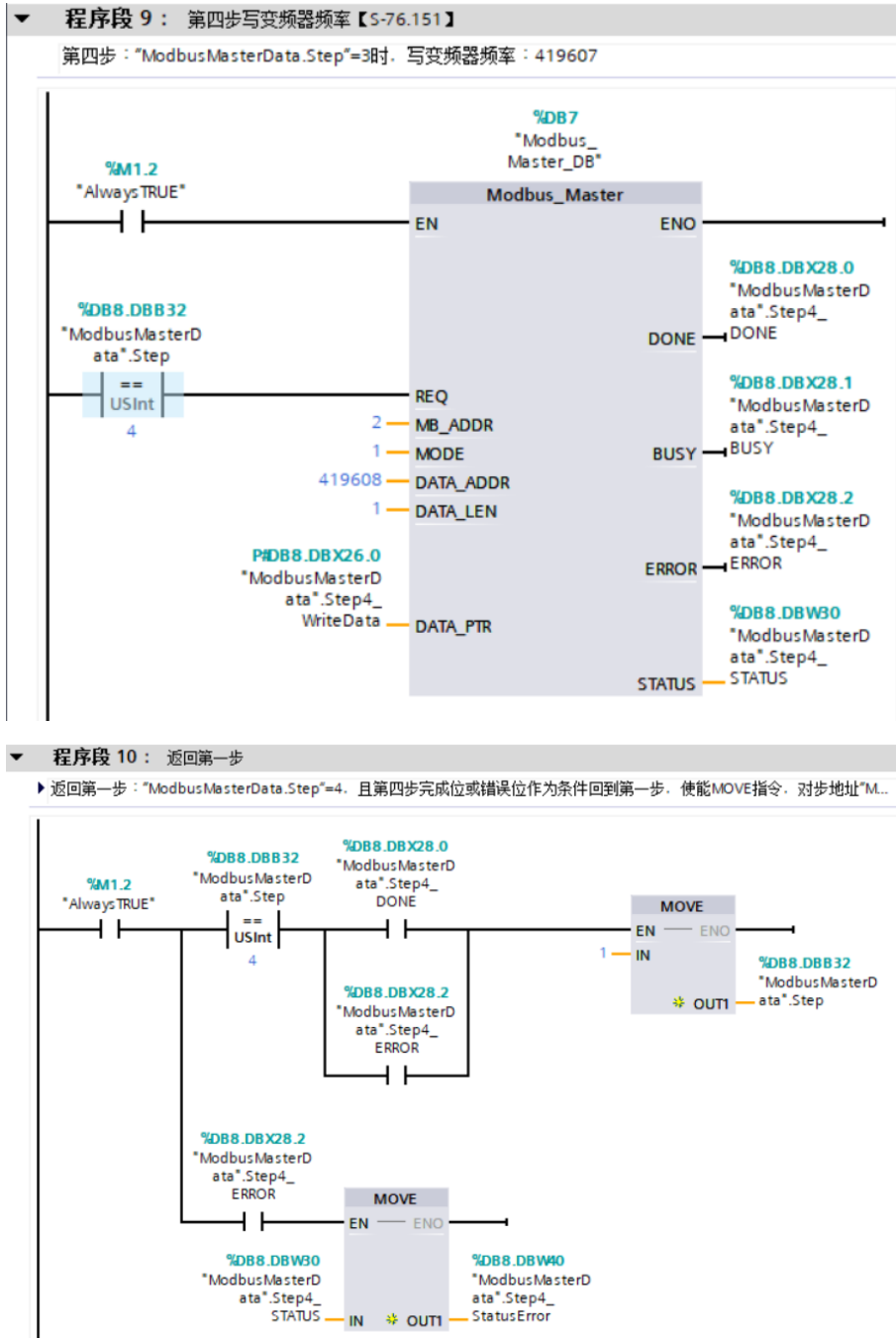
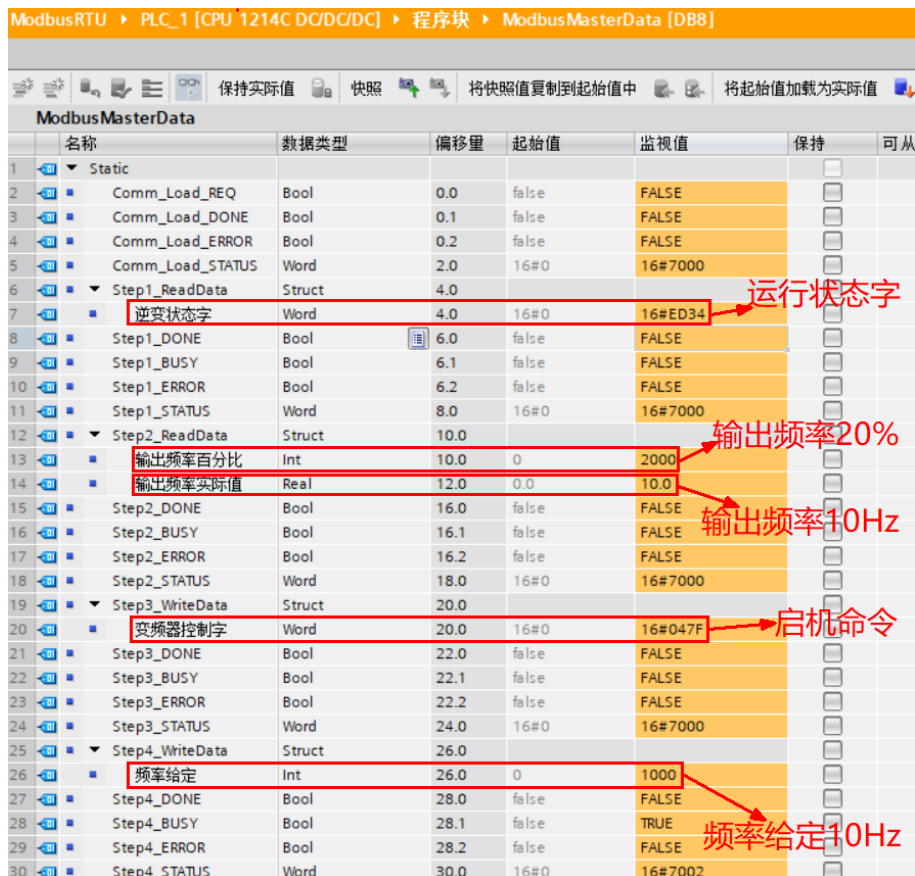


图 4.9 Modbus_Master 轮询程序

5. 通讯状态监控

5.1 PLC 通讯监控

通讯程序编写完成后，将程序下载到 PLC 后，可以看到通讯模块对应的收发 LED 指示灯在闪烁，表示端口正在发送/接收数据。双击打开 DB 数据块【ModbusMasterData】，点击【全部监视】按钮进入在线监控状态。通过 DB 数据块变量【DB8.DBX22.0】变频器控制字赋值 16#047F 进行变频器启机，通过 DB 数据块变量【DB8.DBX26.0】频率给定赋值 10000，给定运行频率 10Hz。此时可看到 PLC 读取到了 HV500 变频器的运行数据，如图 5.1 所示。



名称	数据类型	偏移量	起始值	监视值	保持	可从
Static						
Comm_Load_REQ	Bool	0.0	false	FALSE		
Comm_Load_DONE	Bool	0.1	false	FALSE		
Comm_Load_ERROR	Bool	0.2	false	FALSE		
Comm_Load_STATUS	Word	2.0	16#0	16#7000		
Step1_ReadData	Struct	4.0				
逆变状态字	Word	4.0	16#0	16#ED34		
Step1_DONE	Bool	6.0	false	FALSE		
Step1_BUSY	Bool	6.1	false	FALSE		
Step1_ERROR	Bool	6.2	false	FALSE		
Step1_STATUS	Word	8.0	16#0	16#7000		
Step2_ReadData	Struct	10.0				
输出频率百分比	Int	10.0	0	2000		
输出频率实际值	Real	12.0	0.0	10.0		
Step2_DONE	Bool	16.0	false	FALSE		
Step2_BUSY	Bool	16.1	false	FALSE		
Step2_ERROR	Bool	16.2	false	FALSE		
Step2_STATUS	Word	18.0	16#0	16#7000		
Step3_WriteData	Struct	20.0				
变频器控制字	Word	20.0	16#0	16#047F		
Step3_DONE	Bool	22.0	false	FALSE		
Step3_BUSY	Bool	22.1	false	FALSE		
Step3_ERROR	Bool	22.2	false	FALSE		
Step3_STATUS	Word	24.0	16#0	16#7000		
Step4_WriteData	Struct	26.0				
频率给定	Int	26.0	0	1000		
Step4_DONE	Bool	28.0	false	FALSE		
Step4_BUSY	Bool	28.1	false	TRUE		
Step4_ERROR	Bool	28.2	false	FALSE		
Step4_STATUS	Word	30.0	16#0	16#7002		

图 5.1 PLC 读取 HV500 数据

5.2 变频器通讯监控

通过 RS485 通讯给定运行命令和运行频率后，在变频器侧可以监控到变频器运行状态，如图 5.2 所示。



图 5.2 变频器运行状态监控

6. 注意事项

- 1) 【Modbus_Master】指令的【DATA_PTR】参数用于指向要进行数据写入或数据读取的数据区域地址，该数据区域在 Modbus_Master 【V3.0】支持非优化（标准的）数据块。其收发数据区必须是可通过指针寻址的，即必须是有确切地址的数据区；而优化的 DB 块由于其内部变量没有地址，所以收发数据区均不能使用优化的 DB 块，必须使用标准 DB，否则 PLC 运行会报错。从 Modbus_Master 【V4.0】版本开始支持优化 DB 块。
- 2) 【Modbus_Comm_Load】指令的初始化信号端【REQ】和【Modbus_Master】指令的发送/接收信号端【REQ】必须使用边沿信号触发，否则初始化、发送/接收都无法完成。
- 3) 【Modbus_Comm_Load】指令背景数据块中的静态变量【MODE】用于描述点对点模块的工作模式，有效的工作模式包括：
 - 0 = 全双工（RS232）
 - 1 = 全双工（RS422）四线制模式（点对点）
 - 2 = 全双工（RS422）四线制模式（多点主站）
 - 3 = 全双工（RS422）四线制模式（多点从站）
 - 4 = 半双工（RS485）二线制模式

该静态变量【MODE】默认数据为 0（RS232 全双工模式），需要根据点对点模块实际组态修改该数值，本例中 CM PtP 模块工作在 RS485 半双工模式需要将该数值修改为【4】。