

2 硬件以及通讯原理分析

西门子 S7-200 系列 PLC 拥有 RS-485 串行口，所以要使 MCS51 单片机与 S7-PLC 进行通讯，可以采用几种通讯方式。其中之一就是可以通过 MCS-51 的串行口与 MAX485 芯片相接，然后与 S7-200 PLC 的 RS-485 口进行通讯，其硬件连接如图 1 所示。

S7-200 PLC 是串行通讯方式最为丰富的小型 PLC，支持多种通信协议，如点对点接口协议（PPI 协议）、多点接口协议（MPI 协议）和 PROFIBUS 协议以及自由通信协议等。其中自由通信协议又叫用户定义协议，利用自由端口模式，可以实现用户定义的通信协议，连接多种智能设备，使用起来非常方便，在第三方工程接入中取得了巨大的成功。

在自由端口模式下，PLC 的串行通信接口由用户来控制，通过梯形图程序以及和单片机的汇编语言进行配合，来完成中断、字符接收中断、发送完成中断等，通信协议由用户完全控制。这时单片机处于主机状态，由单片机主动发送握手信号，PLC 接到信号后被动反馈信息即可。

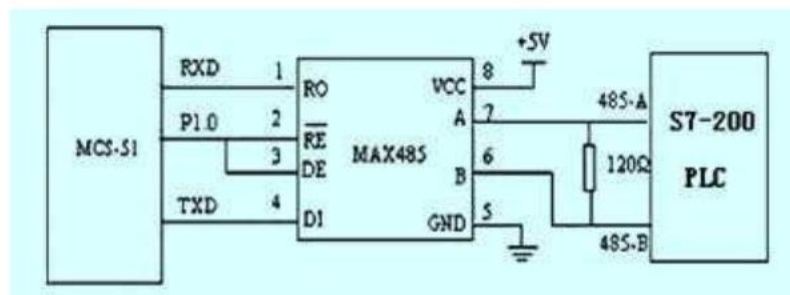


图 1 MCS-51 单片机与 S7-200 的硬件接线图

3 通信系统设计

3.1 通信协议设计

定义根据经验和有关参考资料，定义协议结构和参数。

(1) 通信波特率为 9.6kbps，无校验，8 个数据位，1 个可编程位，1 位起始位，1 位停止位。

(2) 定义通信协议的数据流结构的格式为起始码、命令码、元件首址、字节数、数据块、BCC 校验码和结束码。

● 起始码：表示单片机与 PLC 开始发送数据，是数据流第一个字符，告诉 PLC 开始进行通信了，可以用 00H 表示

● 命令码：表示单片机对 PLC 的各种操作：

40H：读取目标元件 I、Q、V、M、SM、L、T、C 等的的数据或状态；

41H：修改目标元件 I、Q、V、M、SM、L、T、C 等的的数据或状态；

42H：强制目标单元为 ON；

43H：强制目标单元为 OFF；

● 元件首址：表示 PLC 内部的元件类型以及寄存器的地址（但不能表示一个位地址）。前两个字节表示寄存器类型，后两个字节表示寄存器号。00 00 (H)：I 寄存器区 01 00 (H)：Q 寄存器区。02 00 (H)：M 寄存器区 08 00 (H)：V 寄存器区；

● 字节数：从元件首地址起，读取或写入 PLC 元件的数据个数数据块：准备读取或者写入 PLC

的数据或状态：

● BCC 校验码：在传输过程中，指令有可能受到任何的干扰而使原来的数据信号发生扭曲，此时的指令当然是错误的，为了侦测指令在传输过程中发生的错误，接收方必须对指令作进一步的确认工作，以防止错误的指令被执行，最简单的方法就是使用校验码。BCC 校验码的方法就是将要传送的字符串的 ASCII 码以字节为单位作异或和，并将此异或和作为指令的一部分传送出去；同样地，接收方在接到指令后，以相同的方式对接收到的字符串作异或和，并与传送方所送过来的值作对比，若其值相等，则代表接收到的指令是正确的，反之则是错误的

● 结束码：结束字符标志着指令的结束，在本例中被定义为 FFH，不同的 PLC 从站可以定义不同的结束字符以接收针对该 PLC 的指令。

3.2 通信程序的实现

(1) 单片机端程序的实现。单片机在主程序中初始化，采用串行口工作方式 3[2]，波特率为 9.6kbps，采用单片机作为主机，向 PLC 进行呼叫，定期读取数据或者写入数据，其程序流程图参见图 2。

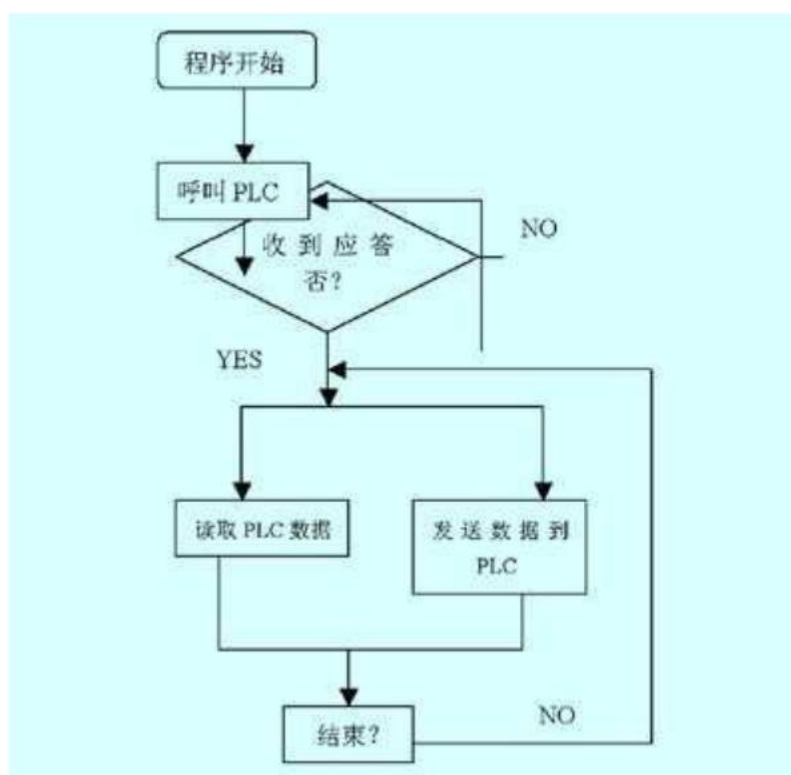


图 2 单片机端通讯程序流程图

(2) PLC 端程序流程图的实现。PLC 端作为从机，采用梯形图或者 STL 编程，主要是先设置通讯协议，然后按照协议把采集到的数据进行处理，再发送给主机单片机，其具体的程序流程图如图 3 所示。

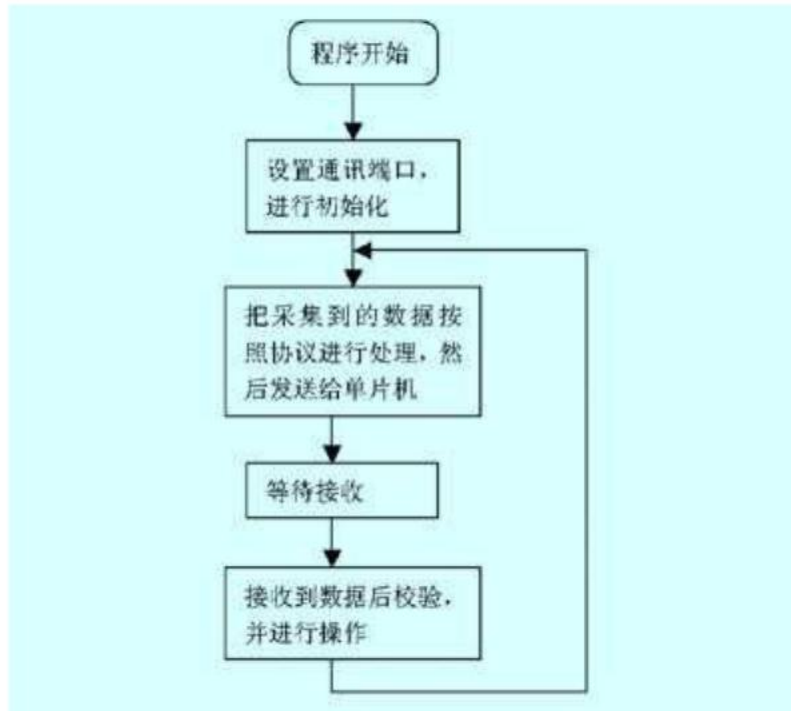


图 3 PLC 端通讯程序流程图

4 结束语

本文利用单片机与 PLC 的串行通信方法, 成功的应用于多个项目中, 实际表明该方法简单可靠, 成本低, 而且易于扩充经济实用的其它功能, 如 A/D、D/A 等功能, 取得了较好的社会效益和经济效益。