

S R 90 调节器通讯指南

2002. 4

本资料和开发的学习软件, 作为用户学习 SR90 系列仪表通讯编程的参考, 不足之处请给与指正。

济南金曼顿自动化技术有限公司

联系电话：0531-86960655 传真：0531-82391088

公司网址：<http://www.sdcystal.cn/>

说明：用户在购买 SR90 系列带通讯接口产品时，将随机提供该学习软盘。

————— 目录 —————

1. 软件清单
2. CC90 的使用方法
3. 进入通讯命令学习前的准备工作
4. 通讯协议以及 BASIC 程序方法
5. SR90 系列仪表的通讯命令
6. BASICA 的程序通讯软件说明：
7. 附录：A. 通讯串口接线方法
B. 有关 RS422/485 通讯口的技术数据

1. 软件清单

在软盘内，提供了下述的软件和资料

SR90 通讯协议.DOC - SR90 调节器通讯指南(WORDS 文件)

BASICA.EXE - 高级 BASIC 语言

SR90.BAS - 通讯测试软件

232T.BAS - BASIC 程序的 PC 机 232 口及先锋 RS422 口测试软件

2. SR90 通讯协议

1) 串口接线

- ①计算机与带 RS-232C 通讯口的连线
- ②计算机与希曼顿 RS-232C/RS-485A 通讯变换器连线
- ③RS-232C/RS-485 通讯变换器与仪表 RS-485 通讯口的连线
- ④D 型 25 针、九针串口接线对照表

2) 通讯协议

3) 参数设置

设置调节器通讯地址和使用的 PC 机串口, 及通讯参数设置。

3. 进入通讯命令学习前的准备工作

3-1. 初次连接系统的准备工作 (仪表未连接)

- 1.) 参照串口接线窗口和附录 A. 通讯串口接线方法, 对系统进行正确的接线。
- 2.) PC 机 RS232 通讯口正常 (包括地线、握手信号), 将 SD, RD 端短接。
- 3.) RS232 接口至 RS232/RS485 转换器连线是否正确。

注意: 9 针与 25 针串口的定义区别。

- 4.) 将 RS232/RS485 转换器的 RS422 输出端发送、接收短接, 测转换器好坏。
- 5.) RS232/RS485 转换器到仪表通讯口的连线正确。
- 如果远距离通讯(1200 米), 利用示波测量发送波形的前沿, 确定通讯线路的传输品质, 选择合适的通讯波特率。

注: 如采用 PC 机内式转换卡, 可省去前两步骤。

其它的操作:

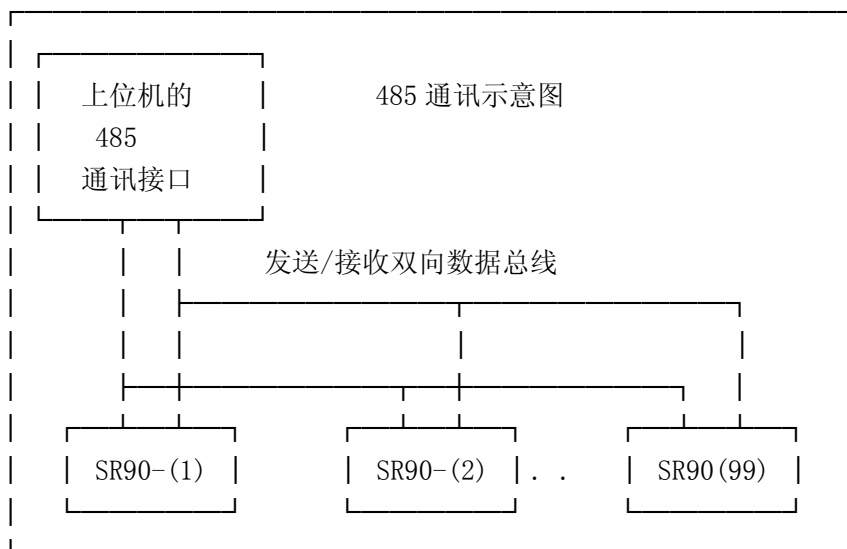
连接仪表且上电, 确信仪表已进行了有关的通讯参数(地址、波特率)设定。

在学习软件中相应的画面应设置与仪表相一致的 PC 机通讯地址和字符参数, 否则将不能正常通讯。**注: 仪表的有关设定请参阅 SR80 操作流程圖**

4. 通讯协议以及 BASIC 程序方法

4-1. 通讯的含意

RS232 接口, 只能单台点对点的通讯, 不能进行总线的并联, 但通讯软件和 485 方式相同



RS485 通讯采用差动的两线发送, 两线接收的双向数据总线两线制方式。上位机和下位调节器的内部接收器的接收高(RD+)和低(RS-)线以及内部发送器的发送高(SD+)和低(SD-)线都挂在数据总线上, 平时内部发送器的发送线处于高阻关闭态。如下图通讯过程示意图所示, 通常上位机是讲者, 下位调节器是听者, 并按主、从方式进行通讯, 多台仪表的通讯靠地址(设备号)的不同来区分。通讯中, 发送方需将发送线置于低阻态。发送完成后, 发送线需重新恢复到高阻关闭态。接收方在接收数据完成后, 又成为发送方。

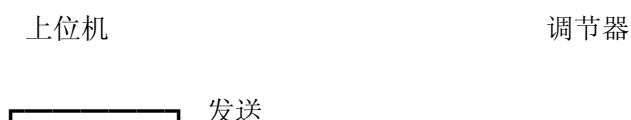
因此, RS485 接口存在着双向数据总线转换冲突问题。在上位机可由软件调整, 下位可由仪表的 RS485 延时时间窗口调整。

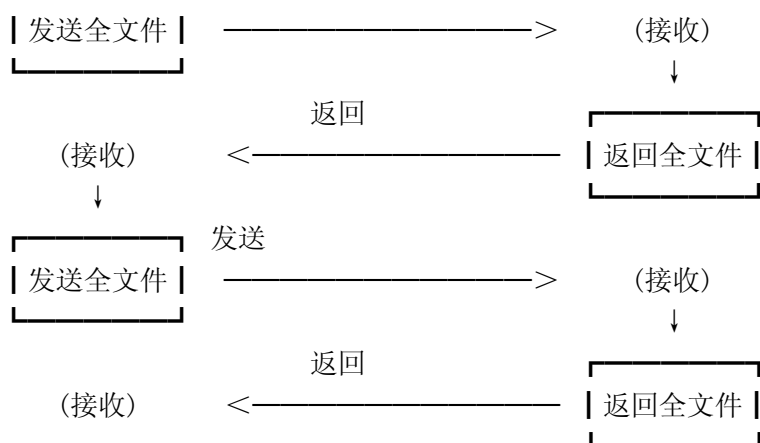
建议: 选用 RS485 接口的仪表时, 可采用研华 5020 型 RS232/RS485 智能通讯转换模块, 编程时无需考虑总线切换的问题。

注意: 通讯时, 上位机必须根据调节器设定的地址, 共同约定的数据格式, 波特率等通讯规约, 发送通讯文件, 下位调节器在接收地址符合, 接收字符格式和校验正确后才能进行正常的通讯。

4-2. 通讯协议说明:

通讯协议的通讯过程示意图





4-3. 发送全文件和返回全文件的组成

4.3.1 通讯控制符的三种格式: 1. STX_ETX_CR 2. STX_ETX_CRLF 3. @_:CR

4.3.2 通讯发送格式

a	b		c		d	e				f	g		h	i		j	
(1)	(2)	(3)	(4)		(5)	(6)	(7)	(8)	(9)	(10)	(11)		(12)	(13)	(14)	(15)	(16)
STX	0	1	1		R	0	1	0	0	0	----		ETX	D	A	CR	
STX	0	1	1		W	0	1	8	c	0	,****		ETX	7	8	CR	

1. 通讯发送格式的解释

- 通讯的起始符, [(1)一位, STX: (02H) 或 "@" (40H)]
- 通讯下位机地址 [(2)、(3)两位], 由 8 位二进制组成。地址范围 1~99 (1:0000 0001~0110 0110), 这 8 位二进制码被分成高 4 位和低 4 位, 其中高 4 位被送入(2)中, 低 4 位被送入(3), 并转换成 ASCII 码。
- 通讯下位机地址的子地址 [(4)一位], 这位被固定为 1。
- 通讯命令类型 [(5)一位]。"R" (52H), 表明在上位机发送或仪表应答中的读命令。"W" (57H), 表明在上位机发送或仪表应答中的写命令。"B" 表明在上位机以广播方式发送命令, 但 SR90 不支持广播方式, "B" 只作为预留命令。
- 通讯命令代码 [(6)、(7)、(8)、(9)四位]。是 16 位二进制代码 (0~65535), 这 16 位被分成四组, 并转换成相应的 ASCII 码。命令代码详见命令代码表。
- 通讯命令连续读代码 [(10)一位]。表明上位机要连续读取多少个参数。这位取值范围 "0" (30H) ~ "9" (39H), 十个数。实际的连续读参数的个数=表明的数值+1。
- 通讯数据 [(11)这位的数据量决定于这位的数据, 既这位的数据长度不定]。数据总是以 ", " (2CH), 数据项与数据项之间不需要任何分割符。数据的长度主要取决于第(10)的方式。每一个数据项由 16 位二进制代码组成 (1 个字), 每 4 位被分成一个数据单元, 同时每个数据单元又被转换成 ASCII 数据。当(5)位为 "R" 读命令时, 此位不写。

	第一数据项				第二数据项					第 N 数据项			
	高位			低位	高位			低位		高位			低位
" , " 2CH	第一单元	第二单元	第三单元	第四单元	第一单元	第二单元	第三单元	第四单元		第一单元	第二单元	第三单元	第四单元

- 数据发送结束符 [(12)一位, ETX (03H) 或 ":" (3AH)]。所有的数据和命令再此位之前都以发送完成, 遇到此字符表明结束。
- BCC 块校验 [(13)、(14)两位] 三种 BCC 块校验和无校验。上位机的 BCC 校验应通过软件处理。仪表的 BCC 校验可在 [1-34] 窗口设置。当 BCC 校验结果有错误时, 将没有应答。BCC 校验数据被分成高 4 位和低 4

位，并被转换成 ASCII 码，

(13):高 4 位的 ASCII 码。(14):低 4 位的 ASCII 码。

1). ADD 块校验

例: STX 0 1 1 R 0 1 0 0 9 EXT E 3 CR LF
(02H)+(30H)+(31H)+(31H)+(52H)+(30H)+(31H)+(30H)+(30H)+(39H)+(03H)=1E3H
BCC 校验结果 (13): "E"=45H (14): "3"=33H

2). ADD_TWO'S CMP 块校验

例: STX 0 1 1 R 0 1 0 0 9 EXT 1 D CR LF
(02H)+(30H)+(31H)+(31H)+(52H)+(30H)+(31H)+(30H)+(30H)+(39H)+(03H)=1E3H
BCC 校验结果 (13): "1"=31H (14): "D"=44H

3). XOR 块校验

例: STX 0 1 1 R 0 1 0 0 9 EXT 1 D CR LF
(02H)⊕(30H)⊕(31H)⊕(31H)⊕(52H)⊕(30H)⊕(31H)⊕(30H)⊕(30H)⊕(39H)⊕(03H)=59H
BCC 校验结果 (13): "5"=35H (14): "9"=39H

j 回车符[(15)、(16)一位或两位 CR (0DH) 或 CRLF] 全文结束符既回车符。CR 或 CR LF 可以选择。

4). None 无校验

4.3.3 通讯应答格式

a	b		c		d	e		g		h	i		j	
(1)	(2)	(3)	(4)		(5)	(6)	(7)	(11)		(12)	(13)	(14)	(15)	(16)
STX	0	1	1		R	0	1	----		ETX	3	C	CR	
STX	0	1	1		W	0	1	,***		ETX	4	E	CR	

1. 通讯应答格式的解释

- a 通讯的起始符，[(1)一位，STX: (02H) 或 "@" (40H)]
- b 通讯下位机地址[(2)、(3)两位]，由 8 位二进制组成。地址范围 1~99(1:0000 0001~0110 0110)，这 8 位二进制码被分成高 4 位和低 4 位，其中高 4 位被送入(2)中，低 4 位被送入(3)，并转换成 ASCII 码。
- c 通讯下位机地址的子地址[(4)一位]，这位被固定为 1。
- d 通讯命令类型[(5)一位]。"R" (52H), 表明在上位机发送或仪表应答中的读命令。"W" (57H), , 表明在上位机发送或仪表应答中的写命令。"B" 表明在上位机以广播方式发送命令,但 SR80 不支持广播方式,"B" 只作为预留命令。
- e 应答代码[(6)、(7)两位]。是 8 位二进制代码 (0~255)，这 8 位被分成高 4 位和低 4 位，并转换成相应的 ASCII 码。应答代码详见应答代码表。(6):高 4 位的 ASCII 码。(7):低 4 位的 ASCII 码。
- g 通讯数据[(11)这位的数据量决定于这位的数据，既这位的数据长度不定]。数据总是以", " (2CH), 数据项与数据项之间不需要任何分割符。数据的长度主要取决于第(11)的方式。每一个数据项由 16 位二进制代码组成(1 个字)，每 4 位被分成一个数据单元，同时每个数据单元又被转换成 ASCII 数据。当(5)位为 "W" 写命令时，此位不用。

	第一数据项				第二数据项					第 N 数据项			
	高位			低位	高位			低位		高位			低位
" , "	第一	第二	第三	第四	第一	第二	第三	第四		第一	第二	第三	第四
2CH	单元	单元	单元	单元	单元	单元	单元	单元		单元	单元	单元	单元

h 数据发送结束符[(12)一位，ETX (03H) 或 ":" (3AH)]。所有的数据和命令再此位之前都以发送完成，遇到此字符表明结束。

- (13):高 4 位的 ASCII 码。(14):低 4 位的 ASCII 码。

例: STX 0 1 1 R 0 1 0 0 9 EXT E 3 CR LF
 (02H)+(30H)+(31H)+(31H)+(52H)+(30H)+(31H)+(30H)+(30H)+(39H)+(03H)=1E3H
 BCC 校验结果 (13): "E"=45H (14): "3"=33H

例: STX 0 1 1 R 0 1 0 0 9 EXT 1 D CR LF
 (02H)+(30H)+(31H)+(31H)+(52H)+(30H)+(31H)+(30H)+(30H)+(39H)+(03H)=1E3H
 BCC 校验结果 (13): "1"=31H (14): "D"=44H

例: STX 0 1 1 R 0 1 0 0 9 EXT 1 D CR LF
 (02H) (30H)+(31H)+(31H)+(52H)+(30H)+(31H)+(30H)+(30H)+(39H)+(03H)=1E3H
 BCC 校验结果 (13): "5"=35H (14): "9"=39H

- 4). None 无校验

1. 读命令

d	e					f
(5)	(6)	(7)	(8)	(9)	(10)	
R	0	4	0	0	9	
52H	30H	34H	30H	30H	39H	

e: 这位表明这个读命令是读 SV1 的控制输出 1 的比例带的参数。

f: 这位表明这读命令要读多少个参数。

具体含义如下:

[illegible]

这位表明这读命令要读多少个参数。	=4H	
	=)0100	(二进制)
	=4	(十进制)

(实际读取参数的个数) = 5 (4+1)

D	e		g																
(5)	(6)	(7)	第一数据项				(11)	第二数据项				第五数据项							
R	0	0	,	0	0	6	4	0	0	6	E								
52H	30H	30H	2CH	30H	30H	36H	34H	30H	30H	36H	45H		0	0	0	0			
													30H	30H	30H	30H			

e: 这位表明这是一个应答代码<0(30H) 0 (30H) 正确的应答>。(详见应答代码表)

g: 这位表明这应答的读命令返回的数据项。返回数据项个数与上位机的 f00位有关。

3. 不正确的读命令应答

D	e	
(5)	(6)	(7)
R	0	7
52H	30H	37H

- d: 这位表明这是一个应答的读命令。
- e: 表明这是一个应答代码<0(30H)7(30H)是数据格式错误的应答>。(详见应答代码表)

4. 写命令

d	e				f	g				
(5)	(6)	(7)	(8)	(9)	(10)	(11)				
W	0	4	0	0	0	,	0	0	2	8
57H	30H	34H	30H	30H	30H	2CH	30H	30H	32H	38H

- d: 这位表明这是一个写命令。
- e: 这位表明这个读命令是写的控制输出 1 的比例带的参数。
- f: 这位表明这读命令要读多少个参数。
- g 通讯数据[11]这位的数据量决定于这位的数据，既这位的数据长度不定]。数据总是以", "(2CH), 数据项与数据项之间不需要任何分割符。数据的长度主要取决于第10的方式。每一个数据项由 16 位二进制代码组成(1 个字)，每 4 位被分成一个数据单元，同时每个数据单元又被转换成 ASCII 数据。

具体含义如下：

这位表明这个写命令是写的控制输出 1 的比例带参数	=0400H (十六进制)
	=0000 0100 0000 0000 (二进制)
这位表明这读命令要写多少个参数。	=0H
	=0000 (二进制)
	=0 (十进制)
(实际写参数的个数)	=1 (0+1)
被写入的具体数据	=0028H (十六进制)
	=0000 0000 0010 1000 (二进制)
	=40 (十进制)

5. 正确的写命令应答

d	e	
(5)	(6)	(7)
W	0	0
57H	30H	30H

- d: 这位表明这是一个写应答的命令。
- e: 表明这是一个应答代码<0(30H)0(30H)是一个写命令的正确应答>。(详见 4-4 应答代码表)

6. 不正确的写命令应答举例

d	e	
(5)	(6)	(7)
W	0	9
57H	30H	30H

- d: 这位表明这是一个写应答的命令。
- e: 表明这是一个应答代码<0 (30H) 9 (39H) 是一个不正确写命令的应答>。(详见应答代码表)

4-4 应答代码表

应答代码		代码类型	代码类型的详细说明
二进制码	ASCII		
0000 0000	"0", "0": 30H, 30H	正确的应答	读、写命令的正确应答
0000 0001	"0", "1": 30H, 31H	硬件错误	当发生硬件错误例如帧溢出或奇偶校验错误被检测到时。
0000 0111	"0", "7": 30H, 37H	数据格式错误	数据格式部分和设计的固定格式不符。
0000 1000	"0", "8": 30H, 38H	命令或数据的数量错误	命令代码或数据的数量和设计的要求不同。
0000 1001	"0", "9": 30H, 39H	数据错误	被写入的数据不是有效的可被设定的范围
0000 1010	"0", "A": 30H, 41H	执行命令错误	执行命令的接收是在一定条件下的 (例如 AT), 否则将不被执行
0000 1011	"0", "B": 30H, 42H	写模式错误	一些类型的数据在某一时刻将不能及时被写入。这种数据写入应在这种数据允许写入的时刻写入。
0000 1100	"0", "C": 30H, 43H	其他或操作错误	写命令中的特殊数据或操作, 不能被加入或接收。

小数点的表示方法：将小数点去掉后，直接连同小数点后的数转换成十六进制数。小数点的位置与使用的量程有关。这四位十六进制代码（16 位二进制码）的使用范围（-32768～32767）。

例： 十进制 十六进制

20.0% → 200 → 008C

100.0℃ → 1000 → 03E8

-40.00℃ → -4000 → F060

4-5 通讯命令表

命令代码 (十六进制)	参数	参数的详细说明		读/写
0040		序列代码 1	这 4 个序列代码不能同时被读取。	读
0041		序列代码 2		读
0042		序列代码 3		读
0043		序列代码 4		读

这些命令由 16 位二进制组成，被分成高 8 位和低 8 位两个单元。不用的地址用"0"填充。

例：SR91 命令 高 4 位 低 4 位 高 4 位 低 4 位

 0040 "S" "R" 53H 52H

 0041 "9" "1" 39H 31H

0042	00H	00H
0043	00H	00H

命令代码 (十六进制)	参数	参数的详细说明	读/写
0100	PV_W	测量值	读
0101	SV_W	当前执行的设定值	读
0102	OUT1W	控制输出 1 的值	读
0103	OUT2W	控制输出 2 的值（无输出时=0000H）	读
0104	EXE_FLG	执行标志（不执行时=0）	读
0105	EV_FLG	事件输出标志（无事件输出时=0000）	读
0109	HB_W	加热器断线报警值（无报警时=0000H）	读
010A	HL_W	欠流报警值（无报警时=0000H）	读

EXE_FLG 和 EV_FLG 的详细说明如下：

	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
EXE_FLG	0	0	0	0	0	0	0	COM	0	0	0	0	0	0	MAN	AT
EV_FLG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	EV2	EV1

上限边包括：PV_S0, CJ_S0, b---, REM_S0, HB_S0=7FFFH

下限边包括：PV_S0, CJ_S0, b---, REM_S0, HB_S0=8000H

无效数据 HB, HL=7FFEh

0182	OUT1_W	在手动方式下设置输出 1 的值	写
0183	OUT2_W	在手动方式下设置输出 2 的值	写
0184	AT	自整定 0=不执行，1=执行	写
0185	MAN	手动 0=自动， 1=手动	写

018C	COM	通讯 0=本机， 1=通讯	写
------	-----	--	---

0300	SV1	设定 SV1 的值	读/写
------	-----	-----------	-----

030A	SV_L	SV 下限值	读/写
030B	SV_H	SV 上限值	读/写

0400	PB1	控制输出 1 的比例带	读/写
0401	IT1	控制输出 1 的积分时间	读/写
0402	DT1	控制输出 1 的微分时间	读/写
0403	MR1	人工补偿	读/写
0404	DF1	回差	读/写
0405	011_L	控制输出 1 下限	读/写
0406	011_H	控制输出 1 上限	读/写
0407	SF1	控制输出 1 抗超调系数	读/写

0460	PB21	控制输出 2 的比例带	读/写
0461	IT21	控制输出 2 的积分时间	读/写

0462	DT21	控制输出 2 的微分时间	读/写
0463	MR21	人工补偿	读/写
0464	DF21	回差	读/写
0465	012_L	控制输出 2 下限	读/写
0466	012_H	控制输出 2 上限	读/写
0467	SF21	控制输出 2 抗超调系数	读/写

0500	EV1_MD	事件报警 1 的模式 (见说明书事件报警)	读/写
0501	EV1_SP	事件报警 1 的设定值 (见说明书事件报警)	读/写
0502	EV1_DF	事件报警 1 的回差	读/写
0503	EV1_STB	事件报警 1 的抑制和非抑制状态 OFF: 无抑制。 1: 初次上电, 报警抑制。 2: 初次上电脱机状态时, 报警抑制。 3: 初次上电脱机状态或改变设定值时, 报警抑制。 4: 脱机状态时抑制, 运行状态时无抑制。	读/写

0508	EV2_MD	事件报警 2 的模式 (见说明书事件报警)	读/写
0509	EV2_SP	事件报警 2 的设定值 (见说明书事件报警)	读/写
050A	EV2_DF	事件报警 2 的回差	读/写
050B	EV2_STB	事件报警 2 的抑制和非抑制状态 OFF: 无抑制。 1: 初次上电, 报警抑制。 2: 初次上电脱机状态时, 报警抑制。 3: 初次上电脱机状态或改变设定值时, 报警抑制。 4: 脱机状态时抑制, 运行状态时无抑制。	读/写

0590	HBS	加热器断线报警设置	读/写
0591	HBL	加热器环路报警设置	读/写
0592	HB_MD	加热器断线报警锁定模式设置 0=非锁定, 1=锁定	读/写
0593	保留		
0594	HB_STB	加热器断线设置 0=OFF, 1=ON	读/写

05A0	A01_MD	模拟变送模式 0=测量值, 1=设定值, 2=偏差值, 3=控制输出 1 的值, 4=控制输出 2 的值	读/写
05A1	A01_L	模拟变送下限	读/写
05A2	A01_H	模拟变送上限	读/写

05B0	COM_MEM	通讯的存贮模式 0=EED, 1=REM, 2=r_E	读/写
------	---------	-----------------------------	-----

0600	ACTMD	输出的特性 0=反作用, 1=整作用	读/写
0601	01_CYC	SV1 的比例周期	读/写
0604	02_CYC	SV2 的比例周期	读/写
0605	SOFTD1	缓起动时间	读/写

0611	KLOCK	键盘锁 0=无锁定 1=除了显示、用户可设定项及通讯以外锁定 2=除了 SV、通讯以外锁定， 3=除了通讯锁定	读/写
------	-------	--	-----

0701	PV_B	PV 值偏移	读/写
0702	PV_F	PV 值滤波	读/写

0704	UNIT	单位 0=℃ 1=°F	读/写
0705	RANGE	测量范围（见测量范围代码表）	读/写
0706	保留		
0707	DP	小数点位置 0=无，1=0.1, 2=0.01, 3=0.001	只有直流输入时作用
0708	SC_L	测量范围下限值 -1999~9989	读/写
0709	SC_H	测量范围上限值 -1989~9999	读/写

4-6 ASCII 代码

	b7b6b5	000	001	010	011	100	101	110	111
b4b3b1		<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>
0000	<u>0</u>	NUL	TC7 (DLE)	SP	0	@	P	`	p
0001	<u>1</u>	TC1 (SOH)	DC1	!	1	A	Q	a	q
0010	<u>2</u>	TC2 (STX)	DC2	”	2	B	R	b	r
0011	<u>3</u>	TC3 (ETX)	DC3	#	3	C	S	c	s
0100	<u>4</u>	TC4 (EOT)	DC4	\$	4	D	T	d	t
0101	<u>5</u>	TC5 (ENQ)	TC8 (NAK)	%	5	E	U	e	u
0110	<u>6</u>	TC6 (ACK)	TC9 (SYN)	&	6	F	V	f	v
0111	<u>7</u>	BEL	TC10 (ETB)	'	7	G	W	g	w
1000	<u>8</u>	FE0 (BS)	CAN	(	8	H	X	h	x
1001	<u>9</u>	FE1 (HT)	EM	)	9	I	Y	i	y
1010	<u>A</u>	FE2 (LF)	SUB	*	:	J	Z	j	z
1011	<u>B</u>	FE3 (VT)	ESC	+	;	K	[	k	{
1100	<u>C</u>	FE4 (FF)	IS4 (FS)	,	<	L	\	l	
1101	<u>D</u>	FE5 (CR)	IS3 (GS)	-	=	M	]	m	}
1110	<u>E</u>	SO	IS2 (RS)	.	>	N	^	n	~
1111	<u>F</u>	SI	IS1 (US)	/	?	O	_	o	DEL

4-7 事件报警类型

报警代码	事件类型	类型号码
OFF	无	0
Hd	上限偏差值报警	1
Ld	下限偏差值报警	2
Od	上或下限偏差值外报警	3
Cd	上或下限偏差值内报警	4
HA	上限绝对值报警	5

LA	下限绝对值报警	6
So	超量程报警	7
Hb	加热器或环路报警	8

4-8 BASICA 程序例

4.8.1 设置起始符, 文件结束, 全文件结束的三个控制符

```

10 STX$ = "@": ETX$ = ":": CR$=CHR$(13)
15 REM 初使化 PC 机口和设数据格式(必需和仪表的设置相同)
20 REM 使用 PC COM1 口, 设置 1200 波特, 偶效验, 7 位数据, 1 停止位, 屏蔽握手信号。
40 BPS$ = "1200": ADR$="01": REM 设置波特率和仪表通讯地址
50 OPEN "COM1:" + BPS$ + ",E,7,1,CD,RS,CS,DS" AS #1

70 CMD$="0100": REM READ PV
80 BC$ = ADR$ + CMD$ + ETX$
90 GOSUB 420
100 PRINT BCC$
110 STOP
420 BCC = ASC(LEFT$(BC$, 1)): REM 发/接的 BCC 块效验程序
430 L = LEN(BC$)
440 FOR N = 2 TO L
450 BCC = BCC XOR ASC(MID$(BC$, N, 1))
460 NEXT N
470 BCC$ = HEX$(BCC)
480 IF LEN(BCC$) = 1 THEN BCC$ = "0" + BCC$: REM 如效验结果为单字节, 需加"0"
490 RETURN

```

原程序

```

10 CLS
20 REM SD20 COM1 BPS=1200 ADR=00 FORMAT =7E1
30 STX$ = "@": ETX$ = ":":
40 BPS$ = "1200"
50 OPEN "COM1:" + BPS$ + ",E,7,1,CD,RS,CS,DS" FOR RANDOM AS #1: REM INITIAL COM(1)
60 'LINE INPUT"NEW ORDER = ";CMD$: REM PV DISPLAY
70 ADR$ = "01": CMD$="1R01000": REM SD20 ADDR=00
80 BC$ = ADR$ + CMD$ + ETX$
90 GOSUB 310
100 TXD$ = STX$ + BC$ + BCC$ + CHR$(13)
110 PRINT #1, TXD$;
120 PRINT "SENDING DATA = "; TXD$
130 T3 = VAL(MID$(TIME$, 7, 2)): 'WAITING TIME 2s
140 IF EOF(1) = 0 THEN 170
150 T4 = VAL(MID$(TIME$, 7, 2))
160 IF ABS(T4 - T3) < 4 THEN 140 ELSE PRINT "OVER 2S AND COMMUNICATION ERROR!":
GOTO 60
170 D$ = "": REM DATA RECEIVE
180 A$ = INPUT$(1, #1)
190 D$ = D$ + A$

```

```

200 IF A$ = CHR$(13) THEN GOTO 220
210 GOTO 180
220 RBCC$ = LEFT$(RIGHT$(D$, 3), 2): REM RECEIVE DATA BCC CHECK
230 LEC = LEN(D$)
240 BC$ = MID$(D$, 2, LEC - 4)
250 GOSUB 310
260 IF RBCC$ = BCC$ THEN 281
270 PRINT "BCC$="; BCC$: PRINT "BCC ERROR !": PRINT D$
280 END
281 LEC = LEN(D$): F$ = "": K = 1
282 FOR P = 6 TO LEC
283 N$ = MID$(D$, P, 1)
284 IF N$ = ", " THEN U$(K) = F$: K = K + 1: F$ = "": GOTO 287
285 IF N$ = ETX$ THEN U$(K) = F$: N = K: GOTO 288
286 F$ = F$ + N$
287 NEXT
288 PRINT "RECEIVING DATA="; D$
289 FOR N = 1 TO K
290 PRINT U$(N),
291 NEXT
292 PRINT
300 GOTO 60
280 END
281 LEC = LEN(D$): F$ = "": K = 1
282 FOR P = 6 TO LEC
283 N$ = MID$(D$, P, 1)
284 IF N$ = ", " THEN U$(K) = F$: K = K + 1: F$ = "": GOTO 287
285 IF N$ = ETX$ THEN U$(K) = F$: N = K: GOTO 288
286 F$ = F$ + N$
287 NEXT
288 PRINT "RECEIVING DATA="; D$
289 FOR N = 1 TO K
290 PRINT U$(N),
291 NEXT
292 PRINT
300 GOTO 60
310 BCC = ASC(LEFT$(BC$, 1)): REM BCC CHECK!
320 L = LEN(BC$)
330 FOR N = 2 TO L
340 BCC = BCC XOR ASC(MID$(BC$, N, 1))
350 NEXT N
360 BCC$ = HEX$(BCC)
370 IF LEN(BCC$) = 1 THEN BCC$ = "0" + BCC$
380 RETURN

```

4.8.2 仪表的通讯设置

1-34 窗口

通讯/机内方式选择	
COMM	COMM
Loc	LOCAL

LOC:机内方式

◎此时, 面板通讯 COM 指示灯灭。

◎仅能由上位机控制命令, 转成通讯方式(COM)。

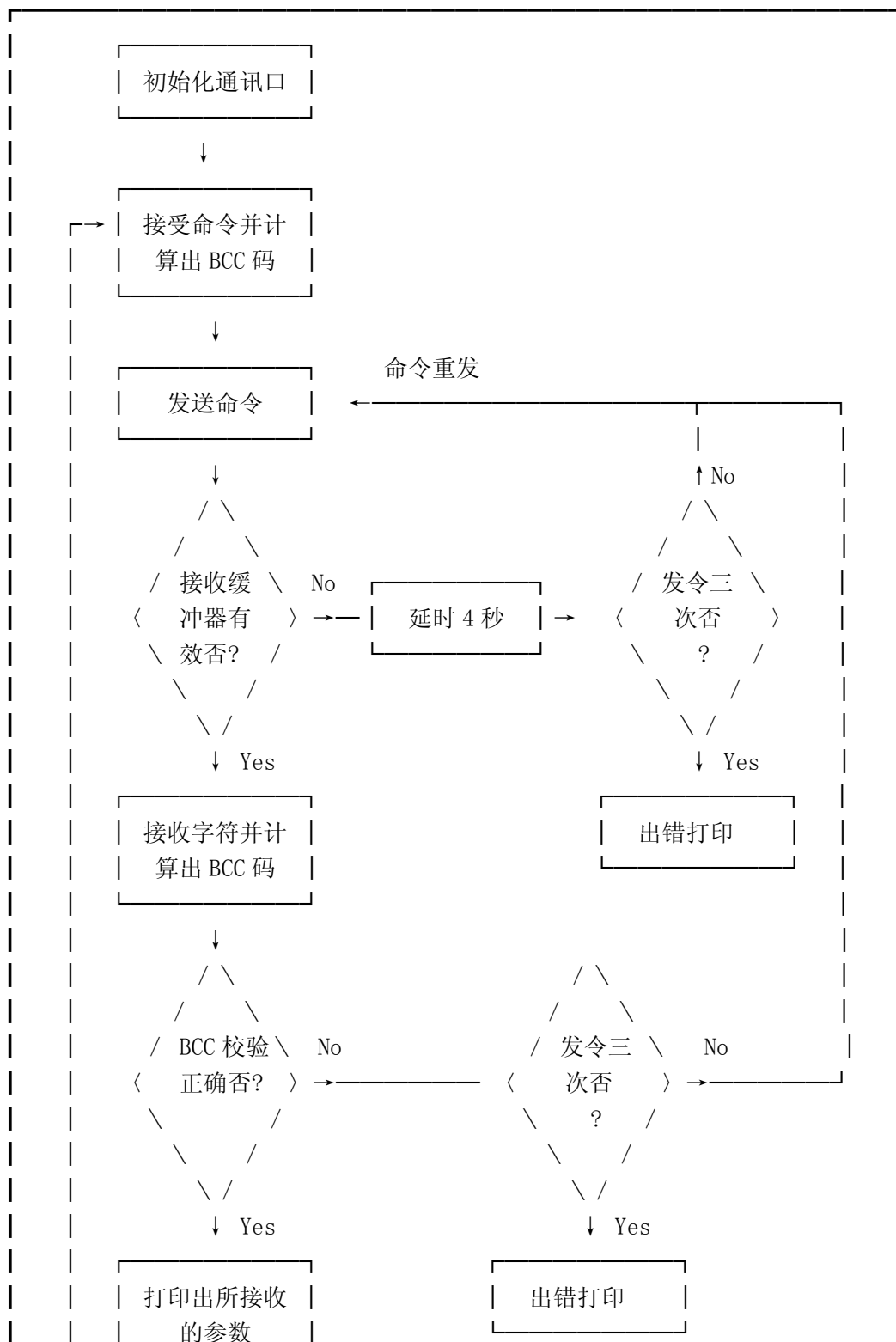
仅能完成上位机的读命令. 可由键设定内部参数。

COM: 上位机通讯方式

◎此时, 面板通讯 COM 指示灯亮

◎仅能由面板键设定或上位机控制命令, 转成 LOC 机内方式。

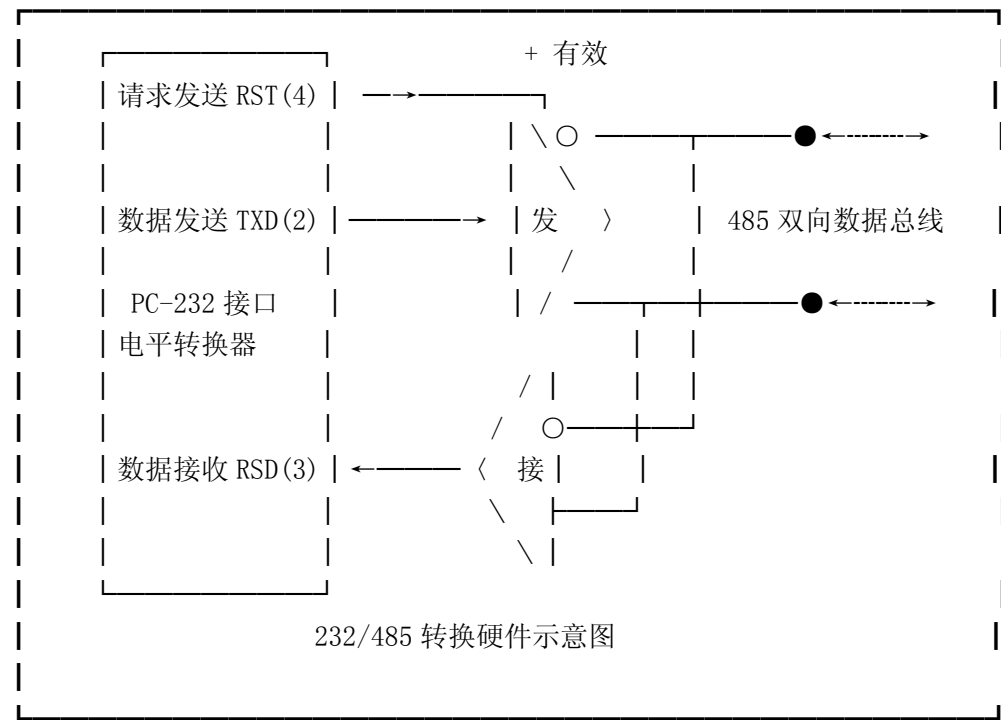
4.8.3 流程图





4. 8. 4 RS485 通讯接口和 BASIC 程序方法

1. RS485 双向数据总线转换硬件示意图



RS485 接口要求在发送数据完成后, 立即关闭发送, 否则无法接收其它设备的通讯。而存在着双向数据总线转换冲突和发送数据被自己接收的问题, 在上位机的通讯软件的设计中, 可采用两种方法:① UART 的发送寄存器空的位测试命令 ② 完整的接收到自发的数据, 来确认发送数据完成, 以便及时地关闭发送。

上位机的 RS232/485 转换器通常是利用 232 口的 RST 请求发送信号的位置位/位复位信号, 作为发送数据总线的转换控制。在 BASIC 程序 OPEN "COM 1, 1200, E, 7, 1, CD, RS, CS, DS" AS #1 命令后, 初始化 PC 机的通讯口, RST 信号置零, 使发送驱动器变成高阻输出。发送数据时, "OUT (&H3FC), &H0B" 的命令, 使经 UART 8250 输出的 RST 信号置高, 令发送驱动器变成低阻输出; 发送数据完成后, 输出 "OUT (&H3FC), &H09" 命令, 又将 RST 信号置零, 发送驱动器恢复成高阻输出。

下位仪表, 可在仪表的 RS485 延时时间窗口, 根据通讯速度, 调整发送数据总线的转换时间。

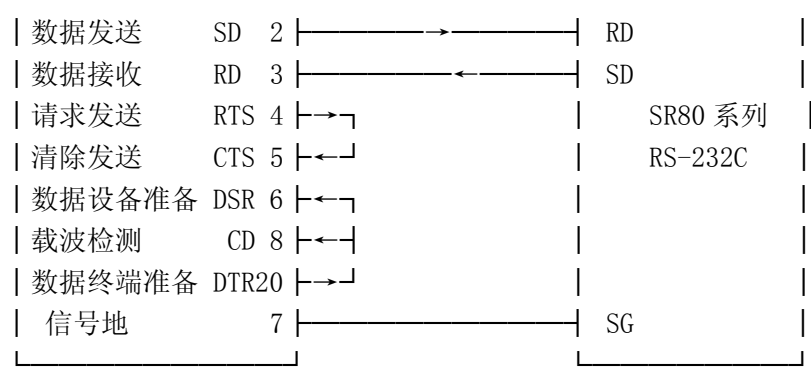
建议: 选用 RS485 接口的仪表时, 可采用研华 5020 型 RS232/RS485 智能通讯转换模块, 编程时无需考虑总线切换的问题。

5. 附录:

A. 通讯串口接线方法

☆RS-232C 通讯口接线示意图

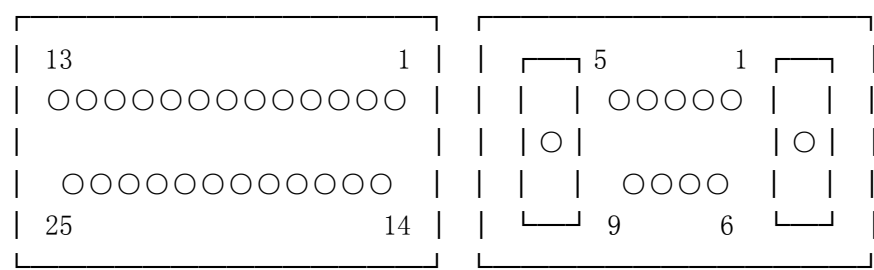




P C 机 25 针 RS-232C 仪表 9 针 RS232 (端子号见使用说明书)

☆PC 机 RS-232C 串口 25 针与 9 针接线对照表:

9PIN	1	2	3	4	5	6	7	8	9
25PIN	8	3	2	20	7	6	4	5	22



B. RS232 通讯口的技术数据

- 1.信号电平: EIA RS-232C 电平(±12V)
- 2.通讯方式: RS232C 3 线半双工
- 3.同步系统: 起始位-停止位, 异步通讯
- 4.通讯距离: RS232C 15 米
- 5.通讯速度: 1200, 2400, 4800, 9600, 19200 波特率
- 6.数据格式: 8 种.
常用格式:数据 7 位, 一个偶校验位, 一个停止位
- 7.数据块校验: 数据异或(双字节)
- 8.通讯码: ASCII
- 9.握手信号: 未使用
- 10.连接台数: RS-232C 1 台

C. RS422/RS485 通讯接口的技术数据

- 1.信号电平: EIA RS422A/485 电平 5V 差动
- 2.通讯方式: RS422A 4 线半双工(多路)/RS485 2 线半双工(多路)
- 3.同步系统: 起始-停止位同位, 异步通讯
- 4.通讯距离: 1200 米
- 5.通讯速度: 1200, 2400, 4800, 9600, 19200 波特率
- 6.数据格式: 8 种.
常用格式:数据 7 位, 一个偶校验位, 一个停止位

- 7.数据块校验： 异或(双字节)
- 8.通讯码： ASCII
- 9.握手信号： 未使用
- 10.连接台数： RS-422/485 32 台 1.5 公里(配先锋 RS232/422 接口转换器)