

单片机的 MODEM 通讯

摘要 探讨单片机之间或单片机与 PC 之间采用 MODEM 的实现方法 以及通讯参数的设定、数据的接收发送和部分 AT 命令的介绍,并给出演示程序。

关键词 MODEM 通讯 AT 命令 单片机 MODEM

我们经常能见到关于 PC 的 MODEM 通讯的文章,但关于单片机 MODEM 通讯的文章却不多见。现在将我个人单片机 MODEM 通讯的实践经历写出来供大家参考。

要写单片机的 MODEM 通讯必须要有两个背景知识,一个是 AT 命令集,另一个是通用异步接收发送器 (UART)。

1.1 AT 命令集

下面介绍我通讯程序例子中涉及到的 AT 命令。

Dn: 拨号命令。该命令使 MODEM 立即进入摘机状态,并拨出跟在后面的号码。D 命令是基本的拨号命令,它受到其它命令的修饰可构成 MODEM 何时拨号以及如何拨号等操作。

T: 音频拨号。例如, ATDT88888888, 其中 88888888 为电话号码。

P: 脉冲拨号。例如, ATDP88888888, 其中 88888888 为电话号码。

,: 标准暂停。我们常常碰到拨打外线电话时需要暂停一下,等听到二次拨号音(外线)之后才能再拨后续的号码。缺省时暂停时间为 2s(秒),它由 S8 寄存器指定。

Sn: 表示 MODEM 内部的寄存器。

S0: 自动应答。如果要求 MODEM 具有自动应答特性,则应该预先将 MODEM 的 S0 寄存器设置为非 0。

S8: 逗号拨号修饰符的暂停时间。该寄存器决定了当 MODEM 在拨号中遇到逗号(,)时应该暂停的时间。

2 通用异步接收发送器 UART

深入理解 UART 内部结构以及内部寄存器各位的含义,详细了解数据发送和接收的过程,有助于编写出高效、稳定的程序。现以 GM16C550 为例介绍编写基本通讯程序需要知道的寄存器。实际的 ADDRESS 由具体接线决定。表 1 为 GM16C550 寄存器的介绍。

表 1 GM16C550 寄存器

A2	A1	A0	ADDRESS	W/R	寄存器
0	0	0	FFF8	W	接收缓冲寄存器 (RHR)
				R	发送保持寄存器 (THR)
0	0	1	FFF9	W	中断允许寄存器 (IER)
0	1	0	FFFA	W	FIFO 控制寄存器 (FCR)
0	1	0	FFFA	R	中断状态寄存器 (ISR)
0	1	1	FFFB	W	线路控制寄存器 (LCR)
1	0	0	FFFC	W	MODEM 控制寄存器 (MCR)
1	0	1	FFFD	R	线路状态寄存器 (LSR)
1	1	0	FFFE	R	MODEM 状态寄存器 (MSR)
1	1	1	FFFF	W/R	临时数据寄存器 (SPR)

(1) 波特率除数锁存器 (LSB、MSB)

在通讯之前要进行一些参数初始化，波特率是首先应该考虑的一项。该寄存器是一个 16 位的寄存器，分为低 8 位 (LSB) 和高 8 位 (MSB) 寄存器。

当 LCR.7=1，且 A2A1A0=000/001 时，单片机访问的是波特率除数锁存器 LSB/MSB。GM16C550 推荐的工作频率是 1.8432MHz。这个频率除以 16 就是波特率的时钟频率，用于控制发送和接收数据的速度。下面给出波特率除数锁存器值的计算公式：

$$\text{波特率除数锁存器值} = \text{工作频率} / (16 \times \text{期望波特率}) = 1843200 / (16 \times \text{期望波特率})$$

表 2 给出了常用波特率与波特率除数锁存器值。

表 2 波特率除数锁存器

波特率 /baud	锁存器 (HEX)	MSB	LSB	波特率 /baud	锁存器 (HEX)	MSB	LSB
300	180	01	80	600	03	00	0C
1200	60	00	60	2400	30	00	30
4800	18	00	18	9600	0C	00	0C
19.2k	06	00	06	38.4k	03	00	03
57.6k	02	00	02	11.5k	00	00	01

MOV DPTR,#LCR ;除数锁定允许

MOV A,#80H

MOVX @DPTR, A

```
MOV DPTR, #LSB ; 波特率为 9600baud
MOV A, #0CH
MOVX @DPTR, A
INC DPTR
CLR A
MOVX @DPTR, A
```

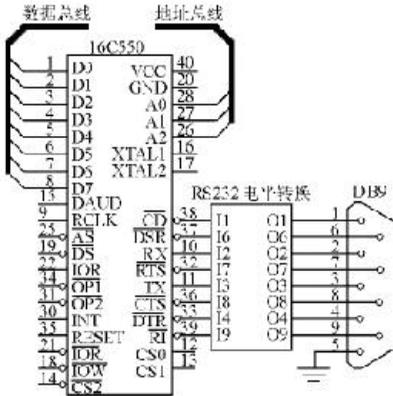


图 1 GM16C550 与 RS232 接线图

- (2) 接收缓冲寄存器和发送保持寄存器(transmit and receive holding register)
 - 当 LCR.7=0, 且 A2A1A0=000 时, 读操作单片机访问接收缓冲寄存器 (RHR), 写操作单片机访问发送保持寄存器 (THR).
- (3) 中断允许寄存器 (interrupt enable register)
 - 当 LCR.7=0, 且 A2A1A0=001 时, 单片机访问中断允许寄存器 (IER).
 - IER.0=1, 允许接收器数据就绪中断。
 - IER.1=1, 允许发送保持寄存器为空时中断。即当从发送保持寄存器把一个字节移到移位寄存器时, 产生一个中断, 使发送保持寄存器能够接收下一个字节。
 - IER.2=1, 表示允许接收有错信息或间断条件中断。
 - IER.3=1, MODEM 状态变化中断。
 - IER.4~7, 没有使用, 设置为零。

```
MOV DPTR, #IER
MOV A, #01H
```

MOVX @ DPTR ,A

(4) FIFO 控制寄存器 (FIFO control register)

数据发送和接收模式的选择。GM16C550 提供了两种模式：FIFO 和 DMA。其中 DMA 又有两种模式 DMA 的模式 0、DMA 的模式 1 可供选择。我的举例采用默认的 DMA 的模式 0。感兴趣的朋友可试一试其它模式，这里不再说明。

(5) 中断状态寄存器 (interrupt status register)

前面介绍了几种中断，它们在各自的条件下产生中断，UART 都会输出一个高电平的中断请求信号，触发同一个中断请求。为了具体判断是哪一种中断，还应该检测 ISR，如表 3 所列。

表 3 中断状态寄存器

中断 优先级	(ISR)				中 断 源
	D3	D2	D1	D0	
1	0	1	1	0	接收奇偶校验错、超越错、帧格式错、间断条件
2	0	1	0	0	接收寄存器就绪
2	1	1	0	0	接收数据超时
3	0	0	1	0	发送保持器准备好
4	0	0	0	0	MODEM 状态变化

ISR.0=1,表示没有中断产生。

ISR.4~5 没有使用。

ISR.6~7,当采用 FIFO 的接收和发送模式时，这两位都设置为 1；反之，都设置为 0。

(6) 线路控制寄存器 (line control register)

LCR.0~1,表示发送和接收时的字节长度，如表 4 所列。

LCR.2,这一位与 LCR.0~1 共同定义了停止位的长度，如表 5 所列。

表 4 线路控制寄存器 LCR.0~1

LCR.0	LCR.1	发送和接收时的字节长度/bit
0	0	5
0	1	6
1	0	7
1	1	8

表 5 线路控制寄存器 LCR.2

LCR. 2	发送和接收时的字节长度/5a	停止位长度/5a
0	5 6 7 8	1
1	5	1.5
1	6 7 8	2

LCR. 3=1, 进行奇偶校验。

LCR. 4=0, 进行奇校验 ; LCR. 4=1, 进行偶校验。

LCR. 5=1, 奇偶校验位恒为 1 或 0。

表 6 为线路控制寄存器 LCR. 3-5。

表 6 线路控制寄存器 LCR. 3 ~ 5

D5D4D3	奇偶校验	D5D4D3	奇偶校验
000	无奇偶	111	奇偶位恒为 1
101	奇偶位恒为 1	011	偶校验
001	奇校验		

LCR. 6=1, 表示允许间断, 即允许发送器寄存器保持一个完整帧时间以上的空号状态。

LCR. 7, 用于区分访问除数锁存寄存器还是访问接收缓冲 / 发送保持和中断允许寄存器。

MOVDPTR, #LCR ; 通讯传输长度为 8 位, 停止位为 1,

; 偶校验

MOVA, #1BH

MOVX @DPTR, A

(7) MODEM 控制寄存器 (MODEM control register)

这是一个 MODEM 和外设接口的寄存器。

MCR. 0=1 时, 强制芯片引脚 DTR=0 ;

MCR. 0=0 时, 强制芯片引脚 DTR=1。

MCR. 1=1 时, 强制芯片引脚 RTS=0 ;

MCR. 1=0 时, 强制芯片引脚 RTS=1。

MCR. 2=1 时, 强制芯片引脚 OP1=0 ;

MCR. 2=0 时, 强制芯片引脚 OP1=1。

MCR. 3=1 时, 强制芯片引脚 OP2=0 ;

MCR. 2=0 时, 强制芯片引脚 OP2=1。

MCR.4=1 时, 循环返回模式, 可用于芯片自测。

其它位保留。

MOVDPTR, #MCR; 异步串口芯片的 DTR、RTS 引脚送出逻辑低电平

MOVA, #03H

MOVX @DPTR, A

(8) 线路状态寄存器 (line status register)

LSR.0: 当接收移位寄存器接收到的字节完全移到接收缓冲寄存器时, 该位置 1。如果读该寄存器, 那么这一位被清零。

LSR.1: 出现超越错时, 这一位被置 1。读该寄存器, 这一位被清零。

LSR.2: 出现奇偶校验错时, 这一位被置 1。读该寄存器, 这一位被清零。

LSR.3: 出现停止位不完整、丢失、空号时, 这一位被置 1。读该寄存器, 这一位被清零。

LSR.4: 当接收寄存器检测到空号状态已持续一个完整帧传输时间时, 这一位被置 1。读该寄存器, 这一位被清零。

LSR.5: 当发送的字节从发送保持寄存器移到发送移位寄存器时, 该位置 1。

LSR.6: 当发送保持寄存器和发送移位寄存器都没用字节时, 该位置 1。

LSR.7: 当奇偶校验错、帧格式错、空号错有一个出现时, 该位置 1。

Setdata

MOV DPTR, #LSR

MOVX A, @DPTR

JNB ACC.5, Setdata

MOV A, DataNumber; DataNumber 记录发送字符的个数

MOV DPTR, #AtCommand

AtCommand 定义 AT 命令参数的起始地址

MOVC A, @A+DPTR

MOV DPTR, #THR; 发送保持寄存器

MOVX @DPTR, A

INC DataNumber

MOV A, DataNumber

CJNE A, #0BH, Setdata; 发送 11 个字符

AtCommand :DB "A", "T", "S", "0", "=", "2", "S", "8", "=", "5",
ODH

(9) MODEM 状态寄存器 (MODEM status register)

MSR.0~3=1 时,表示自上一次单片机读 MSR 寄存器之后,分别反映 MODEM 控制逻辑的四个输入信号的状态发生了变化。

MSR.0=1 时,清除发送信号 (CTS) 已经发生了变化。

MSR.1=1 时,数据设备就绪信号 (DSR) 已经发生了变化。

MSR.2=1 时,振铃信号 (RI) 已经发生了变化。

MSR.3=1 时,载波信号 (DCD) 已经发生了变化。

MSR.4~7 四位分别反映 MODEM 控制逻辑的四个输入信号的当前状态。

MSR.4=1 时,清除发送信号 (CTS) 有效。

MSR.5=1 时,数据设备就绪信号 (DSR) 有效。

MSR.6=1 时,振铃信号 (RI) 有效。

MSR.7=1 时,载波信号 (DCD) 有效。

(10) 临时数据寄存器 (scratchpag register)

可以存储用户信息。

有了上面知识的准备后就可以轻松地写出单片机的 MODEM 通讯程序,现在可把零散的东西组织起来。

GM16C550 芯片初始化模块

Init_16C550:

MOV DPTR,#LCR ;除数锁定允许

MOV A,#80H

MOVX @DPTR, A

MOV DPTR,#LSB ;波特率为 9600baud

MOV A,#0CH

MOVX @DPTR, A

INC DPTR

CLR A

MOVX @DPTR, A

MOV DPTR,#LCR ;通讯传输长度为 8 位,停止位为 1,偶校验

```
MOV A,#1BH
MOVX @DPTR, A
MOV DPTR,#MCR ; 异步串口芯片的 DTR。RTS 引脚送出逻辑低电平
MOV A,#03H
MOVX @DPTR,A
MOV DPTR,#IER
MOV A,#01H
MOVX @DPTR,A
```

采用中断的方式接数据

Interrupt2:

```
PUSH ACC
PUSH DPH
PUSH DPL
PUSH PSW
MOV DPTR,#RHR ; 接受数据
MOVX A,@DPTR
.....
POP PSW
POP DPL
POP DPH
POP ACC
RETI
```

至此，完成了一个小型的单片机 MODEM 通讯系统。其中的检错寄存器、MODEM 状态寄存器等应用限于篇幅没有完全涉及到。有兴趣的朋友可以试一试，那么你就会对 MODEM 通讯有一个比较深刻的认识。