

# 第十章

---

## 串行通信接口

# 10.1 串行通信概述

## 10.1.1 串行通信数据传输方式

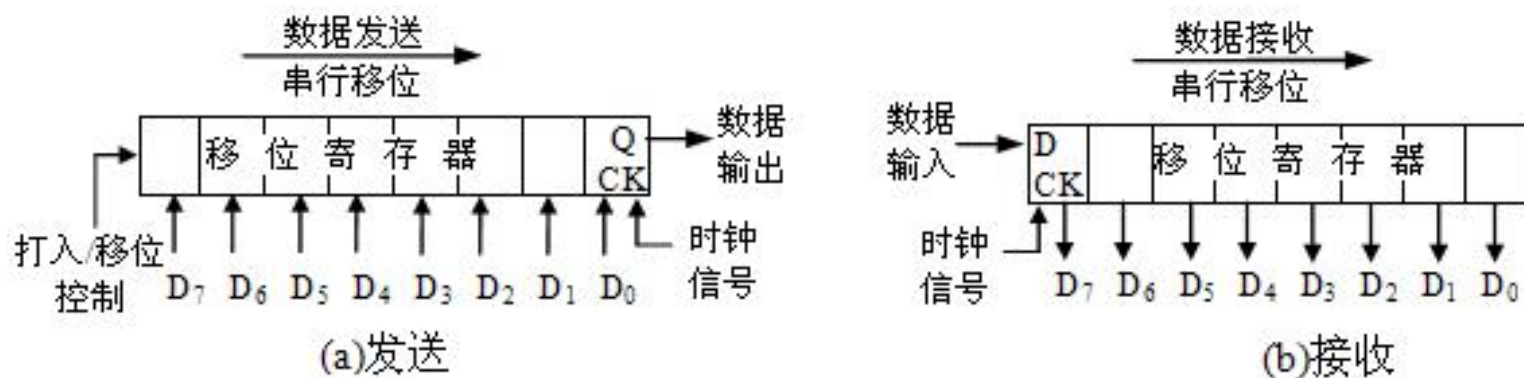


图 10-1 串行与并行数据的转换

# 10.1 串行通信概述

## 10.1.2 串行通信数据收发方式

### 1. 同步串行通信方式

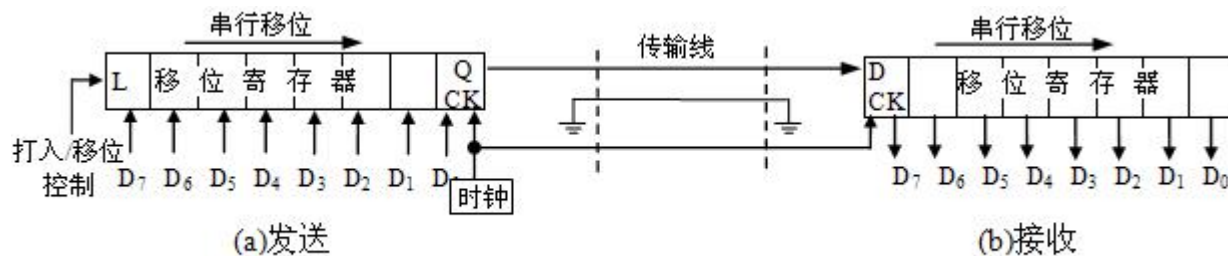


图 10-2 最简单的同步串行通信



图 10-3 同步串行通信

# 10.1 串行通信概述

## 10.1.2 串行通信数据收发方式

### 2. 异步串行通信方式

#### ① 字符格式。

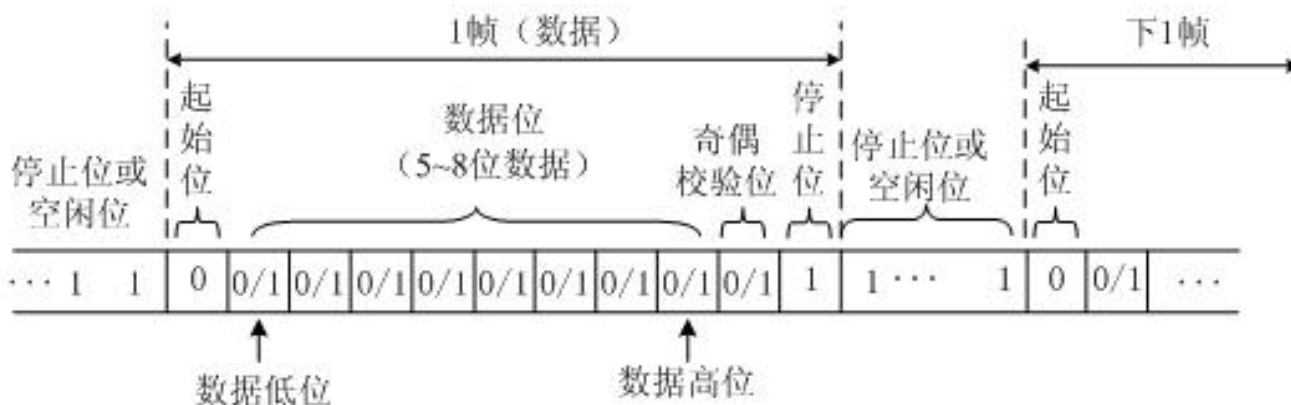


图 10-4 异步通信数据格式

# 10.1 串行通信概述

## 10.1.2 串行通信数据收发方式

### 2. 异步串行通信方式

- ② 数据传送速率。
- ③ 发送时钟与接收时钟。

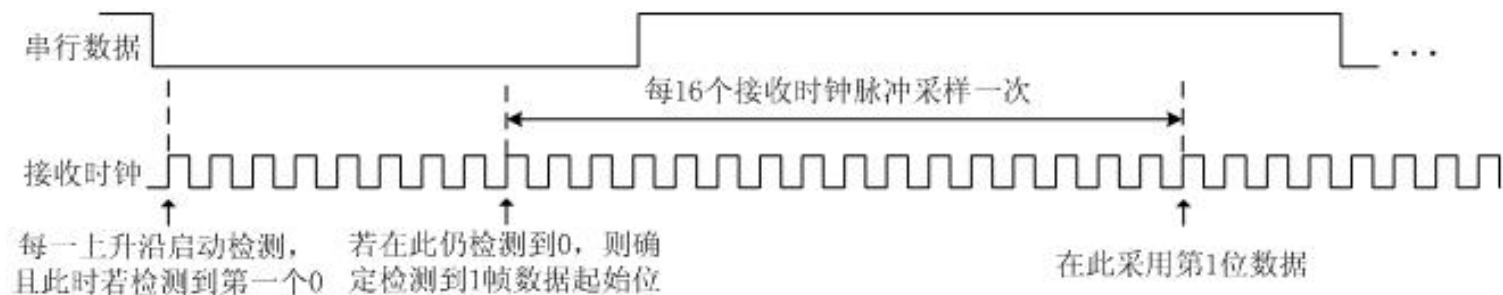


图 10-5 接收时钟采样串行数据说明

# 10.1 串行通信概述

## 10.1.2 串行通信数据收发方式

### 2. 异步串行通信方式

④ 校验方式。

⑤ 调制解调。



图 10-6 利用调制解调器的异步串行通信模型

# 10.1 串行通信概述

## 10.1.3 串行通信数据传送方式

### 1. 单工 (Simplex)

### 2. 半双工 (Half-Duplex)

### 3. 全双工 (Full-Duplex)

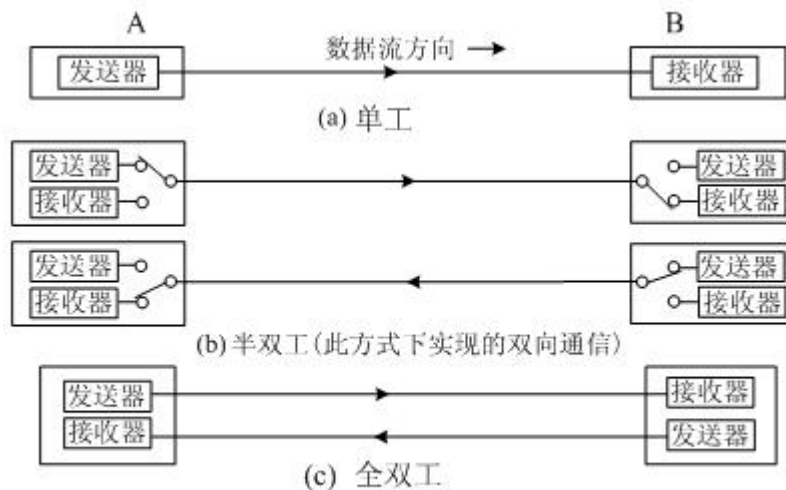


图 10-7 数据传送方式

# 10.2 串行通信接口标准RS-232C

## 10.2.1 RS-232C端口连接器机械特性和信号定义

表 10-1 RS-232C 标准的 DB-25 和 DB-9 接头主要引脚信号

引脚	信号名称	功能说明
1	PGND	保护接地，即机壳接地，起屏蔽保护作用。
2 (3)	TxD	发送数据。从此端将数据发送给 MODEN，或直接发送给对方的 RxD 端。
3 (2)	RxD	接收数据。通过此端接收来自 MODEN 或直接来自对方 TxD 端的数据。
4 (7)	RTS	请求发送。用来表示 DTE 请求 DCE 发送数据，控制 MODEM 发送与否。
5 (8)	CTS	清除发送或允许发送。用来表示 DCE 准备好接收 DTE 发来的数据。
6 (6)	DSR	数据通讯设备准备好。信号有效时表明 MODEM 处于可使用状态。
7 (5)	GND	所有信号的公共参考地
8 (1)	DCD	数据载波检出。用来表示 DCE 已接通通信通道，通知 DTE 准备接收数据。
20 (4)	DTR	数据终端准备就绪。信号有效时表明数据终端可以使用。
22 (9)	RI	振铃指示。当 MODEM 收到对方送来的振铃呼叫信号时，通知终端。



# 10.2 串行通信接口标准RS-232C

## 10.2.2 RS-232C的电气特性

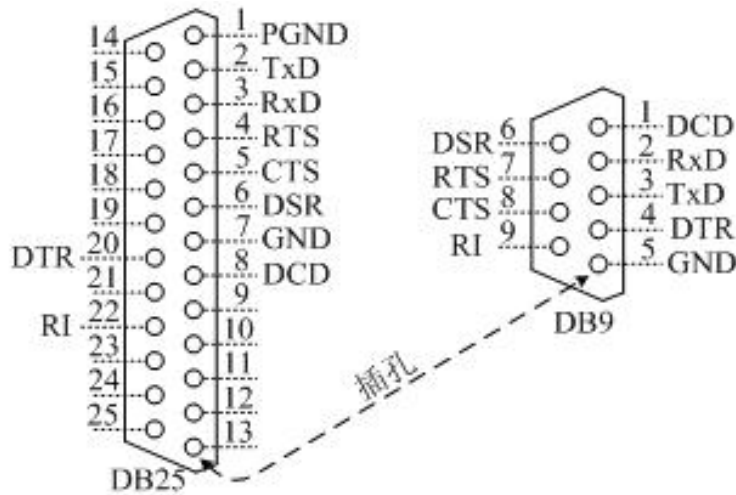


图 10-8 DB25 和 DB9 连接器外形和信号定义

# 10.2 串行通信接口标准RS-232C

## 10.2.3 RS-232C通信线路的连接

### 1. 近距离通信



图 10-9 最简单的三线制通信连接方式

# 10.2 串行通信接口标准RS-232C

## 10.2.3 RS-232C通信线路的连接

### 2. 远距离通信

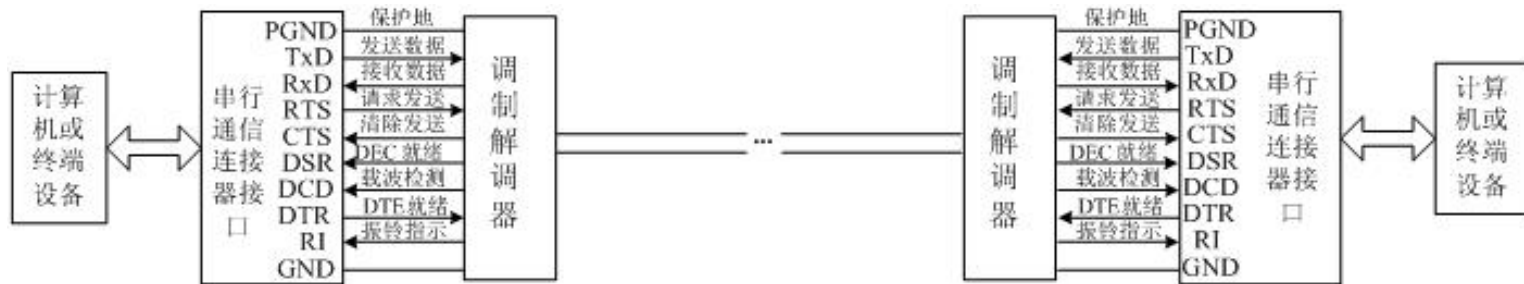


图 10-10 采用 MODEM 和电话线通信时的信号线连接方式

# 10.3 可编程串行通信接口芯片NS16550

## 10.3.1 16550的基本性能

## 10.3.2 16550芯片外部引脚信号

1. 与CPU接口的信号线

2. 串行通信接口信号线

3. 16550自身工作需要的信号线。

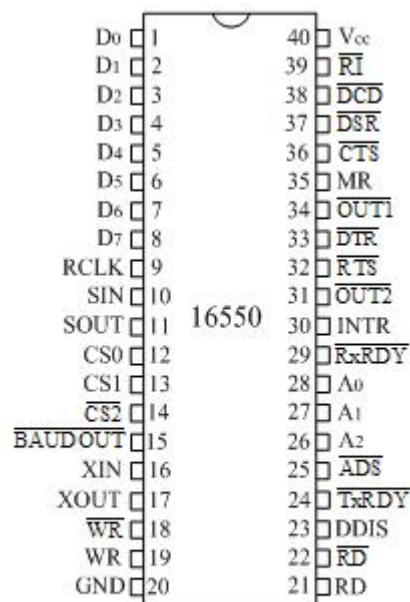


图 10-11 16550 引脚图

# 10.3 可编程串行通信接口芯片NS16550

## 10.3.3 16550的结构及内部寄存器

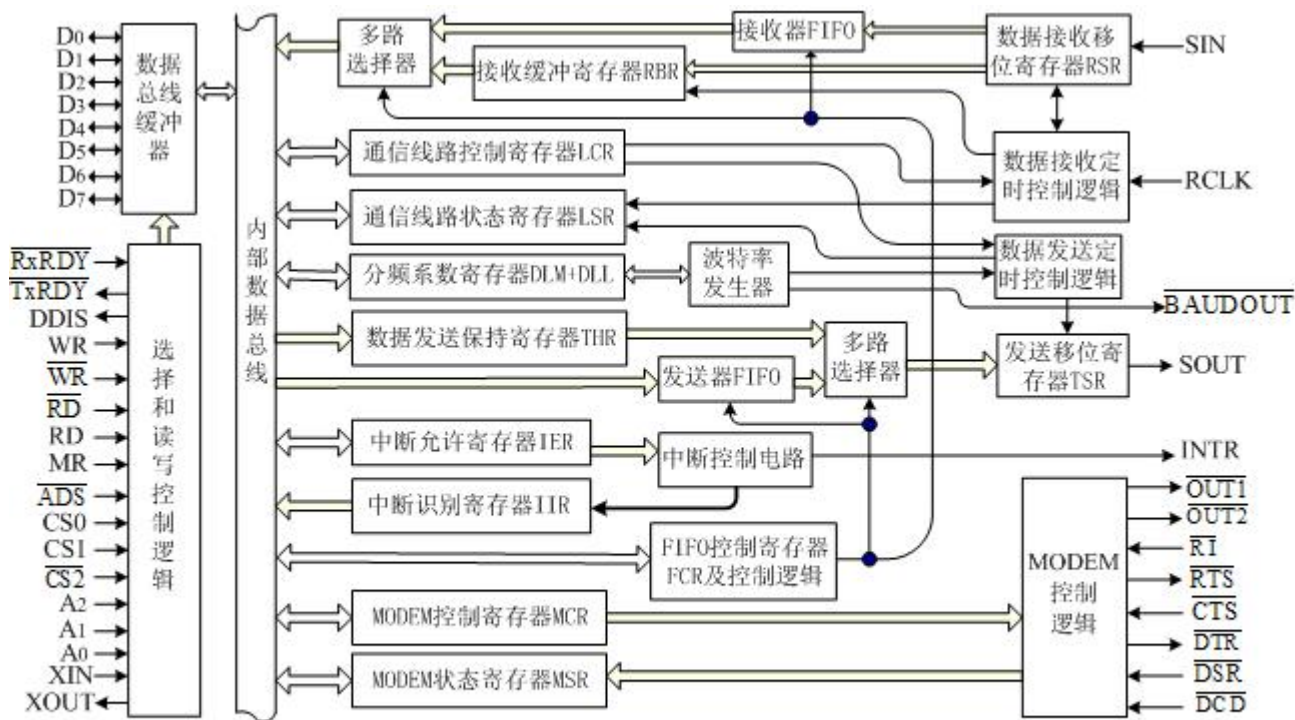


图 10-12 16550 内部结构框图



## 10.3 可编程串行通信接口芯片NS16550

---

### 10.3.3 16550的结构及内部寄存器

1. 数据总线缓冲器
2. 选择和读/写控制逻辑
3. 数据发送模块
4. 数据接收模块



## 10.3 可编程串行通信接口芯片NS16550

---

### 10.3.3 16550的结构及内部寄存器

#### 5. 波特率发生控制模块

$$\text{分频系数} = \text{基准时钟} \div \text{发送时钟} = 1843200 \div (\text{波特率} \times 16) \quad (10-1)$$

#### 6. 调制/解调控制电路

#### 7. 中断控制模块

## 10.3 可编程串行通信接口芯片 NS16550

### 10.3.4 16550内部寄存器数据格式

表 10-2 16550 内部寄存器及端口地址

DLAB	被访问的寄存器	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	COM1 端口	COM2 端口
0	数据发送保持寄存器 THR(写)	0	0	0	3F8H	2F8H
0	数据接收缓冲寄存器 RBR(读)	0	0	0	3F8H	2F8H
1	波特率除数寄存器低 8 位字节 DLL	0	0	0	3F8H	2F8H
0	中断允许寄存器 IER	0	0	1	3F9H	2F9H
1	波特率除数寄存器高 8 位字节 DLM	0	0	1	3F9H	2F9H
×	中断识别寄存器 IIR(读)	0	1	0	3FAH	2FAH
×	FIFO 控制寄存器 FCR(写)	0	1	0	3FAH	2FAH
×	通信线路控制寄存器 LCR	0	1	1	3FBH	2FBH
×	MODEM 控制寄存器 MCR	1	0	0	3FCH	2FCH
×	通信线路状态寄存器 LSR	1	0	1	3FDH	2FDH
×	MODEM 状态寄存器 MSR	1	1	0	3FEH	2FEH
×	暂存 Scratch	1	1	1	3FFH	2FFH



# 10.3 可编程串行通信接口芯片 NS16550

## 10.3.4 16550内部寄存器数据格式

### 1. 通信线路控制寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
DLAB	SB	SP	EPS	PEN	STB	WL1	WL0

图 10-13 通信线路控制寄存器 LCR 中各位的功能定义

### 2. FIFO控制寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
RT1	RT0	0	0	DMA	XMIT RST	REVC RST	EN

图 10-14 FCR 中各位的功能定义

# 10.3 可编程串行通信接口芯片 NS16550

## 10.3.4 16550内部寄存器数据格式

### 3. 通信线路状态寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
ER	TE	TH	BI	FE	PE	OE	DR

图 10-15 LSR 中各位的功能定义

【例 10-1】利用 LSR 的内容进行收发处理：

```
START : MOV  DX, 3FDH    ; 通信线路状态端口地址 3FDH
        IN   AL, DX      ; 读取 LSR 的内容
        TEST AL, 00011110B ; 测 D4D3D2D1, 检查有无数据接收错误及中止信号
        JNZ  ERR        ; 有错, 转出错处理
        TEST AL, 01h     ; 无错, 再查接收数据是否准备好
        JNZ  RECEIVE     ; DR=1, 16550 已经接收到数据, 转数据接收程序
        TEST AL, 20H     ; 再查发送保持寄存器是否变空 (THR2=1) 吗?
        JNZ  TRANS       ; 空, 转数据发送程序
        JMP  START       ; 不空, 循环等待

        ...

ERR :    ...           ; 出错处理
TRANS :  ...           ; 数据发送程序
RECEIVE: ...          ; 数据接收程序
```

# 10.3 可编程串行通信接口芯片 NS16550

## 10.3.4 16550内部寄存器数据格式

### 4. 中断允许寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	EM	EL	ET	ER

图 10-16 IER 中各位的功能定义

### 5. 中断识别寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
X	X	0	0	ID2	ID1	ID0	IP

图 10-17 IIR 中各位的功能定义

# 10.3 可编程串行通信接口芯片 NS16550

## 10.3.4 16550内部寄存器数据格式

### 6. MODEM控制寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	LB	OUT2	OUT1	RTS	DTR

图 10-18 MCR 中各位的功能定义

【例 10-2】若要使 MCR 的 DTR、RTS 有效（对应端口输出 0），OUT<sub>1</sub>、OUT<sub>2</sub> 以及 LB 无效（对应端口输出 1），则可用下列程序段实现：

```
MOV  DX, 3FCH      ; MCR 端口地址
MOV  AL, 00000011B ; MCR 的控制字
OUT  DX, AL
```

若要自发自收进行诊断，即使 LB 有效，程序段如下：

```
MOV  DX, 3FCH      ; MCR 端口地址
MOV  AL, 00010011B ; LB 位置“1”
OUT  DX, AL
```

# 10.3 可编程串行通信接口芯片 NS16550

## 10.3.4 16550内部寄存器数据格式

### 7. MODEM状态寄存器

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
DCD	RI	DSR	CTS	DDCD	TERI	DDSR	DCTS

图 10-19 MSR 中各位的功能定义

### 8. 数据收/发寄存器

### 9. 波特率除数寄存器

### 10. SCRATCHPAD寄存器



# 10.4 16550串行通信编程

---

## 10.4.1 16550的工作过程

### 1. 数据发送过程

### 2. 数据接收过程

# 10.4 16550串行通信编程

## 10.4.2 16550的接口电路

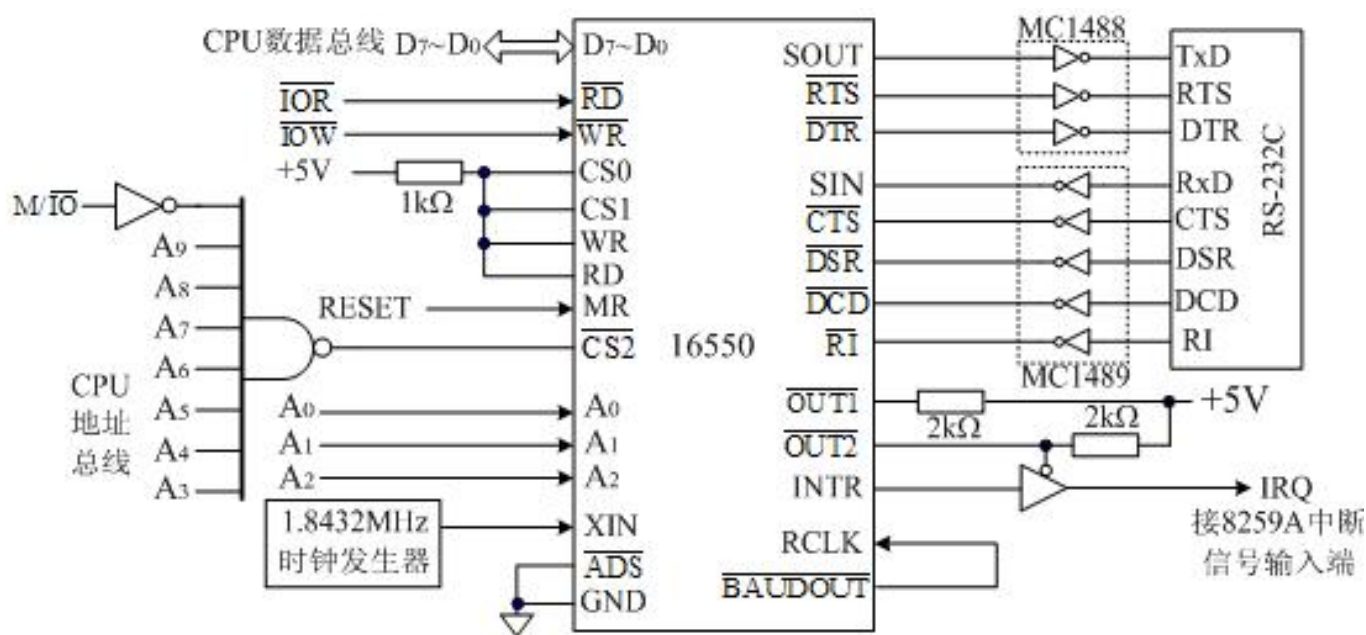


图 10-20 16550 与 CPU 和通信终端连接电路

# 10.4 16550 串行通信编程

## 10.4.3 16550 应用编程

### 1. 16550 的初始化

【例 10-3】设 16550 的端口地址为 3F8H~3FFH，波特率为 1200b/s，异步串行传输的数据位为 8 位长度，偶校验，1 位停止位，设基准时钟频率为 1.8432MHz，除数=60H。编写初始化程序。初始化程序如下：

START	PROC	NEAR	;	过程首
	MOV	AL, 10000000B	;	设置 LCR, 允许访问除数寄存器
	MOV	DX, 3FBH	;	指向通信线路控制寄存器 LCR 端口地址
	OUT	DX, AL	;	设置 DLAB=1
	MOV	AL, 60H	;	1200b/s 对应除数 96=60H
	MOV	DX, 3F8H	;	指向除数寄存器低位 DLL 的端口地址
	OUT	DX, AL		
	MOV	AL, 0		



MOV	DX, 3F9H	; 指向除数寄存器低位 DLM 的端口地址
OUT	DX, AL	
MOV	AL, 00011011B	; 设置 LCR, 8 位数据, 偶校验, 1 位停止位
MOV	DX, 3FBH	; 指向通信线路控制寄存器 LCR 端口地址
OUT	DX, AL	; 设置 DLAB=0, 允许对 FIFO 访问
MOV	AL, 01000111B	; 设置 FCR, 4B 深度, 允许 FIFO, 并清除 FIFO
MOV	DX, 3FAH	; 指向 FIFO 控制字寄存器 FCR 端口地址
OUT	DX, AL	; 允许发生器和接收器工作
MOV	DX, 3FCH	; 指向 MODEN 控制寄存器 MCR 端口地址
MOV	AL, 03H	; 初始化 MODEN 控制寄存器
OUT	DX, AL	; 使 OUT2 输出 1, 禁止中断
MOV	DX, 3F9H	; 指向中断允许寄存器 IER 端口地址
MOV	AL, 0	; 禁止所有中断
OUT	DX, AL	
RET		
START	ENDP	

# 10.4 16550 串行通信编程

## 10.4.3 16550 应用编程

### 2. 以查询方式进行数据收发

【例 10-4】查询方式发送。以下程序段为从 COM1 串行口发送一个字符的子程序，待发送的字符已在 CL 中。

```
SEND:  MOV    DX, 3FDH    ; DX 指向通信线状态寄存器 LSR 地址
        IN     AL, DX     ; 读 LSR 的状态
        TEST   AL, 20H    ; 测 TH=1? , 即发送保持寄存器空否?
        JZ     SEND      ;=0, 不空, 继续查询
        MOV    DX, 3F8H   ;空, DX 指向发送保持寄存器 THR 地址
        MOV    AL, CL     ;待发送字符送 AL
        OUT   DX, AL      ;发送
        RET
```

# 10.4 16550 串行通信编程

## 10.4.3 16550 应用编程

### 2. 以查询方式进行数据收发

【例 10-5】查询方式接收。以下程序段为从 COM1 串行口接收一个字符的子程序，接收的字符暂存于 AL 中。

```
RECV:  MOV    DX, 03FDH    ; DX 指向通信线状态寄存器 LSR 地址
        IN     AL, DX      ; 读 LSR 的状态
        TEST   AL, 0EH;    ; 测试 FE、PE、OE3 位，判断有无错误
        JNZ   ERROR      ; 有错误，转出错处理
        TEST   AL, 01H;    ; 测 DR=1? (已收到字符否?)
        JZ    RECV;       ; 未收到，继续等待
        MOV   DX, 3F8H;    ; DX 指向接收寄存器 RBR 地址
        IN    AL, DX      ; 接收字符
        RET
ERROR:  ...                ; 出错处理
```

# 10.4 16550 串行通信编程

## 10.4.3 16550 应用编程

### 3. 以中断方式进行数据收发

#### (1) 采用中断方式通信的初始化编程。

【例 10-6】初始化程序可在例 10-3 程序的基础上稍作补充。此例设数据接收的基本要求和参数与例 10-3 相同，即传输的数据位是 8 位，采用偶校验和 1 位停止位，波特率为 1200bps 等，并允许接收器中断和线路错误中断。程序如下：

```
START PROC NEAR
    .....
    MOV     AL, 00000101B    ; 中断控制器设置，允许接收器和线路错误中断
    MOV     DX, IER         ; 中断控制端口地址指向 3F9H
    OUT     DX, AL
    MOV     DX, 3FCH        ; 指向 MODEN 控制寄存器 MCR 端口地址
    MOV     AL, 00001011B   ; 初始化 MODEN 控制寄存器
    OUT     DX, AL         ; 使 OUT2 输出 0，允许中断
    .....
ENDPROC
```

# 10.4 16550 串行通信编程

## 10.4.3 16550 应用编程

### 3. 以中断方式进行数据收发

#### (2) 中断处理。

【例 10-7】下面的中断检测程序段就是根据 IIR 的内容，分别控制转移到接收器中断服务程序 RECV、发送器中断访问程序 TRAN 及线路错误中断处理程序 ERR 中。

PUSH	AX	
IN	AL, 3FAH	; 检查中断识别寄存器 IIR, 确定中断类型
CMP	AL, 00000110B	; 是接收器错误中断吗 (测试中断识别码后两位)?
JE	ERR	; 接收器错误, 转移到对应的中断处理程序
CMP	AL, 00000010B	; 是发送寄存器空中断吗?
JE	TRAN	; 发送器寄存器空, 转移到数据发送中断服务程序
CMP	AL, 00000100B	; 是接收寄存器满中断吗?
JE	RCV	; 接收寄存器满, 转移到数据接收中断服务程序

# 10.4 16550 串行通信编程

## 10.4.4 BIOS 串行通信功能用法和示例

### 1. INT 14 指令功能和用法

表 10-3 指令 INT 14H 串行通信 BIOS 功能调用

子功能号	基本功能	入口参数	返回参数
AH=0	初始化串口	AL=初始化参数 DX=通信端口编号: 0: COM1; 1: COM2	AH=通信线路状态寄存器 LSR 的内容; AL=Modem 状态寄存器 MSR 的内容。
AH=1	发送一字符	AL=需要发送的字符 DX=通信端口编号: 0: COM1; 1: COM2	AH 的 D7=0, 发送成功; AH 的 D7=1, 发送失败。 AH 的 D6-D0 为线路状态。
AH=2	接收一字符	DX=通信端口编号: 0: COM1; 1: COM2	AH 的 D7=0 接收成功; AL 中为收到的字符。 AH 的 D7=1 接收失败; AH 的 D6-D0 为线路状态。
AH=3	测试通信线路状态	DX=通信端口编号: =0: COM1; =1: COM2	AH=通信线路状态寄存器 LSR 的内容。 AL=Modem 状态寄存器 MSR 的内容。



# 10.4 16550串行通信编程

## 10.4.4 BIOS串行通信功能用法和示例

### 1. INT 14指令功能和用法

(1) 指令**INT 14**的**0**号子功能的入口参数**AL**中的初始化参数含义

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
最高3位定义串 行通信的波特率			此2位定义 奇偶校验		定义停 止位	此2位定义数 据位长度	

图 10-21 AL 中初始化参数定义

(2) 串行通信端口编号选择

表 10-4 串行通信接口分配

串行接口号	逻辑设备名	基地址	基地址保存单元	保留硬中断
0	COM1	3F8H	0040:0000H	INT OCH (IRQ4)
1	COM2	2F8H	0040:0002H	INT 0BH (IRQ3)



# 10.4 16550串行通信编程

---

## 10.4.4 BIOS串行通信功能用法和示例

### 2. 基于INT 14指令应用的双机串行通信编程



### 【例 10-8】发送端计算机串行通信初始化和发送数据块程序

```
START : MOV AL, 11100011H ; 初始化: 9600 波特、无奇偶校验、数据位长 8 位、1 位停止位
        MOV DX, 0 ; 选择 COM1 通信口
        MOV AH, 0 ; 选择 0 号子功能
        INT 14H ; 完成通信口初始化
        MOV SI, TRANDAT ; 设发送数据块首地址指针
        MOV CX, 100 ; 发送 100 个字节数据的计数值送 CX
SEND :  MOV DX, 0 ; 发送参数设置, 选择 COM1 通信口
        MOV AH, 3 ; 选择 3 号子功能
        INT 14H ; 读通信线路状态寄存器 LSR 的内容于 AH 中
        TEST AH, 60H ; 测试 LSR 的 TE 和 TH 位
        JZ SEND ; 判断发送移位寄存器和数据保持寄存器已空否, 若未空, 继续测试
        MOV DX, 0 ; 发送寄存器已空, 可以发送下一个数据
        MOV AL, [SI] ; 待发送数据送入 AL
        MOV AH, 1 ; 选择 1 号子功能, 发送数据
        INT 14H ; 发送一个数据
        TEST AH, 80H ; 测试 AH 的最高位, 判断是否存在发送错误
        JNZ ERRO ; 发送有错, 进入错误处理程序
        INC SI ; 发送数据缓冲区地址加 1
        LOOP SEND
ERRO :  ...
```

### 【例 10-9】接收端计算机串行通信初始化和接收数据块程序

```
START : MOV AL, 11100011H ; 初始化: 9600 波特、无奇偶校验、数据位长 8 位、1 位停止位
        MOV DX, 0 ; 选择 COM1 通信口
        MOV AH, 0 ; 选择 0 号子功能
        INT 14H ; 完成通信口初始化
        MOV DI, SAVDAT ; 设接收数据块首地址指针
        MOV CX, 100 ; 接收 100 个字节数据的计数值送 CX
RSV : MOV DX, 0 ; 选择 COM1 通信口
      MOV AH, 3 ; 选择 3 号子功能
      INT 14H ; 读通信线路状态寄存器 LSR 的内容于 AH 中
      TEST AH, 01H ; 测试 LSR 的 DR: 判断是否已接收到一个完整的数据
      JZ RSV ; 尚未接收完毕, 返回继续测试
      MOV DX, 0 ; 已接收完毕, 准备将接收到的数据读入。选择 COM1 通信口
      MOV AH, 2 ; 选择接收数据的 2 号子功能,
      INT 14H ; 接收一个数据
      TEST AH, 80H ; 测试 AH 的最高位, 判断是否存在接收错误
      JNZ ERRO ; 接收有错, 进入错误处理程序
      MOV [DI], AL ; 接收的数据正确, 存入缓冲区
      INC DI ; 数据缓冲区地址加 1
      LOOP RSV
ERRO : ...
```

# 10.5 其它常用串行通信标准

## 10.5.1 通用串行总线USB

### 1. USB总线的特点

### 2. USB连接器

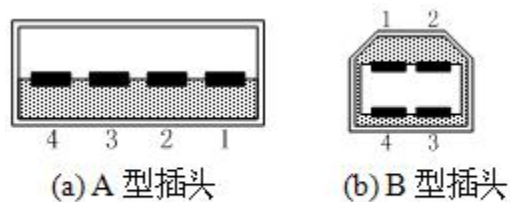


图 10-22 USB 插头示意图

## 10.5.2 IEEE1394串行总线



# 10.6 SOC系统中16550核的应用实例

---

## 10.6.1 SOC系统与PC异步串行通信

### 1. 电路结构

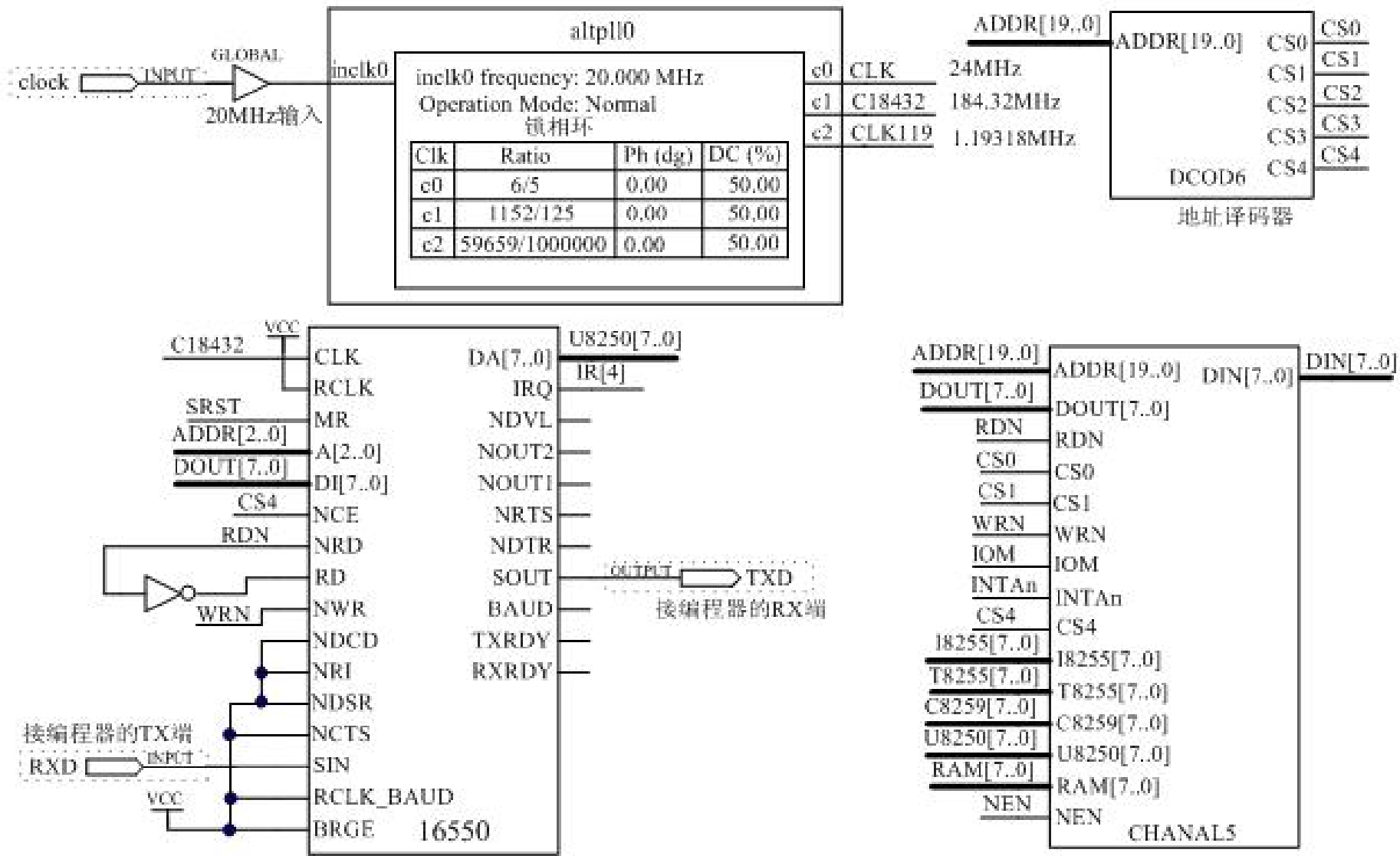


图 10-23 在 SOC 微机系统电路中的 16550 核连接情况



# 10.6 SOC系统中16550核的应用实例

---

## 10.6.1 SOC系统与PC异步串行通信

### 2. 程序说明

### 【例 10-10】

title TEST-UART

.model tiny

.code

.8086

RBR	EQU	3F8H	: 数据接收缓冲寄存器 RBR(读)
THR	EQU	3F8H	: 数据发送保持寄存器 THR(写)
FCR	EQU	3FAH	: FIFO 控制寄存器 FCR(写)
LCR	EQU	3FBH	: 通信线路控制寄存器 LCR
LSR	EQU	3FDH	: 通信线路状态寄存器 LSR
DLL	EQU	3F8H	: 波特率除数寄存器低 8 位字节 DLL
DLM	EQU	3F9H	: 波特率除数寄存器高 8 位字节 DLM

ORG 0000H

DBuf \

dw 2A15h

stackr \

db 256 dup (0)

stacke dw 0000h

start : mov ax, 0000h

mov cs, ax

mov ds, ax

mov es, ax

mov ss, ax

mov sp, offset stacke

mov bx, 0

mov al, 80H

mov dx, 37BH

out dx, al ;为 8255(1)写入控制字 80H



```

mov     dx, LCR
mov     al, 80h
out     dx, al    ; 写入 16550 的控制字: 80H, 使 DLAB=1, 允许访问除数寄存器
mov     dx, DLM
mov     al, 04H   ; 16550 输入的基频: 184.32MHz
out     dx, al    ; 写入波特率除数寄存器高 8 位字节 DLM: 04H
mov     dx, DLL
mov     al, 0B0H
out     dx, al    ; 写入波特率除数寄存器低 8 位字节 DLL: 0B0H
mov     dx, LCR
mov     al, 03h   ; 写入 16550 的控制字: 03H
out     dx, al
mov     dx, FCR
mov     al, 41H
out     dx, al    ; 写入 FIFO 控制寄存器 FCR: 41H
senddat: mov     ax, DBuf
mov     dx, 378h
out     dx, al    ; 通过 8255(1)的 A 口输出缓冲区 DBUF 的低 8 位
mov     al, ah
mov     dx, 379h
out     dx, al    ; 通过 8255(1)的 B 口输出缓冲区 DBUF 的高 8 位
mov     dx, THR
mov     al, 12H
out     dx, al    ; 向数据发送保持寄存器 THR 写入: 12H
mov     al, 34H
out     dx, al    ; 向数据发送保持寄存器 THR 写入: 34H

```



```

    mov     al, 78H
    out    dx, al      ; 向数据发送保持寄存器 THR 写入: 78H
wait1: mov    dx, LSR
    in     al, dx      ; 读通信线路状态寄存器 LSR
    and    al, 40h
    jz     wait1       ; 尚未发送完毕, 等待
    call   wtd         ; 判断 FIFO 中的数据是否就绪
    mov    dx, RBR
    in     al, dx      ; 读上位机发来的数据
    mov    ch, al      ; 读数送 CX 高位
    call   wtd         ; 继续了解 FIFO
    mov    dx, RBR     ; 再读上位机发来的数据
    in     al, dx
    mov    cl, al      ; 读数送 CX 低位
    mov    DBuf, cx   ; CX 中的数送缓冲区 DBUF
    mov    ax, CX
    jmp    senddat    ; 将 CX 中的数通过 8255 显示在数码管
wtd     PROC
    mov    dx, LSR
    in     al, dx
    and    al, 01h
    jz     wtd         ; 判断 FIFO 中的数据是否就绪, 未就绪, 继续等待
    RET
wtd     endp
org     03ff0h
db      0EAh
dw      start
dw      0000h
end

```

# 10.6 SOC系统中16550核的应用实例

## 10.6.1 SOC系统与PC异步串行通信

### 3. 驱动程序安装及端口设置



图 10-24 串行通信测试工具和 USB-RS232 驱动串行存放路径

# 10.6 SOC系统中16550核的应用实例

## 10.6.1 SOC系统与PC异步串行通信

### 3. 驱动程序安装及端口设置



图 10-25 为 USB-串行通信接口设置参数

# 10.6 SOC系统中16550核的应用实例

## 10.6.1 SOC系统与PC异步串行通信

### 3. 驱动程序安装及端口设置



图 10-26 选择端口参数

# 10.6 SOC系统中16550核的应用实例

## 10.6.1 SOC系统与PC异步串行通信

### 4. 系统设计与硬件测试

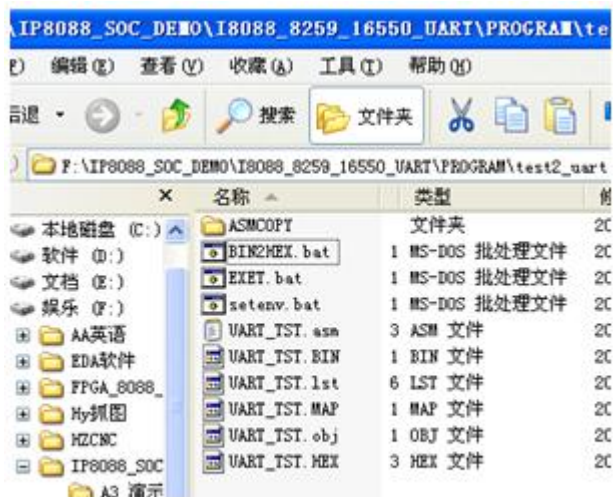


图 10-27 串行通信测试汇编程序路径



图 10-28 串行通信测试工具 Commix 用法

# 10.6 SOC系统中16550核的应用实例

## 10.6.1 SOC系统与PC异步串行通信

### 4. 系统设计与硬件测试

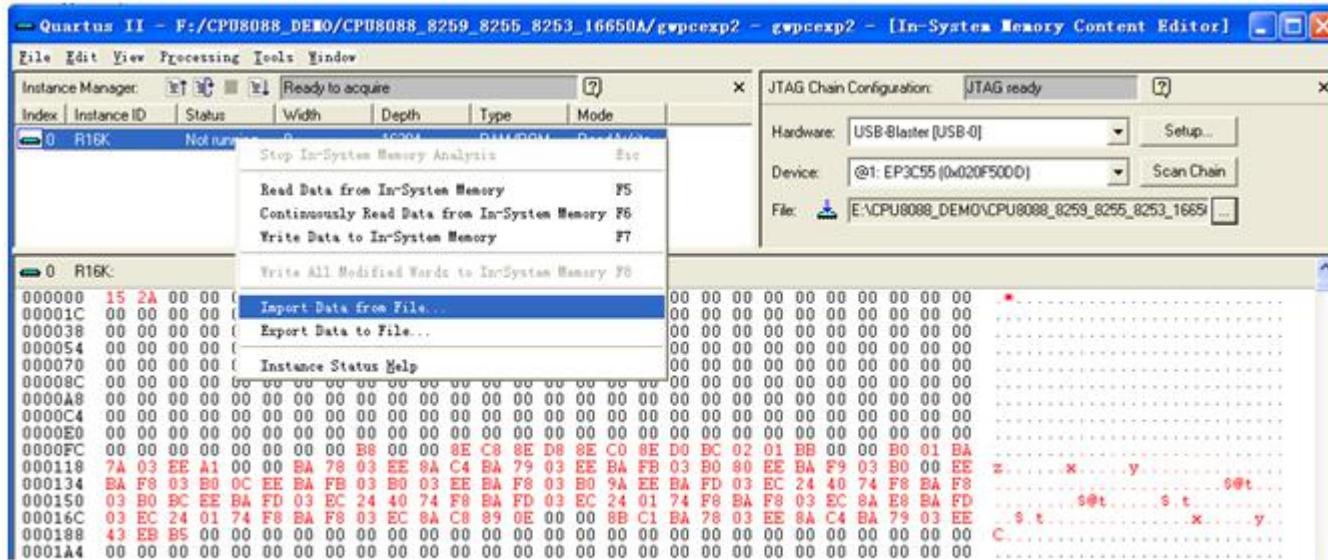


图 10-29 利用 In-System Memory Content Editor 下载 HEX 汇编文件



# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

### 1. 基本原理

表 10-5 GPS 的数据类型

NMEA 信息前缀	信息类型	信息内容
SGPGGA	GPS 定位信息	Time, position and fix type data
SGPGLL	定位地理信息	Latitude, longitude, time of position fix and status
SGPGSA	当前卫星信息	GNSS DOP and active satellites
SGPGSV	可见卫星信息	Setellites in view
SGPRMC	推荐最小定位信息	Recommended minimum specific GNSS data
SGPVTG	地面速度信息	Speed and course over ground

\$信息类型, x, x, x, x, x, x, x, x, x, x, x, x, x

SGPRMC,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>\*hh

# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

### 1. 基本原理

表 10-6 \$GPRMC 各数据的意义

Field	Data	Unit	Notes
Message ID	\$GPRMC		RMC protocol header.
UTC Time	161229.487		hhmmss.sss
Status	A		A: Data valid or V: Data invalid.
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south.
Longitude	12158.3416		ddmm.mmmm
E/W indicator	W		E=east or W=west.
Speed over ground	0.13	knot	Speed over ground
Course over ground	309.62	degree	Course over ground
Date	120598		ddmmyy, current date.
Magnetic variation		degree	Not used.
Mode	A		A=Autonomous, D=DGPS, E=DR.
Checksum	*10		
<CR> <LF>			Message terminator.



# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

表 10-6 SGPRMC 各数据的意义

### 1. 基本原理

Field	Data	Unit	Notes
Message ID	SGPRMC		RMC protocol header.
UTC Time	161229.487		hhmmss.sss
Status	A		A: Data valid or V: Data invalid.
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south.
Longitude	12158.3416		ddmm.mmmm
E/W indicator	W		E=east or W=west.
Speed over ground	0.13	knot	Speed over ground
Course over ground	309.62	degree	Course over ground
Date	120598		ddmmyy, current date.
Magnetic variation		degree	Not used.
Mode	A		A=Autonomous, D=DGPS, E=DR.
Checksum	*10		
<CR> <LF>			Message terminator.

# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

### 2. 硬件电路构建

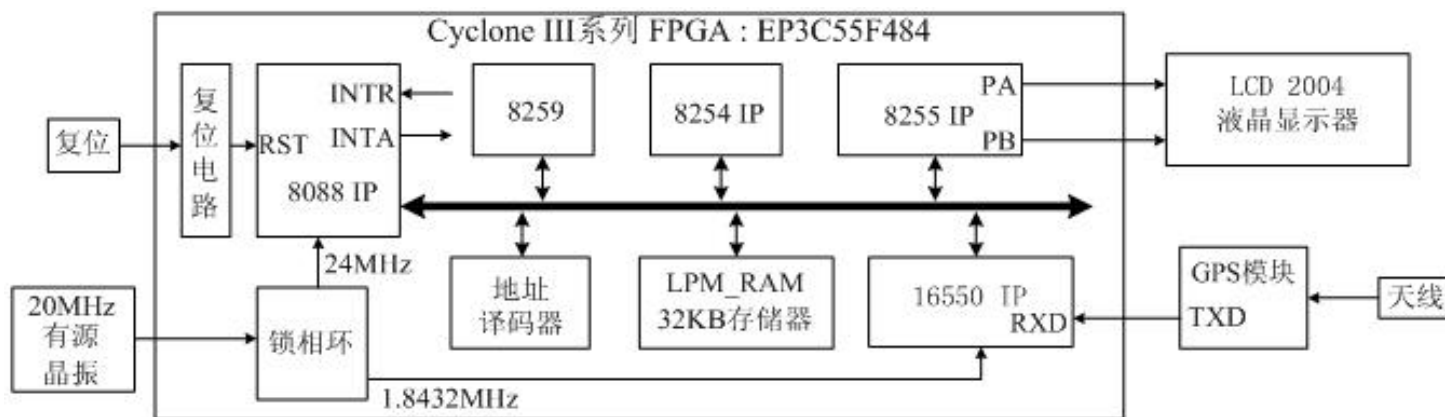


图 10-30 GPS 数据接收处理和显示电路系统

# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

### 3. 程序设计

```
mov     dx, COM1_LCR
mov     al, 80h           ; DIAB = 1, 允许设置波特率
out     dx, al
mov     dx, COM1_DLM     ; 设置波特率, 波特率= 9600 bps
mov     al, 0            ; 波特率系数高位=0
out     dx, al
mov     dx, COM1_DLL
mov     al, 12           ; 波特率系数低位=12
out     dx, al
mov     dx, COM1_LCR     ; 设置数据格式: DIAB = 0,
mov     al, 03h          ; 8 位数据, 无奇偶校验, 1 个停止位
out     dx, al
```

# 10.6 SOC系统中16550核的应用实例

## 10.6.2 利用8088 SOC系统完成GPS信息的接收与显示

### 3. 程序设计

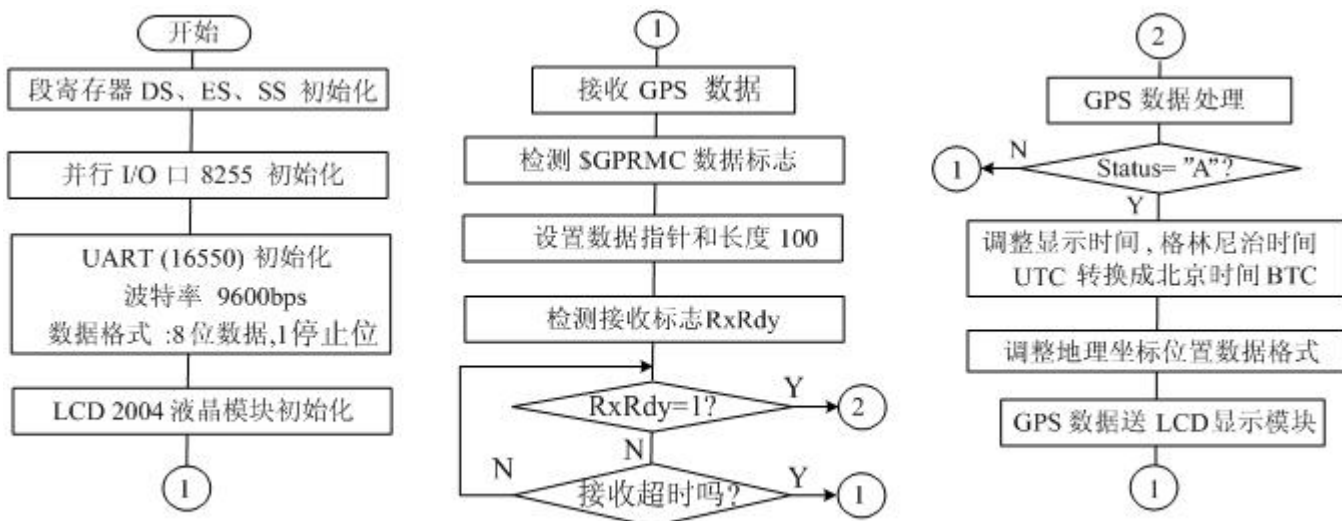


图 10-31 GPS 数据接收程序流程图

# 10.6 SOC系统中16550核的应用实例

## 10.6.3 单片FPGA中双SOC系统间的异步串行通信

### 1. 硬件电路构建

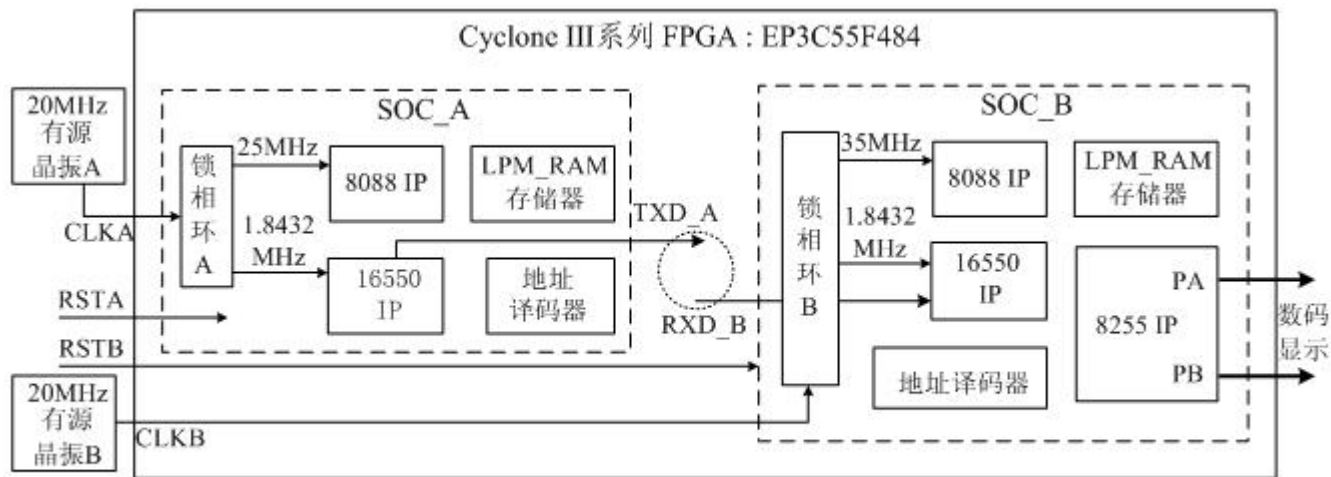


图 10-32 单片 FPGA 中的双 SOC 系统的异步串行通信电路模块图

### 2. 硬件测试



# 10.6 SOC系统中16550核的应用实例

---

## 10.6.3 单片FPGA中双SOC系统间的异步串行通信

### 1. 硬件电路构建