

EM78P451

一 概括描述

EM78P451 是采用高速 CMOS 工艺制造的低功耗 8 位单片机。其操作核心由类 RISC 结构实现。其一次可编程 (OTP) 版本对于大规模生产和工程试验都是很方便的。不论用户购买量是多少, 其价格都是令人满意的。EM78P451 具有 SPI (串行外围接口) 功能、便于实现 RS-232 接口。这对于通信非常有用。指令集仅包含 57 条指令。EMC 在线仿真器、宏汇编和 Easy C 编译器可用于该芯片的应用开发过程。

二 功能特点

- ◆ 工作电压范围: 2.2V~5.5V
- ◆ 工作温度范围: 0℃~70℃
- ◆ 工作频率范围: 晶体模式 DC-20MHz (5V), DC-8MHz (3V)
 RC 模式 DC-4MHz (5V), DC-4MHz (3V)
- ◆ 4K×13 位片内 ROM
- ◆ 具有 SPI 功能
- ◆ 11 个特殊功能寄存器
- ◆ 140×8 位通用寄存器 (SRAM)
- ◆ 5 个双向 I/O 端口 (35 个 I/O 引脚)
- ◆ 3 个 LED 直接驱动引脚, 内部有串联电阻
- ◆ 片内 RC 振荡器
- ◆ 片内上电复位
- ◆ 5 级堆栈
- ◆ 8 位实时定时/计数器 (TCC), 溢出产生中断
- ◆ 每个指令周期为 2 个或 4 个时钟周期
- ◆ 具有低功耗模式
- ◆ 可编程 I/O 引脚变化唤醒功能
- ◆ 片内可编程自由运行 WDT
- ◆ 12 个唤醒引脚
- ◆ 2 个漏极开路引脚
- ◆ 2 个 R-option 引脚
- ◆ 封装形式:
 - (1) EM78P451P, 40 引脚 SDIP (2) EM78P451M, 40 引脚 SOP
 - (3) EM78P451K, 42 引脚 SHRINK (4) EM78P451K, 44 引脚 QFP
- ◆ 可重装计数器
- ◆ 5 种中断:
 - 1 引脚变化 (/INT)
 - 2 端口变化
 - 3 SPI 结束
 - 4 TCC 溢出
 - 5 可重装计数器匹配

三 引脚分配

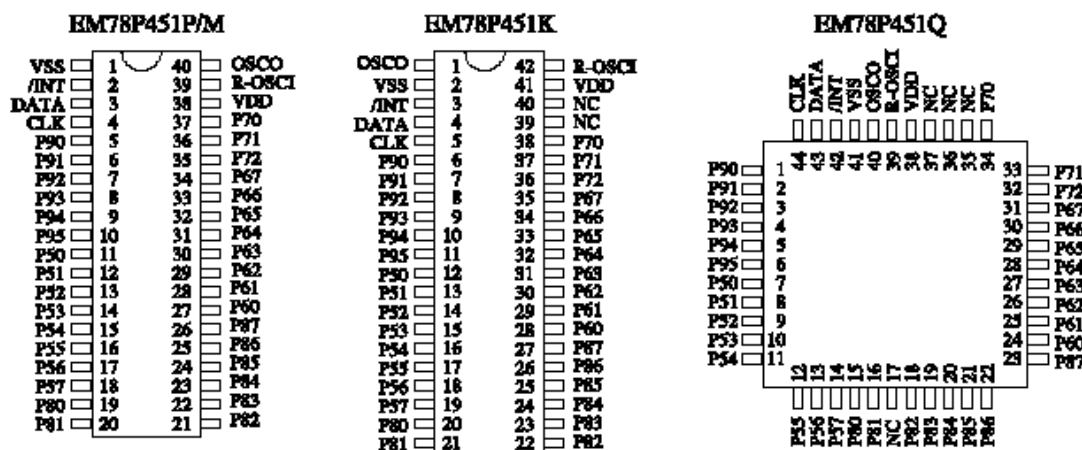


图 1 引脚分配

四 功能框图

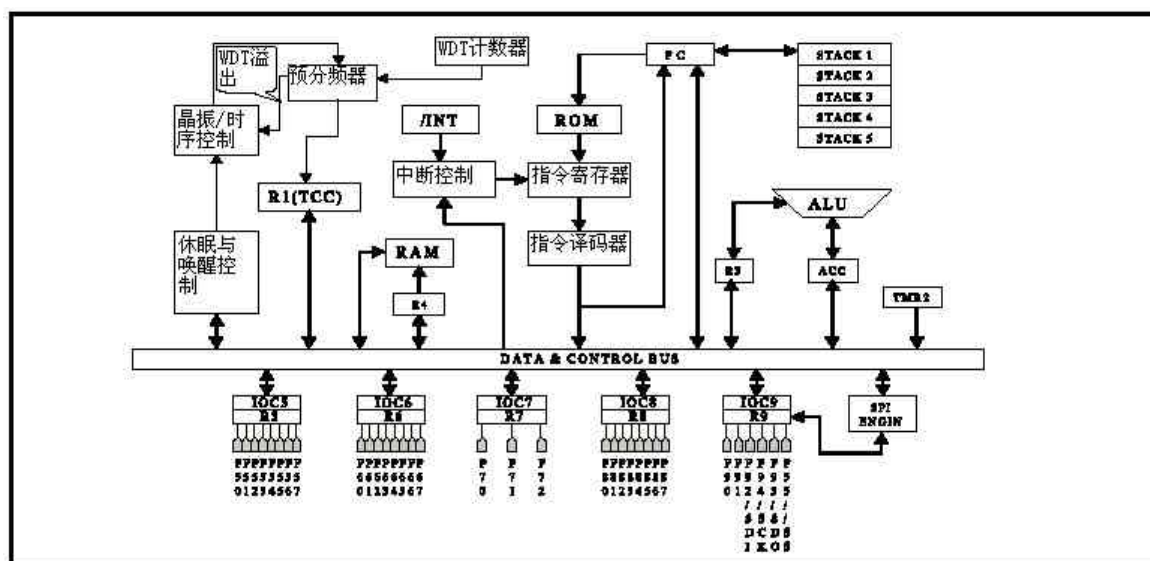


图2 功能框图

五 引脚描述

表 1 引脚描述——EM78P451

符号	类型	功能描述
R-OSCI	I	XTAL 模式中为晶体输入；RC 模式中接 56K 上拉电阻产生 1.8432MHz 频率
OSCO	O	XTAL 模式中为晶体输出；RC 模式中 为指令时钟输出
P90~P95	I/O	端口 9 为 6 位双向 I/O 口，所有引脚可由软件各自设置为上拉
P80~P87	I/O	端口 8 为 8 位双向 I/O 口，P80 与 P81 可用于 R-option
P70~P72	I/O	输出时为 LED 直接驱动引脚，内部有上拉电阻。由软件定义。
CLK	I/O	P74 和 P76 连接，P74 可由软件置为上拉，P76 可置为漏极开路输出

DATA	I/O	P75 和 P77 连接, P75 可由软件置为上拉, P77 可置为漏极开路输出
P60~P67	I/O	端口 6 为 8 位双向 I/O 口, 所有引脚可由软件各自设置为上拉
P50~P57	I/O	端口 5 为 8 位双向 I/O 口, 所有引脚可由软件各自设置为上拉
VDD	-	电源
VSS	-	地
/INT	I	中断史密斯触发引脚, 中断功能由下降沿触发, 用户可由软件使能之, 内部上拉电阻约 50K.
SDI	I/O	串行数据输入
SDO	I/O	串行数据输出
SCK	I/O	串行数据时钟
/SS	I/O	/从器件选择

六 功能描述

6.1 操作寄存器

1) R0 寄存器（间址寄存器）

R0 并非实际存在的寄存器。它的主要功能是做为间接寻址指针。任何以 R0 为指针的指令实际上是对 RAM 选择寄存器 R4 所指的数据进行操作。

2) R1（定时器/计数器）

- 由指令周期时钟触发加 1 操作
- 可读写

3) R2（程序计数器 PC）和堆栈

- R2 与硬件堆栈为 12 位宽, 结构如图 3 所示。

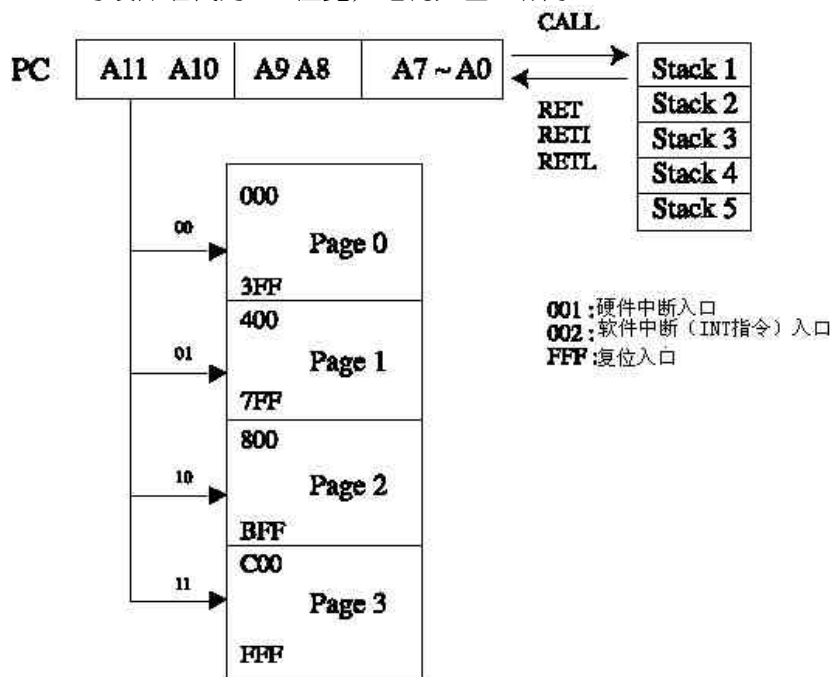


图3 程序计数器结构

- 产生 4096×13 位片内 OTP ROM 地址以获取对应程序指令代码。一个程序页为 512

字长。

- 复位后 R2 所有位均置 1。
- JMP 指令直接装载 R2 低 10 位值，因此 JMP 指令跳转范围为一个程序页面内。
- CALL 指令先将 PC+1 入栈，而后装载 R2 低 10 位值，因此子程序入口地址在同一页面内。
- RET、RETI、RETL 指令将栈顶数据装入 PC。
- “ADD R2, A” 允许将一有关地址加到当前 PC 上，但同时 PC 第 9、10 位将被清 0。
- “MOV R2, A” 将 ACC 中的 8 位地址装入 PC 低 8 位，PC 第 9、10 位被清 0。
- 任何对 R2 内容进行直接修改的指令将引起 R2 第 9、10 位清 0。因此产生的跳转限于同一页面的前 256 个地址。
- TBL 指令将一相应地址加到 PC 上，但 PC 第 9、10 位不变。故由此引起的跳转可在任意程序页第 1 (2 或 3 或 4) 段 256 个地址。
- EM78P451 情况下，最高两位 (A10 和 A11) 将在执行 JMP、CALL 或任何改变 R2 内容的指令时由 R3 的 PS0~PS1 装入。

4) R3 (状态寄存器)

7	6	5	4	3	2	1	0
GP	PS1	PS0	T	P	Z	DC	C

- 第 7 位 (GP) 通用读写位。
- 第 5~6 位 (PS0、PS1) 程序页面选择位。用于选择程序存储器页面，在执行 JMP、CALL 或任何改变 R2 内容的指令时装入 R2 第 11、12 位。注意，RET、RETL、RETI 指令不改变这几位值。也就是说，将返回到程序调用处，不论 PS0、PS1 为何值。

PS1	PS0	程序存储器页面地址
0	0	第 0 页[000-3FF]
0	1	第 1 页[400-7FF]
1	0	第 2 页[800-BFF]
1	1	第 3 页[C00-FFF]

- 第 4 位 (T) 溢出位，执行 SLEEP 或 WDTN 指令或上电后置 1，WDT 溢出时清 0。
- 第 3 位 (P) 低功耗位，执行 WDTN 指令或上电后置 1，执行 SLEEP 指令后清 0。
- 第 2 位 (Z) 零标志位，当算术运算或逻辑运算结果位 0 时该位置 1。
- 第 1 位 (DC) 辅助进位标志。
- 第 0 位 (C) 进位标志。

5) R4 (RAM 选择寄存器)

- 第 0~5 位在间接寻址方式中用于选择 RAM 寄存器 (00~3F)。
- 第 6、7 位用于选择寄存器组。
- 若未使用间接寻址方式，可做 8 位通用寄存器。
- 参见图 4 所示数据存储器结构。

6) R5~R8 I/O 寄存器

- 4 个 I/O 寄存器
- P74 和 P76 可从 DATA 引脚读写数据，P75 和 P77 可从 CLK 引脚读写数据。

7) R9 (端口 9)

- 6 位 I/O 寄存器，高两位读数为 0。

8) RA~R1F

- 参见 6.5 SPI 模式和 6.6 计数器 1

9) R20~R3E 通用寄存器

10) R3F (中断状态寄存器)

7	6	5	4	3	2	1	0
-	-	-	-	TM1IF	RBFIF	EXIF	TCIF

- 1 表示有中断请求，0 表示没有。
- 第 0 位 (TCIF) TCC 溢出中断标志。TCC 溢出时置 1，软件清 0。
- 第 1 位 (EXIF) 外部中断标志。由 /INT 引脚上的下降沿置 1，软件清 0。
- 第 2 位 (RBFIF) 读缓冲器满中断标志。
- 第 3 位 (TM1IF)
- 第 4~7 位未使用，读出为 0。
- R3F 可软件清 0，但不可置 1。
- IOCF 为中断屏蔽寄存器。
- 注意：读 R3F 的结果为 R3F 和 IOCF 相与的结果。

6.2 特殊功能寄存器

1) A (累加器) 用于内部数据传输，或指令操作数保持。不可寻址。

2) CONT (控制寄存器)

7	6	5	4	3	2	1	0
/PHEN	/INT	-	-	PAB	PS2	PS1	PS0

- 第 0 位~第 2 位 (PSR0~PSR2) TCC/WDT 预分频位。

PS2	PS1	PS0	TCC 分频率	WDT 分频率
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

- 第 3 位 (PAB0) 预分频器分配位，为 0 分给 TCC，为 1 分给 WDT。
- 第 4、5 位未用，读出为 0。
- 第 6 位 (/INT) 中断允许标志。0 表示已由 DISI 指令或硬件中断屏蔽中断，1 表示已由 ENI 指令或 RETI 指令允许中断。CONTW 指令不能写该位。
- 第 7 位 (/PHEN) I/O 引脚上拉使能标志。为 1 时禁止上拉功能，为 0 时使能 P60~P67、P74~P75、P90~P91 上拉功能。
- CONT 寄存器第 0~3 位、第 7 位可读写。

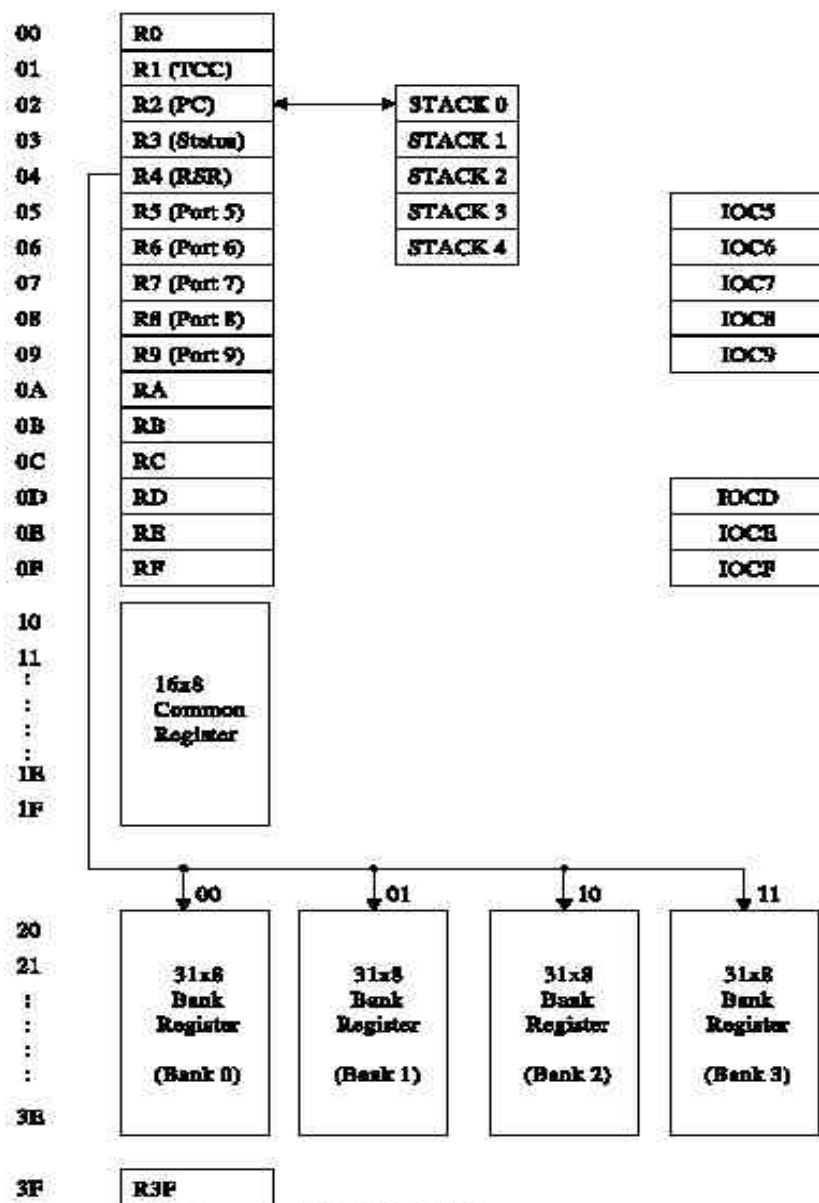


图4 数据寄存器结构

3) IOC5~IOC9 (I/O 控制寄存器)

- 1 定义对应 I/O 引脚为高阻状态，0 定义其为输出
- P74、P76 不可同时定义为输出，P75、P77 也是这样。
- IOC9 只使用了低 6 位。

4) IOCD (上拉控制寄存器)

7	6	5	4	3	2	1	0
S7	-	-	-	/PU9	/PU8	/PU6	/PU5

- /PU5、/PU6、/PU8、/PU9 默认值为 1，禁止上拉功能。
- /PU6、/PU9 和 /PHEN 相与，任意一个为 0 即使能上拉。
- S7 定义了 P70~P72 的驱动能力。0 为正常输出，1 为增强的 LED 驱动输出。

5) IOCE (WDT 控制寄存器)

7	6	5	4	3	2	1	0
-	ODE	WTE	SLPC	ROC	-	-	/WUE

- 第 0 位 (/WUE) P60~P67、P74~P75、P90~P91 唤醒功能控制位。为 0 使能，

为 1 禁止。该位可读写。

- 第 3 位 (ROC) 用于 R-option 功能。ROC 置 1, 则 R-option 引脚 P80、P81 状态可读出, ROC 清 0 则禁止该功能。另外, 介绍一下 R-option 功能。用户必须将 P81 或/和 P80 引脚通过 560K 外部电阻 (Rex) 与 VSS 相连。若 Rex 与 VDD 相连/未连, P80 (P81) 状态读数为 0/1。参见图 7 (B)。该位可读写。

- 第 4 位 (SLPC) 该位在唤醒信号下降沿由硬件置 1, 由软件清 0。该位用于控制振荡器的工作。由高到低变化时振荡器禁止 (振荡器停振, 单片机进入休眠模式 2), 由低到高变化时振荡器使能 (单片机由休眠模式 2 唤醒)。为了保证振荡器输出的稳定, 一旦振荡器再次使能, 在执行下一指令之前有大约 18ms 延时 (OST)。不论代码选项位 WTC 是否为 0, OST 都会由唤醒激活。唤醒后, 若 WTC 为 1 则 WDT 使能。休眠模式 2 的框图和输入触发唤醒如图 5。该位可读写。

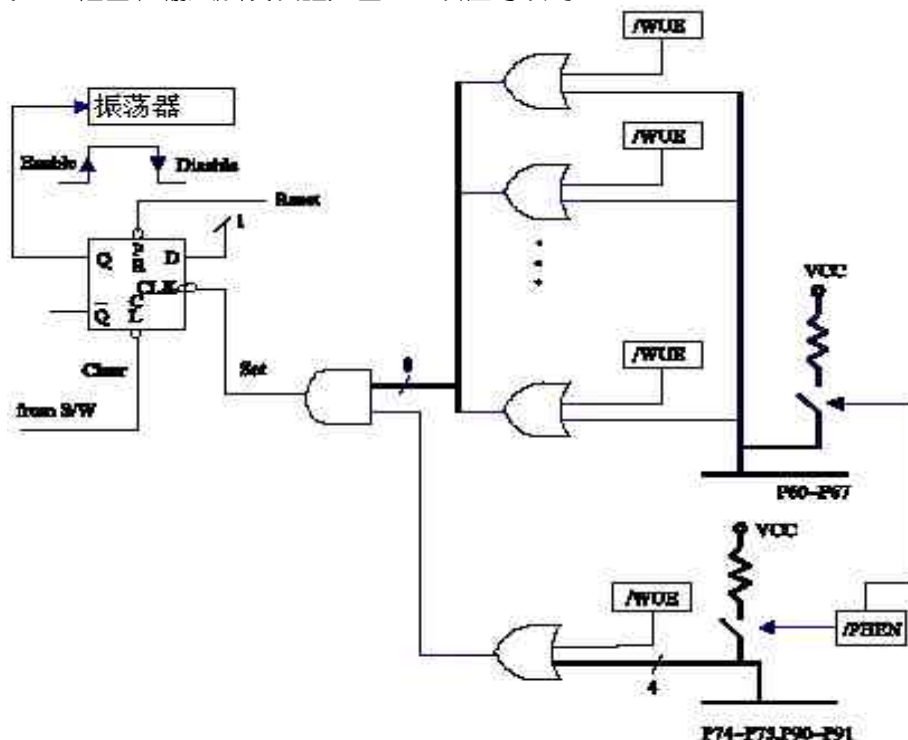


图5 休眠模式和I/O口唤醒电路框图

- 第 5 位 (WTE) WDT 控制位。代码选项寄存器位 WTC 为 1 时该位才有效。此时 WTE 为 1 使能 WDT, 为 0 禁止。WTC 为 0 时该位无效, 无论 WTE 为何值 WDT 均禁止。该位可读写。

- 第 6 位 (ODE) 漏极开路控制位。为 0 时 P76~P77 为通常 I/O 引脚, 为 1 时 P76~P77 为内部漏极开路引脚。该位可读写。

- 第 1~2 和 7 位未使用。

6) IOCF (中断屏蔽寄存器)

7	6	5	4	3	2	1	0
-	-	-	-	T1IE	SPIIE	EXIE	TCIE

- 第 0 位 (TCIE) TCIF 中断使能位。为 0 禁止, 为 1 使能。
- 第 1 位 (EXIE) EXIF 中断使能位。为 0 禁止, 为 1 使能。
- 第 2 位 (SPIIE) SPI 中断使能位。为 0 禁止, 为 1 使能。
- 第 3 位 (T1IE) T1IF 中断使能位。为 0 禁止, 为 1 使能。
- 第 4~7 位未使用。
- 该寄存器可读写。

6.3 TCC/WDT 预分频器

对于 TCC 或 WDT 有一个 8 位寄存器做预分频器。在同一时间它只能分配给其中一方，这由 CONT 寄存器的 PAB 位决定。PSR0~PSR2 确定分频系数。若分配给 TCC，则每次写 TCC 操作均将预分频器清 0。若分配给 WDT，则 WDT 和预分频器均在执行 WDTC 和 SLEP 指令时清 0。TCC/WDT 电路框图如图 6 所示。

R1 (TCC) 为 8 位定时器/计数器。每个指令周期 TCC 加 1 (无预分频器)。

WDT 是一个自由运行的片内 RC 振荡器。当振荡驱动器关闭后，WDT 依然运行，如在休眠模式下即如此。WDT 溢出将引起复位 (若 WDT 使能)。在正常工作时，WDT 可由软件设置 IOCE 的 WTE 位来使能或禁止。在没有预分频情况下，WDT 溢出时间约为 18ms。

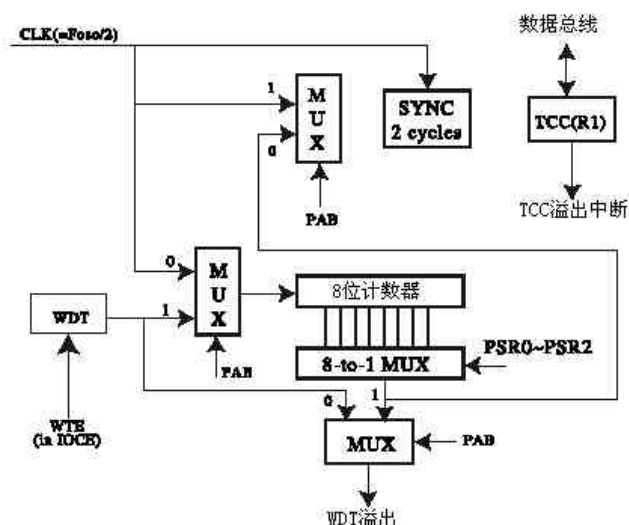


图6 WDT TCC框图

6.4 I/O 端口

I/O 端口 PORT5~PORT9 均为双向三态 I/O 口。P60~P67、P74~P75、P90~P91 具有内部上拉功能和可编程唤醒功能。P76~P77 可由软件设置为漏极开路输出。P80~P81 可由软件设置为 R-option 功能。当时用 R-option 功能时，建议将 P80~P81 设置为输出。在使能 R-option 功能过程中，P80~P81 必须置为输入。低功耗应用时，R-option 功能中若 P80 (P81) 接有外部电阻，可能需要考虑其电流消耗。

程序控制中，I/O 端口可由控制寄存器 (IOC5~IOC9) 设置为输入或输出。I/O 寄存器和 I/O 控制寄存器都是可读写的。I/O 接口电路如图 7 所示。注意，从 I/O 端口读数据时，输入和输出不同状态的读路径不同。

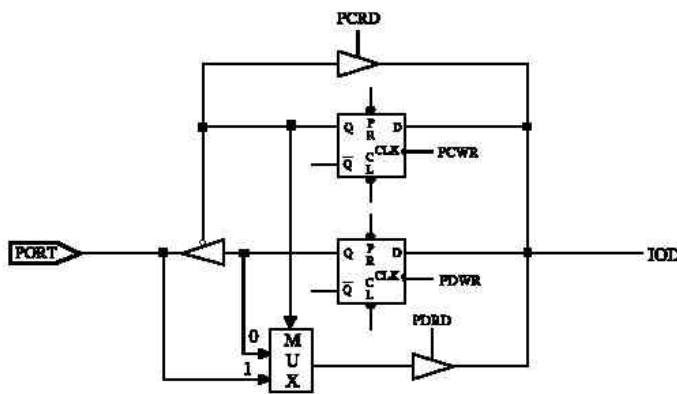


图7 I/O端口、I/O控制寄存器电路

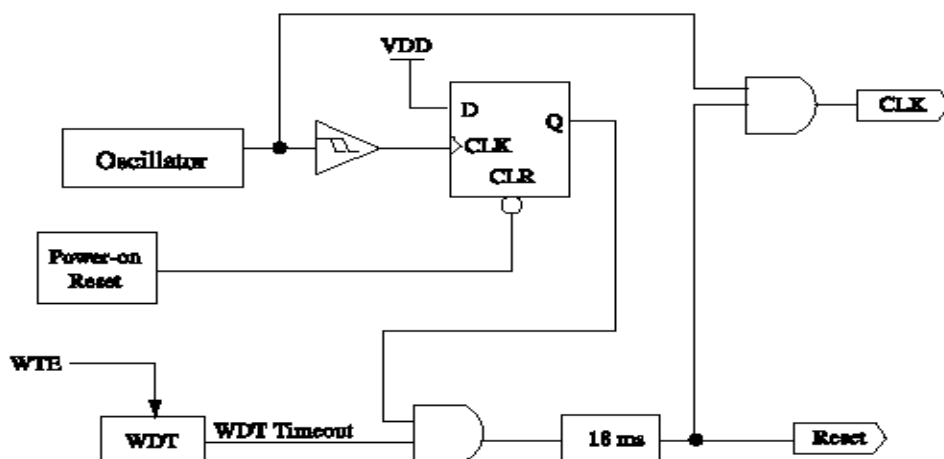


图 8 复位控制方框图

6.5 串行外围接口模式

1) 综述及特点

综述：图 9、10、11 所示为 EM78P451 在 SPI 模式如何与别的设备通信。若 EM78P451 为主控制器，它通过 SCK 发出时钟。同时两个字节数据被发送和接收。若 EM78P451 为从器件，其 SCK 引脚应定义为输入。数据将基于时钟发生率和信号沿连续发送。

特点：

- 可在主模式或从模式下工作
- 3 线或 4 线同步通信，即全双工方式
- 通信波特率可编程
- 时钟信号极性可编程
- 读缓冲器满，产生中断标志
- 位传输速度最大可达 8MHz

2) 功能描述

下述为实现 SPI 通信的信号功能：

- P93/SDO 串行数据输出
- P92/SDI 串行数据输入
- P94/SCK 串行时钟
- P95//SS 从模式选择（可选）。在从模式下需要。
- RBF 由缓冲器满检测器置 1，由软件清 0。

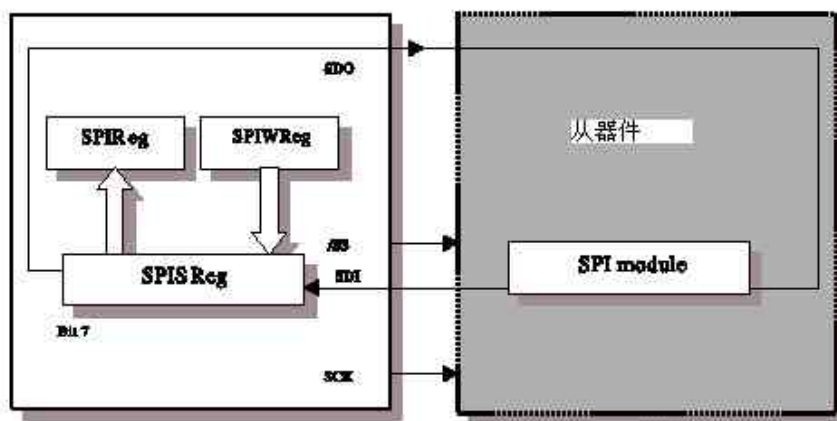


图 9 单主/从 SPI 通信

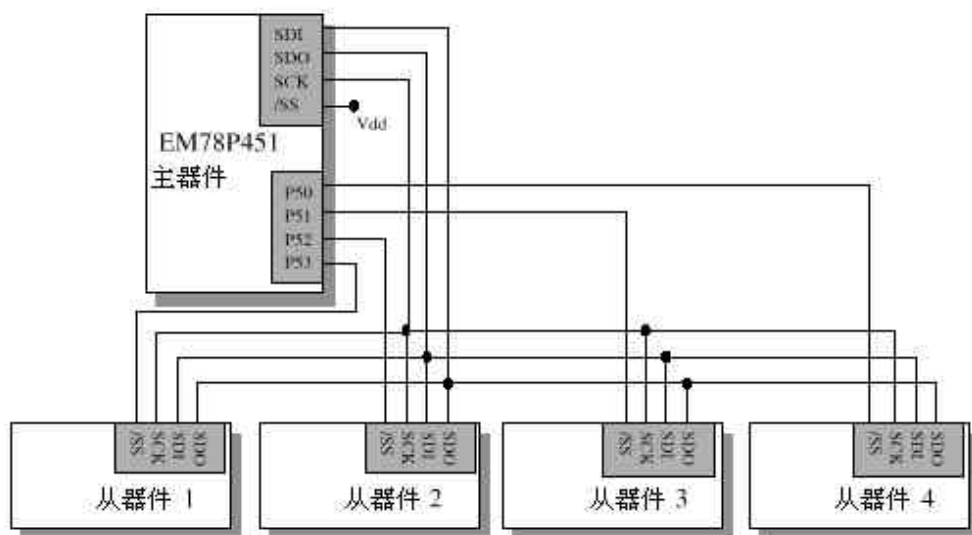


图10 单主器件多从器件SPI通信结构

- ◆ RBF1 由缓冲器满检测器置1，由软件清0
- ◆ Buffer Full Detector 当8位移动结束时置1
- ◆ SE 将数据装入SPIS寄存器，开始传送
- ◆ SPIS寄存器 各个字节数据由此一位一位地移进或移出。高位在前。SPIS和SPIW寄存器同时装入数据。只要数据一写进，SPIS便开始传输。接收完毕后，接收到的数据将被送到SPIR寄存器。RBF和RBF1标志置1。
- ◆ SPIR寄存器 读缓冲寄存器。当8位接收完成后其中数据被更新。在下一接收完成之前，数据必须被读出。读SPIR寄存器时RBF寄存器清0。
- ◆ SPIW寄存器 写缓冲寄存器。在8位传送完成之前，缓冲器不接受任何写操作。传输未完成则SE标志置1。传输结束后该标志清0。用户由此可判断写操作是否完成。
- ◆ SBR2~SBR0 对时钟频率/分频率和时钟源编程。
- ◆ Clock Select 选择内部时钟或外部时钟为传输时钟。
- ◆ Edge Select 对ES编程以选择合适的时钟沿。

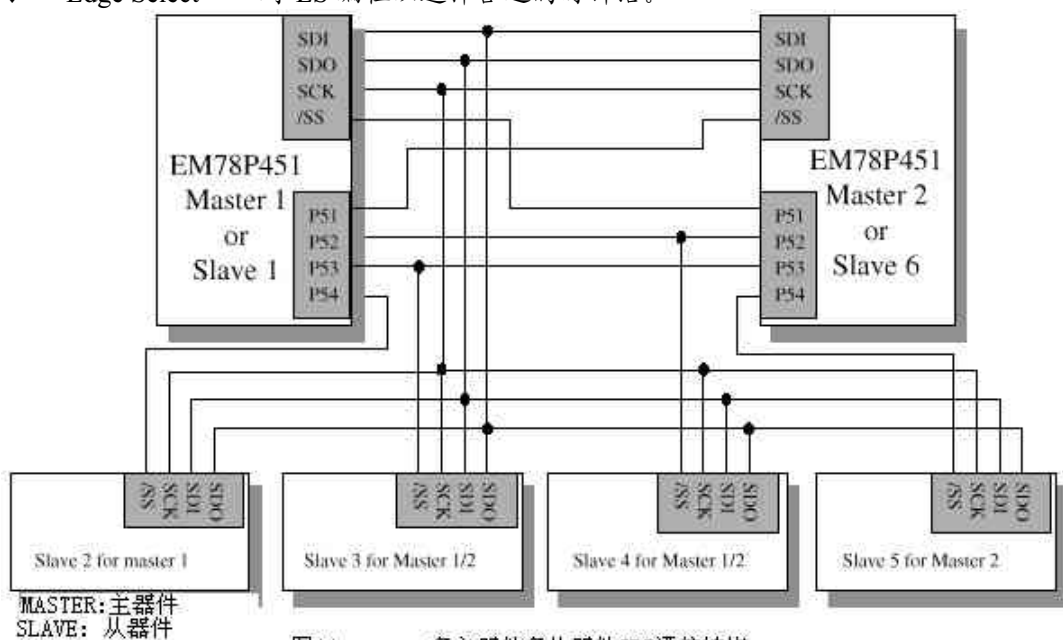


图11 多主器件多从器件SPI通信结构

3) 信号和引脚描述

如图 12 所示的 4 个引脚 SDI、SDO、SCK、/SS 将在此详细描述。

SDI/P92 (引脚 7)

- 串行数据输入。
- 串行接收数据，高位在前，低位在后。
- 未选择应置为高阻状态。
- 主从器件的锁存时钟分频率和时钟沿应设置为相同。
- 本次接收到的数据将冲掉上一次收到的数据。
- SPI 接收完成后，RBF 和 RBFIF 都置 1。
- 时序如图 12、13 所示。

SDO/P93 (引脚 8)

- 串行数据输出
- 串行输出数据，高位在前，低位在后。
- 主从器件的锁存时钟分频率和时钟沿应设置为相同。
- SPI 输出完成后，SE 清 0。
- 时序如图 12、13 所示。

SCK/P94 (引脚 9)

- 串行传输时钟。
- 由主器件产生。
- 对 SDI 和 SDO 引脚的数据通信进行同步。
- ES 位用于选择通信时钟沿。
- SBR0~SBR2 用于选择通信波特率。
- ES, SBR0~SBR2 位对于从模式无效。
- 时序如图 12、13 所示。

/SS/P95 (引脚 10)

- 从模式选择，低电平有效。
- 由主器件产生，让从器件接收数据。
- 在 SCK 第 1 个周期开始前变为低，并保持到第 8 个周期结束。
- 当 /SS 为高时忽略 SDI、SDO 上的数据。
- 时序如图 12、13 所示。

4) 相关寄存器编程

SPI 操作相关寄存器如表 2，表 3 所示。

表 2 SPI 模式相关控制寄存器

地址	名称	7	6	5	4	3	2	1	0
0X0D	*SPIC/RD	ES/0	SPIE/0	SRO/0	SPISE/0	0/0	SBR2/0	SBR1/0	SBR0/0
0X0F	INTC/IOCF	0	0	0	0	T1IE/0	SPIIE/0	EXIE/0	TCIE/0

注：*位名/初始值

表 3 SPI 模式相关状态/数据寄存器

地址	名称	7	6	5	4	3	2	1	0
0X0A	SPIRB/RA	SRB7	SRB6	SRB5	SRB4	SRB3	SRB2	SRB1	SRB0
0X0B	SPIWB/RB	SWB7	SWB6	SWB5	SWB4	SWB3	SWB2	SWB1	SWB0
0X0C	SPIS/RC	0	0	0	T1IF	OD3/0	OD4/0	RBFIF/0	RBF/0

SPIRB:SPI 读缓冲器。串行数据一旦接收完毕，将从 SPISR 寄存器移至 SPIRB 寄存器，RBF 和 RBFIF 置 1。

SPIWB:SPI 写缓冲器。数据转入后从 SPIS 寄存器输出。

SPIC:SPI 控制寄存器。

表 4 为上电复位其初始值。

表 4 SPIC 寄存器初始值

SPIC	7	6	5	4	3	2	1	0
名称	CES	SPIE	SRO	SSE	0	SBR2	SBR1	SBR0
初始值	0	0	0	0	0	0	0	0

- ◆ CES:时钟沿选择
1=数据在下降沿移出，上升沿移进。数据在高电平期间保持。
0=数据在上升沿一处，下降沿移进。数据在低电平期间保持。
- ◆ SPIE:SPI 模式使能位
1=SPI 模式使能
0=SPI 模式禁止
- ◆ SRO:SPI 读溢出位
1=SPIRB 中的数据还未被读出又已完成新数据的接收。在这种情况下，SPISR 寄存器中的数据将丢失。
0=未溢出。
注：这仅在从模式下发生。
- ◆ SSE:SPI 输出使能
1=开始传输，保持为 1 直至当前字节传输结束。
0=传输结束，下一字节准备传输。
注：该位由软件清 0。

表 5 SBR0~SBR2:SPI 波特率选择位

SBR2	SBR1	SBR0	模式	波特率
0	0	0	主	Fosc/2
0	0	1	主	Fosc/4
0	1	0	主	Fosc/8
0	1	1	主	Fosc/16
1	0	0	主	Fosc/32
1	0	1	从	/SS 使能
1	1	0	从	/SS 禁止
1	1	1	主	TMR1/2

SPIS:SPI 状态寄存器

- ◆ RBFIF:读缓冲器满中断标志
1=接收结束，SPIRB 满，产生中断（若使能）
0=接收未结束，SPIRB 空
- ◆ RBF:读缓冲器满标志
1=接收结束，SPIRB 满
0=接收未结束，SPIRB 空
- ◆ OD3:P93 漏极开路控制
1=SDO 漏极开路使能
0=SDO 漏极开路禁止
- ◆ OD4:P94 漏极开路控制
1=SCK 漏极开路使能
0=SCK 漏极开路禁止
- ◆ SPIIE:SPI 中断使能控制
1=SPI 中断使能
0=SPI 中断禁止

5) SPI 模式时序

SCK 边沿由 CES 选择。对于主模式和 /SS 禁止的从模式，图 12 所示波形均适用。图 13 波形仅适用于 /SS 使能的从模式。

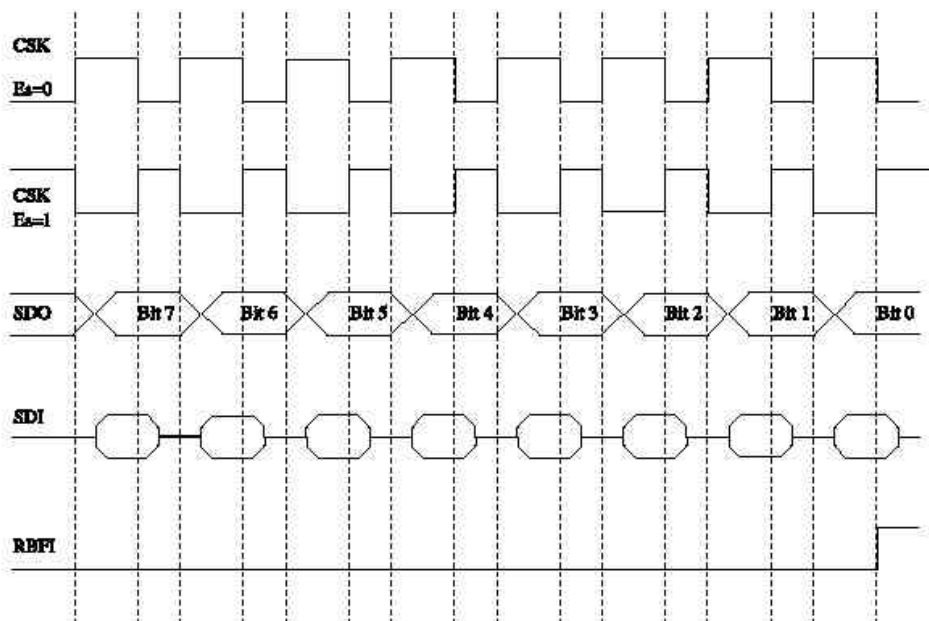


图12 /SS禁止下的SPI模式时序

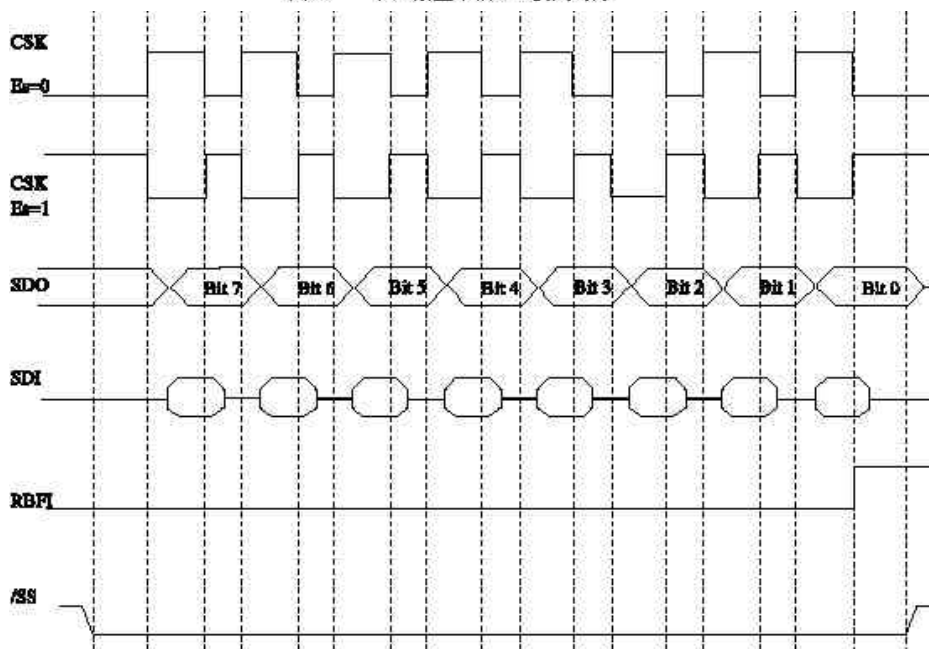


图13 /SS使能的SPI模式时序

6) SPI 软件应用

SPI 范例:

; 定义 RAM

R_0	==0	; 间址寄存器
PSW	==3	; 状态寄存器
RSR	==4	; RAM 选择寄存器
P9	==9	; 端口 9
SPIRB	==0XA	; SPI 读缓冲器
SPIWB	==0XB	; SPI 写缓冲器
SPIS	==0XC	; SPI 状态寄存器
SPIC	==0XD	; SPI 控制寄存器
R_3F	==0X3F	; 中断状态寄存器
I_A	==0X13	; 中断期间累加器保存



```

I_PSW ==0X14           ; 中断期间状态寄存器保存
;定义控制寄存器
C_P9  ==0X9            ; 端口 9 控制寄存器
C_INT ==0XF            ; 中断控制寄存器
;定义位
RUN   ==4              ; SPIC(4)
;定义引脚
SS    ==5              ; P95

INT_VECT:
    ORG 0X02           ;中断子程初始地址
    MOV     I_A,A       ; 保存 ACC
    SWAP    I_A
    SWAPA   PSW
    MOV     I_PSW,A     ; 保存 PSW
    BS      P9,5        ; /SS 置 1, 禁止 SPI
    ;如下 3 位必须按顺序清 0
    BC      SPIS,0      ; RBF=0
    BC      SPIS,1      ; RBFIF=0
    BC      R_3F,2      ; SPIIE=0
    ... ..             ; 用户程序
    SWAPA   I_PSW
    MOV     PSW,A       ; 恢复 PSW
    SWAPA   I_A         ; 恢复 ACC
    RETI

SPI:
    DISI                    ; 中断禁止
    JBS     RX,BX          ; 模式选择
    JMP     SLAVE

MASTER:
    MOV     A,@0BXXXXX1XX ; SDI 输入
    IOW     P9
    MOV     A,@0B01000000 ; 主模式 SPIC 初始值
    MOV     SPIC,A

SLAVE:
    MOV     A,@0BXX1XX1XX  ???????
    IOW     P9
    MOV     A,@0B01000101 ; 从模式 SPIC 初始值
    MOV     SPIC,A
    ; 系统初始化
    MOV     A,@0B00000111
    CONTW
    MOV     A,@0B00000100
    IOW     C_INT          ; 使能 SPI 中断
    CLR     R_3F           ; 清所有中断标志
    BS      P9,5           ; /SS 置 1, 禁止 SPI
    ENI                    ; 使能中断
    ... ..                 ; 用户程序
    DISI

    JBC     SPIC,RUN        ; 检测 RUN 为 0 否
    JMP     STARTTORUN
    BS      SPIC,RUN        ; RUN 置 1

```

STARTTORUN:

```

    ENI                                ; 用户程序
    ....                               ;
    BC      P9,SS                      ; /SS 清 0, SPI 使能
    ....                               ; 用户程序
    MOV     SPIWB,A                    ; 传送值装入 SPIWB
    ....                               ; 用户程序
    BS      SPIC,RUN                   ; 启动 SPI 功能
    ....                               ; 用户程序
    EOP

    ORG 0XFFF                          ; 初始地址
    JMP     SPI

```

6.6 定时器

1) 综述

定时器 1 (TMR1) 是 1 个 8 位时钟计数器, 分频率可编程。它是作为 SPI 模式的波特率时钟发生器设计的。TMR1 可读写, 任何复位情况均清 0。在使用中, 为了降低功耗, 可将 TMR1EN 位清 0 以关闭 TMR1。

2) 功能描述

TMR1 框图如图 14。各信号和功能模块如下述:

OSC/4: 输入时钟

预分频器: 由 T1CLK1, T1CLK2 决定分频率。写 TMR1, T1CON 或任何复位均使其清 0。

PWP: 脉冲宽度预置寄存器。事先写入波特率时钟期望宽度。

TMR1: 定时器 1 寄存器。TMR1 加 1 直至与 PWP 相等, 然后重置为 0。在 SPI 模式中可选择其输出为传输时钟。

比较器: PWP 和 TMR1 相等时改变输出状态。TMR1IF 标志同时置 1。

3) 相关寄存器编程

TMR1 操作相关寄存器如表 6、7 所示。

表 6 TMR1 相关控制寄存器

地址	名称	7	6	5	4	3	2	1	0
0X0C	SPIS/RC	TIP0	T1P10	T1EN	T1IF	OD6	OD4	RBFIF	RBF
0X0F	INTC/IOCF	0	0	0	0	T1IE	SPIIE	EXIE	TCIE

表 7 TMR1 相关状态/数据寄存器

地址	名称	7	6	5	4	3	2	1	0
0X0E	TMR1/RE	TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10
0X0F	PWP/RF	PWP7	PWP6	PWP5	PWP4	PWP3	PWP2	PWP1	PWP0
0X0C	T1CON/IOCC	0	0	0	0	0	T1E	T1P1	T1P0

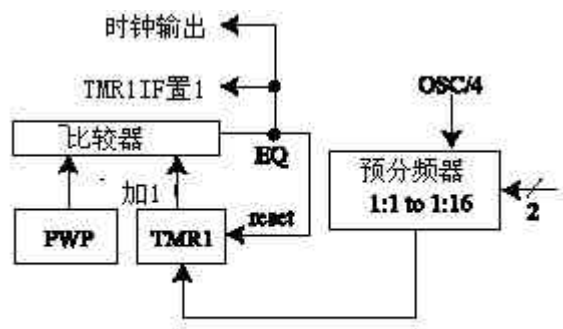


图 14 TMR1框图

表 8 TMR1 分频率

T1P1	T1P0	分频率
0	0	1: 1
0	1	1: 4
1	0	1: 8
1	1	1: 16

6.7 复位与唤醒

复位由下面情况引起：

- 上电复位
- WDT 溢出（若使能）

检测到复位后，系统将保持复位状态 18ms（振荡器起振时间）。一旦复位发生，单片机系统处于如下状态：

- 振荡器继续振荡或起振。
- PC 置为全 1。
- 所有 I/O 引脚定义为输入模式（高阻状态）。
- WDT 和预分频器清 0。
- 上电时，R3 第 5、6 位和 R4 高 2 位清 0。
- CONT 寄存器除第 6 位（INT 标志）外，全置为 1。
- IOCE 寄存器第 0、4、5 位置 1，第 3、6 位清 0。
- R3F、IOCF 寄存器第 0 位清 0。

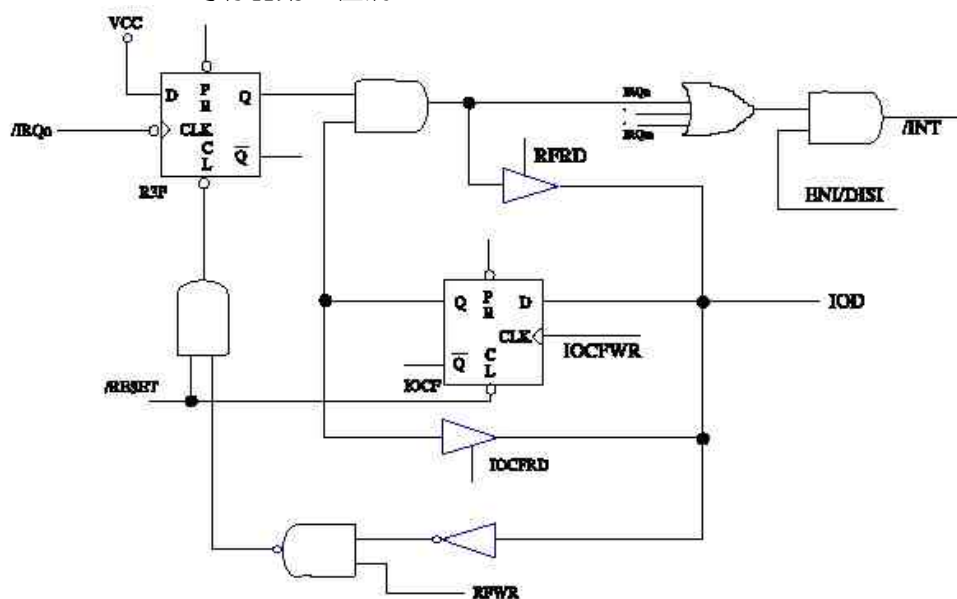


图15 中断输入电路

执行 SLEEP 指令可进入休眠模式 1（低功耗模式）。进入休眠模式时，WDT（若使能）清 0 但继续运行。单片机可被 WDT 溢出（若使能）唤醒并复位。R3 的 T、P 标志可用于确定复位源。

除了基本的休眠模式 1 外，EM78P451 还有一种休眠模式 2，由清 IOCE 的 SLPC 位进入。在休眠模式 2，单片机可被下述情况唤醒：

- 1) 输入触发，如图 15 所示。唤醒后单片机执行下条指令。在进入休眠模式 2 之前，唤醒功能的触发源（P60~P67、P74~P75、P90~P91）需要进行设置（输入状态，上拉），并使能唤醒。值得注意的是，唤醒后若代码选项寄存器 WTC 为 1，则 WDT 使能。因此，WDT 的状态在唤醒后可能需要重新设置。
- 2) WDT 溢出（若使能）。唤醒时引起复位。

6.8 中断

EM78P451 有如下 4 种情况可引起中断：

- (1) TCC 溢出。
- (2) 外部中断信号。
- (3) SPI 结束。
- (4) 可重装计数器匹配。

R3F，中断标志寄存器，在相应位记录了中断请求情况。IOCF 位中断屏蔽寄存器。整体的中断使能或禁止由 ENI 或 DISI 指令完成。当中断发生时，下一指令由地址 0X01 取出。一旦进入中断处理程序，可轮流检测 R3F 寄存器来确定中断源。退出中断处理子程序前，必须清中断标志并使能中断以免重复中断。

不管是否允许中断，R3F 寄存器的相应位会由中断置位。读 RF 的结果是 R3F 和 IOCF 的逻辑与。参见图 15，RETI 指令结束中断子程并使能中断。

当 INT 指令（若使能）产生中断时，下一指令将从地址 0X002 取出。

6.9 指令集

见相关指令集章节

附录一 绝对最大范围

Items	Sym.	Condition	Rating
Temperature under bias	T_{OPR}		0°C to 70°C
Storage temperature	T_{STR}		-65°C to 150°C
Input voltage	V_{IN}		-0.3V to +6.0V
Output voltage	V_O		-0.3V to +6.0V

附录二 直流电气特性（Ta=0°C~70°C,VDD=5V,VSS=0V）

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
IIL	Input Leakage Current	VIN = VDD, VSS			±1	μA
VIH	Input High Voltage		2.0			V
VIL	Input Low Voltage				0.8	V
VIHX	Clock Input High Voltage	OSCI	3.5			V
VILX	Clock Input Low Voltage	OSCI			1.5	V
VOH1	Output High Voltage (ports 5,6,8,9 and P74~P77)	IOH = -12.0mA	2.4			V
VOH2	Output High Voltage (P70~P72)(S7=0)	IOH = -10.0mA		2		V
VOH3	Output High Voltage (P70~P72)(S7=1)	IOH = -10.0mA	2.4			V
VOL1	Output Low Voltage (port5,6,8,9 and P74~P75)	IOL = 5.0mA			0.4	V
VOL2	Output Low Voltage (P70~P72)(S7=0)	IOL = 12.0mA			0.4	V
VOL3	Output Low Voltage (P70~P72)(S7=1), P76~P77)	IOL = 10.0mA		3		V
IPH	Pull-high current	Pull-high active, input pin at VSS	-250	-400	-500	μA
ISB	Power-down current	All input and I/O pin at VDD,output pin floating, WDT enabled			10	μA
ICC1	Operating supply current (VDD=5.0V) at two cycles/two clocks	/RESET = High, Fosc=1.84324MHz(CK2="0"), output pin floating			3	mA

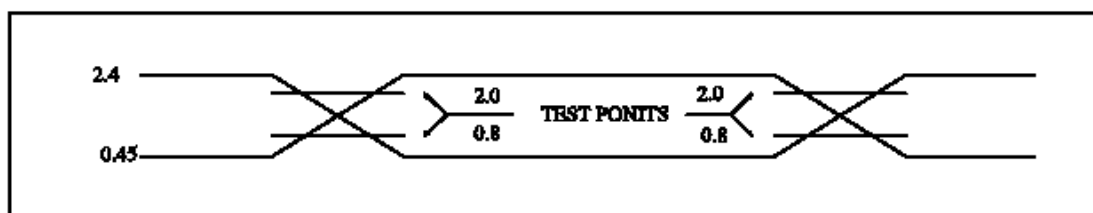
附录三 交流电气特性 (Ta=0℃~70℃, VDD=5V, VSS=0V)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
Delk	Input CLK duty cycle		45	50	55	%
Tins	Instruction cycle time (CLK="0")	RC Type 3.679M	500		DC	ns
Ttcc	TCC input period		(Tins+20)/N*			ns
Twdt	Watchdog Timer period	Ta = 25°C		18		ms
Tdrh	Device reset hold period	Ta = 25°C		18		ms

Note : N* = selected prescaler ratio.

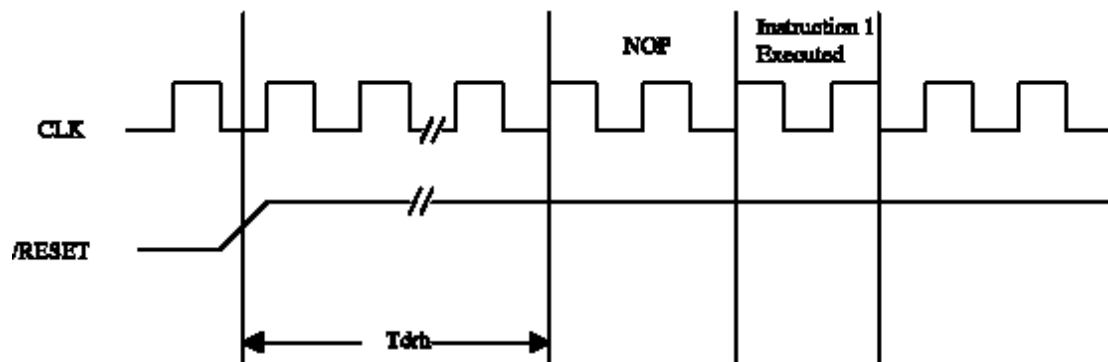
附录四 时隙图

交流输入/输出波形

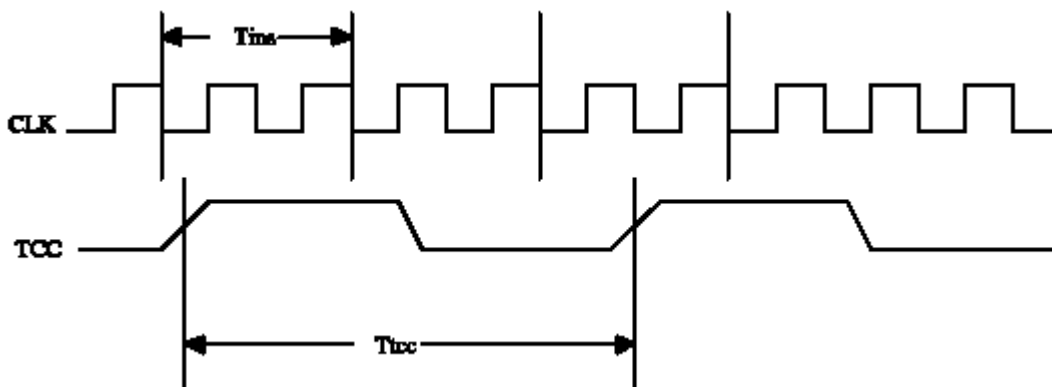


AC Testing : Input is driven at 2.4V for logic "1", and 0.45V for logic "0". Timing measurements are made at 2.0V for logic "1", and 0.8V for logic "0".

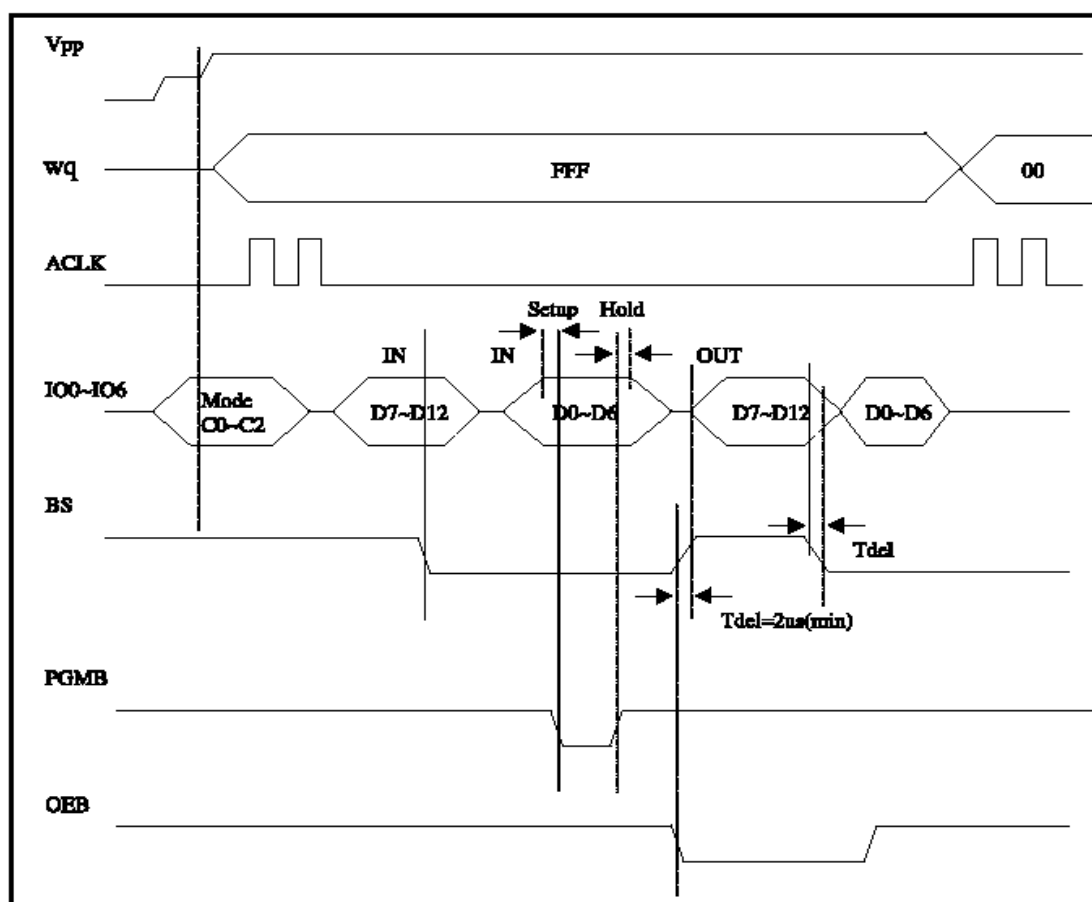
复位时隙 (CLK= "0")



TCC 输入时隙 (CLK= "0")

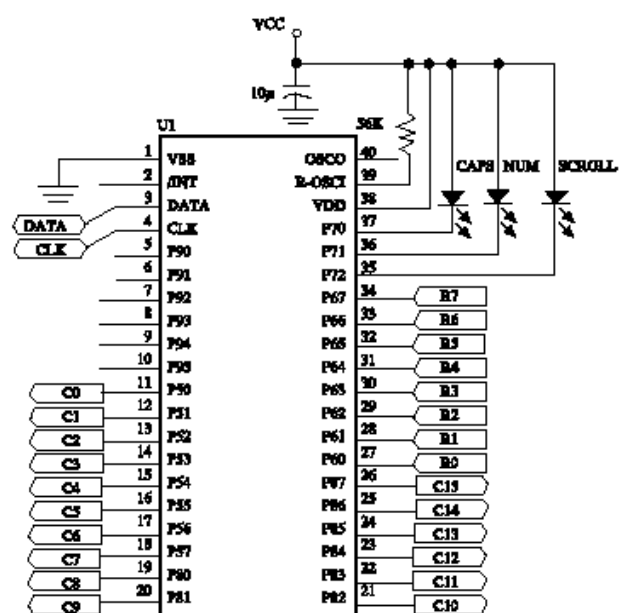


正常和选择模式波形



Attention : 1. Setup & Hold time of data-in D0~D6
2. ACLK

附录五 应用电路



EM78P451

	R0	R1	R2	R3	R4	R5	R6	R7
C0			098				064	
C1		044					037	
C2	110	016	001	002	017	031	046	
C3	045	030	112	003	018	032	047	
C4	113	114	113	004	019	033	048	
C5	095	021	006	005	020	034	049	050
C6	096	022	007	008	023	035	051	
C7		115	119	010	024	036	052	
C8	116	015	120	121		029	043	061
C9	117	028	013	009	024	038	053	
C10	041	027	012	011	026	040	042	055
C11		092	076	122	091	083	090	084
C12	099	097	075	123	096	096	095	089
C13	104	102	085	086	101	103	100	105
C14	083		080	081	016	108	126	079
C15	060			124	125			062



附录六 应用软件（SPI 收发）

#这个演示程序用 EMC++ 编写

```
#EM45
;DEFINE CPU RAM
R_0      == 0      ;DEFINE BIT
TCC      == 1      #DEFINE B_KEY R_BIT.0
PC       == 2      #DEFINE SPI_RUN SPIC.4
PSW      == 3
RSR      == 4      #DEFINE CY PSW.0
P5       == 5      #DEFINE DC PSW.1
P6       == 6      #DEFINE Z PSW.2
P7       == 7      #DEFINE D6 RSR.6
P8       == 8      #DEFINE D7 RSR.7
P9       == 9
SPIRB    == 0XA    #DEFINE P_IN P6
SPIWB    == 0XB    #DEFINE P_OUT P5
SPIS     == 0XC
SPIC     == 0XD
TMR1     == 0XE    ;DEFINE PROT

PWP      == 0XF    #DEFINE P_START P9.0

R_3F     == 0X3F   #DEFINE P_MS_SEL P9.1
                        #DEFINE P_SPI_SS P9.5

R_ACC    == 0X10
R_ACC2   == 0X11    ORG 0
R_ACC3   == 0X12
I_A      == 0X13    MAIN()
I_PSW    == 0X14
I_RSR    == 0X15    ORG 1
I_ACC    == 0X16    NOP
I_A=A
R_BIT    == 0X17    SWAP I_A
R_CODE   == 0X18    SWAPA PSW
                        I_PSW=A
C_P5     == 0X5     I_RSR=RSR
C_P6     == 0X6     I_ACC=R_ACC
C_P7     == 0X7
C_P8     == 0X8     PSW=0
C_P9     == 0X9     RSR=0
C_T1CON  == 0XC
C_HIGH   == 0XD     P8++
C_WDT    == 0XE
```



```
C_INT      == 0XF

                                ENDIF

P_OUT=SPIRB
P_SPI_SS=1      !OPTION=0B111
                                !C_HIGH=0
                                !C_WDT=0B00110001
SPIS.0=0        !C_INT=0B00000100
SPIS.1=0
R_3F.2=0

                                PSW=0
                                RSR=0
                                R_3F=0

RSR=I_RSR
R_ACC=I_ACC      RSR=0X10
SWAPA I_PSW      DO
PSW=A            R_0=0
SWAPA I_A        D6=1
                                R_0=0
RETI              D7=1
                                R_0=0
                                D6=0
                                R_0=0

MAIN() D7=0

                                WHILE++RSR!=0X40
DISI              RSR=0
P5=0              P_SPI_SS=1
!P5=0
P6=0              ENI
!P6=0XFF
P7=0              LOOP:
!P7=0              WDTC
P8=0
!P8=0
P9=0              IF P_MS_SEL==0
IF P_MS_SEL      SPIWB=R_CODE
!P9=0B00000111  DISI
SPIC=0B01000100 IF SPI_RUN==0
ELSE              IF P_SPI_SS==0
!P9=0B00100111  R_CODE++
SPIC=0B01000101
ENDIF
```



```

                                SPI_RUN=1
                                ENDIF
                                ENDIF
                                ENI
                                ENDIF

IF P_START==0
    IF B_KEY==0
        B_KEY=1
        IF SPI_RUN==0
            P_SPI_SS=0
            SPIWB=0xAA
            FOR R_ACC=1,R_ACC!=0,R_ACC++
                NOP
                NOP
            NEXT
            SPI_RUN=1
        ENDIF
    ENDIF
ELSE
    B_KEY=0
ENDIF
GOTO LOOP
EOP
ORG 0XFFF
JMP MAIN
```

目 录

一 概括描述	4-1
二 功能特点	4-1
三 引脚分配	4-2
四 功能框图	4-2
五 引脚描述	4-2
六 功能描述	4-3
6.1 操作寄存器	4-3
6.2 特殊功能寄存器	4-5
6.3 TCC/WDT 预分频器	4-8
6.4 I/O 端口	4-8
6.5 串行外围接口模式	4-9
6.6 定时器	4-15
6.7 复位与唤醒	4-16
6.8 中断	4-17
6.9 指令集	4-17
附录一 绝对最大范围	4-17
附录二 直流电气特性 (Ta=0℃~70℃, VDD=5V, VSS=0V)	4-17
附录三 交流电气特性 (Ta=0℃~70℃, VDD=5V, VSS=0V)	4-18
附录四 时隙图	4-18
附录五 应用电路	4-19
附录六 应用软件 (SPI 收发)	4-20