

## P89LV51RD2 器件手册

### 1. 概述

P89LV51RD2 是一款 80C51 微控制器，包含 64kB Flash 和 1024 字节的数据 RAM。

P89LV51RD2 的典型特性是它的 X2 方式选项。设计者可通过该特性来选择应用程序以传统的 80C51 时钟频率（每个机器周期包含 12 个时钟）或 X2 方式（每个机器周期包含 6 个时钟）的时钟频率运行，其中，选择 X2 方式可在相同时钟频率下获得 2 倍的吞吐量。从该特性获益的另一种方法是将时钟频率减半来保持特性不变，这样可以极大地降低 EMI。

Flash 程序存储器支持并行和串行在系统编程（ISP）。并行编程方式提供了高速的分组编程（页编程）方式，可节省编程花费和推向市场的时间。ISP 允许在软件控制下对成品中的器件进行重复编程。应用固件的产生/更新能力实现了 ISP 的大范围应用。

P89LV51RD2 也可采用在应用中编程（IAP），允许随时对 Flash 程序存储器重新配置，即使应用程序正在运行时也不例外。

### 2. 特性

- 80C51 核心处理单元；
- 3V 的工作电压，操作频率为 0~33MHz；
- 64kB 的片内 Flash 程序存储器，具有 ISP（在系统编程）和 IAP（在应用中编程）功能；
- 通过软件或 ISP 选择支持 12 时钟（默认）或 6 时钟模式；
- SPI（串行外围接口）和增强型 UART；
- PCA（可编程计数器阵列），具有 PWM 和捕获/比较功能；
- 4 个 8 位 I/O 口，含有 3 个高电流 P1 口（每个 I/O 口的电流为 16mA）；
- 3 个 16 位定时器/计数器；
- 可编程看门狗定时器（WDT）；
- 8 个中断源，4 个中断优先级；
- 2 个 DPTR 寄存器；
- 低 EMI 方式（ALE 禁能）；
- 兼容 TTL 和 CMOS 逻辑电平；
- 掉电检测；
- 低功耗模式
  - ◆ 掉电模式，外部中断唤醒；
  - ◆ 空闲模式；
- PDIP40, PLCC44 和 TQFP44 的封装；

### 3. 订购信息

表 1 订购信息

类型号	封装		
	名称	描述	版本
P89LV51RD2BA	PLCC44	塑料行间芯片运载，44 脚	SOT187-2
P89LV51RD2FA			
P89LV51RD2BBC	TQFP44	塑料小型方块扁平封装；44 脚	SOT376-1
P89LV51RD2BN	PDIP40	塑料双列直插；40 脚	SOT129-1

### 3.1 订购选项

表 2 订购选项

类型号	温度范围	频率
P89LV51RDBA	0°C~+70°C	0~33MHz
P89LV51RD2FA	-40°C~+85°C	
P89LV51RD2BBC	0°C~+70°C	
P89LV51RD2BN	0°C~+70°C	

### 4. 功能框图

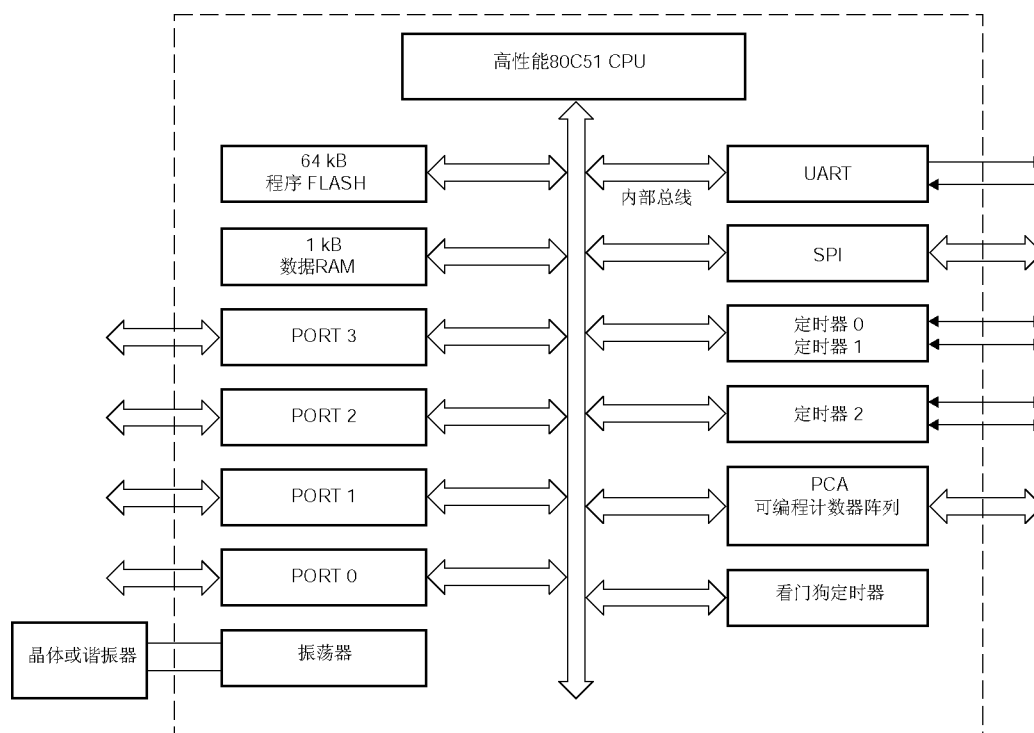


图 1 P89LV51RD2 功能框图

## 5. 管脚信息

### 5.1 管脚

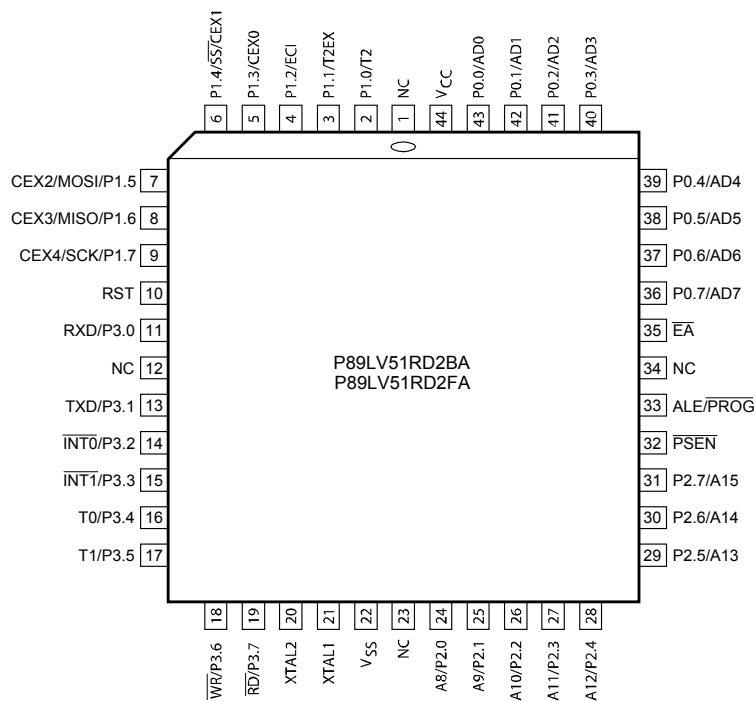


图2 PLCC44 管脚配置

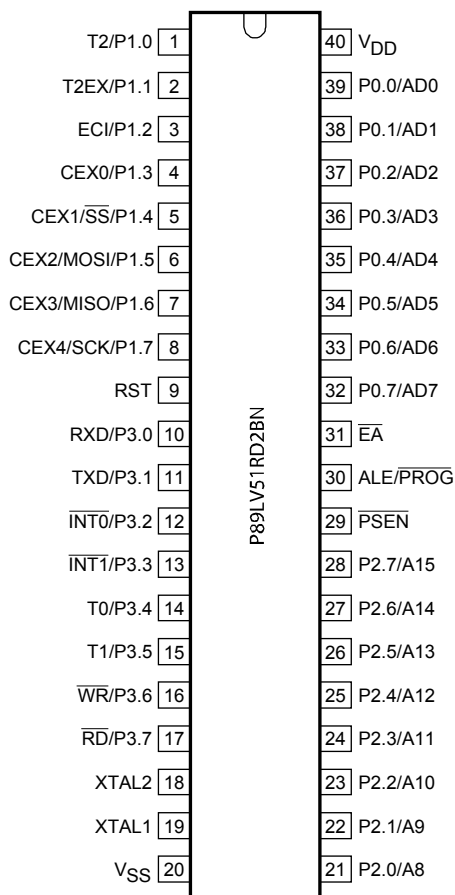


图3 PDIP40 管脚配置

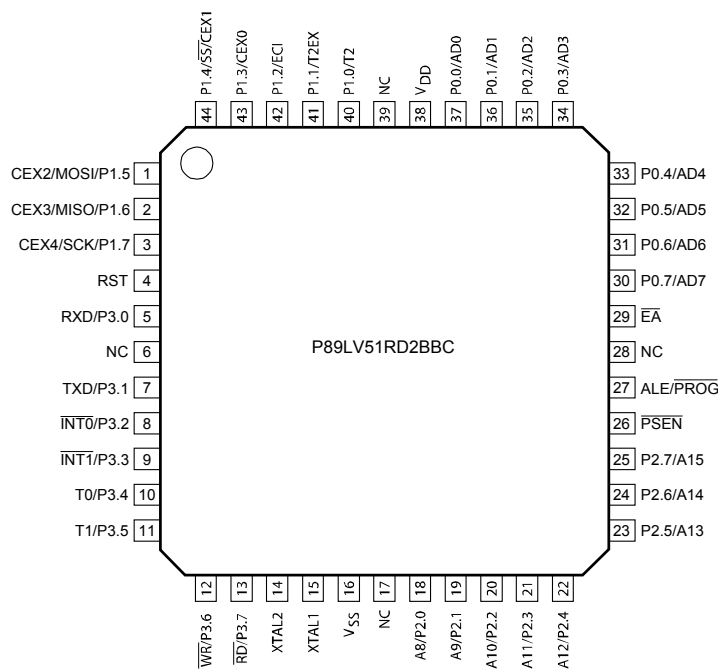


图4 TQFP44 管脚配置

## 5.2 管脚描述

表3 P89LV51RD2 管脚描述

符号	管脚			类型	描述
	DIP40	TQFP44	PLCC44		
P0.0~P0.7	39~32	37~30	43~36	I/O	<b>P0 口:</b> P0 口是一个开漏双向 I/O 口。写入 ‘1’ 时 P0 口悬浮, 可用作高阻态输入。当访问外部程序和数据存储器时, P0 口复用为低位地址和数据总线。应用中 P0 口利用强内部上拉来发送 ‘1’ 电平。P0 口可在外部主机模式编程过程中接收代码字节和在外部主机模式校验过程中发送代码字节。P0 口用作程序校验或通用 I/O 口时均需连接一个外部上拉电阻。
P1.0~P1.7	1~8	40~44, 1~3	2~9	I/O 带内 部上 拉	<b>P1 口:</b> P1 口是一个带内部上拉的 8 位双向口。写入 ‘1’ 时 P1 口被内部上拉拉高, 可用作输入。用作输入时, 由于内部上拉的存在, P1 口被外部器件拉低时将吸收电流 ( $I_{IL}$ )。此外, P1.5, P1.6, P1.7 还有 16mA 的高电流驱动能力。在外部主机模式编程和校验中, P1 口也可接收低位地址字节。
P1.0	1	40	2	I/O	<b>T2:</b> 定时器/计数器 2 的外部计数输入或定时器/计数器 2 的时钟输出。
P1.1	2	41	3	I	<b>T2EX:</b> 定时器/计数器 2 捕获/重装触发和方向控制。
P1.2	3	42	4	I	<b>ECI:</b> 外部时钟输入。PCA 的外部时钟输入。

续上表.....

符号	管脚			类型	描述
	DIP40	TQFP44	PLCC44		
P1.3	4	43	5	I/O	<b>CEX0</b> : PCA 模块 0 的捕获/比较外部 I/O 口。每个捕获/比较模块连接一个 P1 口用作外部 I/O 口。该口线不被 PCA 占用时仍可用作标准 I/O 口。
P1.4	5	44	6	I/O	$\overline{\text{SS}}$ : SPI 从机选择输入。 <b>CEX1</b> : PCA 模块 1 的捕获/比较外部 I/O 口。
P1.5	6	1	7	I/O	<b>MOSI</b> : SPI 主机输出从机输入端。 <b>CEX2</b> : PCA 模块 2 的捕获/比较外部 I/O 口。
P1.6	7	2	8	I/O	<b>MISO</b> : SPI 主机输入从机输出端。 <b>CEX3</b> : PCA 模块 3 的捕获/比较外部 I/O 口。
P1.7	8	3	9	I/O	<b>SCK</b> : SPI 主机输出从机输入端。 <b>CEX4</b> : PCA 模块 4 的捕获/比较外部 I/O 口。
P2.0~P2.7	21~28	18~25	24~31	I/O 带内 部上 拉	<b>P2 口</b> : P2 口是一个带内部上拉的 8 位双向口。写入 '1' 时 P2 口被内部上拉拉高, 可用作输入。用作输入时, 由于内部上拉的存在, P2 口被外部器件拉低时将吸收电流 ( $I_{IL}$ )。在取指外部程序存储器或访问 16 位地址 ( <b>MOVX @DPTR</b> ) 的外部数据存储器时, P2 口发送高位地址。应用中 P2 口利用强内部上拉来发送 '1'。在外部主机模式编程和校验中, P2 口可接收一些控制信号和部分高地址位。
P3.0~P3.7	10~17	5,7~13	11,13~19	I/O 带内 部上 拉	<b>P3 口</b> : P3 口是一个带内部上拉的 8 位双向口。写入 '1' 时 P3 口被内部上拉拉高, 可用作输入。用作输入时, 由于内部上拉的存在, P3 口被外部器件拉低时将吸收电流 ( $I_{IL}$ )。在外部主机模式编程和校验中, P3 口可接收一些控制信号和部分高地址位。
P3.0	10	5	11	I	<b>RxD</b> : 串口输入。
P3.1	11	7	13	O	<b>TxD</b> : 串口输出。
P3.2	12	8	14	I	$\overline{\text{INT0}}$ : 外部中断 0 输入
P3.3	13	9	15	I	$\overline{\text{INT1}}$ : 外部中断 1 输入
P3.4	14	10	16	I	<b>T0</b> : 定时器/计数器 0 的外部计数输入。
P3.5	15	11	17	I	<b>T1</b> : 定时器/计数器 1 的外部计数输入。
P3.6	16	12	18	O	$\overline{\text{WR}}$ : 外部数据存储器写选通信号。
P3.7	17	13	19	O	$\overline{\text{RD}}$ : 外部数据存储器读选通信号。

续上表.....

符号	管脚			类型	描述
	DIP40	TQFP44	PLCC44		
$\overline{\text{PSEN}}$	29	26	32	I/O	<b>程序选通使能:</b> $\overline{\text{PSEN}}$ 是外部程序存储器的读选通信号。 $\overline{\text{PSEN}}$ 在执行内部程序存储器的程序时无效(高电平), 执行外部程序存储器时每个机器周期内两次有效, 但当访问外部数据存储器时两个有效 $\overline{\text{PSEN}}$ 脉冲将被跳过。当 RST 输入引脚的高电平时间大于 10 个机器周期时, 向 $\overline{\text{PSEN}}$ 脚强制输入一个高电平到低电平的跳变将使器件进入外部主机模式编程。
RST	9	4	10	I	<b>复位:</b> 振荡器工作时, 该引脚上 2 个机器周期的高电平逻辑状态将使器件复位。当 RST 输入引脚为高电平时, 如果 $\overline{\text{PSEN}}$ 脚输入一个高电平到低电平的跳变, 器件将进入外部主机模式。否则进入正常工作模式。
$\overline{\text{EA}}$	31	29	35	I	<b>外部访问使能:</b> 若器件要对外部程序存储器取指, $\overline{\text{EA}}$ 就必须与 $V_{\text{SS}}$ 相连。器件执行内部程序存储器的程序时 $\overline{\text{EA}}$ 必须与 $V_{\text{DD}}$ 相连。然而, 4 个安全锁定电平可将 $\overline{\text{EA}}$ 禁能, 使器件只能执行内部程序存储器的程序。 $\overline{\text{EA}}$ 脚可承受 12V 的高压。
$\overline{\text{ALE/PROG}}$	30	27	33	I/O	<b>地址锁存使能:</b> ALE 是一个输出信号, 在访问外部存储器时将地址低字节锁存。该引脚也用作 Flash 的编程脉冲输入 (PROG)。通常, ALE <sup>[1]</sup> 在 1/6 的振荡频率 <sup>[2]</sup> 时输出, 可用作外部定时或外部时钟。每次访问外部数据存储器时都有一个 ALE 脉冲被跳过。但是, 只要 AO 被置 1, ALE 就被禁能。
NC	-	6,17,28,39	1,12,23,34	I/O	不连
XTAL1	19	15	21	I	<b>晶振 1:</b> 反相振荡放大器的输入和内部时钟发生电路的输入。
XTAL2	18	14	20	O	<b>晶振 2:</b> 反相振荡放大器的输出。
$V_{\text{DD}}$	40	38	44	I	电源
$V_{\text{SS}}$	20	16	22	I	地

[1] ALE 负载: 如果复位时 ALE 脚驱动更大的负载 (>30pF), 微控制器可能进入正常工作模式以外的其它工作模式。解决的方法是在管脚上 (如 ALE 脚) 增加一个连接到  $V_{\text{DD}}$  的 3k $\Omega$ ~50k $\Omega$  的上拉电阻。

[2] 6 时钟模式下, ALE 信号在 1/3 振荡频率时输出。

## 6. 特殊功能寄存器

备注：对特殊功能寄存器（SFR）的访问必须遵循以下方式：

- 用户不要试图访问任何未经定义的 SFR 地址。
- 对任何已定义的 SFR 的访问必须符合 SFR 的功能。
- 标注为 ‘-’，‘0’ 或 ‘1’ 的 SFR 位只能以如下方式读或写：
  - ‘-’ 必须写入 0，但当读出时不返回任何确定的值（即使向其写入 0）。这是一个保留位，作为将来功能扩展之用。
  - ‘0’ 必须写入 0，并且当读出时返回 0。
  - ‘1’ 必须写入 1，并且当读出时返回 1。

表 4 特殊功能寄存器

\*表明寄存器可进行位寻址

名称	描述	SFR 地址	位功能和地址							
			MSB				LSB			
位地址			E7	E6	E5	E4	E3	E2	E1	E0
ACC*	累加器	E0H								
AUXR	辅助功能寄存器	8EH	-	-	-	-	-	-	EXTRAM	AO
AUXR1	辅助功能寄存器 1	A2H	-	-	-	-	GF2	0	-	DPS
位地址			F7	F6	F5	F4	F3	F2	F1	F0
B*	B 寄存器	F0H								
CCAP0H	模块 0 捕获高字节	FAH								
CCAP1H	模块 1 捕获高字节	FBH								
CCAP2H	模块 2 捕获高字节	FCH								
CCAP3H	模块 3 捕获高字节	FDH								
CCAP4H	模块 4 捕获高字节	FEH								
CCAP0L	模块 0 捕获低字节	EAH								
CCAP1L	模块 1 捕获低字节	EBH								
CCAP2L	模块 2 捕获低字节	ECH								
CCAP3L	模块 3 捕获低字节	EDH								
CCAP4L	模块 4 捕获低字节	EEH								
CCAPM0	模块 0 模式	DAH	-	ECOM_0	CAPP_0	CAPN_0	MAT_0	TOG_0	PWM_0	ECCF_0
CCAPM1	模块 1 模式	DBH	-	ECOM_1	CAPP_1	CAPN_1	MAT_1	TOG_1	PWM_1	ECCF_1
CCAPM2	模块 2 模式	DCH	-	ECOM_2	CAPP_2	CAPN_2	MAT_2	TOG_2	PWM_2	ECCF_2
CCAPM3	模块 3 模式	DDH	-	ECOM_3	CAPP_3	CAPN_3	MAT_3	TOG_3	PWM_3	ECCF_3
CCAPM4	模块 4 模式	DEH	-	ECOM_4	CAPP_4	CAPN_4	MAT_4	TOG_4	PWM_4	ECCF_4
位地址			DF	DE	DD	DC	DB	DA	D9	D8
CCON*	PCA 计数器控制	D8H	CF	CR	-	CCF4	CCF3	CCF2	CCF1	CCF0
CH	PCA 计数器高字节	F9H								
CL	PCA 计数器低字节	E9H								
CMOD	PCA 计数器模式	D9H	CIDL	WDTE	-	-	-	CPS1	CPS0	ECF
DPTR	数据指针（2 字节）									
DPH	数据指针高字节	83H								
DPL	数据指针低字节	82H								

续上表.....

名称	描述	SFR 地址	位功能和地址							
			MSB				LSB			
FST	Flash 状态寄存器	B6H	-	SB	-	-	EDC	-	-	-
		位地址	AF	AE	AD	AC	AB	AA	A9	A8
IEN0*	中断使能 0	A8H	EA	EC	ET2	ES0	ET1	EX1	ET0	EX0
		位地址	EF	EE	ED	EC	EB	EA	E9	E8
IEN1*	中断使能 1	E8H	-	-	-	-	EBO			
		位地址	BF	BE	BD	BC	BB	BA	B9	B8
IP0*	中断优先级	B8H	-	PPC	PT2	PS	PT1	PX1	PT0	PX0
IP0H	中断优先级 0 高字节	B7H	-	PCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
		位地址	FF	FE	FD	FC	FB	FA	F9	F8
IP1*	中断优先级 1	F8H	-	-	-	-	PBO			
IP1H	中断优先级 1 高字节	F7H	-	-	-	-	PBOH			
FCF		B1H	-	-	-	-	-	-	-	BSEL
		位地址	87	86	85	84	83	82	81	80
P0*	P0 口	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
		位地址	97	96	95	94	93	92	91	90
P1*	P1 口	90H	CEX4/ SPICLK	CEX3/ MISO	CEX2/ MOSI	CEX1/ $\overline{SS}$	CEX0	EC1	T2EX	T2
		位地址	A7	A6	A5	A4	A3	A2	A1	A0
P2*	P2 口	A0H	A15	A14	A13	A12	A11	A10	A9	A8
		位地址	B7	B6	B5	B4	B3	B2	B1	B0
P3*	P3 口	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{INT0}$	TxD	RxD
PCON	电源控制寄存器	87H	SMOD1	SMOD0	BOF	POF	GF1	GF0	PD	IDL
		位地址	D7	D6	D5	D4	D3	D2	D1	D0
PSW*	程序状态字	D0H	CY	AC	F0	RS1	RS0	OV	F1	P
RCAP2H	定时器 2 捕获高字节	CBH								
RCAP2L	定时器 2 捕获低字节	CAH								
		位地址	9F	9E	9D	9C	9B	9A	99	98
SCON*	串口控制	98H	SM0/ FE_	SM1	SM2	REN	TB8	RB8	TI	RI
SBUF	串口数据缓冲寄存器	99H								
SADDR	串口地址寄存器	A9H								
SADEN	串口地址使能	B9H								
		位地址	87 <sup>[1]</sup>	86 <sup>[1]</sup>	85 <sup>[1]</sup>	84 <sup>[1]</sup>	83 <sup>[1]</sup>	82 <sup>[1]</sup>	81 <sup>[1]</sup>	80 <sup>[1]</sup>
SPCTL	SPI 控制寄存器	D5H	SPIE	SPEN	DORD	MSTR	CPOL	CPHA	PSC1	PSC0
SPCFG	SPI 配置寄存器	AAH	SPIF	SPWCOL	-	-	-	-	-	-
SPDAT	SPI 数据	86H								
SP	堆栈指针	81H								
		位地址	8F	8E	8D	8C	8B	8A	89	88
TCON*	定时器控制寄存器	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0



续上表.....

名称	描述	SFR 地址	位功能和地址							
			MSB				LSB			
位地址			CF	CE	CD	CC	CB	CA	C9	C8
T2CON*	定时器 2 控制寄存器	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
T2MOD	定时器 2 模式控制	C9H	-	-	ENT2				T2OE	DCEN
TH0	定时器 0 高字节	8CH								
TH1	定时器 1 高字节	8DH								
TH2	定时器 2 高字节	CDH								
TL0	定时器 0 低字节	8AH								
TL1	定时器 1 低字节	8BH								
TL2	定时器 2 低字节	CCH								
TMOD	定时器 0 和 1 模式	89H	GATE	C/T	M1	M0	GATE	C/T	M1	M0
WDTC	看门狗定时器控制	C0H	-	-	-	WDOUT	WDRE	WDTS	WDT	SWDT
WDTD	看门狗定时器数据/重装	85H								

[1] SFR 中的无效位（标有“-”的位）总是为“X”（未知）。除非特别指明，否则不要向这些位写入 1，因为它们可能在将来扩展用于其它功能。这些位的复位值均为 0，读出来的值不可知。

## 7. 功能描述

### 7.1 存储器结构

P89LV51RD2 器件具有独立的程序存储器和数据存储器地址空间。

#### 7.1.1 Flash 程序存储器

P89LV51RD2 器件含有 2 个内部 Flash 存储模块。模块 0 有 64k 字节，用来存放用户代码。模块 1 用来存放 Philips 提供的 ISP/IAP 程序，可以通过使能模块 1 来覆盖用户代码存储器的前 8k 字节。

64kB 的模块 0 可以分成 512 个扇区，每个扇区含有 128 个字节。

通过清零 FCF 寄存器的 BSEL 位来访问 IAP 程序。然而，在动态修改 BSEL 位时要非常小心。由于 BSEL 位的改变将造成不同的物理存储器都映射到逻辑程序地址空间，因此，当在执行 0000H~1FFFH 地址范围内的用户代码时用户不能清零 BSEL 位。

#### 7.1.2 数据 RAM 存储器

数据 RAM 含有 1024 字节的内部存储器。器件也可对高达 64kB 的外部数据存储器进行寻址。

#### 7.1.3 扩展数据 RAM 寻址

P89LV51RD2 有 1kB 的 RAM。见图 5 “内部和外部数据存储器结构”。

器件的内部数据存储器由 4 部分组成：

1. 低 128 字节 RAM (00H~7FH)，可直接和间接寻址。
2. 高 128 字节 RAM (80H~FFH)，间接寻址。
3. 特殊功能寄存器 (80H~FFH)，只可直接寻址。
4. 扩展的 768 字节 RAM (00H~2FFH)，通过移动外部指令 (MOVX) 和清除 EXTRAM 位间接寻址。（见第 6 节“特殊功能寄存器”的辅助寄存器 (AUXR)）

由于高 128 字节和 SFR 占用相同的地址，因此 RAM 空间必须进行间接寻址。尽管 RAM 和 SFR 的地址相同，但它们在物理上是独立的。

表 5 AUXR—辅助寄存器（地址：8EH）的位分配

不可位寻址；复位值为 00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	-	-	-	EXTRAM	AO

表 6 AUXR—辅助寄存器（地址：8EH）的位描述

位	符号	描述
7~2	-	保留为将来之用。通过用户程序将其清零。
1	EXTRAM	使用 MOVX @Ri/@DPTR 来访问内部/外部 RAM。该位为 0 时，CPU 访问 MOVX 指令指定的内部 XRAM 空间。如果指令中提供的地址超出片内 XRAM 的范围，则选择片外 XRAM 空间进行访问。 该位为 1 时，MOVX @Ri/@DPTR 访问默认的外部数据存储器。
0	AO	ALE 断开：禁止/使能 ALE。AO=0 时，在 1/2 振荡器频率时输出 ALE。AO=1 时，ALE 在使用 MOVX 或 MOVX 指令时有效。

当运用指令来访问高 128 字节空间时（7FH 之上），MCU 根据指令类型来决定访问 SFR 或 RAM 空间。如果指令采用间接寻址，访问 RAM；如果指令采用直接寻址，访问 SFR。见以下例子。

间接寻址：

MOV @R0,#data ; R0 的内容是 90H

寄存器 R0 指向高地址范围内的 90H 单元。‘#data’ 数据写入到 90H RAM 单元，而非 P1 口。

直接寻址：

MOV 90H,#data ; 向 P1 口写入数据

#data’ 数据写入到 P1 口。直接写地址单元的指令就是写 SFR。

将 EXTRAM 清零，使用 MOVX 指令来访问扩展 RAM。外部 768 字节的存储器在物理上位于片内，逻辑上占用的是外部存储器的前 768 字节（地址 000H~2FFH）。

当 EXTRAM=0，扩展 RAM 通过 MOVX 指令和选定寄存器区的 R0、R1 寄存器或 DPTR 间接寻址。对扩展 RAM 的访问不会影响 P0 口、P3.6 ( $\overline{WR}$ )、P3.7 ( $\overline{RD}$ ) 或 P2 口。有关 EXTRAM=0 时扩展 RAM 的访问，请参考下面的例子。

扩展 RAM 访问（只能间接寻址）：

MOVX @DPTR,A DPTR 指向 0A0H 单元

DPTR 指向 0A0H 单元，‘A’ 的内容写入到扩展 RAM（而非外部存储器）的 0A0H 地址单元。利用 MOVX 指令来访问高于 2FFH 单元（0300H~FFFFH）的外部存储器，整个过程的执行方法与标准 8051 器件完全相同：P0 和 P2 口用作数据/地址总线，P3.6 和 P3.7 分别用作写和读信号。

当 EXTRAM=1 时，MOVX @Ri 和 MOVX @DPTR 的用法与标准 8051 类似。MOVX @Ri 提供一个 8 位的地址，此时 P0 口还复用为数据总线。由其它输出端口管脚输出高地址位。这就提供了一种外部页功能。MOVX @DPTR 产生一个 16 位地址，允许对高达 64kB 的外部存储器进行访问。由 P2 口提供高 8 位地址（DPH），P0 口复用为低 8 位地址（DPL）总线。MOVX @Ri 和 MOVX @DPTR 均可产生必需的读和写信号（P3.6— $\overline{WR}$ ，P3.7— $\overline{RD}$ ），供外部存储器使用。表 7 所示为 EXTRAM 位状态给定时外部数据存储器的  $\overline{RD}$  和  $\overline{WR}$  操作。

堆栈指针（SP）可以位于内部 RAM（低 128 字节和高 128 字节）256 字节内的任何位置。但不允许位于扩展 RAM 中。

表 7 EXTRAM 位状态给定时外部数据存储器的  $\overline{RD}$  和  $\overline{WR}$  操作

AUXR	MOVX @DPTR,A 或 MOVX A,@DPTR		MOVX @Ri,A 或 MOVX A,@RI
	ADDR<0300H	ADDR≥0300H	ADDR=任何值
EXTRAM=0	$\overline{RD}/\overline{WR}$ 无效	$\overline{RD}/\overline{WR}$ 有效	$\overline{RD}/\overline{WR}$ 无效
EXTRAM=1	$\overline{RD}/\overline{WR}$ 有效	$\overline{RD}/\overline{WR}$ 有效	$\overline{RD}/\overline{WR}$ 有效

[1] 访问限制在 0~0FFH 的 ERAM 地址内；不能访问 100H~02FFH 单元。

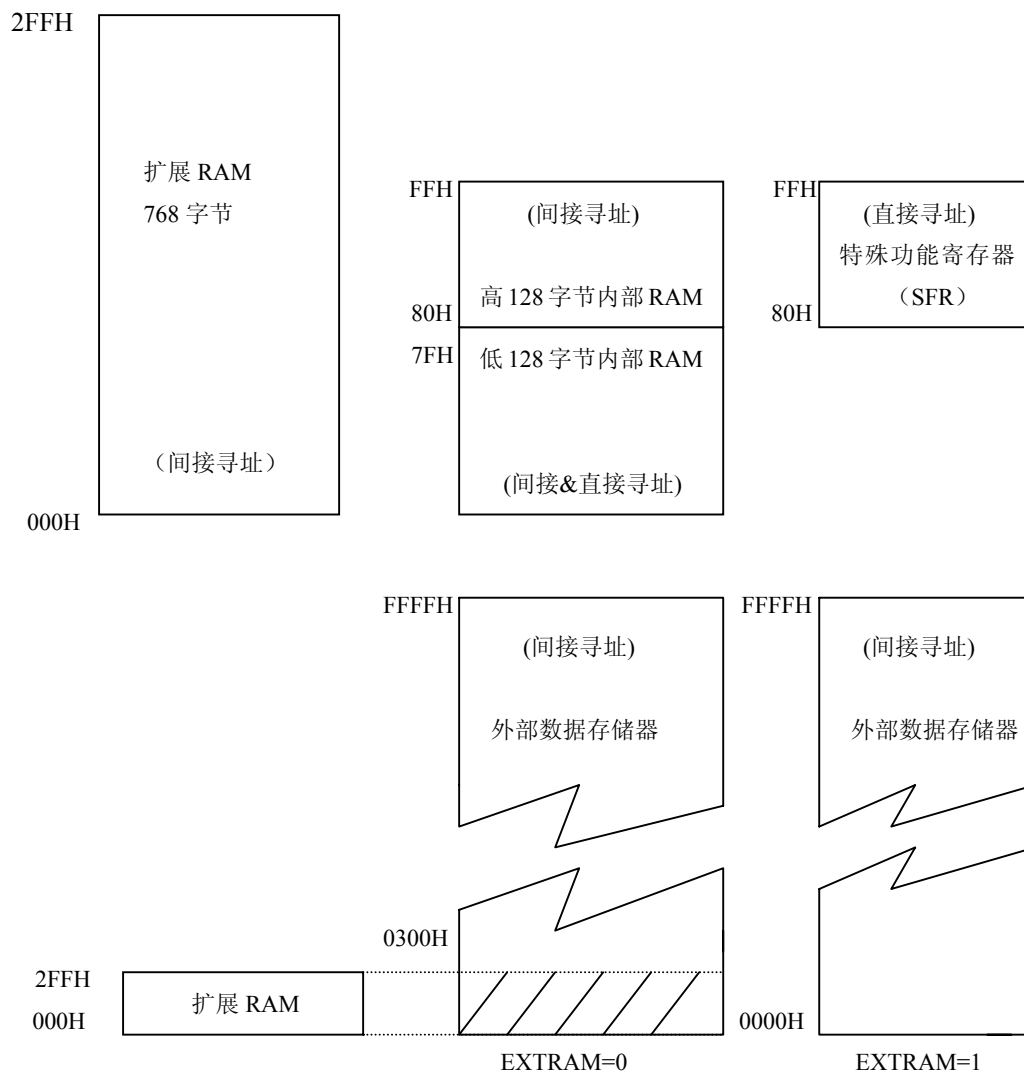


图 5 内部和外部数据存储器结构

#### 7.1.4 双数据指针

P89LV51RD2 器件含有 2 个 16 位的数据指针。AUXR1 寄存器的 DPTR 选择位 (DPS) 用来选择使用的数据指针。DPS=0 时，选择 DPTR0；DPS=1 时，选择 DPTR1。通过对 AUXR1 执行一条简单的 INC 指令就可实现 2 个数据指针的快速切换 (见图 6)。

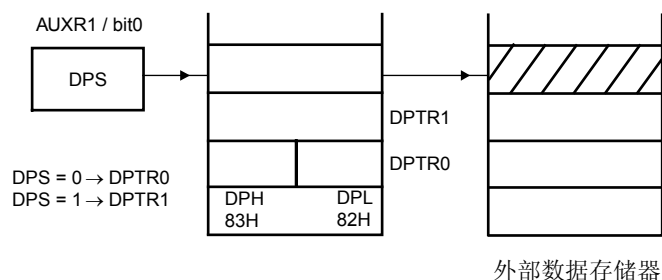


图 6 双数据指针的结构

表 8 AUXR1—辅助寄存器 1 的位分配（地址：A2H）

不可位寻址；复位值为 00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	-	GF2	0	-	DPS

表 9 AUXR1—辅助寄存器 1 的位描述（地址：A2H）

位	符号	描述
7~4	-	保留为将来之用。通过用户程序将其清零。
3	GF2	通用用户定义标志位。
2	0	该位包含一个硬件 0。这样就允许通过将 AUXR1 加 1 来使 DPS 位翻转而不影响寄存器的其它位。
1	-	保留为将来之用。通过用户程序将其清零。
0	DPS	数据指针选择。选择两个数据指针中的一个。详见文中描述。

## 7.2 Flash 存储器在应用中编程

### 7.2.1 Flash 结构

P89LV51RD2 程序存储器包含一个 64kB 的块。第二个 8kB 块中的在系统编程（ISP）功能允许用户代码通过串口进行在电路中编程。有 3 种方法来实现 Flash 存储器的擦除或编程。首先，利用通过一个公共入口来调用低级程序的方法在最终用户应用中进行 Flash 的编程或擦除（IAP）。第二，调用片内的 ISP 引导装载程序。反过来，ISP 引导装载程序也可通过供给最终用户应用的相同公共入口来调用低级子程序。第三，利用支持该器件的商业 EPROM 编程器进行 Flash 的并行编程或擦除。

### 7.2.2 引导块

当微控制器要对其自身的Flash存储器进行编程时，所有细节操作都通过引导块中的代码来处理，引导块与Flash存储器各自独立。所有用户程序只需使用适当的参数调用引导块中的公共入口即可实现所需要的操作。引导块的操作包括以下内容：擦除用户代码、编程用户代码、编程保密位等等。

使用商业并行编程器来擦除芯片。擦除芯片操作将擦除整个引导块的内容。为了运用器件的ISP或IAP功能，用户有必要利用Philips提供的ISP/IAP代码对引导块重新编程。请登陆<http://www.semiconductors.Philips.com>来获取器件的hex文件。有任何问题可直接与[micro.support@philips.com](mailto:micro.support@philips.com)联系。

### 7.2.3 上电复位代码的执行

复位后，P89LV51RD2 将进入 SoftICE 模式（如果之前已通过 ISP 命令将其使能）或尝试自动执行 ISP 引导装载程序。如果该自动执行操作在 400ms 后仍未成功，器件就开始执行用户代码。

### 7.2.4 在系统编程 (ISP)

ISP编程执行时不需要将微控制器从系统中移出。在系统编程特性包含了一系列内部的硬件资源，与内部固件相结合可实现通过串口对P89LV51RD2的远程编程。固件由PHILIPS提供并嵌入到每一个P89LV51RD2芯片当中。PHILIPS的在系统编程特性使嵌入式应用中的在电路编程变得可行，并最大限度减少了额外的元件开销和电路板面积。ISP功能使用芯片的5个管脚(Vdd、Vss、TxD、RxD、RST)。只需要一个小的连接器就可实现通过外部电路使用该特性的应用接口。

### 7.2.5 使用在系统编程

ISP 特性可以适应用户应用中较宽的范围的波特率而不依赖于振荡器的频率。它还可以适应较宽范围的振荡器频率。这是通过对一个接收到的字符中单个位进行位时间测量来实现的。该信息可根据设定的振荡器频率定时器值来对波特率进行编程。为此 ISP 特性要求主机必须向 P89LV51RD2 发送一个初始化字符（大写的 U）以使芯片对波特率进行设定。ISP 固件提供对接收字节的自动回送。一旦波特率的设定完成之后，ISP 固件将只接受 Intel-Hex 格式的记录。Intel Hex 格式的记录包含了用于表示十六进制数的 ASCII 字符。汇总如下：

: NNAAAARRDD..DDCC<crLf>

在 Intel Hex 格式中，‘NN’表示记录中数据字节的个数。P89LV51RD2 最多可接受 32 个数据字节。‘AAAA’字符串表示记录中首字节的地址。如果记录中的字节数为 0，通常该区域设置为‘0000’。‘RR’表示记录的类型：‘00’表示数据类型，‘01’表示文件结束标识。在该应用中，增加额外的记录格式以指示 ISP 应用的命令或数据。

数据字节的最大数目限制为 32（十进制）。ISP 命令汇总见表 10。P89LV51RD2 收到记录后，将数据保存并计算校验和。在整个记录接收完毕后才执行由记录类型所指示的操作。如果校验和发生错误，P89LV51RD2 将从串口发出‘X’指示校验和出错。如果校验结果与记录相符，将会执行指定的命令操作。在大多数情况下，从串口发出‘.’字符表示正确接收。

表 10 在系统编程 (ISP) hex 记录格式

记录类型	命令/数据功能
00	编程用户代码存储器 : nnaaaa00dd..ddcc 此处： nn=编程字节数 aaaa=地址 dd..dd=数据字节 cc=校验和 例如： : 100000000102030405006070809cc
01	文件结束 (EOF)，无任何操作 : xxxxxx01cc 此处： xxxxxx=必须的字段，但其值‘无关’ cc=校验和 例如： : 00000001FF

续上表.....

记录类型	命令/数据功能
02	设置 SoftICE 模式 复位后器件进入 SoftICE 模式。该操作将擦除用户存储器和器件序列号 : 00000002cc 此处: xxxxxx=必须的字段, 但其值‘无关’ cc=校验和 例如: : 00000002FE
03	混合写功能 : nnxxxx03ffssddcc 此处: nn=记录的字节数 xxxx=必须的字段, 但其值‘无关’ ff=子功能代码 ss=选择代码 dd=数据(如果需要) cc=校验和 子功能代码=01(擦除块0) ff=01 子功能代码=05(编程保密位, 加倍时钟) ff=05 ss=01 编程保密位 ss=05 编程加倍时钟位 子功能代码=08(擦除扇区, 128 字节) ff=08 ss=扇区地址高字节(A15:8) dd=扇区地址低字节(A7,A6:0=0) 例如: : 0300000308E000F2(擦除 E000h 的扇区)

续上表.....

记录类型	命令/数据功能
04	<p>显示器件数据或区域校验</p> <p>: 05xxxx04sssseeeffcc</p> <p>此处</p> <p>05=记录的字节数</p> <p>xxxx=必须的字段, 但其值‘无关’</p> <p>04=显示功能代码或区域校验</p> <p>sss=起始地址, MSB 在前</p> <p>eeee=结束地址, MSB 在前</p> <p>ff=子功能</p> <p>    00=显示数据</p> <p>    01=区域校验</p> <p>cc=校验和</p> <p>子功能代码:</p> <p>例如:</p> <p>: 0500000400001FFF00D9 (从 000h~1FFFh 显示)</p>
05	<p>混合读功能</p> <p>: 02xxxx05ffsscc</p> <p>此处:</p> <p>02=记录的字节数</p> <p>xxxx=必须的字节, 但其值‘无关’</p> <p>05=混合读功能代码</p> <p>ffss=子功能和选择代码</p> <p>    0000=读厂商 id</p> <p>    0001=读器件 id</p> <p>    0002=读 ISP/IAP 版本号</p> <p>    0700=读保密位 (00000 SB 0 加倍时钟)</p> <p>cc=校验和</p> <p>例如:</p> <p>: 020000050000F9 (显示厂商 id)</p>
06	<p>波特率直接装载</p> <p>: 02xxxx06HLLcc</p> <p>此处:</p> <p>02=记录字节数</p> <p>xxxx=必须的字段, 但其值‘无关’</p> <p>HH=定时器高字节</p> <p>LL=定时器低字节</p> <p>cc=校验和</p> <p>例如:</p> <p>: 02000007FFFFcc (装载 T2=7FFF)</p>

续上表.....

记录类型	命令/数据功能
07	复位序列号 : xxxxxx07cc 此处: xxxxxx=必须的字段, 但其值‘无关’ 07=复位序列号功能 cc=校验和 例如: : 00000001FF
08	检查序列号 : nnxxxx08ss..sscc 此处 xxxxxx=必须的字段, 但其值‘无关’ 08=检查序列号功能 ss..ss=序列号内容 cc=校验和 例如: : 03000008010203EF (检查 s/n=010203)
09	写序列号 : nnxxxx09ss..sscc 此处 xxxxxx=必须的字段, 但其值‘无关’ 09=写序列号功能 ss..ss=序列号内容 cc=校验和 例如: : 03000009010203EE (写 s/n=010203)
0A	显示序列号 : xxxxxx0Acc 此处: xxxxxx=必须的字段, 但其值‘无关’ 0A=显示序列号功能 cc=校验和 例如: : 0000000AF6

### 7.2.6 使用序列号

器件的 31 字节的序列号和序列号长度 (共 32 字节) 存放在非易失性存储器空间。当器件进入 ISP 模式后, 由其序列号长度来判断序列号是否正被使用。如果序列号长度编程为 00H 或 FFH, 认为序列号未被使用。如果序列号被使用, 用户程序存储器或序列号的读、编程或擦除操作将被禁止直到用户发送一个‘检查序列号’记录, 该记录包含的序列号和序列号长度与原来保存在器件中的序列号和序列号长度相匹配。用户可通过发送‘复位序列号’记录将序列号和序列号长度都设置成 0。而且, ‘复位序列号’记录也可擦除所有的用户代码。



### 7.2.7 在应用中编程的方法

通过一个应用程序对几个在应用中编程（IAP）子程序的调用，可实现有选择地对 Flash 扇区、页、保密位、配置字节和器件 ID 进行擦除、读和编程。所有的调用都是通过一个公共的接口 PGM\_MTP 来实现的。在调用位于地址 1FF0H 的 PGM\_MTP 之前，通过设置微控制器的寄存器可选择编程的功能。IAP 调用见表 11。

表 11 IAP 功能调用

IAP 功能	IAP 调用参数
读 id	输入参数： R1=00h DPH=00H DPL=00H=厂商 id DPL=01H=器件 id 1 DPL=02H=ISP 版本号 DPL=03H=IAP 版本号 返回参数： ACC=所需参数
擦除块 0	输入参数： R1=01h 返回参数： ACC=00=通过 ACC=!00=出错
编程用户代码	输入参数： R1=02h DPH=存储器地址 MSB DPL=存储器地址 LSB ACC=编程字节 返回参数： ACC=00=通过 ACC=!00=出错
读用户代码	输入参数： R1=03h DPH=存储器地址 MSB DPL=存储器地址 LSB 返回参数： ACC=器件数据
编程保密位、加倍时钟	输入参数： R1=05h DPL=01H=保密位 DPL=05H=加倍时钟 返回参数： ACC=00=通过 ACC=!00=出错
读保密位、加倍时钟	输入参数：ACC=07h 返回参数：ACC=000 S/N-匹配 0 SB 0 DBL_CLK

### 7.3 定时器/计数器 0 和 1

2 个 16 位定时器/计数器: 定时器 0 和定时器 1。两者可配置成定时器或事件计数器(见表 12 和表 13)。

用作‘定时器’功能时, 每经过一个机器周期, 寄存器加 1。因此, 可以将一个机器周期看作计数周期。由于一个机器周期由 6 个振荡周期组成, 所以, 定时器的计数率为 1/6 振荡频率。

用作‘计数器’功能时, 每当外部计数管脚, T0 或 T1, 发生一次 1 到 0 的跳变, 寄存器加 1。此功能中, 外部输入脚每个机器周期被采样一次。

当在一个周期内采样为高而下一个周期内采样为低时, 计数值加 1。新的计数值在检测到跳变的周期的下一周期出现在寄存器中。由于识别 1 到 0 的跳变要占用 2 个机器周期(12 个振荡周期), 因此最大计数速率为 1/12 振荡频率。外部输入信号的占空比不受限制, 但为了保证给定电平能在改变之前被检测到, 外部输入信号的状态至少要保持一个完整的机器周期。除了‘定时器’或‘计数器’的选择外, 定时器 0 和定时器 1 还有 4 种工作模式可供选择。

‘定时器’和‘计数器’功能通过特殊功能寄存器 TMOD 的 C/T 位来选择。两个定时器/计数器都有 4 种工作模式, 由 TMOD 的两位(M1, M0)进行选择。两个定时器/计数器的模式 0, 1 和 2 相同, 模式 3 不同。4 种工作模式见下面的描述。

表 12 TMOD—定时器/计数器方式控制寄存器的位分配(地址: 89H)

不可位寻址; 复位值: 0000000B; 复位源: 任何复位

位	7	6	5	4	3	2	1	0
符号	T1GATE	T1C/ $\bar{T}$	T1M1	T1M0	T0GATE	T0C/ $\bar{T}$	T0M1	T0M0

表 13 TMOD—定时器/计数器方式控制寄存器的位描述(地址: 89H)

位	符号	描述
	T1/T0	定时器 1/定时器 0 的位控制。
	GATE	该位被置位时为门控位。仅当‘INTx’脚为高且‘TRx’控制位被置位时使能定时器/计数器‘x’。当该位被清零时, 只要‘TRx’位被置位, 定时器 x 就被使能。
	C/ $\bar{T}$	门控定时器或计数器选择器。该位被清零时用作定时器功能(从内部系统时钟输入脉冲)。该位被置位时用作计数器功能(从‘Tx’脚输入脉冲)。

表 14 TMOD—定时器/计数器方式控制寄存器 M1/M0 工作模式(地址: 89H)

M1	M0	工作模式	描述
0	0	0	8048 定时器, ‘TLx’ 用作 5 位的预分频器
0	1	1	16 位定时器/计数器。‘THx’ 和 ‘TLx’ 级联; 无预分频器。
1	0	2	8 位自动重装定时器/计数器。定时器溢出时, ‘THx’ 的值重装到 ‘TLx’ 中。
1	1	3	(定时器 0) TL0 用作 8 位的定时器/计数器, 由标准定时器 0 的控制位来控制。TH0 用作 8 位的定时器, 仅受定时器 1 的控制位来控制。
1	1	3	(定时器 1) 禁能定时器/计数器 1。

表 15 TCON—定时器/计数器控制寄存器的位分配(地址: 88H)

可位寻址; 复位值: 0000000B; 复位源: 任何复位

位	7	6	5	4	3	2	1	0
符号	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

表 16 TCON—定时器/计数器控制寄存器的位描述（地址：88H）

位	符号	描述
7	TF1	定时器 1 溢出标志。定时器/计数器溢出时由硬件置位。中断执行时硬件清零，或由软件清零。
6	TR1	定时器 1 运行控制位。通过软件将其置位/清零来起动/关闭定时器/计数器 1。
5	TF0	定时器 0 溢出标志。定时器/计数器溢出时由硬件置位。中断执行时硬件清零，或由软件清零。
4	TR0	定时器 0 运行控制位。通过软件将其置位/清零来起动/关闭定时器/计数器 0。
3	IE1	中断 1 边沿标志。当检测到外部中断 1 边沿/低电平时由硬件置位该标志。中断处理时由硬件清零，或通过软件清零。
2	IT1	中断 1 类型控制位，通过软件置位/清零该位来选择外部中断 1 以下降沿/低电平方式触发。
1	IE0	中断 0 边沿标志。当检测到外部中断 0 边沿/低电平时由硬件置位该标志。中断处理时由硬件清零，或通过软件清零。
0	IT0	中断 0 类型控制位。通过软件置位/清零该位来选择外部中断 0 以下降沿/低电平方式触发。

### 7.3.1 模式 0

工作在模式 0 的定时器类似于一个 8048 定时器，是一个具有固定 32 分频预分频器的 8 位计数器。模式 0 的操作如图 7 所示。

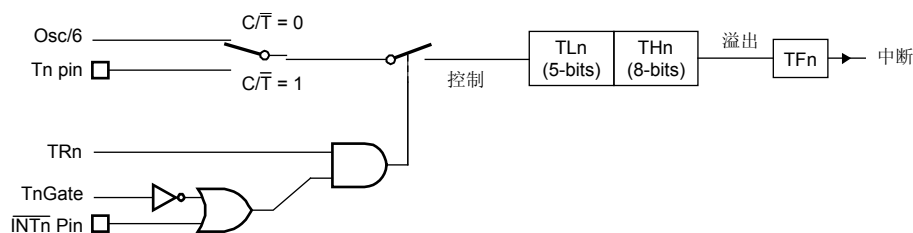


图 7 定时器/计数器 0 或 1 工作在模式 0（13 位计数器）

该模式下，定时器寄存器配置成一个 13 位的寄存器。当计数值由全 1 变成全 0 时，定时器中断标志 TF<sub>n</sub> 置位。当 TR<sub>n</sub>=1 和 GATE=0 或  $\overline{INTn}=1$ （GATE=1 时，外部输入  $\overline{INTn}$  控制定时器，定时器可用来测量脉冲宽度）时，计数输入使能到定时器。TR<sub>n</sub> 是特殊功能寄存器 TCON 的一个控制位（图 6）。GATE 是 TMOD 寄存器中的一位。

该 13 位寄存器由 TH<sub>n</sub> 的 8 位和 TL<sub>n</sub> 的低 5 位组成。TL<sub>n</sub> 的高 3 位不确定，可被忽略。置位运行标志（TR<sub>n</sub>）不会清零寄存器。

定时器 0 和定时器 1 的模式 0 操作相同（见图 70）。有 2 个不同的 GATE 位，一个用于定时器 1（TMOD.7），一个用于定时器 0（TMOD.3）。

### 7.3.2 模式 1

除定时器寄存器为 16 位（TH<sub>n</sub> 和 TL<sub>n</sub>）外，模式 1 的操作与模式 0 完全相同。

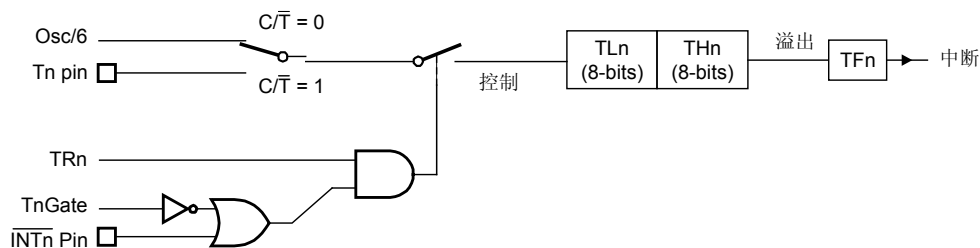


图 8 定时器/计数器 0 或 1 工作在模式 1（16 位计数器）

### 7.3.3 模式 2

模式 2 中，定时器寄存器配置一个 8 位的计数器 (TLn)，具有自动重装功能，如图 9 所示。TLn 的溢出不仅使 TFn 置位，而且将 THn 的内容重新装入 TLn，THn 的值由软件预先设置。重装时 THn 的内容不变。定时器 0 和定时器 1 的模式 2 操作相同。

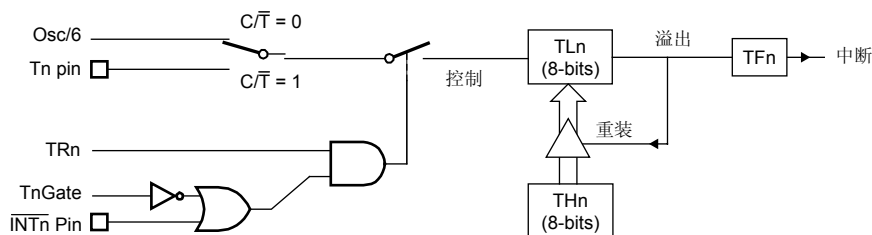


图 9 定时器/计数器 0 工作在模式 2 (8 位自动重装)

### 7.3.4 模式 3

模式 3 中，定时器 1 停止工作 (计数值保持不变)。等效于 TR1=0。

模式 3 中，定时器 0 将 TL0 和 TH0 分成 2 个独立的计数器。模式 3 的逻辑和定时器 0 的工作如图 10 所示。TL0 使用定时器 0 控制位：T0C/T, T0GATE, TR0, INT0 和 TF0。TH0 锁定成定时器 (计数器周期)，占用定时器 1 的 TR1 和 TF1 位。这样，TH0 就控制着 ‘定时器 1’ 的中断。

模式 3 用在需要一个额外的 8 位定时器的场合。当定时器 0 工作在模式 3 时，P89LV51RD2 可看成一个拥有额外定时器的器件。

注：当定时器 0 工作在模式 3 时，定时器 1 可通过进入/退出模式 3 来启动/关闭。定时器 1 还可用作串口的波特率发生器或用在不需要中断的应用中。

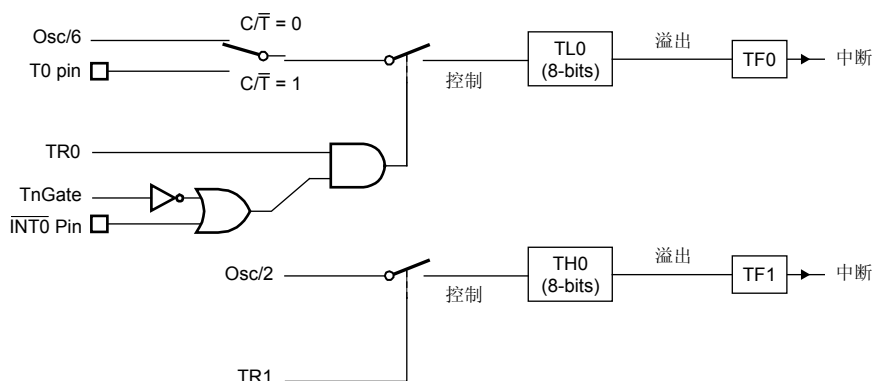


图 10 定时器/计数器 0 工作在模式 3 (2 个 8 位计数器)

## 7.4 定时器 2

定时器 2 是一个 16 位的定时器/计数器，可用作事件定时器或事件计数器，由特殊功能寄存器 T2CON 的 C/T2 位选择。定时器 2 有 4 种工作模式：捕获、自动重装 (向上或向下计数)、时钟输出和波特率发生器。工作模式的选择见表 17，需要使用寄存器 T2CON (表 18 和表 19) 和 T2MOD (表 20 和表 21)。

表 17 定时器 2 的工作模式

RCLK+TCLK	CP/RL2	TR2	T2OE	模式
0	0	1	0	16 位自动重装
0	1	1	0	16 位捕获
0	0	1	1	可编程时钟输出
1	X	1	0	波特率发生器
X	X	0	X	关闭

表 18 T2CON—定时器/计数器 2 控制寄存器位分配（地址：C8H）

可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/ $\overline{T2}$	CP/ $\overline{RL2}$

表 19 T2CON—定时器/计数器 2 控制寄存器的位描述（地址：C8H）

位	符号	描述
7	TF2	定时器 2 溢出标志。该位在定时器 2 溢出时置位，由软件清零。当 RCLK 或 TCLK=1 或者定时器 2 工作在时钟输出模式时，TF2 将不会置位。
6	EXF2	定时器 2 外部标志。当定时器 2 处于捕获、重装或波特率模式，EXEN2=1 和 T2EX 脚出现负跳变时，该位置位。如果定时器 2 中断使能，EXF2=1 时，CPU 将执行定时器 2 中断服务程序。EXF2 必须由软件清零。
5	RCLK	接收时钟标志。当该位被置位时，UART 将定时器 2 的溢出脉冲作为模式 1 和 3 的接收时钟。当 RCLK=0 时，定时器 1 的溢出脉冲被用作接收时钟。
4	TCLK	发送时钟标志。该位置位时，UART 将定时器 2 的溢出脉冲作为模式 1 和 3 的发送时钟。当 TCLK=0 时，定时器 1 的溢出脉冲被用作发送脉冲。
3	EXEN2	定时器 2 外部使能标志。该位置位时，如果定时器 2 不用作串口时钟，当 T2EX 脚上负跳变时，使能定时器的捕获或重装操作。当 EXEN2=0 时，定时器 2 忽略 T2EX。
2	TR2	定时器 2 的起动/关闭控制位。该位为 1 时使能定时器 2 运行。
1	C/ $\overline{T2}$	定时器或计数器选择。（定时器 2） 0=内部定时器（fosc/6） 1=外部事件计数器（下降沿触发；外部时钟的最大计数速率=fosc/12）
0	CP/ $\overline{RL2}$	捕获/重装标志。该位置位时，如果 EXEN2=1，T2EX 上出现负跳变时捕获。该位被清零时，如果 EXEN2=1，当定时器 2 溢出或 T2EX 上出现负跳变时自动重装。当 RCLK=1 或 TCLK=1 时，该位被忽略，定时器 2 溢出时被强制自动重装。

表 20 T2MOD—定时器 2 方式控制寄存器的位分配（地址：C9H）

不可位寻址；复位值：xx000000B

位	7	6	5	4	3	2	1	0
符号	-	-	-	-	-	-	T2OE	DCEN

表 21 T2MOD—定时器 2 方式控制寄存器的位描述（地址：C9H）

位	符号	描述
7~2	-	保留为将来之用。通过用户程序将其清零。
1	T2OE	定时器 2 输出使能位。仅用于可编程时钟输出模式。
0	DCEN	递减计数使能位。该位置位时，定时器 2 可配置成递增/递减计数器。

#### 7.4.1 捕获模式

定时器 2 工作在捕获模式时有两种情况，由 T2CON 的 EXEN2 位来选择。如果 EXEN2=0，定时器 2 用作一个 16 位的定时器或计数器（由 T2CON 的 C/ $\overline{T2}$  位决定），溢出时将置位 TF2（定时器 2 的溢出位）。捕获模式原理如图 11 所示。

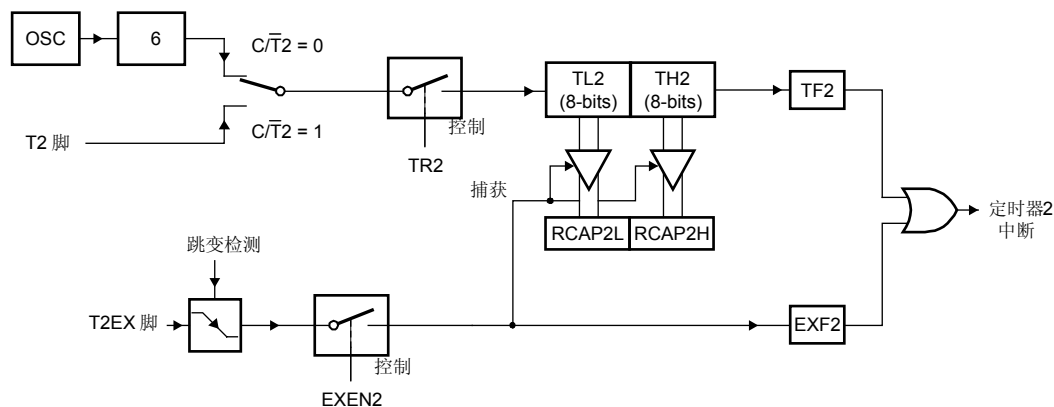


图 11 定时器 2 工作在捕获模式

只要 IEN0 寄存器的定时器 2 中断位被使能，TF2 位就可用来产生中断。如果 EXEN2=1，除上述操作外，定时器 2 还增加了一种特性——外部输入 T2EX 脚上 1 到 0 的跳变将使定时器 2 寄存器，TL2 和 TH2，的当前值分别装载到寄存器 RCAP2L 和 RCAP2H 中。

而且，T2EX 脚的跳变还将置位 T2CON 的 EXF2 位。EXF2 位和 TF2 位一样，也可用来产生中断（两者共用定时器 2 溢出的中断向量）。由此可见，定时器 2 的中断可由 TF2 或 EXF2 产生，由产生中断的事件决定。

这种情况下不会重装 TL2 和 TH2。即使 T2EX 脚出现捕获事件，计数器仍继续对 T2 脚的跳变或 fosc/6 的脉冲进行计数。因为 RCAP2L 和 RCAP2H 重装后其值并不受保护，所以一旦定时器 2 产生中断，中断必须在 T2EX 脚出现新的捕获事件之前进行处理。否则，T2EX 脚上新的下降沿将使 TL2 和 TH2 的当前值重装到 RCAP2L 和 RCAP2H，从而破坏引起中断的 RCAP2L 和 RCAP2H 值。

#### 7.4.2 自动重装模式（递增或递减计数器）

在 16 位自动重装模式下，定时器 2 可配置成定时器或计数器（通过 T2CON 的  $\overline{C/T2}$ ），并可通过编程设定成递增或递减计数。定时器 2 的计数方向由 T2MOD（见表 20 和表 21）的 DCEN 位（递减计数使能）决定。复位后，DCEN=0，定时器 2 默认为递增计数。如果 DCEN 位被置位，定时器 2 根据 T2EX 脚的状态进行递增或递减计数。

图 12 所示为定时器 2 的自动递增计数模式（DCEN=0）。

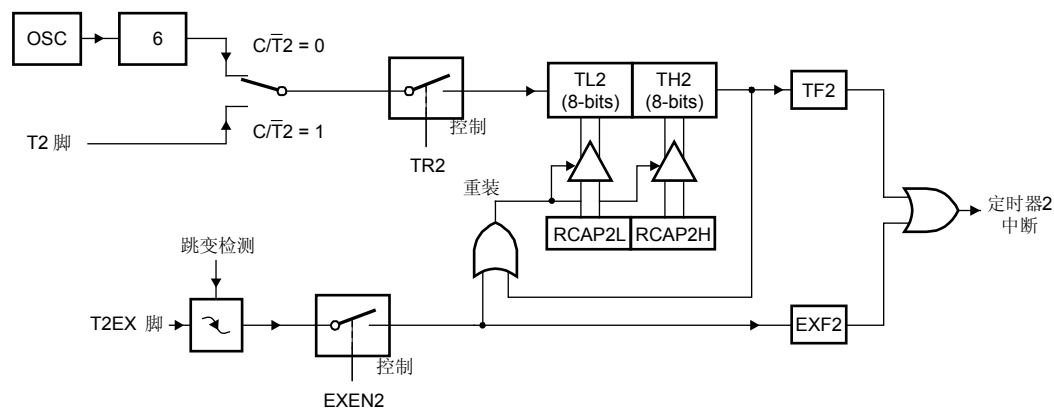


图 12 定时器 2 工作在自动重装模式（DCEN=0）

定时器 2 工作在自动重装模式时有 2 种情况，由 T2CON 寄存器的 EXEN2 位来选择。如果 EXEN2=0，定时器 2 可计数到 0FFFFH，溢出时置位 TF2（溢出标志）。定时器 2 溢出时，RCAP2L 和 RCAP2H 的 16 位内容将装载到定时器的寄存器中。RCAP2L 和 RCAP2H 的值由软件预先设置。

定时器 2 递增计数的自动重装频率由下式决定：

$$\frac{\text{提供的频率}}{(65536 \angle (\text{RCAP2H}, \text{RCAP2L}))} \quad (1)$$

此处，提供的频率为  $f_{osc} (C/\overline{T2} = 0)$  或 T2 脚信号的频率 ( $C/\overline{T2} = 1$ )。

如果 EXEN2=1，定时器的溢出或 T2EX 脚输入的负跳变都将触发 16 位的重装操作。T2EX 脚的负跳变还会置位 EXF2 位。如果定时器 2 中断使能，TF2 或 EXF2 为 ‘1’ 时都将产生中断。

微处理器硬件识别 T2EX 上的下降沿和置位 EXF2 需要 3 个连续的机器周期：第 1 个机器周期采样 T2EX 为 ‘1’；第 2 个机器周期采样 T2EX 为 ‘0’；第 3 个机器周期置位 EXF2。

图 13 所示为 DCEN=1 时定时器 2 的递增或递减计数。该模式允许 T2EX 脚控制计数方向。当 T2EX 脚为 ‘1’ 时，定时器 2 递增计数。定时器 2 在计数值为 0FFFFH 时溢出，并置位 TF2 标志，产生中断（如果中断被使能）。定时器的溢出也将使 RCAP2L 和 RCAP2H 的 16 位值重装到定时器寄存器 TL2 和 TH2 中。

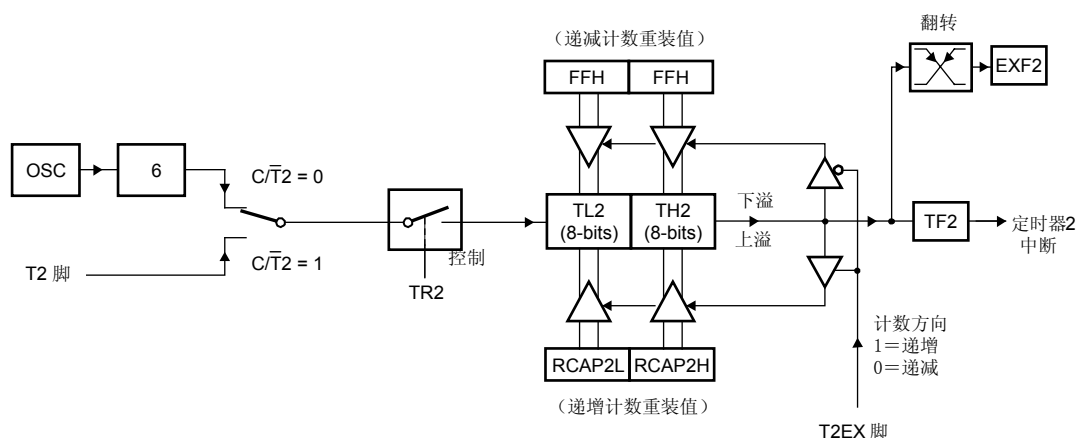


图 13 定时器 2 工作在自动重装模式 (DCEN=1)。

当 T2EX 脚为 ‘0’ 时，定时器 2 递减计数。当 TL2 和 TH2 的值与 RCAP2L 和 RCAP2H 的值相等时定时器溢出。定时器 2 的溢出将置位 TF2 标志并将 0FFFFH 重装到定时器寄存器 TL2 和 TH2 中。定时器 2 的下溢或上溢都会使 EXF2 翻转。如果需要，EXF2 位可用作第 17 位。

### 7.4.3 可编程时钟输出

通过编程，可从 T2 脚 (P1.0) 输出占空比为 50% 的时钟信号。P1.0 口，除用作通用 I/O 口外，还含有 2 个附加功能。P1.0 口可编程为：

1. 定时器/计数器 2 的外部时钟输入或
2. 在 16MHz 的工作频率下，输出占空比为 50% 的时钟信号，时钟信号的频率范围为 122Hz~8MHz。

通过清零位  $C/\overline{T2}$ （位于 T2CON 寄存器）和置位 T2MOD 的 T0E 位来将定时器/计数器 2 配置成一个时钟发生器。通过置位 TR2 位 (T2CON.2) 就可启动定时器的运行。

时钟输出频率取决于振荡器频率和定时器 2 捕获寄存器 (RCAP2H,RCAP2L) 的重装值，见等式 2：

$$\frac{\text{振荡器频率}}{2 \times (65536 \angle (\text{RCAP2H}, \text{RCAP2L}))} \quad (2)$$

此处，(RCAP2H,RCAP2L) = RCAP2H 和 RCAP2L 的内容，将其看作一个 16 位的无符号整数。

在时钟输出模式中，定时器 2 计数器的循环翻转不会产生中断。这一点与定时器 2 用作波特率发生器时一致。

### 7.4.4 波特率发生器模式

通过设置 T2CON 的 TCLK 和/或位 RCLK，定时器 1 或定时器 2 可用来产生 UART 的发送和接收波特率（见 7.5 节“UART”）。当 TCLK=0 时，定时器 1 用作 UART 发送波特率发生器。当 TCLK=1 时，定时器 2 用作 UART 发送波特率发生器。RTCLK 的设置方法与 TCLK 类似，但它用来产生接收波特率。利

用这两位，UART 可以得到不同的接收和发送波特率——定时器 1 或定时器 2。

图 14 所示为定时器工作在波特率发生器模式：

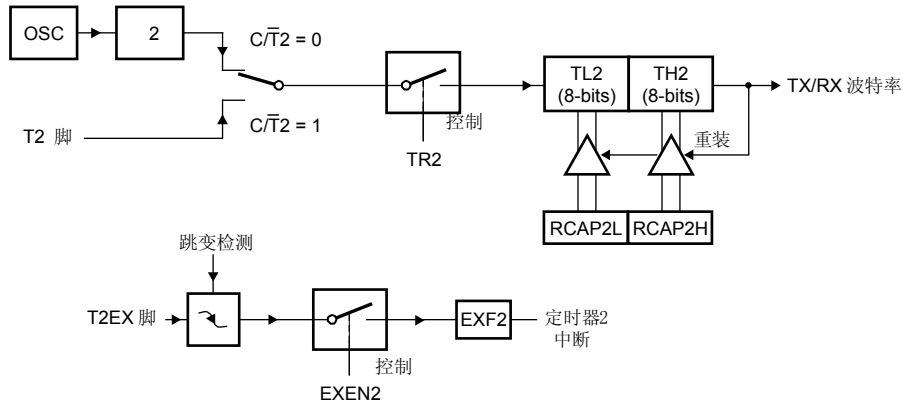


图 14 定时器 2 工作在波特率发生器

波特率模式和自动重装模式一样，TH2 值的翻转将使得 RCAP2H 和 RCAP2L 的内容重装到定时器 2 寄存器中。RCAP2H 和 RCAP2L 的值由软件预先设置。

UART 模式 1 和 3 的波特率由定时器 2 的溢出速率决定：

模式 1 和 3 的波特率 = 定时器 2 的溢出速率 / 16

定时器 2 通过配置可用作‘定时器’或‘计数器’功能。在大量应用中，它都用作‘定时器’( $C/\overline{T2} = 0$ )。定时器 2 用作定时器时的操作与用作波特率发生器时不同。

正常情况下，定时器 2 用作定时器时，每经过一个机器周期计数值加 1 (即计数频率为 1/6 振荡器频率)。但用作波特率发生器时，定时器 2 每经过一个振荡周期加 1。波特率可由下式得到：

$$\text{模式 1 和 3 的波特率} = \frac{\text{振荡器频率}}{(16 \times (65536 - (\text{RCAP2H}, \text{RCAP2L})))} \quad (3)$$

此处，(RCAP2H,RCAP2L) = RCAP2H 和 RCAP2L 的内容，将其看作一个 16 位的无符号整数。

仅当 T2CON 寄存器的 RCLK 和/或 TCLK=1 时，定时器 2 的波特率发生器模式才有效。注意：波特率模式中，TH2 值的翻转不会置位 TF2 和产生中断。因此，当定时器 2 工作在波特率发生器模式时不必禁用定时器 2 中断。而且，当 EXEN2 (T2 外部使能标志) 被置位时，T2EX (定时器/计数器 2 触发输入) 上的负跳变可将 EXF2 (T2 外部标志) 置位，但并不会将 (RCAP2H,RCAP2L) 装载到 (TH2,TL2)。因此，当定时器 2 用作波特率发生器时，如果需要，T2EX 可用作一个额外的外部输入。

当定时器 2 处于波特率发生器模式时，不能读或写 TH2 和 TL2 寄存器。因为在该模式下读出或写入的 TH2 或 TL2 值都是不准确的。RCAP2 寄存器可以读，但不能写，因为写操作会将装载值覆盖从而产生写和/或重装错误。在访问定时器 2 或 RCAP2 寄存器之前必须先关闭定时器 (清除 TR2)。表 22 列出了常用的波特率及其由来。

#### 7.4.5 波特率公式小结

定时器 2 工作在波特率发生器模式。外部时钟信号由 T2 (P1.0) 输入，波特率为：

$$\text{波特率} = \text{定时器 2 溢出速率} / 16$$

若定时器 2 使用内部时钟信号，波特率为：

$$\text{波特率} = f_{\text{osc}} / (16 \times (65536 - (\text{RCAP2H}, \text{RCAP2L})))$$

此处  $f_{\text{osc}}$  = 振荡器频率。

对上式经过改写，可得出 RCAP2H 和 RCAP2L 的重装值：

$$\text{RCAP2H}, \text{RCAP2L} = 65536 - f_{\text{osc}} / (16 \times \text{波特率})$$



表 22 由定时器 2 产生的常用波特率

波特率	Osc 频率	定时器 2	
		RCAP2H	RCAP2L
750K	12MHz	FF	FF
19.2K	12MHz	FF	D9
9.6K	12MHz	FF	B2
4.8K	12MHz	FF	64
2.4K	12MHz	FF	C8
600	12MHz	FB	1E
220	12MHz	F2	AF
600	6MHz	FD	8F
220	6MHz	F9	57

## 7.5 UART

P89LV51RD2 的 UART 除可工作在所有的标准模式之外，还包含一些标准 80C51 UART 的增强特性：帧错误检测和自动地址识别。

### 7.5.1 模式 0

串行数据通过 RxD 进出。TxD 输出移位时钟。每次发送或接收都为 8 位，LSB（最低位）在前。波特率固定为 CPU 时钟频率的 1/6。UART 工作在模式 0 时，不论数据是否通过 RxD 发送或接收，串行时钟信号都从 TxD 输出。

### 7.5.2 模式 1

TxD 脚发送，RxD 脚接收，每次数据为 10 位：1 个起始位（逻辑 0），8 个数据位（LSB 在前）以及 1 个停止位（逻辑 1）。当接收数据时，停止位保存在 SCON 中的 RB8。该模式的波特率可变，由定时器 1/2 的溢出速率决定。

### 7.5.3 模式 2

TxD 脚发送，RxD 脚接收，每次数据为 11 位：1 个起始位（逻辑 0），8 个数据位（LSB 在前），一个可编程第 9 位数据及 1 个停止位（逻辑 1）。发送数据时，第 9 个数据位（SCON 中的 TB8 位）可置为 0 或 1。例如可将奇偶位（PSW 内 P 位）放入 TB8。接收时，第 9 位数据存入 SCON 的 RB8 位，而停止位不会被保存。波特率可编程为 CPU 时钟频率的 1/16 或 1/32，由 PCON 的 SMOD1 位决定。

### 7.5.4 模式 3

TxD 脚发送，RxD 脚接收，每次数据为 11 位：1 个起始位（逻辑 0），8 个数据位（LSB 在前），1 个可编程的第 9 位数据及 1 个停止位（逻辑 1）。实际上，模式 3 除了波特率外其它均与模式 2 相同。模式 3 的波特率可变并由定时器 1/2 的溢出率决定。

表 23 SCON—串口控制寄存器的位分配（地址：98H）

位	7	6	5	4	3	2	1	0
符号	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI

表 24 SCON—串口控制寄存器的位描述（地址：98H）

位	符号	描述
7	SM0/FE	该位的用途由 PCON 寄存器中的 SMOD0 决定。如果 SMOD0=0，该位作为 SM0，和 SM1 一起定义串口模式；如果 SMOD0=1，该位作为 FE(帧错误)。当检测到一个无效的停止位时，FE 被接收器置位。一旦置位后，该位不能由有效帧清零，只能通过软件清零。（注：UART 模式位 SM0 和 SM1 应当在 SMOD0=1 之前设置）
6	SM1	和 SM0 定义串行口操作模式（见下面的表 25）。
5	SM2	使能模式 2 和 3 中的多机通信功能。在模式 2 或 3 中，如果 SM2=1 而且接收到的第 9 位数据（RB8）为 0 时，则 RI 不会被激活。在模式 1 中，如果 SM2=1 而且未接收到一个有效的停止位时，RI 不会被激活。在模式 0 中，SM2 应当为 0。
4	REN	使能串行接收。由软件置位以使能接收。软件清零则禁止接收。
3	TB8	模式 2 和 3 中将要发送的第 9 位数据，可以根据需要由软件置位或清零。
2	RB8	模式 2 和 3 中接收的第 9 位数据，在模式 1 中（SM2 必须为 0），RB8 是接收到的停止位。在模式 0 中，RB8 未定义。
1	TI	发送中断标志。模式 0 中，在第 8 位数据发送结束时由硬件置位。在其它任何串行发送模式中，在发送停止位时由硬件置位。必须通过软件清零。
0	RI	接收中断标志，模式 0 中，第 8 位接收完成时由硬件置位。在其它串行发送模式中，在接收停止位的中间时刻由硬件置位（例外情况请参考上面的 SM2 描述）。必须通过软件清零。

表 25 SCON—串口控制寄存器（地址：98H）SM0/SM1 串口工作模式的定义

SM0,SM1	UART 模式	波特率
00	0: 移位寄存器	CPU 时钟/6
01	1: 8 位 UART	可变
10	2: 9 位 UART	CPU 时钟/32 或 CPU 时钟/16
11	3: 9 位 UART	可变

### 7.5.5 帧错误

如果 SMOD0 (PCON.6) =1，帧错误 (FE) 在 SCON.7 位报告。如果 SMOD0=0，SCON.7 作为 SM0。建议在 SMOD0 置位前对 SM0 进行设置。

### 7.5.6 更多关于 UART 模式 1 的信息

接收在检测到 RxD 端电平负跳变时启动，CPU 对 RxD 不断采样，采样速率为波特率的 16 倍。当检测到负跳变时，16 分频计数器立即复位来将其翻转值按接收到的位时间为边界进行分配。16 个计数状态把每个位时间分成 16 份。在每个位时间的第 7、8、9 计数状态时，位检测器对 RxD 端的值进行采样。取值为三个采样值中取多数（至少 2 个）作为读入值，这样做是为了抑制噪声。如果在第一个位时间所接收的位不为 0，接收电路复位并等待另一个负跳变的到来。这样可以防止错误的起始位。如果起始位被证明是有效的，则被移入输入移位寄存器，并开始接收这一帧剩余的位。

当且仅当产生最后一位移位脉冲时满足下列条件：(a) RI=0 以及(b) SM2=0 或接收到的停止位=1。

上述两个条件中的任何一个不满足，所接收到的数据帧都会丢失。两个条件都满足时，停止位就进入 RB8，而 8 位数据则进入 SBUF，并且 RI 置位。

### 7.5.7 更多关于模式 2 和模式 3 的信息

接收的工作方式与模式 1 相同。

当且仅当产生最后一位移位脉冲时满足下列条件: (a) RI=0, 以及(b) SM2=0 或接收到的第 9 位数据=1, 数据才会被装入 SBUF 和 RB8, 并置位 RI。

上述两个条件中的任何一个不满足, 所接收到的数据帧都会丢失, RI 不被置位。两个条件都满足时, 接收到的第 9 位数据就进入 RB8, 而前 8 位数据则进入 SBUF。

### 7.5.8 多机通信

UART 模式 2 及模式 3 有一个专门的应用领域即多机通信。在这些模式时, 发送及接收均为 9 位数据。接收时第 9 位数据存入 RB8。UART 可编程为: 接收到停止位时, 仅当 RB8=1 时串口中断才激活。可通过置位 SCON 内 SM2 位来使能这一特性。下面讲述多机系统使用该特性的一种方法:

当主机需要发送一数据块给某一从机时, 首先发送一个地址字节以识别目标从机。地址字节与数据字节的区别在于第 9 位数据, 地址字节的第 9 位为 1, 而数据字节的第 9 位为 0。SM2=1 时, 数据字节不会使从机产生中断, 而地址字节则会使所有从机都产生中断, 这样每个从机可以检查接收到的字节并判断是否被寻址。被寻址的从机将清零 SM2 位以准备接收随后的数据字节 (数据长度仍是 9 位)。未被寻址的从机的 SM2 位仍为 1, 这样就忽略随后的数据继续各自工作。

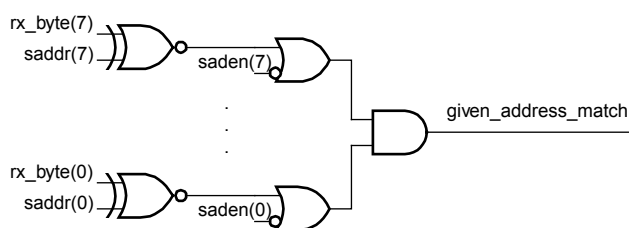
在模式 0 中 SM2 无效, 在模式 1 中 SM2 可用来检测停止位是否有效, 尽管该功能通过检查帧错误标志来实现更好。当 SM2=1 且模式 1 中, UART 接收数据时, 接收中断不会激活除非接收到一个有效的停止位。

### 7.5.9 自动地址识别

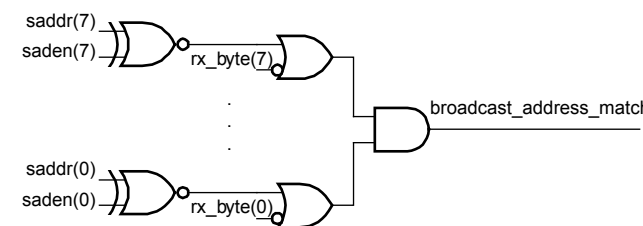
自动地址识别是这样一种特性, 它允许 UART 使用硬件进行比较, 从串行数据流中识别出特定的地址。这样就不必花费大量软件资源去检查每一个从串口输入的串行地址。将 SCON 内 SM2 置位可使能该特性。在 9 位 UART 模式 (模式 2 和模式 3) 下, 如果接收的字节中包含“给定”地址或“广播”地址, 接收中断标志 (RI) 将自动置位。在 9 位模式下要求第 9 个信息位为 1 以表明该信息内容是地址而非数据。

使用自动地址识别特性时, 主机通过调用给定的从机地址选择与一个 (或多个) 从机通信。使用广播地址可以联系所有的从机。有两个特殊功能寄存器用来定义从机地址: SADDR 和地址屏蔽 SADEN。SADEN 用于定义 SADDR 内哪几位需要使用而哪几位是无关位。SADEN 可以与 SADDR 逻辑“与”得出给定的地址, 主机使用该地址对每一从机进行寻址。使用给定地址可以识别多个从机而排除另外的从机。

器件使用下面图 15 的方法来检测是否接收到‘给定’或‘广播’地址。



P89LV51RD2 UART 使用该逻辑来检测接收的数据中的‘给定地址’



P89LV51RD2 UART 使用该逻辑来检测接收的数据中的‘广播地址’

图 15 多机通信使能时 UART 检测‘给定’和‘广播’地址的方法

下面给出了几个例子来说明上述方法的通用性。

表 26 从机 0 和 1 举例

从机 0            SADDR=1100 0000  
                   SADEN=1111 1101  
                   给定地址=1100 00X0

从机 1            SADDR=1100 0000  
                   SADEN=1111 1110  
                   给定地址=1100 000X

上例中 SADDR 相同，而 SADEN 的数据用于区分两个从机。从机 0 要求位 0 为 0 而忽略位 1。从机 1 则要求位 1 为 0 而忽略位 0。由于从机 1 要求位 1 必须为 0，从机 0 唯一的地址应当是 1100 0010。由于从机 1 的位 0 必须为 1 来排除从机 0，从机 1 唯一的地址应当为 1100 0001。通过给定一个位 0=0（从机 0）和位 1=0（从机 1）的地址可同时选择两个从机，从而取地址 1100 0000 时两从机都可被寻址。

下例所示为选择从机 1、2 而不选从机 0：

表 27 从机 0, 1 和 2 举例

从机 0            SADDR=1100 0000  
                   SADEN=1111 1001  
                   给定地址=1100 0XX0

从机 1            SADDR=1110 0000  
                   SADEN=1111 1010  
                   给定地址=1100 0X0X

从机 2            SADDR=1110 0000  
                   SADEN=1111 1100  
                   给定地址=1110 00XX

上述三个从地址只有低 3 位不同。从机 0 要求位 0=0，它可通过 1110 0110 单独寻址；从机 1 要求位 1=0，可通过 1110 0101 单独寻址；从机 2 要求位 2 为 0，可通过 1110 0011 单独寻址。由于必须使地址字节的第 2 位为“1”以屏蔽从机 2，因此使用地址 1110 0100 可选择从机 0 和 1 同时屏蔽从机 2。将 SADDR 和 SADEN 相“或”后产生每个从机的“广播”地址，结果为零的位视为无关位。大多数情况下，无关位被解释为 1，这样，“广播”地址为 FFH。复位时 SADDR 和 SADEN 均为 00H，此时产生了一个所有位都是无关位的给定地址，即所有位都“无关”的广播地址。这样有效地禁止了自动寻址模式，允许微处理器使用不带有上述特性的标准 UART 驱动器。

## 7.6 串行外围接口

### 7.6.1 SPI 接口

- 主机或从机操作
- 10MHz 位频率（最大）
- 数据传输时 LSB 在前或 MSB 在前
- 4 个可编程位速率
- 传输终止（SPIF）
- 写冲突标志保护（WCOL）
- 空闲模式唤醒（仅用于从机模式）

### 7.6.2 SPI 描述

串行外围接口 (SPI) 实现了 P89LV51RD2 和外围器件之间以及多个 P89LV51RD2 器件之间数据的高速同步传输。图 16 所示为主机和从机 SPI 器件之间的连接。SCK 是主机和从机模式的时钟输入、输出脚。主器件 SPI 数据寄存器的写操作完成后，起动 SPI 时钟发生器。写入的数据便从主器件的 MOSI 脚移出，移入到从器件的 MOSI 脚。一个字节的数据传输结束后，SPI 时钟发生器关闭，SPIF 标志置位。如果 SPI 中断使能位 (SPIE) 和串口中断使能位 (ES) 置位，产生 SPI 中断请求。

当外部主机驱动从机选择输入脚， $\overline{SS}/P1[4]$ ，为低电平时，SPI 模块用作从机。如果  $\overline{SS}/P1[4]$  脚不为低电平，则 SPI 单元不被激活，MOSI/P1[5] 只能用作输入/输出。

CPHA 和 CPOL 位控制着 SPI 时钟的相位和极性。图 17 和图 18 列出了两位的 4 种可能组合。

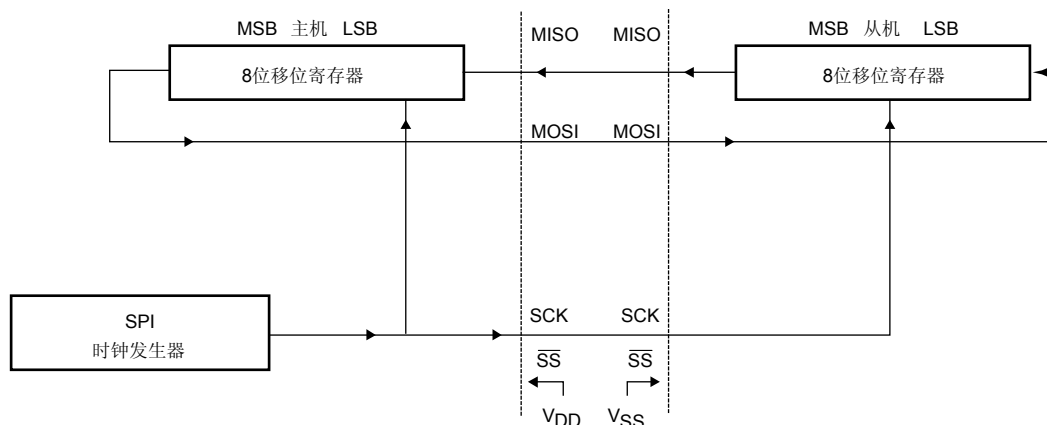


图 16 SPI 主机—从机之间的连接

表 28 SPCR—SPI 控制寄存器的位分配 (地址: D5H)

可位寻址; 复位源: 任何复位; 复位值: 0000000B

位	7	6	5	4	3	2	1	0
符号	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0

表 29 SPCR—SPI 控制寄存器的位描述 (地址: D5H)

位	符号	描述
7	SPIE	如果 SPIE 和 ES 都置 1，SPI 中断使能。
6	SPE	SPI 使能位。该位置位时 SPI 使能。
5	DORD	数据传输顺序。0=MSB 在前; 1=LSB 在前。
4	MSTR	主机/从机选择。1=主机模式, 0=从机模式。
3	CPOL	时钟极性。1=空闲时 SCK 为高 (低有效); 0=空闲时 SCK 为低 (高有效)。
2	CPHA	时钟相位控制位。1=移位在时钟的后沿触发; 0=移位在时钟的前沿触发。
1	SPR1	SPI 时钟分频率选择位 1。当器件用作主机时, 该位和 SPR0 位一起控制器件 SCK 的分频率。SPR1 和 SPR0 对从机无影响。见下面的表 30。
0	SPR0	SPI 时钟分频率选择位 0。当器件用作主机时, 该位和 SPR1 位一起控制器件 SCK 的分频率。SPR1 和 SPR0 对从机无影响。见下面的表 30。

表 30 SPCR—SPI 控制寄存器时钟的分频率选择 (地址: D5H)

SPR1	SPR0	SCK=fosc 分频
0	0	4
0	1	16
1	0	64
1	1	128

表 31 SPSR—SPI 状态寄存器的位分配 (地址: AAH)

可位寻址; 复位源: 任何复位; 复位值: 0000000B

位	7	6	5	4	3	2	1	0
符号	SPIF	WCOL	-	-	-	-	-	-

表 32 SPSR—SPI 状态寄存器的位描述 (地址: AAH)

位	符号	描述
7	SPIF	SPI 中断标志。数据传输结束时, 该位被置位。如果 SPIE=1 和 ES=1, 产生中断。该位由软件清零。
6	WCOL	写冲突标志。数据传输过程中 SPI 数据寄存器被写入时该位置位。该位由软件清零。
5~0	-	保留为将来之用。通过用户程序将其清零。

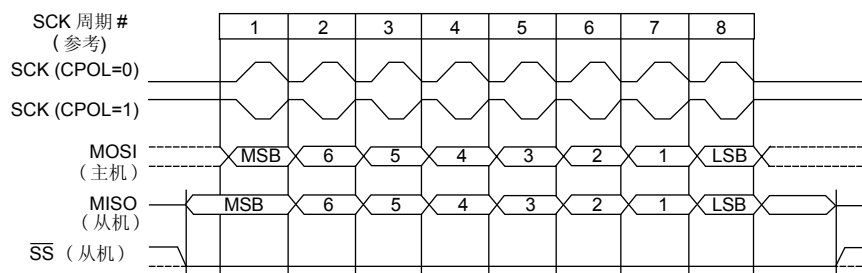


图 17 SPI 传输格式 (CPHA=0)

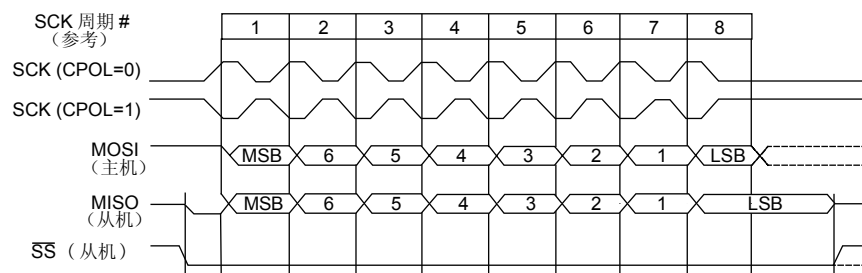


图 18 SPI 传输格式 (CPHA=1)

### 7.7 看门狗定时器

P89LV51RD2 具有一个可编程的看门狗定时器 (WDT), 可用于系统故障安全保护, 防止软件死锁和自动恢复。

用户软件可以通过在用户定义的时间周期内刷新 WDT 来保护系统来防止软件的死锁。如果用户软件在定义的时间周期内刷新 WDT 失败, 如果使能 (WDRE=1), 则启动内部硬件复位。软件可设计成如果程序运行不正确, WDT 就溢出。

P89LV51RD2 的 WDT 以系统时钟 (XTAL1) 为时间基准。所以严格说来, WDT 是一个看门狗计数器,

而不是看门狗定时器。每经过 344,064 个振荡周期，WDT 寄存器加 1。时间基准寄存器（WDTD）的高 8 位用作 WDT 的重装寄存器。

WDT 溢出时 WDTS 标志位置位，WDT 复位不能改变 WDTS 位的值。该标志位可通过软件写入 ‘1’ 清零。

图 19 所示为 WDT 方框图。看门狗定时器操作受到 2 个 SFR（WDTC 和 WDTD）的控制。空闲模式下，WDT 被暂时挂起，当出现中断时从空闲模式中恢复。

WDT 的溢出周期可由下式计算得出：

$$\text{周期} = (255 - \text{WDTD}) \times 344064 \times 1/f_{\text{CLK(XTAL1)}}$$

此处 WDTD 为 WDTD 寄存器的装载值，fosc 是振荡器频率。

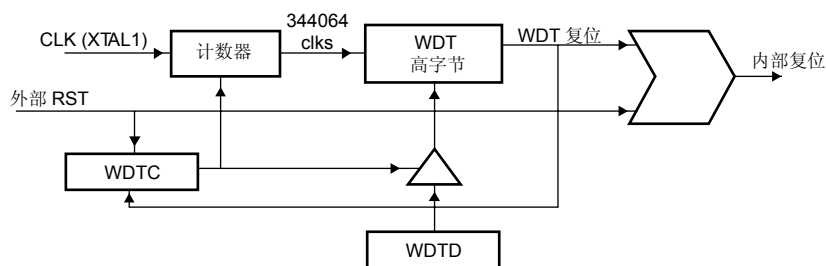


图 19 可编程看门狗定时器的方框图

表 33 WDTC—看门狗控制寄存器的位分配（地址：C0H）

可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	WDOUT	WDRE	WDTS	WDT	SWDT

表 34 WDTC—看门狗控制寄存器的位描述（地址：C0H）

位	符号	描述
7~5	-	保留为将来之用。用户程序将其清零。
4	WDOUT	看门狗输出使能。当该位和 WDRE 置位时，看门狗复位将驱动复位脚在 32 个时钟周期内有效。
3	WDRE	看门狗定时器复位使能。该位置位时，看门狗定时器复位。
2	WDTS	看门狗定时器复位标志。如果该位被置位，表明发生了 WDT 复位。复位由软件产生。
1	WDT	看门狗定时器刷新。该位通过软件置位，可强制产生一次 WDT 复位。
0	SWDT	起动看门狗定时器。该位被置位时起动 WDT；该位被清零时关闭 WDT。

### 7.8 可编程计数器阵列（PCA）

PCA 含有一个特殊的 16 位定时器，有 5 个 16 位的捕获/比较模块与之相连。每个模块可编程工作在 4 种模式下：上升/下降沿捕获、软件定时器、高速输出或脉宽调节器。每个模块都与 P1 口向连。模块 0 连接到 P1.3（CEX0），模块 1 连接到 P1.4（CEX1），依此类推。寄存器 CH 和 CL 的内容是正在自由递增计数的 16 位 PCA 定时器的当前值。PCA 定时器是 5 个模块的公共时间基准，可通过编程工作在：1/6 振荡频率、1/2 振荡频率、定时器 0 溢出或 ECI 脚的输入（P1.2）。定时器的计数源由 CMOD SFR 的 CPS1 和 CPS0 位来确定（见表 35 和表 36）。

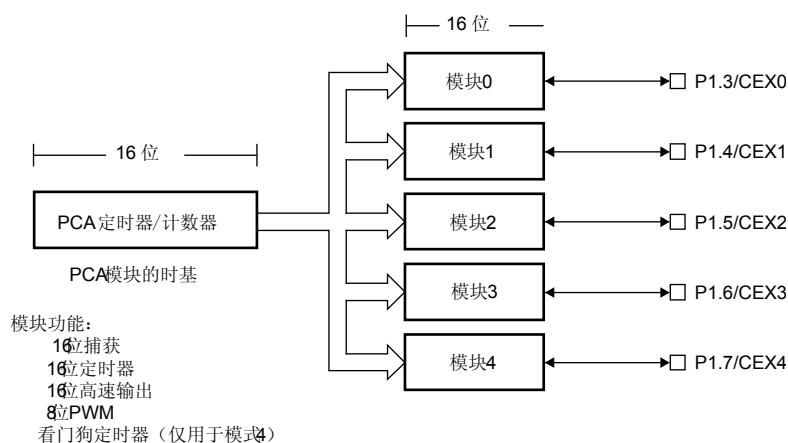


图 20 PCA 方框图

CMOD SFR 还有 3 位与 PCA 相关。它们分别是：CIDL，空闲模式下允许关闭 PCA；WDTE，使能或禁能模块 4 的看门狗功能；ECF，置位时产生中断和当 PCA 定时器溢出将 PCA 溢出标志 CF (CCON SFR) 置位。

看门狗定时器功能由 PCA 的模块 4 来实现。

CCON SFR 包含 PCA 的运行控制位 (CR) 和 PCA 定时器标志 (CF) 以及各个模块的标志 (CCF4:0)。通过软件置位 CR 位 (CCON.6) 来运行 PCA。CR 位被清零时 PCA 关闭。当 PCA 计数器溢出时，CF 位 (CCON.7) 置位，如果 CMOD 寄存器的 ECF 位置位，就产生中断。CF 位只可通过软件清除。CCON 寄存器的位 0~4 是 PCA 各个模块的标志 (位 0 对应模块 0，位 1 对应模块 2，依此类推)，当发生匹配或比较时由硬件置位。这些标志也只能通过软件清除。所有模块共用一个中断向量。PCA 的中断系统如图 21 所示。

PCA 的每个模块都对应一个特殊功能寄存器。它们分别是：模块 0 对应 CCAPM0，模块 1 对应 CCAPM1，依此类推。特殊功能寄存器包含了相应模块的工作模式控制位。

当模块发生匹配或比较时，ECCF 位 (CCAPMn.0, n=0, 1, 2, 3 或 4, 由工作的模块决定) 使能 CCON SFR 的 CCFn 标志来产生中断 (见图 21)。

PWM (CCAPMn.1) 用来使能脉宽调节模式。

当 PCA 计数值与模块的捕获/比较寄存器的值相匹配时，如果 TOG 位 (CCAPMn.2) 置位，模块的 CEX 输出将发生翻转。

当 PCA 计数值与模块的捕获/比较寄存器的值相匹配时，如果匹配位 MAT (CCAPMn.3) 置位，CCON 寄存器的 CCFn 位将被置位。

CAPN (CCAPMn.4) 和 CAPP (CCAPMn.5) 用来设置捕获输入的有效沿。CAPN 位使能下降沿有效，CAPP 位使能上升沿有效。如果两位都置位，则两种跳变沿都被使能，捕获可在两种跳变沿产生。

通过置位 CCAPMn 寄存器的最后一位 ECOM (CCAPMn.6) 来使能比较器功能。

每个 PCA 模块还对应另外两个寄存器，CCAPnH 和 CCAPnL。当出现捕获或比较时，它们用来保存 16 位的计数值。当 PCA 模块用在 PWM 模式中时，它们用来控制输出的占空比。



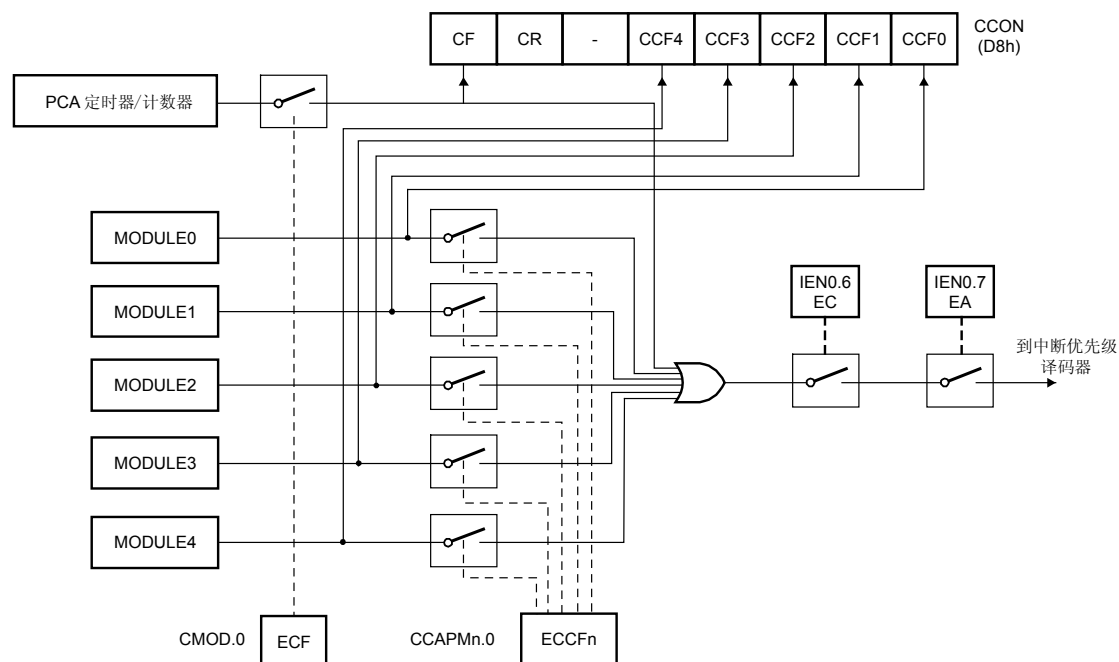


图 21 PCA 中断系统

表 35 CMOD—PCA 计数器方式寄存器的位分配 (地址: D9H)

不可位寻址; 复位值: 00H

位	7	6	5	4	3	2	1	0
符号	CIDL	WDTE	-	-	-	CPS1	CPS0	ECF

表 36 CMOD—PCA 计数器方式寄存器的位描述 (地址: D9H)

位	符号	描述
7	CIDL	计数器空闲控制: CIDL=0 时, 空闲模式下 PCA 计数器继续工作。CIDL=1 时, 空闲模式下 PCA 计数器禁能。
6	WDTE	看门狗定时器使能: WDTE=0 时, 禁止模块 4 的看门狗定时器功能。WDTE=1 时使能看门狗定时器。
5~3	-	保留为将来之用。通过用户程序将其清零。
2~1	CPS1,CPS0	PCA 计数脉冲选择 (见下面的表 37)。
0	ECF	PCA 使能计数溢出中断: ECF=1 时, 使能寄存器 CCON CF 位的中断。ECF=0 时, 禁止该功能。

表 37 CMOD—PCA 计数方式寄存器的计数脉冲选择 (地址: D9H)

CPS1	CPS0	选择 PCA 输入
0	0	0, 内部时钟, fosc/6
0	1	1, 内部时钟, fosc/6
1	0	2, 定时器 0 溢出
1	1	3, ECI/P1.2 脚的外部时钟 (最大速率=fosc/4)

表 38 CCON-PCON 计数器控制寄存器的位分配 (地址: 0D8H)

可位寻址; 复位值: 00H

位	7	6	5	4	3	2	1	0
符号	CF	CR	-	CCF4	CCF3	CCF2	CCF1	CCF0

表 39 CCON-PCA 计数器控制寄存器的位描述 (地址: 0D8H)

位	符号	描述
7	CF	PCA 计数器溢出标志。计数值翻转时该位由硬件置位。如果 CMOD 寄存器的 ECF 位置位, CF 标志可用来产生中断。CF 位可通过硬件或软件置位, 但只可通过软件清零。
6	CR	PCA 计数器运行控制位。该位通过软件置位, 用来起动 PCA 计数器计数。该位通过软件清零, 用来关闭 PCA 计数器。
5	-	保留为将来之用。通过用户程序将其清零。
4	CCF4	PCA 模块 4 中断标志。当出现匹配或捕获时该位由硬件置位。该位必须通过软件清零。
3	CCF3	PCA 模块 3 中断标志。当出现匹配或捕获时该位由硬件置位。该位必须通过软件清零。
2	CCF2	PCA 模块 2 中断标志。当出现匹配或捕获时该位由硬件置位。该位必须通过软件清零。
1	CCF1	PCA 模块 1 中断标志。当出现匹配或捕获时该位由硬件置位。该位必须通过软件清零。
0	CCF0	PCA 模块 0 中断标志。当出现匹配或捕获时该位由硬件置位。该位必须通过软件清零。

表 40 CCAPMn-PCA 模块比较/捕获寄存器 (CCAPM0 地址: 0DAH CCAPM1 地址: 0DBH CCAPM2 地址: 0DCH CCAPM3 地址: 0DDH CCAPM4 地址: 0DEH) 的位分配

不可位寻址; 复位值: 00H

位	7	6	5	4	3	2	1	0
符号	-	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn

表 41 CCAPMn-PCA 模块比较/捕获寄存器 (CCAPM0 地址: 0DAH CCAPM1 地址: 0DBH CCAPM2 地址: 0DCH CCAPM3 地址: 0DDH CCAPM4 地址: 0DEH) 的位描述

位	符号	描述
7	-	保留为将来之用。通过用户程序将其清零。
6	ECOMn	使能比较器。ECOMn=1 时使能比较器功能。
5	CAPPn	正捕获。CAPPn=1 时使能上升沿捕获。
4	CAPNn	负捕获。CAPNn=1 时使能下降沿捕获。
3	MATn	匹配。当 MATn=1 时, PCA 计数值与模块的比较/捕获寄存器的值的匹配将置位 CCON 寄存器的中断标志位 CCFn。
2	TOGn	翻转。当 TOGn=1 时, PCA 计数值与模块的比较/捕获寄存器的值的匹配将使 CEXn 脚翻转。
1	PWMn	脉宽调节模式。当 PWMn=1 时, 使能 CEXn 脚用作脉宽调节输出。
0	ECCFn	使能 CCF 中断。使能寄存器 CCON 的比较/捕获标志 CCFn, 用来产生中断。

表 42 PCA 模块工作模式 (CCAPMn 寄存器)

ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	模块功能
0	0	0	0	0	0	0	无此操作
X	1	0	0	0	0	X	16 位捕获模式, 由 CEXn 的上升沿触发
X	0	1	0	0	0	X	16 位捕获模式, 由 CEXn 的下降沿触发
X	1	1	0	0	0	X	16 位捕获模式, 由 CEXn 的跳变触发
1	0	0	1	0	0	X	16 位软件定时器
1	0	0	1	1	0	X	16 位高速输出
1	0	0	0	0	1	0	8 位 PWM
1	0	0	1	X	0	X	看门狗定时器

### 7.8.1 PCA 捕获模式

要使一个 PCA 模块工作在捕获模式 (图 22), 寄存器 CCAPM 的两位 (CAPN 和 CAPP) 或其中任何一位必须置 1。对模块的外部 CEX 输入 (P1 口) 的跳变进行采样。当采样到有效跳变时, PCA 硬件就将 PCA 计数器寄存器 (CH 和 CL) 的值装载到模块的捕获寄存器中 (CCAPnL 和 CCAPnH)。

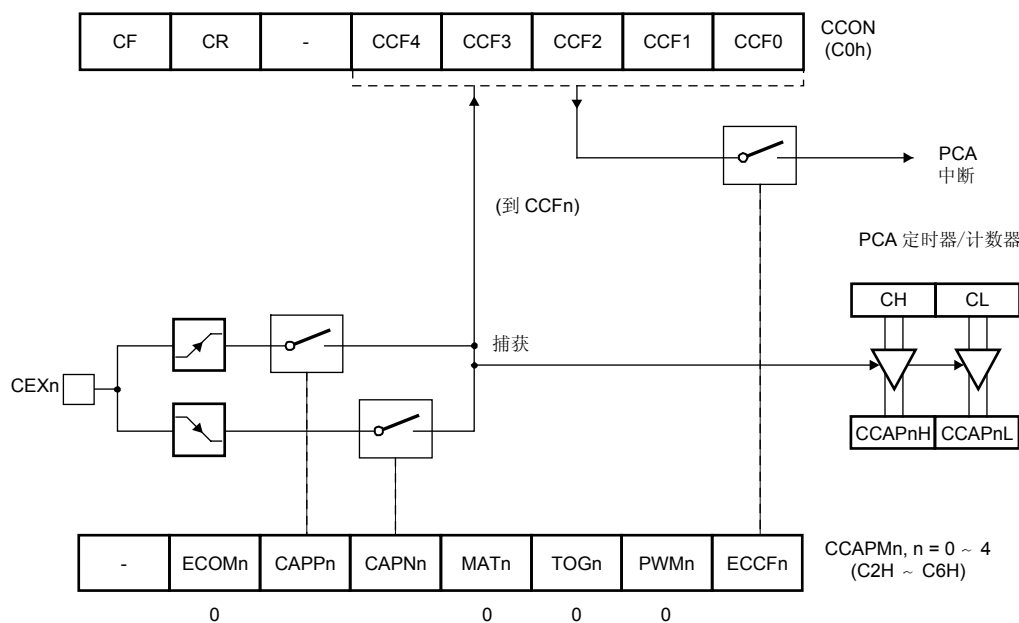


图 22 PCA 捕获模式

如果 CCON SFR 的位 CCFn 和 CCAPMn SFR 的位 ECCFn 位被置位, 将产生中断。

### 7.8.2 16 位软件定时器模式

通过置位 CCAPMn 寄存器的 ECOM 和 MAT 位, 可使 PCA 模块用作软件定时器 (见图 23)。PCA 定时器的值与模块捕获寄存器的值相比较, 当两者相等时, 如果位 CCFn (CCON SFR) 和位 ECCFn (CCAPMn SFR) 都置位, 将产生中断。

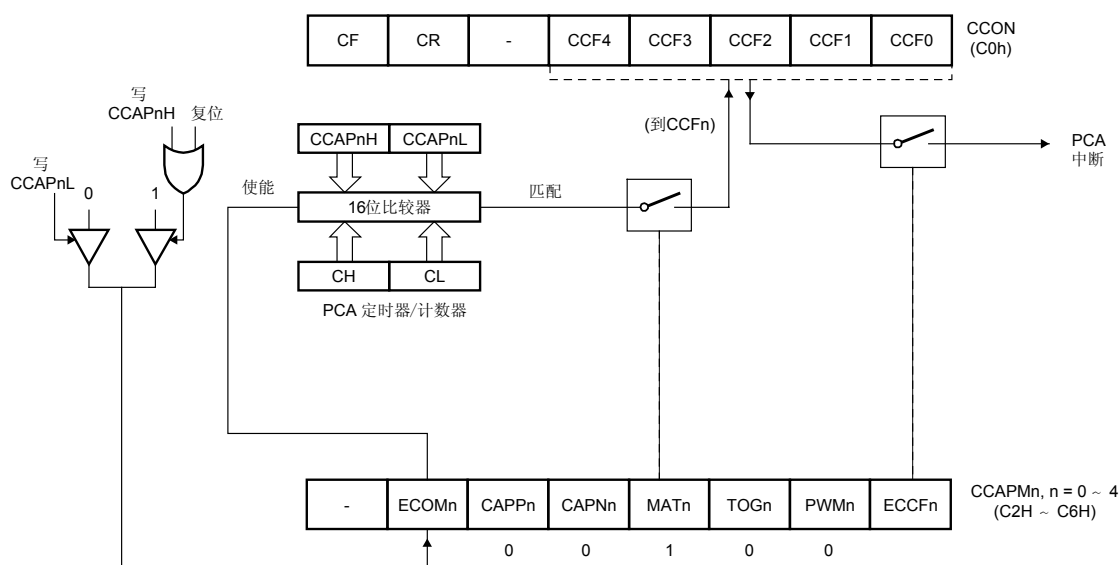


图 23 PCA 比较模式

### 7.8.3 高速输出模式

该模式中（图 24），当 PCA 计数器的计数值与模块捕获寄存器的值相匹配时，PCA 模块的 CEX 输出将发生翻转。要激活高速输出模式，模块 CCAPMn SFR 的 TOG, MAT 和 ECOM 位必须都置位。

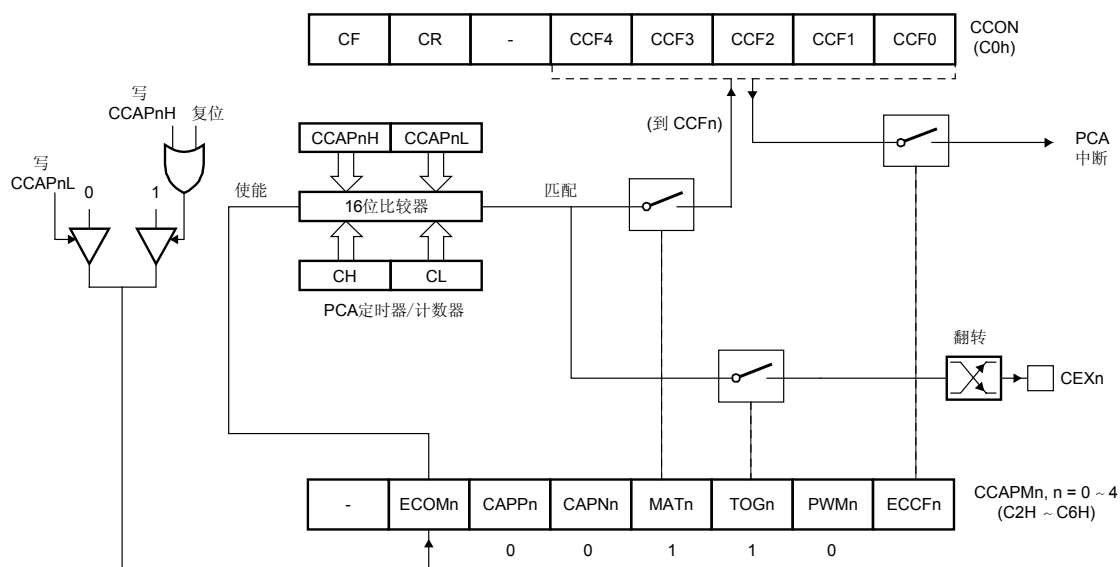


图 24 PCA 高速输出模式

### 7.8.4 脉宽调节模式

所有 PCA 模块都可用作 PWM 输出（图 25）。输出频率取决于 PCA 定时器的时钟源。

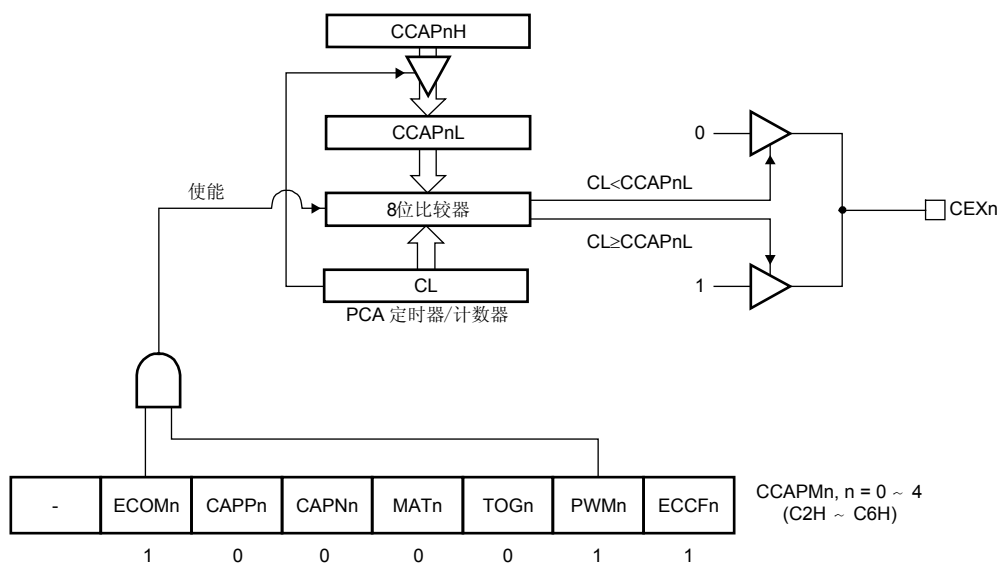


图 25 PCA PWM 模式

由于所有模块共用仅有的 PCA 定时器，所有它们的输出频率相同。各个模块的输出占空比是独立变化的，与使用的捕获寄存器 CCAPnL 有关。当 PCA CL SFR 的值小于 CCAPnL SFR 时，输出为低，当 PCA CL SFR 的值等于或大于 CCAPnL 时，输出为高。当 CL 的值由 FF 变为 00 溢出时，CCAPnH 的内容装载到 CCAPnL 中。这样就可实现无干扰地更新 PWM。要能使 PWM 模式，模块 CCAPm 寄存器的 PWM 和 ECOM 位必须置位。

### 7.8.5 PCA 看门狗定时器

PCA 还具有看门狗定时器功能，该功能的存在实现了在不增加芯片管脚的条件下系统可靠性的提高。看门狗定时器对于对噪声、电源干扰或静电放电敏感的系统很适用。模块 4 是 PCA 模块中惟一可编程用作看门狗的模块。而且，在不需要使用看门狗的情况下，该模块仍可工作在其它模式下。图 25 所示为看门狗工作原理图。比较寄存器的 16 位值由用户预先装载。与其它模式一样，这个 16 位的值将和 PCA 定时器的计数值相比较，如果两者匹配，则产生内部复位。但不会拉高 RST 脚。

用户软件必须定时地改变 (CCAP4H,CCAP4L) 的内容，防止它与 PCA 定时器 (CH,CL) 的内容相匹配。这段代码在下面的 WATCHDOG 程序中给出。

为了防止内部复位的产生，用户有 3 种选择：

1. 定时地改变比较的值，使之永远不和 PCA 定时器的内容相匹配。
2. 定时地改变 PCA 定时器的值，使之永远不和比较的内容相匹配。
3. 匹配发生前通过清除 WDTE 位先将看门狗禁能，然后再将其使能。

由于看门狗定时器并不能象 #3 选项那样被禁能，因此，前两种选择相对来说更可靠。如果程序计数器出错，匹配最终仍会产生，进而导致产生内部复位。当其它 PCA 模块正在使用时，建议大家也不要使用第二种方法，这是因为 PCA 定时器是所有模块的时间基准，为了实现模块 4 的看门狗定时器功能而改变其它模块的时间基准的做法并不明智。因此，在大多数应用中，第一种方法是最好的选择。

；定时地调用下面的 WATCHDOG 程序

```

CLR EA ; 禁能中断
MOV CCAP4L, #00 ;
MOV CCAP4H, CH
SETB EA ; 重新使能中断
RET
    
```

这个程序不能用作中断服务程序，因为一旦程序计数器出错而进入死循环，仍会对中断提供服务 and 产

生看门狗复位。这样，就不能达到使用看门狗定时器的目的。因此，这段程序应该在主程序中调用，在主程序中，PCA 定时器应当设置成  $2^{16}$  的计数工作模式（模式 1）。

### 7.9 保密位

保密位可用来防止软件被盗用和并行编程模式下未被授权团体对 flash 的读操作。它也可以避免内部 flash 存储器的偶然擦除和编程引起的代码混乱。

当保密位被激活后，除芯片擦除外的所有并行编程命令都被忽略（这样就不能对器件执行读操作）。但是，只要序列号及其长度未被编程，仍然可以执行用户代码的 ISP 读操作。所以，当用户要编程保密位时，编程器应当提示用户和程序需要一个序列号才可对器件执行操作。

### 7.10 复位

系统复位后，MCU 被初始化，程序从 0000H 单元开始执行。器件的复位输入是 RST 脚。要使器件复位，振荡器运行稳定后，RST 脚上的逻辑高电平至少要保持 2 个机器周期（24 个时钟周期）。复位时 ALE、 $\overline{\text{PSEN}}$  被微弱拉高。为了实现正确的复位，ALE 和  $\overline{\text{PSEN}}$  必须在复位过程中输出高电平。而且要求该电平不受外部元件的影响。当器件正在运行时，系统复位不会影响 1k 字节的片内 RAM 的内容，但是，片内 RAM 的内容在上电时是不确定的。

#### 7.10.1 上电复位

初始上电后，端口管脚可能是任何一种状态，直到振荡器稳定起振和内部复位逻辑将所有管脚微弱拉高。无有效复位的器件上电会使 MCU 从一个不确定的地址开始执行程序。这样未定义的状态必然会破坏 flash 的代码。

器件通电后，RST 脚上的高电平除了要保持有效上电复位所需的 2 个机器周期外，还要保持一段时间，以便振荡器能稳定起振（低频振荡器需要几个毫秒）。下面给出一个延长 RST 信号的方法：将 RST 脚通过一个 10uF 的电容器连接到  $V_{DD}$  和通过一个 8.2kΩ 的电阻连接到  $V_{SS}$ ，即搭建一个 RC 电路，如图 26 所示。注意：如果使用了 RC 电路，要确保  $V_{DD}$  的上升时间小于 1ms，振荡器的起振时间小于 10ms。

对于起振时间较长的低频振荡器来说，必须延长复位信号的时间以满足长时间起振的要求。要求解决这个问题的方法应当维护  $V_{DD}$  和 RST 之间的必然联系，以避免在一个不确定的地址上编程而造成 flash 内代码的破坏。可设计一个用作初始上电的上电检测电路，该电路在电压达到掉电检测电平之前工作。PCON 寄存器的 POF 标志用来指示初始上电的条件。POF 标志将一直有效，直到被软件清零。详情请参考有关 PCON 寄存器的描述。

复位后，P89LV51RD2 将进入 SoftICE 模式（如果之前已通过 ISP 命令将其使能）或尝试自动执行 ISP 引导装载程序。如果该自动执行操作在 400ms 后仍未成功，器件就开始执行用户代码。

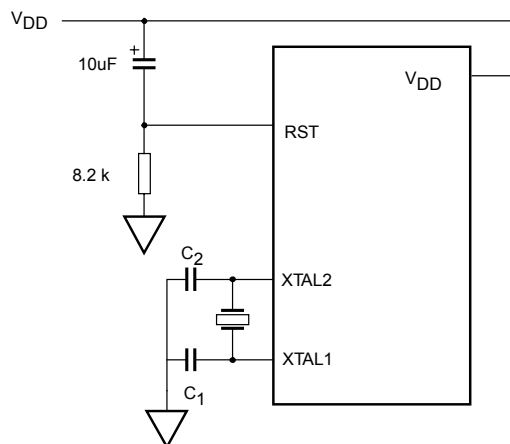


图 26 上电复位电路

### 7.10.2 软件复位

通过将 FCF[1] (SWR) 从 ‘0’ 变为 ‘1’ 来实现软件复位。软件复位后，程序计数器指向 0000H 地址。所有 SFR 寄存器都被设置成各自的复位值，但 FCF[1] (SWR)、WDTC[2] (WDTS) 和 RAM 数据将保持不变。

### 7.10.3 掉电检测复位

器件含有掉电检测电路，可保护系统免受电源电压  $V_{DD}$  大纹波的影响。P89LV51RD2 掉电检测的阈值为 2.35V。有关掉电电压的参数，请参考表 67 和表 68。

当  $V_{DD}$  下降到低于阈值电压时，掉电检测器将触发电路产生一个掉电中断，但 CPU 仍然继续运行，直至电压返回到掉电检测电压  $V_{BOD}$ 。掉电检测的默认操作是产生一次处理器复位。

在掉电检测电路响应前， $V_{DD}$  必须保持至少 4 个振荡器周期的低于  $V_{BOD}$  的状态。

通过置位 IEA 寄存器的 EBO 位（地址：E8H，位 3）来使能掉电中断。如果 EBO 位置位并且检测到掉电条件，则产生掉电中断，程序从 004BH 单元开始执行。EBO 位必须在进入中断服务程序后通过软件清零。掉电条件有效时清除 EBO 位将使器件正确复位。如果掉电中断未被使能，掉电条件将复位程序，使程序恢复到从 0000H 单元开始执行。

### 7.10.4 中断优先级和查询序列

器件支持 8 中断源，4 中断优先级的中断结构。表 43 总结了支持中断的查询序列。注意：SPI 串行接口和 UART 共用同一个中断向量（见图 27）。

表 43 中断查询序列

描述	中断标志	向量地址	中断使能	中断优先级	服务优先级	掉电唤醒
外部中断 0	IE0	0003H	EX0	PX0/H	1 (最高)	能
掉电	-	004BH	EBO	PBO/H	2	不能
T0	TF0	000BH	ET0	PT0/H	3	不能
外部中断 1	IE1	0013H	EX1	PX1/H	4	能
T1	TF1	001BH	ET1	PT1/H	5	不能
PCA	CF/CCFn	0033H	EC	PPCH	6	不能
UART/SPI	TI/RI/SPIF	0023H	ES	PS/H	7	不能
T2	TF2,EXF2	002BH	ET2	PT2/H	8	不能

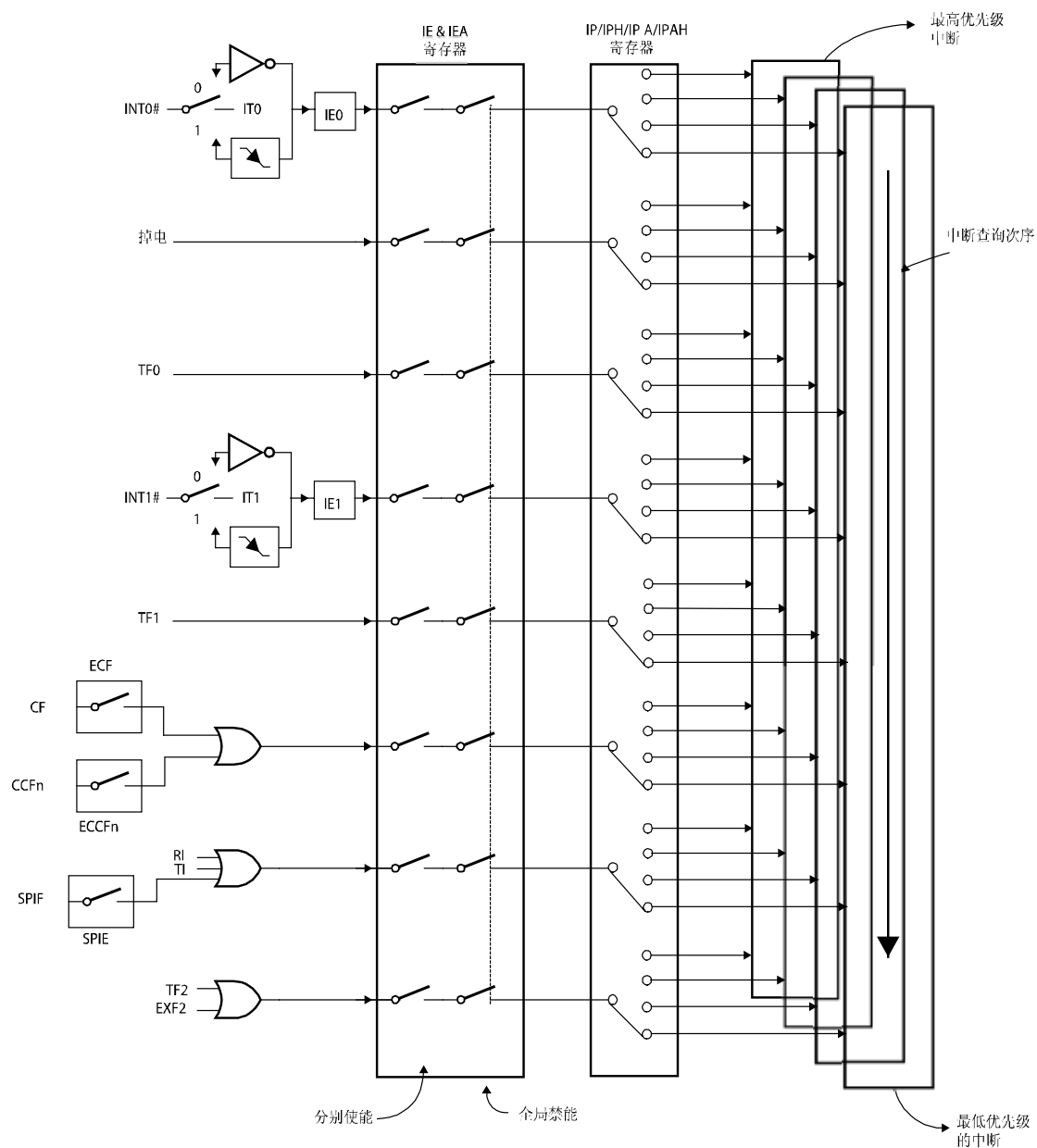


表 27 中断结构

表 44 IEN0—中断使能寄存器 0 的位分配 (地址: A8H)

可位寻址; 复位值: 00H

位	7	6	5	4	3	2	1	0
符号	EA	EC	ET2	ES	ET1	EX1	ET0	EX0



表 45 IEN0—中断使能寄存器 0 的位描述（地址：A8H）

位	符号	描述
7	EA	中断使能位：EA=1，允许中断服务；EA=0，禁能中断服务。
6	EC	PCA 中断使能位。
5	ET2	定时器 2 中断使能。
4	ES	串口中断使能。
3	ET1	定时器 1 溢出中断使能。
2	EX1	外部中断 1 使能。
1	ET0	定时器 0 溢出中断使能。
0	EX0	外部中断 0 使能。

表 46 IEN1—中断使能寄存器 1 的位分配（地址：E8H）

可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	-	EBO	-	-	-

表 47 IEN1—中断使能寄存器 1 的位描述（地址：E8H）

位	符号	描述
7~4	-	保留为将来之用。通过用户程序将其清零。
3	EBO	掉电中断使能。1=使能，0=禁能。
2~0	-	保留为将来之用。通过用户程序将其清零。

表 48 IP0—中断优先级 0 低字节寄存器的位分配（地址：B8H）

可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	PPC	PT2	PS	PT1	PX1	PT0	PX0

表 49 IP0—中断优先级 0 低字节寄存器的位描述（地址：B8H）

位	符号	描述
7	-	保留为将来之用。通过用户程序将其清零。
6	PPC	PCA 中断优先级低位。
5	PT2	定时器 2 中断优先级低位。
4	PS	串口中断优先级低位。
3	PT1	定时器 1 中断优先级低位。
2	PX1	外部中断 1 优先级低位。
1	PT0	定时器 0 中断优先级低位。
0	PX0	外部中断 0 优先级低位。

表 50 IP0H—中断优先级 0 高字节寄存器的位分配（地址：B7H）

不可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

表 51 IP0H—中断优先级 0 高字节寄存器的位描述（地址：B7H）

位	符号	描述
7	-	保留为将来之用。通过用户程序将其清零。
6	PPCH	PCA 中断优先级高位。
5	PT2H	定时器 2 中断优先级高位。
4	PSH	串口中断优先级高位。
3	PT1H	定时器 1 中断优先级高位。
2	PX1H	外部中断 1 优先级高位。
1	PT0H	定时器 0 中断优先级高位。
0	PX0H	外部中断 0 优先级高位。

表 52 IP1—中断优先级 1 寄存器的位分配（地址：F8H）

可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	PBO	-	-	-	-

表 53 IP1—中断优先级 1 寄存器的位描述（地址：F8H）

位	符号	描述
7~5	-	保留为将来之用。通过用户程序将其清零。
4	PBO	掉电中断优先级位。
3~0	-	保留为将来之用。通过用户程序将其清零。

表 54 IP1H—中断优先级 1 高字节寄存器的位分配（地址：F7H）

不可位寻址；复位值：00H

位	7	6	5	4	3	2	1	0
符号	-	-	-	PBOH	-	-	-	-

表 55 IP1H—中断优先级 1 高字节寄存器的位描述（地址：F7H）

位	符号	描述
7~5	-	保留为将来之用。通过用户程序将其清零。
4	PBOH	掉电中断优先级位。
3~0	-	保留为将来之用。通过用户程序将其清零。

## 7.11 节电模式

器件提供了 2 种节电模式，用在将功耗作为重要因素的应用中。这两种节电模式是空闲模式和掉电模式，见表 56。

### 7.11.1 空闲模式

通过置位 PCON 寄存器的 IDL 位来进入空闲模式。在空闲模式下，程序计数器（PC）停止工作。系统时钟继续运行，所有中断和外围功能仍然有效。片内 RAM 和特殊功能寄存器的内容在该模式下保持不变。

系统复位或硬件复位都可将器件从空闲模式唤醒。如果通过系统中断来退出空闲模式，中断开始时就立即将 IDL 位清零，使器件退出空闲模式。退出中断服务程序后，被中断的程序立刻恢复到唤醒空闲模式指令的下条指令开始执行。也可通过硬件复位来重新启动器件，方法与上电复位类似。

### 7.11.2 掉电模式

通过置位 PCON 寄存器的 PD 位来进入掉电模式。在掉电模式下，时钟停止运行，只有电平触发的外部中断有效。在掉电模式下，SRAM 的内容保持不变，V<sub>DD</sub> 的最低电平为 2.0V。

通过使能电平触发的外部中断或硬件复位来将器件从掉电模式中唤醒。中断开始时立即将 PD 位清零，使器件退出掉电模式。外部中断管脚上的低电平可以使振荡器重新启动，但该信号的低电平必须保持至少 1024 个时钟周期再返回到高电平，从而使器件退出掉电模式。当中断信号恢复到 V<sub>IH</sub> 时，中断服务程序立刻恢复到唤醒掉电模式指令的下条指令开始执行。也可通过硬件复位来重新启动器件，方法与上电复位类似。

为了使器件正确地退出掉电模式，复位或外部中断的操作必须在 V<sub>DD</sub> 恢复到正常工作电压之后执行。为了重新启动振荡器并使之稳定运行，要求 V<sub>DD</sub> 保持在其正常工作电平足够长的时间（通常大于 10ms）。

表 56 节电模式

模式	起动	MCU 的状态	退出
空闲模式	软件方式 (置位 PCON 的 IDL 位) MOV PCON,#01H	该模式下，CLK 继续运行；中断、串口和定时器/计数器有效；程序计数器停止工作；ALE 和 $\overline{\text{PSEN}}$ 信号为高电平；所有寄存器的内容保持不变。	通过使能中断或硬件复位可将器件从空闲模式唤醒，中断开始时清除 IDL 位，使器件退出空闲模式，执行完 ISR RETI 指令后，程序恢复到唤醒空闲模式指令的下条指令开始执行。用户可以考虑在唤醒空闲模式的指令后添加两三条 NOP 指令来消除任何情况出现带来的影响。也可通过硬件复位来重新启动器件，方法与上电复位类似。
掉电模式	软件方式 (置位 PCON 的 PD 位) MOV PCON,#02H	该模式下，CLK 停止运行；片内 SRAM 和 SFR 的内容保持不变；ALE 和 $\overline{\text{PSEN}}$ 信号为低电平；只有电平触发的外部中断有效（如果中断被使能）。	通过使能电平触发的外部中断或硬件复位可将器件从掉电模式唤醒。中断开始时清除 PD 位，使器件退出掉电模式。执行完 ISR RETI 指令后，程序恢复到唤醒掉电模式指令的下条指令开始执行。用户可以考虑在唤醒掉电模式的指令后添加两三条 NOP 指令来消除任何情况出现带来的影响。也可通过硬件复位来重新启动器件，方法与上电复位类似。

## 7.12 系统时钟和时钟选项

### 7.12.1 时钟输入选项和振荡器电容值的选取

图 28 所示为内部反相放大器的输入和输出 (XTAL1, XTAL2)，通过配置，它们可用作片内振荡器。当器件使用外部时钟源时，XTAL2 脚悬空，XTAL1 脚被驱动。

晶振开始起振时，由于放大器和放大器本身的反馈电容之间的相互作用，外部振荡器的 XTAL1 脚会出现一个较高的容性负载。但是，只要外部信号满足 V<sub>IL</sub> 和 V<sub>IH</sub> 的要求，振荡器的电容将不会超过 15pF。

晶振生产商、供给电压以及其它因素都将影响电路特性，使之在各个应用中都不同。在每个设计中，C1 和 C2 都要调节到一个合适的值。表 57 给出了不同晶振类型对应的 C1 和 C2 典型值（可得到不同的频

率)。

表 57 不同晶振类型对应的 C1 和 C2 值

晶体	C1=C2
石英晶体	20pF~30pF
陶瓷晶体	40pF~50pF

更多有关片内振荡器设计的内容请参考 *FlashFlex51 振荡器电路设计精要* 应用笔记。

### 7.12.2 时钟加倍模式选项

通常情况下，器件每个机器周期运行 12 个时钟 (X1 模式)。器件含有一个时钟加倍选项，可以加速器件的运行速度，每个机器周期运行 6 个时钟 (见表 58)。时钟加倍模式可通过外部编程器或 IAP 来实现。当该模式被选择时，FST 寄存器的 EDC 位用来指示 6 时钟模式。

时钟加倍模式只可用于加倍内部系统时钟和内部 flash 存储器 (即  $\overline{EA}=1$ )。在访问外部存储器和外围器件时要特别小心，还要注意晶振的输出 (XTAL2) 是不能加倍的。



图 28 振荡器特性

表 58 时钟加倍特性

器件	标准模式 (x1)		时钟加倍模式 (x2)	
	每个机器周期的时钟数	最大外部时钟频率 (MHz)	每个机器周期的时钟数	最大外部时钟频率 (MHz)
P89LV51RD2	12	33	6	16

表 59 FST—Flash 状态寄存器的位分配 (地址: B6H)

不可位寻址; 复位值: xxxxx0xxB

位	7	6	5	4	3	2	1	0
符号	-	SB	-	-	EDC	-	-	-

表 60 FST—Flash 状态寄存器的位描述 (地址: B6H)

位	符号	描述
7	-	保留为将来之用。通过用户程序将其清零。
6	SB	保密位。
5~4	-	保留为将来之用。通过用户程序将其清零。
3	EDC	使能加倍时钟。
2~0	-	保留为将来之用。通过用户程序将其清零。

## 8. 极限参数

表 61 极限参数

遵循绝对最大额定系统规范 (IEC 60134)

除非特别说明, 表中参数在工作温度范围内有效。所有电压都以  $V_{SS}$  为参考点。

符号	参数	条件	最小	最大	单位
Tamb(bias)	工作环境温度		-55	+125	°C
Tstg	存储温度范围		-65	+150	°C
$V_{\overline{EA}}$	$\overline{EA}$ 管脚电压 (相对 $V_{SS}$ )		-0.5	14	V
Vn	任意管脚对地的 DC 电压		-0.5	$V_{DD}+0.5$	V
Vit	其它管脚到 $V_{SS}$ 的瞬时电压 (<20ns)		-1.0	$V_{DD}+1.0$	V
$I_{OL(I/O)}$	I/O 口 P1.5,P1.6,P1.7 的最大 $I_{OL}$		-	20	mA
$I_{OL(I/O)}$	其它 I/O 口的最大 $I_{OL}$		-	15	mA
Ptot(pack)	每种封装的总功耗	Tamb=25°C	-	1.5	W
	通孔焊接温度	10 秒	-	300	°C
	表面安装焊接温度	3 秒	-	240	°C
	输出短路电流	[1]	-	50	mA

[1] 输出短路时间不能超过 1s。一次只能发生一个输出短路。(基于封装热传递的限制, 而非器件功耗。)

## 9. 建议工作条件

表 62 工作范围

符号	描述	最小	最大	单位
Tamb	环境温度			
	商业级	0	+70	°C
	工业级	-40	+85	°C
$V_{DD}$	电源电压	2.7	3.6	V
fosc	振荡器频率	0	33	MHz
	应用编程时的振荡器频率	0.25	33	MHz

表 63 可靠性

符号	参数	最小规范	单位	测试方法
$N_{END}^{[1]}$	耐久性	10,000	周期	JEDEC Standard A117
$T_{DR}^{[1]}$	数据保持	100	年	JEDEC Standard A103
$I_{LTH}^{[1]}$	锁定	$100+I_{DD}$	mA	JEDEC Standard 78

[1] 该参数在初始条件限制下测得, 设计或操作改变后参数值将受影响。

表 64 AC 测试条件<sup>[1]</sup>

输入上升/下降时间	10ns
输出负载	$C_L=100\text{pf}$

[1] 见图 35 和图 37。

表 65 建议系统上电时序

符号	参数	最小	单位
$T_{PU-READ}^{[1]}$	上电到读操作	100	us
$T_{PU-WRITE}^{[1]}$	上电到写操作	100	us

[1] 该参数在初始条件限制下测得，设计或操作改变后参数值将受影响。

表 66 管脚阻抗

( $V_{DD}=3.3V$ ,  $T_{amb}=25^{\circ}C$ ,  $f=1MHz$ , 其它管脚开路)

参数	描述	测试条件	最大	单位
$C_{I/O}^{[1]}$	I/O 口电容	$V_{I/O}=0V$	15	pF
$C_{IN}^{[1]}$	输入电容	$V_{IN}=0V$	12	pF
$L_{PIN}$	管脚电感		20	nH

[1] 该参数在初始条件限制下测得，设计或操作改变后参数值将受影响。

## 10. 静态特性

表 67 DC 电气特性

$T_{amb}=0^{\circ}C \sim +70^{\circ}C$  或  $-40^{\circ}C \sim +85^{\circ}C$ ;  $V_{DD}=2.7V \sim 3.6V$ ;  $V_{SS}=0V$

符号	参数	条件	最小	最大	单位
$V_{IL}$	低电平输入电压	$2.7V < V_{DD} < 3.6V$	-0.5	0.7	V
$V_{IH}$	高电平输入电压	$2.7V < V_{DD} < 3.6V$	$0.2V_{DD} + 0.9$	$V_{DD} + 0.5$	V
$V_{IH1}$	高电平输入电压 (XTAL1,RST)	$2.7V < V_{DD} < 3.6V$	$0.7V_{DD}$	$V_{DD} + 0.5$	V
$V_{OL}$	低电平输出电压(P1.5,P1.6,P1.7)	$V_{DD}=2.7V; I_{OL}=16mA$	-	1.0	V
$V_{OL}$	低电平输出电压(P1,P2,P3) <sup>[1]</sup>	$V_{DD}=2.7V$			
		$I_{OL}=100\mu A$	-	0.3	V
		$I_{OL}=1.6mA$	-	0.45	V
		$I_{OL}=3.5mA$	-	1.0	V
$V_{OL1}$	低电平输出电压(P0 口, $\overline{ALE}$ , $\overline{PSEN}$ ) <sup>[1][3]</sup>	$V_{DD}=2.7V$			
		$I_{OL}=200\mu A$	-	0.3	V
		$I_{OL}=3.2mA$	-	0.45	V
$V_{OH}$	高电平输出电压 (P1,P2,P3, $\overline{ALE}$ , $\overline{PSEN}$ ) <sup>[4]</sup>	$V_{DD}=2.7V$			
		$I_{OH}=-10\mu A$	$V_{DD}-0.3$	-	V
		$I_{OH}=-30\mu A$	$V_{DD}-0.7$	-	V
$V_{OH1}$	高电平输出电压 (外部总线模式的 P0 口) <sup>[4]</sup>	$V_{DD}=2.7V$			
		$I_{OH}=-200\mu A$	$V_{DD}-0.3$	-	V
		$I_{OH}=-3.2mA$	$V_{DD}-0.7$	-	V
$V_{BOD}$	掉电检测电压		2.35	2.55	V
$I_{IL}$	逻辑 0 输入电流 (P1,P2,P3)	$V_{IN}=0.4V$	-	-75	$\mu A$
$I_{TL}$	逻辑 1 到 0 的跳变电流 (P1,P2,P3) <sup>[5]</sup>	$V_{IN}=2V$	-	-650	$\mu A$
$I_{LI}$	输入漏电流 (P0)	$0.45V < V_{IN} < V_{DD}-0.3V$	-	$\pm 10$	$\mu A$
$R_{RST}$	RST 下拉电阻		-	225	k $\Omega$
$C_{IO}$	管脚电容 <sup>[6]</sup>	@1MHz, $T_{amb}=25^{\circ}C$	-	15	pF

续上表.....

符号	参数	条件	最小	最大	单位
I <sub>DD</sub>	电源电流				
	激活模式	@12MHz	-	11.5	mA
		@33MHz	-	30	mA
	空闲模式	@12MHz	-	8.5	mA
		@33MHz	-	21	mA
	掉电模式 (最小 V <sub>DD</sub> =2V)	T <sub>amb</sub> =0°C~+70°C	-	45	uA
T <sub>amb</sub> =-40°C~+85°C		-	55	uA	

[1] 稳态 (非瞬态) 条件下, I<sub>OL</sub> 必须受到如下限制:

- a) 每个 8 位 I/O 口的最大 I<sub>OL</sub>: 26mA
- b) 所有输出的最大总 I<sub>OL</sub>: 71mA
- c) 如果 I<sub>OL</sub> 超出测试条件, V<sub>OH</sub> 也会超出相应的规定。不能保证管脚提供的灌电流大于表中测试条件的列出值。

[2] P0 和 P2 口的容性负载会在 ALE、P1 和 P3 的 V<sub>OL</sub> 上产生叠加的寄生噪声。噪声是由于总线操作过程中管脚发生 1 到 0 的跳变时外部总线对 P0 和 P2 口的放电产生的。最坏情况下 (容性负载>100pF) ALE 管脚上的噪声脉冲可能大于 0.8V。这时, 必须利用施密特触发器或带施密特触发器选通输入的地址锁存器对 ALE 脚的信号进行限制。

[3] P0 口、ALE 和 PSEN 的负载电容=100pF, 其它输出的负载电容=80pF。

[4] 当地址位固定时, P0 和 P2 口的容性负载将使 ALE 和 PSEN 的 V<sub>OH</sub> 暂时低于规定的 V<sub>DD</sub>-0.7。

[5] 当 P1、P2 和 P3 口的电平被外部器件从 1 拉到 0 时, 由 3 个 I/O 口来提供跳变电流。当 V<sub>IN</sub> 接近 2V 时跳变电流达到最大值。

[6] 管脚电容是一个特性值, 并非测试所得。EA =25pF(max)。

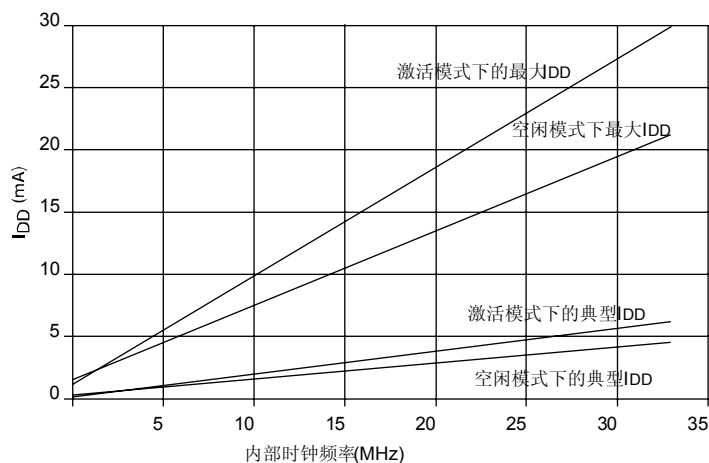


图 29 I<sub>DD</sub> 和频率的对应关系

## 11. 动态特性

表 68 AC 特性

工作条件: P0 口、ALE 和  $\overline{\text{PSEN}}$  的负载电容=100pF; 其它输出的负载电容=80pF。

$T_{\text{amb}}=0^{\circ}\text{C}\sim+70^{\circ}\text{C}$ 或 $-40^{\circ}\text{C}\sim+85^{\circ}\text{C}$ ;  $V_{\text{DD}}=2.7\text{V}\sim3.6\text{V}@33\text{MHz}$ ;  $V_{\text{SS}}=0\text{V}$

符号	参数	33MHz(X1 模式)		可变		单位
		16MHz(X2 模式) <sup>[1]</sup>		最小	最大	
		最小	最大			
$1/T_{\text{CLCL}}$	X1 模式振荡器频率	0	33	0	33	MHz
$1/2T_{\text{CLCL}}$	X2 模式振荡器频率	0	16	0	16	MHz
$t_{\text{LHLL}}$	ALE 脉宽	46	-	$2T_{\text{CLCL}}-15$	-	ns
$t_{\text{AVLL}}$	地址有效到 ALE 低	5	-	$T_{\text{CLCL}}-25$	-	ns
$t_{\text{LLAX}}$	ALE 变低后地址保持	5	-	$T_{\text{CLCL}}-25$	-	ns
$t_{\text{LLIV}}$	ALE 低到有效指令输入	-	56	-	$4T_{\text{CLCL}}-65$	ns
$t_{\text{LLPL}}$	ALE 低到 $\overline{\text{PSEN}}$ 低	5	-	$T_{\text{CLCL}}-25$	-	ns
$t_{\text{PLPH}}$	$\overline{\text{PSEN}}$ 脉宽	66	-	$T_{\text{CLCL}}-25$	-	ns
$t_{\text{PLIV}}$	$\overline{\text{PSEN}}$ 低到有效指令输入	-	35	-	$3T_{\text{CLCL}}-55$	ns
$t_{\text{PXIX}}$	$\overline{\text{PSEN}}$ 后输入指令保持	-	-	0	-	ns
$t_{\text{PXIZ}}$	$\overline{\text{PSEN}}$ 后输入指令悬浮	-	25	-	$T_{\text{CLCL}}-5$	ns
$t_{\text{PXAV}}$	$\overline{\text{PSEN}}$ 到地址有效	22	-	$T_{\text{CLCL}}-8$	-	ns
$t_{\text{AVIV}}$	地址到有效地址输入	-	72	-	$5T_{\text{CLCL}}-80$	ns
$t_{\text{PLAZ}}$	$\overline{\text{PSEN}}$ 低到地址悬浮	-	10	-	10	ns
$t_{\text{RLRH}}$	$\overline{\text{RD}}$ 脉宽	142	-	$6T_{\text{CLCL}}-40$	-	ns
$t_{\text{WLWH}}$	写脉宽( $\overline{\text{WR}}$ )	142	-	$6T_{\text{CLCL}}-40$	-	ns
$t_{\text{RLDV}}$	$\overline{\text{RD}}$ 低到有效数据输入	-	62	-	$5T_{\text{CLCL}}-90$	ns
$t_{\text{RHDX}}$	$\overline{\text{RD}}$ 后的数据保持	0	-	0	-	ns
$t_{\text{RHDZ}}$	$\overline{\text{RD}}$ 后的数据悬浮	-	36	-	$2T_{\text{CLCL}}-25$	ns
$t_{\text{LLDV}}$	ALE 低到有效数据输入	-	152	-	$8T_{\text{CLCL}}-90$	ns
$t_{\text{AVDV}}$	地址到有效数据输入	-	183	-	$9T_{\text{CLCL}}-90$	ns
$t_{\text{LLWL}}$	ALE 低到 $\overline{\text{RD}}$ 或 $\overline{\text{WR}}$ 低	66	116	$3T_{\text{CLCL}}-25$	$3T_{\text{CLCL}}+25$	ns
$t_{\text{AVWL}}$	地址到 $\overline{\text{RD}}$ 或 $\overline{\text{WR}}$ 低	46	-	$4T_{\text{CLCL}}-75$	-	ns
$t_{\text{WHQX}}$	$\overline{\text{WR}}$ 后的数据保持	3	-	$T_{\text{CLCL}}-27$	-	ns
$t_{\text{QVWH}}$	数据有效到 $\overline{\text{WR}}$ 高	142	-	$7T_{\text{CLCL}}-70$	-	ns
$t_{\text{RLAZ}}$	$\overline{\text{RD}}$ 低到地址悬浮	-	0	-	0	ns
$t_{\text{WHLH}}$	$\overline{\text{RD}}$ 到 $\overline{\text{WR}}$ 高到 ALE 高	5	55	$T_{\text{CLCL}}-25$	$T_{\text{CLCL}}+25$	ns

[1] 由 X1 模式计算所得。

### 11.1 符号说明

每个时序符号包含 5 个字母。第一个字母通常是 ‘T’ (代表时间)。其它字母根据所处的位置来代表信号的名称或此信号的逻辑状态。下面列出了所有的字母以及它们代表的含意。

- A—地址
- C—时钟
- D—输入数据
- H—逻辑高电平



- I—指令（程序存储器的内容）
- L—逻辑低电平或 ALE
- P— $\overline{\text{PSEN}}$
- Q—输出数据
- R— $\overline{\text{RD}}$  信号
- T—时间
- V—有效
- W— $\overline{\text{WR}}$  信号
- X—非有效逻辑电平
- Z—高阻抗（悬浮）

例如：

$T_{\text{AVLL}}$  = 地址有效到 ALE 低的时间

$T_{\text{LLPL}}$  = ALE 低到  $\overline{\text{PSEN}}$  低的时间

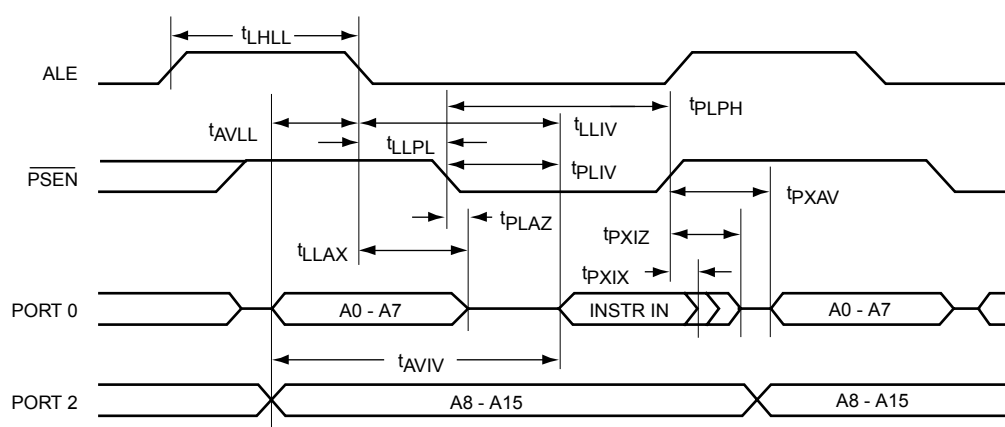


图 30 外部程序存储器读周期

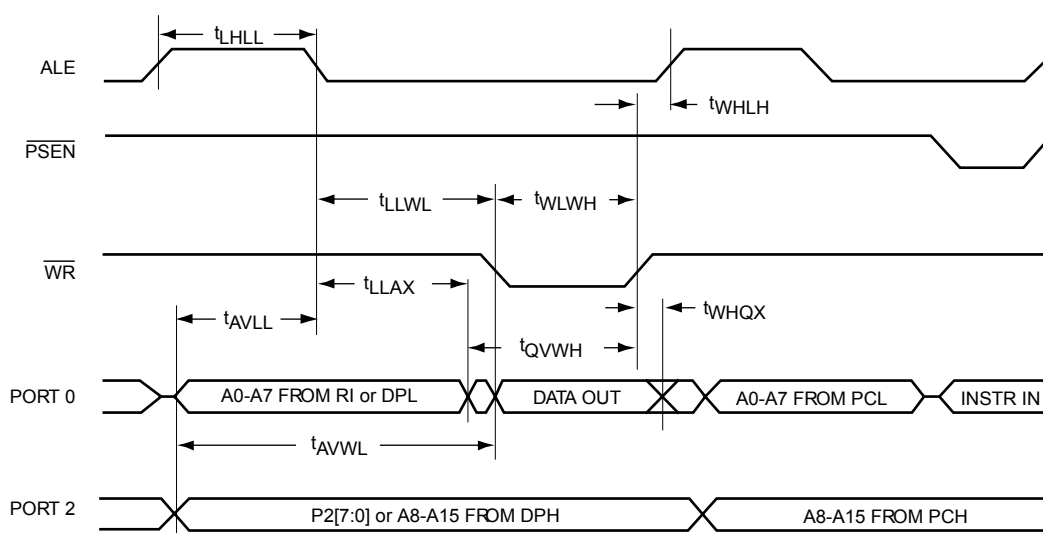


图 31 外部数据存储器读周期

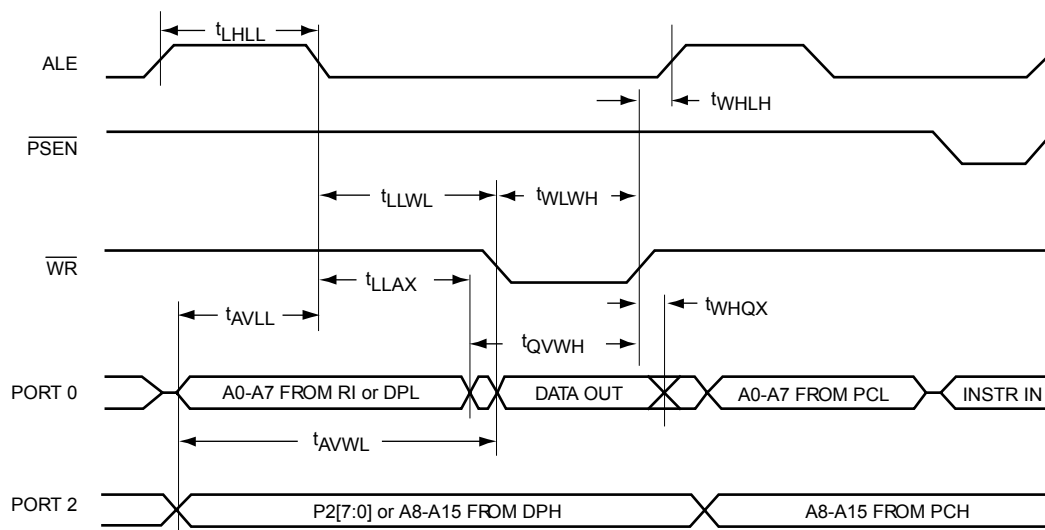


图 32 外部数据存储写周期

表 69 外部时钟驱动

符号	参数	振荡器				单元
		12MHz		可变		
		最小	最大	最小	最大	
$1/T_{CLCL}$	振荡器频率	-	-	0	33	MHz
$t_{CLCL}$		83	-	-	-	ns
$t_{CHCX}$	高电平时间	-	-	$0.35T_{CLCL}$	$0.65T_{CLCL}$	ns
$t_{CLCX}$	低电平时间	-	-	$0.35T_{CLCL}$	$0.65T_{CLCL}$	ns
$t_{CLCH}$	上升时间	-	20	-	-	ns
$t_{CHCL}$	下降时间	-	20	-	-	ns

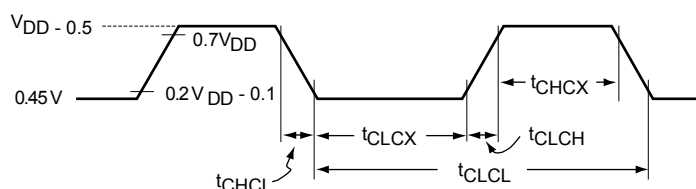


图 33 外部时钟驱动波形

表 70 串口时序

符号	参数	振荡器				单元
		12MHz		可变		
		最小	最大	最小	最大	
$t_{XLXL}$	串口时钟周期	1.0	-	$12t_{CLCL}$	-	us
$t_{QVXH}$	输出数据建立时钟上升沿	700	-	$10t_{CLCL} - 133$	-	ns
$t_{XHQX}$	时钟上升沿后的输出数据保持	50	-	$2t_{CLCL} - 117$ $2t_{CLCL} - 50$	-	ns
$t_{XHDX}$	时钟上升沿后的输入数据保持	0	-	0	-	ns
$t_{XHDX}$	时钟上升沿到输入数据有效	-	700	-	$10t_{CLCL} - 133$	ns

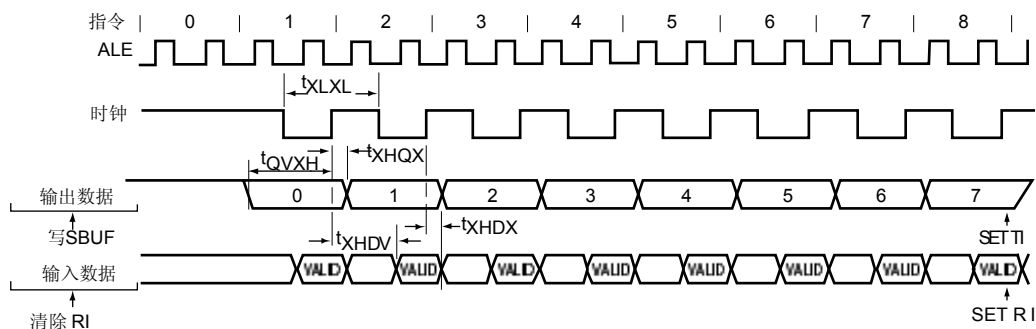
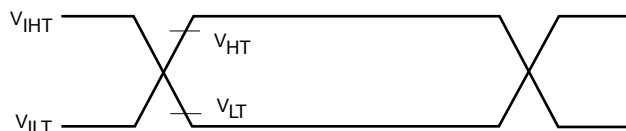


图 34 移位寄存器模式时序波形



注:  $V_{HT}$ -V 高电平测试  
 $V_{LT}$ -V 低电平测试  
 $V_{IHT}$ -V 输入高电平测试  
 $V_{ILT}$ -V 输入低电平测试

测试过程中，当 AC 输入被驱动到  $V_{IHT}(V_{DD}-0.5V)$  时为逻辑 1 电平； $V_{ILT}(0.45V)$  时为逻辑 0 电平。输入和输出的测试参考点分别为  $V_{HT}(0.2V_{DD}+0.9)$  和  $V_{LT}(0.2V_{DD}-0.1)$ 。

图 35 AC 测试输入/输出测试波形



出于时序的考虑，端口管脚不再在负载电压发生 100mV 变化时悬浮，而是开始在  $V_{OH}/V_{OL}$  电平发生 100mV 的变化时悬浮。 $V_{OH}/V_{OL} = \pm 20mA$ 。

图 36 悬浮波形

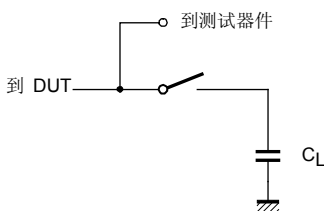
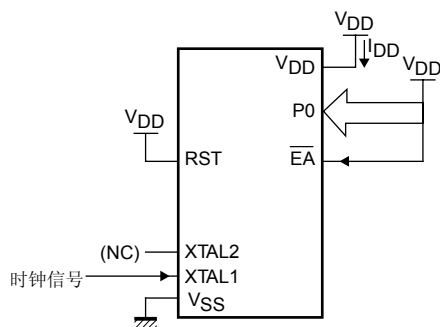
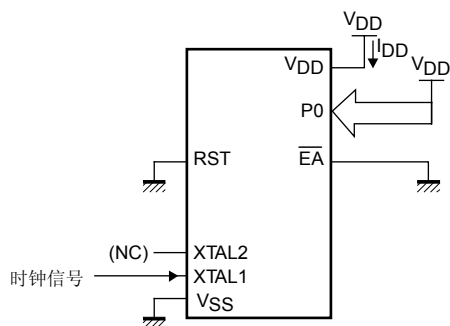


图 37 测试电阻举例



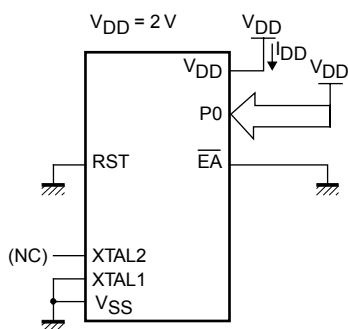
其它管脚都不连

图 38  $I_{DD}$  测试条件，激活模式



其它管脚都不连

图 39  $I_{DD}$  测试条件, 空闲模式

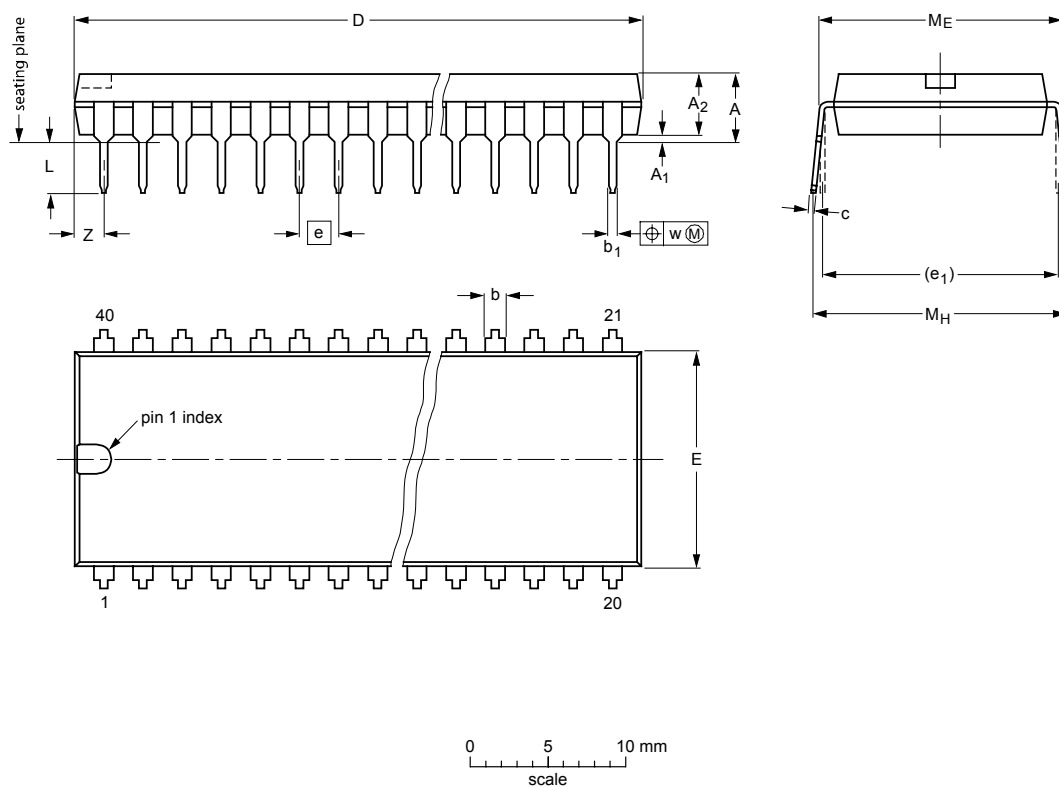


其它管脚都不连

图 40  $I_{DD}$  测试条件, 掉电模式

## 12. 封装

DIP40: 40 脚

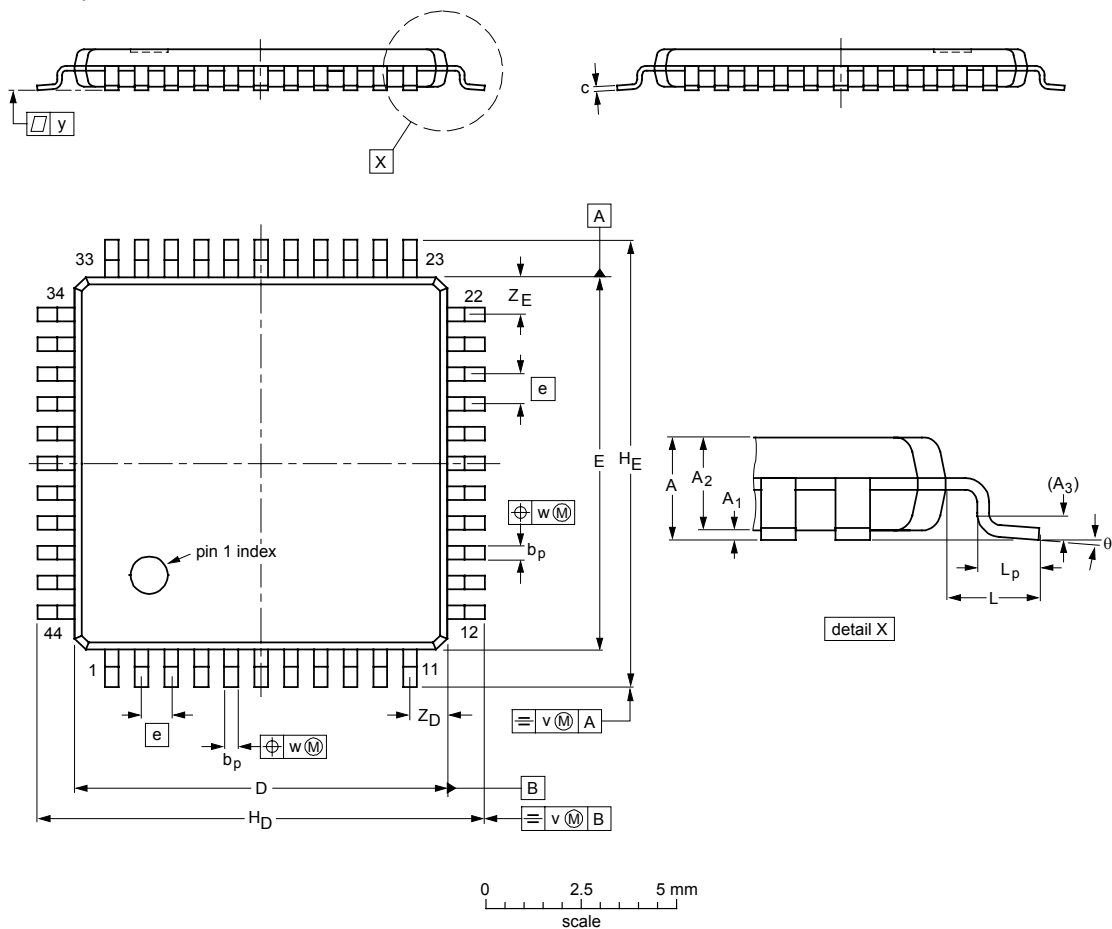


DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A <sub>1</sub> min.	A <sub>2</sub> max.	b	b <sub>1</sub>	c	D <sup>(1)</sup>	E <sup>(1)</sup>	e	e <sub>1</sub>	L	M <sub>E</sub>	M <sub>H</sub>	w	Z <sup>(1)</sup> max.
mm	4.7	0.51	4	1.70 1.14	0.53 0.38	0.36 0.23	52.5 51.5	14.1 13.7	2.54	15.24	3.60 3.05	15.80 15.24	17.42 15.90	0.254	2.25
inches	0.19	0.02	0.16	0.067 0.045	0.021 0.015	0.014 0.009	2.067 2.028	0.56 0.54	0.1	0.6	0.14 0.12	0.62 0.60	0.69 0.63	0.01	0.089

图 41 PDIP40 封装

TQFP44: 44 脚, 本体大小  $10 \times 1.0\text{mm}$

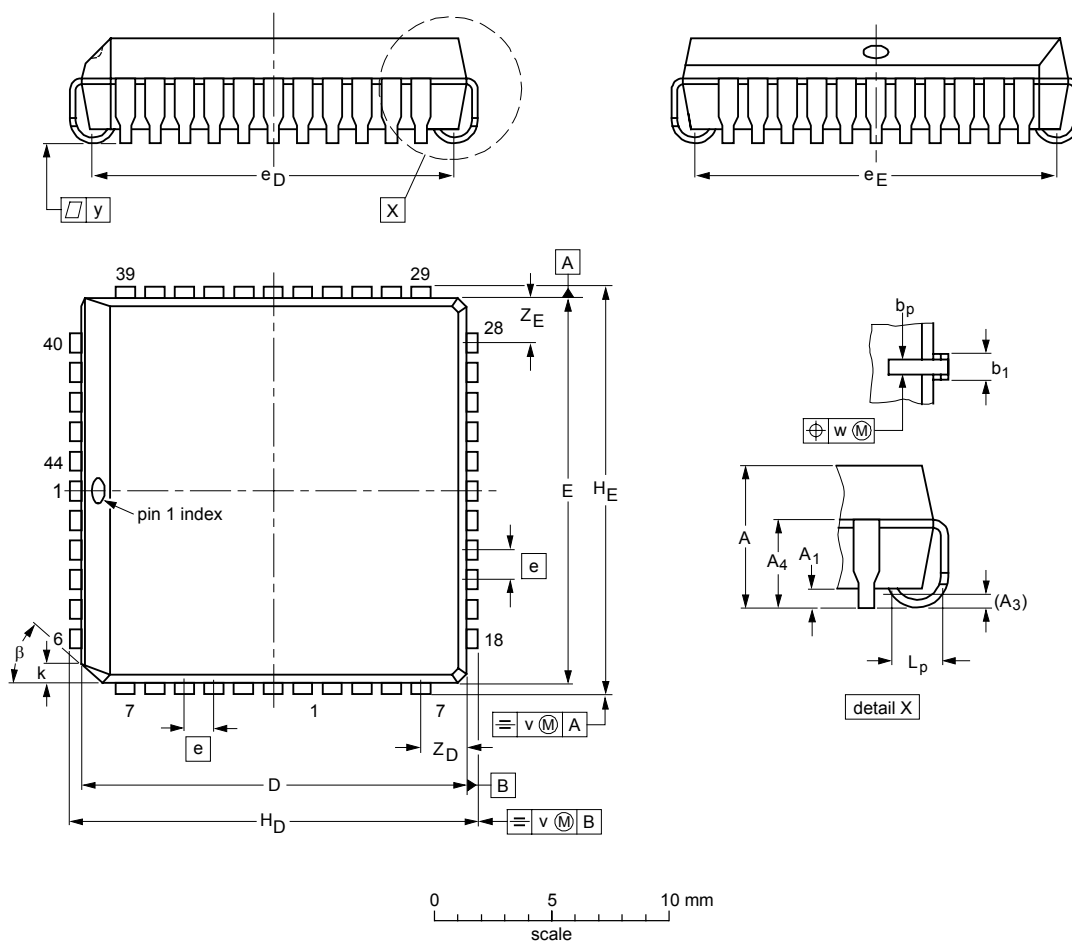


DIMENSIONS (mm are the original dimensions)

UNIT	A <sub>max.</sub>	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	b <sub>p</sub>	c	D <sup>(1)</sup>	E <sup>(1)</sup>	e	H <sub>D</sub>	H <sub>E</sub>	L	L <sub>p</sub>	v	w	y	Z <sub>D</sub> <sup>(1)</sup>	Z <sub>E</sub> <sup>(1)</sup>	θ
mm	1.2	0.15 0.05	1.05 0.95	0.25	0.45 0.30	0.18 0.12	10.1 9.9	10.1 9.9	0.8	12.15 11.85	12.15 11.85	1	0.75 0.45	0.2	0.2	0.1	1.2 0.8	1.2 0.8	7° 0°

图 42 TQFP44 封装

PLCC44: 44 端



DIMENSIONS (mm dimensions are derived from the original inch dimensions)

UNIT	A	A <sub>1</sub> min.	A <sub>3</sub>	A <sub>4</sub> max.	b <sub>p</sub>	b <sub>1</sub>	D <sup>(1)</sup>	E <sup>(1)</sup>	e	e <sub>D</sub>	e <sub>E</sub>	H <sub>D</sub>	yH <sub>E</sub>	k	L <sub>p</sub>	v	w		Z <sub>D</sub> <sup>(1)</sup> max.	Z <sub>E</sub> <sup>(1)</sup> max.	$\beta$
mm	4.57 4.19	0.51	0.25	3.05	0.53 0.33	0.81 0.66	16.66 16.51	16.66 16.51	1.27	16.00 14.99	16.00 14.99	17.65 17.40	17.65 17.40	1.22 1.07	1.44 1.02	0.18	0.18	0.1	2.16	2.16	45°
inches	0.180 0.165	0.02	0.01	0.12	0.021 0.013	0.032 0.026	0.656 0.650	0.656 0.650	0.05	0.63 0.59	0.63 0.59	0.695 0.685	0.695 0.685	0.048 0.042	0.057 0.040	0.007	0.007	0.004	0.085	0.085	

图 43 PLCC44 封装

13. 修订记录

2003 年 11 月 13 日

第二版