概述

P89C51RB2/RC2/RD2 具有 16K/32K/64K 并行可编程的非易失性 FLASH 程序存储器，并可实现对器件串行在系统编程（ISP）和在应用中编程(IAP)。在系统编程（ISP：In-System Programming）：当 MCU 安装在用户板上时，允许用户下载新的代码。在应用中编程（IAP：In-Application Programming）：MCU 可以在系统中获取新代码并对自己重新编程。这种方法允许通过调制解调器连接进行远程编程。片内 ROM 中固化的默认的加载程序（Boot Loader）允许 ISP 通过 UART 将程序代码装入 Flash 存储器，而 Flash 代码中则不需要加载程序。对于 IAP，用户程序擦除和重编程 Flash Memory 的操作是通过使用片内 ROM 中的标准程序。

该器件的 1 个机器周期由 6 个时钟周期组成，因此运行速度是传统 80C51 的 2 倍。一个 OTP 配置位可让用户选择传统的 12 时钟周期。该系列单片机是 80C51 微控制器的派生器件，是采用先进 CMOS 工艺制造的 8 位微控制器，指令系统与 80C51 完全相同。

有 4 组 8 位 I/O 口，3 个 16 位定时/计数器，多个中断源，4 个中断优先级、嵌套中断结构，1 个增强型 UART，片内振荡器及时序电路。

新增的特性使得 89C51RB2/RC2/RD2 成为功能更强大的微控制器，更好地支持应用于脉宽调制，高速 I/O，递增/递减计数能力（如电机控制）等场合。

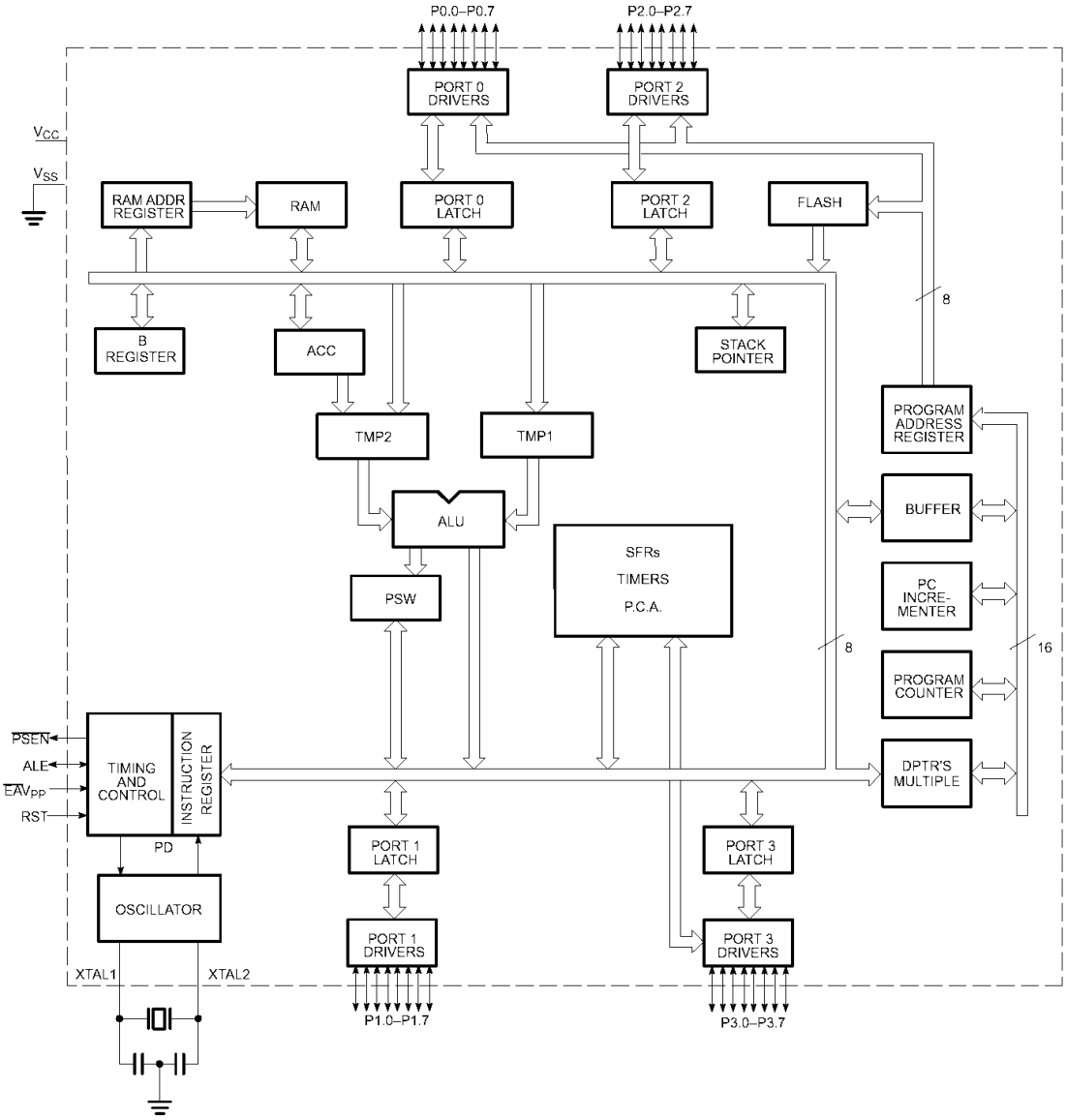
特性

- 80C51 核心处理单元
- 具有 ISP 和 IAP 功能的片内 FLASH 程序存储器
- 片内 Boot ROM 包含底层 FLASH 编程子程序，以实现通过 UART 下载程序
- 可实现最终用户应用的编程（IAP）
- 与 87C51 兼容的并行编程硬件接口
- 每个机器周期为 6 个时钟周期（标准）
- 可选择 12 个时钟周期的机器周期
- 采用 6 时钟周期时频率可高达 20MHz（相当于 40MHz），采用 12 时钟周期时频率可达 33MHz
- 全静态操作
- RAM 可扩展到 64K 字节
- 4 个中断优先级
- 7 个中断源
- 4 个 8 位 I/O 口
- 全双工增强型 UART
  - －帧错误检测
  - －自动地址识别
- 电源控制模式
  - －时钟可停止和恢复
  - －空闲模式
  - －掉电模式
- 可编程时钟输出
- 异步端口复位
- 双 DPTR 寄存器
- 低 EMI（禁止 ALE）
- 可编程计数器阵列（PCA）
  - －PWM
  - －捕获/比较

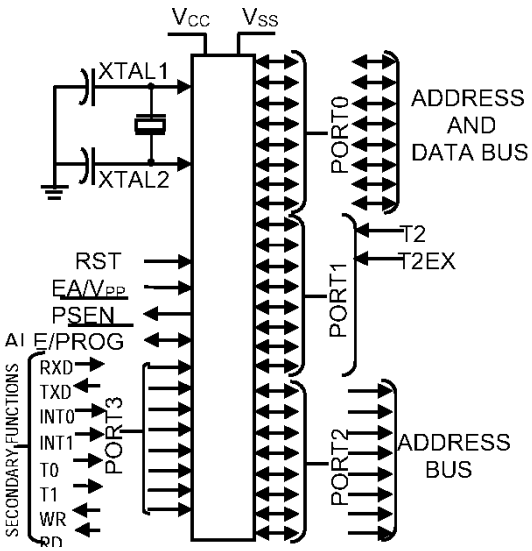
订购信息

	货品号	存储器		温度范围和封装	电压范围	频率（MHz）	
		Flash	RAM			6 时钟	12 时钟
1	P89C51RB2HBA	16KB	512B	0－70,PLCC	4.5-5.5V	0－20	0－33
2	P89C51RB2HBBD	16KB	512B	0－70,LQFP	4.5-5.5V	0－20	0－33
3	P89C51RC2HBP	32KB	512B	0－70,PDIP	4.5-5.5V	0－20	0－33
4	P89C51RC2HBA	32KB	512B	0－70,PLCC	4.5-5.5V	0－20	0－33
5	P89C51RC2HFA	32KB	512B	-40－85,PLCC	4.75-5.25V	0－20	0－33
6	P89C51RC2HBBD	32KB	512B	0－70,LQFP	4.5-5.5V	0－20	0－33
7	P89C51RC2HFBD	32KB	512B	-40－85,LQFP	4.75-5.25V	0－20	0－33
8	P89C51RD2HBP	64KB	1 KB	0－70,PDIP	4.5-5.5V	0－20	0－33
9	P89C51RD2HBA	64KB	1 KB	0－70,PLCC	4.5-5.5V	0－20	0－33
10	P89C51RD2HBBD	64KB	1 KB	0－70,LQFP	4.5-5.5V	0－20	0－33

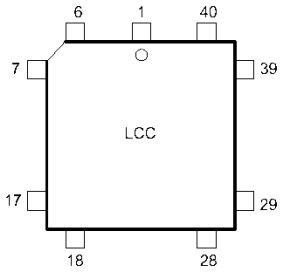
框图



逻辑符号



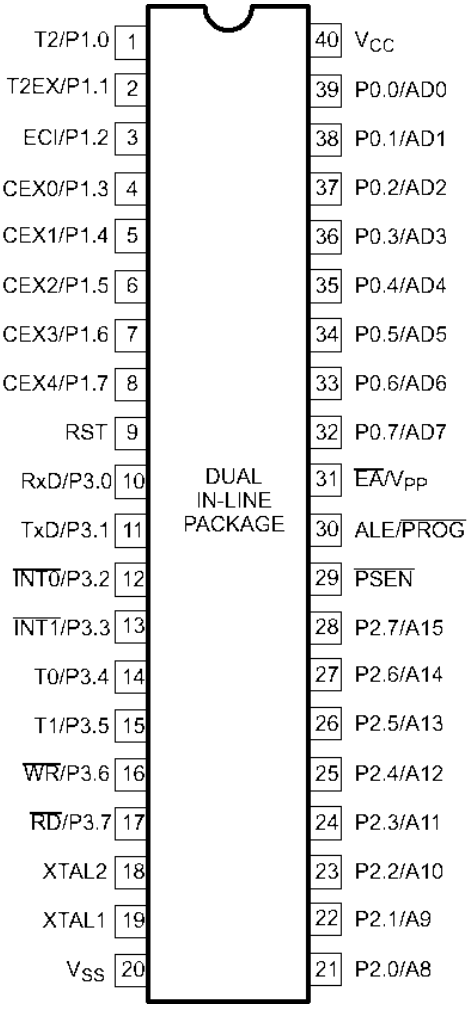
PLCC 和 CLCC 封装及管脚功能



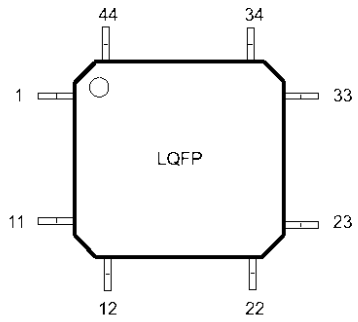
Pin	Function	Pin	Function	Pin	Function
1	NIC*	16	P3.4/T0	31	P2.7/A15
2	P1.0/T2	17	P3.5/T1	32	$\overline{PSEN}$
3	P1.1/T2EX	18	P3.6/$\overline{WR}$	33	ALE/$\overline{PROG}$
4	P1.2/ECI	19	P3.7/$\overline{RD}$	34	NIC*
5	P1.3/CEX0	20	XTAL2	35	$\overline{EA}$/V_{PP}
6	P1.4/CEX1	21	XTAL1	36	P0.7/AD7
7	P1.5/CEX2	22	V_{SS}	37	P0.6/AD6
8	P1.6/CEX3	23	NIC*	38	P0.5/AD5
9	P1.7/CEX4	24	P2.0/A8	39	P0.4/AD4
10	RST	25	P2.1/A9	40	P0.3/AD3
11	P3.0/RxD	26	P2.2/A10	41	P0.2/AD2
12	NIC*	27	P2.3/A11	42	P0.1/AD1
13	P3.1/TxD	28	P2.4/A12	43	P0.0/AD0
14	P3.2/$\overline{INT0}$	29	P2.5/A13	44	V_{CC}
15	P3.3/$\overline{INT1}$	30	P2.6/A14		

* NO INTERNAL CONNECTION

DIP 封装及管脚功能



LQFP 封装及管脚功能



Pin	Function	Pin	Function	Pin	Function
1	P1.5/CEX2	16	V_{SS}	31	P0.6/AD6
2	P1.6/CEX3	17	NIC*	32	P0.5/AD5
3	P1.7/CEX4	18	P2.0/A8	33	P0.4/AD4
4	RST	19	P2.1/A9	34	P0.3/AD3
5	P3.0/RxD	20	P2.2/A10	35	P0.2/AD2
6	NIC*	21	P2.3/A11	36	P0.1/AD1
7	P3.1/TxD	22	P2.4/A12	37	P0.0/AD0
8	P3.2/$\overline{INT0}$	23	P2.5/A13	38	V_{CC}
9	P3.3/$\overline{INT1}$	24	P2.6/A14	39	NIC*
10	P3.4/T0	25	P2.7/A15	40	P1.0/T2
11	P3.5/T1	26	$\overline{PSEN}$	41	P1.1/T2EX
12	P3.6/$\overline{WR}$	27	ALE/$\overline{PROG}$	42	P1.2/ECI
13	P3.7/$\overline{RD}$	28	NIC*	43	P1.3/CEX0
14	XTAL2	29	$\overline{EA}$/V_{PP}	44	P1.4/CEX1
15	XTAL1	30	P0.7/AD7		

* NO INTERNAL CONNECTION

管脚描述

名称	管脚号			类型	名称和功能
	DIP	LCC	QFP		
Vss	20	22	16	I	地
Vcc	40	44	38	I	电源：提供掉电、空闲、正常工作电压
P0.0-0.7	39-32	43-36	37-30	I/O	P0 口： P0 口是开漏双向口，可以写为 1 使其状态为悬浮，用作高阻输入。P0 也可以在访问外部程序存储器时作地址的低字节，在访问外部数据存储器时作数据总线，此时通过内部强上拉传送 1。
P1.0-1.7	1-8	2-9	40-44 1-3	I/O	P1 口： P1 口是带内部上拉的双向 I/O 口，向 P1 口写入 1 时，P1 口被内部上拉为高电平，可用作输入口。当作为输入脚时，被外部拉低的 P1 口会因为内部上拉而输出电流(见 DC 电气特性)。P1 口第 2 功能：
	1	2	40	I/O	T2(P1.0)： 定时/计数器 2 的外部计数输入/时钟输出
	2	3	41	I	T2EX(P1.1)：定时/计数器 2 重装载/捕捉/方向控制
	3	4	42	I	ECI(P1.2): PCA 的外部时钟输入
	4	5	43	I/O	CEX0(P1.3): PCA 模块0 捕获/比较模式的外部I/O脚
	5	6	44	I/O	CEX1(P1.4): PCA 模块1 捕获/比较模式的外部I/O脚
	6	7	1	I/O	CEX2(P1.5): PCA 模块2 捕获/比较模式的外部I/O脚
	7	8	2	I/O	CEX3(P1.6): PCA 模块3 捕获/比较模式的外部I/O脚
	8	9	3	I/O	CEX4(P1.7): PCA 模块 4 捕获/比较模式的外部 I/O 脚
P2.0-2.7	21-28	24-31	18-25	I/O	P2 口： P2 口是带内部上拉的双向 I/O 口，向 P2 口写入 1 时，P2 口被内部上拉为高电平，可用作输入口。当作为输入脚时，被外部拉低的 P2 口会因为内部上拉而输出电流(见 DC 电气特性)。在访问外部程序存储器和外部数据时分别作为地址高位字节和 16 位地址(MOVX @DPTR)，此时通过内部强上拉传送 1。当使用 8 位寻址方式(MOV @Ri)访问外部数据存储器时,P2 口发送 P2 特殊功能寄存器的内容。P2.7 在编程/擦除时必须为“1”
P3.0-3.7	10-17	11, 13-19	5, 7-13	I/O	P3 口：P3 口是带内部上拉的双向 I/O 口，向 P3 口写入 1 时，P3 口被内部上拉为高电平，可用作输入口。当作为输入脚时，被外部拉低的 P3 口会因为内部上拉而输出电流(见 DC 电气特性)。P89C51RX2 的 P3 口脚具有以下特殊功能：
	10	11	5	I	RxD(p3.0)： 串行输入口
	11	13	7	O	TxD(P3.1)： 串行输出口
	12	14	8	I	$\overline{\text{INT0}}$(P3.2)： 外部中断 0
	13	15	9	I	$\overline{\text{INT1}}$(P3.3)： 外部中断
	14	16	10	I	T0(P3.4)： 定时器 0 外部输入
	15	17	11	I	T1(P3.5)： 定时器 1 外部输入
	16	18	12	I	$\overline{\text{WR}}$(P3.6)： 外部数据存储器写信号
	17	19	13	I	$\overline{\text{RD}}$(P3.7)： 外部数据存储器读信号
RST	9	10	4	I	复位： 当晶振在运行中，只要复位管脚出现 2 个机器周期高电平即可复位，内部有扩散电阻连接到 Vss，仅需要外接一个电容到 Vcc 即可实现上电复位。

ALE	30	33	27	O	地址锁存使能： 在访问外部存储器时，输出脉冲锁存地址的低字节，在正常情况下，ALE 输出信号恒定为 1/6 振荡频率。并可用作外部时钟或定时，注意每次访问外部数据时一个 ALE 脉冲将被忽略。ALE 可以通过置位 SFR auxililary.0 禁止，置位后 ALE 只能在执行 MOVX 指令时被激活。
$\overline{\text{PSEN}}$	29	32	27	O	程序存储使能： 读外部程序存储。当从外部读取程序时，$\overline{\text{PSEN}}$ 每个机器周期被激活两次，在访问外部数据存储器 $\overline{\text{PSEN}}$ 无效，访问内部程序存储器时 $\overline{\text{PSEN}}$ 无效。
$\overline{\text{EA}}$/Vpp	31	35	29	I	外部寻址使能/编程电压：在访问整个外部程序存储器时，$\overline{\text{EA}}$ 必须外部置低。如果 $\overline{\text{EA}}$ 为高时，将执行内部程序。当 RST 释放后 EA 脚的值被锁存，任何时序的改变都将无效。该引脚在对 FLASH 编程时用于输入编程电压(Vpp)。
XTAL1	19	21	15	I	晶体 1：振荡反向放大器输入端和内部时钟发生电路输入端
XTAL2	18	20	14	O	晶体 2：振荡反向放大器输出端

注：为了避免"latch-up"对上电的影响,任意管脚（Vpp 除外）上的电压最大不能高于 Vcc+0.5,最低不能低于 Vss- 0.5。

表 1 P89C51RX2 特殊功能寄存器

名称	定义	地址	位功能和位地址								复位值
ACC*	累加器	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H
AUXR#	辅助功能寄存器	8EH	–	–	–	–	–	–	–	AO	xxxxxxx0B[1]
AUXR1#	辅助功能寄存器 1	A2H	–	–	–	–	GF2	0	–	DPS	02H[1]
B*	B 寄存器	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H
CCAP0H#	模块 0 捕获高字节	FAH									xxxxxxxxB
CCAP1H#	模块 1 捕获高字节	FBH									xxxxxxxxB
CCAP2H#	模块 2 捕获高字节	FCH									xxxxxxxxB
CCAP3H#	模块 3 捕获高字节	FDH									xxxxxxxxB
CCAP4H#	模块 4 捕获高字节	FEH									xxxxxxxxB
CCAP0L#	模块 0 捕获低字节	EAH									xxxxxxxxB
CCAP1L#	模块 1 捕获低字节	EBH									xxxxxxxxB
CCAP2L#	模块 2 捕获低字节	ECH									xxxxxxxxB
CCAP3L#	模块 3 捕获低字节	EDH									xxxxxxxxB
CCAP4L#	模块 4 捕获低字节	EEH									xxxxxxxxB
CCAPMO#	模块 0 模式	DAH	–	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	x0000000B
CCAPM1#	模块 1 模式	DBH	–	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	x0000000B
CCAPM2#	模块 2 模式	DCH	–	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	x0000000B
CCAPM3#	模块 3 模式	DDG	–	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	x0000000B
CCAPM4#	模块 4 模式	DEH	–	ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	x0000000B
			DF	DE	DD	DC	DB	DA	D9	D8	
CCON*#	PCA 计数器控制	D8H	CF	CR	–	CCF4	CCF3	CCF2	CCF1	CCF0	00x00000B
CH#	PCA 计数器高字节	F9H									00H
CL#	PCA 计数器低字节	E9H									00H

CMOD#	PCA 计数器模式	D9H	CIDL	WDTE	—	—	—	CPS1	CPS0	ECF	00xxx000B
DPTR:											
DPH	指针高字节	83H									00H
DPL	指针低字节	82H									00H
			AF	AE	AD	AC	AB	AA	A9	A8	
IE*	中断使能	A8H	EA	EC	ET2	ES	ET1	EX1	ET0	EX0	00H
			BF	BE	BD	BC	BBB	BA	B9	B8	
IP*	中断优先级	B8H	—	PPC	PT2H	PSH	PT1H	PX1H	PT0H	PX0H	x0000000B
			B7	B6	B5	B4	B3	B2	B1	B0	
IPH#	中断优先级高字节	B7H	—	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H	x0000000B
			87	86	85	84	83	82	81	80	
P0*	P0 口	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	FFH
			97	96	95	94	93	92	91	90	
P1*	P1 口	90H	CEX4	CEX3	CEX2	CEX1	CEX0	ECI	T2EX	T2	FFH
			A7	A6	A5	A4	A3	A2	A1	A0	
P2*	P2 口	A0H	AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8	FFH
			B7	B6	B5	B4	B3	B2	B1	B0	
P3*	P3 口	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{INT0}$	TxD	RxD	FFH
PCON#[1]	电源控制寄存器	87H	SMOD1	SMOD0	—	POF[2]	GF1	GF0	PD	IDL	00xxx000B
			D7	D6	D5	D4	D3	D2	D1	D0	
PSW*	程序状态字	D0H	CY	AC	F0	RS1	RS0	OV	F1	P	00000000B
RACAP2H#	定时器 2 捕获高字节	CBH									00H
RACAP2L#	定时器 2 捕获低字节	CAH									00H
SADDR#	从地址	A9H									00H
SADEN#	从地址屏蔽	B9H									00H
SBUF	串口数据缓冲区	99H									xxxxxxxxB
			9F	9E	9D	9C	9B	9A	99	98	
SCON*	串行口控制	98H	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SP	堆栈指针	81H									07H
			8F	8E	8D	8C	8B	8A	89	88	
TCON*	定时器控制	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
			8F	8E	8D	8C	8B	8A	89	88	
T2CON*	定时器 2 控制	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$	00H
T2MOD#	定时器 2 模式控制	C9H	—	—	—	—	—	—	T2OE	DCEN	xxxxxx00B
TH0	定时器 0 高字节	8CH									00H
TH1	定时器 1 高字节	8DH									00H
TH2#	定时器 2 高字节	CDH									00H
TL0	定时器 0 低字节	8AH									00H
TL1	定时器 1 低字节	8BH									00H
TL2#	定时器 2 低字节	CCH									00H
TMOD	定时器模式	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	00H

注： 带“*”号的 SFR 可位寻址。

带“#”号的 SFR 表示从 80C51 的 SFR 修改而来或新增加的。

“—”表示保留位

1．复位值由复位源确定。

2．此位不受复位影响。

FLASH EPROM 存储器

概述

P89C51/52/54/58 在 10000 次擦除和编程之后仍能可靠保存 FLASH 存储器的内容。存储单元的设计使擦除和编程结构最优化。此外，先进的沟道氧化工艺和低内部电场的结合使擦除和编程操作更加可靠。

特性

- FLASH EPROM 带片内擦除的内部程序存储器
- 内部程序存储器禁止时（$\overline{EA}$=0），外部程序存储器最多可达 64K
- 可编程加密位
- 每字节最少 10000 次擦除/编程周期
- 数据最少可保存 10 年
- 从一般供应商处可获得编程支持

振荡器特性

XTAL1 和 XTAL2 为输入和输出，可分别作为一个反向放大器的输入和输出。此管脚可配置为使用内部振荡器。要使用外部时钟源驱动器件时，XTAL2 可以不连接而由 XTAL1 驱动。外部时钟信号无占空比的要求，因为时钟通过触发器二分频输入到内部时钟电路。但高低电平的最长和最短时间必须符合手册的规定。

复位

在振荡器工作时，将 RST 脚保持至少两个机器周期高电平（24 个振荡器周期）可实现复位。为了保证上电复位的可靠，RST 必须保持足够长时间的高电平以使振荡器产生两个机器周期的脉冲（通常为几个微秒）。上电时 Vcc 和 RST 必须同时上升以实现正确的启动。当复位电压大于 V_{IH1}(min.)时，I/O 口 1、2 和 3 不同步复位。当 RST 撤除时，$\overline{EA}$ 的值被锁存。

低功耗模式

时钟停止模式

静态设计使时钟频率可以降至 0MHz(停止)。当振荡器停振时，RAM 和 SFR 的值保持不变。该模式允许逐步应用并可将时钟频率降至任意值以实现系统功耗的降低。如要实现最低功耗则建议使用掉电模式。

空闲模式

空闲模式（见表 2）中，CPU 进入睡眠状态，但片内的外围电路仍然保持工作状态。正常操作模式的最后一条指令执行进入空闲模式。空闲模式下，CPU 内容、片内 RAM 和所有 SFR 保持原来的值。任何被使能的中断（此时，程序从中断服务程序处恢复并继续执行）或硬件复位（与上电复位使用相同的方式启动处理器）均可终止空闲模式。

掉电模式

为了进一步降低功耗，通过软件可实现掉电模式(见表 2)。该模式中，振荡器停振并且在最后一条指令执行进入掉电模式。降到 2.0 伏时，片内 RAM 和 SFR 保持原值，在退出掉电模式之前 Vcc 必须升至规定的最低操作电压。

硬件复位或外部中断均可结束掉电模式。硬件复位使 SFR 值重新设置，但不改变片内 RAM 的值。外部中断允许 SFR 和片内 RAM 都保持原值。

要正确退出掉电模式，在 Vcc 恢复到正常操作电压范围之后，复位或外部中断开始执行并且要保持足够长的时间（通常小于 10ms）以使振荡器重新启动并稳定下来。

使用外部中断时，INT0 和 INT1 必须使能且配置为电平触发。将管脚电平拉低使振荡器重新启动，退出掉电模式后将管脚恢复为高电平。一旦中断被响应，RETI 之后所执行的是进入掉电模式指令的后一条指令。

表 2 空闲模式和掉电模式时外部管脚的状态

模式	程序存储器	ALE	PSEN	口 0	口 1	口 2	口 3
空闲	内部	1	1	数据	数据	数据	数据
空闲	外部	1	1	悬浮	数据	地址	数据
掉电	内部	0	0	数据	数据	数据	数据
掉电	外部	0	0	悬浮	数据	数据	数据

设计中的注意事项

当空闲模式被硬件复位所中止时，器件在内部复位之前从停止处恢复程序正常运行，时间为 2 个机器周期。这段时间内片内硬件禁止对内部 RAM 的访问，但对 I/O 口的访问未被禁止。当 Idle 模式被复位所中止时，为了消除可能产生的误写操作，应用 Idle 模式指令后的指令不应执行写 I/O 口或写外部存储器操作。

ONCE™模式

ONCE(在线仿真)模式实现了对系统的测试和调试而不需要将器件从电路中移去。进入 ONCE 模式的条件：

1. 当器件复位且 $\overline{\text{PSEN}}$ 为高电平时，将 ALE 置低电平；
2. 在 RST 撤除时，ALE 保持低电平。

当器件处于 ONCE 模式时，P0 口处于悬浮状态，其它 I/O 口、ALE 和 $\overline{\text{PSEN}}$ 为弱上拉。振荡电路保持工作状态，器件处于该模式时，可用仿真器或测试 CPU 驱动电路。执行正常复位时恢复正常操作。

可编程时钟输出

可从 P1.0 编程输出 50%占空比的时钟信号。P1.0 除了作为常规 I/O 口外，还有两个可选功能。它可编程为：

1. 用于定时/计数器 2 的外部时钟输入；
2. 在 16MHz 操作频率下输出频率从 122 Hz 到 8MHz 的 50%占空比的时钟信号（12 时钟模式时为 61Hz~4MHz）。

要将定时/计数器 2 配置为时钟发生器,C/$\overline{\text{T2}}$(T2CON.1)必须清零，T2MOD 中的 T20E 位必须置位。启动定时器 2 必须将 TR2(T2CON.2)置位。

时钟输出频率由振荡器频率和定时器 2 捕获寄存器的重新装入值确定，公式如下：

$$\frac{\text{振荡器频率}}{n\times(65536-\text{RCAP2H,RCAP2L})}$$

n＝2（6 时钟模式）
＝4（12 时钟模式）

此处（RCAP2H,RCAP2L）＝RCAP2H 和 RCAP2L 的内容作为一个 16 位无符号整数

在时钟输出模式中，定时器 2 的翻转将不会产生中断。这和它作为波特率发生器时相似。定时器 2 可同时作为波特率发生器和时钟发生器。但需要注意的是，波特率和时钟输出频率相同。

定时器 2 的操作

定时器 2

定时器 2 是一个 16 位定时/计数器。通过设置特殊功能寄存器 T2CON 中的 C/$\overline{\text{T2}}$ 位，可将其作为定时器或计数器。定时器 2 有三种操作模式：捕获、自动重新装载（递增或递减计数）和波特率发生器，这三

种模式由 T2CON 中的位进行选择（见表 3）。

捕获模式

在捕获模式中，通过 T2CON 中的 EXEN2 设置两个选项。如果 EXEN2＝0，定时器 2 作为一个 16 位定时器或计数器（由 T2CON 中 C/$\overline{T2}$ 位选择），溢出时置位 TF2（定时器 2 溢出标志位）。该位可用于产生中断（通过使能 IE 寄存器中的定时器 2 中断使能位）。如果 EXEN2＝1，与以上描述相同，但增加了一个特性，即外部输入 T2EX 由 1 变 0 时将定时器 2 中 TL2 和 TH2 的当前值各自捕获到 RCAP2L 和 RCAP2H。另外，T2EX 的负跳变使 T2CON 中的 EXF2 置位，EXF2 也象 TF2 一样能够产生中断（其向量与定时器 2 溢出中断地址相同，定时器 2 中断服务程序通过查询 TF2 和 EXF2 来确定引起中断的事件）。捕获模式如图 2 所示（在该模式中，TL2 和 TH2 无重新装载值。甚至当 T2EX 产生捕获事件时，计数器仍以 T2EX 的负跳变或振荡频率的 1/12 计数）。

自动重装模式（递增/递减计数器）

16 位自动重装模式中，定时器 2 可通过 C/$\overline{T2}$ 配置为定时器/计数器，编程控制递增/递减计数。计数的方向是由 DCEN（递减计数使能位）确定的，DCEN 位于 T2MOD 寄存器（见图 3）中。当 DCEN＝0 时，定时器 2 默认为向上计数；当 DCEN＝1 时，定时器 2 可通过 T2EX 确定递增或递减计数。

图 4 显示了当 DCEN＝0 时，定时器 2 自动递增计数。在该模式中通过设置 EXEN2 位进行选择。如果 EXEN2＝0，定时器 2 递增计数到 0FFFFH 并在溢出后将 TF2 置位，然后将 RCAP2L 和 RCAP2H 中的 16 位值作为重新装载值装入定时器 2。RCAP2L 和 RCAP2H 的值是通过软件预设的。

如果 EXEN2＝1，16 位重新装载可通过溢出或 T2EX 从 1→0 的负跳变实现。此负跳变同时将 EXF2 置位。如果定时器 2 中断被使能，则当 TF2 或 EXF2 置 1 时产生中断。

在图 5 中 DCEN＝1 时，定时器 2 可递增或递减计数。此模式允许 T2EX 控制计数的方向。当 T2EX 置 1 时，定时器 2 递增计数，计数到 0FFFFH 后溢出并置位 TF2。还将产生中断（如果中断被使能），定时器 2 的溢出将使 RCAP2L 和 RCAP2H 中的 16 位值作为重新装载值放入 TL2 和 TH2。

当 T2EX 置零时，将使定时器 2 递减计数。当 TL2 和 TH2 计数到等于 RCAP2L 和 RCAP2H 时，定时器产生溢出。定时器 2 溢出置位 TF2，并将 0FFFFH 重新装入 TL2 和 TH2。

当定时器 2 递增/递减产生溢出时，外部标志位 EXF2 翻转。如果需要，可将 EXF2 位作为第 17 位。在此模式中，EXF2 标志不会产生中断。

（最高位） （最低位）

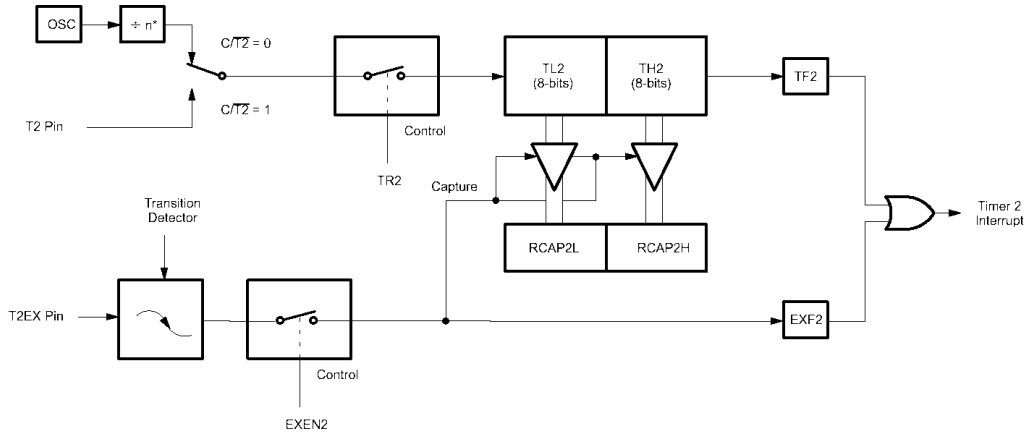
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

符号	位	名称和意义
TF2	T2CON.7	定时器 2 溢出标志。定时器 2 溢出时置位，必须由软件清除。当 RCLK 或 TCLK＝1 时，TF2 将不会置位。
EXF2	T2CON.6	定时器 2 外部标志。当 EXEN2＝1 且 T2EX 的负跳变产生捕获或重装时，EXF2 置位。定时器 2 中断使能时，EXF2＝1 将使 CPU 从中断向量处执行定时器 2 中断子程序。EXF2 位必须用软件清零。在递增/递减计数器模式（DCEN＝1）中，EXF2 不会引起中断。
RCLK	T2CON.5	接收时钟标志。RCLK 置位时，定时器 2 的溢出脉冲作为串行口模式 1 和模式 3 的接收时钟。RCLK＝0 时，将定时器 1 的溢出脉冲作为接收时钟。
TCLK	T2CON.4	发送时钟标志。TCLK 置位时，定时器 2 的溢出脉冲作为串行口模式 1 和模式 3 的发送时钟。TCLK＝0 时，将定时器 1 的溢出脉冲作为发送时钟。
EXEN2	T2CON.3	定时器 2 外部使能标志。当其置位且定时器 2 未作为串行口时钟时，允许 T2EX 的负跳变产生捕获或重装。EXEN2＝0 时，T2EX 的跳变对定时器 2 无效。
TR2	T2CON.2	定时器 2 启动/停止控制位。置 1 时启动定时器。
C/$\overline{T2}$	T2CON.1	定时器/计数器选择。（定时器 2） 0＝内部定时器（OSC/12） 1＝外部事件计数器（下降沿触发）
CP/$\overline{RL2}$	T2CON.0	捕获/重装标志。置位：EXEN2＝1 时 T2EX 的负跳变产生捕获。清零：EXEN2＝1 时定时器 2 溢出或 T2EX 的负跳变都可使定时器自动重装。当 RCLK＝1 或 TCLK＝1 时，该位无效且定时器强制为溢出时自动重装。

图 1. 定时器/计数器 2（T2CON）控制寄存器

表 3 定时器 2 工作方式

RCLK+TCLK	CP/$\overline{RL2}$	TR2	模式
0	0	1	16 位自动重装载
0	1	1	16 位捕获
1	X	1	波特率发生器
X	X	0	（关闭）



* n = 6 in 6 clock mode, or 12 in 12 clock mode.

图 2 定时器 2 捕获模式

T2MOD 地址=0C9H 复位值=XXXX XX00B

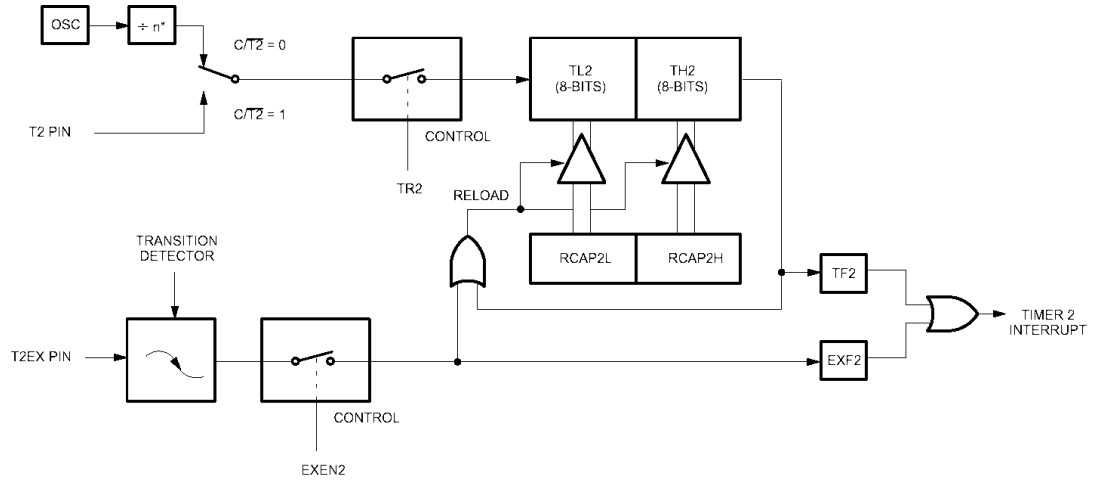
不可位寻址

—	—	—	—	—	—	T2OE	DCEN
位	7	6	5	4	3	2	1	0

符号	功能
—	不可用，保留将来之用。*
T2OE	定时器 2 输出使能位。
DCEN	向下计数使能位。定时器 2 可配置成向上/向下计数器。

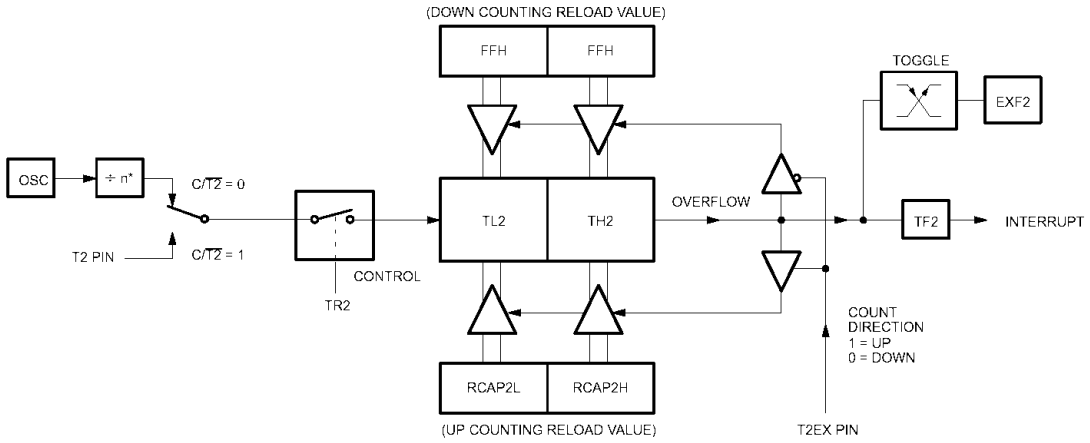
* 用户勿将其置 1。这些位在将来 8051 系列产品中用来实现新的特性，这种情况下，以后用到保留位，复位时或非有效状态时，它的值应为 0，而这些位为有效状态时，它的值为 1。从保留位读到的值是不确定的。

图 3 定时器 2 模式（T2MOD）控制寄存器



* n = 6 in 6 clock mode, or 12 in 12 clock mode.

图 4 定时器 2 自动装载模式（DCEN=0）



* n = 6 in 6 clock mode, or 12 in 12 clock mode.

图 5 定时器 2 自动装载模式（DCEN=1）

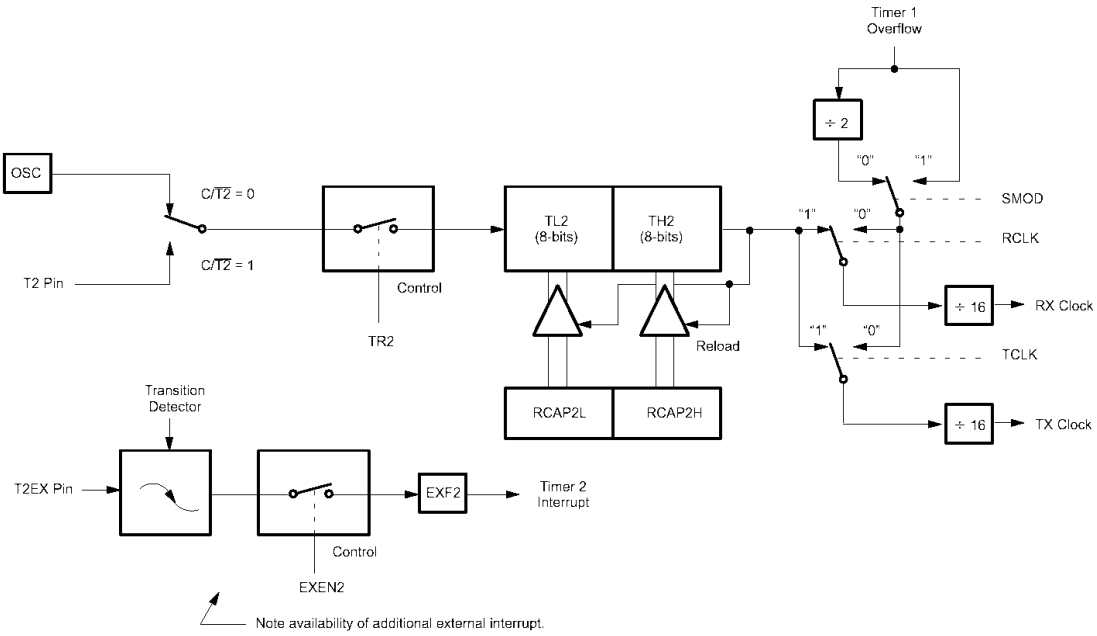


图 6　定时器 2 波特率发生器模式

表 4　由定时器 2 产生的常用波特率

波特率		振荡器频率	定时器 2	
			RCAP2H	RCAP2L
375k	750k	12MHz	FF	FF
9.6k	19.2k	12MHz	FF	D9
2.8k	5.6k	12MHz	FF	B2
2.4k	4.8k	12MHz	FF	64
1.2k	2.4k	12MHz	FE	C8
300	600	12MHz	FB	1E
110	220	12MHz	F2	AF
300	600	6MHz	FD	8F
110	220	6MHz	F9	57

波特率发生器模式

寄存器 T2CON（表 4）的位 TCLK 和（或）RCLK 允许从定时器 1 或定时器 2 中得到串行口发送和接收的波特率。当 TCLK=0 时 ，定时器 1 作为串行口发送波特率发生器；当 TCLK=1 时，定时器 2 作为串行口发送波特率发生器。RCLK 对串行口接收波特率有同样的效果。通过这两位，串行口能得到不同的接收和发送波特率——一个通过定时器 1 产生，另一个通过定时器 2 产生。

图 6 所示为定时器 2 工作在波特率发生器模式。与自动重装载模式相似，当 TH2 溢出时，波特率发生器模式使定时器 2 寄存器重新装载来自寄存器 RCAP2H 和 RCAP2L 的 16 位的值，寄存器 RCAP2H 和 RCAP2LR 的值由软件预置。

当工作于模式 1 和模式 3 时，波特率由下面给出的定时器 2 溢出率所决定：

$$\text{模式 1 和模式 3 的波特率} = \frac{\text{定时器 2 溢出率}}{16}$$

定时器可配置成“定时”或“计数”方式，在许多应用上，定时器被设置在“定时”方式（C/T2*=0）。当定时器 2 作为定时器时，它的操作不同于波特率发生器。

通常，定时器 2 作为定时器，每过一个机器周期计数加 1（例如 6 时钟模式时为 1/6 振荡器频率，12 时钟模式时为 1/12 振荡器频率）。当定时器 2 作为波特率发生器时，计数会以振荡器频率递增（12 时钟模式为 1/2 振荡频率）。这样，波特率公式如下：

$$\text{模式 1 和模式 3 的波特率}=\frac{\text{振荡器频率}}{[n\times[65536-(RCAP2H,RCAP2L)]]}$$

n＝16（6 时钟模式）

＝32（12 时钟模式）

此处：RCAP2H,RCAP2L)=RCAP2H 和 RCAP2L 的内容，为 16 位无符号整数。

如图 6 所示，定时器 2 作为波特率发生器，仅当寄存器 T2CON 中的 RCLK 和（或）TCLK=1 时，定时器 2 作为波特率发生器才有效。注意 TH2 溢出并不置位 TF2，也不产生中断。这样，当定时器 2 作为波特率发生器时，不必禁止定时器 2 中断。如果 EXEN2（T2 外部使能标志）被置位，在 T2EX（定时/计数器 2 触发输入）上由 1 到 0 的负跳变会置位 EXF2（T2 外部标志位），但不会导致（TH2，TL2）重新装入（RCAP2H，RCAP2L）。因此，当定时器 2 用作波特率发生器时，T2EX 在需要的时候可用作一个额外的外部中断。

当定时器工作在波特率发生器模式下时，不要对TH2和TL2进行读写。因为其脉冲源无论用的是外部的或者内部的，其计数频率与机器的读写钟频是不同步的，在此情况下对TH2和TH1进行读写是不准确的。可对RCAP2寄存器进行读，但不要进行写，否则将导致自动重装错误。当对定时器2或寄存器RCAP进行访问时，应关闭定时器（TR2清零）。

表 4 列出了常用的波特率和如何用定时器 2 得到这些波特率。

波特率公式汇总

定时器 2 工作在波特率发生器模式，外部时钟信号由 T2 脚进入，波特率为：

$$\text{波特率}=\frac{\text{定时器 2 溢出率}}{16}$$

如果定时器 2 采用内部时钟信号，则波特率为：

$$\text{波特率}=\frac{fosc}{[n\times[65536-(RCAP2H,RCAP2L)]]}$$

n＝16（6 时钟模式）

＝32（12 时钟模式）

fosc 为晶振频率

自动重装值可由下式得到：

RCAP2H,RCAP2L=65536-[fosc/(n*波特率)]

定时器/计数器 2 的设置

除了波特率发生器模式，T2CON 不包括 TR2 位的设置，TR2 位需单独设置来启动定时器。表 5，表 6 给出了 T2 作为定时器和计数器的设置。

表 5 T2 作为定时器

模式	T2CON	
	内部控制（注 1）	外部控制（注 2）
16 位重装	00H	08H
16 位捕获	01H	09H
波特率发生器接收和发送相同波特率	34H	36H
只接收	24H	26H
只发送	14H	16H

表 6 T2 作为计数器

模式	TMOD	
	内部控制（注 1）	外部控制（注 2）
16 位	02H	0AH
自动重装	03H	0BH

注： 1.定时器溢出时进行捕获和重装

2.当定时/计数器溢出并且 T2EX(P1.1)发生电平负跳变时产生捕获和重装（波特率方式除外）。

增强型 UART

UART 操作的一般模式，前部分已有描述。除了通过查询丢失的停止位进行帧错误检测和地址自动识别外，UART 和标准的 80C51 异步串行通信一样，支持多机通讯。

通过查询丢失的停止位对帧错误进行检测，丢失停止位时，将会置位 SCON 的 FE 位，FE 位与 SM0 共用 SCON.7 位地址，通过对 PCON.6（SMOD0）位进行设置来决定 SCON.7 位的功能。SMOD0 为 1 时，SCON.7 为 FE,SMOD0 为 0 时，SCON.7 为 SM0。SCON.7 位只能被软件清除。见图 8。

自动地址识别

自动地址识别是串行通讯一个特性，它使 UART 可以通过硬件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。通过置位 SCON 的 SM2 位来激活该功能。在 9 位 UART 模式中，模式 2 和模式 3，当接收完特定地址或广播地址后，接收中断标志位 RI 将自动置位。第 9 位为 1，说明所接收为地址，而非数据。自动地址识别见图 9。

模式 1 为 8 位 UART 模式，当 SM2 使能，所接收的 8 位数据后有有效的停止位 时，RI 将置位，所接收的 8 位数据为特定地址或广播地址。

模式 0 为移位寄存器模式，与 SM2 无关。

使用地址自动识别特性时，主机通过调用特定从机地址选择与一个（或多个）从机通信。使用广播地址时，所有从机都被联系。在此使用了两个特殊功能寄存器：SADDR 表示从机地址，SADEN 表示地址屏蔽。SADEN 用于定义 SADDR 内哪几位需使用而哪几位不予考虑。SADEN 可以与 SADDR 逻辑“与”得出给定的地址，用于对每一从机进行寻址。示例如下：

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

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

上例中 SADDR 相同，而 SADEN 不同以区分两个从机。从机 0 要求 0 位为 0 而忽略 1 位。从机 1 则要求 1 位为 0 而忽略 0 位。由于从机 1 的 1 位必须为 0，从机 0 只能取独有的地址 1100 0010 以区别。由于从机 0 的 0 位必须为 1，从机 1 只能取独有的地址 1100 0001 以区别。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：

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

从机 1 SADDR=1110 0000
SADEN=1111 1010

特定地址=1100 0X0X

从机 2 SADDR=1110 0000

SADEN=1111 1100

特定地址=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 可选通从机 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，无关位被认为是 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，也即“广播”地址。这样有效地禁止了自动寻址模式，并允许微处理器使用不带有上述特性的标准 UART 驱动器。

SCON Address=98H 复位值 00000000B

可位寻址

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

（SMOD0=0/1）*

符号 功能

FE 帧错误 位。当接收到无效停止位 时，该位置位。必须由软件清零。
如果要对 FE 进行访问，则应置位 SMOD。

SM0 串行口模式位 0（如果要对 SM0 进行访问，则应复位 SMOD）

SM1 串行口模式位 1

SM0	SM1	Mode	功能描述	波特率**
0	0	0	移位寄存器	fosc/12
0	1	1	8 位 UART	波特率可变
1	0	2	9 位 UART	fosc/64 或 fosc/32
1	1	3	9 位 UART	波特率可变

SM2 在模式 2 和 3 方式下使能自动地址识别功能。当 SM2=1 时，只有当传送的数据第 9 位（RB8）为 1，说明传送的是地址时，RI 才会置位。在模式 1 方式下，如果 SM2=1,只有当接收到有效的停止位 时，RI 才会置位，并且接收的位为特定或广播地址。模式 0 时，SM2 应该为 0。

REN 使能串行接收，软件置位允许串行接收，软件复位禁止串行接收。

TB8 在模式 2 和 3 方式下所发送的第 9 位数据，软件置位或清零。

RB8 在模式 2 和 3 方式下所接收的第 9 位数据，在模式 1 方式下，为所接收的停止位。模式 0 方式下，该位无效。

TI 发送中断标志位，模式 0 方式下发送完第 8 位后，由硬件置位。在其它模式下，在发送停止位之前由硬件置位。该位由软件复位。

RI 接收中断标志位，模式 0 方式下接收完第 8 位后，由硬件置位。在其它模式下，在接收停止位当中由硬件置位。该位由软件复位。

注： *SMOD0 在 PCON.6 中

**fosc=晶振频率

图 7 SCON 串行口控制寄存器

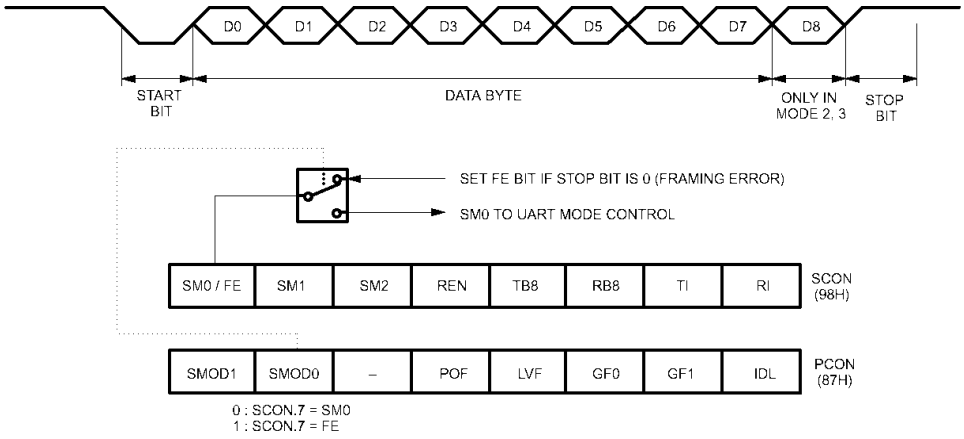


图 8 UART 帧错误检测

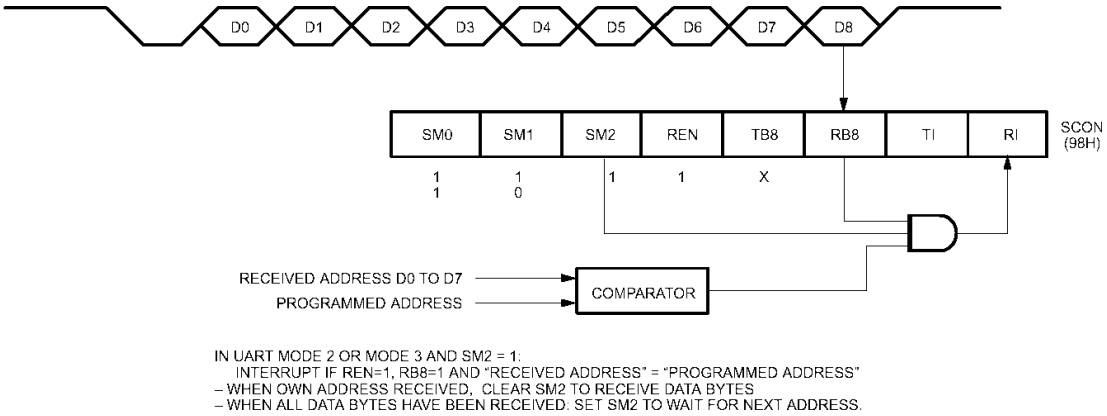


图 9 UART 多机通讯和地址的自动识别

中断优先级

P89C51RB2/RC2/RD2 有 7 个中断源，4 个中断优先级。

4 个中断优先级和 3 个特殊功能寄存器相关，他们分别是 IE，IP 和 IPH(见图 10、11、12)。

寄存器 IPH（中断优先级高）组成 4 级中断结构，IPH 的地址位于 SFR 中的 B7H。IPH 寄存器及其位的描述见图 12。

IPH 寄存器的功能很简单，IPH 和 IP 组合使用决定每一个中断的优先级。如下表所示：

优先级位		中断优先级
IPH. x	IP. x	
0	0	0 级(最低级)
0	1	1 级
1	0	2 级
1	1	3 级(最高级)

表 7 中断表

中断源	优先顺序	请求位	硬件清除	入口地址
X0	1	IE0	N(L)[1] Y(T)[2]	03H
T0	2	TP0	Y	0BH
X1	3	IE1	N(L) Y(T)	13H
T1	4	TF1	Y	1BH
PCA	5	CF,CCFn(n=0-4)	N	33H
SP	6	R1,T1	N	23H
T2	7	TF2,EXF2	N	2BH

注：

1. L=中断级激活
2. T= 转换激活

IE(0A8H)

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

使能位=1，使能中断

使能位=0，禁止中断

位	标号	功能
IE.7	EA	全局禁止位。如果 EA=0，禁止所有中断，如果 EA=1，通过置位或清除使能位，对应的每个中断被使能或禁止。
IE.6	EC	PCA 中断使能位
IE.5	ET2	定时器 2 中断使能位
IE.4	ES	串行口中断使能位
IE.3	ET1	定时器 1 中断使能位
IE.2	EX1	外部中断 1 使能位
IE.1	ET0	定时器 0 中断使能位
IE.0	EX0	外部中断 0 使能位

图 10 IE 特殊功能寄存器

IP(0B8H)

7	6	5	4	3	2	1	0
—	—	PT2	PS	PT1	PX1	PT0	PX0

中断优先级控制位=1，定义为高优先级中断

中断优先级控制位=0，定义为低优先级中断

位	标号	功能
IP.7	—	—
IP.6	PPC	PCA 中断优先级控制位
IP.5	PT2	定时器 2 中断优先级控制位
IP.4	PS	串行口中断优先级控制位
IP.3	PT1	定时器 1 中断优先级控制位
IP.2	PX1	外部中断 1 中断优先级控制位
IP.1	PT0	定时器 0 中断优先级控制位
IP.0	PX0	外部中断 0 中断优先级控制位

图 11 IP 寄存器

IPH(B7H)	7	6	5	4	3	2	1	0
	—	—	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

中断优先级控制位=1，定义为高优先级中断

中断优先级控制位=0，定义为低优先级中断

位	标号	功能
IPH.7	—	—
IPH.6	PPCH	PCA 中断优先级控制位高
IPH.5	PT2H	定时器 2 中断优先级控制位高
IPH.4	PSH	串行口中断优先级控制位高
IPH.3	PT1H	定时器 1 中断优先级控制位高
IPH.2	PX1H	外部中断 1 中断优先级控制位高
IPH.1	PT0H	定时器 0 中断优先级控制位高
IPH.0	PX0H	外部中断 0 中断优先级控制位高

图 12 IPH 寄存器

4 个中断级比 80C51 多 2 个。在没有产生同级的中断和更高级的中断情况下，中断将被执行，如果同级的中断或更高级的中断正在执行，新的中断只有等到正在执行的中断结束才能被执行。在更低级的中断正在执行情况下产生新的中断时，低级的中断停止，转而执行新的中断，直到新中断完成才可以执行被停止的中断。

可编程计数器阵列(PCA)

P89C51RB2/RC2/RD2的可编程计数器阵列是由五个16位捕获/比较模块组成的特殊定时器。每个模块都可以经编程实现下列多种模式： 捕捉模式，软件定时器模式，高速输出模式和PWM 脉宽调制模式。每个模块都有一个P1口的管脚与之对应。例如，模块0连接到P1.3(CEX0)，模块1连接到P1.4(CEX1)。PCA的基本配置见图14。

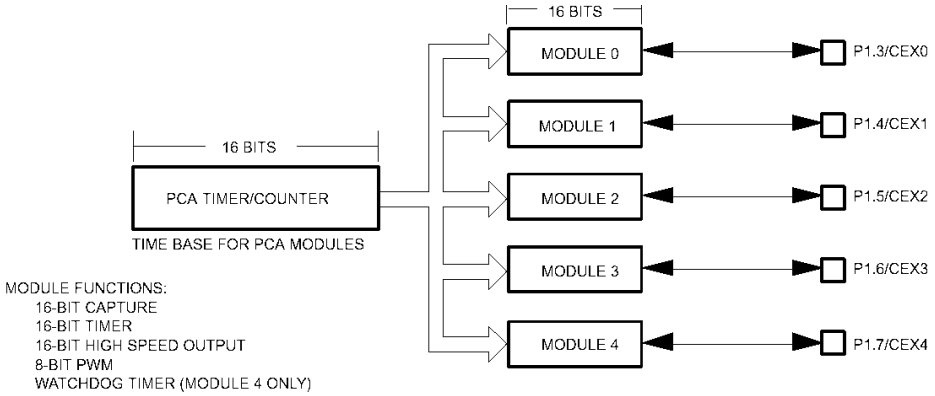


图14 可编程计数器阵列（PCA）

PCA定时器所有5个模块有一个公共的时基，并可编程为：1/6振荡器频率，1/2振荡器频率，定时器0溢出或者ECI脚（P1.2）的输入。定时器的计数源由CMOD中的CPS1和CPS0决定（见图17）。

CPS1	CPS0	PCA定时器计数源
0	0	1/6振荡器频率（6时钟模式）
		1/12振荡器频率（12时钟模式）
0	1	1/2振荡器频率（6时钟模式）
		1/4振荡器频率（12时钟模式）
1	0	定时器0溢出
1	1	ECI脚的外部输入

在寄存器CMOD中有3个附加的位与PCA相关。其中CIDL位允许PCA在空闲模式时停止，WDTE使能/禁止模块4的看门狗功能，ECF可以控制产生中断，而PCA溢出标志CF（位于CCON中）在PCA定时器溢出时置位。这些功能如图15所示。看门狗定时器功能由模块4实现（见图24）。

寄存器CCON包含了PCA的运行控制位CR，溢出标志CF以及每个模块的中断标志（如图18所示）。要运行PCA，CR位必须由软件置位，清零该位则关闭PCA。CF位(CCON.7)在PCA计数器溢出时置位，此时若ECF位（位于CMOD中）为1则产生中断。CF位只能由软件清零。CCON寄存器中的位0～4为各个模块的中断标志位。当发生捕获/匹配时由硬件置位，只能由软件清零。PCA中断系统如图16所示。PCA的每个模块都有一个SFR与之对应，例如，CCAPM0用于模块0，CCAPM1用于模块1（见图19）。这些寄存器包含了控制各模块工作方式的位。当相关模块发生匹配或者比较时，ECCF位（CCAPMn.0，n=0,1,2,3或4）使能CCON中的CCF标志以产生中断。PWM位（CCAPMn.1）使能脉宽调制模式。TOG位（CCAPMn.2）置位且PCA计数器与模块的捕获/比较寄存器匹配时，CEX输出发生翻转。MAT位（CCAPMn.3）置位且PCA计数器与模块的捕获/比较寄存器匹配时，CCON中的CCFn位置位。

CAPN(CCAPMn.4)和CAPP(CCAPMn.5)决定捕获输入被激活的边沿。CAPN使能下降沿，CAPP使能上升沿。如果两个位都置位则在上升和下降沿均可使能产生捕获。ECOM(CCAPMn.6)置位可使能比较器功能。如图20所示为CCAPMn设置实现不同的PCA功能。

有两个额外的寄存器与每个PCA模块相关。分别为CCAPnH和CCAPnL，它们保存用于产生捕获或者比较的16位计数值。当模块用于PWM模式时，这两个寄存器用于控制输出信号的占空比。

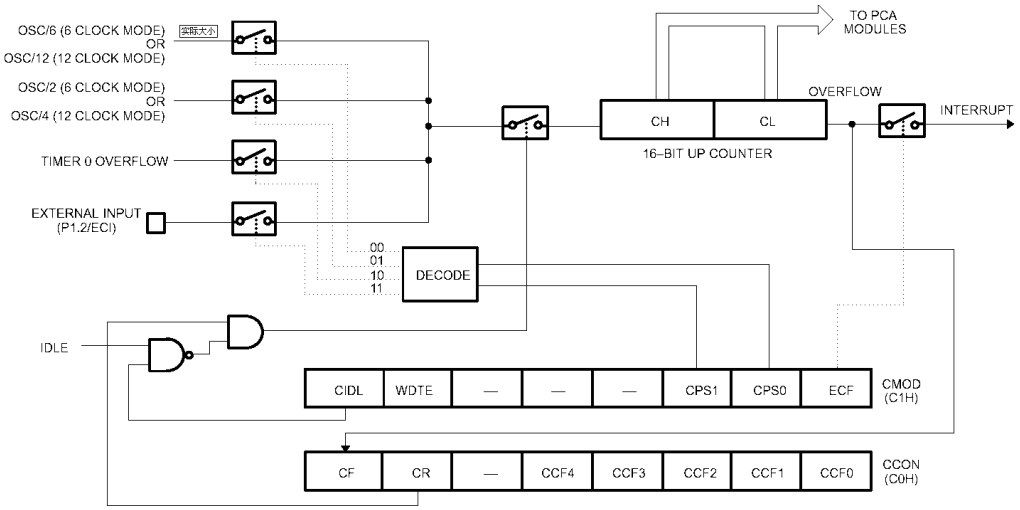


图15 PCA定时/计数器

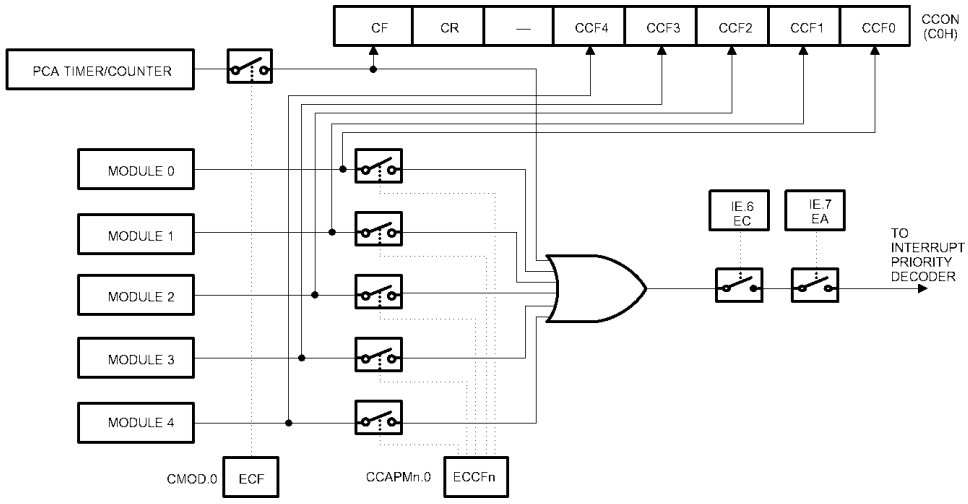


图16 PCA中断系统

CMOD 地址=D9H 复位值=00XX X000B

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

符号	功能
CIDL	计数器空闲控制:CIDL=0 时，PCA 在空闲模式下继续工作。CIDL=1 时，PCA 在空闲模式下关闭。
WDTE	看门狗定时器使能：WDTE=0 时禁止 PCA 模块 4 的看门狗功能。WDTE=1 时使能。
—	保留位，编程时勿将其置位。
CPS1	PCA 计数脉宽选择位 1
CPS0	PCA 计数脉宽选择位 0

CPS1	CPS0	选择 PCA 输入*	
0	0	0	内部时钟，6 时钟模式下为 fosc/6(12 时钟模式下为 fosc/12)
0	1	1	内部时钟，6 时钟模式下为 fosc/2(12 时钟模式下为 fosc/4)
1	0	2	定时器 0 溢出
1	1	3	ECI/P1.2 脚输入的外部时钟(6 时钟模式最大为 fosc/4,12 时钟模式下为 fosc/8)

ECF	PCA 使能计数器溢出中断：ECF=1 使能 CCON 中的 CF 位产生中断；ECF=0 时禁止 CF 的功能。

注：* fosc=振荡器频率

图17 CMOD：PCA计数器模式寄存器

CCON 地址=D8H 复位值=00X0 0000B

可位寻址

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

符号	功能
CF	PCA 计数器溢出标志。计数器溢出时由硬件置位。CMOD 中 ECF 置位时，CR 为中断标志位。CF 可由硬件或软件置位，但只能由软件清零。
CR	PCA 计数器运行控制位。由软件置位将 PCA 计数器打开；要关闭 PCA 计数器必须由软件清零。
—	保留位，编程时勿将其置位。
CCF4	PCA 模块 4 中断标志位。当产生匹配或捕获时由硬件置位。必须由软件清零。
CCF3	PCA 模块 3 中断标志位。当产生匹配或捕获时由硬件置位。必须由软件清零。
CCF2	PCA 模块 2 中断标志位。当产生匹配或捕获时由硬件置位。必须由软件清零。
CCF1	PCA 模块 1 中断标志位。当产生匹配或捕获时由硬件置位。必须由软件清零。
CCF0	PCA 模块 0 中断标志位。当产生匹配或捕获时由硬件置位。必须由软件清零。

图18 CCON:PCA计数器控制寄存器

CCAPMn 地址 CCAPM0 0DAH 复位值=X000 0000B

CCAPM1 0DBH

CCAPM2 0DCH

CCAPM3 0DDH

不可位寻址 CCAMP4 0DEH

—	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn
7	6	5	4	3	2	1	0

符号	功能
—	保留位，编程时勿将其置位。
ECOMn	比较器使能位。ECOMn=1 使能比较器功能。
CAPPn	上升沿捕获，CAPPn=1 使能上升沿捕获。
CAPNn	下降沿捕获，CAPNn=1 使能下降沿捕获。
MATn	匹配。MATn=1 时，PCA 计数器值与该模块比较/捕获寄存器值的匹配使 CCON 中的 CCFn 位置位，标志中断。
TOGn	触发。TOGn=1 时，PCA 计数器值与该模块比较/捕获寄存器值的匹配使 CEXn 脚的电平发生翻转。
PWMn	脉宽调制模式。PWMn=1 使能 CEXn 脚作为脉宽调制信号的输出。
ECCFn	使能 CCF 中断。使能 CCON 中的比较/捕获标志 CCFn 以产生中断。

图19 PCA模块比较/捕获寄存器

-	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCn	模块功能
X	0	0	0	0	0	0	0	无
X	X	1	0	0	0	0	X	16位捕获(CEXn脚上升沿触发)
X	X	0	1	0	0	0	X	16位捕获(CEXn脚下降沿触发)
X	X	1	1	0	0	0	X	16位捕获(CEXn脚双边沿触发)
X	1	0	0	1	0	0	X	16位软件定时器
X	1	0	0	1	1	0	X	16位高速输出
X	1	0	0	0	0	1	0	8位PWM
X	1	0	0	1	X	0	X	看门狗定时器

图20 PCA模块模式（CCAPMn寄存器）

PCA捕获模式

要使用PCA模块中的捕获模式，必须将CCAPMn中的位CAPN和/或CAPP置位。CEXn输入的信号出现上升或下降沿时，PCA硬件将PCA计数器寄存器（CH和CL）的值装入模块捕获寄存器（CCAPnL和CCAPnH）。与此同时置位CCFn 标志。如果CCAPMn中的ECCFn位已经置位，则会产生中断。如图21所示。

16位软件定时器

要使用PCA模块中的软件定时器模式，必须将CCAPMn中的位ECOM和MAT置位。PCA定时器和模块的捕获寄存器进行比较。当两者的值相同且CCAPMn中的ECCFn位已经置位时，将会产生中断。如图22所示。

高速输出模式

在该模式下，每次当PCA定时器和模块的捕获寄存器的值相匹配时，模块对应的输出脚的电平将会发生翻转。要使能该模式，必须将CCAPMn中的TOG、MAT和ECOM都置位。见图23。

脉宽调制模式

所有的PCA模块都可用作PWM输出。其功能如图24所示。输出的频率由PCA定时器决定。由于所有的模

块共用PCA定时器，因此它们具有相同的输出频率。每个模块的占空比可通过各自的捕获寄存器CCAPLn单独设置。当PCA CL寄存器的值小于CCAPLn的值，输出为低；当CL的值等于或大于CCAPLn的值时，输出为高。当CL从FF溢出到00时，CCAPLn将CCAPHn的值重新装入。这就使PWM不会出现误操作。要使能PWM模式，必须将CCAPMn中的位PWM和ECOM都置位。

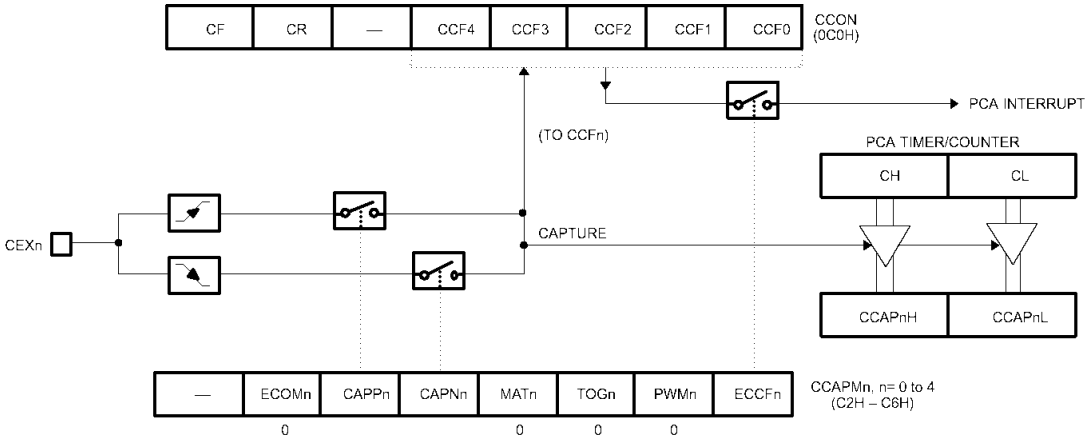


图21 PCA捕获模式

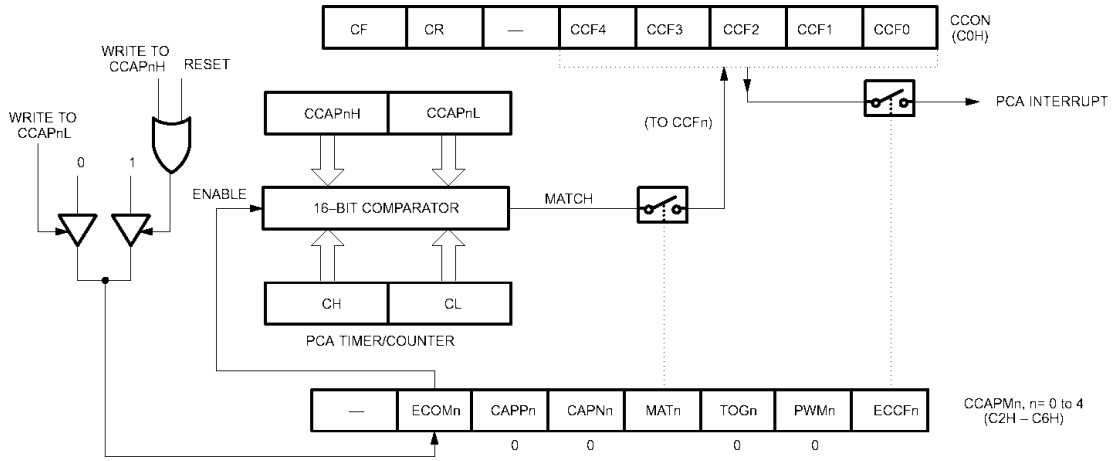


图22 PCA比较模式

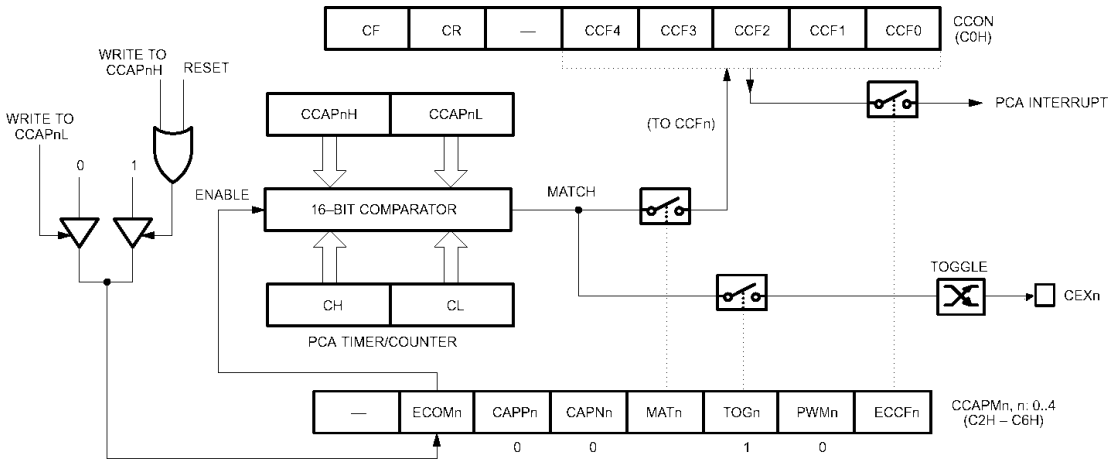


图23 PCA高速输出模式

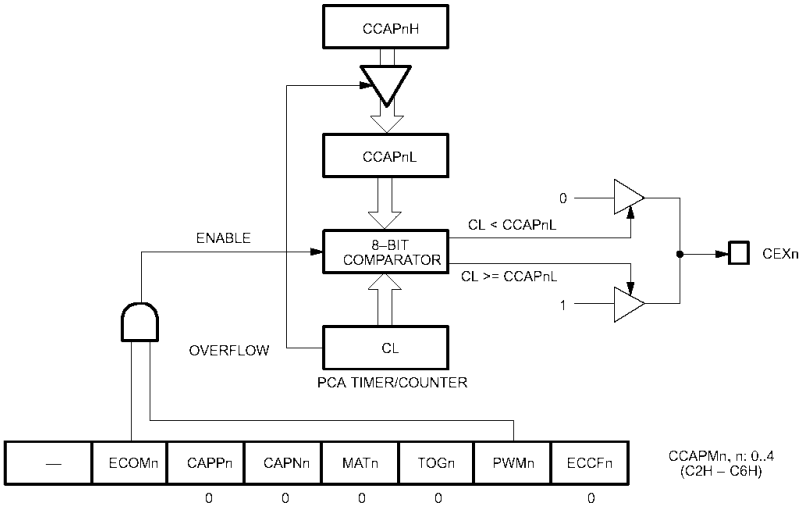


图24 PCA PWM模式

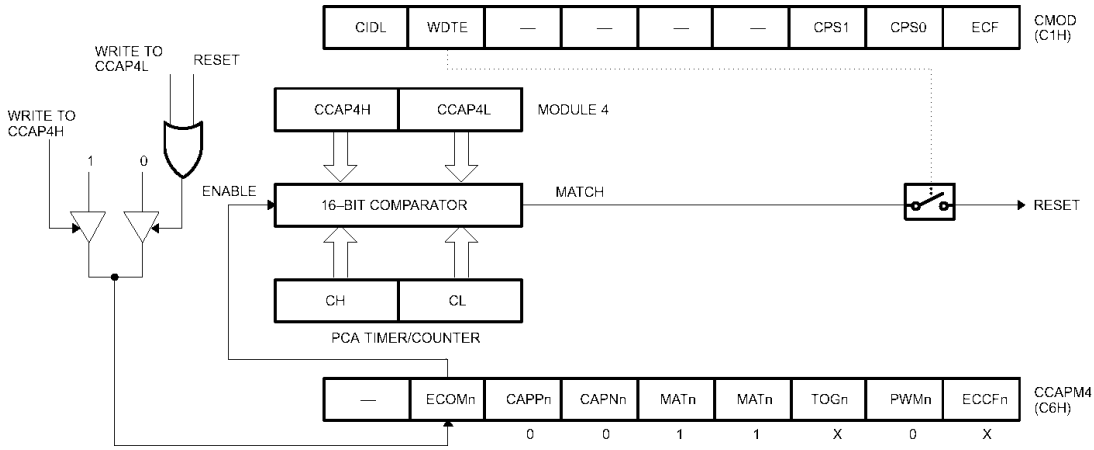


图25 PCA看门狗定时器(模块4)

PCA看门狗定时器

由PCA实现的片内看门狗定时器可改善系统的稳定性而不需要增加芯片的成本。看门狗定时器对于那些易受噪声、电源杂波或者静电放电影响的系统很有用处。PCA模块中只有模块4可编程为看门狗。如果不需要看门狗，该模块还可以用作其它模式。如图25所示为看门狗工作的示意图。用户将一个16位数值预先装入比较寄存器。就像其它比较模式一样，该16位值与PCA定时器值相比较。如果相同则产生一个内部复位，但不会将复位管脚RST拉高。

为了防止发生复位，用户有下列3个选择：

1． 周期性改变比较的数值，使之不会与PCA定时器的值相同。

2． 周期性改变PCA定时器的值，使之不会与比较的数值相同。

3． 在发生匹配之前将WDTE位清零，然后再重新使能它。

前两个选项更为可行，因为看门狗定时器从来不像第3项那样被禁止。如果程序指针跑飞，最终会发生匹配并产生内部复位信号。如果其它模块在使用时，最好不用第2个选项，因为PCA定时器是所有模块的时基，改变了它对其它模块都有影响。因此，在大多数应用中，第1个选项是最好的。

图26所示为看门狗定时器的初始化代码。模块4配置为比较模式，同时CMOD中的WDTE位必须置位。用户软件必须周期性改变（CCAP4H,CCAP4L）防止发生复位。该程序不能作为一个中断服务程序，因为如果程序指针跑飞并进入一个死循环，仍然执行中断将看门狗复位，这样将起不到使用看门狗的目的。应当在主程序小于PCA定时器2^{16}个计数的间隔内调用该程序。

```
INIT_WATCHDOG:
    MOV  CCAPM4,#4CH        ;模块4为比较模式
    MOV  CCAP4L,#0FFH       ;先写低字节
    MOV  CCAP4H,#0FFH       ;在PCA定时器计数到FFFF之前必须改变该比较值
    ORL  CMOD,#40H          ;置位WDTE位以使能看门狗定时器,不改变CMOD的其它位
;-------------------------------------------------
;主程序,周期性调用子程序WATCHDOG
;-------------------------------------------------
WATCHDOG:
    CLR  EA                 ;关中断
    MOV  CCAP4L,#00         ;下一个比较值与当前PCA定时器值相差不超过255个计数
    MOV  CCAP4H,CH
    SETB EA
    RET
```

图26 PCA看门狗定时器初始化代码

扩展数据RAM寻址

P89C51RB2/RC2/RD2内部数据存储器分为四部分:RAM低128字节、高128字节、128字节特殊功能寄存器（SFR）和256字节扩展RAM(ERAM)（RD2为768字节）。

1. RAM低128字节（地址00H到7FH）可直接或间接寻址
2. RAM高128字节（地址80H到FFH）只能间接寻址
3. 特殊功能寄存器（地址80H到FFH）只能直接寻址
4. 256/768字节扩展RAM（地址00H到1FFH/2FFH）通过将EXTRAM位清零实现MOVX间接寻址，见图27。

AUXR 地址＝ 0DAH　　复位值=xxxx xx00B

不可位寻址

—	—	—	—	—	—	EXTRAM	AO
7	6	5	4	3	2	1	0

符号	功能
AO	禁止/使能 ALE
	AO 操作模式
	0 以恒定的 1/3 振荡器频率发出 ALE 信号（12 时钟模式下为 1/6 fosc）
	1 ALE 仅当执行 MOVX 或 MOVC 指令时才有效
EXTRAM	使用 MOVX @Ri/@DPTR 访问内部/外部 RAM
	EXTRAM 操作模式
	0 使用 MOVX @Ri/@DPTR 访问内部 RAM
	1 访问外部数据存储器
—	保留位，编程时勿将其置位。

图27 AUXR：辅助寄存器

RAM低128字节可直接或间接寻址。高128字节只能间接寻址，高128字节与SFR占用相同的地址空间。不过它们在物理上是分开的。当指令访问高于7FH的内部地址时，CPU通过使用的寻址方式确定是对高128字节RAM还是对SFR进行访问。指令使用直接寻址方式访问SFR，例如：

```
MOV  0A0H,#data
```

使用间接寻址方式访问高128字节RAM，例如：

```
MOV  @R0,#data
```

ERAM可通过将EXTRAM位清零用MOVX实现间接寻址。这部分ERAM在物理上位于片内，在逻辑上占用了外部数据存储器开始的7936字节。EXTRAM=0时，ERAM通过MOVX和R0、R1的组合使用实现间接寻址。对ERAM的访问不会影响P0口、P3.6(WR#)和P3.7(RD#)。在对外部寻址时，P2 SFR为输出。例如，EXTRAM=0时

```
MOVX @R0,#data
```

此处R0=0A0H，是对地址为0A0H的ERAM寻址而不是对外部存储器寻址。对高于ERAM地址的外部数据地址进行访问的方式与标准80C51相同。P0和P2作为数据/地址总线，P3.6和P3.7作为读/写时序信号。见图28。

EXTRAM=1时，MOVX @Ri和MOVX @DPTR指令与标准80C51相似。MOVX @Ri提供一个8位地址与数据复用P0口，任意的输出口都可用作输出高位地址。这提供了外部翻页功能。MOVX @DPTR产生一个16位地址，P2口输出高8位地址（DPH），低8位地址（DPL）与数据复用P0口。MOVX @Ri和MOVX @DPTR在P3.6(WR#)和P3.7(RD#)上产生读或者写信号。堆栈指针可位于内部256字节RAM中的任意位置。堆栈不能位于ERAM中。

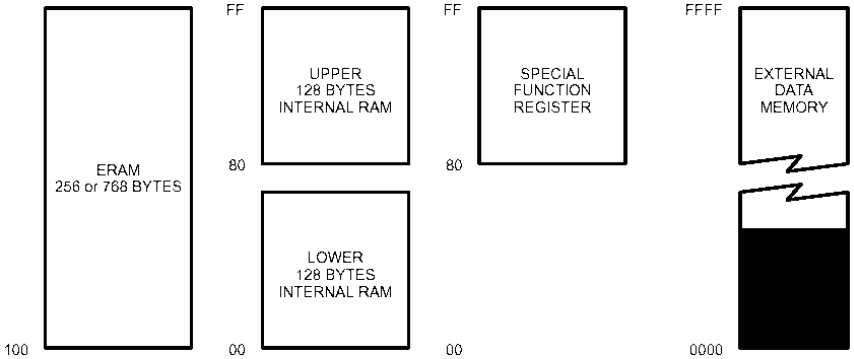


图28　内部和外部数据存储器地址空间(EXTRAM=0)

硬件看门狗定时器(WDT)

WDT在CPU落入软件运行失控的情况下可作为一种恢复的方法。WDT包含一个14位计数器和看门狗定时器复位SFR（WDTRST）。WDT在上电时被禁止。若想使能WDT，用户必须将01EH和0E1H依次写入WDTRST（地址0A6H）。当WDT使能后，在振荡器运行的情况下每过一个机器周期加1。此后无法禁止WDT，除非通过复位（硬件复位或者WDT溢出复位）。当WDT溢出时，它会在RST脚产生一个复位脉冲。

使用WDT

要使能 WDT，用户必须将 01EH 和 0E1H 依次写入 WDTRST。当 WDT 使能后，用户需要定期将 01EH 和 0E1H 写入 WDTRST 以避免 WDT 溢出。14 位计数器在到达 16383（3FFFH）后溢出并将芯片复位。当 WDT 使能后，在振荡器运行的情况下每过一个机器周期加 1。这意味着用户必须在 16383 个机器周期内至少复位 WDT 一次。要想复位 WDT，用户必须将 01EH 和 0E1H 写入 WDTRST。WDTRST 是一个只写寄存器。复位脉冲持续 98＊Tosc（12 时钟模式下为 196），此处 Tosc=1/fosc。

极限参数

参数	额定值	单位
操作温度	0～+70 或－40～+85	℃
贮存温度范围	－65～+150	℃
$\overline{EA}$/Vpp 脚相对于 Vss 的电压	0～+13.0	V
其它任何脚相对于 Vss 的电压	－0.5～+6.5	V
每个 I/O 脚的最大 I_{OL}	15	mA
功率损耗（指器件表面的发热，而非器件的功耗）	1.5	W

DC 电气特性

Tamb=0℃～+70℃或－40℃～+85℃;5V±10%;Vss=0V

标号	参数	测试条件	极限			单位
			最小	典型[1]	最大	
V_{IL}	输入低电压	4.5V<Vcc<5.5V	－0.5		0.2Vcc-0.1	V
V_{IH}	输入高电压(P0,1,2,3,$\overline{EA}$)		0.2Vcc+0.9		Vcc+0.5	V
V_{IH1}	输入高电压,XTAL1,RST		0.7Vcc		Vcc+0.5	V
V_{OL}	输出低电压,P1,2,3[8]	Vcc=4.5V I_{OL}=1.6mA[2]			0.4	V
V_{OL1}	输出低电压,P0 口,ALE,$\overline{PSEN}$[7,8]	Vcc=4.5V I_{OL}=3.2mA[2]			0.45	V
V_{OH}	输出高电压,P1,2,3[3]	Vcc=4.5V I_{OH}=－30uA	Vcc-0.7			V
V_{OH1}	输出高电压 ALE[9],$\overline{PSEN}$[3]	Vcc=4.5V I_{OH}=－3.2mA	Vcc-0.7			V
I_{IL}	逻辑 0 输入电流,P1,2,3	V_{IN}=0.4V	－1		－75	V
I_{TL}	逻辑 1 到 0 的转变电流,P1,2,3[6]	V_{IN}=2.0V 参见注释 4			－650	V
I_{LI}	输入漏电流,P0	0.45<V_{IN}<Vcc－0.3			±10	uA
I_{CC}	电源电流(参见图 36): 激活模式,空闲模式,掉电模式或 时钟停止(参见图 42 的条件) 编程和擦除模式	参见注 5 Tamb=0℃～+70℃ Tamb=－40℃～+85℃ fosc=20MHz		 <1 60	 40 50 	 uA uA mA
R_{RST}	内部复位下拉电阻		40		225	kΩ
C_{IO}	管脚电容[10]（$\overline{EA}$ 脚除外）				15	pF

注：

1) 不能保证典型值，因为这些值是在室温、5V 下测得。
2) P0 口和 P2 口上的容性负载会产生噪声叠加到 1 口、3 口和 ALE 的低电平上。噪声产生的原因是在总线操作期间，0 口和 2 口从 1 到 0 的跳变会使外部总线电容对口 0 和口 2 管脚放电，在最恶劣的情况下（容性负载>100pF），ALE 管脚上的噪声脉冲可超过 0.8V。在这种境况下，可以通过施密特触发器或者带有施密特触发 STROBE 输入的地址锁存器来校正 ALE。I_{OL} 会超过测试条件下的电流。
3) 容性负载加到口 0 和口 2 会导致 ALE 和 PSEN 管脚瞬时低于 VCC—0.7V，当地址位稳定下来。
4) 当口 1、2、3 被外部电路拉低时，口上从 1 到 0 的跳变将产生跳变电流，当输入电压大约在 2v 时，跳变电流达到最大。
5) 图 22 到 25 所示为 Icc 的测试条件，图 21 为 Icc 和频率的关系。
   激活模式： $I_{CC(MAX)}$=(0.56×FREQ.+8.0)mA
   空闲模式： $I_{CC(MAX)}$=(0.30×FREQ.+2.0)mA
6) 应用温度 T＝0℃~+70℃。
7) 口 0、 ALE 和 $\overline{PSEN}$ 脚的负载电容为 100pF，其他输出口为 80 pF。
8) 在稳定的状态条件下，Io 低被外部限制如下：
   i. 每个管脚的最大 I_{OL} 15mA（注：85℃规格）
   ii. 每个 8 位口的最大 I_{OL} 26 mA
   iii. I_{OL} 输出最大总和 71mA

iv. 如果 I_{OL} 超过测试条件，V_{OL} 可能会超过相应规格。不能保证超过测试电流。

9) ALE 的测试是 ALE 关断情况下，测出 ALE 的高电位值。

10) 管脚电容特性并不由测试得出，而是由其特性保证。管脚电容小于 25 pF。陶瓷电容小于 15pF（EA 是 25pF）。

AC 电气特性

T_{amb} = 0°C to +70°C or –40°C to +85°C, V_{CC} = 5 V ±10%, V_{SS} = 0V[1, 2, 3]

SYMBOL	FIGURE	PARAMETER	VARIABLE CLOCK[4]		33MHz CLOCK		UNIT
			MIN	MAX	MIN	MAX	
$1/t_{CLCL}$	14	Oscillator frequency Speed versions: I;J;U (33 MHz)	3.5	33	3.5	33	MHz
t_{LHLL}	14	ALE pulse width	$2t_{CLCL}$–40		21		ns
t_{AVLL}	14	Address valid to ALE low	t_{CLCL}–25		5		ns
t_{LLAX}	14	Address hold after ALE low	t_{CLCL}–25		5		ns
t_{LLIV}	14	ALE low to valid instruction in		$4t_{CLCL}$–65		55	ns
t_{LLPL}	14	ALE low to $\overline{PSEN}$ low	t_{CLCL}–25		5		ns
t_{PLPH}	14	$\overline{PSEN}$ pulse width	$3t_{CLCL}$–45		45		ns
t_{PLIV}	14	$\overline{PSEN}$ low to valid instruction in		$3t_{CLCL}$–60		30	ns
t_{PXIX}	14	Input instruction hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	14	Input instruction float after $\overline{PSEN}$		t_{CLCL}–25		5	ns
t_{AVIV}	14	Address to valid instruction in		$5t_{CLCL}$–80		70	ns
t_{PLAZ}	14	$\overline{PSEN}$ low to address float		10		10	ns
Data Memory							
t_{RLRH}	15, 16	$\overline{RD}$ pulse width	$6t_{CLCL}$–100		82		ns
t_{WLWH}	15, 16	$\overline{WR}$ pulse width	$6t_{CLCL}$–100		82		ns
t_{RLDV}	15, 16	$\overline{RD}$ low to valid data in		$5t_{CLCL}$–90		60	ns
t_{RHDX}	15, 16	Data hold after $\overline{RD}$	0		0		ns
t_{RHDZ}	15, 16	Data float after $\overline{RD}$		$2t_{CLCL}$–28		32	ns
t_{LLDV}	15, 16	ALE low to valid data in		$8t_{CLCL}$–150		90	ns
t_{AVDV}	15, 16	Address to valid data in		$9t_{CLCL}$–165		105	ns
t_{LLWL}	15, 16	ALE low to $\overline{RD}$ or $\overline{WR}$ low	$3t_{CLCL}$–50	$3t_{CLCL}$+50	40	140	ns
t_{AVWL}	15, 16	Address valid to $\overline{WR}$ low or $\overline{RD}$ low	$4t_{CLCL}$–75		45		ns
t_{QVWX}	15, 16	Data valid to $\overline{WR}$ transition	t_{CLCL}–30		0		ns
t_{WHQX}	15, 16	Data hold after $\overline{WR}$	t_{CLCL}–25		5		ns
t_{QVWH}	16	Data valid to $\overline{WR}$ high	$7t_{CLCL}$–130		80		ns
t_{RLAZ}	15, 16	$\overline{RD}$ low to address float		0		0	ns
t_{WHLH}	15, 16	$\overline{RD}$ or $\overline{WR}$ high to ALE high	t_{CLCL}–25	t_{CLCL}+25	5	55	ns
External Clock							
t_{CHCX}	18	High time	17	t_{CLCL}–t_{CLCX}			ns
t_{CLCX}	18	Low time	17	t_{CLCL}–t_{CHCX}			ns
t_{CLCH}	18	Rise time		5			ns
t_{CHCL}	18	Fall time		5			ns
Shift Register							
t_{XLXL}	17	Serial port clock cycle time	$12t_{CLCL}$		360		ns
t_{QVXH}	17	Output data setup to clock rising edge	$10t_{CLCL}$–133		167		ns
t_{XHQX}	17	Output data hold after clock rising edge	$2t_{CLCL}$–80		50		ns
t_{XHDX}	17	Input data hold after clock rising edge	0		0		ns
t_{XHDV}	17	Clock rising edge to input data valid		$10t_{CLCL}$–133		167	ns

AC 特性符号说明

每一时序符号有 5 个字符。首先是“t”（=时间），其字符基于他们的位置、名称和逻辑状态说明如下：

A－地址

C－时钟

D－输入数据

H－逻辑高电平

I－指令

L－逻辑低低电平或 ALE

Z－悬浮

P－$\overline{PSEN}$

Q－数据输出

R－$\overline{RD}$ 信号

t－时间

W－$\overline{WR}$ 信号

X－不再是有效逻辑电平

V－有效

例如：t_{AVLL}=从地址有效到 ALE 为低的时间。

t_{llpl}=从 ALE 为低到 PSEN 为低的时间。

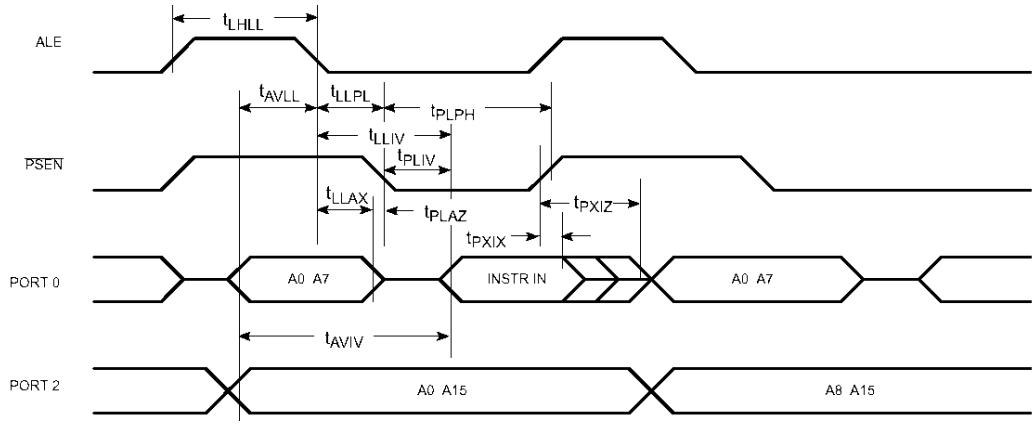


图 29 外部程序存储器的读周期

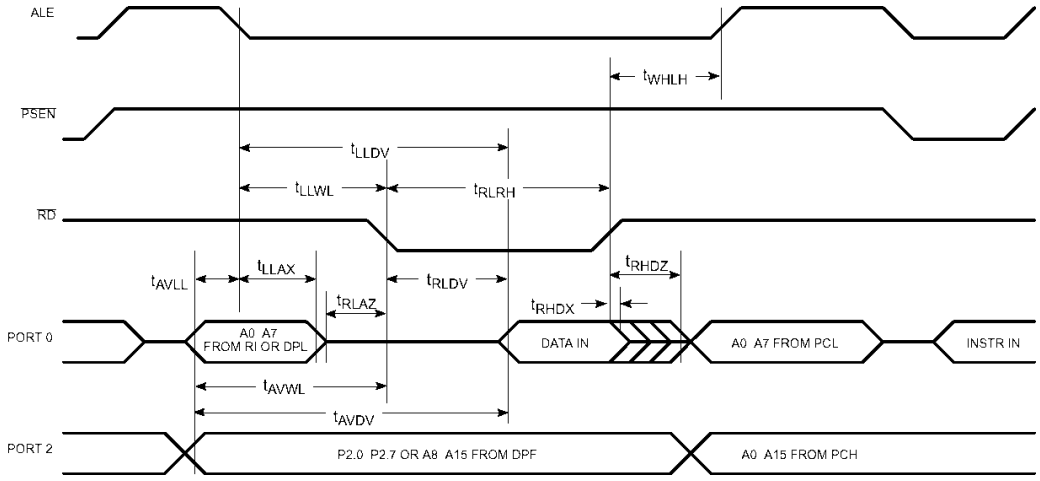


图 30 外部数据存储器的读周期

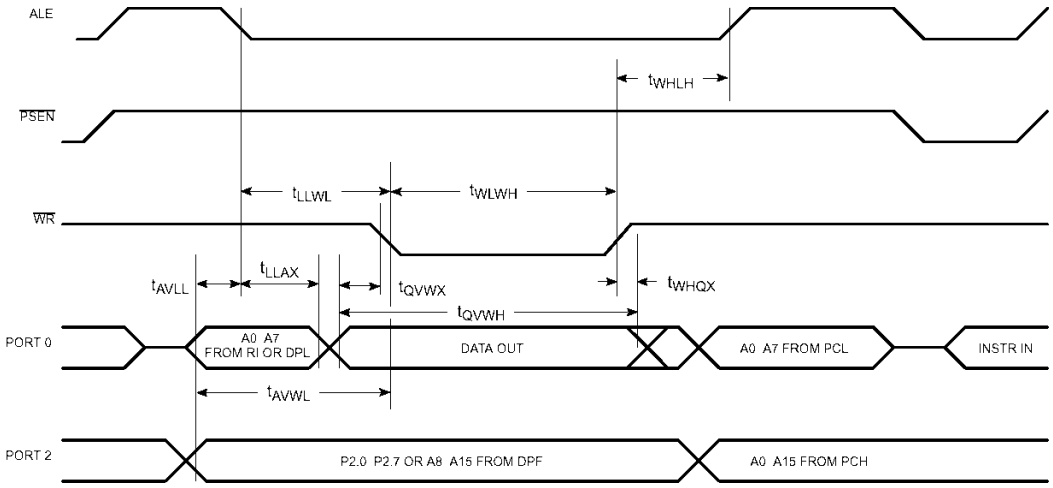


图 31 外部数据存储器的写周期

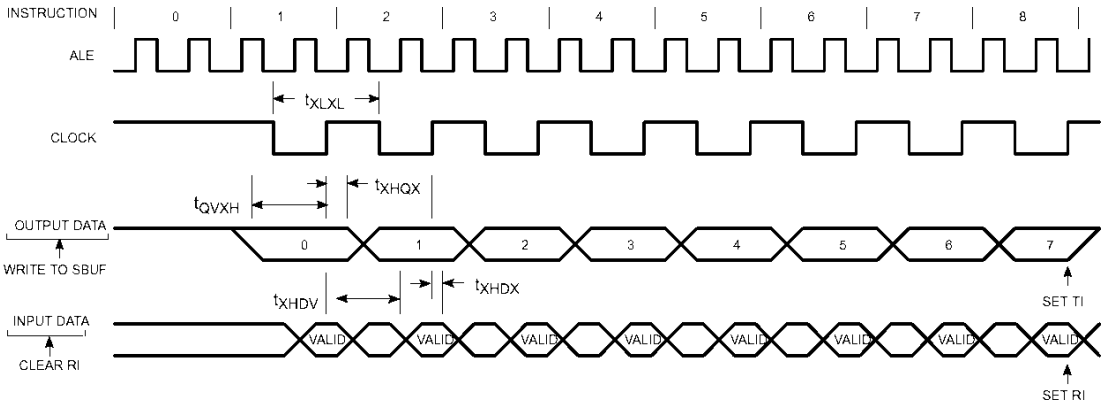


图 32 移位寄存器模式时序

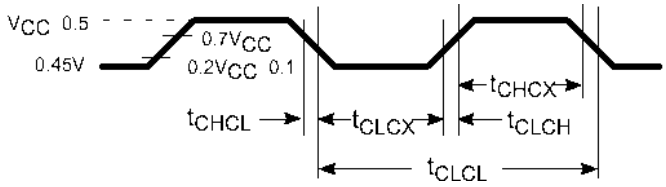
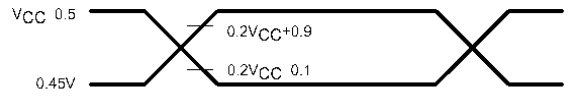
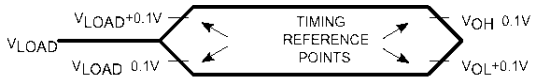


图 33 外部时钟驱动



NOTE:
AC inputs during testing are driven at V_{CC} 0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at V_{IH} min for a logic '1' and V_{IL} max for a logic '0'.

图 34 AC 输入/输出测试



NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded V_{OH}/V_{OL} level occurs. $I_{OH}/I_{OL} \geq \pm 20mA$.

图 35 悬浮波形

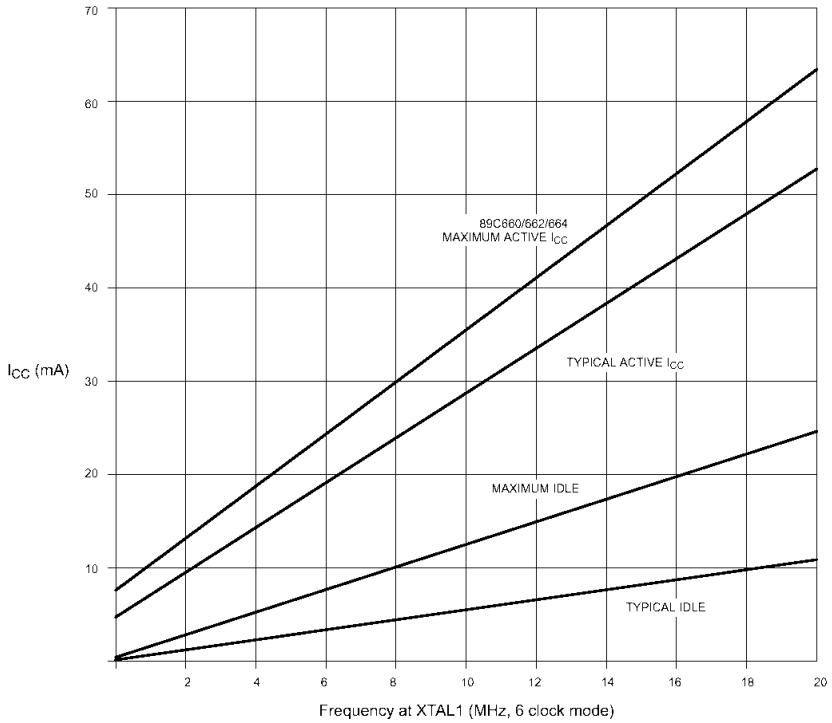


图 36 Icc 与频率的关系(频率范围内测得的有效数据)

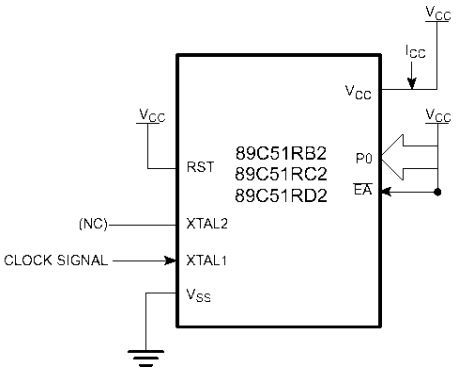


图 37 Icc 测试条件(激活模式;未连接其他管脚)

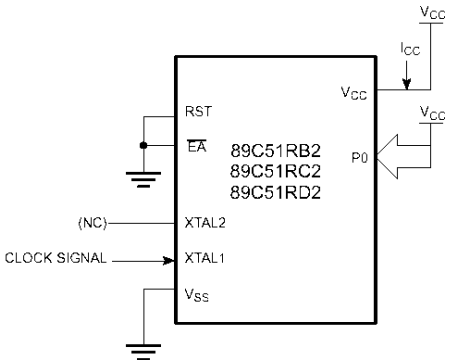


图 38 Icc 测试条件(空闲模式; 未连接其他管脚)

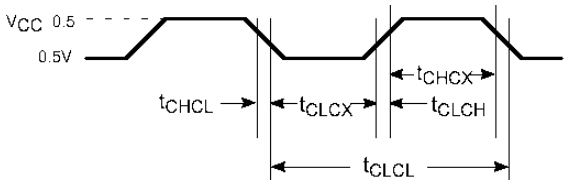


图 39 用于 Icc 测试的时钟信号波形(激活和空闲模式下)

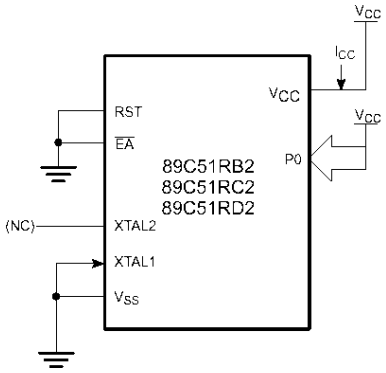


图 40 Icc 测试条件(掉电模式; Vcc=2V~5.5V)

FLASH EPROM 存储器

概述

P89C51RB2/RC2/RD2 Flash 存储器增加了 EPROM 所没有的电可擦除和编程特性。对芯片的擦除操作将整个程序存储区都擦除。块擦除功能可实现对任意 Flash 块进行擦除。在系统编程和标准的并行编程都是可行的。片内产生的擦除和写入时序为用户提供了良好的编程接口。

P89C51RB2/RC2/RD2 Flash 存储器甚至在经过 10,000 次擦除和编程之后仍然能可靠地保存存储器的内容。存储单元的设计使擦除和编程结构最优化。此外，先进的沟道氧化工艺和低内部电场的结合使擦除和编程操作更加可靠。P89C51RB2/RC2/RD2 使用+5VVpp 即可实现编程和擦除算法。

在系统编程(ISP)和在应用中编程(IAP)特性

- Flash EPROM 带块擦除的内部程序存储器
- 内部固化了 1K 字节的 Boot ROM，包含了一个低级的在系统编程子程序和一个默认的串行装载程序。用户可调用这些程序来实现在应用中编程(IAP)。可将 Boot ROM 关闭以提供对整个 64k Flash 存储器的访问。
- Boot 向量允许用户将 Flash 装载代码放入 Flash 存储器内的任何位置。这种配置为用户提供了应用的灵活性。
- Boot ROM 中的默认装载子程序允许通过串口进行编程，不需要用户提供装载程序。
- 如果内部程序存储区被禁止(EA=0)，外部程序空间最多可达 64k 字节。
- 编程和擦除电压为+5V(可承受+12V)。
- 使用 ISP/IAP 进行读/编程/擦除:

  －字节编程(20μs)

  －典型的快速擦除时间：

  块擦除(8k 或 16k 字节)小于 10 秒

  完全擦除(64k 字节)小于 20 秒

- 并行编程的硬件接口与 87C51 兼容
- 在系统编程
- 可对 Flash 程序代码进行编程加密
- 每个字节至少 10,000 次擦除/编程周期
- 数据至少可保存 10 年

89C51Flash 微控制器的性能

Flash 的结构

P89C51RB2/RC2/RD2 包含 16KB/32KB/64KB 的 Flash 程序存储器。程序存储器分为 5 个单独的块。前两个程序块大小为 8k 字节，占据地址 0 到 3FFFH。后 3 个程序块大小为 16k 字节，占据地址 4000H 到 0FFFFH。如图 41 所示。

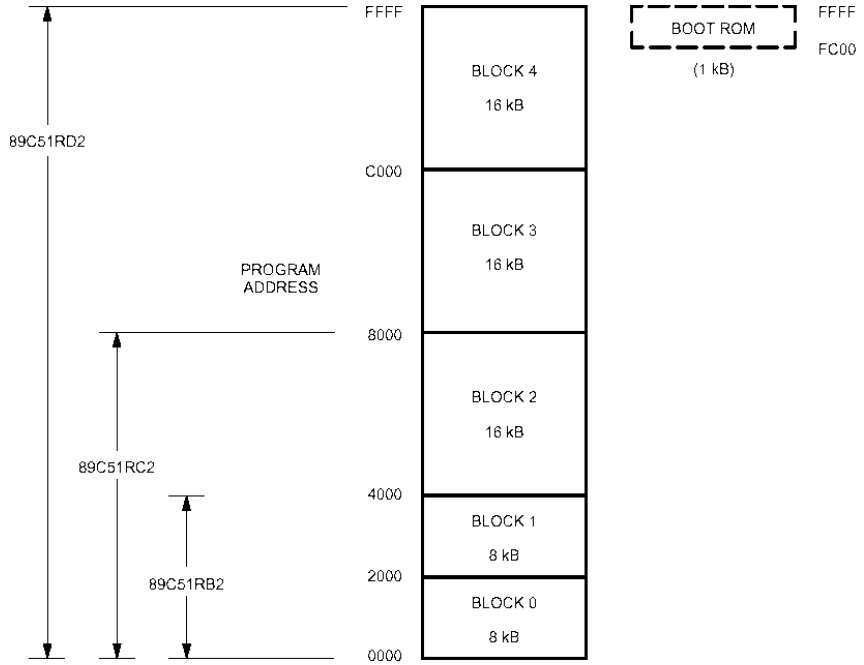


图 41 Flash 存储器结构

Flash 的编程和擦除

有三种方法可实现对Flash的编程或擦除。第一，在应用程序中调用Boot ROM中的擦除/写入等子程序进行串行的在运行中编程(IAP)。第二，用Boot ROM固件由主机进行串行的在系统中编程ISP。第三，使用商用编程器进行并行编程。对该器件进行并行编程的方法与87C51相似，但不完全相同。商用编程器必须支持该器件。

Boot ROM

当微控制器对自身的Flash存储器进行编程时，所有操作的细节都由固化在1k字节Boot ROM中的代码进行处理。Boot ROM与Flash存储器是各自独立的。用户程序简单调用Boot ROM中带适当参数的公共入口即可实现所需要的操作。Boot ROM的操作包括以下内容：擦除程序块，编程字节，校验字节，编程保密位等等。当Boot ROM打开时，它与从0FC00到0FFFFH的程序存储器的地址相重叠。可以将Boot ROM关闭，这样就可对这一部分的程序存储区进行访问。

上电复位代码的执行

P89C51RB2/RC2/RD2包含两个特殊的Flash寄存器：BOOT VECTOR和STATUS BYTE（注意：它们不在SFR中而在FLASH空间里）。在复位信号的下降沿，P89C51RB2/RC2/RD2检查STATUS BYTE中的内容。如果为0，则转去0000H地址开始执行程序。这是用户应用代码的正常起始地址。如果STATUS BYTE不为0，则将BOOT

VECTOR的值作为程序计数器的高字节，低字节固定为00H。工厂的默认设定为0FCH，对应Boot ROM的地址0FC00H。

注：当对FLASH进行擦除时，FLASH寄存器STATUS BYTE和BOOT VECTOR也同时被擦除。所以必须对它们重新编程。

在系统编程(ISP)

ISP编程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P89C51RB2/RC2/RD2的远程编程。固件由Pilips提供并嵌入到每一个芯片当中。

Pilips的在系统编程特性使嵌入式应用中的在电路编程成为可行，并最大限度减小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TxD、RxD、Vcc、Vss和Vpp（见图42）。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。

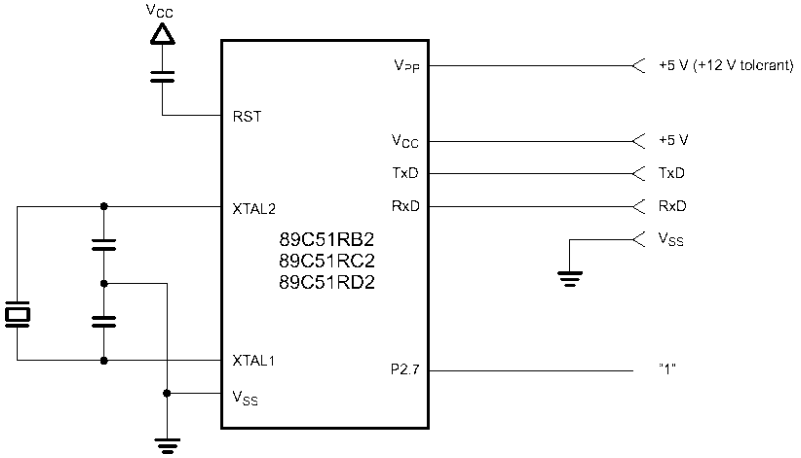


图42 在系统编程的管脚配置

使用在系统编程(ISP)

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。并用这个信息根据基于振荡器频率的定时器值对波特率进行编程。为此ISP特性要求主机必须向芯片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>

其中‘NN’表示记录中数据字节的个数。P89C51RB2/RC2/RD2最多可接受16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。数据字节的最大数目限制为16。ISP命令汇总见表8。

收到记录后，P89C51RB2/RC2/RD2将数据保存并计算校验和。在整个记录接收完毕后，执行由记录类型所指示的操作。如果校验和发生错误，芯片从串口发出‘X’表示校验和错。如果校验结果与记录相符，在大多数情况下，从串口发出‘.’表示正确接收。随后按照类型号完成指定的操作。

对于数据记录(记录类型00)会有一个额外的校验。在校验和相符且对记录中所有字节都成功编程后，才发送字符‘.’。对于数据记录，‘X’表示校验和不相符，‘R’表示其中一个字节编程错误。需要在对数据编程之前发送类型02记录（指定振荡器频率）。ISP的特性使得它不要求指定振荡器频率用以产生波特率和编程脉冲的时序。用户只需要向芯片提供要求产生正确时序的信息。记录类型02可实现这一目的。

从Pilips Semiconductor网站可以得到通过PC实现ISP编程的软件WinISP。此外，网站上还提供第三方的商用串行和并行编程器的清单。

表 8 ISP 所用 Intel-Hex 记录

记录类型	命令/数据 功能
00	编程数据 :nnaaaa00dd....ddcc 此处: nn = 记录的字节数（hex） aaaa = 记录中首字节的地址 dd....dd = 编程数据字节 cc = 校验和 例:10008000AF5F67F0602703E0322CFA92007780C3FD
01	文件结束，无操作 :xxxxxx01cc 此处: xxxxxx = 无关位 cc = 校验和 例:00000001FF
02	定义振荡器频率 :01xxxx02ddcc 此处: xxxx = 无关位 dd = 与振荡器频率最近的整数（MHz） cc = 0100000210ED（dd = 10h = 16,适用于 16.0~16.9MHz)
03	不同的写功能：nnxxxx03ffssddcc 此处：nn = 记录的字节数（hex） xxxx= 无关值 03 = 写功能 ff = 子功能代码 ss = 选择的代码 dd = 数据输入（如果需要） cc = 校验和 子功能代码=01（擦除块） ff = 01 ss = 块代码 块 0， 0k 到 8k, 00H 块 1， 8k 到 16k,20H 块 2，16k 到 32k,40H 块 3，32k 到 48k,80H 块 4，48k 到 64k,C0H 例：0200000301C03A 擦除块 4 子功能代码=04（擦除 Boot 向量和状态字节） ff = 04 ss = 无关 例：020000030400F7 擦除Boot向量和状态字节 子功能代码=05(编程加密位)

	ff = 05 ss = 00 对加密位1编程(禁止写入FLASH) 01 对加密位2编程(禁止校验FLASH) 02 对加密位3编程(禁止执行外部存储器代码) 例：020000030501F5 对加密位2编程 子功能码=06(编程Boot向量和状态字节) ff = 06 ss = 00 对状态字节编程 01 对Boot向量编程 例：010000030601FCF7 Boot 向量编程为 0FCH 子功能码=07（整片擦除） 擦除所有FLASH块、加密位以及将Boot向量和状态字节设置为缺省值 ff = 07 ss = 无关 dd = 无关 例：0100000307F5 整片擦除
04	显示器件数据或查空—整个Flash阵列的内容经串口送出以一定的格式显示。包括一个地址以及其后的16个字节数据。如果加密位2已编程，将不会显示器件的内容。向串口发送数据由接收到的任意字符启动并由接收到的任意字符终止。 格式：05xxxx04sssseeeeffcc 其中 05 = 记录中数据字节数 xxxxx = 所要求的字段，但其值为“无关” 04 = “显示器件数据或查空”功能代码 ssss = 起始地址 eeee = 结束地址 ff = 子功能代码： 00 或01 00 = 显示数据 01 = 查空 cc = 校验和 例：0500000440004FFF0069 显示地址4000~4FFF FLASH段内容
05	杂项读功能 格式：02xxxx05ffsscc 其中 02 = 记录中数据字节数 xxxx = 所要求的字段，但其值为“无关” 05 = “杂项读”功能代码 ffss = 子功能和选择码 0000 = 读签名字节—生产厂ID(15H) 0001 = 读签名字节—器件ID #1(C2H) 0002 = 读签名字节—器件ID #2 0700 = 读加密位 0701 = 读状态字节 0702 = 读Boot向量 cc = 校验和 例 :020000050001F8 读签名字节—器件 ID #1

06	直接装载波特率 格式：02xxxx06hhllcc 其中 02 = 记录中数据字节数 xxxx = 所要求的字段，但其值为“无关” 06 = “直接装载波特率”功能代码 hh = 定时器2高字节 ll = 定时器2低字节 cc = 校验和 例 :02000006F500F3 定时器 2 值 F500

在应用中编程方法

通过一个应用程序对几个IAP子程序的调用，可实现有选择地对FLASH块进行擦除和编程。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FFF0H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。振荡器频率以兆赫为单位向下舍入到最近的整数。例如，晶振为11.0592MHz时，R0设定为11。结果返回到寄存器中。IAP调用见表9。

表 9 IAP 调用

IAP 调用	参数
编程数据字节	输入参数： R0 = 振荡器频率(整数) R1 = 06h R1 = 86h(清零WDT) DPTR= 编程字节地址 ACC = 编程字节 返回参数 ACC = 00 表示通过，!00 表示失败
擦除 Flash 块	输入参数： R0 = 振荡器频率(整数) R1 = 0(快速擦除) R1 = 81h(清零WDT) DPH = 块代码如下： 块0， 0k~8k，00H 块1， 8k~16k，20H 块2，16k~32k，40H 块3，32k~48k，80H 块4，48k~64k，C0H DPL = 00H 返回参数 无
擦除 Boot 向量	输入参数： R0 = 振荡器频率(整数) R1 = 04h R1 = 84h(清零WDT) DPH = 00h DPL = 无关

	返回参数 无
编程保密位	输入参数： R0 = 振荡器频率(整数) R1 = 05h R1 = 85h(清零WDT) DPH = 00h DPL = 00h — 保密位#1(禁止写入Flash) 01h — 保密位#2(禁止对Flash进行校验) 02h — 保密位#3(禁止外部程序) 返回参数 无
编程状态字节	输入参数： R0 = 振荡器频率(整数) R1 = 06h R1 = 86h(清零WDT) DPH = 00h DPL = 01H –编程状态字节 ACC = 状态字节 返回参数 ACC = 状态字节
编程 Boot 向量	输入参数： R0 = 振荡器频率(整数) R1 = 06h R1 = 86h(清零WDT) DPH = 00h DPL = 01H –编程Boot向量 ACC = Boot向量 返回参数 ACC = Boot 向量
读器件数据	输入参数： R1 = 06h R1 = 86h(清零WDT) DPTR= 读出字节的地址 返回参数 ACC = 读出字节的值
读制造厂商 ID	输入参数： R0 = 振荡器频率(整数) R1 = 00h R1 = 80h(清零WDT) DPH = 00h DPL = 00H(制造厂商ID) 返回参数 ACC = 读出字节的值

读器件 ID #1	输入参数： R0 = 振荡器频率(整数) R1 = 00h R1 = 80h(清零WDT) DPH = 00h DPL = 01H(器件ID #1) 返回参数 ACC = 读出字节的值
读器件 ID #2	输入参数： R0 = 振荡器频率(整数) R1 = 00h R1 = 80h(清零WDT) DPH = 00h DPL = 02H(器件ID #2) 返回参数 ACC = 读出字节的值
读保密位	输入参数： R0 = 振荡器频率(整数) R1 = 07h R1 = 87h(清零WDT) DPH = 00h DPL = 00H(保密位) 返回参数 ACC = 读出字节的值
读状态字节	输入参数： R0 = 振荡器频率(整数) R1 = 07h R1 = 87h(清零WDT) DPH = 00h DPL = 01H(状态字节) 返回参数 ACC = 读出字节的值
读 Boot 向量	输入参数： R0 = 振荡器频率(整数) R1 = 07h R1 = 87h(清零WDT) DPH = 00h DPL = 02H(Boot向量) 返回参数 ACC = 读出字节的值

保密性

保密特性是为了保护软件的版权，防止 FLASH 的内容被读出。保密位位于 FLASH 中。P89C51RX2 有 3 个可编程保密位，可为片内的代码和数据提供不同级别的保护（见表 8）。与 ROM 和 OTP 版本不同的

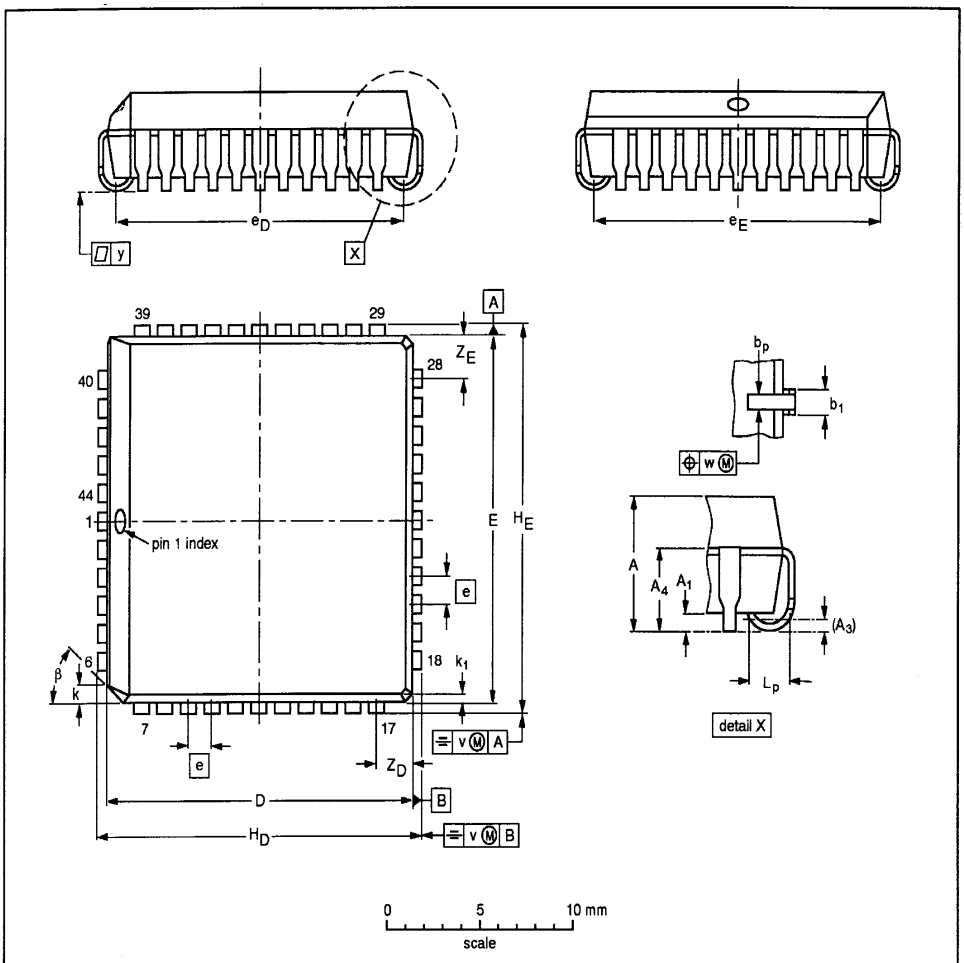
是，这 3 个保密位是独立的，LB3 包含了 LB1 的保密功能。

表 10

保密位 级别	保护描述
LB1	禁止外部程序存储器中的指令 MOVC 读取内部存储器的代码字节
LB2	禁止对程序进行校验
LB3	禁止执行外部程序

注：保密位是独立的

PLCC 封装:44 脚



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

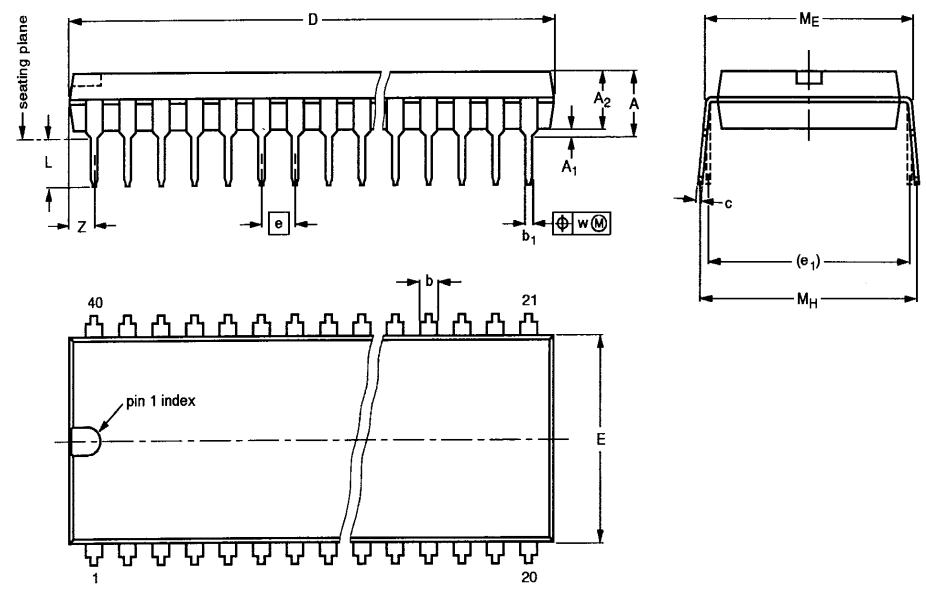
UNIT	A	A_1 min.	A_3	A_4 max.	b_p	b_1	$D^{(1)}$	$E^{(1)}$	e	e_D	e_E	H_D	H_E	k	k_1 max.	L_p	v	w	y	$Z_D^{(1)}$ max.	$Z_E^{(1)}$ max.	β
mm	4.57 4.19	0.51	0.25	3.05	0.53 0.33	0.81 0.66	16.66 16.51	16.66 16.51	1.27	16.00 14.99	16.00 14.99	17.65 17.40	17.65 17.40	1.22 1.07	0.51	1.44 1.02	0.18	0.18	0.10	2.16	2.16	
inches	0.180 0.165	0.020	0.01	0.12	0.021 0.013	0.032 0.026	0.656 0.650	0.656 0.650	0.05	0.630 0.590	0.630 0.590	0.695 0.685	0.695 0.685	0.048 0.042	0.020	0.057 0.040	0.007	0.007	0.004	0.085	0.085	$\pm 5^{o}$

Note

1. Plastic or metal protrusions of 0.01 inches maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT187-2	112E10	MO-047AC				~~95-02-25~~ 97-12-16

DIP 封装：40 脚



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

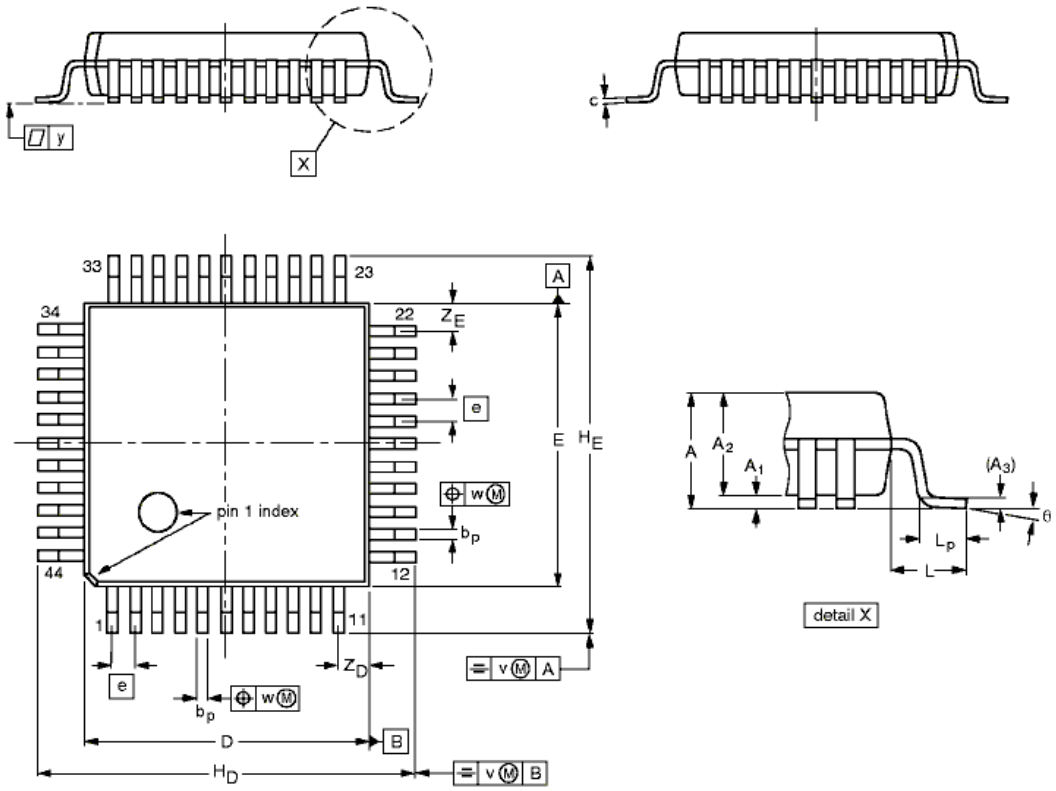
UNIT	A max.	A_1 min.	A_2 max.	b	b_1	c	$D^{(1)}$	$E^{(1)}$	e	e_1	L	M_E	M_H	w	$Z^{(1)}$ max.
mm	4.7	0.51	4.0	1.70 1.14	0.53 0.38	0.36 0.23	52.50 51.50	14.1 13.7	2.54	15.24	3.60 3.05	15.80 15.24	17.42 15.90	0.254	2.25
inches	0.19	0.020	0.16	0.067 0.045	0.021 0.015	0.014 0.009	2.067 2.028	0.56 0.54	0.10	0.60	0.14 0.12	0.62 0.60	0.69 0.63	0.01	0.089

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT129-1	051G08	MO-015AJ				~~92-11-17~~ 95-01-14

QFP 封装:44 脚



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	1.60	0.15 0.05	1.45 1.35	0.25	0.45 0.30	0.20 0.12	10.10 9.90	10.10 9.90	0.80	12.15 11.85	12.15 11.85	1.0	0.75 0.45	0.20	0.20	0.10	1.14 0.85	1.14 0.85	7^o 0^o