



第3章 STC单片机CPU子系统

何宾
2015.02

本章主要内容

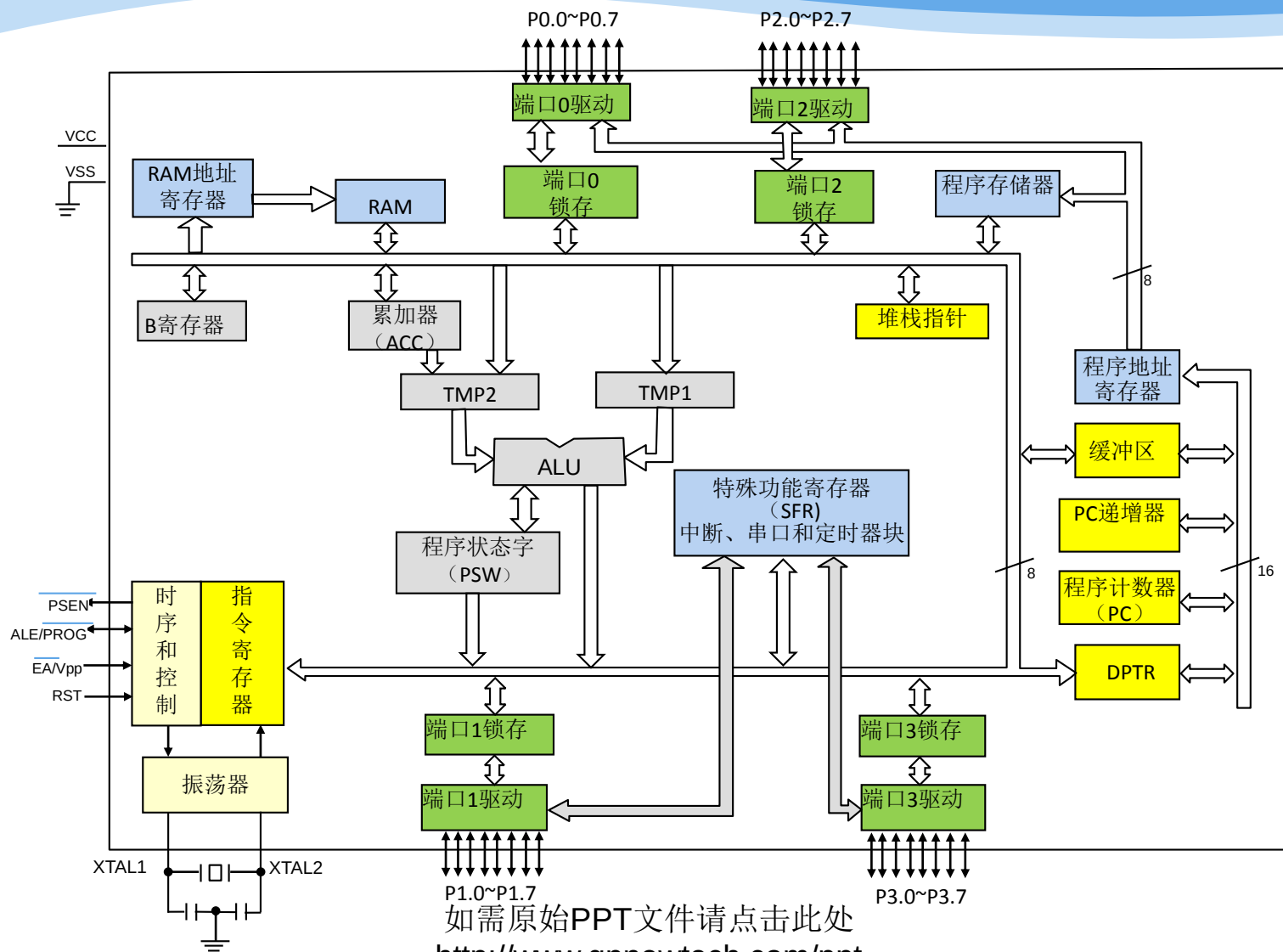
- STC单片机CPU内核功能单元
- STC单片机存储器结构和地址空间
- STC单片机中断系统原理及功能

STC单片机CPU内核功能单元

8051单片机自诞生的那天开始，到现在已经持续了30多年。

- 在这期间，人们对其性能不断的进行改进，使得其整体性能提高了10倍以上。
- 目前，以8051 CPU内核为核心的单片机仍然发挥着其巨大的生命力。
- 虽然8051 CPU的内核比较简单，但是以其为核心的单片机系统却包含了构成计算机系统的全部要素。

STC单片机CPU内核功能单元



STC单片机CPU内核功能单元

STC内的8051 CPU核是高性能、运行速度经过优化的8位中央处理单元（ Central Processing Unit , CPU ）。8051 CPU外围主要包括：

- 内部数据RAM；
- 外部数据空间；
- 特殊功能寄存器；
- CPU时钟分频器。

STC单片机CPU内核功能单元

STC 8051 CPU的特性主要包括：

- 采用流水线RISC结构，执行速度比工业标准8051快十几倍
- 与工业标准8051指令集100%兼容；
- 大多数指令使用1个或2个时钟周期执行；
- 256个字节的内部数据RAM；
- 使用双DPTR扩展标准8051结构；
- 提供了片外扩展的64KB外部数据存储器；
- 提供了多达21个中断源；
- 新特殊功能寄存器使能：快速访问STC单片机I/O端口，以及控制CPU时钟频率；

STC单片机CPU内核功能单元

任何一个中央处理单元CPU都包含有控制器和运算器两大基本模块。下面将通过STC单片机分析8051 CPU子系统的功能。

STC单片机CPU内核功能单元

--控制器

控制器是CPU中最重要的功能部件之一。其作用是控制CPU内的各个组成部件协调的工作，保证CPU的正常运行。

控制器

--程序计数器

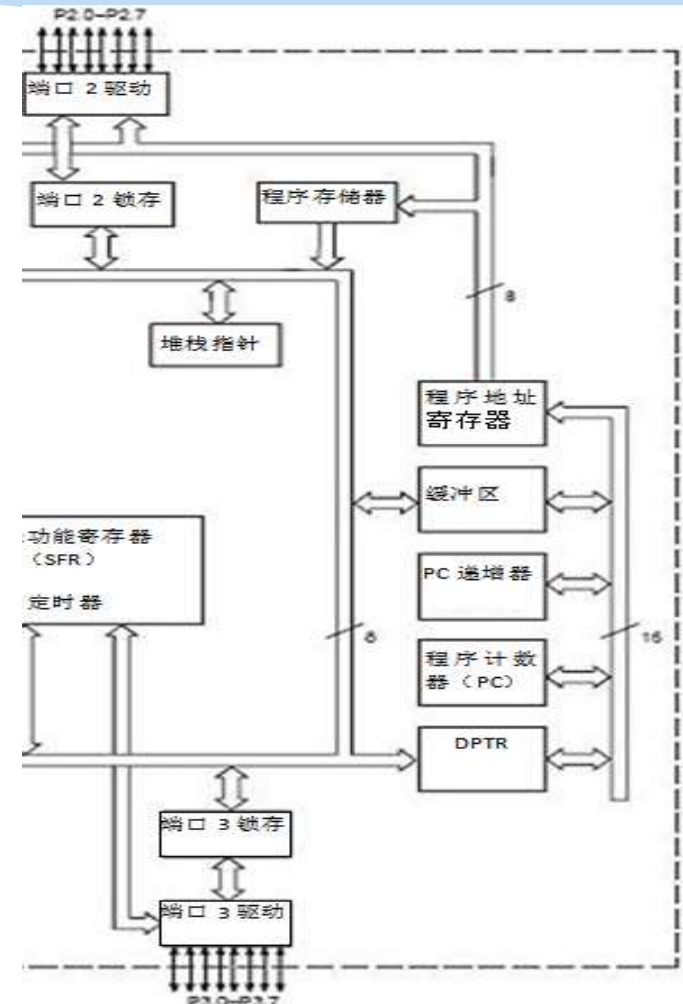
单片机最重要的特点之一就是采用了存储程序的体系结构，即需要执行的代码保存在一个称之为程序存储器的单元中。

- 通过程序计数器（Program Counter，PC）从程序存储器中源源不断地取出所要执行的代码。因此，程序计数器PC是CPU中最基本的控制部分。
- PC的特点就是总是指向下一条所要执行的指令的地址空间。
- 程序计数器、PC递增计数器、缓冲区、程序地址寄存器都挂在其结构右侧的一条总线上。程序地址寄存器的输出连接到程序存储器上，而程序存储器连接到内部总线上。

控制器

--程序计数器

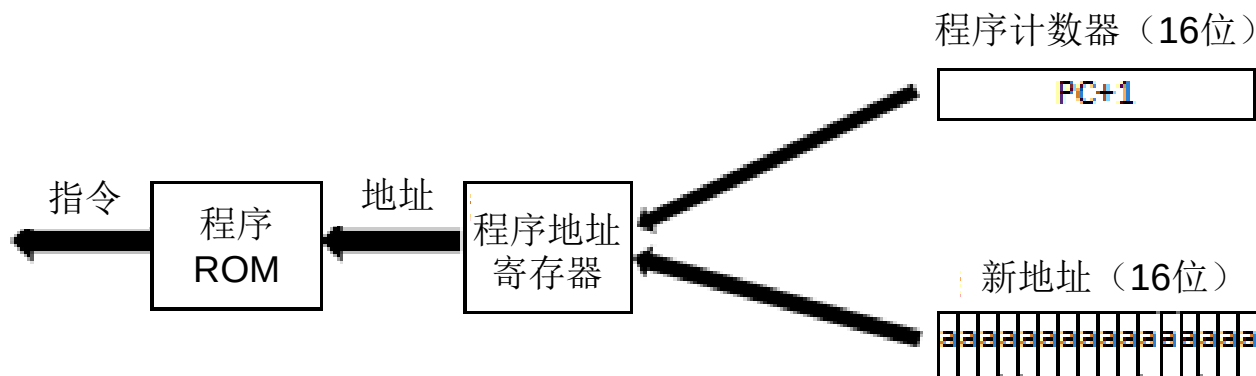
- 前面已经提到在程序存储器中，保存的是程序的机器代码，即机器指令。
- 从图中可以知道，程序地址寄存器的输出用于给程序存储器提供地址，而程序存储器的输出用于提供机器指令的内容。



控制器

--程序计数器

- 程序计数器其实质就是实现递增功能的计数器而已，只不过是因为计数器的计数值作为程序存储器的地址而已。
- 在上一图中，程序计数器的宽度为16位。也就是说，地址深度为 2^{16} ，地址的范围为0~65535，即64K。因此，程序存储器的深度最大为64K。
- 程序计数器并不能总是让程序地址寄存器递增。这是因为，执行机器指令可以分成顺序执行和非顺序执行，如下图。



控制器

--程序计数器

■ 顺序执行

- 是指按机器指令的前后顺序，顺序执行执行指令，即：把PC+1后的值送给程序地址寄存器，作为程序ROM的地址。
- 然后，程序ROM送出指令。这就是所说的，程序计数器总是指向下一条要执行的指令。

■ 非顺序执行

- 在编写的软件代码中，经常出现条件判断语句、跳转语句、程序调用语句和中断调用等。
- 因此，当执行程序代码的过程中遇到这些指令时，程序的执行顺序并不是按照PC+1->PC来执行程序，而是将这些语句所指向的新指令所在的新目标地址赋给程序地址寄存器。

STC单片机CPU内核功能单元

--指令通道

指令通道包含取指单元、译码单元、执行指令单元。
本质上，取值、译码和执行指令实质上就是一个有限自动状态机。也就是经常所说的，微指令控制器。

■ 取指单元

- 根据PC所指向的存放指令的程序存储器地址，取出指令。
- 在8051单片机中，程序存储器的宽度为8位。但是，8051的机器指令有8位、16位或24位，即：1个字节、2个字节或3个字节。
- 所以，对于不同的指令来说，取值令所需要的时钟周期并不相同。因此，可能需要几个时钟周期才能完成取指操作。

STC单片机CPU内核功能单元

--指令通道

■ 译码单元

- 根据取出指令的操作码部分，对指令进行翻译。
- 这个翻译过程，就是将机器指令转换成一系列的逻辑控制序列，这些控制序列将直接控制CPU内的运算单元。

■ 执行指令单元

- 当完成译码过程后，根据逻辑控制序列（微指令）所产生的逻辑行为，控制运算器单元，完成指令需要的实现的操作行为。

STC单片机CPU内核功能单元

--指令通道

比如，假设此时取出一条机器指令（汇编助记符表示）：

ADD A, Rn

其机器指令的格式为：

0010, 1rrr // rrr表示寄存器的编号

该指令完成寄存器Rn和累加器A数据相加的操作。

STC单片机CPU内核功能单元

--指令通道

■ 其译码和执行指令的过程应该包含：

□ 从寄存器Rn中取出数据送入ALU的一个输入端口TMP1，表示为：

$(Rn) \rightarrow (TMP1)$

□ 从累加器ACC中取出数据送入ALU另一个输入端口TMP2，表示为：

$(ACC) \rightarrow (TMP2)$

□ 将TMP1和TMP2的数据送到ALU进行相加，产生结果，表示为：

$(TMP2) + (TMP1) \rightarrow (总线)$

□ 将ALU产生的结果，通过内部总线送入到ACC累加器中。

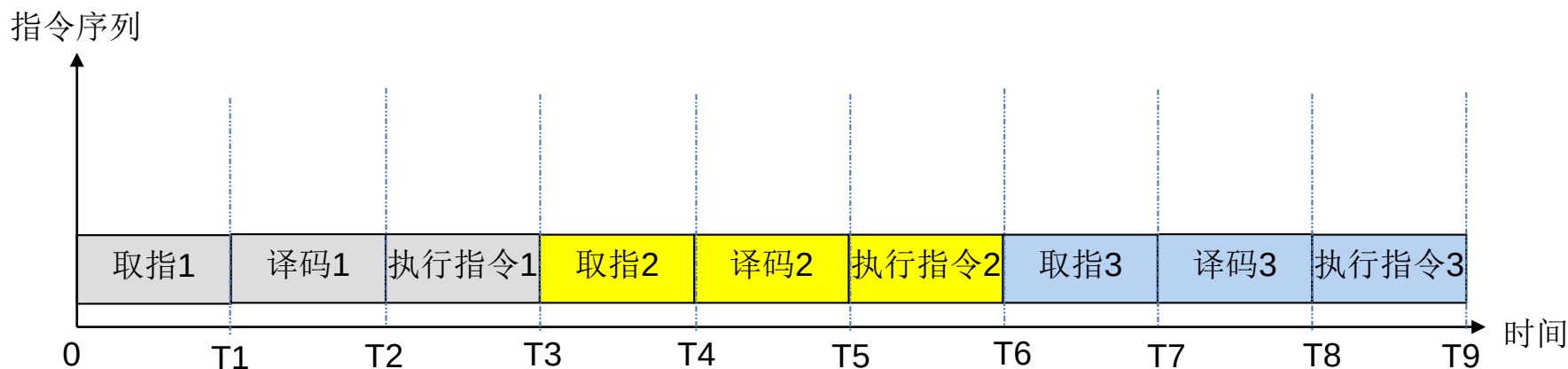
$(总线) \rightarrow (ACC)$

STC单片机CPU内核功能单元

--指令通道

流水线技术

传统8051单片机的指令通道采用的是串行结构，如下图所示。为了分析问题方便，下面假设每条指令的取值周期、译码周期和执行指令的周期都是一样的。当采用串行执行结构时，对于3条指令来说，需要T9个时间周期才能执行完成。



3条指令执行的串行执行结构原理

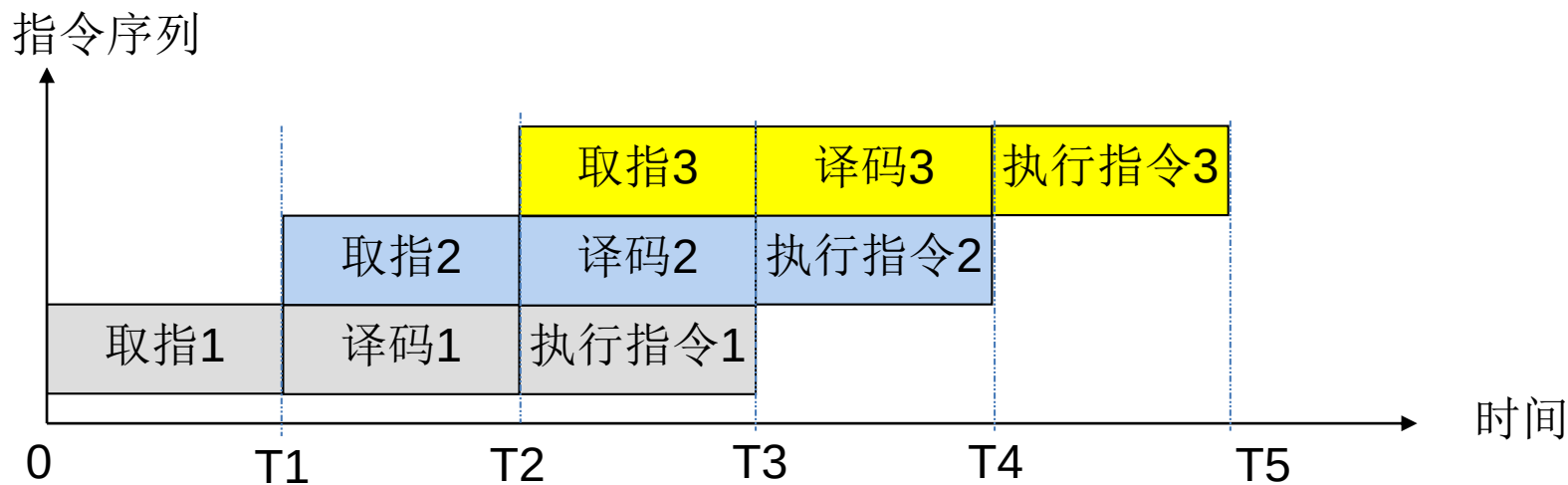
如需原始PPT文件请点击此处

<http://www.gpnewtech.com/ppt>

STC单片机CPU内核功能单元

--指令通道

与传统的8051 CPU取指通道相比，STC指令通道采用了改进后的（二级/三级）流水线结构，如图所示。从图中可以看出，当采用三级流水线结构以后，只需要T5个周期就能执行完3条指令。基于前面的假设条件，指令通道的吞吐量提高了1倍。当执行指令的数量增加时，流水线结构优势将更加明显。



3条指令执行的流水线结构原理

STC单片机CPU内核功能单元

--双数据指针

数据指针（Dual Data Pointer, DPTR）是一个16位的专用寄存器。

- 由DPL（低8位）和DPH（高8位）组成，其地址为82H（DPL低字节）和83H（DPH，高字节）。
- DPTR是8051中唯一可以直接进行16位操作的寄存器。
- 此外，也可以按照字节分别对DPH和DPL进行操作。

STC单片机CPU内核功能单元

--双数据指针

如果STC单片机没有外部数据总线，则该单片机只存在一个16位的数据指针。

- 如果单片机有外部数据总线，则该单片机设计了两个16位的数据指针DPTR0和DPTR1，这两个数据指针公用一个地址空间。
- 通过软件设置特殊功能寄存器（Special Function Register，SFR）中P_SW1（地址为A2H）的第0位选择数据指针。

Mnemonic	Address	Name	7	6	5	4	3	2	1	0	Reset Value
AUXR1 P_SW1	A2H	Auxiliary Register 1	S1_S1	S1_S0	CCP_S1	CCP_S0	SPI_S1	SPI_S0	0	DPS	0000,0000

STC单片机CPU内核功能单元

--双数据指针

- 数据指针选择 (Data Pointer Select , DPS) 来选择所使用的DPTR。
 - $DPS=0$, 选择DPTR0 ($0x83:0x82$) , 其中 : $0x83$ 为16位DPTR0的高寄存器DPH0 ; $0x82$ 为16位DPTR0的低寄存器DPL0 ;
 - $DPS=1$, 选择DPTR1 ($0x83:0x82$) , 其中 : $0x83$ 为16位DPTR1的高寄存器DPH1 ; $0x82$ 为16位DPTR1的低寄存器DPL1 ;

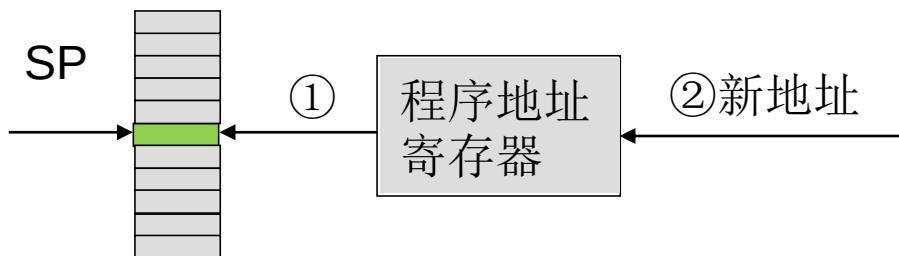
STC单片机CPU内核功能单元

--堆栈及指针

在单片机中，有一个称为堆栈的特殊存储空间。

■ 其作用主要用于保存现场。

□ 典型的，当在执行程序的过程中遇到跳转指令时，就需要将当前PC+1指向的下一跳指令的地址保存起来，等待执行完跳转指令的时候，再将所保存的下一条指令的地址恢复到程序地址寄存器中。



STC单片机CPU内核功能单元

--堆栈及指针

在STC单片机中，用于控制指向存储空间位置的是一个堆栈指针（Stack Pointer，SP）

- 它实际上就是一个8位的专用寄存器
- 该寄存器的内容就是栈顶的地址，也就是用于表示当前栈顶在内部RAM块中的位置。

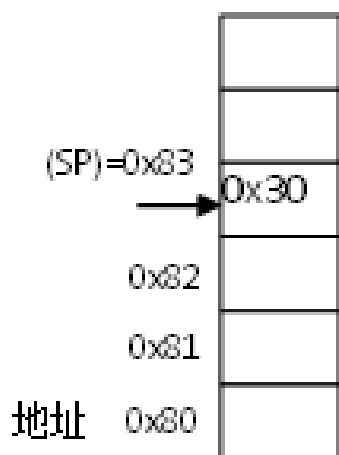
以三个数据0x30、0x31和0x32入栈和出栈为例。假设在对堆栈进行操作前，当前SP的内容为0x82，也就是SP指向堆栈存储空间地址为0x82的位置。

STC单片机CPU内核功能单元

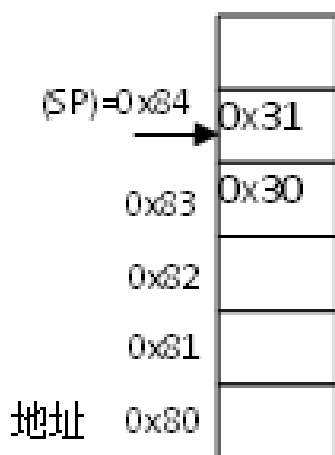
--堆栈及指针

入栈操作

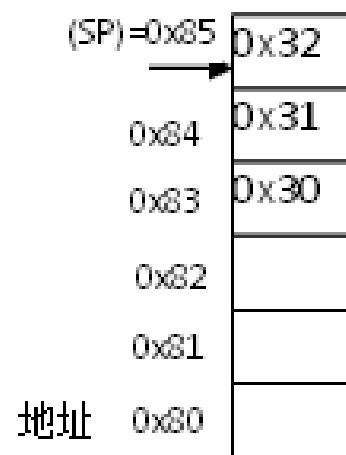
- 从该过程可以看出，随着数据的入栈操作，（SP）递增，SP总是指向最新保存的数据的存储器的位置，也就是通常所说的，SP总是指向栈顶的位置



(a) 数据 0x30 入栈



(b) 数据 0x31 入栈



(c) 数据 0x32 入栈

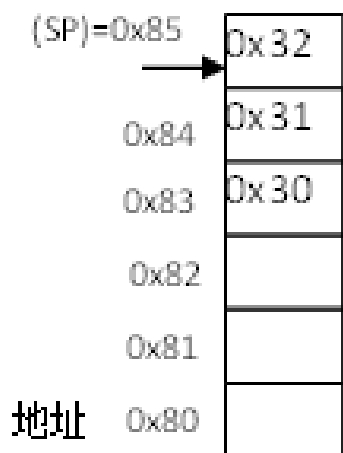
数据入栈操作

STC单片机CPU内核功能单元

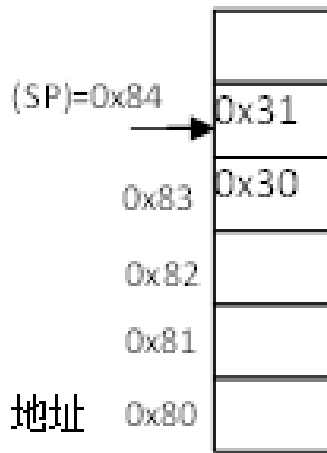
--堆栈及指针

出栈操作时

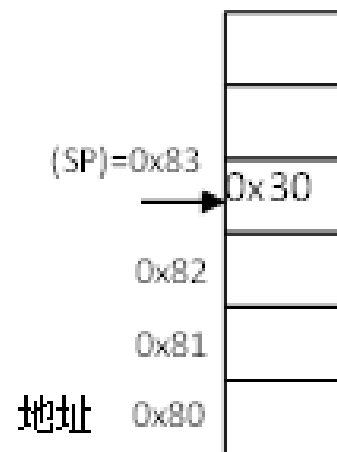
- 从该过程可以看出，随着数据的入栈操作，（SP）递增，SP总是指向最新保存的数据的存储器的位置，也就是通常所说的，SP总是指向栈顶的位置。



(a) 数据 0x32 出栈



(b) 数据 0x31 出栈



(c) 数据 0x30 出栈

数据出栈操作

STC单片机CPU内核功能单元

--运算器

运算器用于执行丰富的数据操作功能。STC的8051 CPU内的运算器包括：

- 8位算术逻辑单元
- 累加器
- B寄存器
- 程序状态字

运算器

--8位算术逻辑单元

STC的8051 CPU内的运算器中，最核心的部件就是**算术逻辑单元**（Arithmetic and Logic Unit，ALU）。8051 CPU的ALU位宽为8位，它可以实现的功能包括：

■ 算术运算

- 典型地，实现8位的加、减、乘和除运算；

■ 其它运算

- 典型地，实现递增、递减、BCD十进制调整和比较运算；

■ 逻辑运算

- 典型地，实现逻辑与（AND）、逻辑或（OR）、逻辑异或（XOR）逻辑取反（NOT）和旋转/移位操作；

■ 按位运算

- 典型地，置位、复位、取补、如果没有设置则跳转操作、如果设置则跳转并且清除操作和移入/移出进位标志寄存器；

运算器

--累加器

累加器 (Accumulator , ACC) , 也简写为A

- 用于大多数指令结果的累加器。
- 为累加器所分配的地址空间位于特殊功能寄存器 (Special Function Register , SFR) 0xE0的位置。

运算器

--B寄存器

在乘法和除法操作中，B寄存器有特殊用途；而在其它情况，它用于通用寄存器。

■ 乘法操作

- 参与乘法运算的一个操作数保存在B寄存器中，另一个保存在A寄存器中。
- 在乘法运算后，所得乘积的高8位保存在B寄存器中，而乘积的低8位保存在A寄存器中。

■ 除法操作

- 参与除法运算的被除数保存在A寄存器中，除数保存在B寄存器中
- 在除法运算后，所得的商保存在A寄存器中，而余数保存在B寄存器中。

运算器

--程序状态字

在程序状态字（Program Status Word，PSW）寄存器中，保存一些具有特殊含义的比特位，这些位反映当前8051 CPU内的工作状态。

■ 该寄存器位于SFR空间0xD0地址。下表给出PSW寄存器的内容。

比特位	7	6	5	4	3	2	1	0
名字	CY	AC	F0	RS1	RS0	OV	RSV	P

运算器

--程序状态字

■ CY

- 进位标志。
- 算术和位指令操作影响该位。
- 在进行加法运算时，如果最高位有进位；或者在进行减法运算时，如果最高位有借位，则将CY设置为1；否则设置为0。

■ AC

- 辅助进位标志。
- ADD，ADDC，SUBB指令影响该位。
- 在进行加法运算时，如果第3位向第4位有进位；或者在进行减法运算时，如果第3位向第4位有借位，则将AC设置为1；否则设置为0。
- 设置辅助进位标志的目的是为了便于BCD码加法、减法运算的调整。

运算器

--程序状态字

■ F0

□ 通用标志0。

■ RS1和RS0

□ 寄存器组选择位。用于选择不同的寄存器组，其含义如下表所示。

RS1	RS0	当前使用的工作寄存器组（（R0~R7）
0	0	0
0	1	1
1	0	2
1	1	3

运算器

--程序状态字

■ OV

- 溢出标志。
- ADD、ADDC、SUBB、MUL和DIV指令影响该位状态。

■ RSV

- 保留位。

■ P

- 奇偶标志位。
- 在每条指令执行后，设置或者清除该比特位，该位用于表示累加器ACC中1的个数。如果ACC中有奇数个1时，则将P设置为1；否则，如果ACC中有偶数个1（包括0个1的情况）时，则将P设置为0。

STC单片机CPU内核功能单元

--特殊功能寄存器

特殊功能寄存器（ Special Function Register , SFR ） , SFR是具有特殊功能的RAM区域 , 它是一系列控制寄存器和状态寄存器的集合。

- **这些寄存器用于对STC单片机内的各个功能模块进行管理、控制和监视。**

注：（ 1 ） STC15系列单片机的SFR和高128字节的RAM共用相同的地址范围 , 都使用80H~FFH的区域。因此 , 需要使用直接寻址方式访问SFR。

（ 2 ） 当寄存器地址能够被8整除时才可以进行位操作 , 其他区域不可以进行位操作。

STC单片机CPU内核功能单元

--特殊功能寄存器

地址	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	地址
0xF8	P7	CH	CCAP0H	CCAP1H	CCAP2H				0xFF
0xF0	B	PWMCFG	PCA_PWM0	PCA_PWM1	PCA_PWM0	PWMCR	PWMIF	PWMFDCR	0xF7
0xE8	P6	CL	CCAP0L	CCAP1L	CCAP2L				0xEF
0xE0	ACC	P7M1	P7M0				CMPCR1	CMPCR2	0xE7
0xD8	CCON	CMOD	CCAPM0	CCAPM1	CCAPM2				0xDF
0xD0	PSW	T4T3M	T4H RL_TH4	T4L RL_TL4	T3H RL_TH3	T3L RL_TL3	T2H RL_TH2	T2L RL_TL2	0xD7
0xC8	P5	P5M1	P5M0	P6M1	P6M0	SPSTAT	SPCTL	SPDAT	0xCF
0xC0	P4	WDT_CONTR	IAP_DATA	IAP_ADDRH	IAP_ADDRL	IAP_CMD	IAP_TRIG	IAP_CONTR	0xC7
0xB8	IP	SADEN	P_SW2		ADC_CONTR	ADC_RES	ADC_RESL		0xBF
0xB0	P3	P3M1	P3M0	P4M1	P4M0	IP2	IP2H	IPH	0xB7
0xA8	IE	SADDR	WKTCL WKCTL_CNT	WKTCH WKTCH_CNT	S3CON	S3BUF		1E2	0xAF
0xA0	P2	BUS_SPEED	AUXR1 P_SW1						0xA7
0x98	SCON	SBUF	S2CON	S2BUF		P1ASF			0x9F
0x90	P1	P1M1	P1M0	P0M1	P0M0	P2M1	P2M0	CLK_DIV PCON2	0x97
0x88	TCON	TMOD	TL0 RL_TL0	TL1 RL_TL1	TH0 RL_TH0	TH1 RL_TH1	AUXR	INT_CLK0 AUXR2	0x8F
0x80	P0	SP	DPL	DPH	S4CON	S4BUF		PCON	0x87

如需原始PPT文件请点击此处

<http://www.gpnewtech.com/ppt>

35

特殊功能寄存器

--端口控制寄存器组

在SFR中，提供了用于对单片机P0组端口、P1组端口、P2组端口、P3组端口、P4组端口和P5组端口的控制寄存器。

■ P0端口寄存器组

□ P0寄存器、P0M0寄存器、P0M1寄存器。

■ P1端口寄存器组

□ P1寄存器、P1M0寄存器、P1M1寄存器。

特殊功能寄存器

--端口控制寄存器组

■ P2端口寄存器组

□ P2寄存器、P2M0寄存器、P2M1寄存器。

■ P3端口寄存器组

□ P3寄存器、P3M0寄存器、P3M1寄存器。

■ P4端口寄存器组

□ P4寄存器、P4M0寄存器、P4M1寄存器。

特殊功能寄存器

--端口控制寄存器组

PyM1.x和PyM0.x的组合含义

P0M1	P0M0	含义
0	0	准双向端口，与传统8051 I/O模式兼容。其灌电流可达20mA，拉电流为270μA (注：由于制造误差，拉电流实际在150~270μA之间)
0	1	推挽输出，即：强上拉输出，电流可达到20mA。因此，在使用时需要接入限流电阻
1	0	仅为输入（高阻）
1	1	开漏（Open Drain），内部上拉电阻断开。在该模式下，既可以读外部状态也可以对外输出高电平/低电平。因此，需要加上拉电阻，否则读不到外部状态，也不能对外输出高电平。

特殊功能寄存器

--端口控制寄存器组

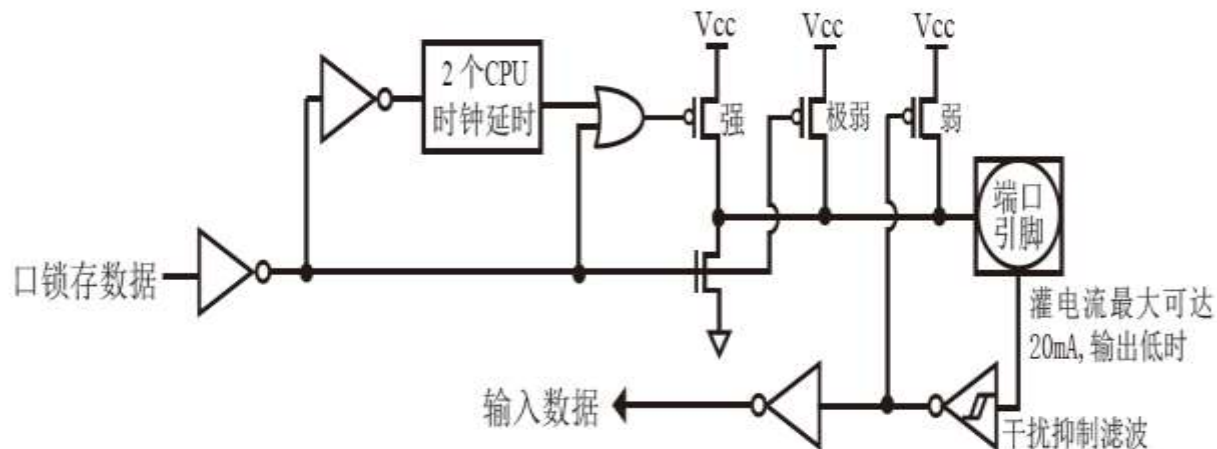
I/O口有四种不同的配置模式

- 准双向输出配置
- 强推挽输出配置
- 仅为输入（高阻）配置
- 开漏输出配置

特殊功能寄存器

--端口控制寄存器组

■ 准双向输出配置



- 准双向输出类型可以用作输出和输入功能，而不需要重新配置I/O口输出状态。
 - 当I/O输出为高时，驱动能力很弱，允许外部设备将其拉低（要尽量避免出现这种情况）；
 - 而当引脚的输出为低时，驱动能力很强，可吸收很大的电流。
- 在准双向口提供3个上拉晶体管（“弱上拉”、“极弱上拉”和“强上拉”）以满足不同的要求。

特殊功能寄存器

--端口控制寄存器组

第1个晶体管，称为弱上拉晶体管。

- 当口锁存数据置1且引脚本身为1时打开，此上拉提供基本驱动电流使准双向口输出为1。如果一个引脚输出为1而由外部设备下拉到低时，弱上拉晶体管关闭而极弱上拉晶体管维持打开状态，为了把这个引脚强拉为低，外部设备必须有足够的灌电流能力使引脚上的电压降到门限电平以下。
 - 对于5V供电的单片机来说，弱上拉晶体管的电流大约为250 μ A；
 - 对于3.3V供电的单片机来说，弱上拉晶体管的电流大约为150 μ A。

特殊功能寄存器

--端口控制寄存器组

第二个上拉晶体管，称为极弱上拉晶体管。

- 当口锁存数据置为1时，该晶体管导通。
- 当引脚悬空时，这个极弱的上拉源产生很弱的上拉电流将引脚上拉到高电平。
 - 对于5V供电的单片机来说，极弱上拉晶体管的电流约为 $18\mu\text{A}$ ；
 - 对于3.3V单片机来说，极弱上拉晶体管的电流约为 $5\mu\text{A}$ 。

特殊功能寄存器

--端口控制寄存器组

第三个上拉晶体管，称为强上拉晶体管。

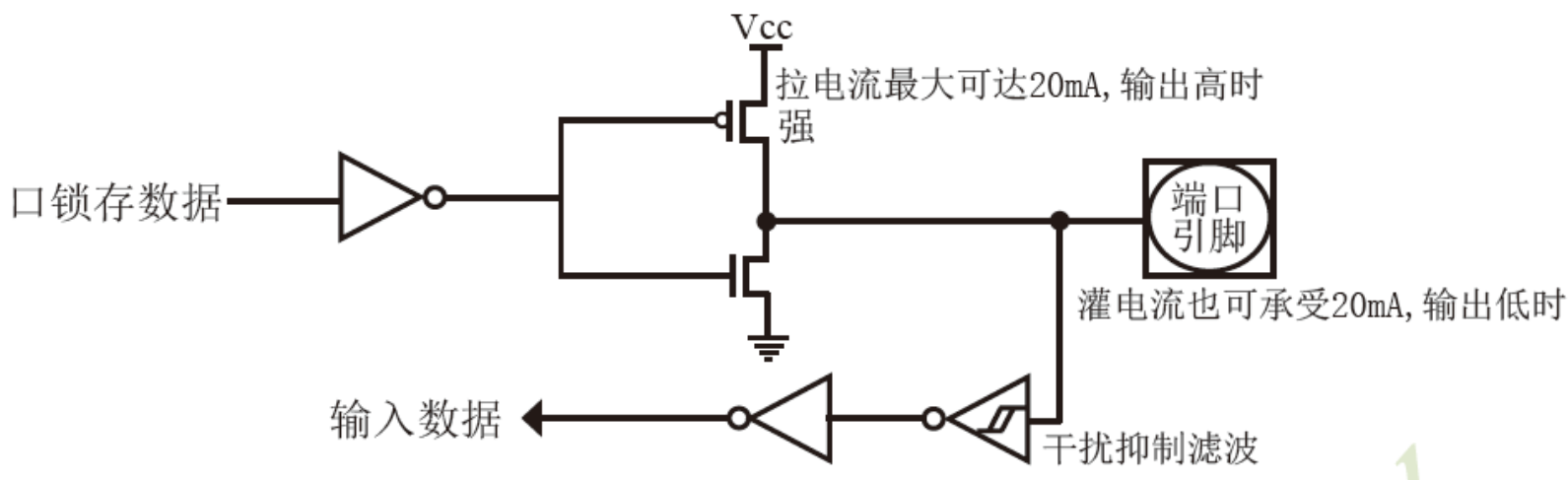
- **当口锁存数据由0变化到1时，这个上拉用于加快准双向口由逻辑0到逻辑1的转换。**
- **当出现这种情况时，强上拉打开约2个时钟以使引脚能够迅速地上拉到高电平。**

特殊功能寄存器

--端口控制寄存器组

强推挽输出配置

- 强推挽输出配置的下拉结构与开漏输出以及准双向口的下拉结构相同。
- 但当口锁存数据为1时，提供持续的强上拉。
- 推挽模式一般用于需要更大驱动电流的情况。

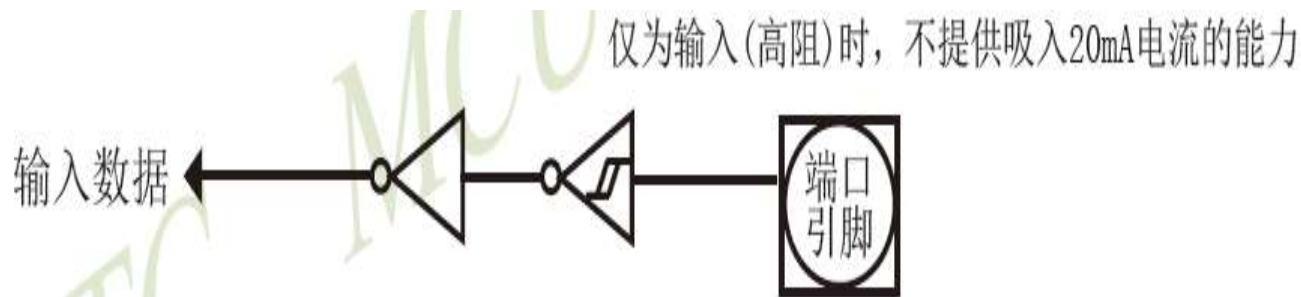


特殊功能寄存器

--端口控制寄存器组

仅为输入（高阻）配置

- 输入口带有一个施密特触发器输入以及一个干扰抑制电路。

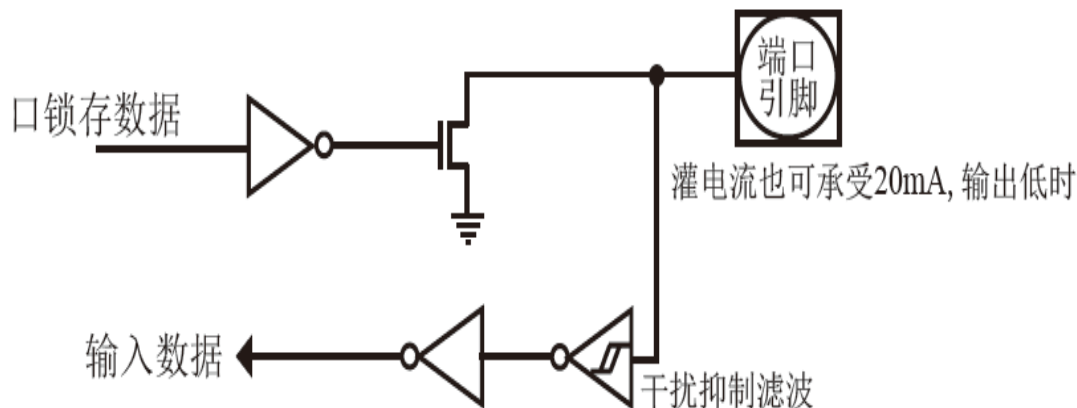


特殊功能寄存器

--端口控制寄存器组

开漏输出配置

- 开漏模式下，既可以读取外部状态也可以对外输出高电平/低电平。
- 如果要正确地读取外部状态或者需要对外输出高电平时，需要外加上拉电阻。



注：由于8051 CPU的时钟速度较高。因此，当软件执行由低变高的指令后，加入1~2个空操作延迟指令，在读取外部状态。

特殊功能寄存器

--时钟分频器

CPU分频器允许CPU运行在不同的速度。

□ 用户通过配置用户SFR空间地址为0x97的CLK_DIV (PCON2) 寄存器来控制CPU的时钟速度。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	MCKO_S1	MCKO_S0	ADRJ	Tx_Rx	MCLKO_2	CLKS2	CLKS1	CLKS0

特殊功能寄存器

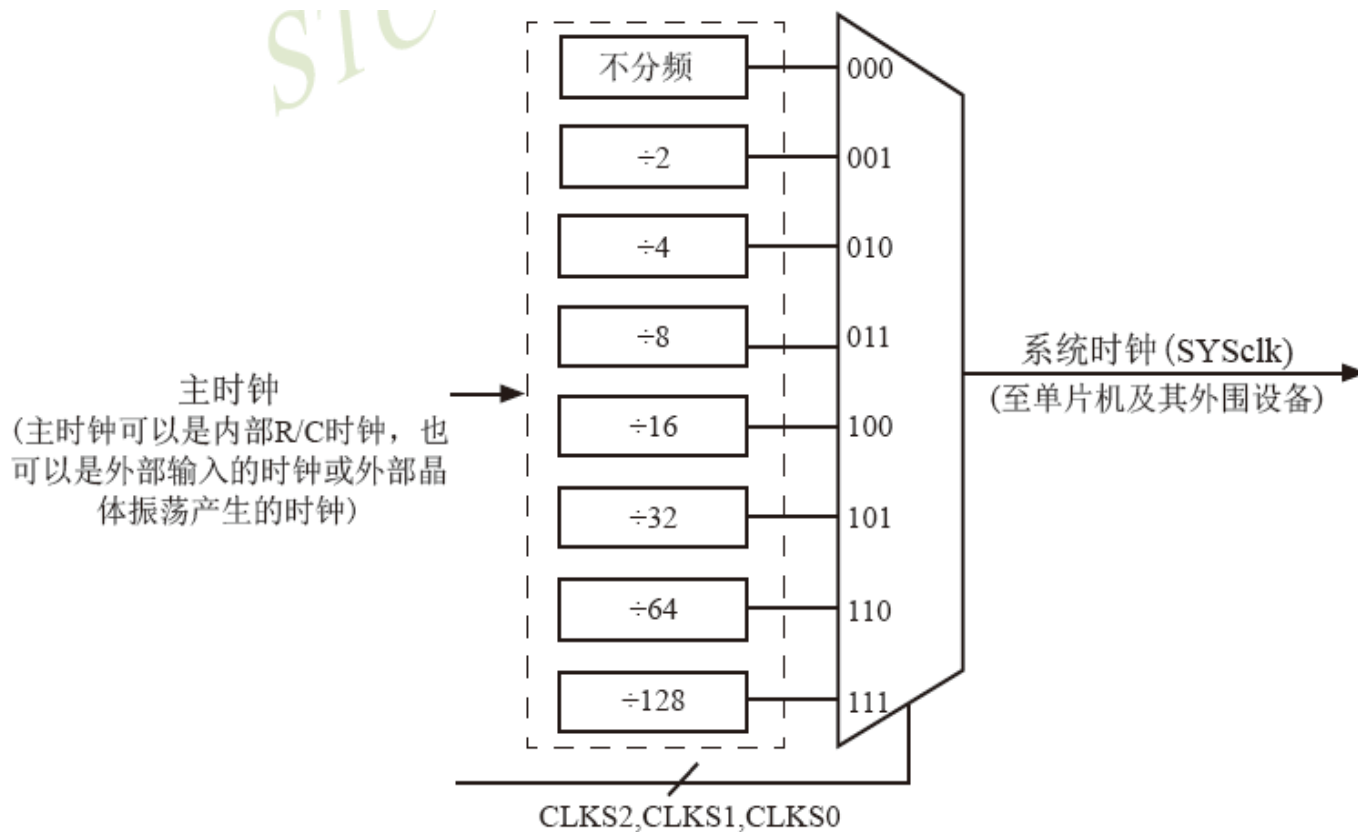
--时钟分频器

其中，CLKS2~CLKS0比特位用于对主时钟进行分频，如下所示。

CLKS2	CLKS1	CLKS0	含义
0	0	0	主时钟频率/1
0	0	1	主时钟频率/2
0	1	0	主时钟频率/4
0	1	1	主时钟频率/8
1	0	0	主时钟频率/16
1	0	1	主时钟频率/32
1	1	0	主时钟频率/64
1	1	1	主时钟频率/128

特殊功能寄存器

--时钟分频器



注：主时钟可以是内部的R_C时钟，也可以是外部输入时钟/外部晶体振荡器产生的时钟。

特殊功能寄存器

--时钟分频器

其中，B6和B7比特位用于控制在STC单片机引脚MCLKO（P5.4）或者MCLKO_2（P1.6）输出时钟的频率，如下表所示。

MCKO_S1	MCKO_S0	含义
0	0	主时钟不对外输出时钟
0	1	输出时钟，输出时钟频率=SYSclk的时钟频率
1	0	输出时钟，输出时钟频率=SYSclk的时钟频率/2
1	1	输出时钟，输出时钟频率=SYSclk的时钟频率/4

注：SYSclk为系统时钟。

特殊功能寄存器

--时钟分频器

STC15W系列单片机通过CLK_DIV寄存器的B3位
(MCLKO_2) 选择:

- 在MCLKO/P5.4口对外输出时钟.
- 还是在MCLKO_2/XTAL2/P1.6口对外输出时钟。

CLKO_2	含义
0	在MCLKO/P5. 4口对外输出时钟
1	在MCLKO_2/XTAL2/P1. 6口对外输出时钟

程序Flash存储器

--程序存储器空间映射

STC单片机程序空间位于0x0000~0xFFFF的地址范围。

- 16位的PC指向下一条要执行的指令。
- CPU只能通过使用MOVC指令，从程序空间读取数据。
- 当单片机复位后，程序计数器（PC）的内容为0x0000。
 - 因此，从程序存储器地址为0x0000的地方开始执行程序。另外，中断服务程序的入口地址（也称为中断向量）也放在程序存储单元。

注：在STC15W4K32S4系列的单片机内部集成了8K~61K字节的Flash程序存储器，如下表所示。

程序Flash存储器

--程序存储器空间映射

类型	程序存储器
STC15W4K08S4	0x0000~0x1FFF (8K)
STC15W4K16S4	0x0000~0x3FFF (16K)
STC15W4K24S4	0x0000~0x5FFF (24K)
STC15W4K32S4	0x0000~0x7FFF (32K)
STC15W4K40S4	0x0000~0x9FFF (40K)
STC15W4K48S4	0x0000~0xBFFF (48K)
STC15W4K56S4	0x0000~0xDFFF (56K)
STC15W4K60S4	0x0000~0xEFFF (60K)
IAP15W4K58S4	0x0000~0xE7FF (58K)
IAP15W4K61S4	0x0000~0xF3FF (61K)

程序Flash存储器

--程序存储器特点



STC的程序存储器可以保存用户程序、数据和表格信息，其具有下面的特点：

- 提供10万次以上擦写能力；
- 低压保护功能，即：在低压状态下，禁止对程序存储器进行擦除和编程；
- 程序存储器对外不提供读电路。因此，有效地防止对用户程序的破解；
- 只有对程序存储器进行擦除操作后，才能对其进行编程操作；
- 在对程序存储器编程时，可以将程序代码乱序后存放；
- 程序存储器的最后7个字节设置全球唯一的ID号；
- 以扇区为单位擦除；
- 以字节为单位进行编程；
- STC提供了通过通用异步串口，对Flash进行擦除、编程和代码加密的能力。

STC单片机存储器结构和地址空间

--数据Flash存储器

STC系列单片机内部提供了大容量的数据Flash存储器，用于实现电可擦除的只读存储器（Electrically Erasable Programmable Read-Only Memory，EEPROM）功能。

- 数据Flash存储器和程序Flash存储器空间是分开的。

STC单片机存储器结构和地址空间

--数据Flash存储器

其特点主要包括：

- 通过ISP/IAP技术可以将内部的数据Flash当作EEPROM使用；
- 擦写次数在10万次以上；
- 以扇区位单位，每个扇区包含512个字节。
- 数据存储器的擦除操作是按扇区进行的。

注：由于EEPROM是以扇区为单位的，数据存储器的擦除操作是按扇区进行的。因此，建议同一次修改的数据保存在同一个扇区，不是同一次修改的数据需要保存在不同扇区。

STC单片机存储器结构和地址空间

--数据Flash存储器

在STC单片机内提供了用于IAP和EEPROM进行数据擦除和保存的特殊功能寄存器，这些寄存器位于SFR内。

■ ISP/IAP数据寄存器

- 用于ISP/IAP的数据寄存器称为IAP_DATA，它位于STC单片机SFR内地址为0xC2的位置。
- 当ISP/IAP从EEPROM读出的数据放在该寄存器中，向EEPROM写的数据也放在这个寄存器中。
- 当复位时，该寄存器值设置为0xFF。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ ISP/IAP地址寄存器

- 用于ISP/IAP的地址寄存器由IAP_ADDRH和IAP_ADDRL构成，这两个寄存器分别位于STC单片机SFR内地址为0xC3和0xC4的位置。
- 当进行ISP/IAP时，分别用于保存高八位和第八位地址。
- 当复位时，这两个寄存器的值设置为0x00。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ ISP/IAP命令寄存器

- 用于ISP/IAP的命令寄存器称为IAP_CMD，它位于STC单片机SFR内地址为0xC5的位置。

IAP_CMD寄存器各位的含义

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	--	--	--	--	--	--	MS1	MS0

注：当复位时，该寄存器的值设置为xxxxxx00。在该寄存器中只有MS1和MS0比特位有意义，用于控制对数据Flash（EEPROM）的操作。

STC单片机存储器结构和地址空间

--数据Flash存储器

MS1	MS0	功能
0	0	待机模式，无
0	1	从用户的应用程序区对数据
1	0	从用户的应用程序区对数据
1	1	从用户的应用程序区对数据

注：（1）当程序在用户应用程序区时，仅可以对数据Flash（EEPROM）进行字节读/字节编程/扇区擦除。

（2）IAP15系列单片机可以在用户应用程序区修改用户应用程序区。

（3）可以通过MOVC指定读数据Flash（EEPROM），但是起始地址不是0x0000，而是程序存储器空间结束地址的下一个地址。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ IAP/ISP命令触发寄存器

- 用于IAP/ISP的命令触发寄存器称为IAP_TRIG，它位于STC单片机SFR内地址为0xC6的位置。当复位时，该寄存器的值设置为XXXXXXXX。
- 当ICAP_CONTR寄存器B7位的IAPEN设置为1时，对命令触发寄存器先写入0x5A，然后再写入0xA5。这样，ISP/IAP命令才能生效。
- IAP/ISP操作完成后，IAP地址高8位寄存器IAP_ADDRH、IAP地址低8位寄存器IAP_ADDRL，以及IAP命令寄存器IAP_CMD的内容不变。如果接着要对下一个地址的数据进行ISP/IAP操作，需要手动将该地址的高8位和低8位分别写到IAP_ADDRH和IAP_ADDRL寄存器。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ ISP/IAP控制寄存器

□ 用于ISP/IAP的控制寄存器称为IAP_CONTR，它位于STC单片机SFR内地址为0xC7的位置，如下表所示。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	IAPEN	SWBS	SWRST	CMD_FAIL	—	WT2	WT1	WT0

□ 当复位时，该寄存器的值设置为0000x000。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ IAPEN

- ISP/IAP功能使能位。当该位为0时，禁止IAP读/写/擦除数据Flash（EEPROM）；当该位为1时，允许IAP读/写/擦除数据Flash。

■ SWBS

- 在复位后，软件选择从用户应用程序区启动还是从系统ISP监控程序区启动。当该位为1时，选择从系统ISP监控程序区启动；当该位为0时，选择从用户应用程序区启动。

注：需要与SWRST一起配合使用。

■ SWRST

- 软件复位控制位。当该位为0时，表示没有复位操作；当该位为1时，软件控制产生复位，单片机自动复位

STC单片机存储器结构和地址空间

--数据Flash存储器

■ CMD_FAIL：命令失败指示位。

□ 如果IAP地址（由IAP_ADDRH和IAP_ADDRL寄存器确定）指向非法地址或者无效地址，且已经发送了ISP/IAP命令，并对IAP_TRIG发送0x5A~和0xA5触发失败，则CMD_FAIL为1。

注：需要使用软件清除该位。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ WT2~WT0

□ 设置等待时间，如下表所示。

设置等待时间			CPU			
WT2	WT1	WT0	读（2个周期）	编程（=55 μ S）	扇区擦除（=21ms）	推荐的系统时钟
1	1	1	2个周期	55个时钟	21012个时钟	$\geq 1\text{MHz}$
1	1	0	2个周期	110个时钟	42024个时钟	$\geq 2\text{MHz}$
1	0	1	2个周期	165个时钟	63036个时钟	$\geq 3\text{MHz}$
1	0	0	2个周期	330个时钟	126072个时钟	$\geq 6\text{MHz}$
0	1	1	2个周期	660个时钟	252144个时钟	$\geq 12\text{MHz}$
0	1	0	2个周期	1100个时钟	420240个时钟	$\geq 20\text{MHz}$
0	0	1	2个周期	1320个时钟	504288个时钟	$\geq 24\text{MHz}$
0	0	0	2个周期	1760个时钟	672384个时钟	$\geq 30\text{MHz}$

STC单片机存储器结构和地址空间

--数据Flash存储器

■ PCON

- 电源控制寄存器，它位于STC单片机SFR内地址为0x87的位置，如下表所示。
- 当复位时，该寄存器的值设置为00110000。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	SMOD	SMOD0	LVDF	POF	GF1	GF0	PD	IDL

STC单片机存储器结构和地址空间

--数据Flash存储器

■ LVDF

- 低电压检测标志位。当单片机的供电电压低于检测门限电压时，该位设置为1。

注：（1）该位需要由软件清除。

（2）当STC单片机内的低压检测电路发现STC单片机的供电电压Vcc偏低时，不要对数据Flash进行任何操作。在STC_ISP软件中，选中“低压时禁止EEPROM操作”选项前的复选框。

■ POF

- 上电复位标志位。当单片机断电后，上电复位标志设置为1，该位由软件清除。

注：该位可以用于判断复位源。

STC单片机存储器结构和地址空间

--数据Flash存储器

■ GP0和GF1

- 两个通用工作标志位，用户可以任意使用。

■ PD

- 掉电模式控制位。当该位设置为1时，进入掉电模式。在掉电模式时，内部时钟停止振荡。因此，CPU、定时器等部件停止工作，只有外部中断继续工作。当进入掉电模式时，可以由外部中断上升沿或者下降沿触发事件进行唤醒。

■ IDL

- 空闲模式控制位。当该位设置为1时，进入空闲模式。在该模式下，除系统不给8051 CPU提供时钟，以及CPU不执行指令外，其余功能部件仍然正常工作。在该模式时，可以由外部中断、定时器中断、低压检测中断以及A/D转换中断将其从空闲模式唤醒。

数据Flash存储器

--数据Flash (EEPROM) 空间映射

STC15W4K32S4系列单片机数据Flash (EEPROM) 空间容量和地址

型号	EEPROM	扇区数	用IAP字节读时, 数据Flash的起 始扇区首地址	用IAP字节读时, 数据Flash的结 束扇区首地址	用MOVC指令读时, 数据Flash的起 始扇区首地址	用MOVC指令读时, 数据Flash的结束 扇区首地址
STC15W4K16S4	42KB	84	0x0000	0xA7FF	0x4C00	0xF3FF
STC15W4K32S4	26KB	52	0x0000	0x67FF	0x8C00	0xF3FF
STC15W4K40S4	18KB	36	0x0000	0x47FF	0xAC00	0xF3FF
STC15W4K48S4	10KB	20	0x0000	0x27FF	0xCC00	0xF3FF
STC15W4K56S4	2K	4	0x0000	0x07FF	0xEC00	0xF3FF

以下系列特殊，用户可以在用户程序区直接修改用户程序，所有注：没有专门的数据Flash，但是用户可以将用户程序区的程序Flash当作数据Flash使用，使用时不要擦除字节的有效程序

IAP15W4K61S4	--	122	0x0000	0xF3FF	--	--
IRC15W4K63S4	--	126	0x0000	0xFBFF	--	--

注：对于其它类型的单片机数据Flash的容量和映射关系，请参阅STC提供的单片机数据手册。

STC单片机存储器结构和地址空间

--内部数据RAM存储器

STC15系列的单片机内部集成了RAM存储器，可用于存放程序执行的中间结果和过程数据。在逻辑和物理上，将其分为两个地址空间：

- 内部RAM，其容量为256个字节；
- 内部扩展RAM，其容量为3840个字节。

内部数据RAM存储器

--内部RAM

STC15系列单片机内部RAM空间可以分成三个部分。



内部数据RAM存储器

--内部RAM

低128字节RAM（兼容传统8051）



这部分的存储空间，既可采用直接寻址方式又可采用间接寻址方式。这部分RAM区域也称为通用RAM区域，如左图所示。这个区域分为：

- 工作寄存器组区
- 可位寻址区
- 用户RAM区域和堆栈区

内部数据RAM存储器

--内部RAM

■ 工作寄存器组区域

- 该区域的地址从0x00~0x1F，用32个字节单元，分为四组，每个称为一个寄存器组。每个寄存器组包含8个工作寄存器区，范围是R0~R7，但是它们属于不同的物理空间。

■ 可位寻址区域

- 在地址0x20~0x2F的16个字节的区域中，可实现按位寻址。也就是说，可以对这16个单元中的每一位进行单独的寻址，这个区域一共有128位，所对应的地址范围是0x00~0x7F。

■ 用户RAM区域和堆栈区

- 在地址0x30~0xFF区域（包含了高128字节区域）是用户RAM区域和堆栈区，可以采用直接寻址或者间接寻址的方式访问该区域。

内部数据RAM存储器

--内部RAM

高128字节RAM（Intel在8052中扩展了高128字节RAM）

- 这部分区域虽然和SFR区域的地址范围重合，都在0x80~0xFF的区域。但是，它们在物理上是相互独立的，通过不同的寻址方式来区分它们。对于高128字节RAM区域来说，只能采用间接寻址方式。

特殊功能寄存器

- 对于SFR来说，只能采用直接寻址的方式。

内部数据RAM存储器

--内部扩展RAM

在STC15W4K32S4系列的单片机中，除了集成256字节的内部RAM外，还集成了3840字节的扩展RAM区，其地址范围是0x0000~0x0EFF。

■ 在STC15W4K32S4系列单片机中，访问内部RAM的方法如下

□ 使用汇编语言，通过MOVX指令访问内部扩展RAM区域，访问的命令为：

MOVX @DPTR 或 MOVX @Ri

□ 使用C语言，通过使用xdata声明存储类型来访问内部扩展RAM区域。

内部数据RAM存储器

--内部RAM

在STC该系列单片机中，由SFR内地址为0x8E的辅助寄存器AUXR控制，如下表所示。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	T0X12	T1x12	UAR_M0x6	T2R	T2_C/T	T2x12	EXTRAM	S1ST2

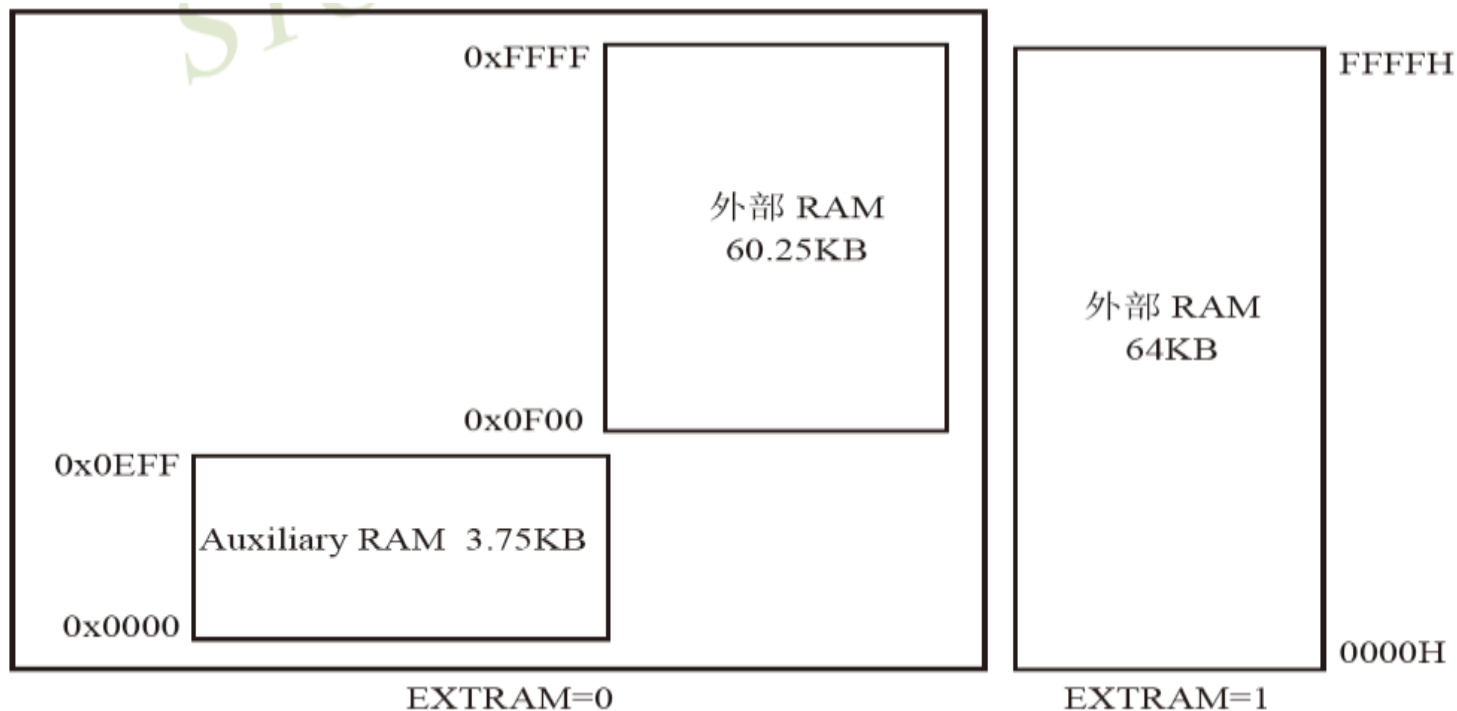
- 当复位时，该寄存器的值为0x01。
- EXTRAM为0时，可以存取内部扩展的EXT_RAM。在STC15W4K32S4系列单片机中，通过指令：MOVX @DPTR，访问0x0000~0x0EFF单元（3840字节）。
- 当访问地址超过0x0F00时，总是访问外部数据存储器。

注：MOVX @Ri 只能访问0x00~0xFF单元。

内部数据RAM存储器

--内部RAM

- EXTRAM为1时，禁止访问内部扩展RAM，此时指令：
MOVX @DPTR/MOVX @Ri 的使用同普通8052单片机。



外部数据存储器

--外部数据存储器访问控制

STC15系列40引脚以上的单片机具有扩展64KB外部数据存储器 and I/O口的能力。

- 当访问STC单片机外扩的数据存储器时，WR和RD信号有效。
- 在STC15系列单片机中，增加了一个控制外部数据存储器的数据总线速度的特殊功能寄存器BUS_SPEED。该寄存器在SFR位置为0xA1的位置。当复位时，该寄存器设置为xxxxx10。

外部数据存储器

--外部数据存储器访问控制

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	---	---	---	---	---	---	EXRTS[1:0]	

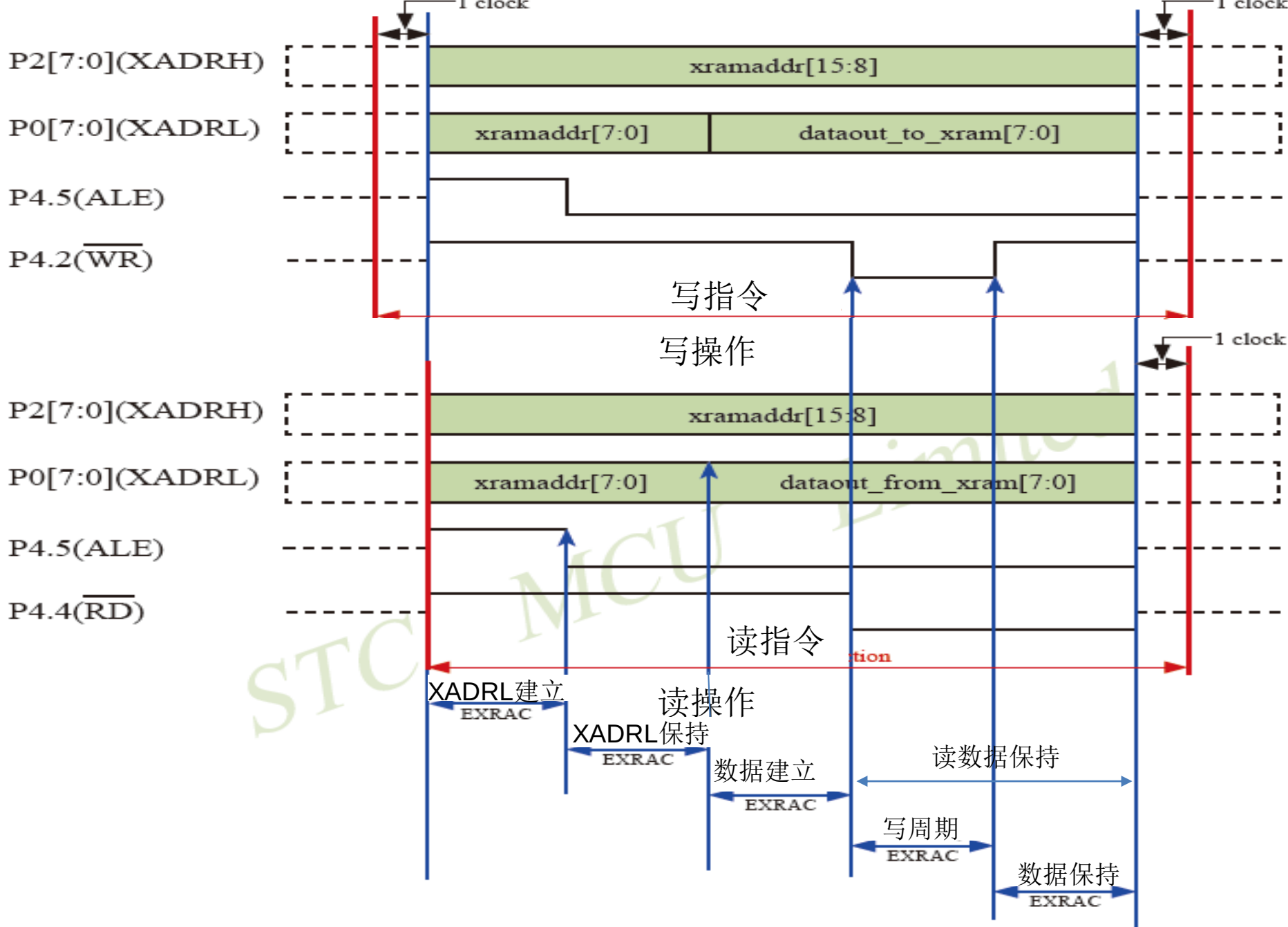
EXRTS[1:0]	建立/保持/读和写时钟周期个数
00	1
01	2
10	4
11	8

外部数据存储器

--外部数据存储器访问控制

为了方便读者理解外部数据存储器的访问原理，给出了写外部数据存储器 and 读外部数据存储器的时序。

- STC单片机的低8位地址线和数据线引脚是复用在P0[7:0]端口上的，在XADL建立和保持周期内，P0[7:0]端口上产生出所要访问外部数据存储器的低8位地址。
- P0[7:0]产生的低8位地址和P2[7:0]产生的高8位地址，拼成一个16位的地址。
- 其可访问的外部数据存储器的地址范围0x0000~0xFFFF，即64KB的范围。



外部数据存储器

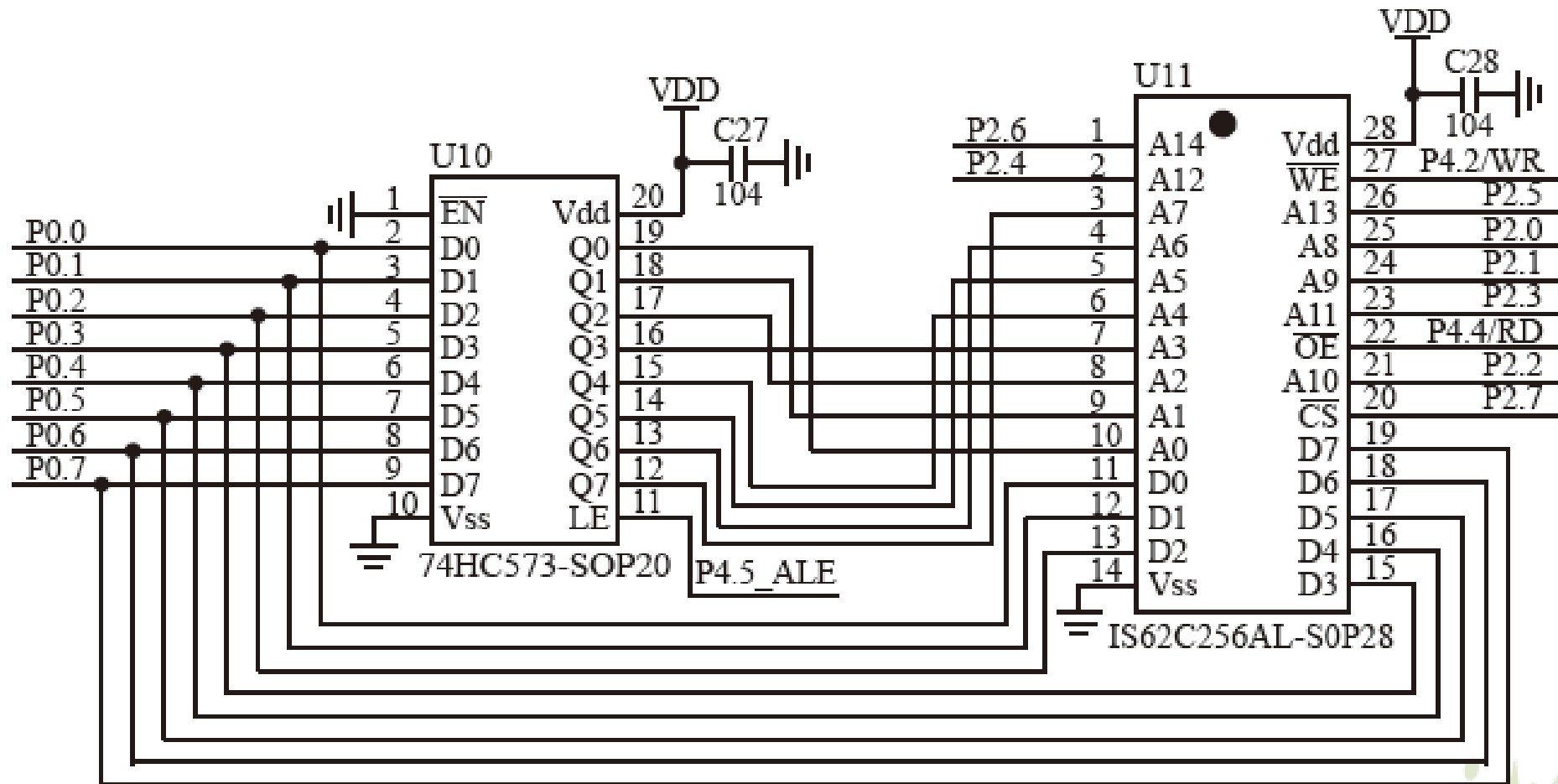
--外部数据存储器访问控制

单片机和外部数据存储器的硬件电路

- 由于单片机上的低8位地址和8位数据复用在P0端口上，因此需要将复用的低8位地址和8位数据分离。在实际应用中，通过使用74HC573器件将地址和数据进行分离。

外部数据存储器

--外部数据存储器访问控制



外部数据存储器

--外部数据存储器访问控制

- 74HC573是8位带3态输出的锁存器，其功能如下表所示。

输入			输出
EN	LE	D	
0	1	1	1
0	1	0	0
0	0	X	Q_0
1	X	X	Z

外部数据存储器

--外部数据存储器访问控制

- 在设计中，74HC573器件的LE引脚与STC单片机P4.5/ALE引脚连接。
 - 当ALE为高时，P0[7:0]端口给出的是用于访问外部数据存储器的低8位地址。通过这个器件，就产生出可以连接到外部型号为IS62C256的SRAM存储器的低8位地址线；而STC单片机的P2[7:0]端口直接产生用于访问外部数据存储器的高8位地址。
 - 在非地址周期时，P0[7:0]端口和需要访问的外部数据存储器的8位数据线直接进行连接。此外，P2[7]引脚作为访问外部存储器的片选信号。

STC单片机中断系统原理及功能

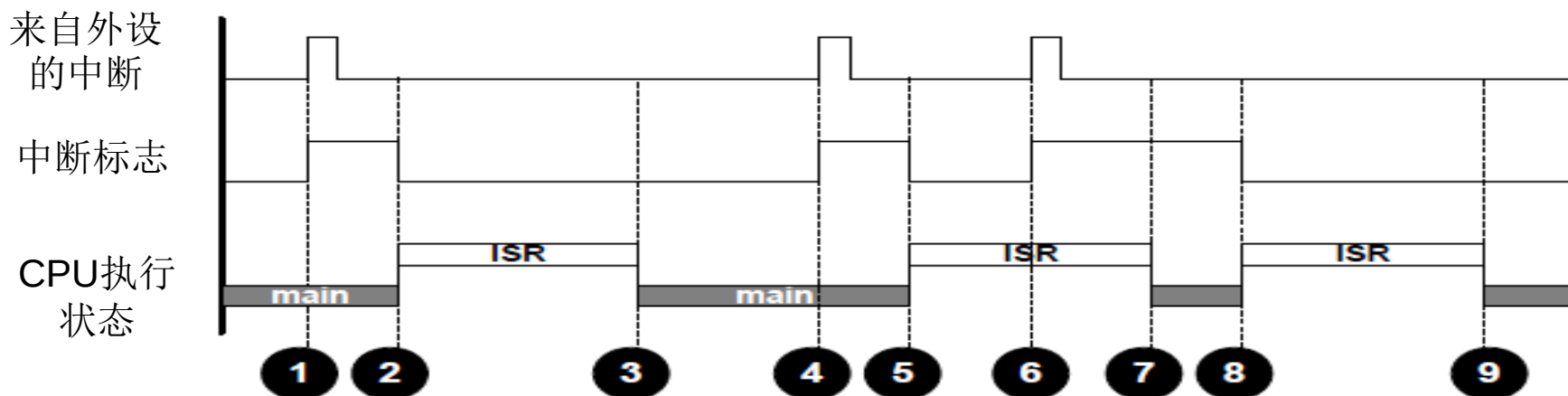
--中断原理

STC单片机中的中断系统是为了8051 CPU具有实时处理外界紧急事件能力而设置的一种机制。

- 当8051 CPU正在处理某个事件的时候，外界发生了紧急事件，这个紧急事件通过STC单片机的外部引脚送给了8051 CPU，8051 CPU就需要做出判断是不是需要立即处理这个紧急事件。
- 如果CPU允许立即处理这个事件，则暂时停止继续执行当前的程序代码，而跳转到中断服务程序。当处理完紧急事件后，再继续处理前面所打断的正常执行的程序代码。这个过程称为中断。

STC单片机中断系统原理及功能

--中断原理



1. 外设在中断线上产生脉冲。中断控制器设置中断标志，并向CPU产生请求；

STC单片机中断系统原理及功能

--中断原理

2. CPU开始执行中断服务程序（ Interrupt Service Routine , ISR ）。中断控制器在进入ISR入口时，清除中断标志；
3. CPU从中断返回；
4. 外设在中断线上产生脉冲。中断控制器设置中断标志，并向CPU产生请求；
5. CPU开始执行中断服务程序ISR。中断控制器在进入ISR入口时，清除中断标志；

STC单片机中断系统原理及功能

--中断原理

6.当正在执行ISR时，外设在中断线上产生脉冲。中断控制器设置中断待处理标志；

7.CPU从中断服务程序返回。当设置中断标志时，中断控制器产生一个中断；

8.CPU开始执行ISR。中断控制器在进入ISR入口时，清除中断标志；

9.CPU从中断服务程序返回；

STC单片机中断系统原理及功能

--中断系统结构

STC15W4K32S4系列的单片机中，提供了21个中断源。

- 在这21个中断中，外部中断2、外部中断3、定时器T2中断、外部中断4、串口3中断、串口4中断、定时器3中断、定时器4中断和比较器中断的优先级总是固定为最低优先级（0级）外，其它的中断都具有2个中断优先级，可以实现两级中断嵌套。
- 用户可以通过中断允许寄存器IE的B7比特位EA，以及相应中断的允许位，使能或者禁止相应的中断进入到8051 CPU中。通常把禁止中断进入8051 CPU，称为屏蔽中断（中断屏蔽）。

注意：当定时器/计数器0工作在不可屏蔽中断的16位自动重载模式时，其优先级是所有中断中最高的，而且不受总中断允许位EA控制。

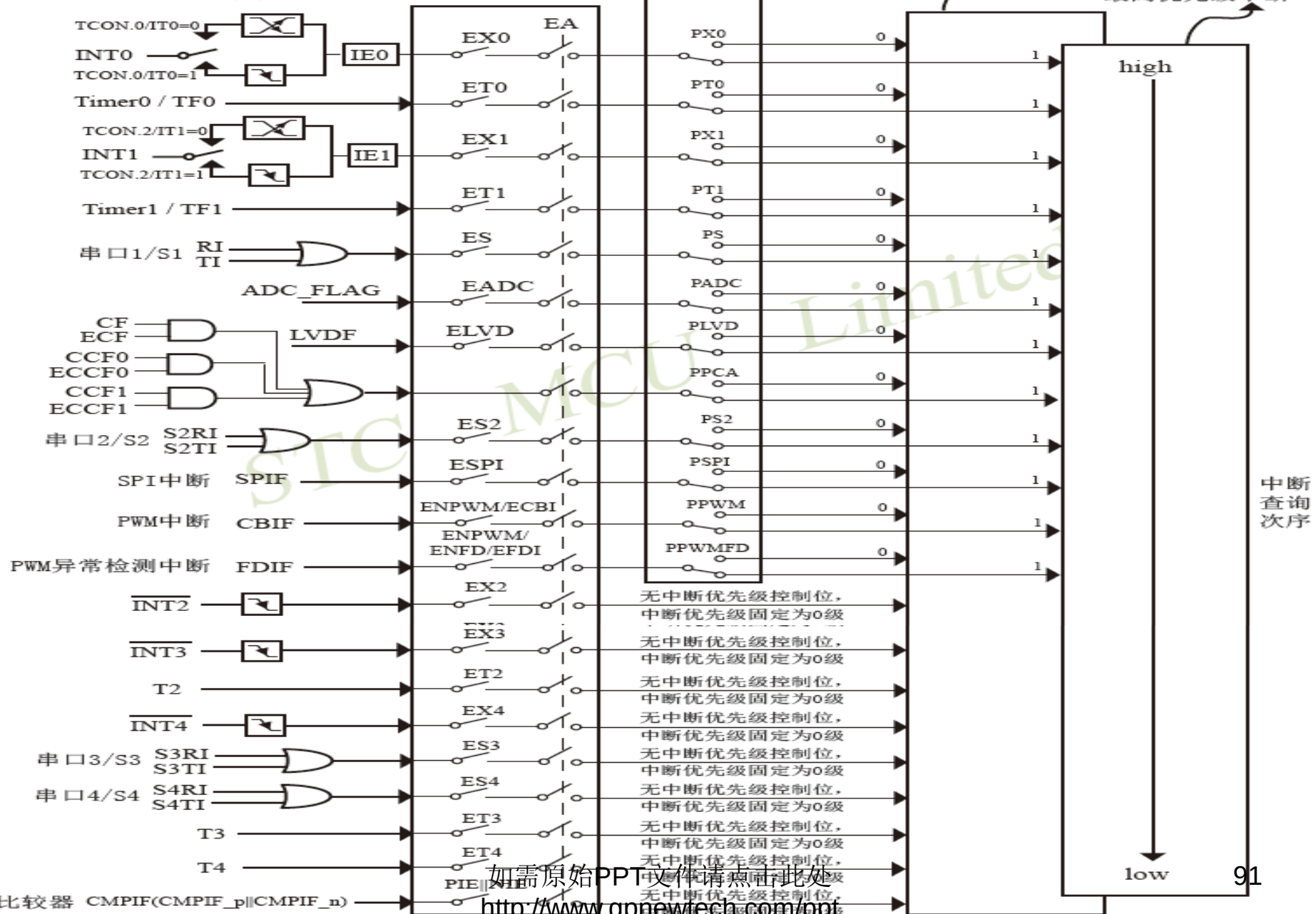
上升沿和下降沿均可中断

中断允许控制寄存器

中断优先级控制寄存器

最低优先级中断

最高优先级中断



如需原始PPT文件请点击此处

<http://www.gpnewtech.com/ppt>

STC单片机中断系统原理及功能

--IE寄存器

STC15W4K32S4系列单片机内的8051 CPU对是否允许和禁止响应所有中断源，以及是否允许每一个中断源，这都是由内部的中断允许寄存器IE控制。

- 该寄存器位于SFR地址为0xA8的位置。当复位时，将该寄存器设置为0x00。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	EA	ELVD	EADC	ES	ET1	EX1	ET0	EX0

注：该寄存器可以位寻址。

STC单片机中断系统原理及功能

--IE寄存器

- **EA**：8051CPU的全局中断使能控制位。当该位为1时，表示CPU可以响应各种不同类型的紧急事件（中断）；当该位为0时，表示CPU禁止响应任何类型的紧急事件（中断）。
- **ELVD**：低电压检测中断允许位。当该位为1时，表示允许低压检测产生中断事件；当该位为0时，表示禁止低电压检测产生中断事件。
- **EADC**：ADC转换中断允许位。当该位为1时，表示允许ADC转换产生中断事件；当该位为0时，表示禁止ADC转换产生中断事件。
- **ES**：串行接口1中断允许位。当该位为1时，表示允许串行接口1产生中断事件；当该位为0时，表示禁止串行接口1产生中断事件。

STC单片机中断系统原理及功能

--IE寄存器

- **ET1**：定时/计数器T1的溢出中断允许位。当该位为1时，表示T1溢出产生中断事件；当该位为0时，表示禁止T1溢出产生中断事件。
- **EX1**：外部中断1中断允许位。当该位为1时，表示允许外部中断1产生中断事件；当该位为0时，表示禁止外部中断1产生中断事件。
- **ET0**：定时/计数器T0的溢出中断允许位。当该位为1时，表示允许T0溢出产生中断事件；当该位为0时，表示禁止T0溢出产生中断事件。
- **EX0**：外部中断0中断允许位。当该位为1时，表示允许外部中断0产生中断事件；当该位为0时，表示禁止外部中断0产生中断事件。

STC单片机中断系统原理及功能

--IE2寄存器

该寄存器用于使能和禁止其它紧急事件

- 该寄存器位于SDR地址为0xAF的位置。当复位时，该寄存器的内容设置为x0000000。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	—	ET4	ET3	ES4	ES3	ET2	ESP1	ES2

■ ET4

- 定时器4中断允许位。当该位为1时，表示允许定时器4产生中断事件；当该位为0时，表示禁止定时器4产生中断事件。

STC单片机中断系统原理及功能

--IE2寄存器

■ ET3

- 定时器3中断允许位。当该位为1时，表示允许定时器3产生中断事件；当该位为0时，表示禁止定时器3产生中断事件。

■ ES4

- 串行接口4中断允许位。当该位为1时，表示允许串行接口4产生中断事件；当该位为0时，表示禁止串行接口4产生中断事件。

■ ES3

- 串行接口3中断允许位。当该位为1时，表示允许串行接口3产生中断事件；当该位为0时，表示禁止串行接口3产生中断事件。

STC单片机中断系统原理及功能

--IE2寄存器

■ ET2

- 定时器2中断允许位。当该位为1时，表示允许定时器2产生中断事件；当该位为0时，表示禁止定时器2产生中断事件。

■ ESPI

- SPI接口中断允许位。当该位为1时，表示允许SPI接口产生中断事件；当该位为0时，表示禁止SPI接口产生中断事件。

■ ES2

- 串行接口2中断允许位。当该位为1时，表示允许串行接口2产生中断事件；当该位为0时，表示禁止串行接口2产生中断事件。

STC单片机中断系统原理及功能

-- INT_CLKO寄存器

外部中断允许和时钟输出寄存器。

- 该寄存器是STC15系列单片机新增加的寄存器，该寄存器位于SFR地址为0x8F的位置。在复位时，该寄存器的内容设置为x0000000。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	--	EX4	EX3	EX2	MCKO_S2	T2CLK0	T1CLK0	T0CLK0

注：在此只介绍和中断有关的比特位。

STC单片机中断系统原理及功能

-- INT_CLKO寄存器

- EX4 :外部中断4中断允许位。当该位为1时，表示允许外部中断4产生中断事件；当该位为0时，表示禁止外部中断4产生中断事件。

注：外部中断4只能通过下降沿进行触发。

- EX3:外部中断3中断允许位。当该位为1时，表示允许外部中断3产生中断事件；当该位为0时，表示禁止外部中断3产生中断事件。

注：外部中断3只能通过下降沿进行触发。

- EX2:外部中断2中断允许位。当该位为1时，表示允许外部中断2产生中断事件；当该位为0时，表示禁止外部中断2产生中断事件。

注：外部中断2只能通过下降沿进行触发。

STC单片机中断系统原理及功能

-- TCON寄存器

TCON寄存器称为定时器/计数器控制寄存器

- 该寄存器在SFR地址为0x88的位置。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	TFI	TR1	TF0	TR0	IE1	IT1	IE0	IT0

注：在此只介绍和该设计有关的比特位。

STC单片机中断系统原理及功能

-- TCON寄存器

■ IE1 :

外部中断1 (INT1/P3.3) 中断请求标志。当该位为1，外部中断1向CPU请求中断。当CPU响应该中断时，由硬件清除该位。

■ IT1 :

外部中断1中断源类型选择位。当该位为0时，表示INT1/P3.3引脚上的上升沿或者下降沿信号均可以触发外部中断1；当该位为1时，表示外部中断1为下降沿触发方式。

STC单片机中断系统原理及功能

-- TCON寄存器

■ IE0 :

外部中断1 (INT0/P3.2) 中断请求标志。当该位为1, 外部中断向CPU请求中断。当CPU响应该中断时, 由硬件清除该位。

■ IT0 :

外部中断0中断源类型选择位。当该位为0时, 表示INT0/P3.2 引脚上的上升沿或者下降沿信号均可以触发外部中断0; 当该位为1时, 表示外部中断0为下降沿触发方式。

STC单片机中断系统原理及功能

对于上面的寄存器来说：

- 对于IE寄存器的设置操作，可以通过下面的位操作指令：

SETB BIT

CLR BIT

或通过下面的字节操作指令实现：

MOV IE, #DATA

ANL IE, #DATA

ORL IE, #DATA

MOV IE, A

- 对于IE2和INT_CLKO的设置操作，只能通过字节操作指令完成。

STC单片机中断系统原理及功能

--中断优先级处理

如果一个8051 CPU确认需要响应中断INTB，而CPU此时正在执行另一个中断INTA，这里有三种可能性用于处理这种情况：

■ 如果INTA的优先级比INTB低，则：

- 在当前INTA正在执行的指令上停下来，即：暂时停止运行INTA；
- 将INTA的现场入栈，即：保护现场，CPU开始转向执行INTB；
- 当执行完INTB后，CPU跳转到刚才中断执行INTA指令的地方，继续执行INTA；

STC单片机中断系统原理及功能

--中断优先级处理

■ 如果INTA的优先级比INTB高，则：

- INTB一直等待，直到执行完INTA；
- 一旦执行完INTA后，立即开始执行INTB；

■ 如果INTA和INTB有相同的优先级，则：

- 如果正在执行INTA，则INTB等待执行完INTA。当执行完INTA后，开始执行INTB；
- 如果正在执行INTB，则INTA等待执行完INTB。当执行完INTB后，开始执行INTA。

STC单片机中断系统原理及功能

--中断优先级处理

同时产生中断

- 如果INTA优先级低于INTB，则INTB获得仲裁权，开始执行；
- 如果INTA优先级高于INTB，则INTA获得仲裁权，开始执行；
- 如果INTA和INTB有相同的优先级，则具有低索引值的中断获得仲裁权，开始执行。

STC单片机中断系统原理及功能

--中断优先级控制器

在STC15系列单片机中，通过设置特殊功能寄存器IP1和IP2中的相应位，可以将部分中断设为带有2个中断优先级，除外部中断2、外部中断3和外部中断4以外，可以将其他所有中断请求源设置为2个优先级。

STC单片机中断系统原理及功能

--中断优先级寄存器IP1

该寄存器用于设置部分中断源的优先级

- 该寄存器位于SFR地址为0xB8的位置。当复位时，该寄存器的内容设置为0x00。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	PPCA	PLVD	PADC	PS	PT1	PX1	PT0	PX0

注：该寄存器可位寻址

STC单片机中断系统原理及功能

--中断优先级寄存器IP1

■ PPCA :

PCA中断优先级控制位。当该位为0时，PCA中断为最低优先级（优先级0）；当该位为1时，PCA中断为最高优先级（优先级1）。

■ PLVD :

PLVD中断优先级控制位。当该位为0时，低电压检测中断为最低优先级（优先级0）；当该位为1时，低电压检测中断为最高优先级（优先级1）。

■ PADC :

PADC中断优先级控制位。当该位为0时，ADC转换中断为最低优先级（优先级0）；当该位为1时，ADC转换中断为最高优先级（优先级1）。

■ PS :

PS中断优先级控制位。当该位为0时，串口1中断为最低优先级（优先级0）；当该位为1时，串口1中断为最高优先级（优先级1）。

STC单片机中断系统原理及功能

--中断优先级寄存器IP1

■ PT1 :

PT1中断优先级控制位。当该位为0时，定时器1中断为最低优先级（优先级0）；当该位为1时，定时器1中断为最高优先级（优先级1）。

■ PX1 :

PX1中断优先级控制位。当该位为0时，外部中断1为最低优先级（优先级0）；当该位为1时，外部中断1为最高优先级（优先级1）。

■ PT0 :

PT0中断优先级控制位。当该位为0时，定时器0中断为最低优先级（优先级0）；当该位为1时，定时器0中断为最高优先级（优先级1）。

■ PX0 :

PX0中断优先级控制位。当该位为0时，外部中断0为最低优先级（优先级0）；当该位为1时，外部中断0为最高优先级（优先级1）。

STC单片机中断系统原理及功能

--中断优先级寄存器IP2

该寄存器用于设置部分中断源的优先级

- 该寄存器位于SFR地址为0xB5的位置。当复位时，该寄存器的内容设置为xxx00000。

比特	B7	B6	B5	B4	B3	B2	B1	B0
名字	--	--	--	PX4	PPWMFD	PPWM	PSPI	PS2

- PX4：PX4中断优先级控制位。当该位为0时，外部中断4为最低优先级（优先级0）；当该位为1时，外部中断4为最高优先级（优先级1）。

STC单片机中断系统原理及功能

--中断优先级寄存器IP2

- **PPWMFD** : PPWMFD中断优先级控制位。当该位为0时，PWM异常检测中断为最低优先级（优先级0）；当该位为1时，PWM异常检测中断为最高优先级（优先级1）。
- **PPWM** : PPWM中断优先级控制位。当该位为0时，PWM中断为最低优先级（优先级0）；当该位为1时，PWM中断为最高优先级（优先级1）。
- **PSPI** : PSPI中断优先级控制位。当该位为0时，SPI中断为最低优先级（优先级0）；当该位为1时，SPI中断为最高优先级（优先级1）。
- **PS2** : PS2中断优先级控制位。当该位为0时，串口2中断为最低优先级（优先级0）；当该位为1时，串口2中断为最高优先级（优先级1）。

STC单片机中断系统原理及功能

--中断向量表

在计算机中，把一个用于保存处理不同类型事件的程序代码起始地址的存储空间称为中断向量表。

- 实际上，中断向量表就是程序存储器中的一块特定的存储空间而已，只不过这个存储空间的位置已经实现规定好了，用户不可以修改中断向量表所在地址空间的位置。
- 用户可以做的工作是可以修改中断向量表中每个中断向量的内容，也就是为处理每个不同类型事件的程序代码指定起始地址，这就是所说的中断映射，如下图所示。

STC单片机中断系统原理及功能

程序Flash存储器空间

地址0x0000		
地址0x0003	LJMP指令	外部中断0的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理外部中断0产生的中断事件）
	地址高8位	
	地址低8位	
地址0x000B	LJMP指令	定时器0的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理定时器0产生的中断事件）
	地址高8位	
	地址低8位	
		
地址0x0013	LJMP指令	外部中断1的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理外部中断1产生的中断事件）
	地址高8位	
	地址低8位	
		
地址0x001B	LJMP指令	定时器1的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理定时器1产生的中断事件）
	地址高8位	
	地址低8位	
		

STC单片机中断系统原理及功能

地址0x0023	LJMP指令 地址高8位 地址低8位	} 串口1的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理串口1产生的中断事件）
地址0x002B	LJMP指令 地址高8位 地址低8位	
地址0x0033	LJMP指令 地址高8位 地址低8位	
地址0x003B	LJMP指令 地址高8位 地址低8位	
		} PCA的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理可编程计数阵列产生的中断事件）
地址0x0043	LJMP指令 地址高8位 地址低8位	} 串口2的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理串口2产生的中断事件）
地址0x004B	LJMP指令 地址高8位 地址低8位	
		} SPI的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理SPI产生的中断事件）

STC单片机中断系统原理及功能

地址0x0053	LJMP指令 地址高8位 地址低8位	外部中断2的中断向量。其内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理外部中断2产生的中断事件）
地址0x005B	LJMP指令 地址高8位 地址低8位	外部中断3的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理外部中断3产生的中断事件）
地址0x0063	LJMP指令 地址高8位 地址低8位	定时器2的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理定时器2产生的中断事件）
地址0x0083	LJMP指令 地址高8位 地址低8位	外部中断4的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理外部中断4产生的中断事件）
地址0x008B	LJMP指令 地址高8位 地址低8位	串口3的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理串口3产生的中断事件）

STC单片机中断系统原理及功能

地址0x0093

LJMP指令

地址高8位

地址低8位

.....

串口4的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理串口4产生的中断事件）

地址0x009B

LJMP指令

地址高8位

地址低8位

.....

定时器3的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理定时器3产生的中断事件）

地址0x00A3

LJMP指令

地址高8位

地址低8位

.....

定时器4的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理定时器4产生的中断事件）

地址0x00AB

LJMP指令

地址高8位

地址低8位

.....

比较器的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理比较器产生的中断事件）

地址0x00B3

LJMP指令

地址高8位

地址低8位

.....

PWM的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于处理PWM产生的中断事件）

地址0x00BB

LJMP指令

地址高8位

地址低8位

.....

PWM异常的中断向量。内容为指向中断服务程序入口地址的跳转指令（该ISR用于PWM异常产生的中断事件）