

# 单片机



1

单片机的基本概念

2

单片机的开发

3

STC系列单片机

4

STC89C52单片机



# 什么是单片机

- 单片机一词最初源于“Single Chip Microcomputer”，简称“SCM”。单片机也叫做“微控制器”或者“嵌入式微控制器”。
- 它不是完成某一个逻辑功能的芯片（芯片也称为集成电路块），而是把一个微型计算机系统集成到一个芯片上。概括的讲：一块芯片就成了一台计算机。
- 它体积小、质量轻、价格便宜，为学习、应用和开发提供了便利条件。同时，学习使用单片机是了解计算机原理与结构的最佳选择。



单片机是具有CPU运算处理能力，集成了多种硬件功能及IO接口的一种集成电路芯片。

内存

南桥 (ICH)

硬盘

端口

430mm\*410mm\*190mm

4mm\*4mm\*0.75mm

ANALOG PERIPHERALS		DIGITAL I/O	
-----		UART0	Port 0
		Port 1	Port 1
		Port 2	Port 2
		Port 3	Port 3
		Port 4	Port 4
		I2C Only	
		SER/	
64/32 kB ISP FLASH	8051 CPU (48/25 MIPS)	4/2 kB SRAM	
FLEXIBLE INTERRUPTS	DEBUG CIRCUITRY	POR	WDT



# 单片机的发展

年代	事件
1971	Intel公司研制出世界上第一个4位的微处理器Intel 4004，标志着微处理器芯片的诞生
1971	Intel公司推出MCS-4微型计算机系统（包括4004微处理器）
1972	Intel公司研制出第一个8位微处理器Intel 8008
1973	Intel公司研制出8位微处理器8080，标志第二代微处理器的诞生
1975	MITS公司发布第一个通用型Altair 8800，标志世界上第一台微型计算机的诞生



# 单片机的发展

1976	Intel公司研制出MCS-48系列8位单片机，标志着单片机的诞生
1976	Zilog公司开发出Z80微处理器，用于微型计算机和工业自动控制设备
1981	在MCS-48单片机的基础上，Intel推出了MCS-51系列的单片机
1985	Acorn公司的Roger Wilson和Steve Furber设计了32位的RISC指令集计算机，简称ARM
1990	Acorn公司改组为现在的ARM计算机公司





航空航天  
智能仪器仪表



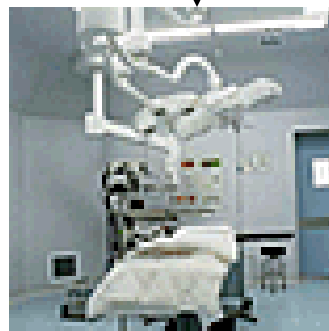
工业控制



计算机网络和通  
讯技术



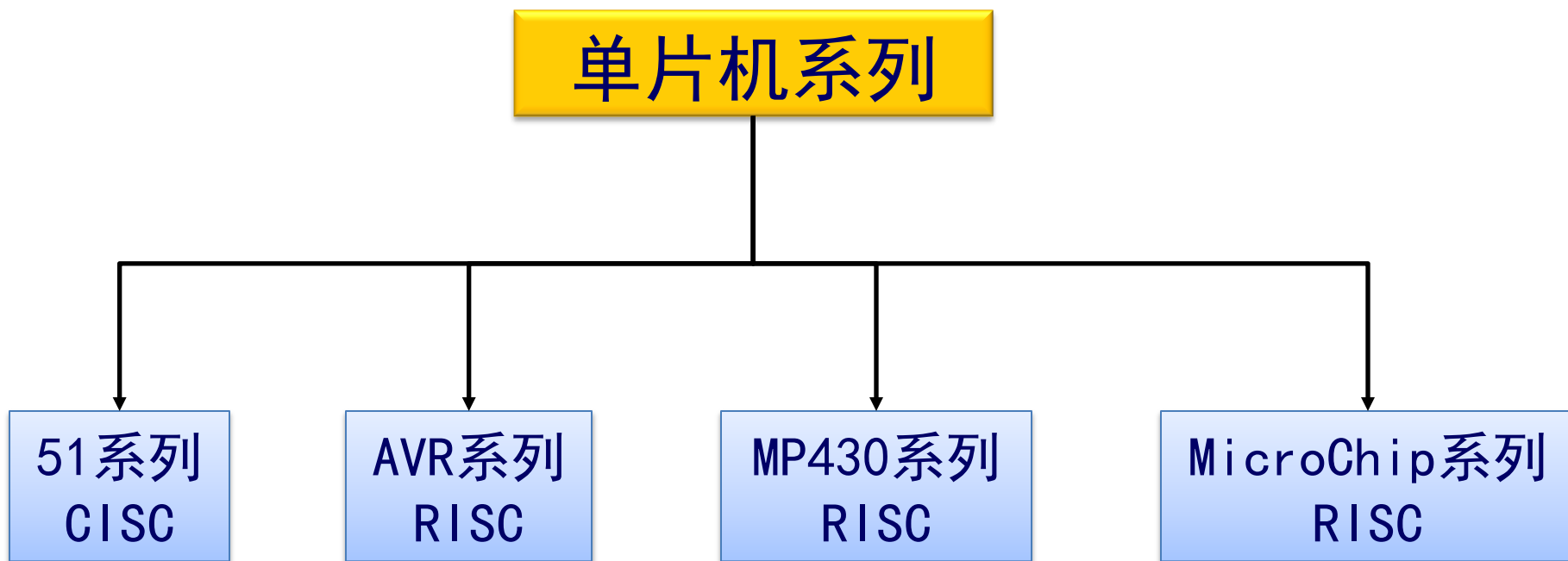
日常生活及家电



医疗设备



电动玩具



复杂指令系统——Complex Instruction Set Computing

精简指令系统——Reduced Instruction Set Computing





# 51系列单片机

- Intel
- ATMEL
- Silicon Lab
- Cypress
- Analog Device
- NXP
- MAXIM



# 51系列单片机

- Texas Instruments
- Freescale
- MicroChip
- STC



# 电平特性

- 数字电路中只有两种电平：高和低
- （本课程中）定义单片机为TTL电平：

	高	低
TTL电平	+5V (VCC)	0V
数字逻辑	1	0



# 51单片机内部结构

外部晶体振荡器与内部振荡电路共同产生单片机运行所需的时钟信号

用于存放用户程序、常数、表格等信息

用于存放单片机运行过程中的运算数据及运算地址等

用来对片内各功能模块进行管理、控制和监控的寄存器 and 反映各功能运行状态的寄存器

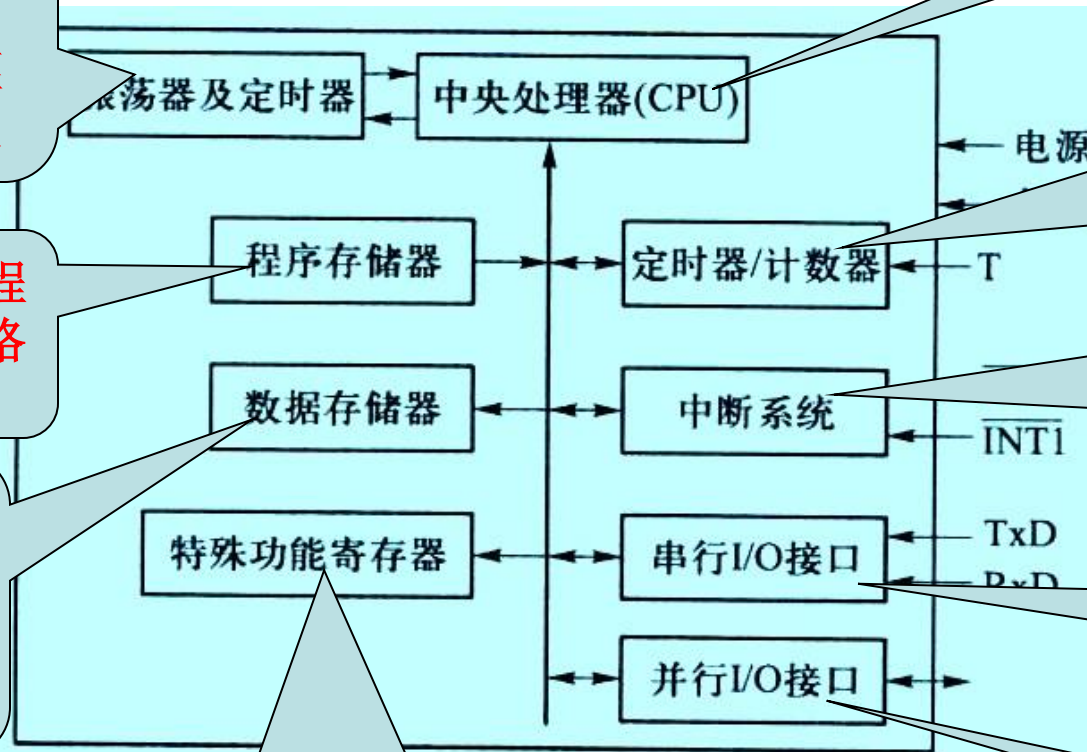
负责运算、控制和调度

2个16位的定时/计数器，实现定时和计数功能

5个中断源、2级中断优先级的中断系统

1个全双工异步串行口

4个8位的并行I/O口

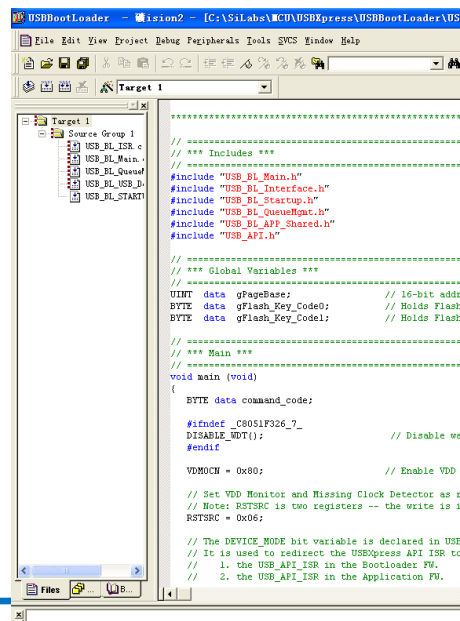


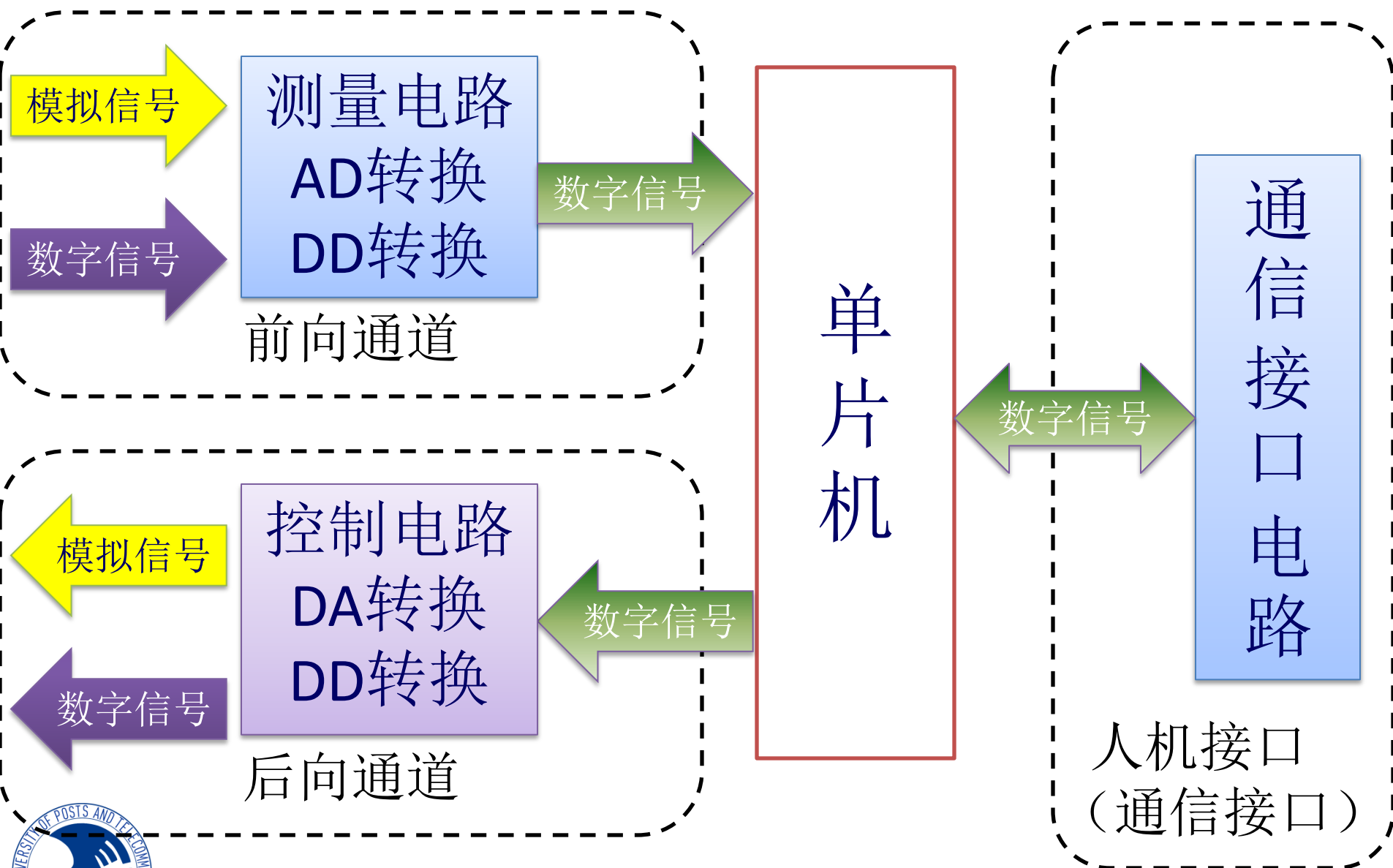
# 单片机应用系统

• 单片机应用系统 = 硬件 + 软件

— 硬件是基础

— 软件是核心





# 利用单片机进行开发的一般过程

- 每一个系统都是在完成一个特定的功能，这些特定功能的实现依靠单片机来指派。单片机知道怎样来进行指派，是由我们编写的程序来控制，从这个过程反过来推导就是单片机开发的一个过程。
- 最简流程：根据硬件设计编写程序，并把程序加载到单片机里。
- 总体来说单片机开发的一般过程是首先进行硬件设计，然后根据硬件和系统的要求在开发环境中编写程序，经多次使用仿真器把程序调试成功后，再通过烧录器把程序写到单片机里。



- 开发环境（也称“编译器”）
  - 由于单片机只认识“0”和“1”，这就需要一个“中间人”来充当翻译，把程序翻译成“0”和“1”的一系列组合（“0”和“1”的一系列组合也称之为目标码或机器码）
- 下载器（或称“烧录器”）
  - 为了把翻译的结果“灌入”（存入）单片机
- 仿真机
  - 为了更加方便的检查我们编写的程序是否符合设计的系统的要求（进行程序调试）
  - 当程序仿真成功以后，再下载到我们设计的系统上，这样一来不仅为我们的程序调试提供了方便，也减少了把一个有误的程序下载到设计的系统上的可能。





# 单片机的编程语言

- 世界上不论是哪个厂商开发出来的单片机，它们都具有下面共同的特点：
  - 采用存储程序的体系结构，即将程序保存在程序存储器中。
  - 本质上都是串行执行程序的，都是靠程序计数器（Program Counter, PC）来控制程序的运行。
  - 工作在二进制状态，也就是通常所说的在PC的控制下，通过取出指令、翻译指令和运行指令，运行二进制组成的机器代码。通过运行机器代码，来控制CPU内各个功能部件的协调运行。



# 单片机的编程语言

- 单片机编译语言：
  - 机器语言
  - 汇编语言
  - C语言



# 单片机的编程语言

## 机器语言

- 输入给CPU，通知其执行运算的称为机器语言。
- 机器语言是由0和1构成的序列，序列中包含操作码和操作数两部分。
- 8051单片机中，ACC累加器和一个常数（立即数）相加的机器语言的格式如下：

0	0	1	0	0	1	0	0
---	---	---	---	---	---	---	---

立即数
-----



# 单片机的编程语言

## 机器语言

0 0

1 0

0 1 0 0

立即数

### ■ 操作码

- 操作码告诉CPU所需要执行的操作。操作码部分包含了操作的类型编码，同时也包含了一部分的操作数内容，指明了参与加法运算的一个数来自ACC累加寄存器中。

### ■ 操作数

- 操作数是执行操作所针对的对象。这些对象包括：立即数（常数），寄存器和存储器等。

### ■ 纯粹意义上的机器语言对程序员太难理解了！



# 单片机的编程语言

## 汇编语言

- 用汇编助记符描述机器指令完整的形式为：

[标号:] 助记符 [操作数] [; 注释]

- 标号用来表示一行指令，
- 助记符表示所要执行的逻辑操作行为，
- 操作数为逻辑操作行为所操作具体对象
- 现在用汇编语言来描述下面的机器指令：

ADD A, #25



# 单片机的编程语言

## ■ 汇编语言的特点：

- 程序员必须清楚CPU的指令集，寄存器单元和存储器映射等硬件规则。
- 执行效率高，但编程效率低。

## ■ 汇编语言的重要性：

- 对理解CPU内部的结构和运行的原理非常重要；
- 很多与CPU打交道的软件驱动程序，尤其是程序的引导代码必须用汇编语言开发。
- 在一些对程序执行时间比较苛刻的场合也需要使用汇编语言进行开发。



# 单片机的编程语言

## C语言

- C语言不能直接在CPU上，它必须通过编译器，生成可执行文件，才能在CPU上运行。

C语言、汇编语言和机器语言之间的对应关系

C语言描述	汇编语言描述	机器语言	功能
F=C+D	MOV A, 0x08	E508	将数据空间地址为0x08的内容送给累加器A
	ADD A, 0x09	2509	将数据空间地址为0x09的内容和累加器A相加后送给累加器A
	MOV 0x0A, A	F50A	将累加器A的内容送到地址为0x0A的数据空间

# STC系列单片机

- STC micro ( 宏晶科技公司 )
  - 1999年成立，创始人姚永平
  - 现已成长为全球最大的8051单片机设计公司，提供专用MCU设计服务。
- 
- 低功耗, 超低价, 高可靠
  - 无法解密, 并且软硬件完全兼容全球各厂家均已被解密的51系列单片机，可直接取代





# STC系列单片机

- STC系列单片机主要有STC89C51xx、STC90C51xx、STC11/10xx、STC12xx、STC15xx等系列，除STC15xx系列外，其他系列的引脚能够相互兼容，各系列新增的功能不完全相同。



# STC单片机ISP和IAP

**当设计者在计算机上完成单片机的程序开发后，就需要将程序固化到单片机内部的程序存储器中。**

- **将本地固化程序的方式称为在系统编程（ In System Programming , ISP ）**
- **另一种固化程序的方式称为在应用编程（ In Application Programming , IAP ）。**



# STC单片机ISP和IAP

## ISP

通过单片机专用的串行编程接口和STC提供的串口固化程序软件，对单片机内部的Flash存储器进行编程。一般来说，实现ISP只需要很少外部电路。

## IAP

IAP技术是从结构上将Flash存储器映射为两个存储空间。当运行一个存储体空间的程序时，可对另一个存储空间重新编程。然后，将控制从一个存储空间转向另一个存储空间。IAP的实现更加灵活。

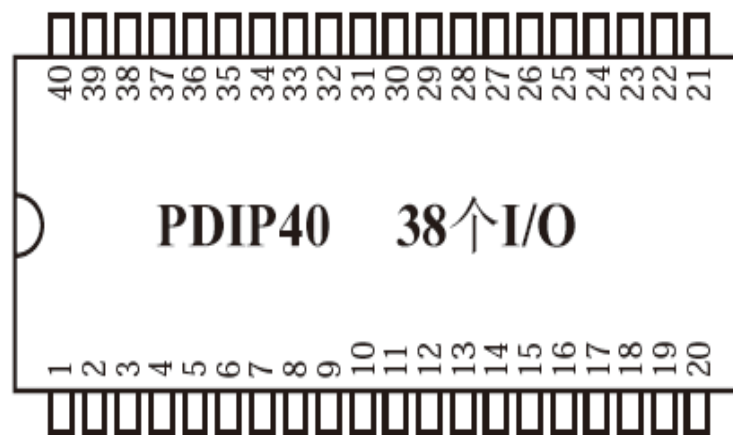
**注：支持ISP方式的单片机，不一定支持IAP方式；但是，支持IAP方式的单片机，一定支持ISP方式。**



# STC单片机封装类型

## 双列直插式封装

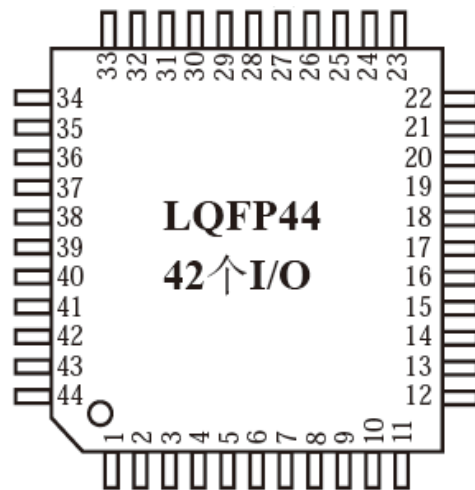
双列直插式封装（Dual Inline-pin Package，DIP），也称为双列直插式封装技术。早期的集成电路芯片大多采用双列直插形式封装，其引脚数一般不超过100。



# STC单片机封装类型

## 薄型四方扁平式封装

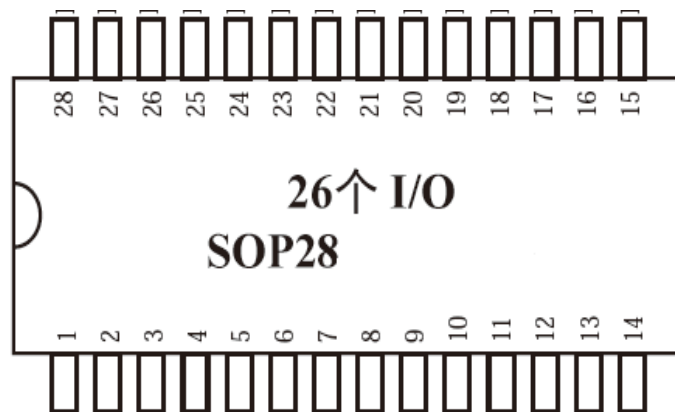
采用薄型四方扁平式（Low-profile Quad Flat Package，LQFP）封装的集成电路芯片引脚之间距离很小，引脚很细。



# STC单片机封装类型

## 小外形封装

小外形封装 ( Small Out-Line Package , SOP ) 是一种很常见的元器件形式。表面贴装型封装之一，引脚从封装两侧引出呈海鸥翼状(L字形)。



# STC单片机封装类型

## 薄的缩小型小外形封装

薄的缩小型小外形 ( Thin Shrink Small Outline Package , TSSOP ) 封装 , 比SOP封装薄 , 引脚更密 , 封装尺寸更小。



# STC单片机封装类型

## 方形扁平无引脚封装

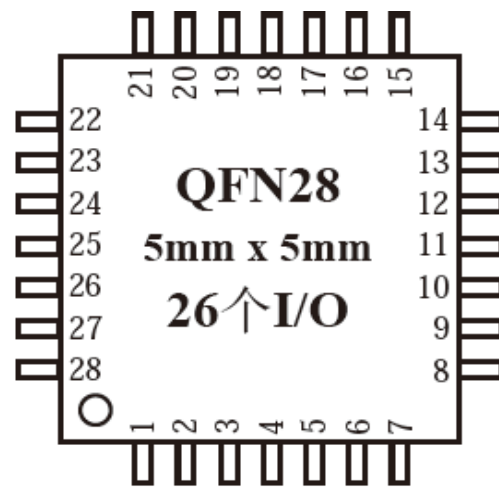
方形扁平无引脚（Quad Flat No-lead，QFN）封装，表面贴装型封装之一。现在多称为LCC。



(a) 正面



(b) 背面





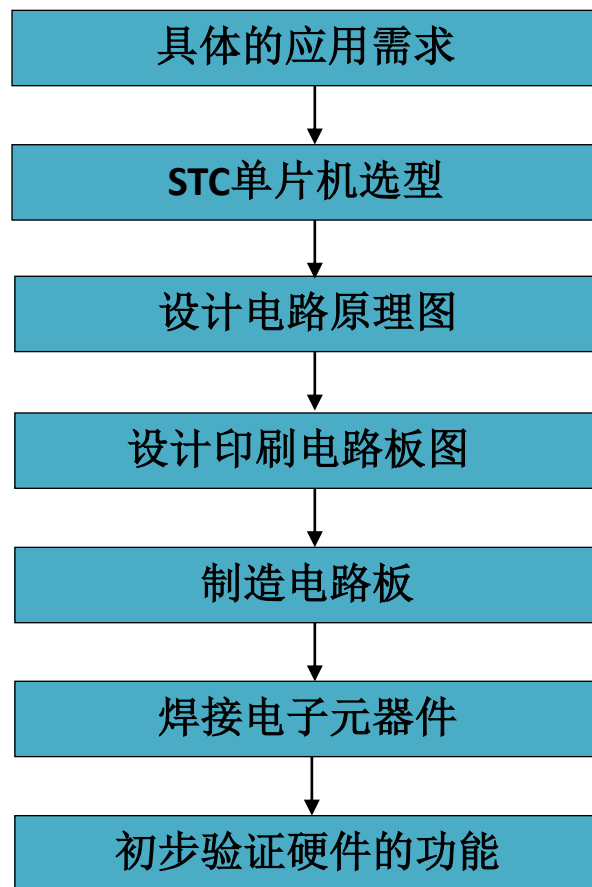
# STC单片机开发流程

**STC**单片机开发流程，主要包括硬件设计流程和软件设计流程。

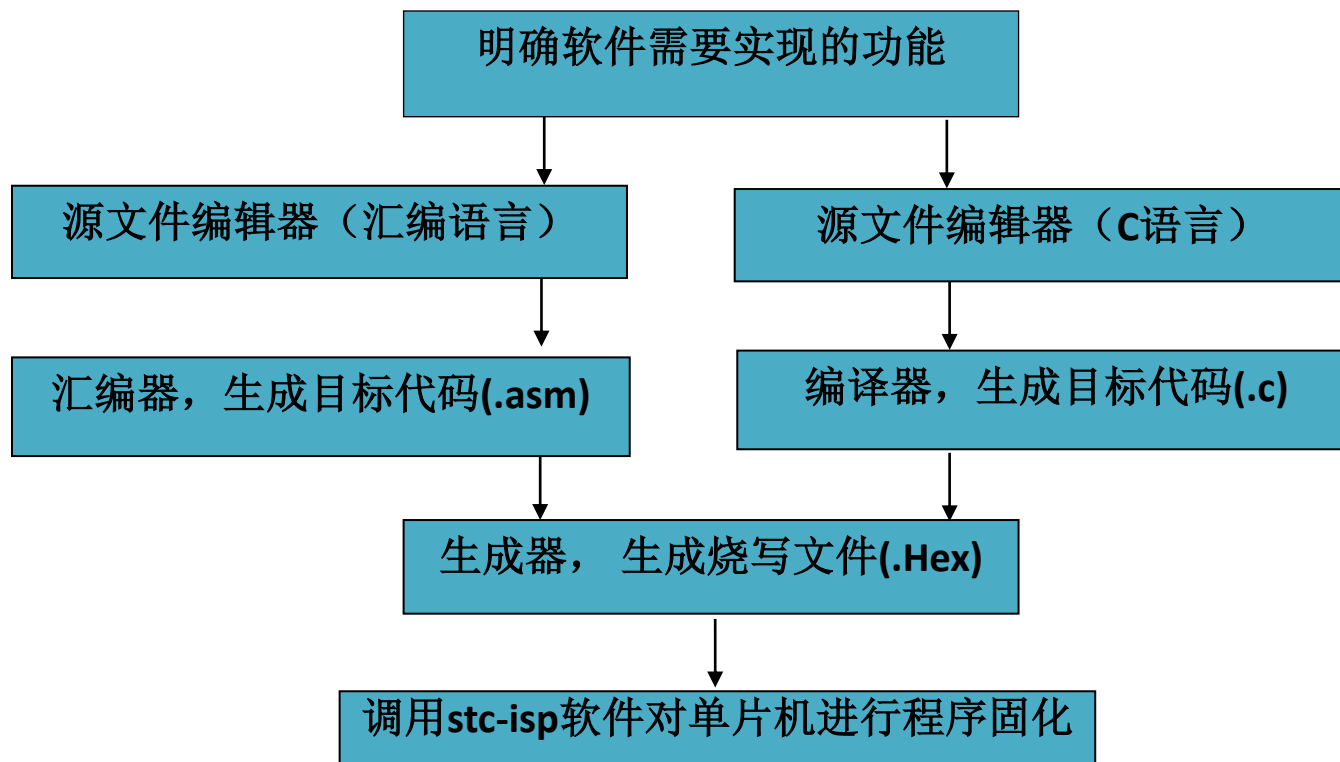
- 当硬件和软件设计流程结束后，通过**USB**-串口下载电缆，以及**STC**提供的**stc-isp**软件将**HEX**文件下载到设计者所选择的单片机程序存储器中。



# STC单片机开发流程（硬件）

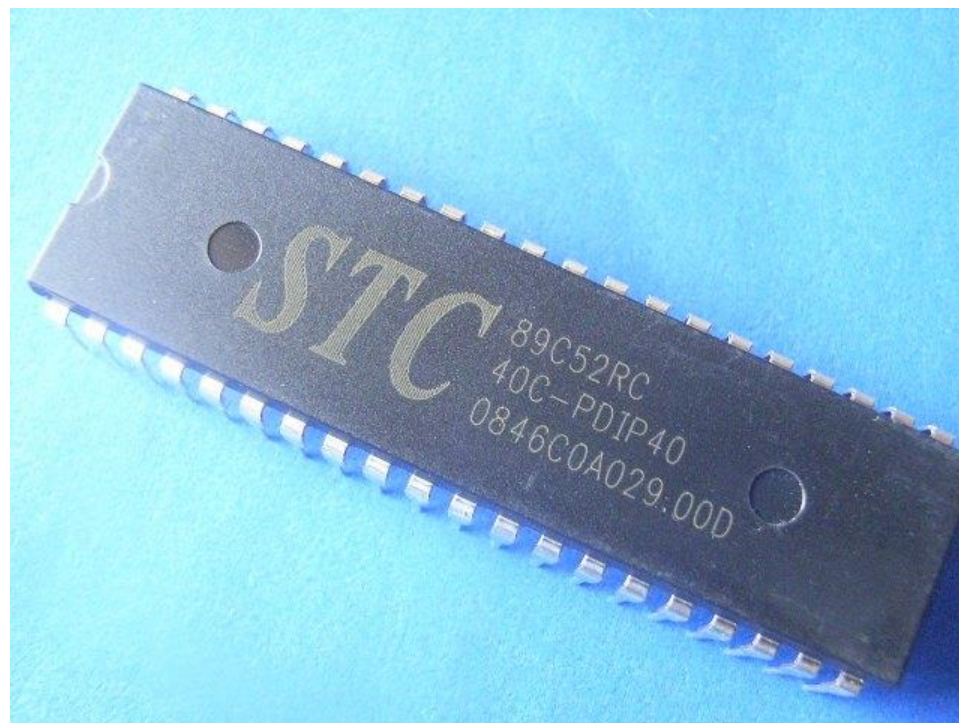


# STC单片机开发流程（软件）



# STC89C52单片机

## — 智能循迹小车

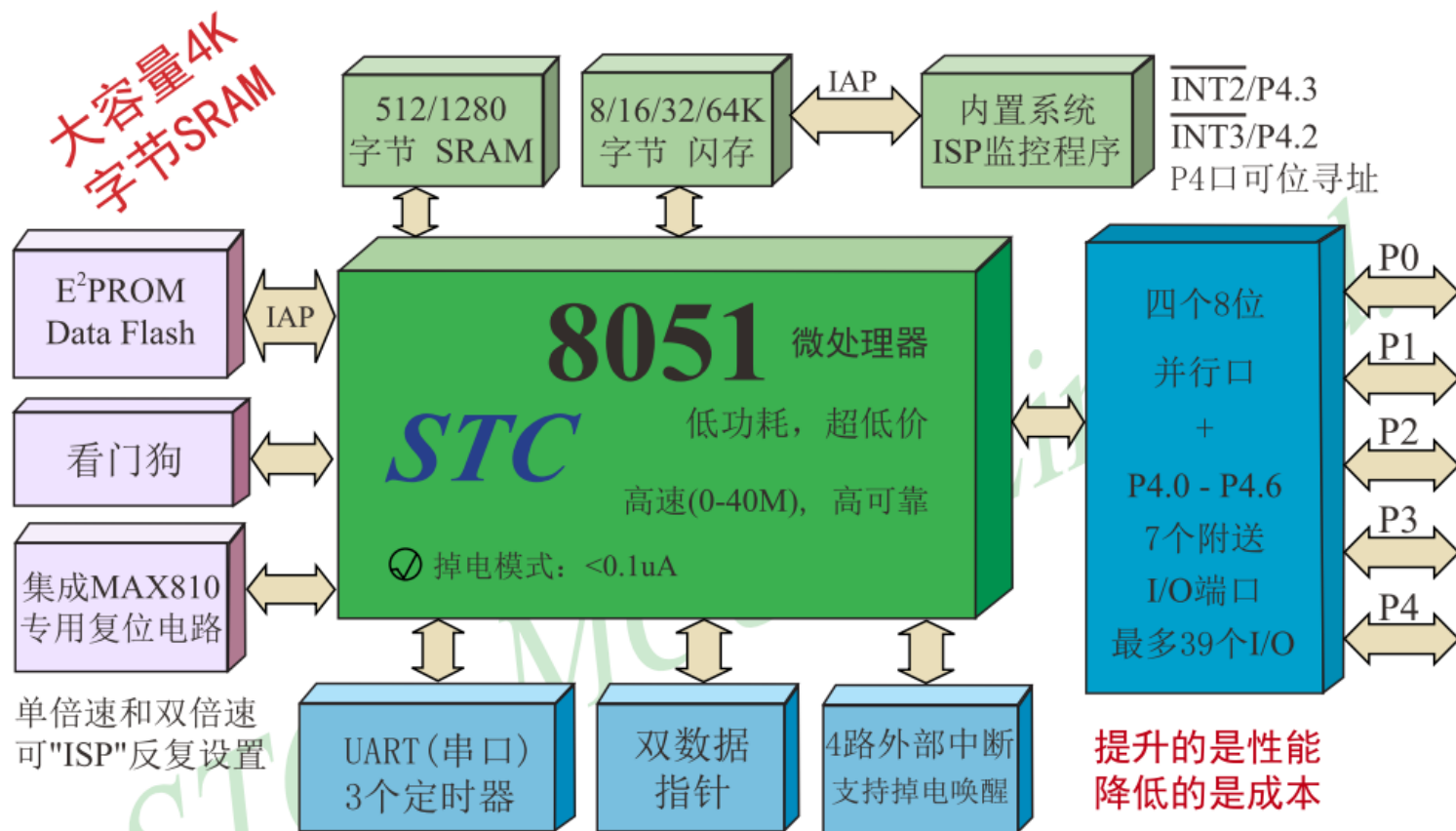


# STC89C52单片机

- 内部结构及特点
- 外部引脚及功能
- 时钟与复位



# STC89C52单片机内部结构及特点



# STC89C52单片机内部结构及特点

## 功能部件和特性：

1. 增强型6时钟/机器周期和12时钟/机器周期任意设置。
2. 指令代码完全兼容传统8051
3. 工作电压：5.5 ~ 3.3V（5V单片机）。
4. 工作频率：0 ~ 40MHz，相当于普通8051单片机的0 ~ 80MHz，实际工作频率可达48MHz。
5. 用户应用程序空间：8KB片内Flash程序存储器，擦写次数10万次以上。
6. 片上集成512B RAM数据存储器。



# STC89C52单片机内部结构及特点

7. 通用I/O口（35/39个），复位后为：P1、P2、P3、P4是准双向口/弱上拉（与普通MCS-51传统I/O口功能一样）；P0口是开漏输出口，作为总线扩展时用，不用加上拉电阻；P0口作为I/O口用时，需加上拉电阻。
8. ISP在系统可编程/IAP在应用可编程，无需专用编程器/仿真器，可通过串口（RxD/P3.0，TxD/P3.1）直接下载用户程序，8KB程序3s即可完成一片。
9. 芯片内置EEPROM功能。
10. 硬件看门狗（WDT）。
11. 内部集成MAX810专用复位电路（HD版本和90C版本才有），外部晶体20M以下时，可不需要外部复位电路。



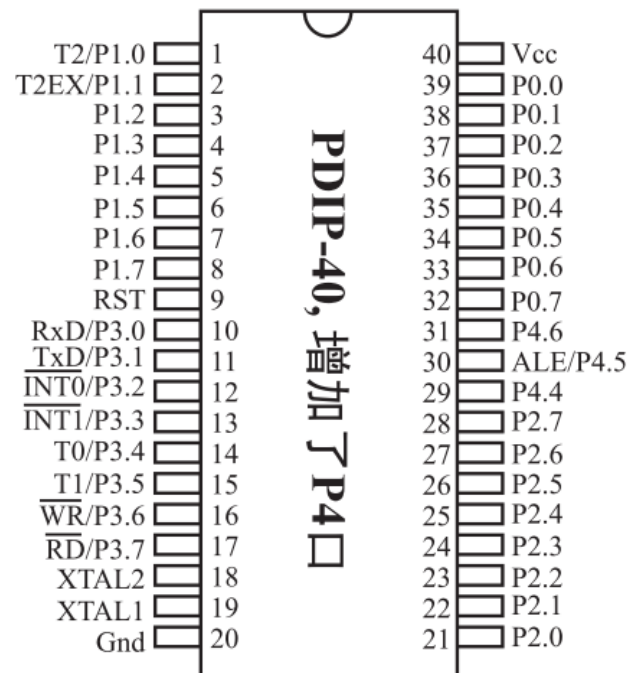
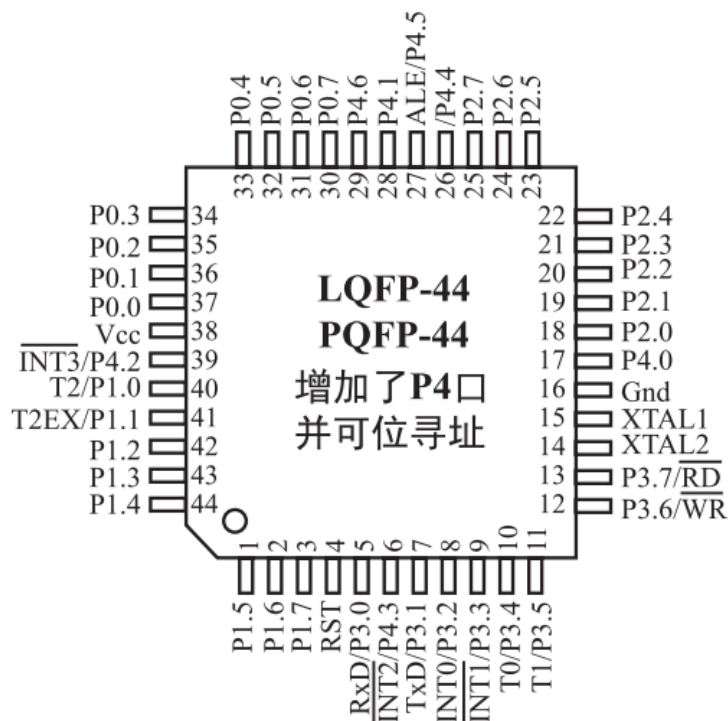


# STC89C52单片机内部结构及特点

- 12.共3个16位定时器/计数器，兼容普通MCS-51单片机的定时器，其中定时器T0还可以当成2个8位定时器使用。
- 13.外部中断4路，下降沿中断或低电平触发中断，掉电模式可由外部中断低电平触发中断方式唤醒。
- 14.通用异步串行口（UART），还可用定时器软件实现多个UART。
- 15.工作温度： $0^{\circ}\text{C} \sim 75^{\circ}\text{C}$ （商业级）/ $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ （工业级）。
- 16.封装形式有：LQFP-44、PDIP-40、PLCC-44、PQFP-44。由于LQFP-44具有体积小、扩展了P4口、外部中断2和3及定时器T2的功能。PDIP-40的封装与传统的89C52芯片兼容。



# STC89C52单片机外部引脚及功能



# STC89C52单片机I/O口

STC89C52RC单片机I/O端口有3种工作类型：

- 准双向口/弱上拉（标准8051输出模式）
- 输入（高阻）
- 开漏输出



# STC89C52单片机I/O口

## 准双向口/弱上拉：

- 可用作输出和输入功能，而不需重新配置引脚输出状态。
- 引脚输出为1时，驱动能力很弱，允许外部装置将其拉低。引脚输出为低时，驱动能力很强，可吸收相当大的电流。
- 因此要想获得较大的驱动能力，采用低电平输出。



# STC89C52单片机I/O口

输入（高阻）：

- 只能作为输入使用，这在模拟比较器和ADC（模数转换）应用中是必需的。
- 高阻态的输入端口可以作模拟输入输出，准双向口只能作数字I/O口输入。如果输入高低电平时，高阻态与准双向口读出的值相同，悬空时，高阻态读出的值不确定，准双向口读出的值是1。



# STC89C52单片机I/O口

## 开漏输出:

- 开漏输出就是不输出电压，低电平时接地，高电平时不接地。如果外接上拉电阻，则在输出高电平时电压会拉到上拉电阻的电源电压。这种方式适合在连接的外设电压比单片机电压低的时候。
- 要得到高电平状态需要上拉电阻。适合于做电流型的驱动，其吸收电流的能力相对强（一般20mA以内）。



# STC89C52单片机I/O口

## P0端口

- 上电复位后处于开漏模式，当P0管脚作I/O口时，需外加10K~4.7K的上拉电阻，当P0管脚作为地址/数据复用总线使用时，不用外加上拉电阻。

## P1/P2/P3/P4端口

- 上电复位后为准双向口/弱上拉（传统8051的I/O口）模式。



# I/O口定义与赋值

## I/O口定义

- 特殊功能位声明 `sbit IN1=P1^3; //自定义一位`

## I/O口赋值

- `IN1=0;` //将P1^3置0（仅高低）
- `P2=0xff;` //将P2口的8位全部置1（高电平VCC）





# STC89C52单片机时钟与复位

## 51单片机时序

单片机工作时，是在统一的时钟脉冲控制下一拍一拍地进行的。时钟电路产生单片机工作时所必需的控制信号，在时钟信号的控制下，严格按时序执行指令。由于指令的字节数不同，取这些指令所需要的时间也就不同，即使是字节数相同的指令，由于执行操作有较大的差别，不同的指令执行时间也不一定相同，即所需的拍节数不同。为了便于对CPU时序进行分析，一般按指令的执行过程规定了几种周期，即**时钟周期、机器周期和指令周期**，也称为时序定时单位。



# STC89C52单片机时钟与复位

## 1. 时钟周期

- 也称为振荡周期，定义为时钟脉冲的倒数，**是计算机中最基本的、最小的时间单位**。可以这么理解，时钟周期就是单片机外接晶振的倒数，例如12M的晶振，它的时钟周期就是 $1/12\text{ us}$ 。
- 显然，对同一种机型的单片机，时钟频率越高，单片机的工作速度就越快。但是，由于不同的单片机硬件电路和器件的不完全相同，所以其所需要的时钟频率范围也不一定相同。在单片机中把**一个时钟周期定义为一个节拍（用P表示）**，**二个节拍定义为一个状态周期（用S表示）**。



# STC89C52单片机时钟与复位

## 2. 机器周期

- 在单片机中，为了便于管理，常把一条指令的执行过程划分为若干个阶段，每一阶段完成一项工作。例如，取指令、存储器读、存储器写等，这每一项工作称为一个基本操作。**完成一个基本操作所需要的时间称为机器周期。**
- 一般情况下，一个机器周期由若干个s周期（状态周期）组成。51系列单片机的一个机器周期由6个s周期（状态周期）组成。因一个时钟周期为一个节拍，二个节拍为一个状态周期，51单片机的机器周期由6个状态周期组成，也就是说**一个机器周期=6个状态周期=12个时钟周期。**

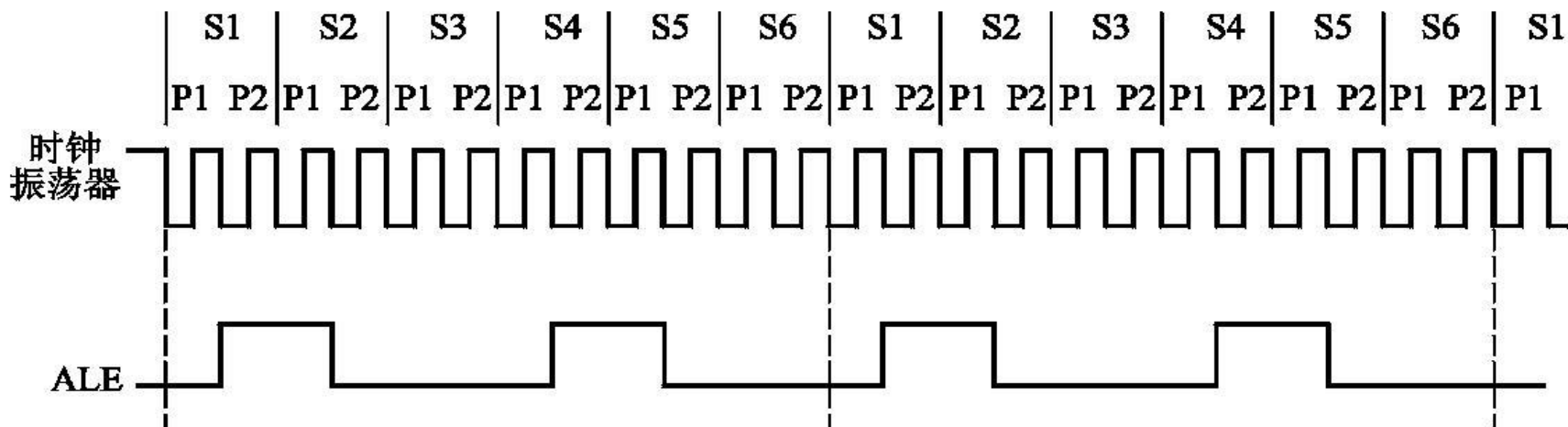


# STC89C52单片机时钟与复位

1个机器周期包括12个时钟周期，分6个状态：S1 ~ S6。

每个状态又分两拍：P1和P2。

因此，一个机器周期中的12个时钟周期表示为S1P1、S1P2、S2P1、S2P2、...、S6P2。



# STC89C52单片机时钟与复位

## 3. 指令周期

- **指令周期是执行一条指令所需要的时间**，一般由若干个机器周期组成。指令不同，所需的机器周期数也不同。对于一些简单的单字节指令，在取指令周期中，指令取出到指令寄存器后，立即译码执行，不再需要其它的机器周期。对于一些比较复杂的指令，例如转移指令、乘法指令，则需要两个或者两个以上的机器周期。
- 通常含一个机器周期的指令称为单周期指令，包含两个机器周期的指令称为双周期指令。51单片机的指令系统中，按它们的长度可分为单字节指令、双字节指令和三字节指令。
- 从指令执行时间看：单字节和双字节指令一般为单机器周期和双机器周期；三字节指令都是双机器周期；乘、除指令占用四个机器周期。



# STC89C52单片机时钟与复位

## 时钟电路：

- CPU发出的时序信号作用：
  - 用于对片内各个功能部件控制，用户无须了解；
  - 用于对片外存储器或I/O口的控制，这部分时序对于分析、设计硬件接口电路至关重要。
- 时钟频率直接影响单片机的速度，时钟电路的质量也直接影响单片机系统的稳定性。常用的时钟电路有两种方式，**一种是内部时钟方式，另一种是外部时钟方式。**



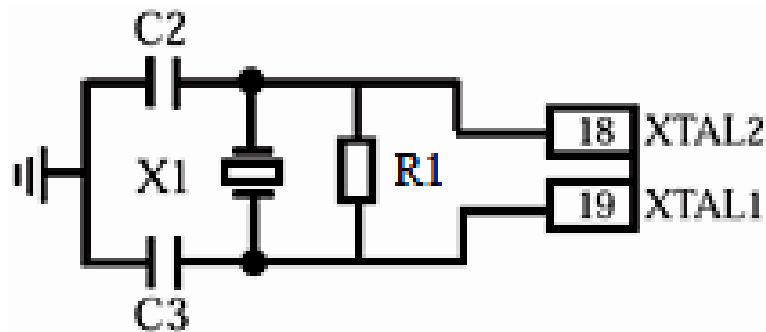
# STC89C52单片机时钟与复位

## 1. 内部时钟方式

- STC89C52内部有一个用于构成振荡器的高增益反相放大器，输入端为XTAL1，输出端为XTAL2。这两个引脚接石英晶体振荡器和微调电容，构成一个稳定的自激振荡器。

## 2. 外部时钟方式

- 利用外部振荡脉冲接入XTAL1，XTAL2端悬空。
- 用现成的外部振荡器产生脉冲信号，常用于多片单片机同时工作，以便于多片单片机之间的同步，一般采用内部时钟方式产生工作时序。



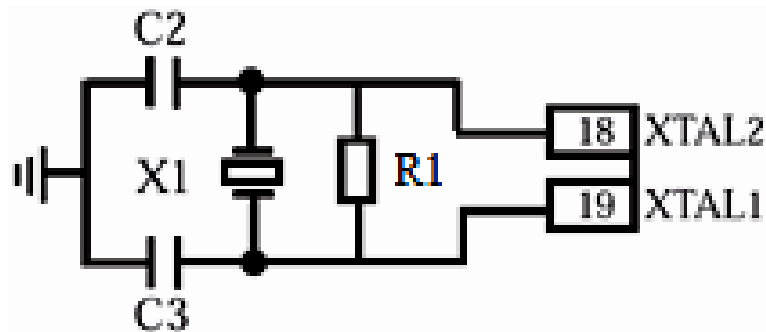
# STC89C52单片机时钟与复位

## 1. 内部时钟方式

- STC89C52内部有一个用于构成振荡器的高增益反相放大器，输入端为XTAL1，输出端为XTAL2。这两个引脚接石英晶体振荡器和微调电容，构成一个稳定的自激振荡器。

## 2. 外部时钟方式

- 利用外部振荡脉冲接入XTAL1，XTAL2端悬空。
- 用现成的外部振荡器产生脉冲信号，常用于多片单片机同时工作，以便于多片单片机之间的同步，一般采用内部时钟方式产生工作时序。





# STC89C52单片机时钟与复位

## STC89C52单片机的复位电路

- 复位是单片机的初始化操作。单片机启动运行时，都需要先复位，其作用是使CPU和系统中其他部件处于一个确定的初始状态，并从这个状态开始工作。因而，复位是一个很重要的操作方式。但单片机本身是不能自动进行复位的，必须配合相应的外部电路才能实现。
- STC89C52RC系列单片机有4种复位方式：外部RST引脚复位、软件复位、掉电复位、上电复位、看门狗复位。



## 参考资料：

1. 陈中，朱代忠 著: 基于STC89C52单片机的控制系统设计，清华大学出版社，2015
2. 朱兆优 著：单片机原理与应用——基于STC系列增强型80C51单片机(第3版)，2016
3. STC89C51RC--RD+系列单片机使用手册，2015

