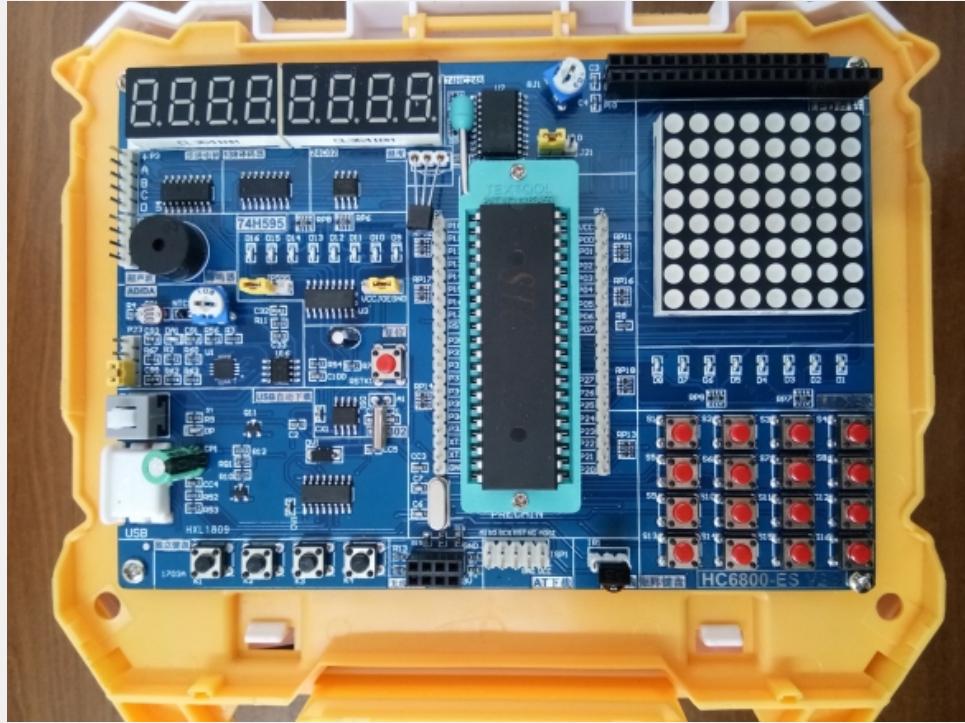


课程简介

硬件设备



51单片机开发板



Windows电脑

软件设备



Keil5
编写程序



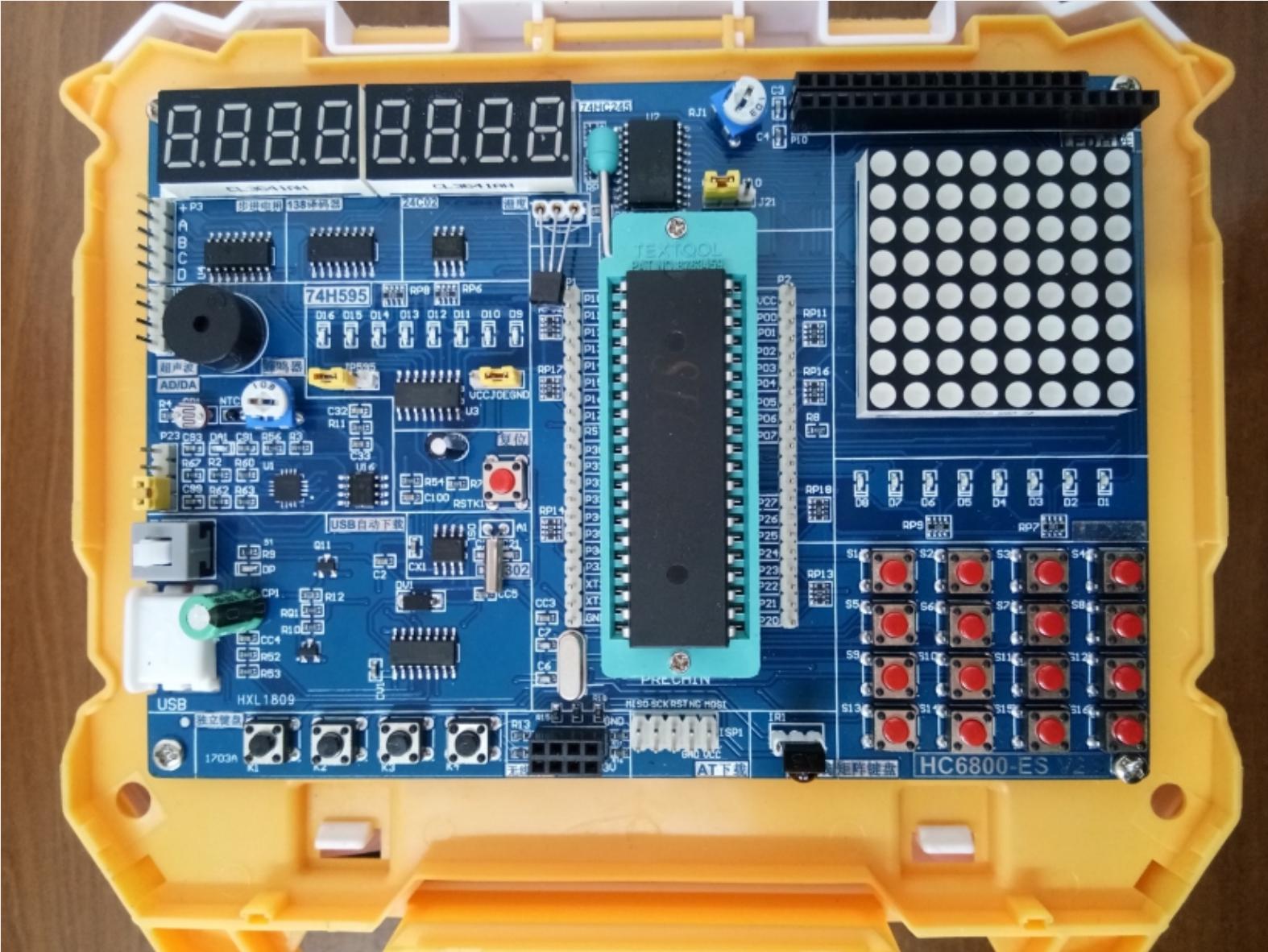
STC-ISP
下载程序

资料准备

开发板资料：

密码：

视频目的



软件安装



Keil5（必装
）



有道词典（选装
）



STC-ISP（必装
）



福昕阅读器（选装
）

Keil5软件注意事项

- Keil5 C51和Keil5 MDK的区别

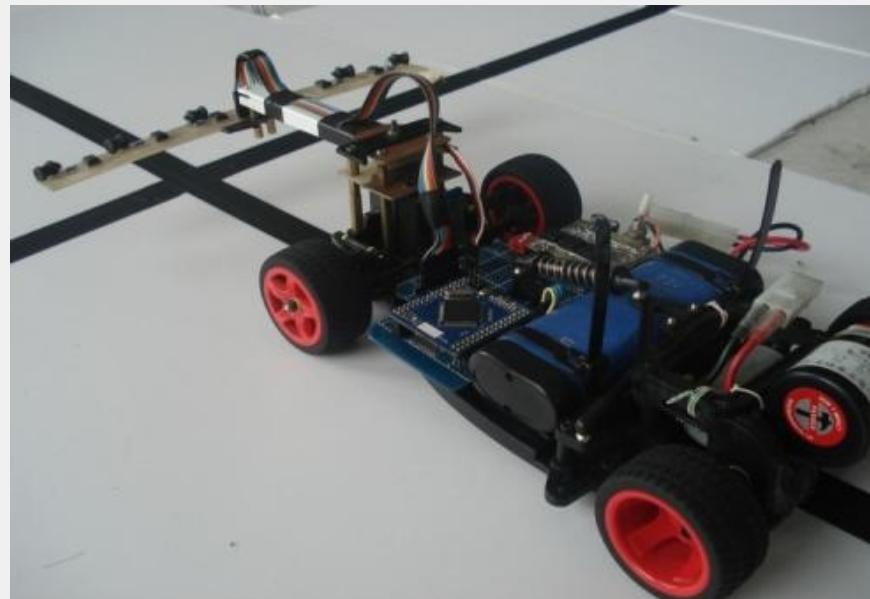
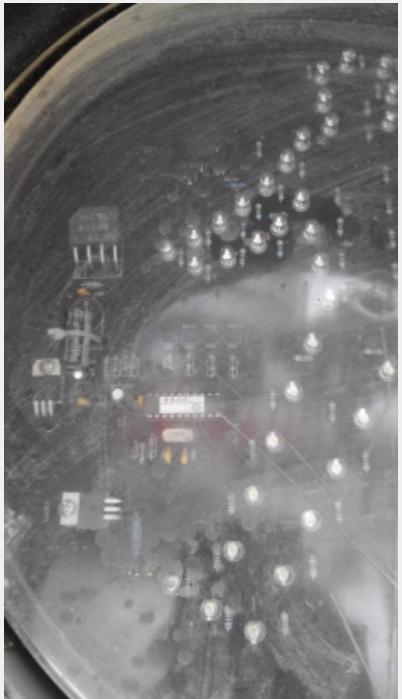
两者都是Keil系列软件，但前者是用来开发51单片机的，后者是用来开发ARM系列，比如STM32的。

单片机介绍

- 单片机，英文Micro Controller Unit，简称MCU
- 内部集成了CPU、RAM、ROM、定时器、中断系统、通讯接口等一系列电脑的常用硬件功能
- 单片机的任务是信息采集（依靠传感器）、处理（依靠CPU）和硬件设备（例如电机，LED等）的控制
- 单片机跟计算机相比，单片机算是一个袖珍版计算机，一个芯片就能构成完整的计算机系统。但在性能上，与计算机相差甚远，但单片机成本低、体积小、结构简单，在生活和工业控制领域大有所用
- 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

单片机的应用领域

- 单片机的使用领域已十分广泛，如智能仪表、实时工控、通讯设备、导航系统、家用电器等。各种产品一旦用上了单片机，就能起到使产品升级换代的功效，常在产品名称前冠以形容词——“智能型”，如智能型洗衣机等



STC89C52单片机

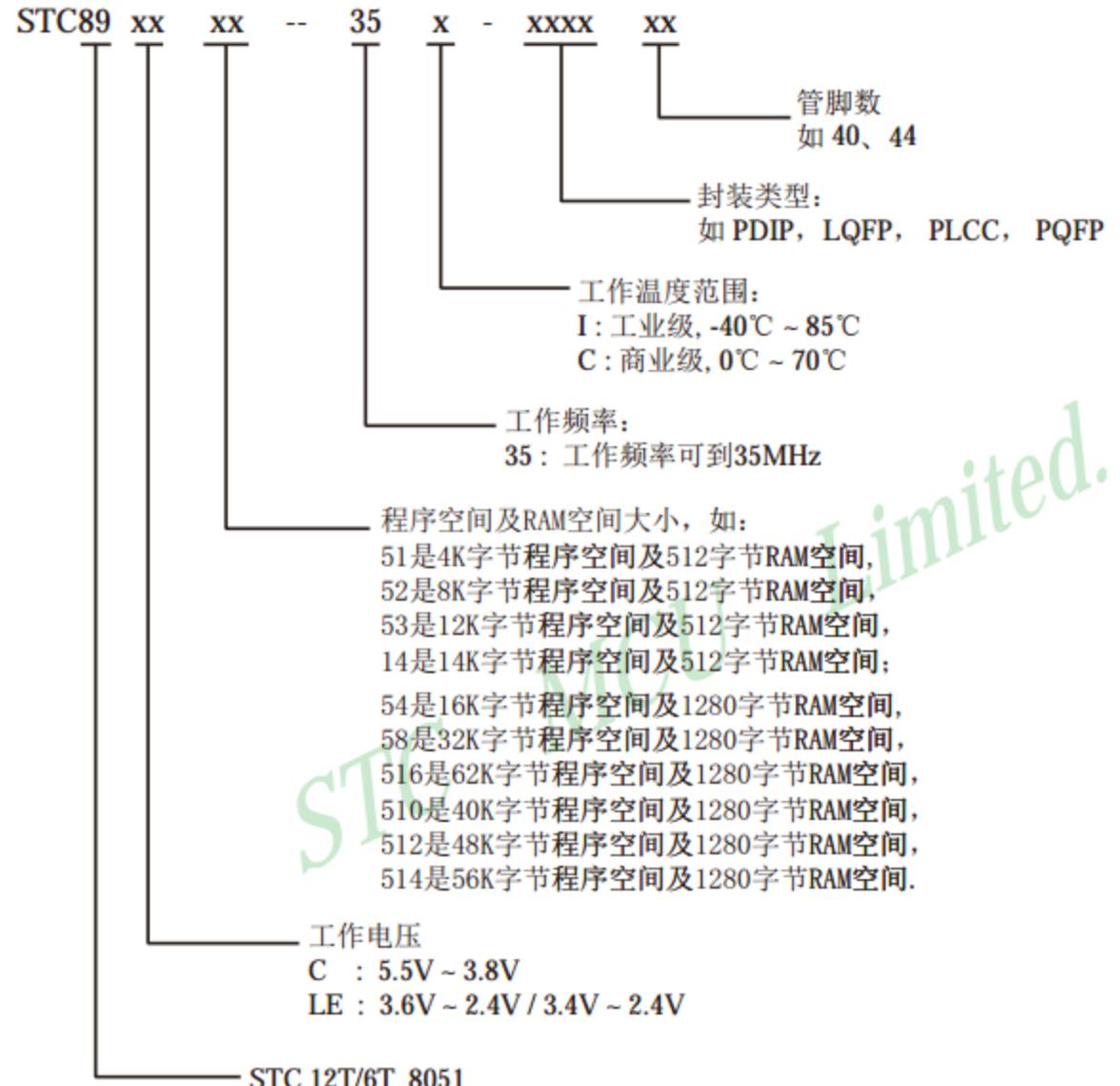
- 所属系列：51单片机系列
- 公司：STC公司
- 位数：8位
- RAM：512字节
- ROM：8K（Flash）
- 工作频率：12MHz（本开发板使用）



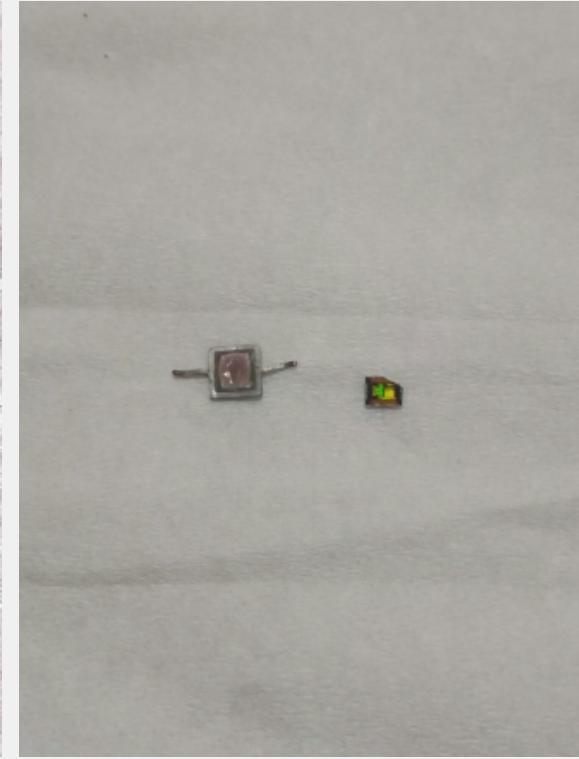
命名规则



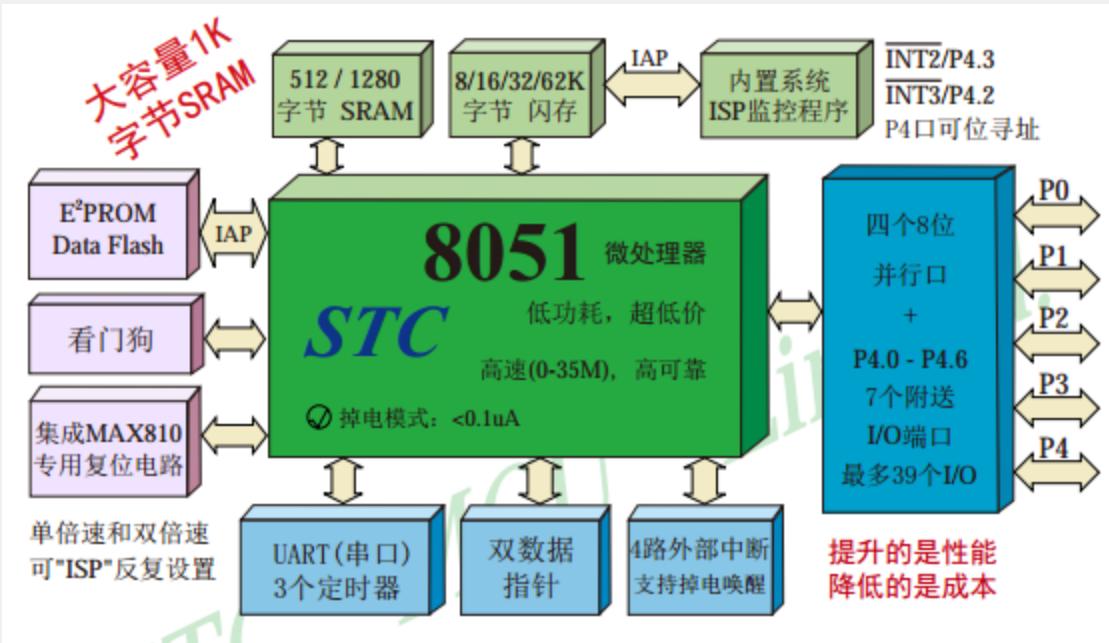
1.5 STC89C52系列单片机命名规则



单片机内部拆解

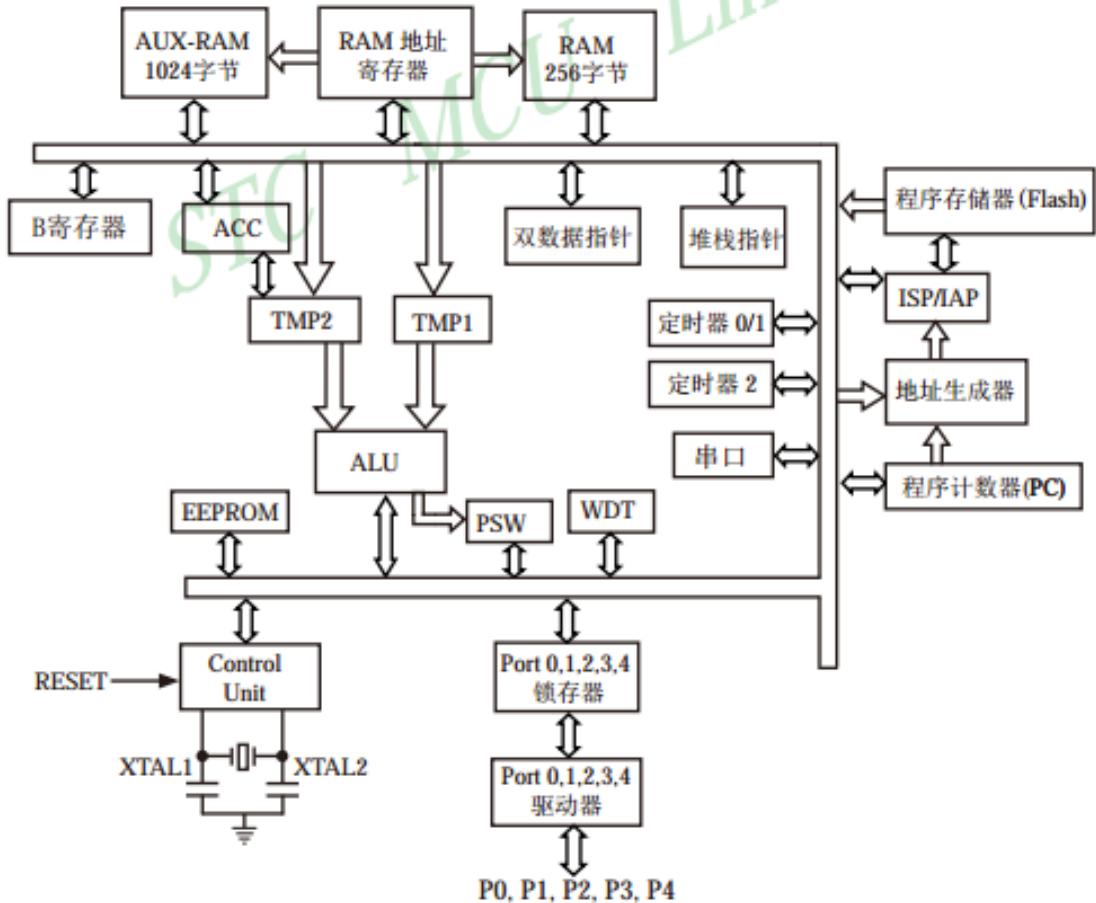


单片机内部结构图



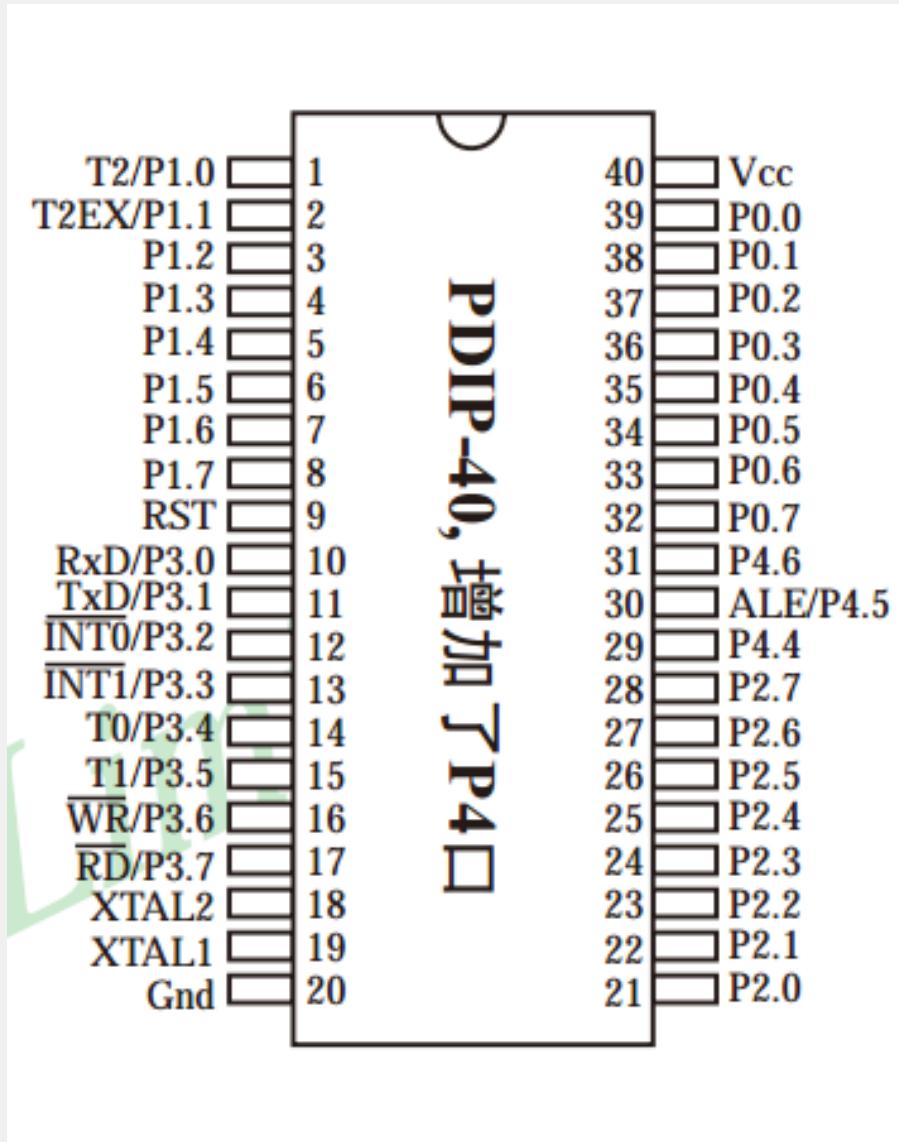
1.2 STC89C52系列单片机的内部结构

STC89C52系列单片机的内部结构框图如下图所示。STC89C52单片机中包含中央处理器(CPU)、程序存储器(Flash)、数据存储器(SRAM)、定时/计数器、UART串口、I/O接口、EEPROM、看门狗等模块。STC89C52系列单片机几乎包含了数据采集和控制中所需的所有单元模块，可称得上一个片上系统。



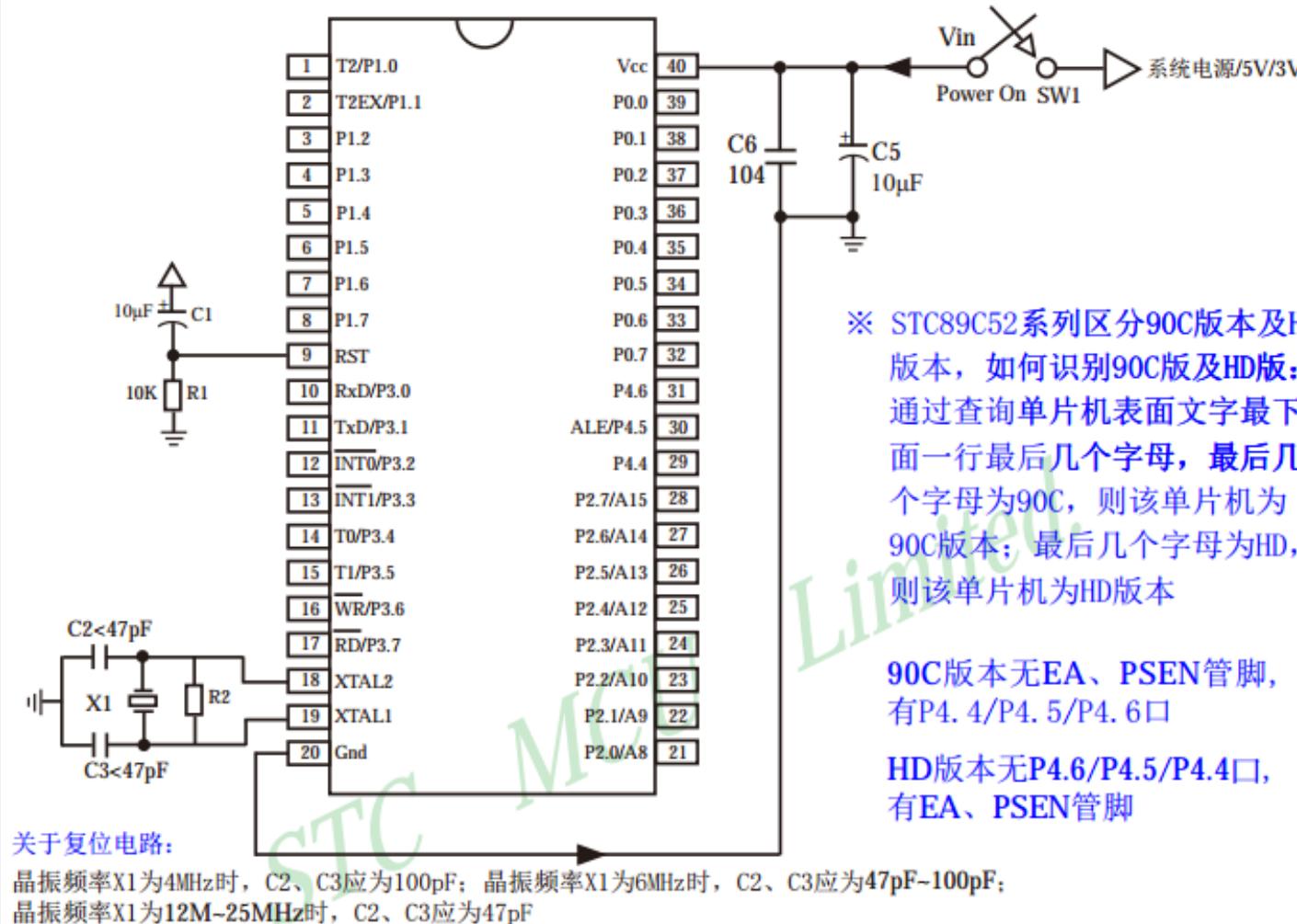
STC89C52系列内部结构框图

单片机管脚图

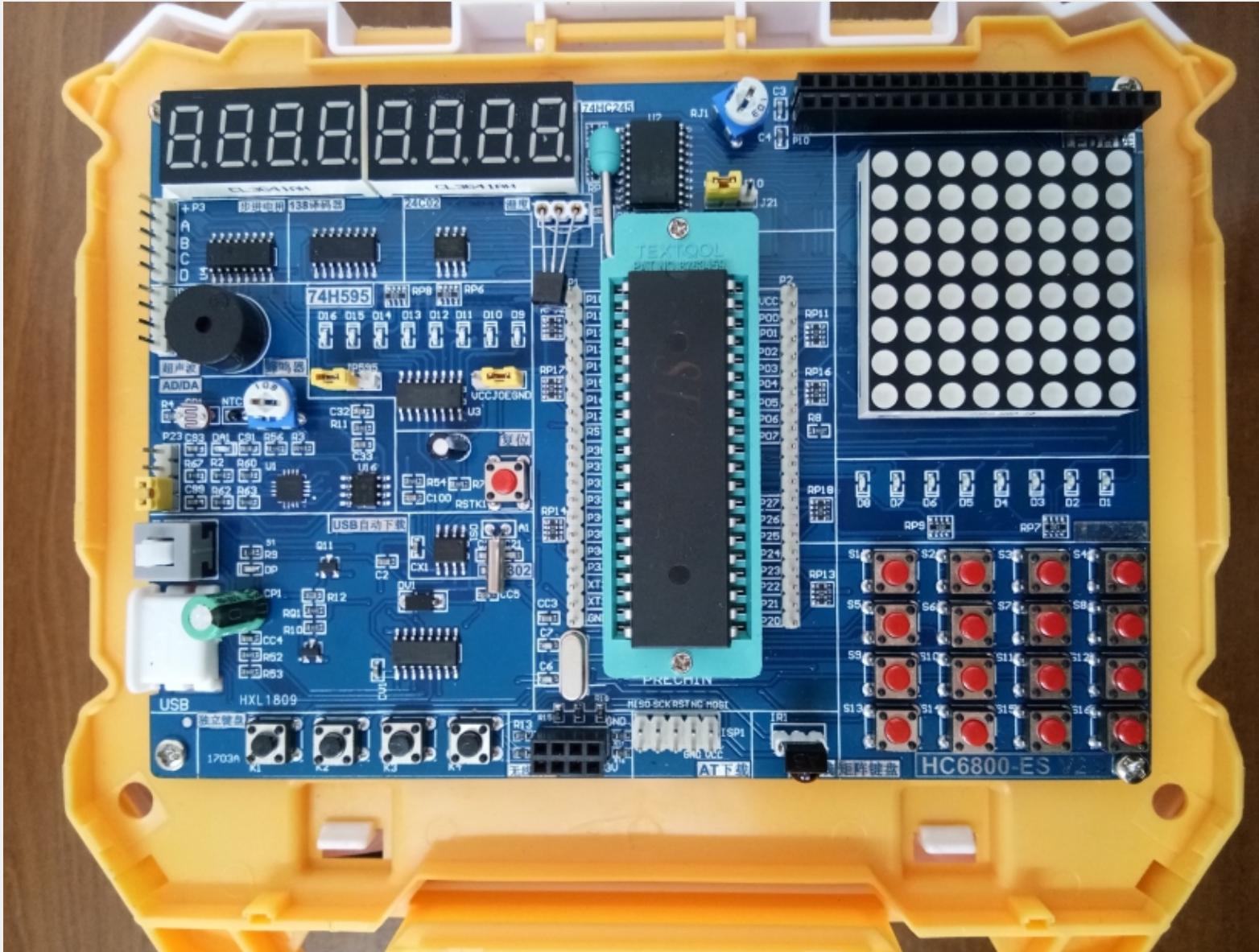


单片机最小系统

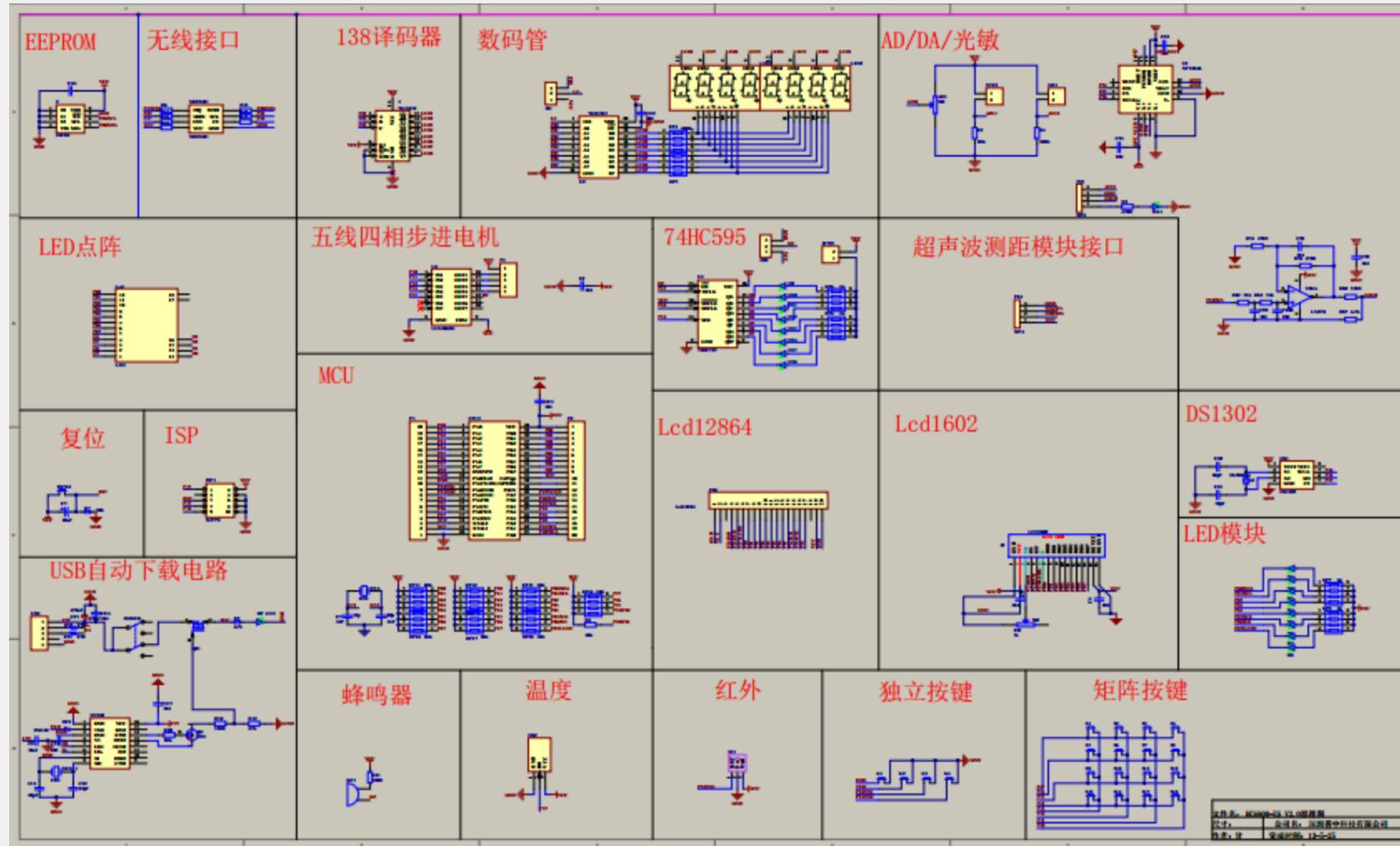
1.6 STC89C52系列单片机最小应用系统



开发板介绍



开发板原理图



LED介绍

- 中文名：发光二极管
- 外文名：Light Emitting Diode
- 简称：LED
- 用途：照明、广告灯、指引灯、屏幕

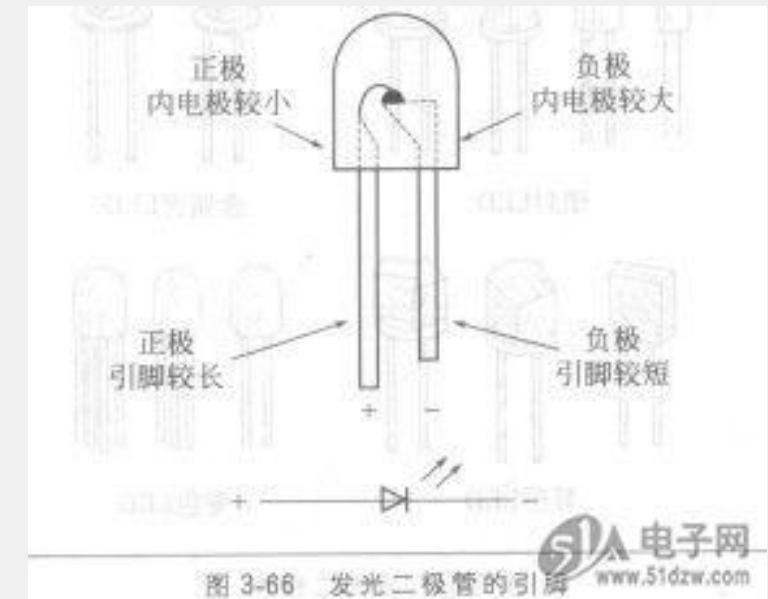
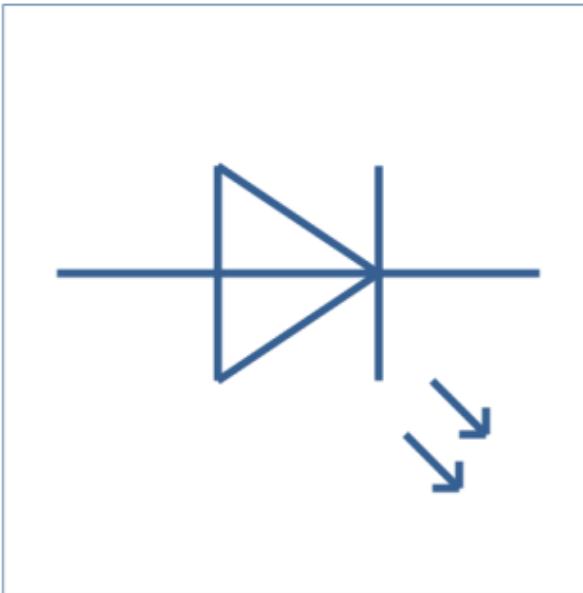


图 3-66 发光二极管的引脚
www.51dzw.com

进制转换

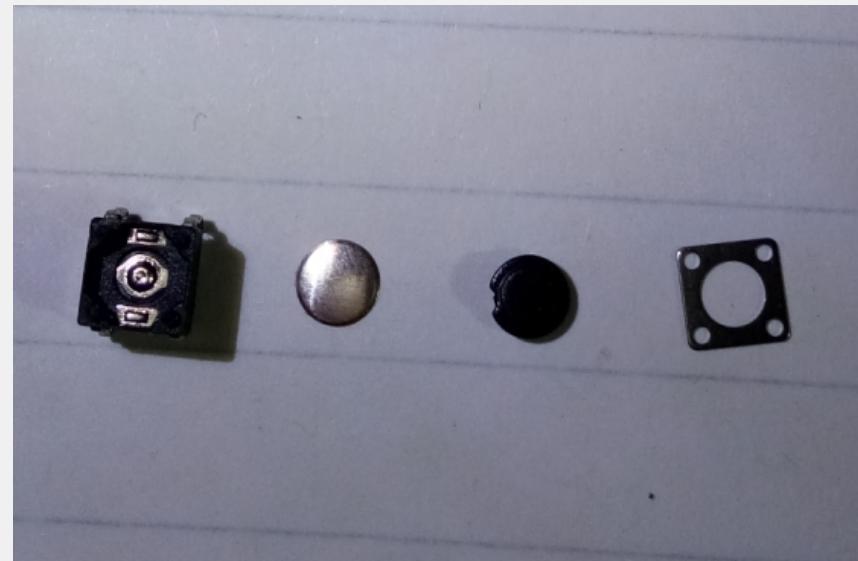
十进制	二进制	十六进制	十进制	二进制	十六进制
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

C51数据类型

类型	符号	关键字	所占位数	数的表示范围
整型	有	(signed) int	16	-32768~32767
		(signed) short	16	-32768~32767
		(signed) long	32	-2147483648~2147483647
	无	unsigned int	16	0~65535
		unsigned short int	16	0~65535
		unsigned long int	32	0~4294967295
实型	有	float	32	3.4e-38~3.4e38
	有	double	64	1.7e-308~1.7e308
字符型	有	char	8	-128~127
	无	unsigned char	8	0~255

按键介绍

- 轻触按键：相当于是一种电子开关，按下时开关接通，松开时开关断开，实现原理是通过轻触按键内部的金属弹片受力弹动来实现接通和断开



C51数据运算

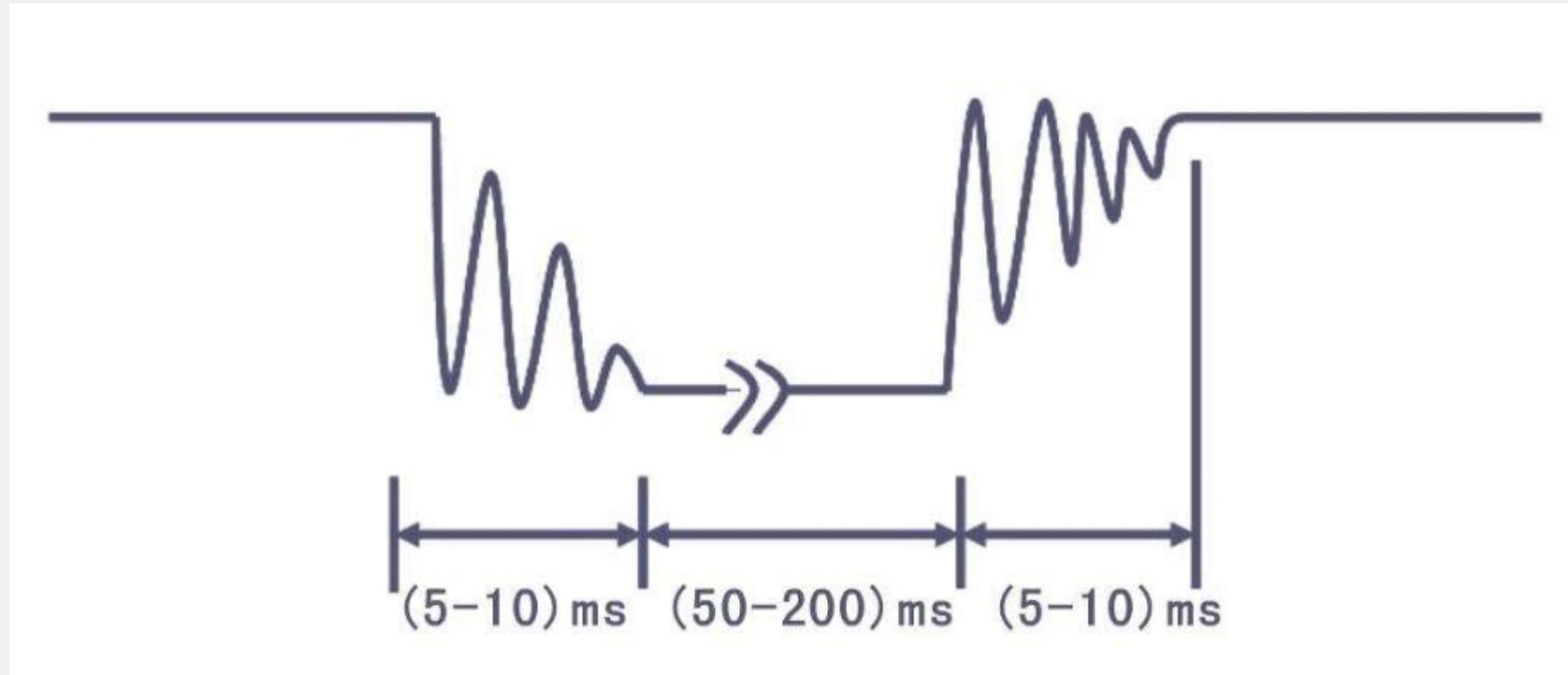
类别	运算符	意义	类别	运算符	意义
算术	+	加	逻辑	&&	逻辑与
	-	减			逻辑或
	*	乘		!	逻辑非
	/	除		<<	按位左移
	%	取余		>>	按位右移
	=	赋值		&	按位与
判断	>	大于	位运算		按位或
	>=	大于等于		^	按位异或
	<	小于		~	按位取反
	<=	小于等于			
	==	等于			
	!=	不等于			

C51基本语句

语句	解释	语句	解释
<pre>if(逻辑表达式) { 语句体1; } else { 语句体2; }</pre>	<p>如果逻辑表达式成立 执行语句体1 否则 执行语句体2 (else可以不写)</p>	<pre>for(初始化;逻辑表达式;更改条件) { 循环体; }</pre>	<p>先执行初始化 再判断逻辑表达式 若成立则执行循环体 执行后更改条件 再判断逻辑表达式 直到表达式不成立</p>
<pre>while(逻辑表达式) { 循环体; }</pre>	<p>如果逻辑表达式成立 执行循环体 执行后再次判断 若还成立则继续执行直到表达式不成立</p>	<pre>switch(变量) { case 常量1:语句体1;break; case 常量2:语句体2;break; ... default:语句体x;break; }</pre>	<p>将变量与case后的各个常量对比 若有相等，则执行相应的语句体 若没有一个相等，则执行default后的语句体 (default可以不写)</p>

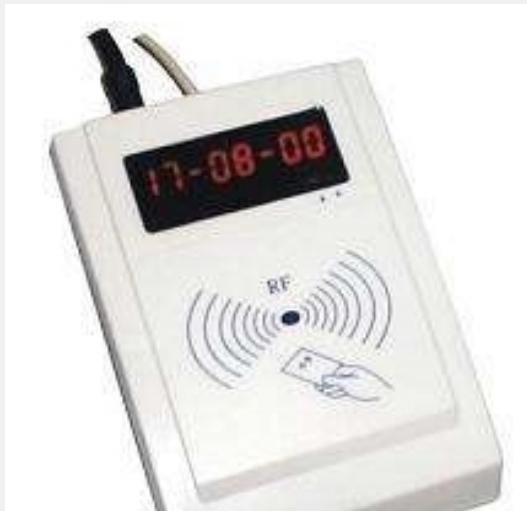
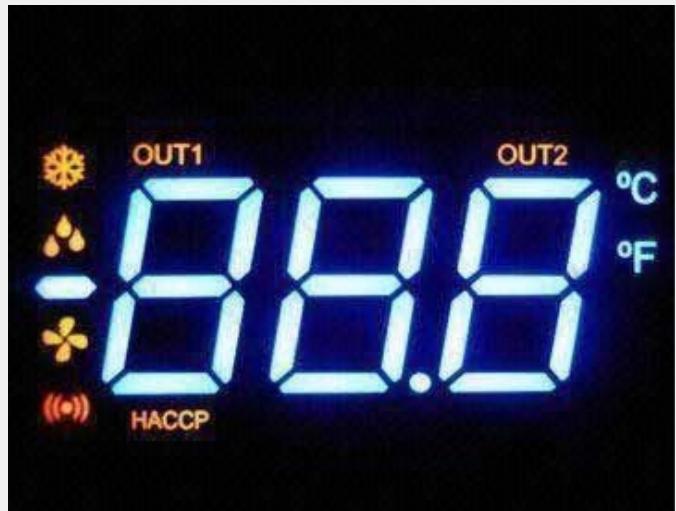
按键的抖动

- 对于机械开关，当机械触点断开、闭合时，由于机械触点的弹性作用，一个开关在闭合时不会马上稳定地接通，在断开时也不会一下子断开，所以在开关闭合及断开的瞬间会伴随一连串的抖动

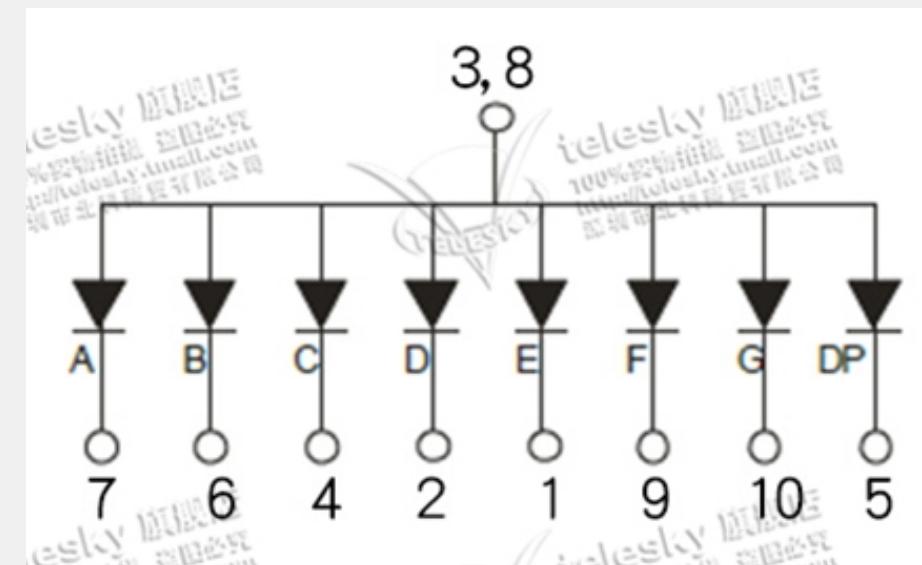
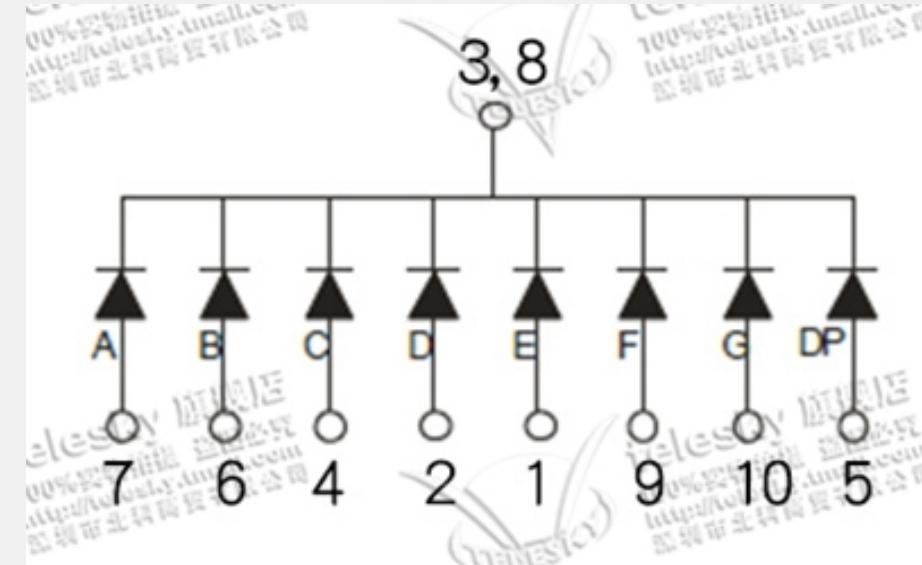
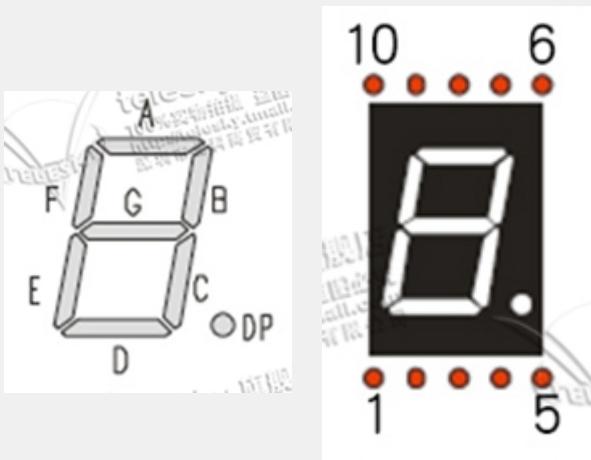
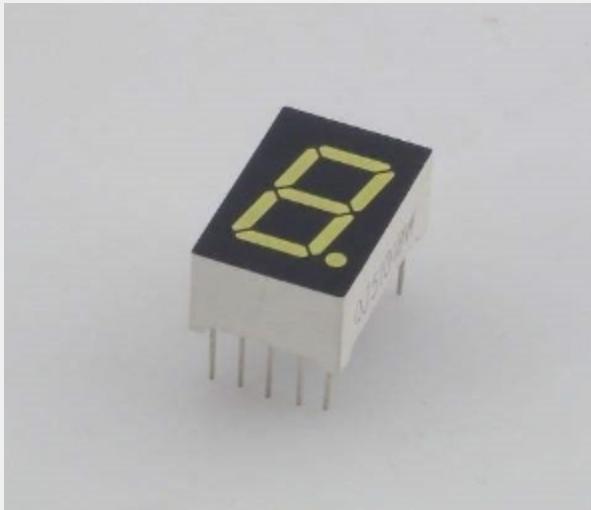


数码管介绍

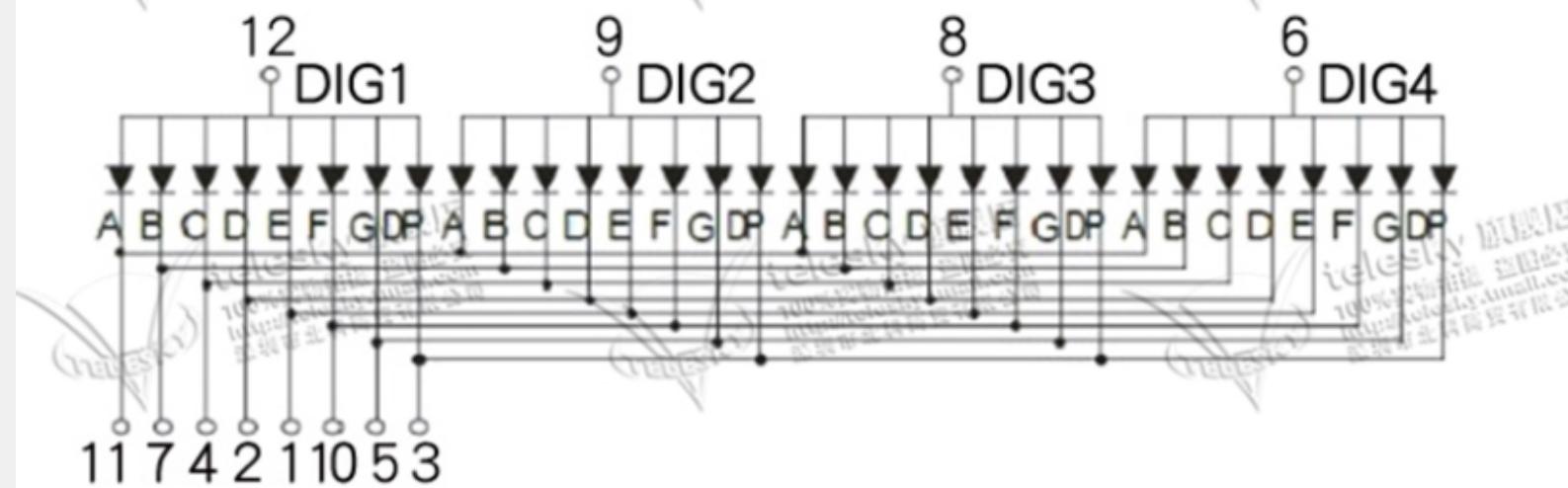
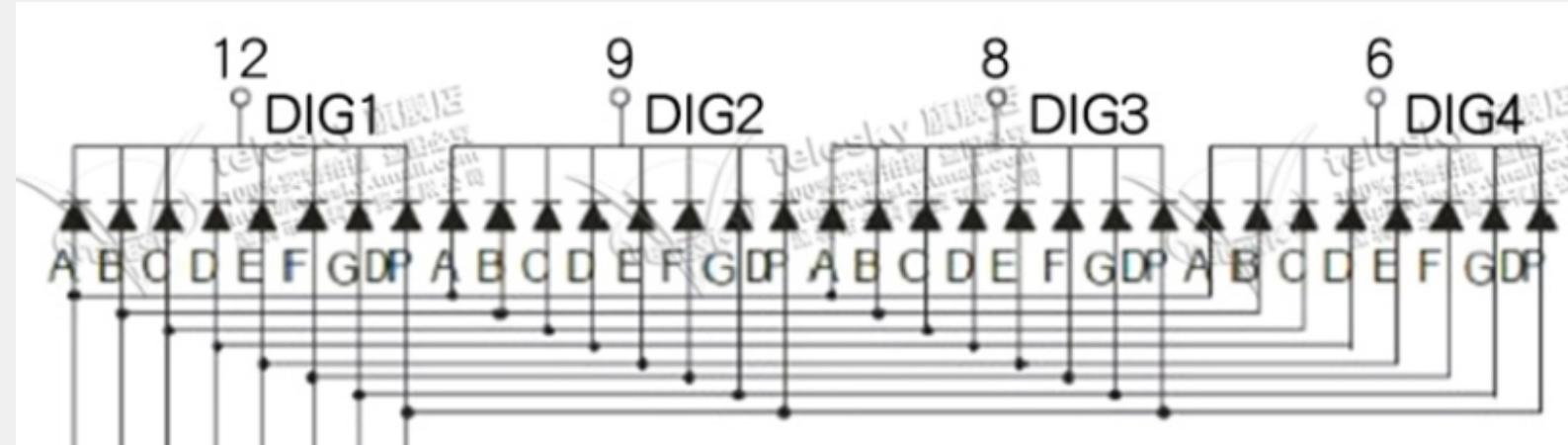
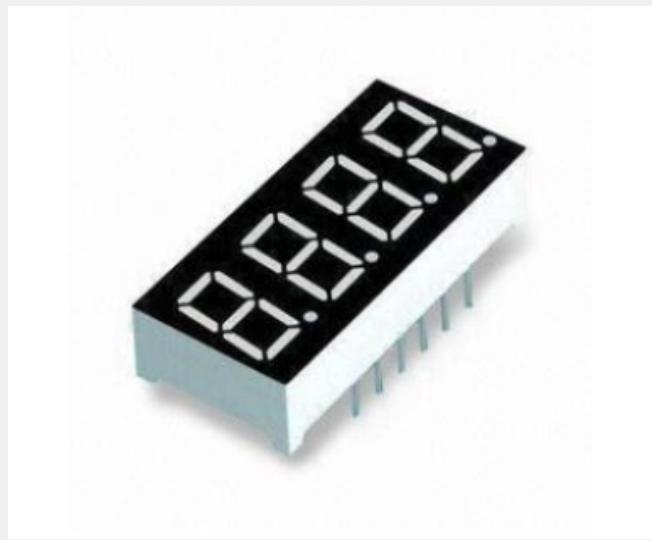
- LED数码管：数码管是一种简单、廉价的显示器，是由多个发光二极管封装在一起组成“8”字型的器件



数码管引脚定义



数码管引脚定义



- 数组：把相同类型的一系列数据统一编制到某一个组别中，可以通过数组名+索引号简单快捷的操作大量数据

```
int x[3];           // 定义一组变量（3个）
int x[]={1,2,3};   // 定义一组变量并初始化
```

```
x[0]           // 引用数组的第0个变量
x[1]           // 引用数组的第1个变量
x[2]           // 引用数组的第2个变量
```

引用x[3]时，数组越界，读出的数值不确定，应避免这种操作

C51子函数

- 子函数：将完成某一种功能的程序代码单独抽取出来形成一个模块，在其它函数中可随时调用此模块，以达到代码的复用和优化程序结构的目的

```
void Function(unsigned char x, y)
{
```

```
}
```

返回值 函数名 (形参)

```
{
```

 函数体

```
}
```

数码管段码表

0 1 2 3 4 5 6 7 8 9

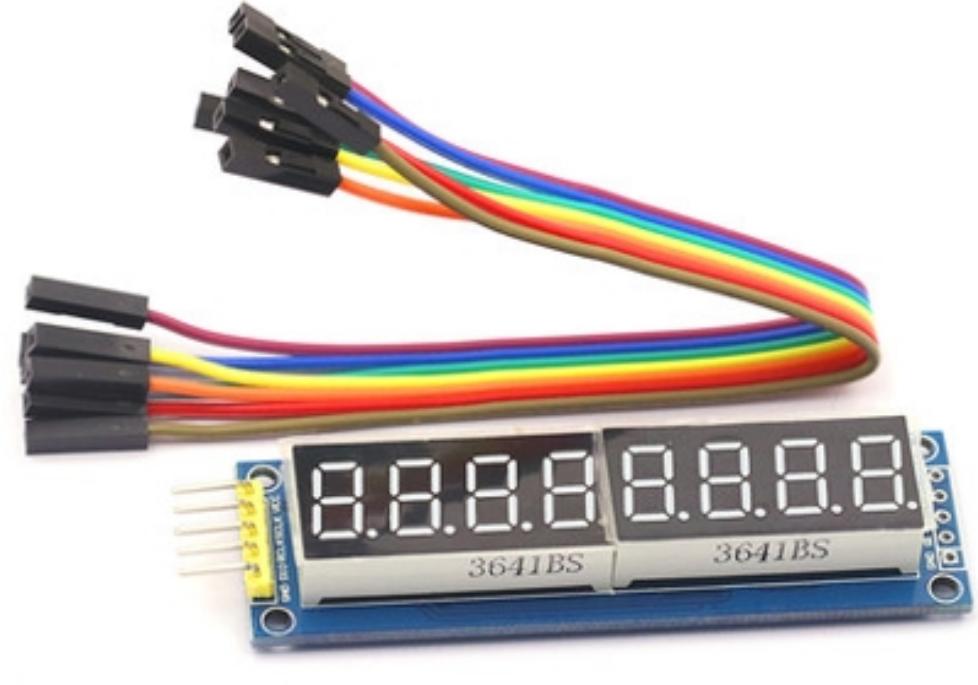
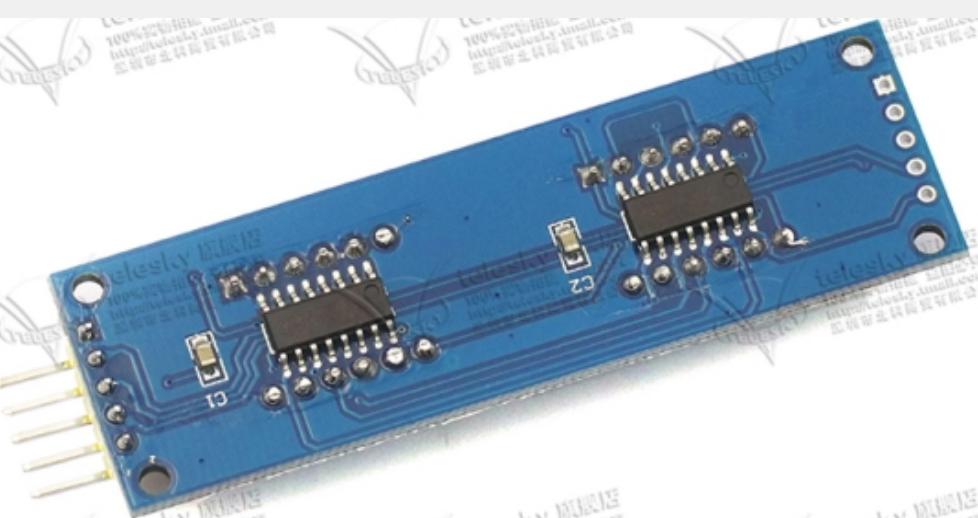
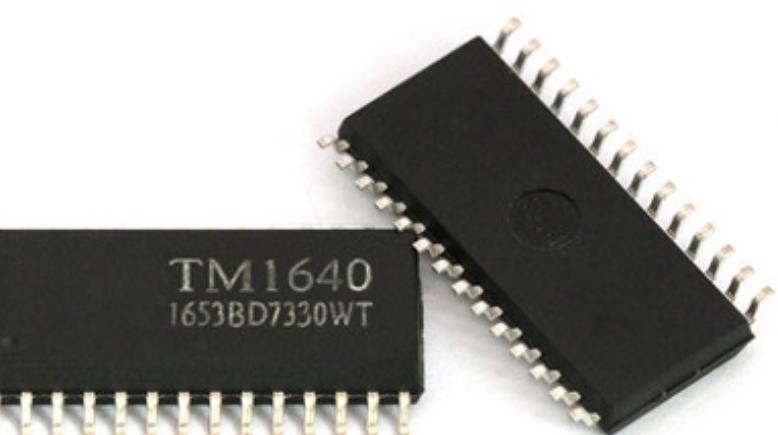
0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F,

A B C D E F 空

0x77, 0x7C, 0x39, 0x5E, 0x79, 0x71, 0x00,

数码管驱动方式

- 单片机直接扫描：硬件设备简单，但会耗费大量的单片机CPU时间
- 专用驱动芯片：内部自带显存、扫描电路，单片机只需告诉它显示什么



模块化编程

- 传统方式编程：所有的函数均放在main.c里，若使用的模块比较多，则一个文件内会有很多的代码，不利于代码的组织和管理，而且很影响编程者的思路
- 模块化编程：把各个模块的代码放在不同的.c文件里，在.h文件里提供外部可调用函数的声明，其它.c文件想使用其中的代码时，只需要#include "XXX.h"文件即可。使用模块化编程可极大的提高代码的可阅读性、可维护性、可移植性等

模块化编程框图

```
#include "Delay.h"

void main()
{
    Delay(10);
}
```

main.c

```
#ifndef __DELAY_H__
#define __DELAY_H__

void Delay(unsigned int
xms);

#endif
```

Delay.h

```
void Delay(unsigned int xms)
{
    unsigned char i, j;
    while(xms--)
    {
        i = 2;
        j = 239;
        do
        {
            while (--j);
        } while (--i);
    }
}
```

Delay.c

Delay函数模块化

模块化编程注意事项

- .c文件：函数、变量的定义
- .h文件：可被外部调用的函数、变量的声明
- 任何自定义的变量、函数在调用前必须有定义或声明（同一个.c）
- 使用到的自定义函数的.c文件必须添加到工程参与编译
- 使用到的.h文件必须要放在编译器可寻找到的地方（工程文件夹根目录、安装目录、自定义）

C预编译

- C语言的预编译以#开头，作用是在真正的编译开始之前，对代码做一些处理（预编译）

预编译	意义
#include <REGX52.H>	把REGX52.H文件的内容搬到此处
#define PI 3.14	定义PI，将PI替换为3.14
#define ABC	定义ABC
#ifndef __XX_H__	如果没有定义__XX_H__
#endif	与#ifndef,#if匹配，组成“括号”

- 此外还有#define,#if,#else,#elif,#undef等

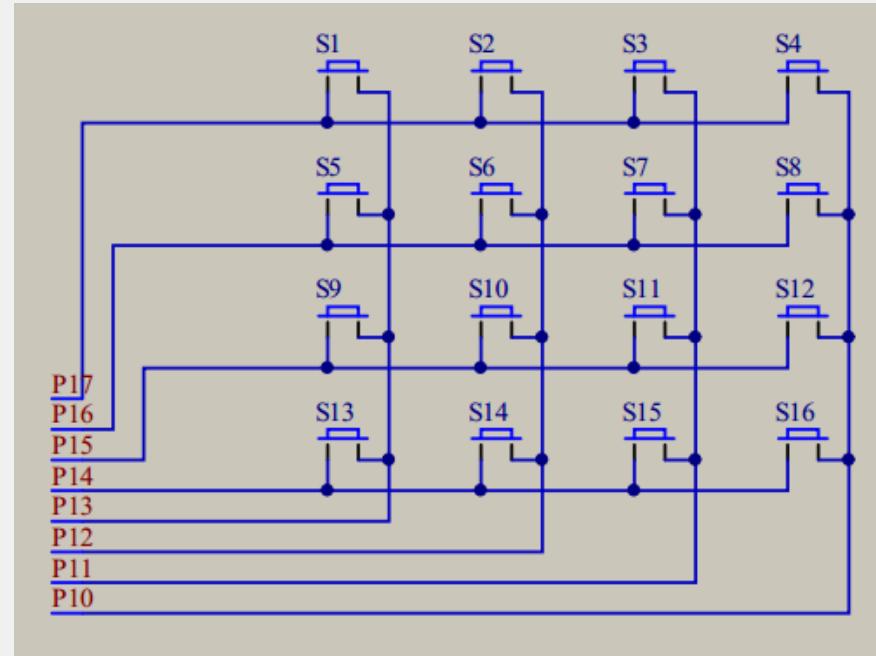
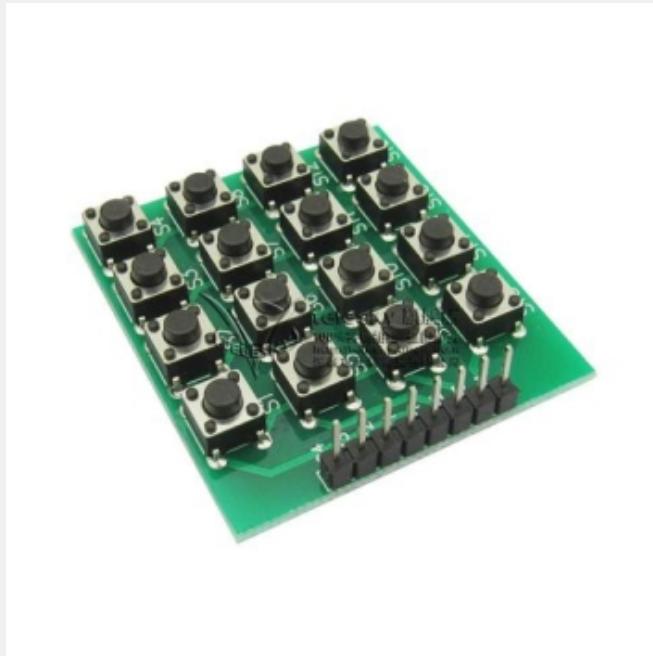
LCD1602调试工具

- 使用LCD1602液晶屏作为调试窗口，提供类似printf函数的功能，可实时观察单片机内部数据的变换情况，便于调试和演示。
- 本视频提供的LCD1602代码属于模块化的代码，使用者只需要知道所提供函数的作用和使用方法就可以很容易的使用LCD1602

函数	作用
LCD_Init();	初始化
LCD_ShowChar(1,1,'A');	显示一个字符
LCD_ShowString(1,3,"Hello");	显示字符串
LCD_ShowNum(1,9,123,3);	显示十进制数字
LCD_ShowSignedNum(1,13,-66,2);	显示有符号十进制数字
LCD_ShowHexNum(2,1,0xA8,2);	显示十六进制数字
LCD_ShowBinNum(2,4,0xAA,8);	显示二进制数字

矩阵键盘介绍

- 在键盘中按键数量较多时，为了减少I/O口的占用，通常将按键排列成矩阵形式
- 采用逐行或逐列的“扫描”，就可以读出任何位置按键的状态



扫描的概念

- 数码管扫描（输出扫描）

原理：显示第1位→显示第2位→显示第3位→.....，然后快速循环这个过程，最终实现所有数码管同时显示的效果

- 矩阵键盘扫描（输入扫描）

原理：读取第1行(列)→读取第2行(列)→读取第3行(列)→.....，然后快速循环这个过程，最终实现所有按键同时检测的效果

- 以上两种扫描方式的共性：节省I/O口

定时器介绍

- 定时器介绍：51单片机的定时器属于单片机的内部资源，其电路的连接和运转均在单片机内部完成
- 定时器作用：
 - (1) 用于计时系统，可实现软件计时，或者使程序每隔一固定时间完成一项操作
 - (2) 替代长时间的Delay，提高CPU的运行效率和处理速度
 - (...)

STC89C52定时器资源

- 定时器个数：3个（T0、T1、T2），T0和T1与传统的51单片机兼容，T2是此型号单片机增加的资源
- 注意：定时器的资源和单片机的型号是关联在一起的，不同的型号可能会有不同的定时器个数和操作方式，但一般来说，T0和T1的操作方式是所有51单片机所共有的

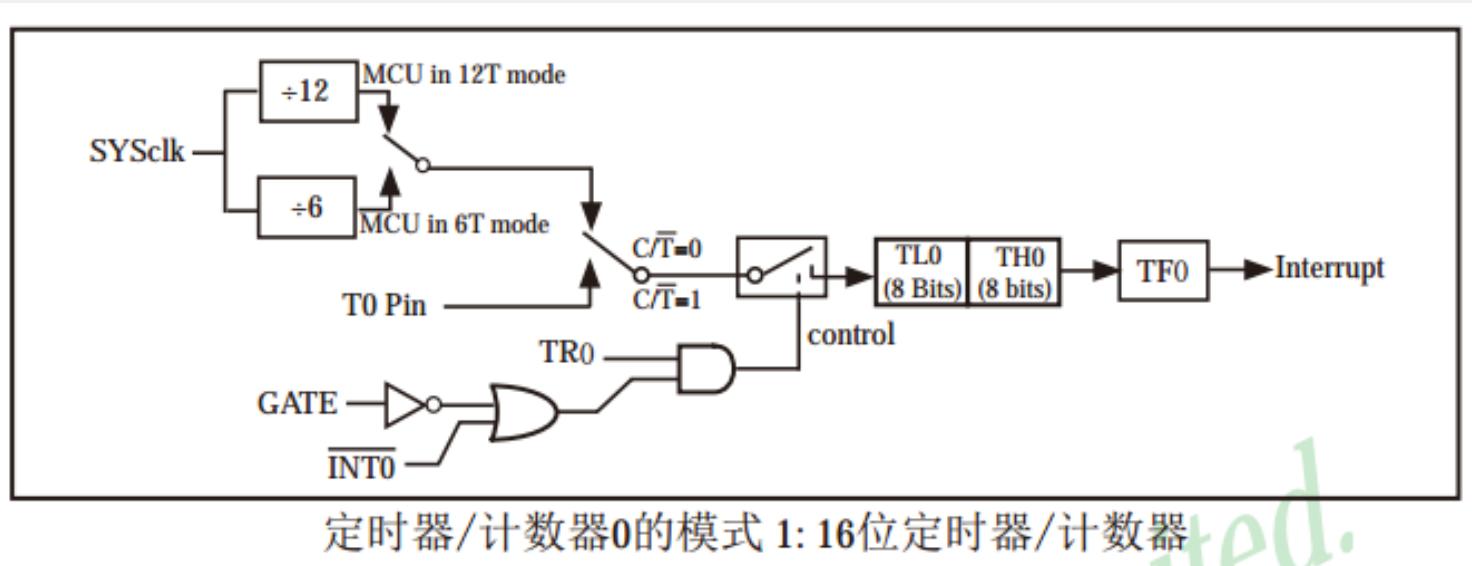
定时器框图

- 定时器在单片机内部就像一个小闹钟一样，根据时钟的输出信号，每隔“一秒”，计数单元的数值就增加一，当计数单元数值增加到“设定的闹钟提醒时间”时，计数单元就会向中断系统发出中断申请，产生“响铃提醒”，使程序跳转到中断服务函数中执行

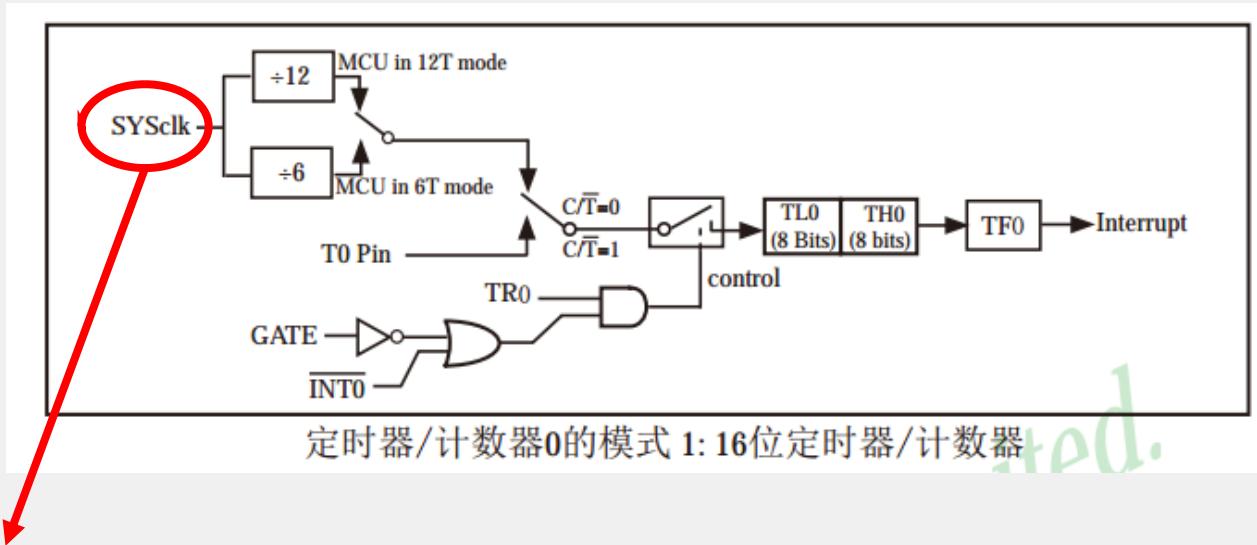


定时器工作模式

- STC89C52的T0和T1均有四种工作模式：
 - 模式0：13位定时器/计数器
 - 模式1：16位定时器/计数器（常用）
 - 模式2：8位自动重装模式
 - 模式3：两个8位计数器
- 工作模式1框图：



定时器时钟



- SYSclk : 系统时钟，即晶振周期，本开发板上的晶振为 12MHz

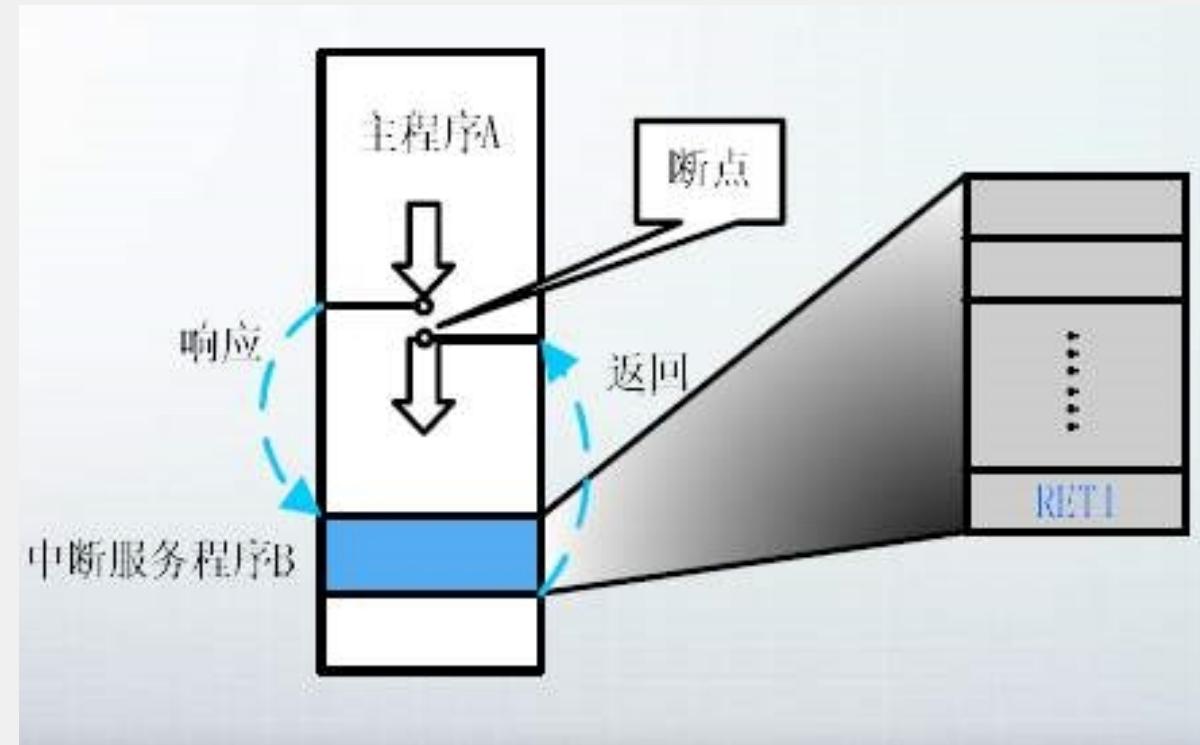
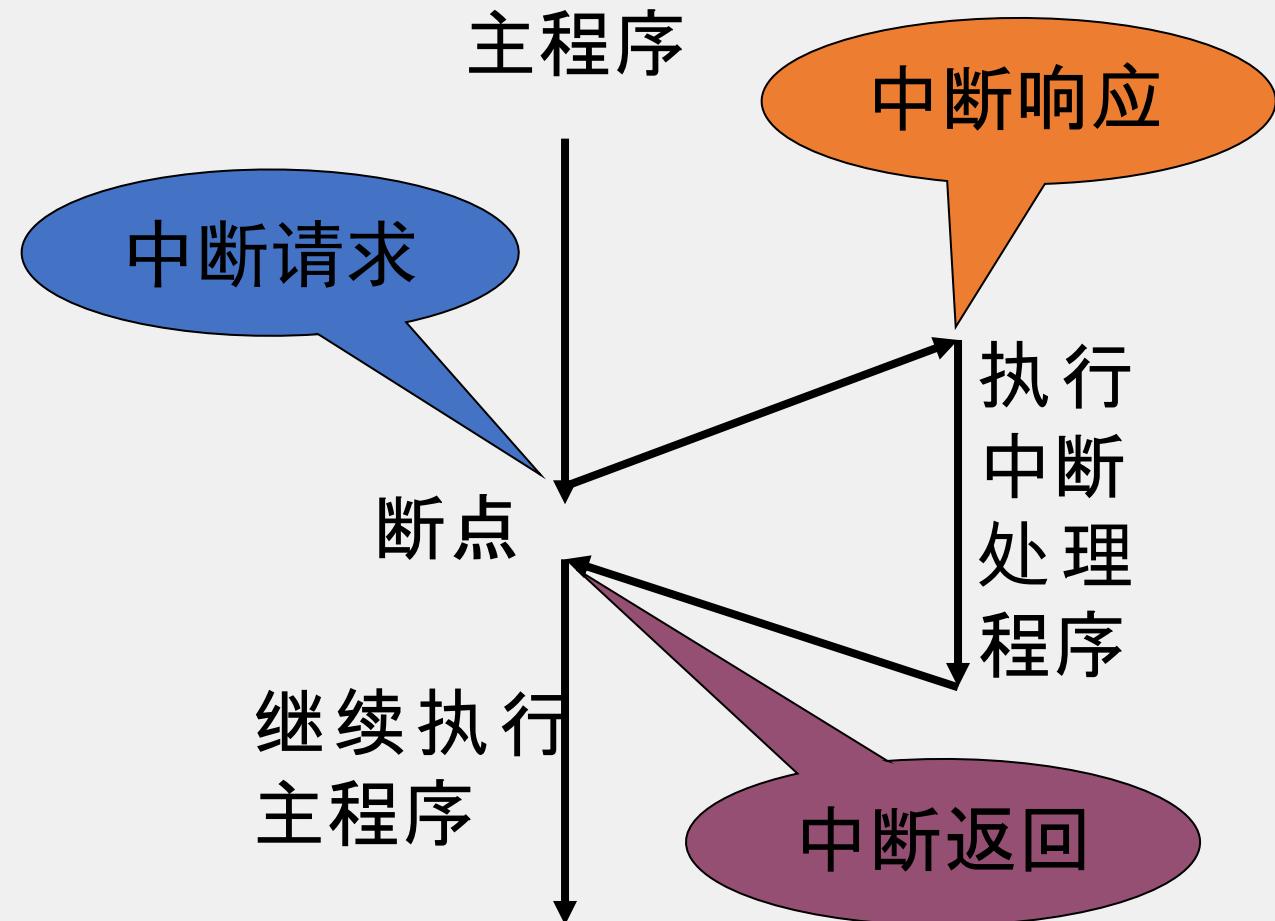
第6章 中断系统

中断系统是为使CPU具有对外界紧急事件的实时处理能力而设置的。

当中央处理机CPU正在处理某件事的时候外界发生了紧急事件请求，要求CPU暂停当前的工作，转而去处理这个紧急事件，处理完以后，再回到原来被中断的地方，继续原来的工作，这样的过程称为中断。实现这种功能的部件称为中断系统，请示CPU中断的请求源称为中断源。微型机的中断系统一般允许多个中断源，当几个中断源同时向CPU请求中断，要求为它服务的时候，这就存在CPU优先响应哪一个中断源请求的问题。通常根据中断源的轻重缓急排队，优先处理最紧急事件的中断请求源，即规定每一个中断源有一个优先级别。CPU总是先响应优先级别最高的中断请求。

当CPU正在处理一个中断源请求的时候（执行相应的中断服务程序），发生了另外一个优先级比它还高的中断源请求。如果CPU能够暂停对原来中断源的服务程序，转而去处理优先级更高的中断请求源，处理完以后，再回到原低级中断服务程序，这样的过程称为中断嵌套。这样的中断系统称为多级中断系统，没有中断嵌套功能的中断系统称为单级中断系统。

中断程序流程



STC89C52中断资源

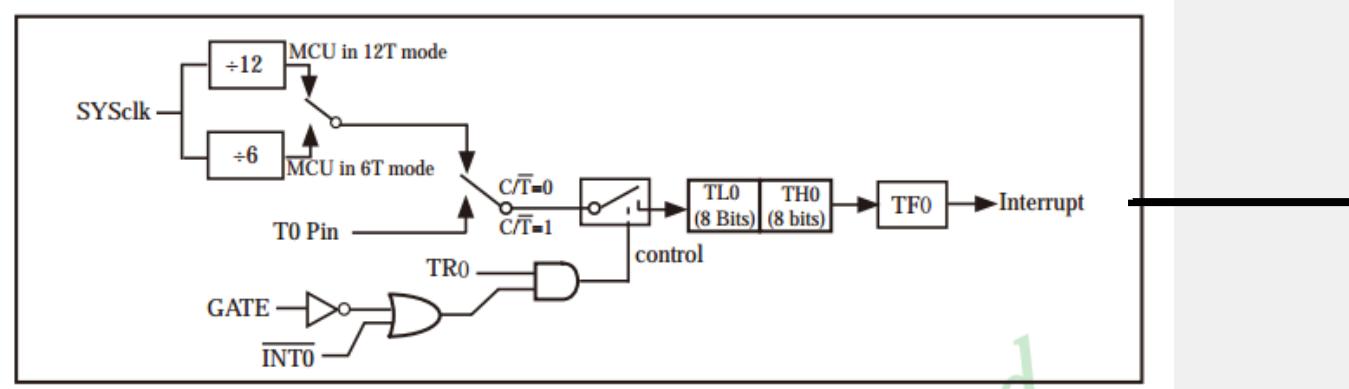
- 中断源个数：8个（外部中断0、定时器0中断、外部中断1、定时器1中断、串口中断、定时器2中断、外部中断2、外部中断3）
- 中断优先级个数：4个
- 中断号：

如果使用C语言编程，中断查询次序号就是中断号，例如：

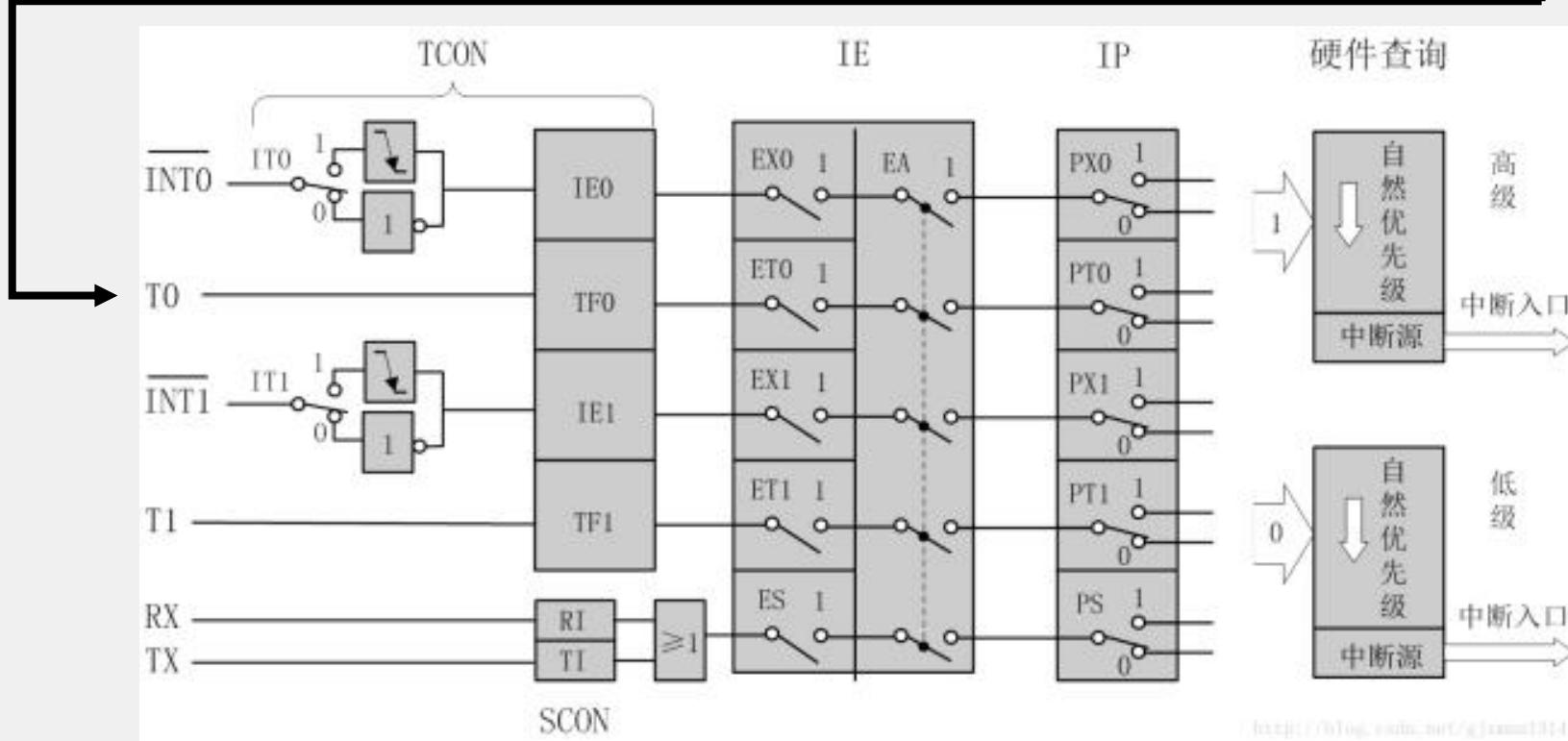
```
void    Int0_Routine(void)           interrupt 0;  
void    Timer0_Routine(void)         interrupt 1;  
void    Int1_Routine(void)           interrupt 2;  
void    Timer1_Routine(void)         interrupt 3;  
void    UART_Routine(void)          interrupt 4;  
void    Timer2_Routine(void)         interrupt 5;  
void    Int2_Routine(void)           interrupt 6;  
void    Int3_Routine(void)           interrupt 7;
```

- 注意：中断的资源和单片机的型号是关联在一起的，不同的型号可能会有不同的中断资源，例如中断源个数不同、中断优先级个数等等

定时器和中断系统



定时器/计数器0的模式1: 16位定时器/计数器



为了方便讲解，这里使用的中断系统图是传统51单片机的图，STC89C52的中断系统图可参考手册

定时器相关寄存器

7.1.1 定时器/计数器0和1的相关寄存器

符号	描述	地址	位地址及其符号									复位值
			MSB				LSB					
TCON	定时器控制寄存器	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0		0000 0000B
TMOD	定时器模式寄存器	89H	GATE	C/T	M1	M0	GATE	C/T	M1	M0		0000 0000B
TL0	Timer Low 0	8AH										0000 0000B
TL1	Timer Low 1	8BH										0000 0000B
TH0	Timer High 0	8CH										0000 0000B
TH1	Timer High 1	8DH										0000 0000B

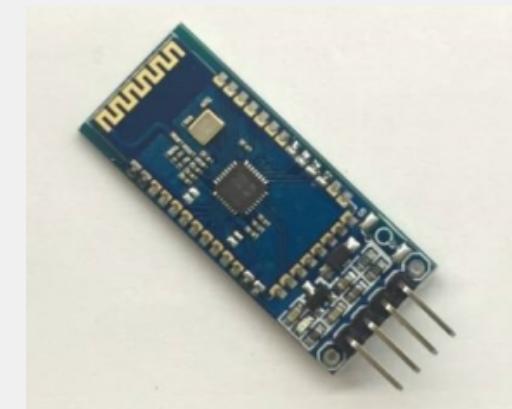
6.2 中断寄存器

符号	描述	地址	位地址及符号									复位值
			MSB				LSB					
IE	Interrupt Enable	A8H	EA	-	ET2	ES	ET1	EX1	ET0	EX0		0x00 0000B
IP	Interrupt Priority Low	B8H	-	-	PT2	PS	PT1	PX1	PT0	PX0		xx00 0000B
IPH	Interrupt Priority High	B7H	PX3H	PX2H	PT2H	PSH	PT1H	PX1H	PT0H	PX0H		0000,0000B
TCON	Timer/Counter 0 and 1 Control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0		0000 0000B
SCON	Serial Control	98H	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI		0000 0000B
T2CON	Timer/Counter 2 Control	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2		0000 0000B
XICON	Auxiliary Interrupt Control	C0H	PX3	EX3	IE3	IT3	PX2	EX2	IE2	IT2		0000 0000B

- 寄存器是连接软硬件的媒介
- 在单片机中寄存器就是一段特殊的RAM存储器，一方面，寄存器可以存储和读取数据，另一方面，每一个寄存器背后都连接了一根导线，控制着电路的连接方式
- 寄存器相当于一个复杂的“操作按钮”

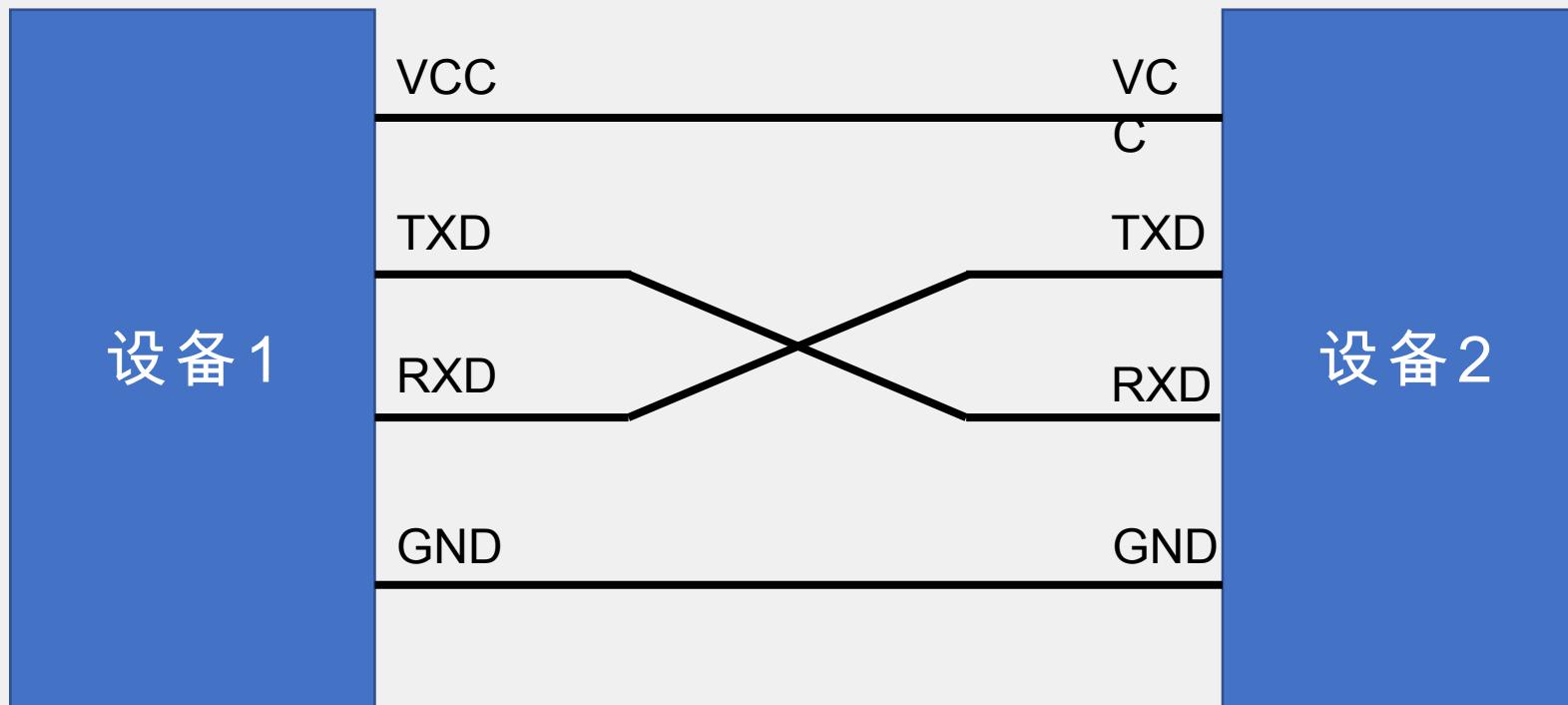
串口介绍

- 串口是一种应用十分广泛的通讯接口，串口成本低、容易使用、通信线路简单，可实现两个设备的互相通信。
- 单片机的串口可以使单片机与单片机、单片机与电脑、单片机与各式各样的模块互相通信，极大的扩展了单片机的应用范围，增强了单片机系统的硬件实力。
- 51单片机内部自带UART（Universal Asynchronous Receiver Transmitter，通用异步收发器），可实现单片机的串口通信。



硬件电路

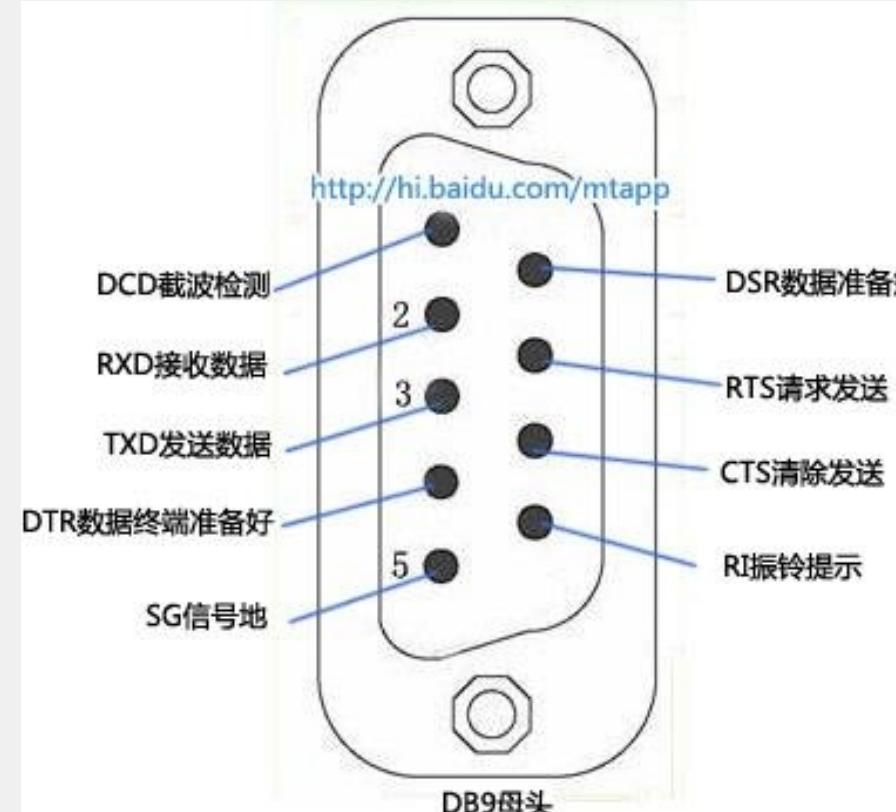
- 简单双向串口通信有两根通信线（发送端TXD和接收端RXD）
- TXD与RXD要交叉连接
- 当只需单向的数据传输时，可以直接一根通信线
- 当电平标准不一致时，需要加电平转换芯片



电平标准

- 电平标准是数据1和数据0的表达方式，是传输线缆中人为规定的电压与数据的对应关系，串口常用的电平标准有如下三种：
- TTL电平：+5V表示1，0V表示0
- RS232电平：-3~-15V表示1，+3~+15V表示0
- RS485电平：两线压差+2~+6V表示1，-2~-6V表示0（差分信号）

接口及引脚定义



常见通信接口比较

名称	引脚定义	通信方式	特点
UART	TXD、RXD	全双工、异步	点对点通信
I ² C	SCL、SDA	半双工、同步	可挂载多个设备
SPI	SCLK、MOSI、MISO、CS	全双工、同步	可挂载多个设备
1-Wire	DQ	半双工、异步	可挂载多个设备

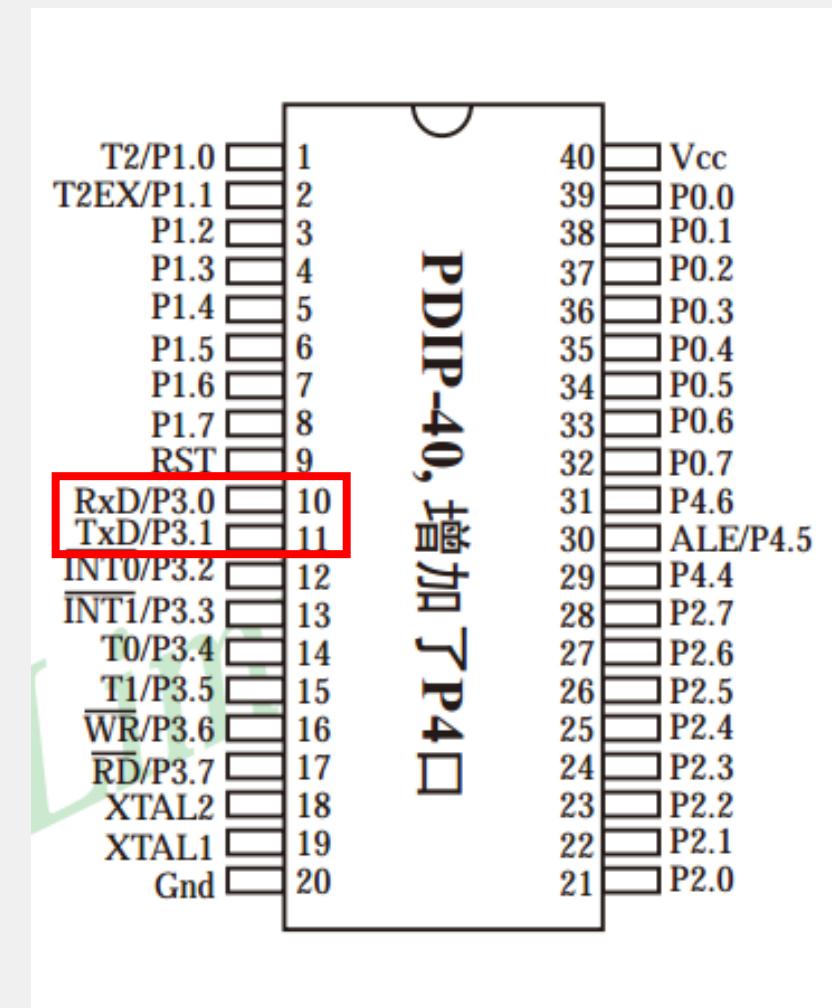
- 此外还有：CAN、USB等

相关术语

- 全双工：通信双方可以在同一时刻互相传输数据
- 半双工：通信双方可以互相传输数据，但必须分时复用一根数据线
- 单工：通信只能有一方发送到另一方，不能反向传输
- 异步：通信双方各自约定通信速率
- 同步：通信双方靠一根时钟线来约定通信速率
- 总线：连接各个设备的数据传输线路（类似于一条马路，把路边各住户连接起来，使住户可以相互交流）

51单片机的UART

- STC89C52有1个UART
- STC89C52的UART有四种工作模式：
 - 模式0：同步移位寄存器
 - 模式1：8位UART，波特率可变（常用）
 - 模式2：9位UART，波特率固定
 - 模式3：9位UART，波特率可变



串口参数及时序图

- 波特率：串口通信的速率（发送和接收各数据位的间隔时间）
- 检验位：用于数据验证
- 停止位：用于数据帧间隔

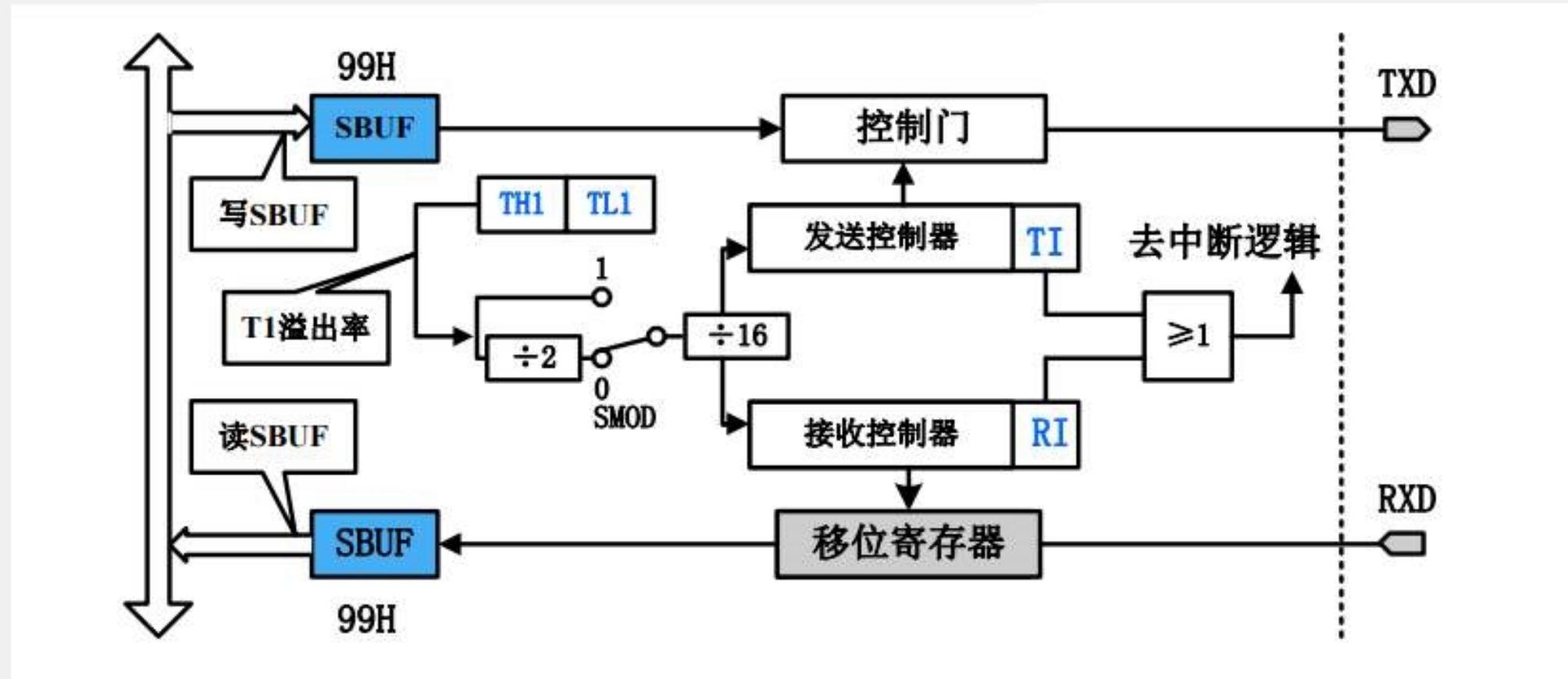


8位数据格式



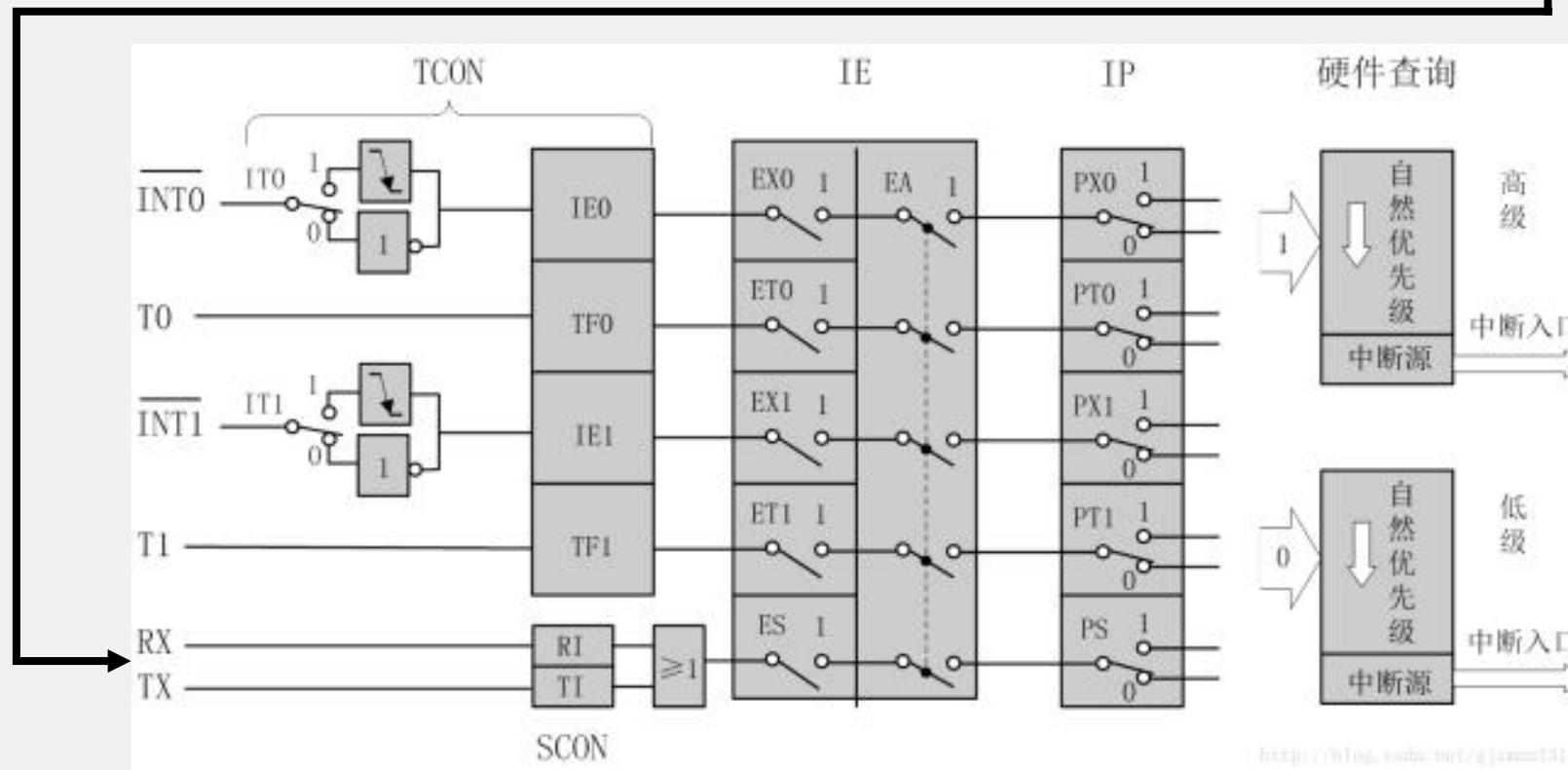
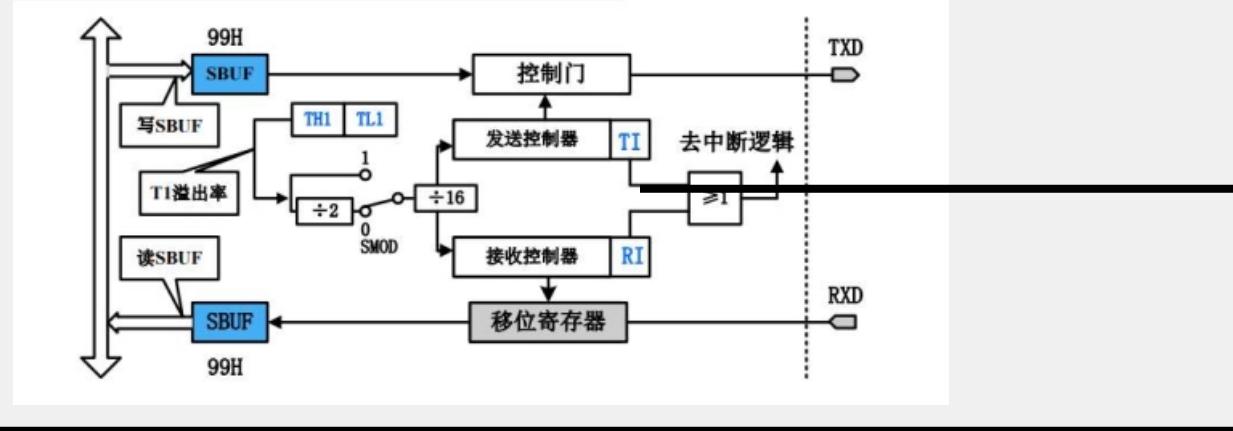
9位数据格式

串口模式图



- SBUF：串口数据缓存寄存器，物理上是两个独立的寄存器，但占用相同的地址。写操作时，写入的是发送寄存器，读操作时，读出的是接收寄存器

串口和中断系统



为了方便讲解，这里使用的中断系统图是传统51单片机的图，STC89C52的中断系统图可参考手册

串口相关寄存器

8.1 串行口相关寄存器

数据显示模式

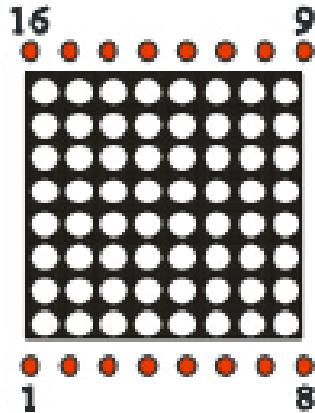
- HEX模式/十六进制模式/二进制模式：以原始数据的形式显示
- 文本模式/字符模式：以原始数据编码后的形式显示

LED点阵屏介绍

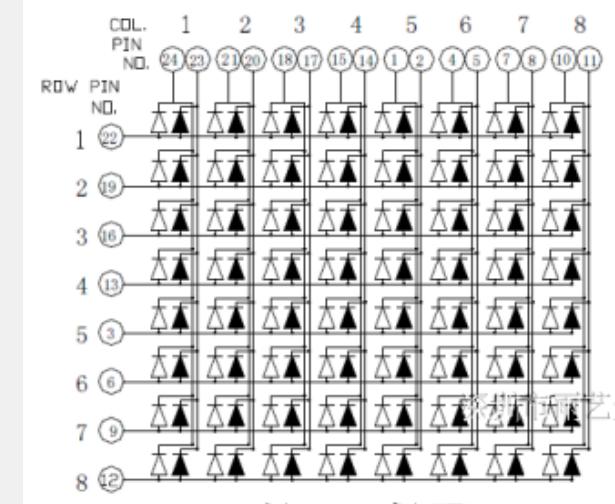
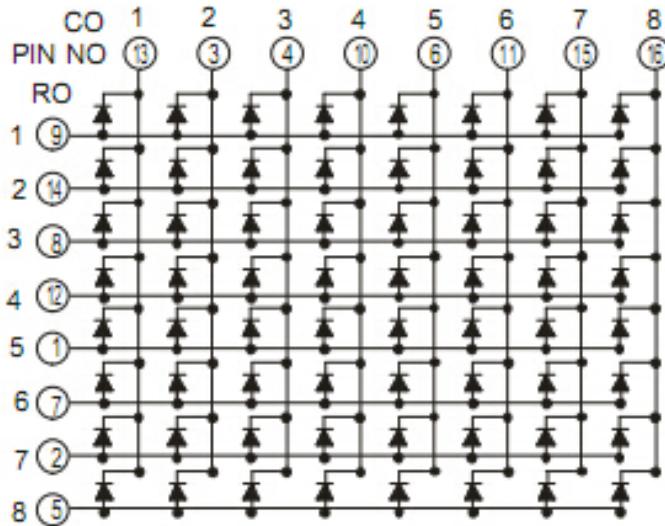
- LED点阵屏由若干个独立的LED组成，LED以矩阵的形式排列，以灯珠亮灭来显示文字、图片、视频等。LED点阵屏广泛应用于各种公共场合，如汽车报站器、广告屏以及公告牌等
- LED点阵屏分类
 - 按颜色：单色、双色、全彩
 - 按像素：8*8、16*16等（大规模的LED点阵通常由很多个小点阵拼接而成）



显示原理



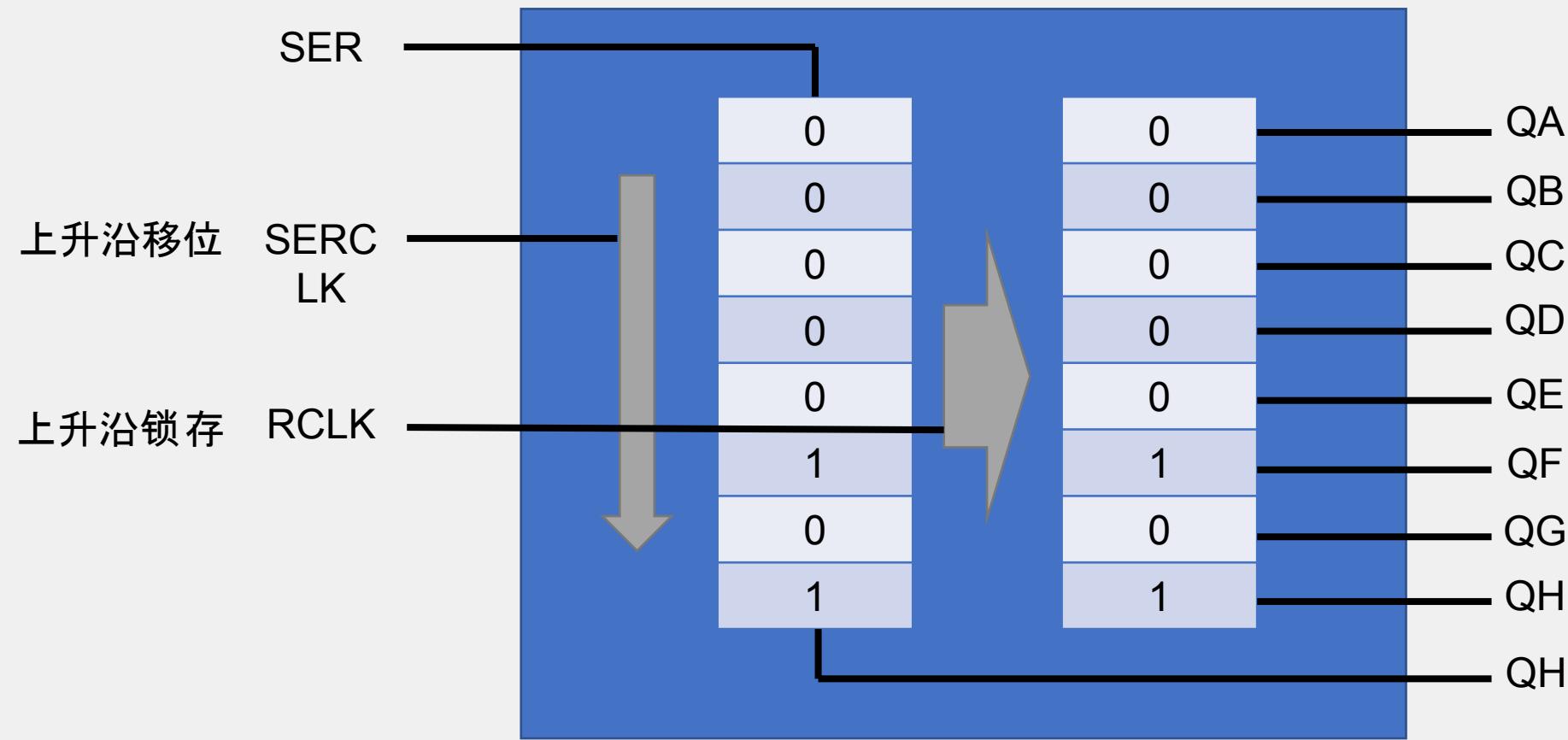
数码管显示及引脚位置图



- LED点阵屏的结构类似于数码管，只不过是数码管把每一列的像素以“8”字型排列而已
- LED点阵屏与数码管一样，有共阴和共阳两种接法，不同的接法对应的电路结构不同
- LED点阵屏需要进行逐行或逐列扫描，才能使所有LED同时显示

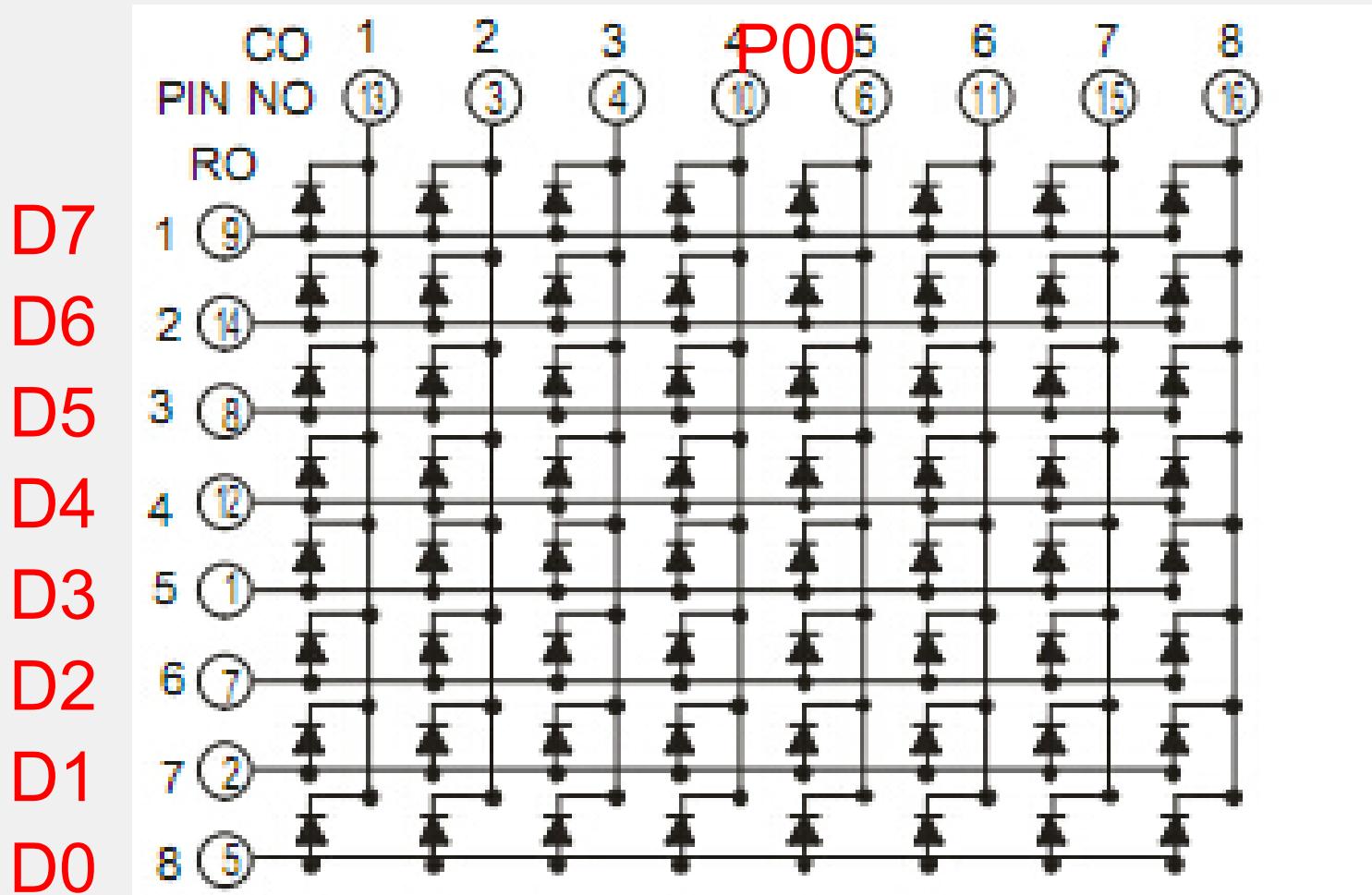
74HC595

- 74HC595是串行输入并行输出的移位寄存器，可用3根线输入串行数据，8根线输出并行数据，多片级联后，可输出16位、24位、32位等，常用于IO口扩展。



开发板引脚对应关系

P07 P06 P05 P04 P03 P02 P01



C51的sfr、sbit

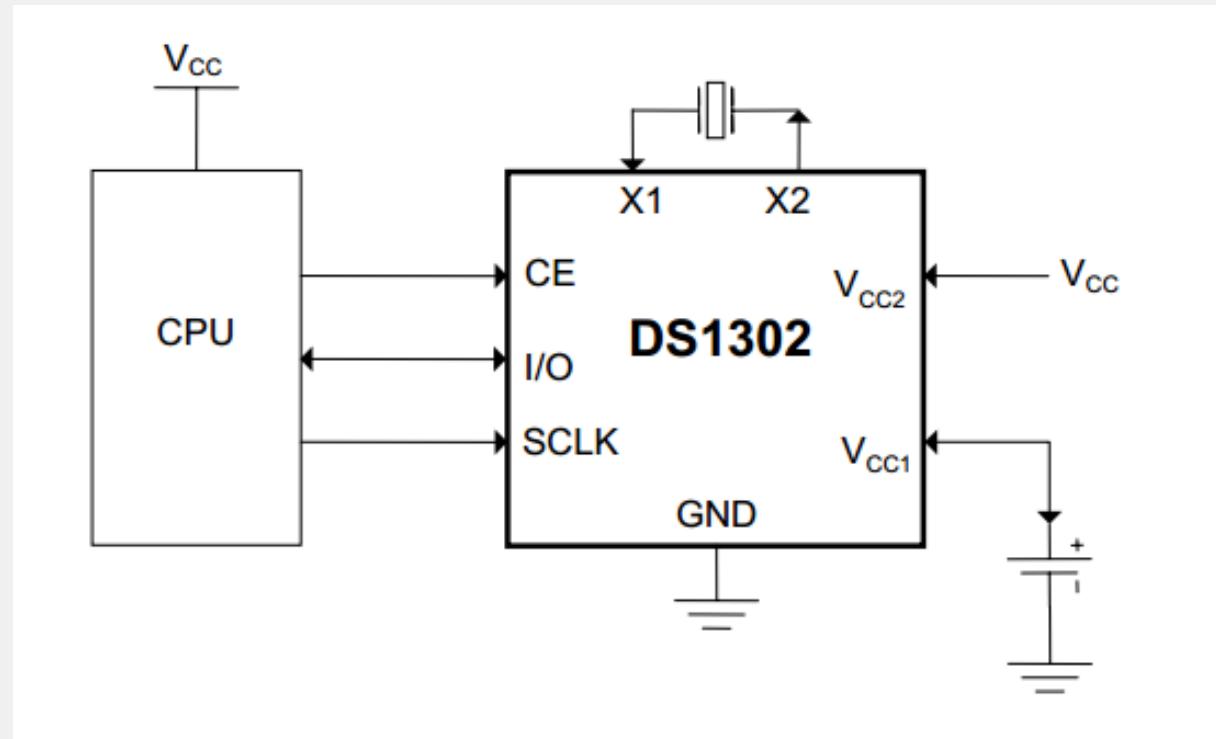
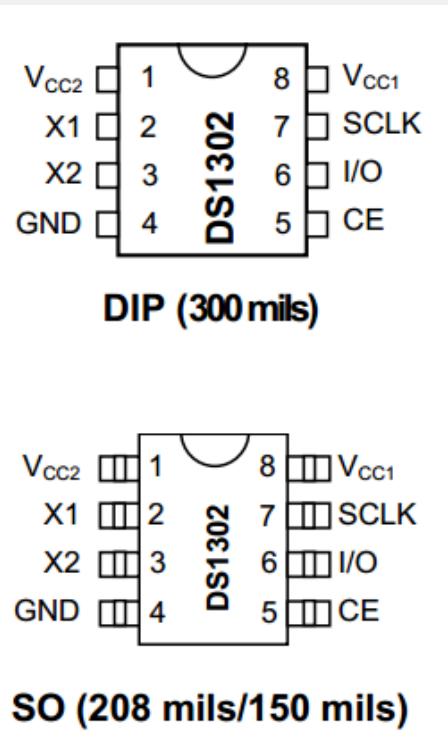
- sfr (special function register) : 特殊功能寄存器声明
例 : sfr P0 = 0x80;
 声明P0口寄存器，物理地址为 0x80
- sbit (special bit) : 特殊位声明
例 : sbit P0_1 = 0x81; 或 sbit P0_1 = P0^1;
 声明P0寄存器的第1位
- 可位寻址/不可位寻址 : 在单片机系统中，操作任意寄存器或者某一位的数据时，必须给出其物理地址，又因为一个寄存器里有8位，所以位的数量是寄存器数量的8倍，单片机无法对所有位进行编码，故每8个寄存器中，只有一个是可以位寻址的。对不可位寻址的寄存器，若要只操作其中一位而不影响其它位时，可用“&=”、“|=”、“^=”的方法进行位操作

DS1302介绍

- DS1302是由美国DALLAS公司推出的具有涓细电流充电能力的低功耗实时时钟芯片。它可以对年、月、日、周、时、分、秒进行计时，且具有闰年补偿等多种功能
- RTC(Real Time Clock)：实时时钟，是一种集成电路，通常称为时钟芯片

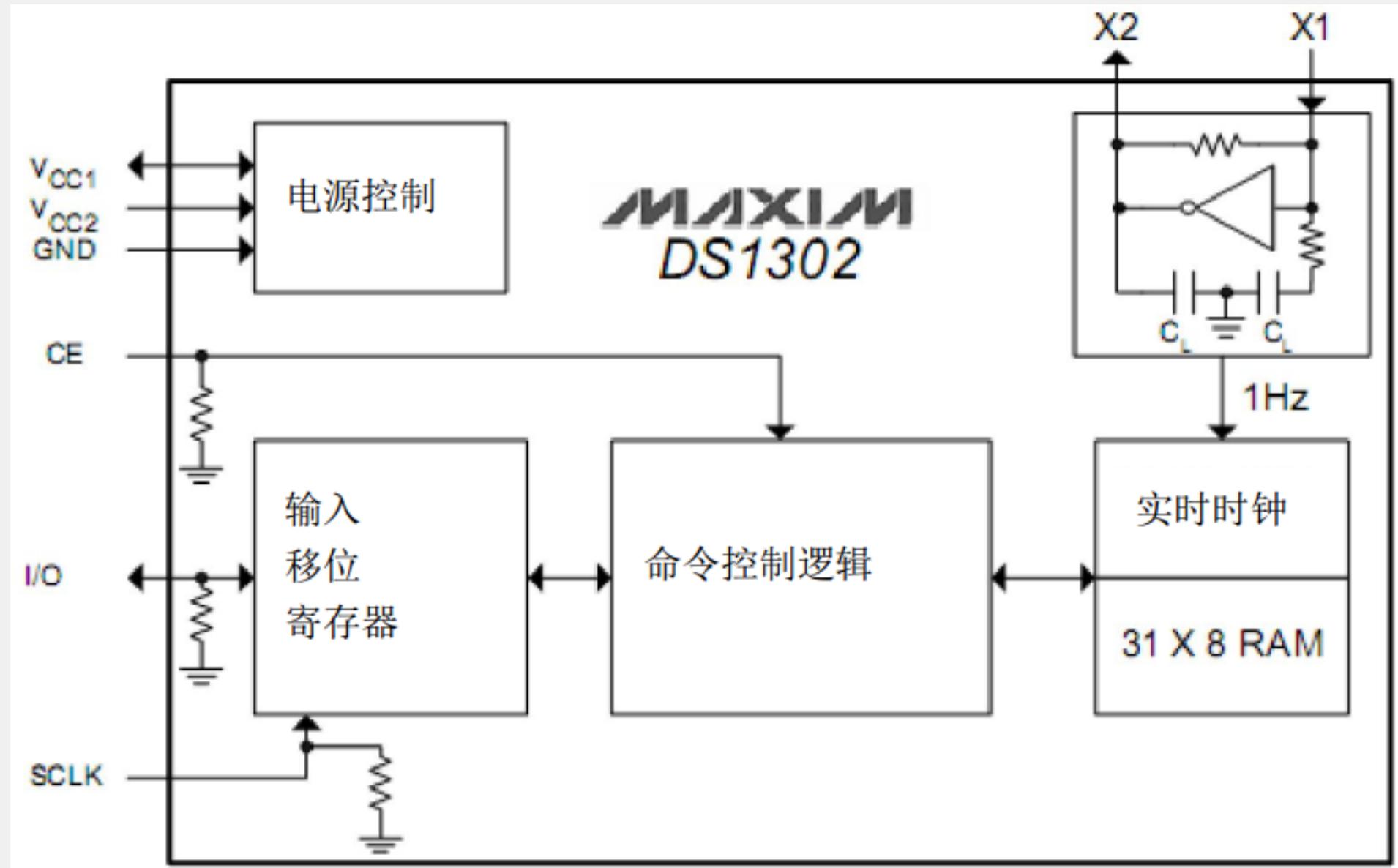


引脚定义和应用电路



引脚名	作用	引脚名	作用
VCC2	主电源	CE	芯片使能
VCC1	备用电池	I/O	数据输入/输出
GND	电源地	SCLK	串行时钟
X1、X2	32.768KHz晶振		

内部结构框图

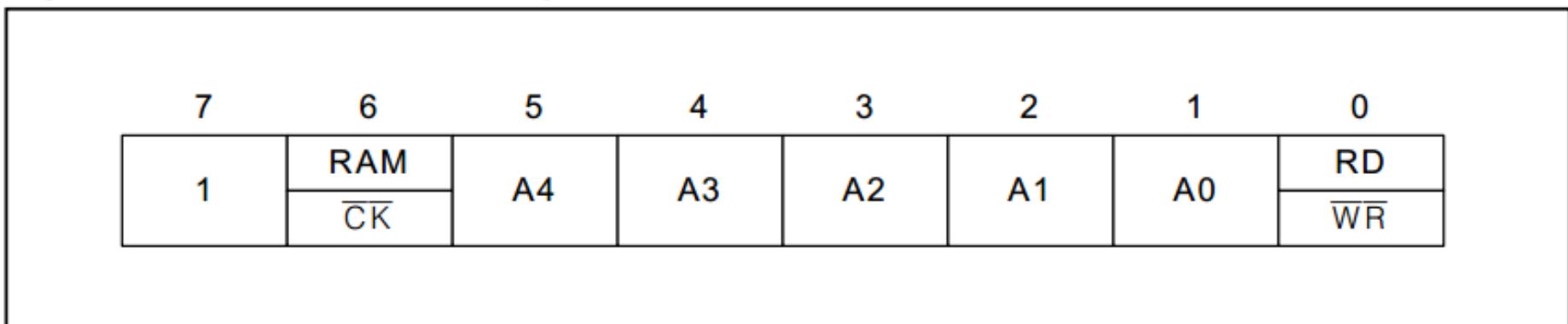


寄存器定义

RTC

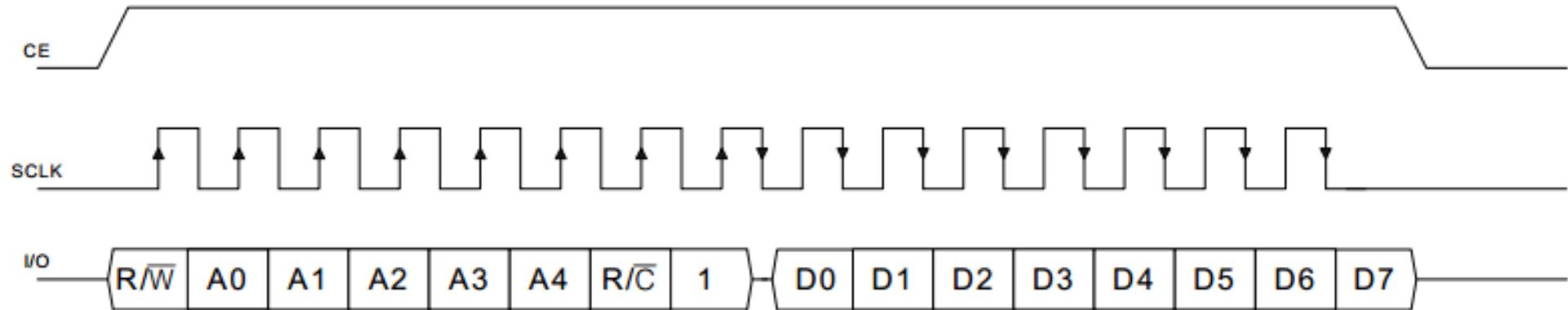
READ	WRITE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	RANGE
81h	80h	CH	10 Seconds			Seconds			00–59	
83h	82h		10 Minutes			Minutes			00–59	
85h	84h	12/24	0	10	Hour	Hour			1–12/0–23	
				AM/PM						
87h	86h	0	0	10 Date		Date			1–31	
89h	88h	0	0	0	10	Month			1–12	
8Bh	8Ah	0	0	0	0	0	Day		1–7	
8Dh	8Ch	10 Year			Year			00–99		
8Fh	8Eh	WP	0	0	0	0	0	0	0	—
91h	90h	TCS	TCS	TCS	TCS	DS	DS	RS	RS	—

Figure 3. Address/Command Byte

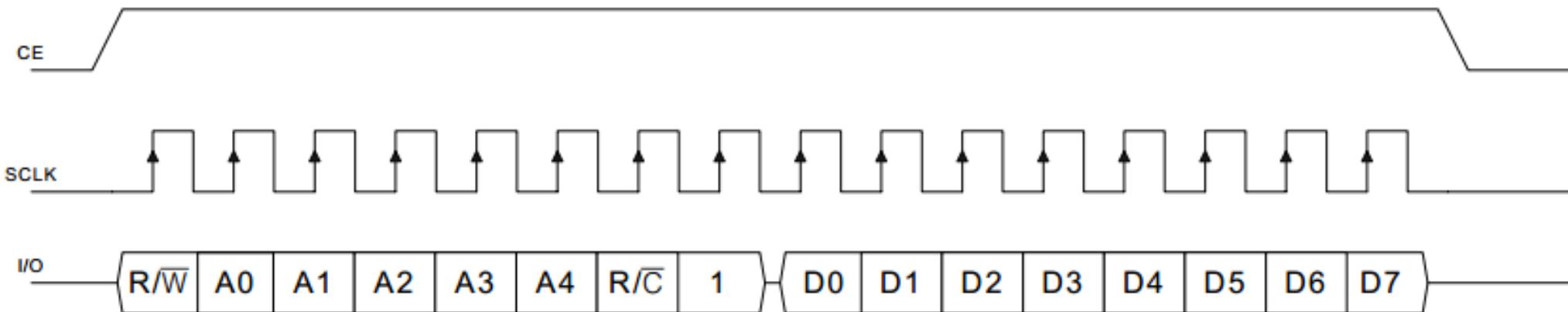


时序定义

SINGLE-BYTE READ



SINGLE-BYTE WRITE

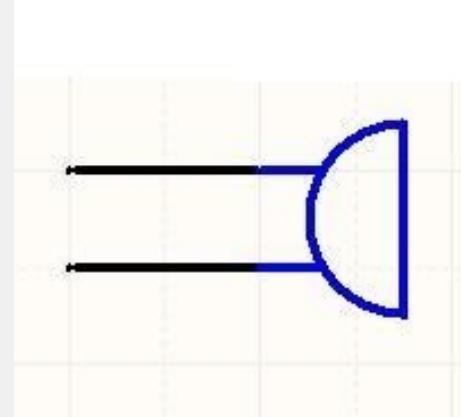


BCD码

- BCD码 (Binary Coded Decimal) , 用4位二进制数来表示1位十进制数
- 例 : 0001 0011表示13, 1000 0101表示85, 0001 1010不合法
- 在十六进制中的体现 : 0x13表示13, 0x85表示85, 0x1A不合法
- BCD码转十进制 : $DEC=BCD/16*10+BCD\%16;$ (2位BCD)
- 十进制转BCD码 : $BCD=DEC/10*16+DEC\%10;$ (2位BCD)

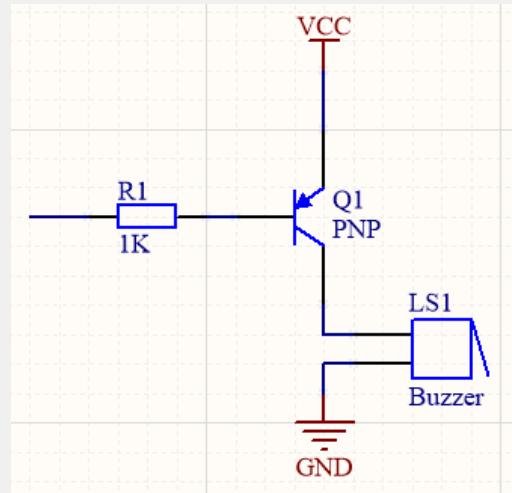
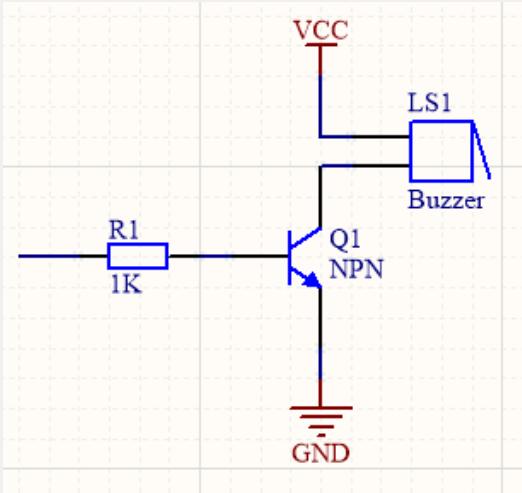
蜂鸣器介绍

- 蜂鸣器是一种将电信号转换为声音信号的器件，常用来产生设备的按键音、报警音等提示信号
- 蜂鸣器按驱动方式可分为有源蜂鸣器和无源蜂鸣器
- 有源蜂鸣器：内部自带振荡源，将正负极接上直流电压即可持续发声，频率固定
- 无源蜂鸣器：内部不带振荡源，需要控制器提供振荡脉冲才可发声，调整提供振荡脉冲的频率，可发出不同频率的声音

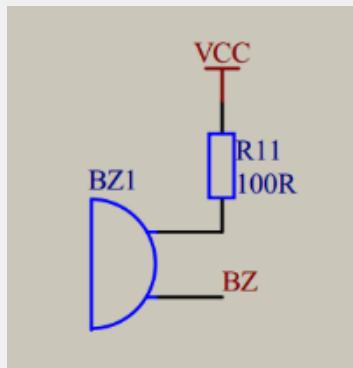
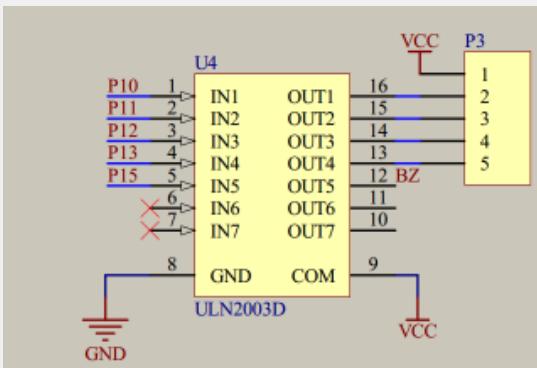


驱动电路

- 三极管驱动



- 集成电路驱动



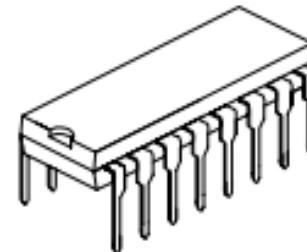
达林顿晶体管阵列

ULN2003是一个单片高电压、高电流的达林顿晶体管阵列集成电路。它是由**7对NPN**达林顿管组成的，它的高电压输出特性和阴极箝位二极管可以转换感应负载。单个达林顿对的集电极电流是**500mA**。达林顿管并联可以承受更大的电流。此电路主要应用于继电器驱动器，字锤驱动器，灯驱动器，显示驱动器（**LED**气体放电），线路驱动器和逻辑缓冲器。

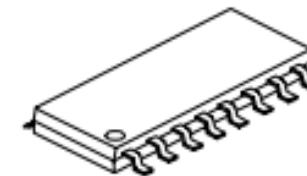
ULN2003的每对达林顿管都有一个**2.7kΩ**串联电阻，可以直接和**TTL**或**5V CMOS**装置。

主要特点

- * **500mA** 额定集电极电流（单个输出）
- * 高电压输出: **50V**
- * 输入和各种逻辑类型兼容
- * 继电器驱动器



DIP-16



SOP-16

键盘与音符对照

钢琴键盘与五线谱、简谱音高对照表

音名	C	D	E	F	G	A	B	c	d	e	f	g	a	b	c'	d'	e'	f'	g'	a'	b'	c^2	d^2	e^2	f^2	g^2	a^2	b^2	c^3
五线谱	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do
简谱	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1
唱名	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do	re	mi	fa	sol	la	si	do

音符与频率对照

C调音符与频率对照表

音符	频率/Hz	音符	频率/Hz	音符	频率/Hz
低音1	262	中音1	523	高音1	1046
低音1#	277	中音1#	554	高音1#	1109
低音2	294	中音2	587	高音2	1175
低音2#	311	中音2#	622	高音2#	1245
低音3	330	中音3	659	高音3	1318
低音4	349	中音4	698	高音4	1397
低音4#	370	中音4#	740	高音4#	1480
低音5	392	中音5	784	高音5	1568
低音5#	415	中音5#	831	高音5#	1661
低音6	440	中音6	880	高音6	1760
低音6#	466	中音6#	932	高音6#	1865
低音7	494	中音7	988	高音7	1976

小星星

1=C $\frac{4}{4}$

佚名 词曲

1 1 5 5 | 6 6 5 - | 4 4 3 3 | 2 2 1 - | 5 5 4 4 | 3 3 2 - |
一 闪 一 闪 亮 晶 晶， 满 天 都 是 小 星 星。 挂 在 天 上 放 光 明，

5 5 4 4 | 3 3 2 - | 1 1 5 5 | 6 6 5 - | 4 4 3 3 | 2 2 1 - |
好 像 许 多 小 眼 睛。 一 闪 一 闪 亮 晶 晶， 满 天 都 是 小 星 星。

简谱

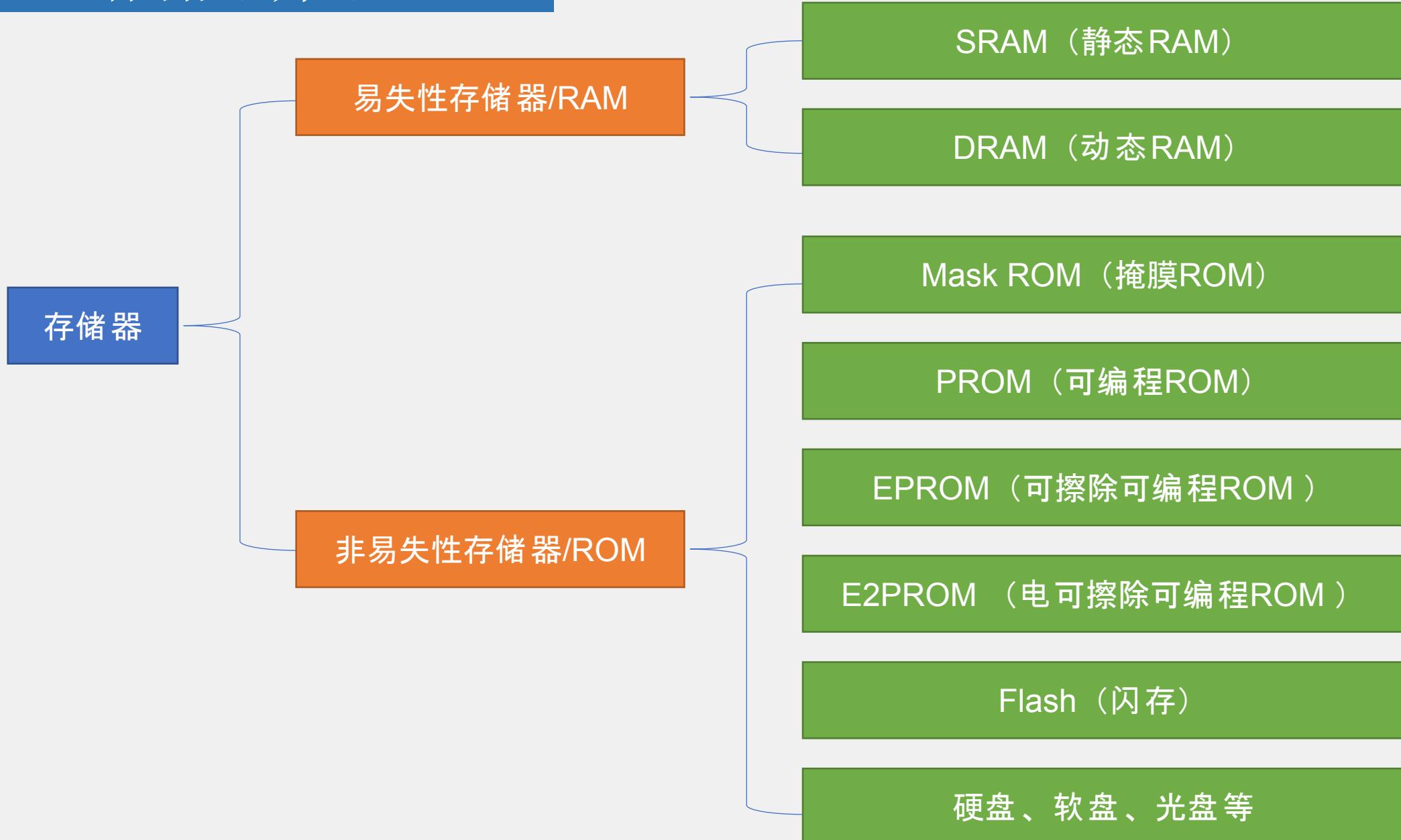
天空之城

1=D $\frac{4}{4}$

质谱：赵海洋

0 0 0 6 7 | i. 7 i 3 | 7 - - 3 3 |
6. 5 6 i | 5 - - 3 | 4. 3 4 i |
3 - 0 i i i | 7. 4 4 7 | 7 - 0 6 7 |
i. 7 i 3 | 7 - - 3 3 | 6. 5 6 i |
5 - - 2 3 | 4 i 7 7 i i | 2 2 3 i i - |
i 7 6 6 7 5 | 6 - - i 2 | 3. 2 3 5 |
2 - - 5 5 | i. 7 i 3 | 3 - - - |
6 7 i 7 2 2 | i. 5 5 - | 4 3 2 i |
3 - - 3 | 6 - 5 5 | 3 2 i - 0 i |
2 i 2 2 5 | 3 - - 3 | 6 - 5 - |
3 2 i - 0 i | 2 i 2 2 7 | 6 - - 6 7 :|

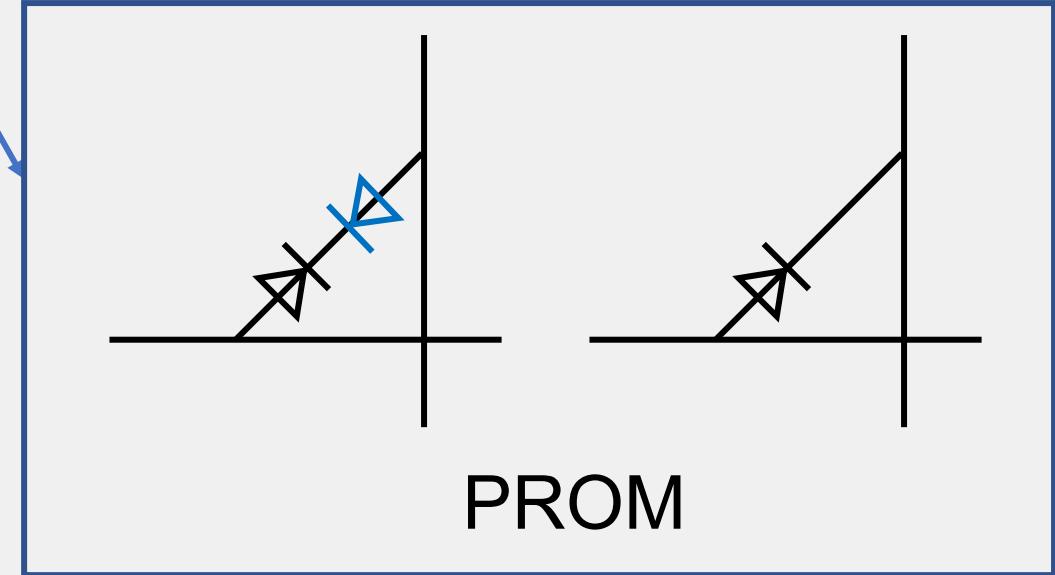
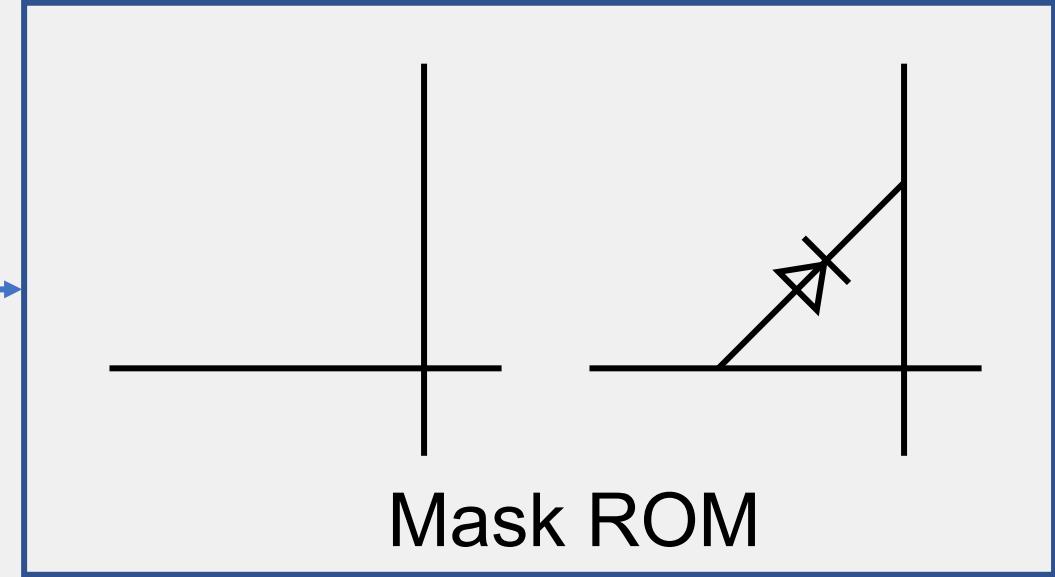
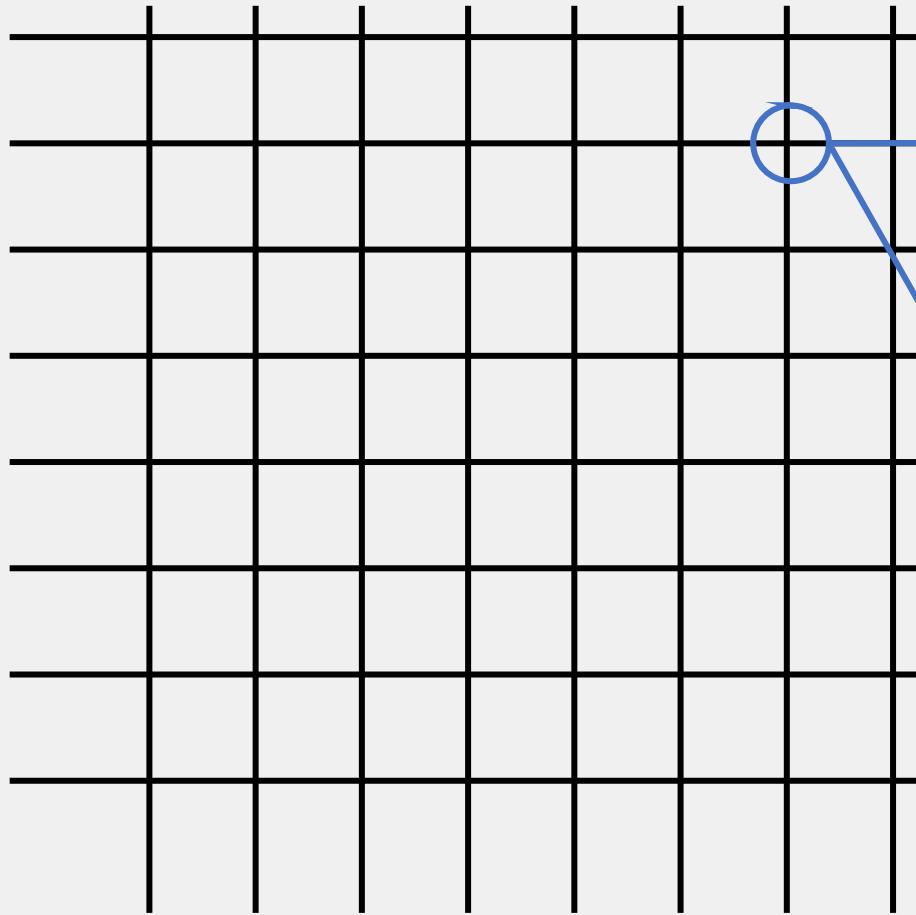
存储器介绍



存储器简化模型

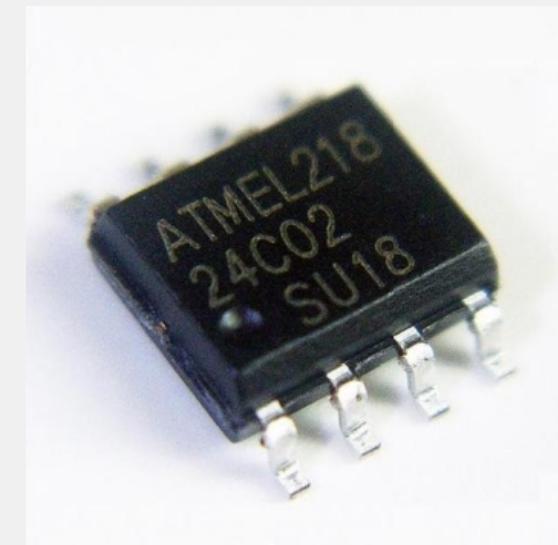
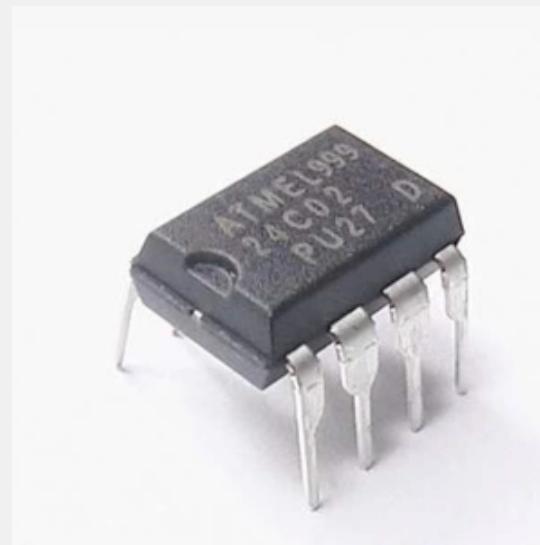
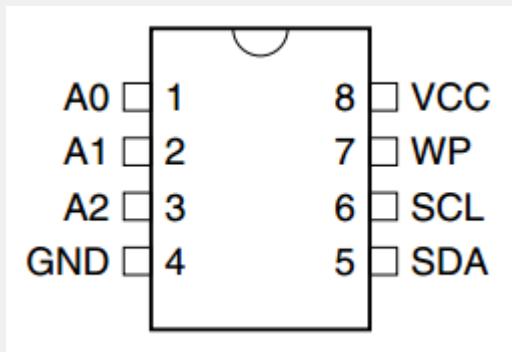
地址
总线

数据总线



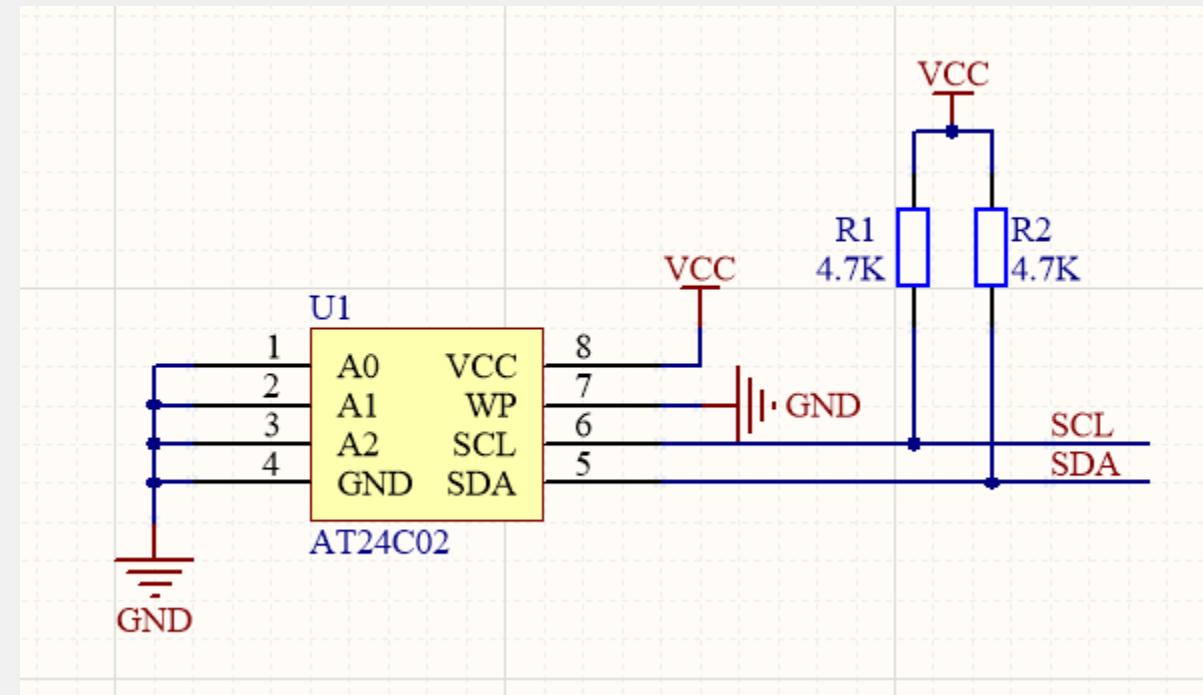
AT24C02介绍

- AT24C02是一种可以实现掉电不丢失的存储器，可用于保存单片机运行时想要永久保存的数据信息
- 存储介质：E2PROM
- 通讯接口：I2C总线
- 容量：256字节

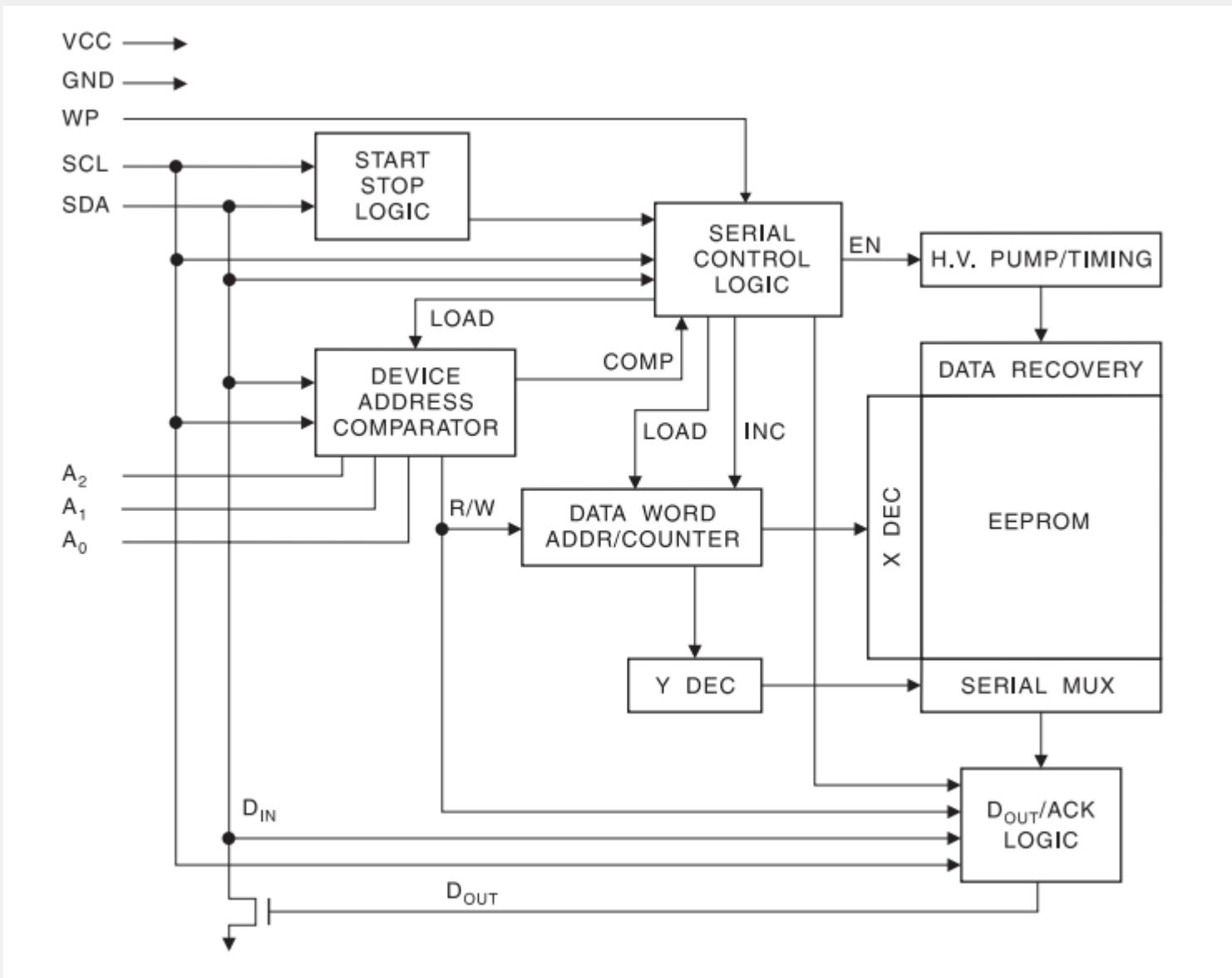


引脚及应用电路

引脚	功能
VCC、GND	电源 (1.8V~5.5V)
WP	写保护 (高电平有效)
SCL、SDA	I2C接口
A0、A1、A2	I2C地址

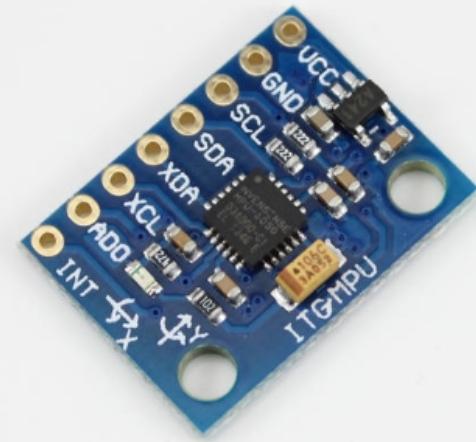


内部结构框图



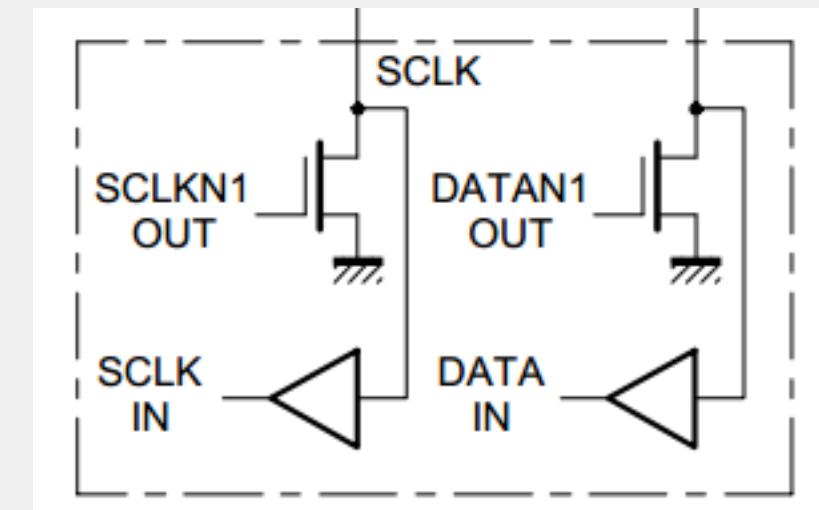
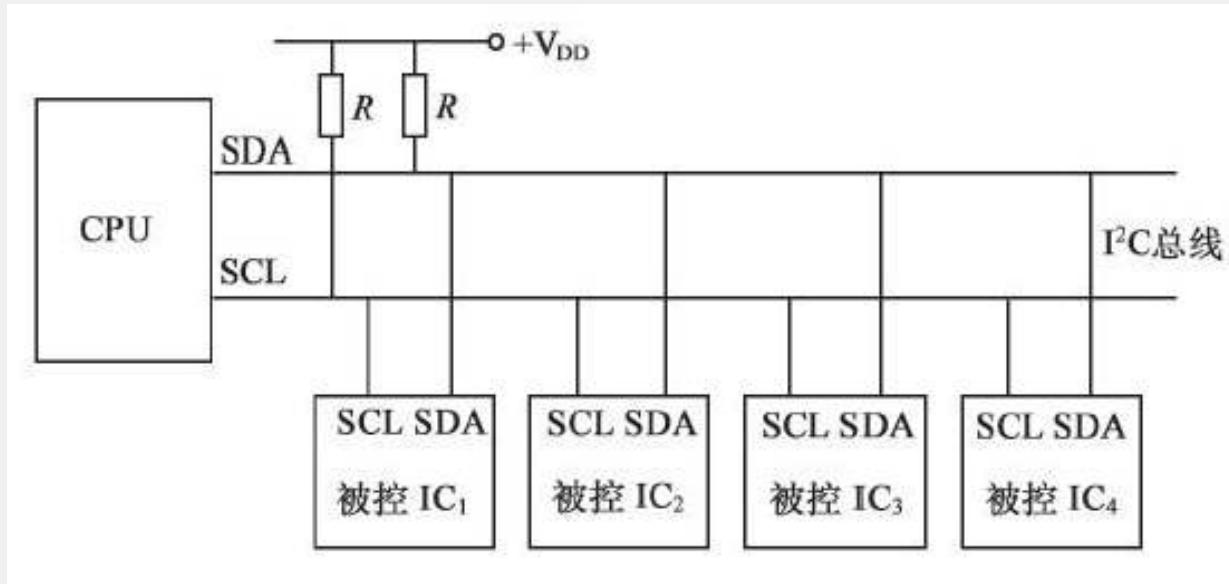
I2C总线介绍

- I2C总线（Inter IC BUS）是由Philips公司开发的一种通用数据总线
- 两根通信线：SCL（Serial Clock）、SDA（Serial Data）
- 同步、半双工，带数据应答
- 通用的I2C总线，可以使各种设备的通信标准统一，对于厂家来说，使用成熟的方法可以缩短芯片设计周期、提高稳定性，对于应用者来说，使用通用的通信协议可以避免学习各种各样的自定义协议，降低了学习和应用的难度



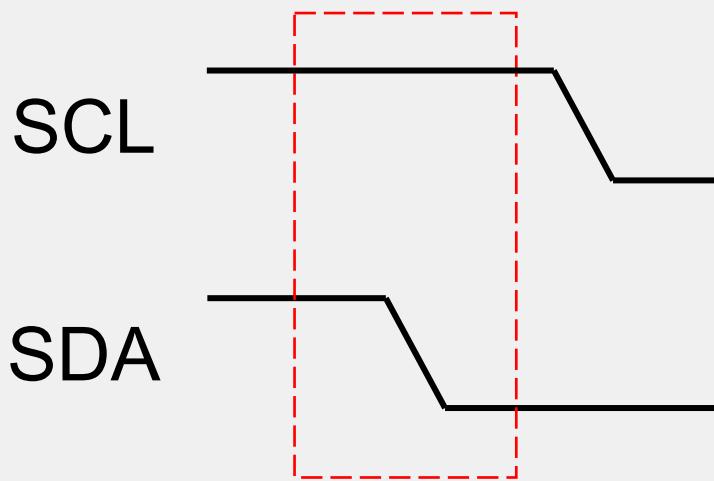
I2C电路规范

- 所有I2C设备的SCL连在一起，SDA连在一起
- 设备的SCL和SDA均要配置成开漏输出模式
- SCL和SDA各添加一个上拉电阻，阻值一般为 $4.7\text{K}\Omega$ 左右
- 开漏输出和上拉电阻的共同作用实现了“线与”的功能，此设计主要是为了解决多机通信互相干扰的问题

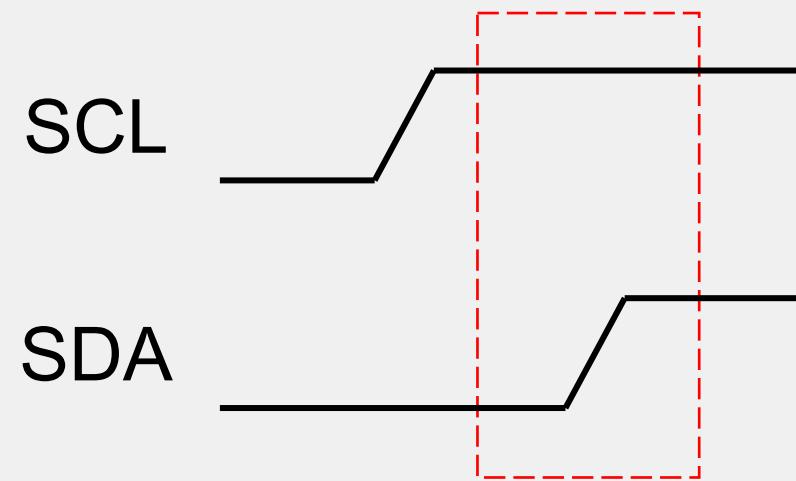


I2C时序结构

- 起始条件：SCL高电平期间，SDA从高电平切换到低电平
- 终止条件：SCL高电平期间，SDA从低电平切换到高电平



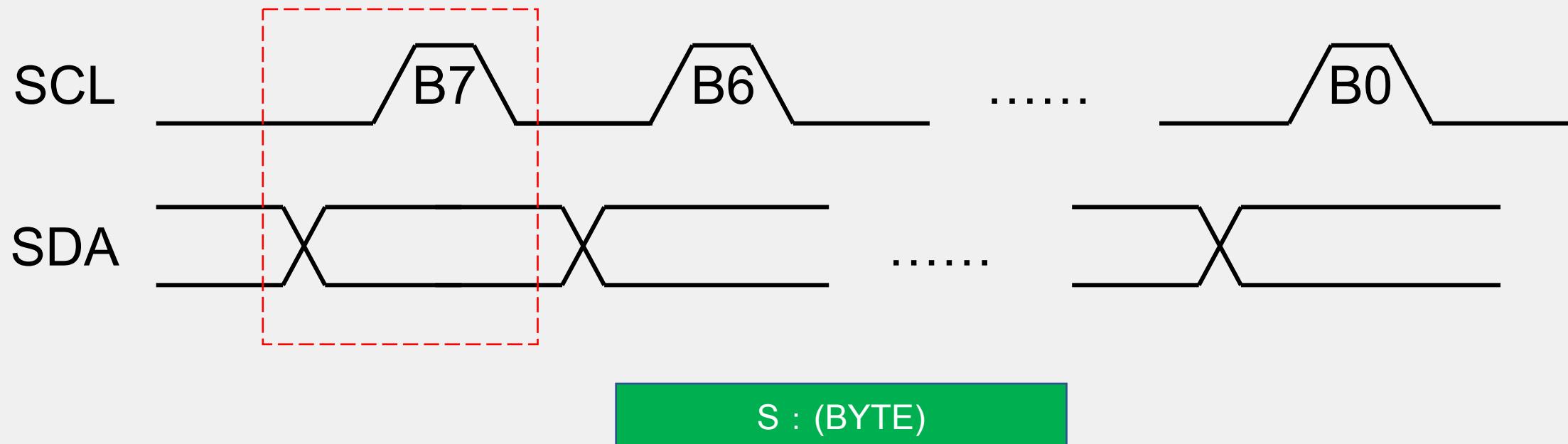
S



P

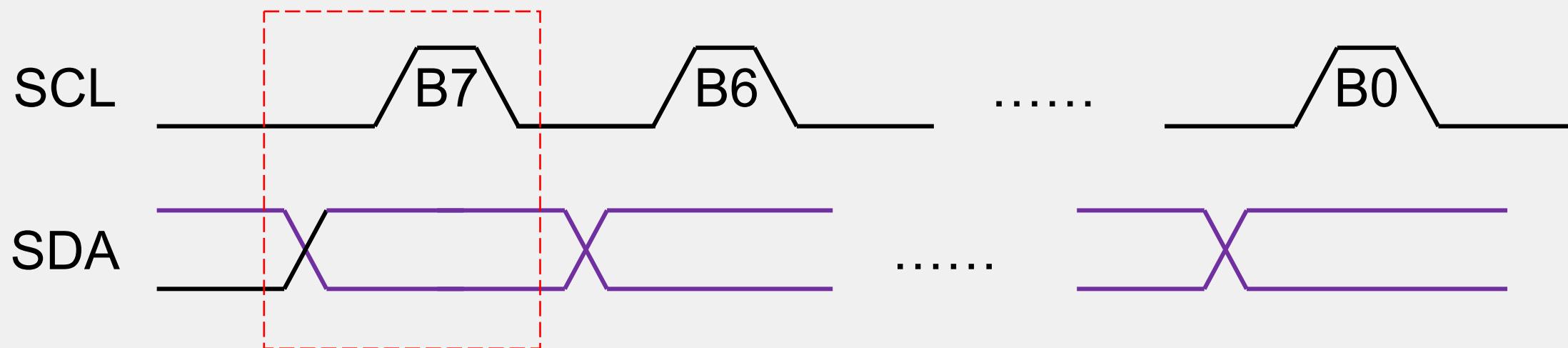
I2C时序结构

- 发送一个字节：SCL低电平期间，主机将数据位依次放到SDA线上（高位在前），然后拉高SCL，从机将在SCL高电期间读取数据位，所以SCL高电期间 SDA不允许有数据变化，依次循环上述过程8次，即可发送一个字节



I2C时序结构

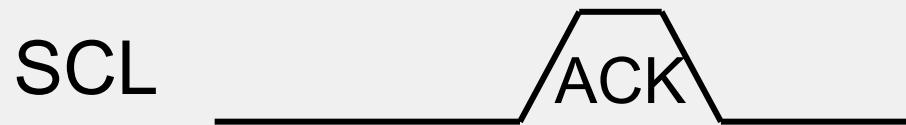
- 接收一个字节：SCL低电平期间，从机将数据位依次放到SDA线上（高位在前），然后拉高SCL，主机将在SCL高电期间读取数据位，所以SCL高电期间 SDA不允许有数据变化，依次循环上述过程8次，即可接收一个字节（主机在接收之前，需要释放SDA）



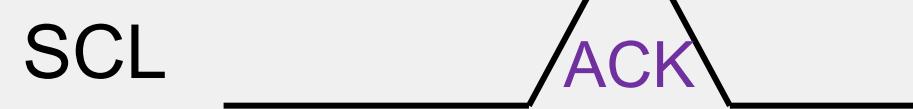
R : (BYTE)

I2C时序结构

- 发送应答：在接收完一个字节之后，主机在下一个时钟发送一位数据，数据0表示应答，数据1表示非应答
- 接收应答：在发送完一个字节之后，主机在下一个时钟接收一位数据，判断从机是否应答，数据0表示应答，数据1表示非应答（主机在接收之前，需要释放SDA）



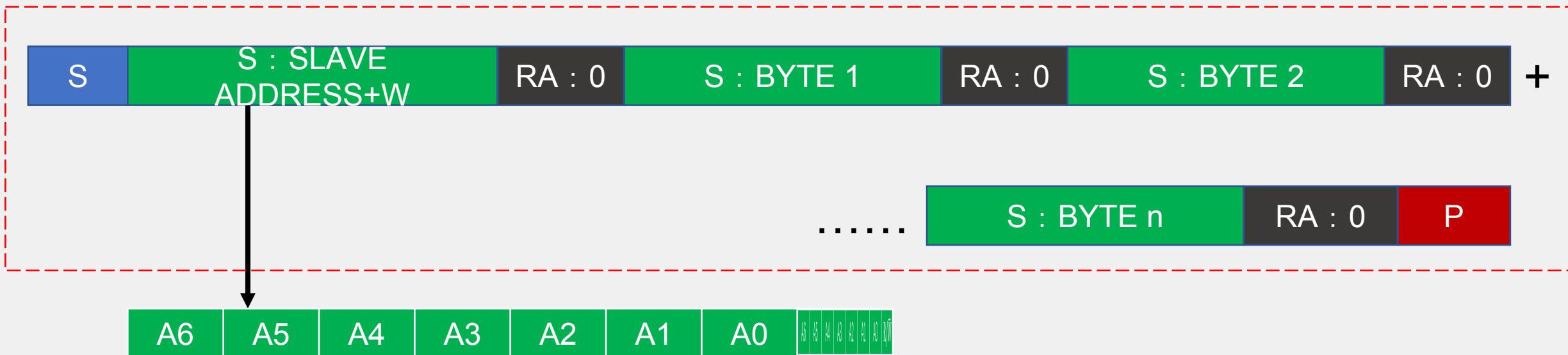
SA : (1/0)



RA : (1/0)

I2C数据帧

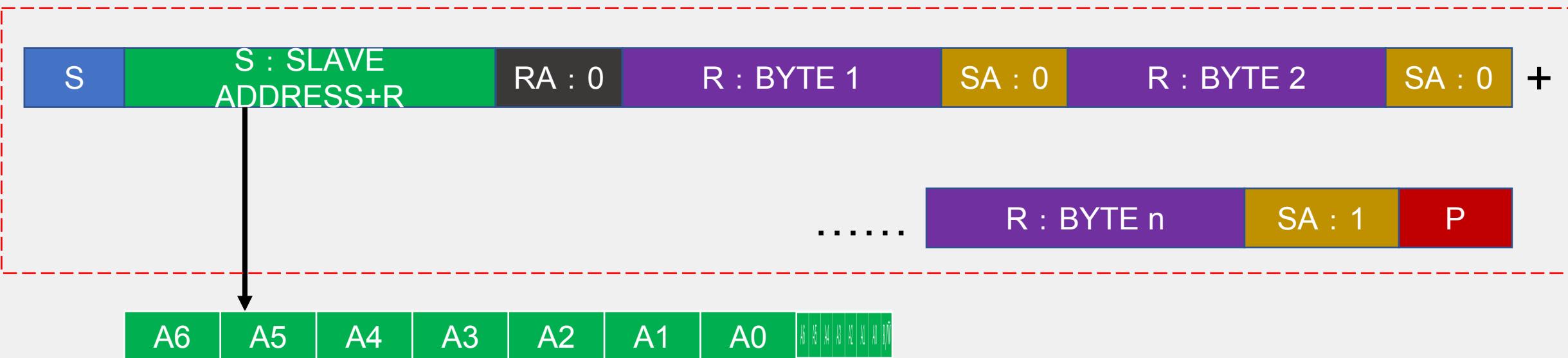
- 发送一帧数据



- 完成任务：向谁发什么

I2C数据帧

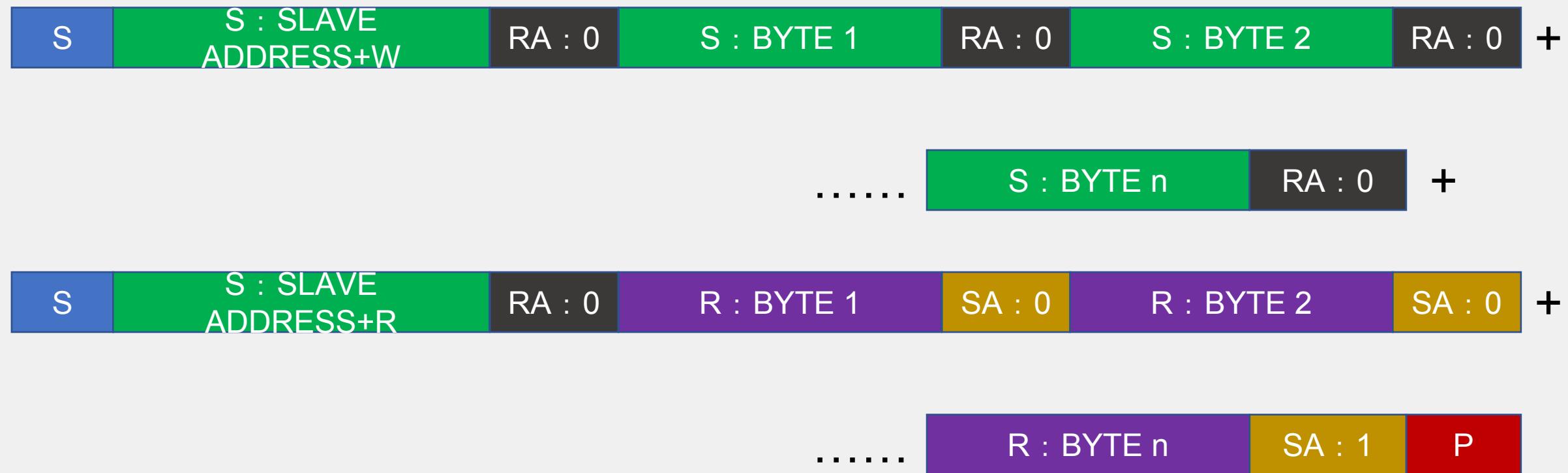
- 接收一帧数据



- 完成任务：向谁收什么

I2C数据帧

- 先发送再接收数据帧（复合格式）



- 完成任务：向谁收指定的什么

AT24C02数据帧

- 字节写：在WORD ADDRESS处写入数据DATA



- 随机读：读出在WORD ADDRESS处的数据DATA



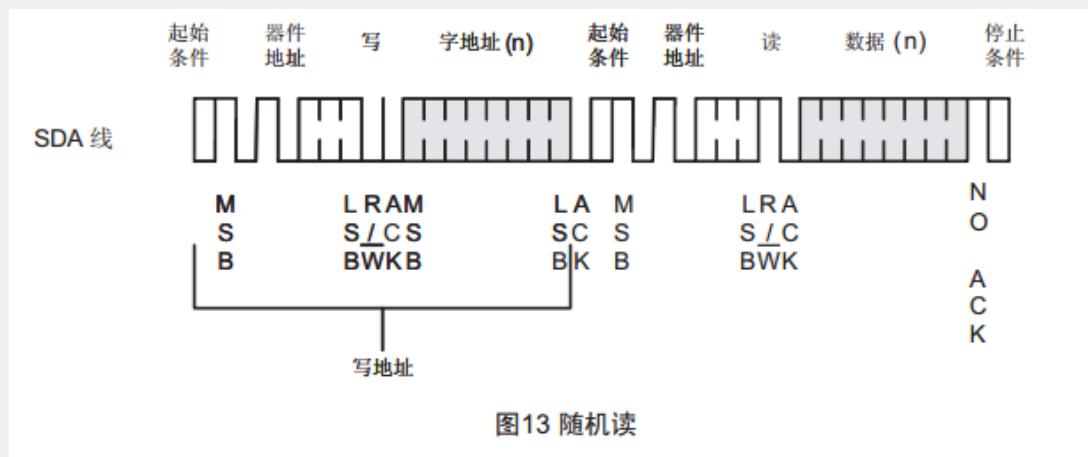
- AT24C02的固定地址为1010，可配置地址本开发板上为000
所以SLAVE ADDRESS+W为0xA0，SLAVE ADDRESS+R为0xA1

AT24C02数据帧

- 字节写：在“字地址”处写入“数据”

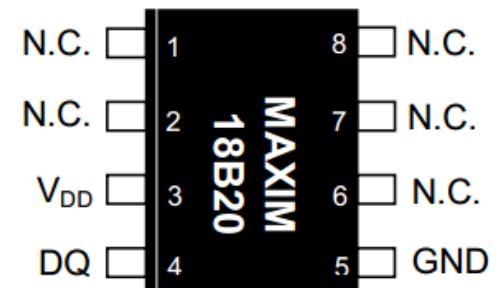
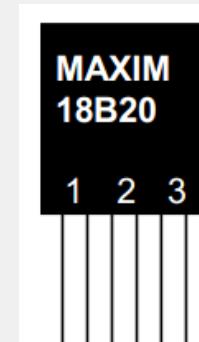
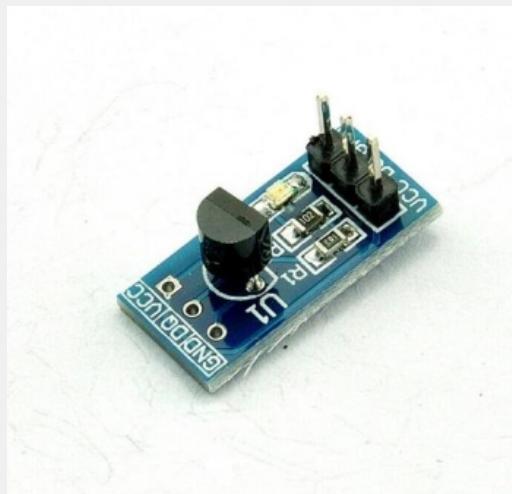
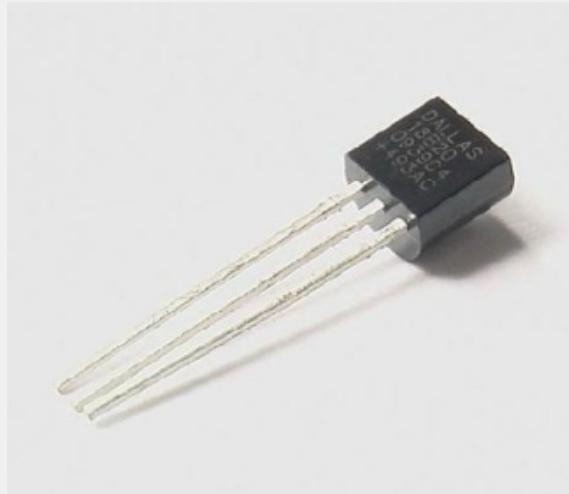


- 随机读：读出在“字地址”处的“数据”



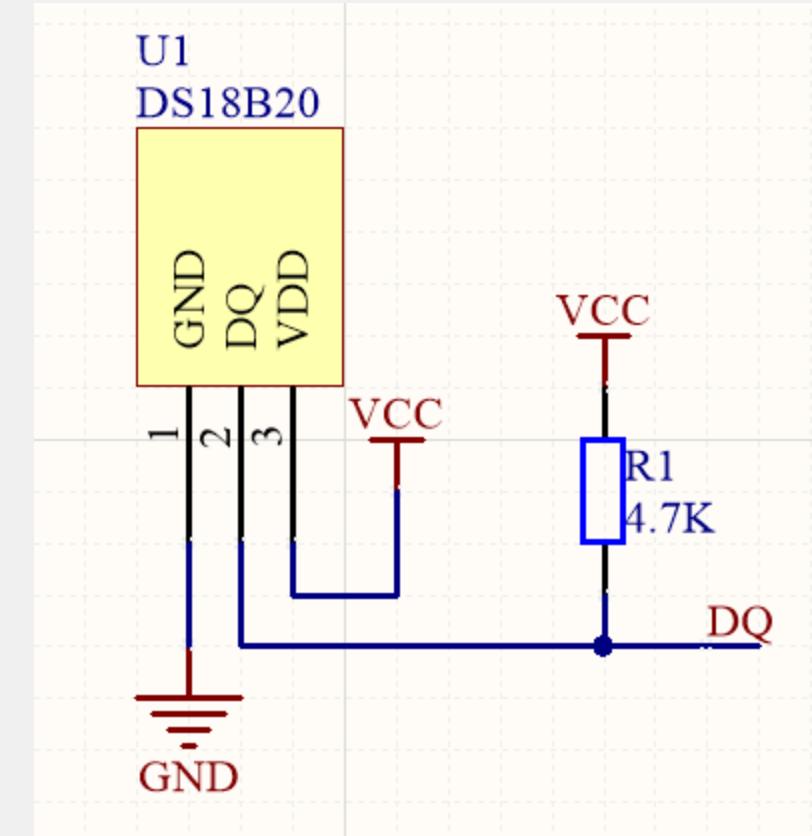
DS18B20介绍

- DS18B20是一种常见的数字温度传感器，其控制命令和数据都是以数字信号的方式输入输出，相比较于模拟温度传感器，具有功能强大、硬件简单、易扩展、抗干扰性强等特点
- 测温范围：-55°C 到 +125°C
- 通信接口：1-Wire（单总线）
- 其它特征：可形成总线结构、内置温度报警功能、可寄生供电

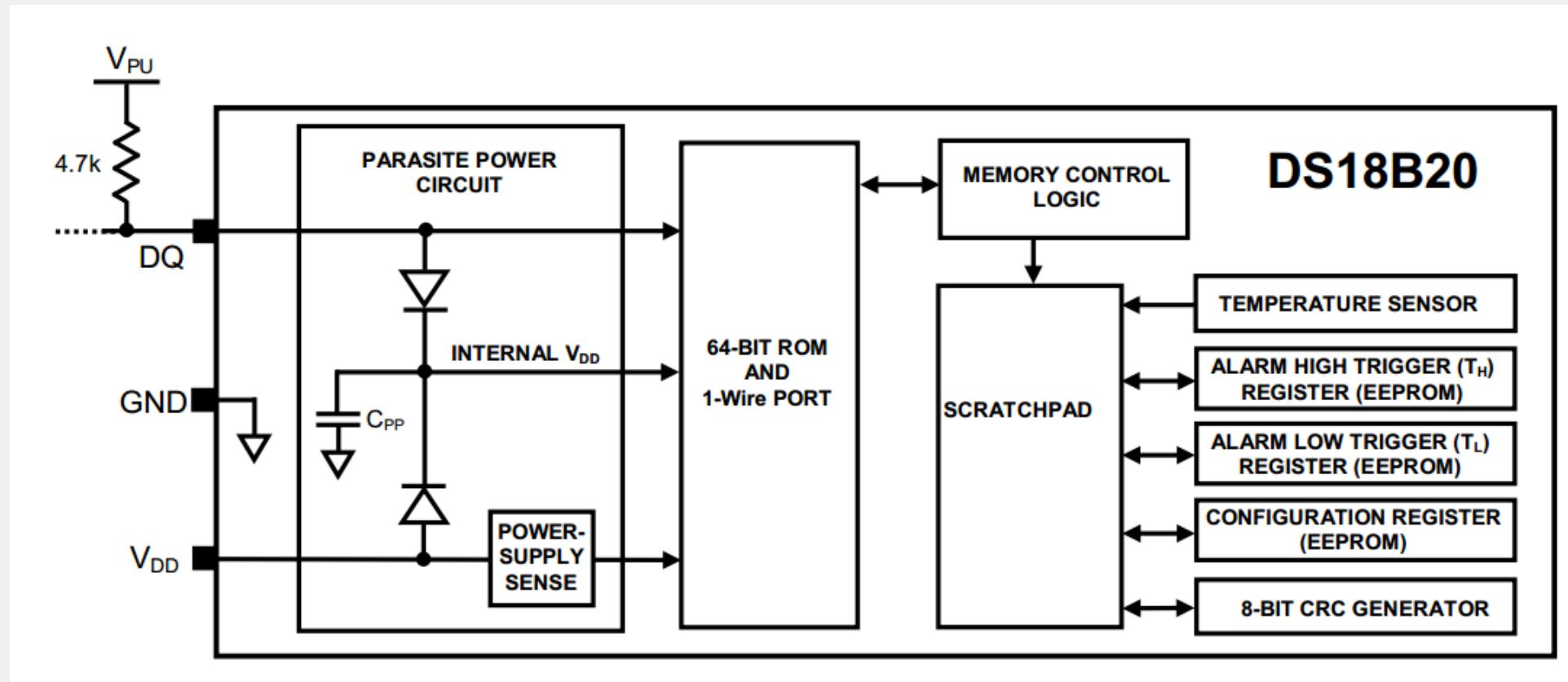


引脚及应用电路

引脚	功能
VDD	电源 (3.0V ~ 5.5V)
GND	电源地
DQ	单总线 接口

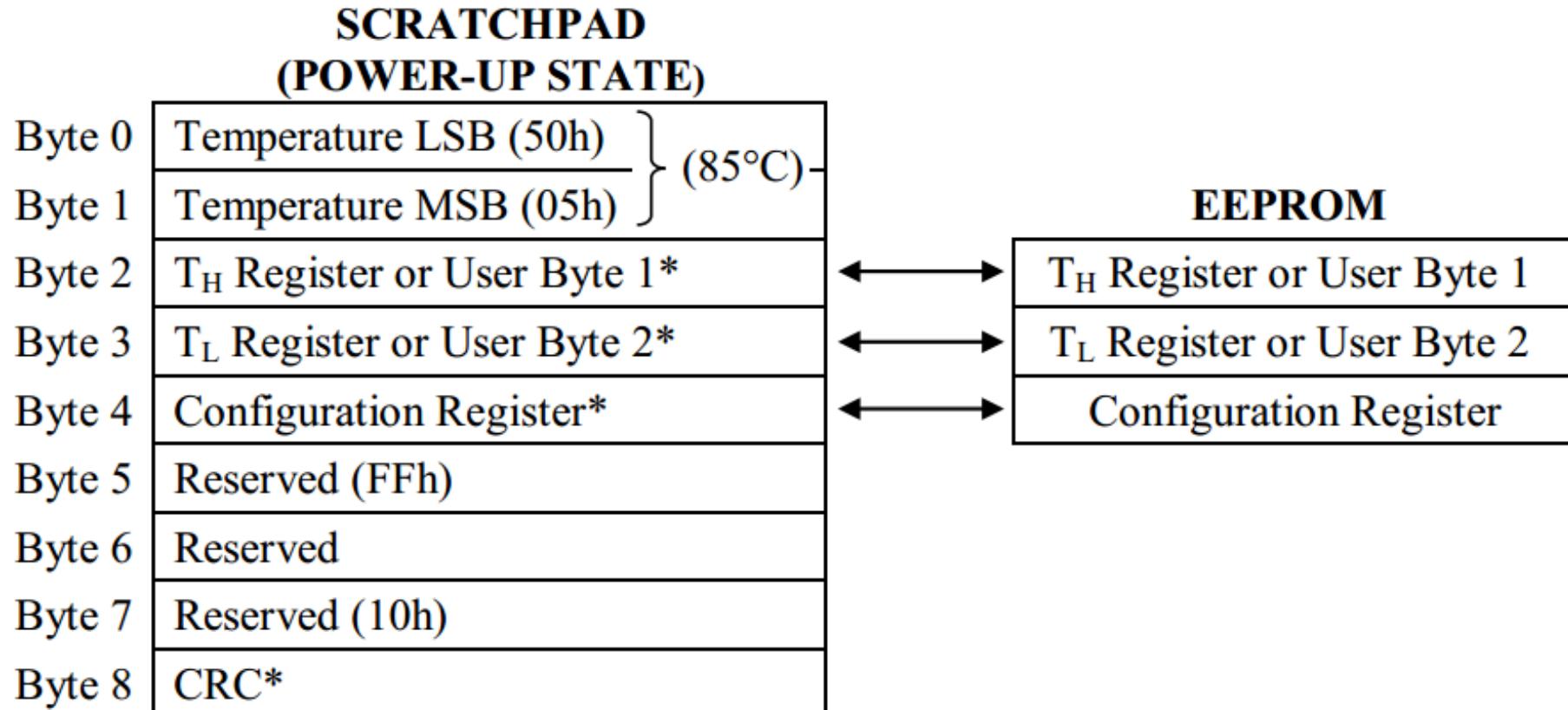


内部结构框图



- 64-BIT ROM：作为器件地址，用于总线通信的寻址
- SCRATCHPAD（暂存器）：用于总线的数据交互
- EEPROM：用于保存温度触发阈值和配置参数

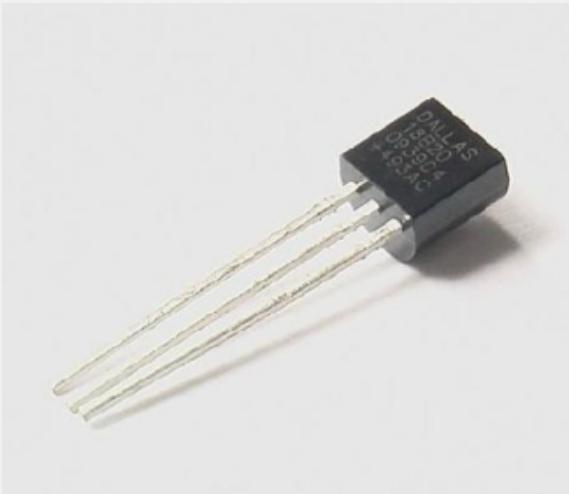
存储器结构



*Power-up state depends on value(s) stored in EEPROM.

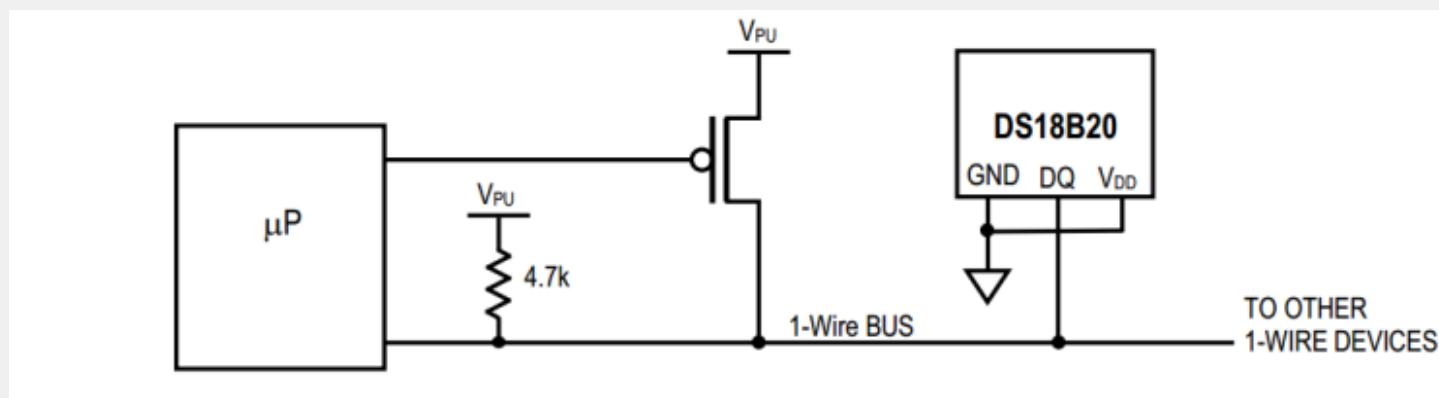
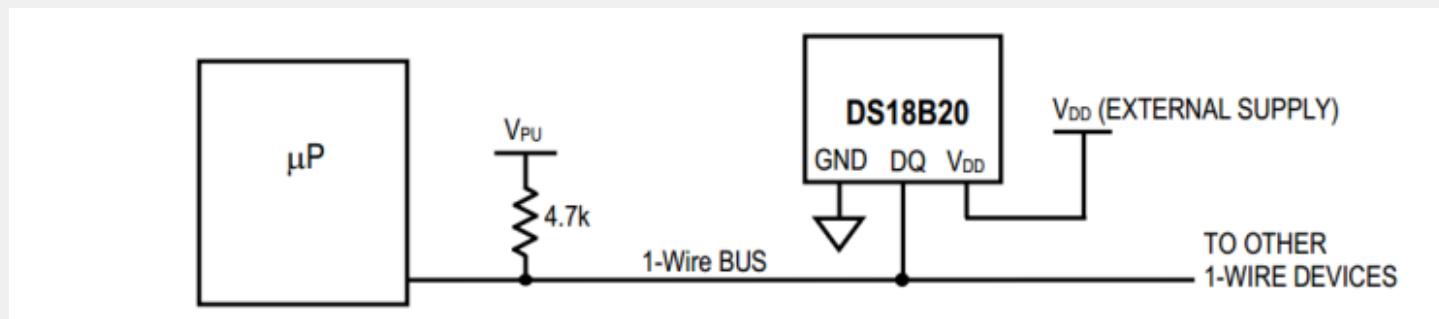
单总线介绍

- 单总线（1-Wire BUS）是由Dallas公司开发的一种通用数据总线
- 一根通信线：DQ
- 异步、半双工
- 单总线只需要一根通信线即可实现数据的双向传输，当采用寄生供电时，还可以省去设备的VDD线路，此时，供电加通信只需要DQ和GND两根线



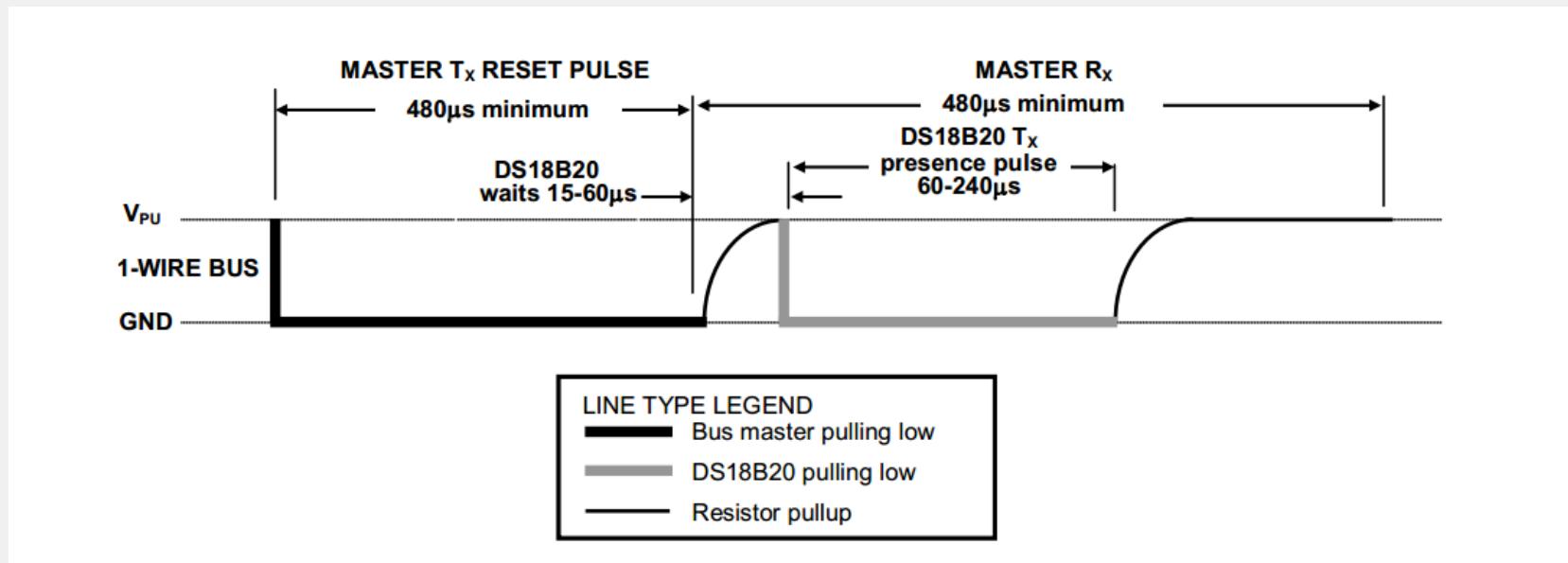
单总线电路规范

- 设备的DQ均要配置成开漏输出模式
- DQ添加一个上拉电阻，阻值一般为4.7KΩ左右
- 若此总线的从机采取寄生供电，则主机还应配一个强上拉输出电路



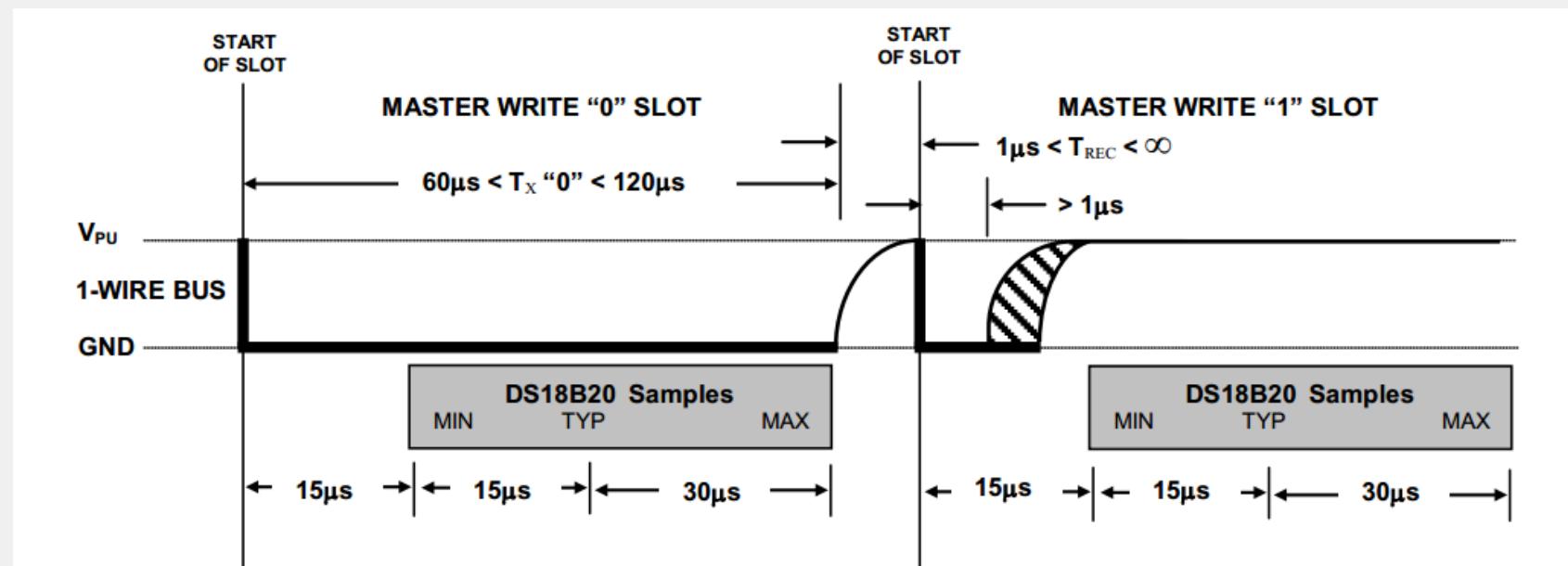
单总线时序结构

- 初始化：主机将总线拉低至少480us，然后释放总线，等待15~60us后，存在的从机会拉低总线60~240us以响应主机，之后从机将释放总线



单总线时序结构

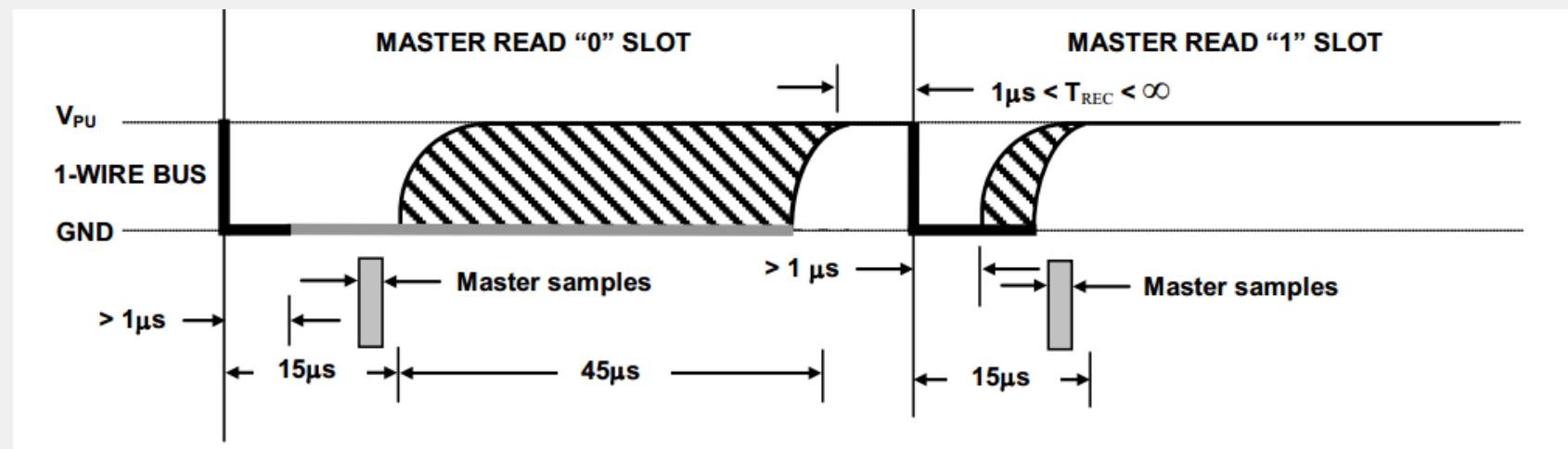
- 发送一位：主机将总线拉低60~120us，然后释放总线，表示发送0；主机将总线拉低1~15us，然后释放总线，表示发送1。从机将在总线拉低30us后（典型值）读取电平，整个时间片应大于60us



S : (BIT)

单总线时序结构

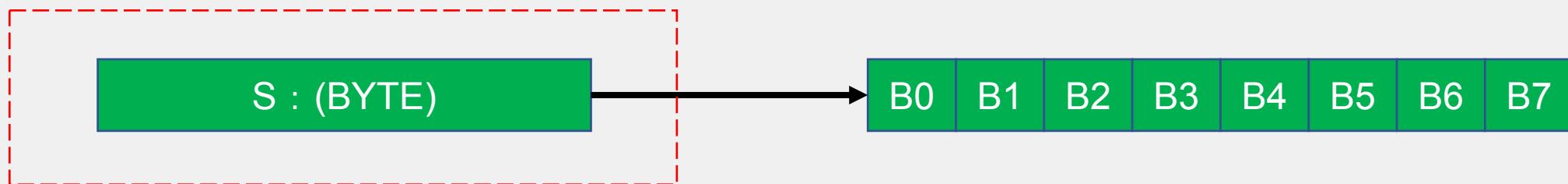
- 接收一位：主机将总线拉低1~15us，然后释放总线，并在拉低后15us内读取总线电平（尽量贴近15us的末尾），读取为低电平则为接收0，读取为高电平则为接收1，整个时间片应大于60us



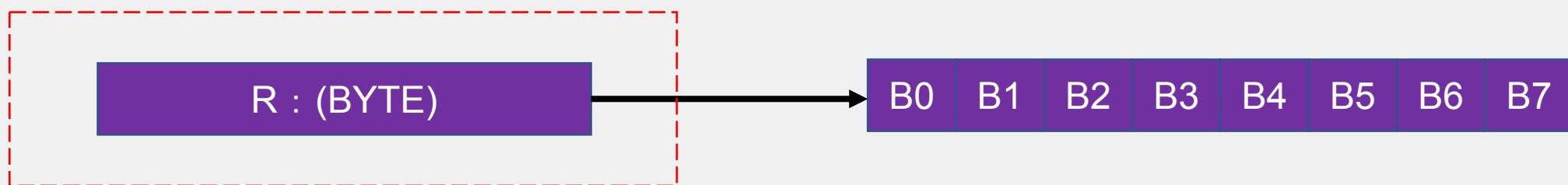
R : (BIT)

单总线时序结构

- 发送一个字节：连续调用8次发送一位的时序，依次发送一个字节的8位（低位在前）



- 接收一个字节：连续调用8次接收一位的时序，依次接收一个字节的8位（低位在前）



DS18B20操作流程

- 初始化：从机复位，主机判断从机是否响应
- ROM操作：ROM指令+本指令需要的读写操作
- 功能操作：功能指令+本指令需要的读写操作

ROM指令	功能指令
SEARCH ROM [F0h]	CONVERT T [44h]
READ ROM [33h]	WRITE SCRATCHPAD [4Eh]
MATCH ROM [55h]	READ SCRATCHPAD [BEh]
SKIP ROM [CCh]	COPY SCRATCHPAD [48h]
ALARM SEARCH [ECh]	RECALL E2 [B8h]
	READ POWER SUPPLY [B4h]

DS18B20数据帧

- 温度变换：初始化→跳过ROM →开始温度变换

S	S : SKIP ROM [CCh]	S : CONVERT T [44h]
---	--------------------	---------------------

- 温度读取：初始化→跳过ROM →读暂存器→连续的读操作

S	S : SKIP ROM [CCh]	S : READ SCRATCHPAD [BEh]	+
---	--------------------	---------------------------	---

R : Temperature LSB	R : Temperature MSB	...
---------------------	---------------------	-----

温度存储格式

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

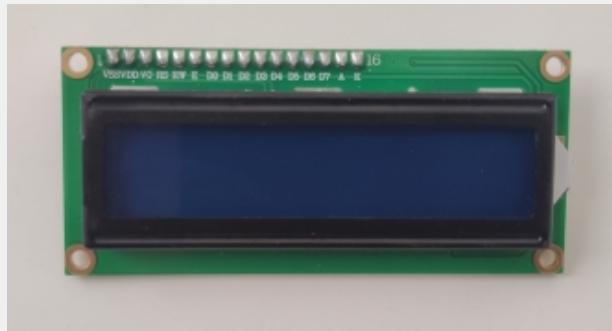
S = SIGN

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D0h
+85*	0000 0101 0101 0000	0550h
+25.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

*The power-on reset value of the temperature register is +85°C.

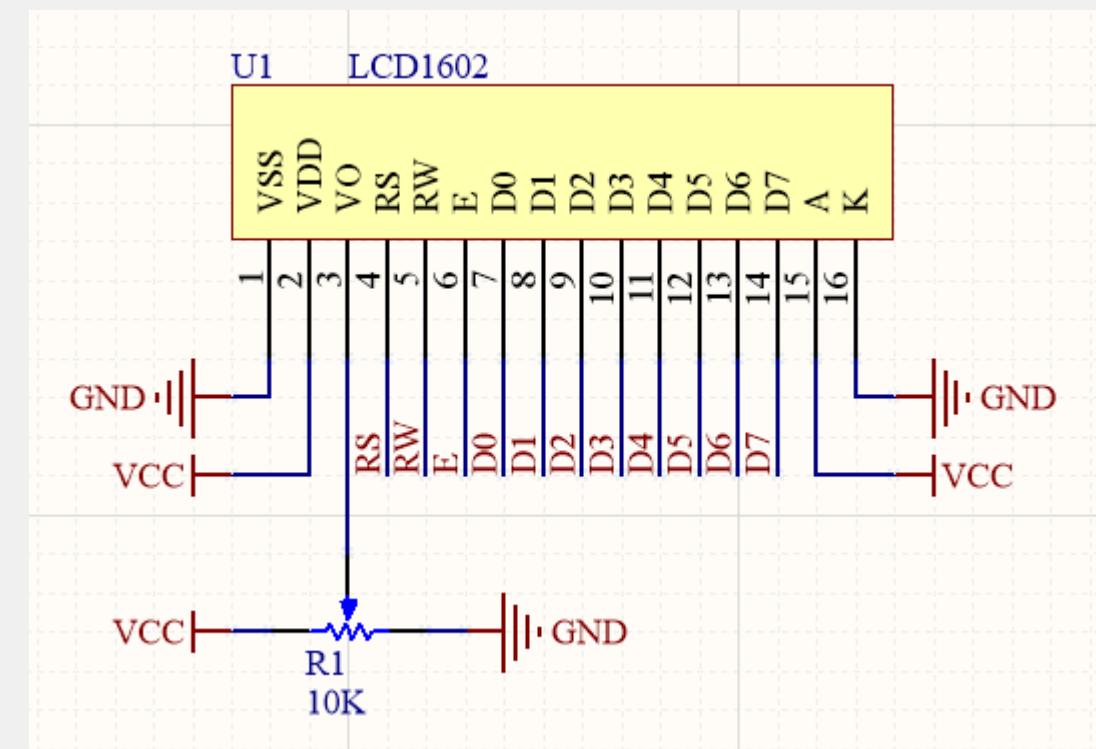
LCD1602介绍

- LCD1602 (Liquid Crystal Display) 液晶显示屏是一种字符型液晶显示模块，可以显示ASCII码的标准字符和其它的一些内置特殊字符，还可以有8个自定义字符
- 显示容量：16×2个字符，每个字符为 5*7 点阵

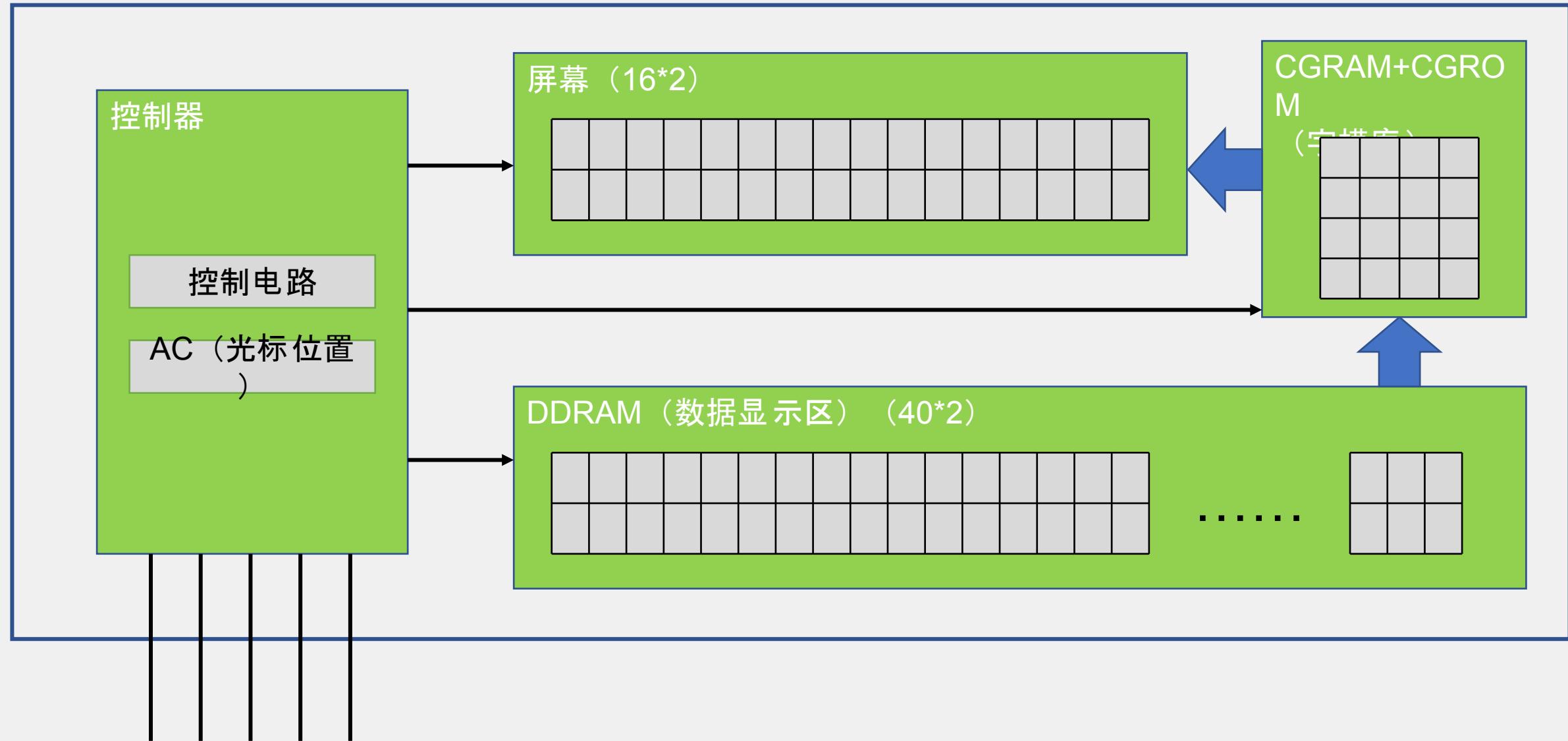


引脚及应用电路

引脚	功能
VSS	地
VDD	电源正极 (4.5~5.5V)
VO	对比度调节电压
RS	数据/指令选择， 1为数据, 0为指令
RW	读/写选择， 1为读， 0为写
E	使能， 1为数据有效， 下降沿执行命令
D0~D7	数据输入/输出
A	背光灯电源正极
K	背光灯电源负极



内部结构框图



存储器结构

DDRAM (数据显示区)

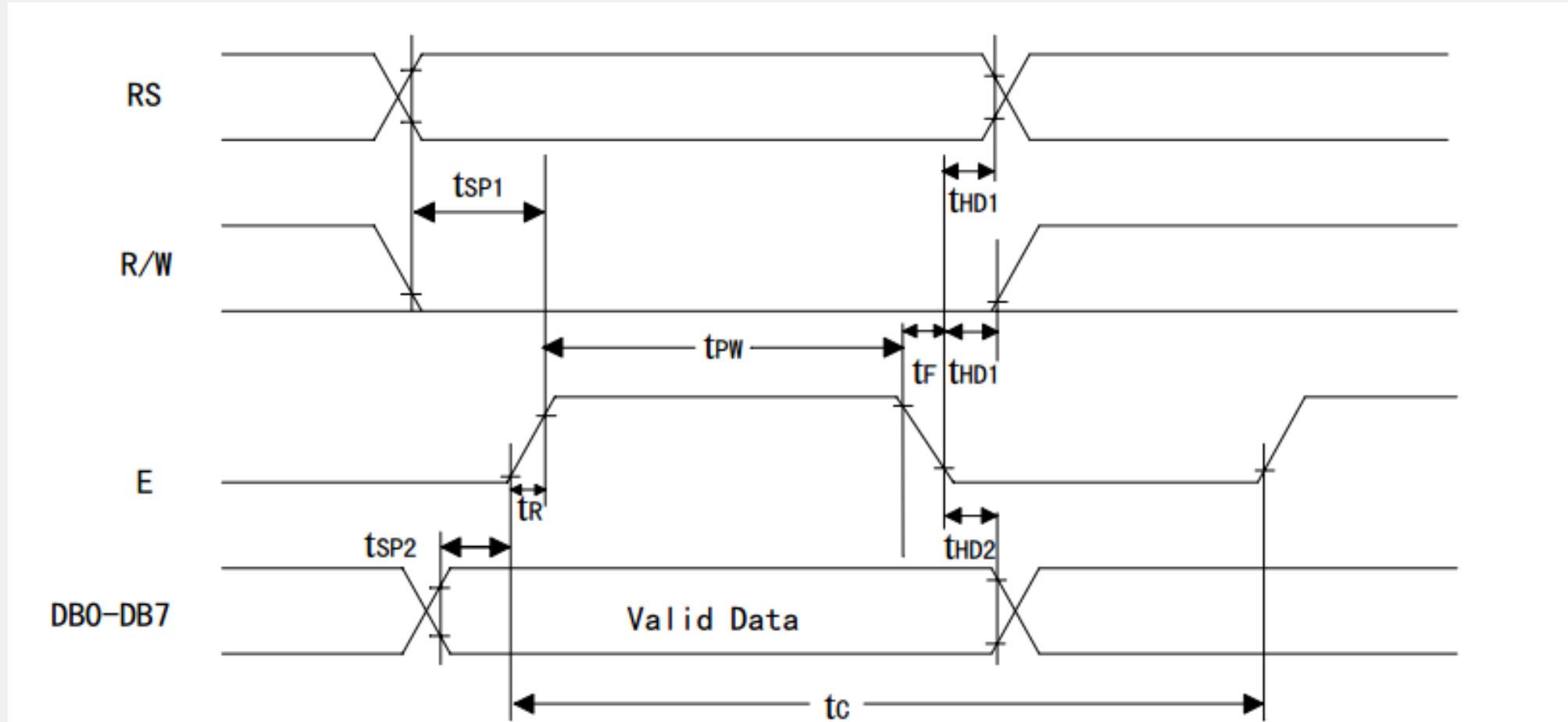
	显示位置	1	2	3	4	5	6	7	40
DDRAM	第一行	00H	01H	02H	03H	04H	05H	06H	27H
地址	第二行	40H	41H	42H	43H	44H	45H	46H	67H

CGRAM+CGROM (字模库)

CGROM中字符码与字字符字模关系对照表																
↓	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000 (1)	0	0	P	^	P				-	タ	ミ	エ	ロ			
xxxx0001 (2)	!	1	A	Q	a	q			.	ア	チ	4	ä	q		
xxxx0010 (3)	"	2	B	R	b	r			「	イ	ツ	×	β	0		
xxxx0011 (4)	#	3	C	S	c	s			」	ウ	テ	モ	~	~		
xxxx0100 (5)	\$	4	D	T	d	t			、	エ	ト	ト	μ	Ω		
xxxx0101 (6)	%	5	E	U	e	u			・	オ	ナ	ユ	σ	0		
xxxx0110 (7)	&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ		
xxxx0111 (8)	'	7	G	W	g	w			ア	キ	ヌ	ラ	¤	π		
xxxx1000 (1)	(8	H	X	h	x			イ	ク	ネ	リ	ゞ	ゞ		
xxxx1001 (2))	9	I	Y	i	y			ウ	ケ	ノ	ル	ヽ	ヽ		
xxxx1010 (3)	*	:	J	Z	j	z			エ	コ	ハ	レ	ゞ	ゞ		
xxxx1011 (4)	+	;	K	C	k	c			オ	サ	ヒ	ロ	¤	¤		
xxxx1100 (5)	,	<	L	¥	l	¥			ヤ	シ	フ	ワ	¥	円		
xxxx1101 (6)	-	=	M]	m	}			ユ	ス	ヘ	ン	モ	÷		
xxxx1110 (7)	.	>	N	^	n	^			ヨ	セ	ホ	^	＼			
xxxx1111 (8)	/	?	O	_	o	_			ツ	ソ	マ	¶	ö	■		

时序结构

- 写数据/指令



LCD1602指令集

LCD1602 字符型液晶显示模块指令集

指令	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	功能	执行时间
1.清屏	0	0	0	0	0	0	0	0	1	*	清除 DDRAM 和 AC 值, 光标复位 (清屏) 01	1.64us
2.归位	0	0	0	0	0	0	0	0	1	*	AC=0 , 光标复位, DDRAM 内容不变。(复位) 02	1.64us
3.输入方式设置	0	0	0	0	0	0	1	1	1	0	数据读、写操作后, AC 自动加一; 画面平移。 07	40us
										0	数据读、写操作后, AC 自动加一; 画面不动。 06	
										1	数据读、写操作后, AC 自动减一; 画面平移。 05	
										0	数据读、写操作后, AC 自动减一; 画面不动。 04	
4.显示开关控制	0	0	0	0	0	0	1	0	0	0	显示关, 光标关, 闪烁关。 08	40us
										0	显示关, 光标关, 闪烁开。 09	
										1	显示关, 光标开, 闪烁关。 0a	
										1	显示关, 光标开, 闪烁开。 0b	
										0	显示开, 光标关, 闪烁关。 0c	
										1	显示开, 光标关, 闪烁开。 0d	
										1	显示开, 光标开, 闪烁关。 0e	
										1	显示开, 光标开, 闪烁开。 0f	
5.光标、画面位移	0	0	0	0	0	1	0	0	*	*	光标向左平移一个字符位, AC 值减 1。 10	40us
										1	*	
										0	光标向右平移一个字符位, AC 值加 1。 14	
										0	画面向左平移一个字符位, 但光标不动。移屏 18	
6.功能设置	0	0	0	0	1	0	0	0	*	*	画面向右平移一个字符位, 但光标不动。移屏 1c	40us
										1	四位数据接口, 一行显示, 5x7 点阵 20	
										0	四位数据接口, 一行显示, 5x10 点阵 24	
										1	四位数据接口, 两行显示, 5x7 点阵 28	
										1	四位数据接口, 两行显示, 5x10 点阵 29	
										0	八位数据接口, 一行显示, 5x7 点阵 30	
										1	八位数据接口, 一行显示, 5x10 点阵 34	
										1	八位数据接口, 两行显示, 5x7 点阵 38	
7.CGRAM 地址设置	0	0	0	1	A5	A4	A3	A2	A1	A0	设置 CGRAM 地址。 A5~A0=0~3FH	40us
8.DDRAM 地址设置	0	0	1	A6	A5	A4	A3	A2	A1	A0	设定下一个要存入数据的 DDRAM 的地址。	40us
9.读 BF 及 AC 值	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	BF=1: 忙; BF=0: 准备好。 AC 值意义为最近一次地址设置 (CGRAM 或 DDRAM) 定义。	40us

LCD1602操作流程

- 初始化：
 - 发送指令0x38 //八位数据接口，两行显示，5*7点阵
 - 发送指令0x0C //显示开，光标关，闪烁关
 - 发送指令0x06 //数据读写操作后，光标自动加一，画面不动
 - 发送指令0x01 //清屏
 - 显示字符：
 - 发送指令0x80|AC //设置光标位置
 - 发送数据 //发送要显示的字符数据
 - 发送数据 //发送要显示的字符数据
-

字符、字符串

- 字符：根据一定规则建立的数字到字符的映射（ASCII码表）
例如：0x21='!', 0x41='A', 0x00='\0'
定义方法：char x='A'; （等效于char x=0x41;）
- 字符数组：存储字符变量的一个数组
定义方法：char y[]={‘A’, ‘B’, ‘C’};
（等效于char y[]={0x41,0x42,0x43}; ）
- 字符串：在字符数组后加一个字符串结束标志，本质上是字符数组
定义方法：char z[]="ABC"; （等效于char z[]={‘A’, ‘B’, ‘C’, ‘\0’};
）

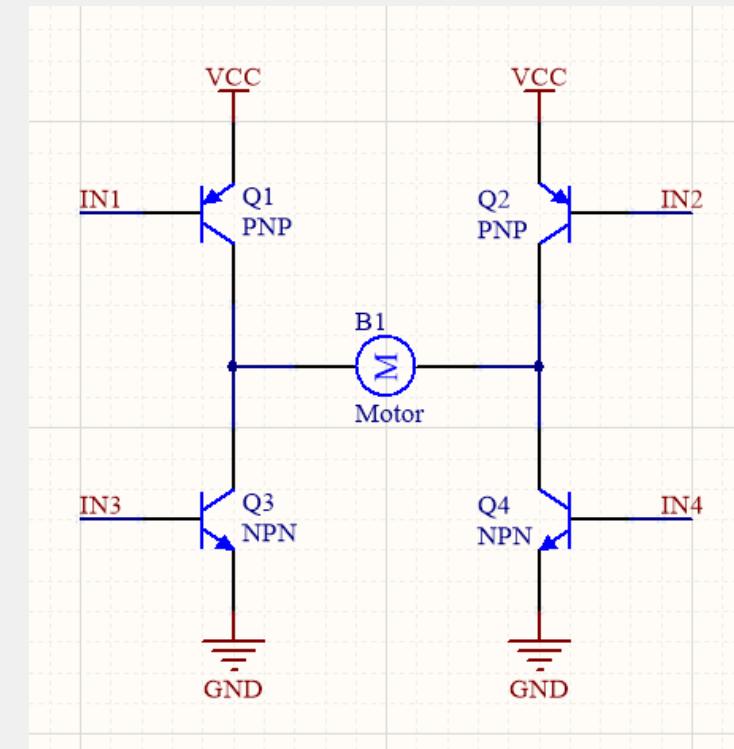
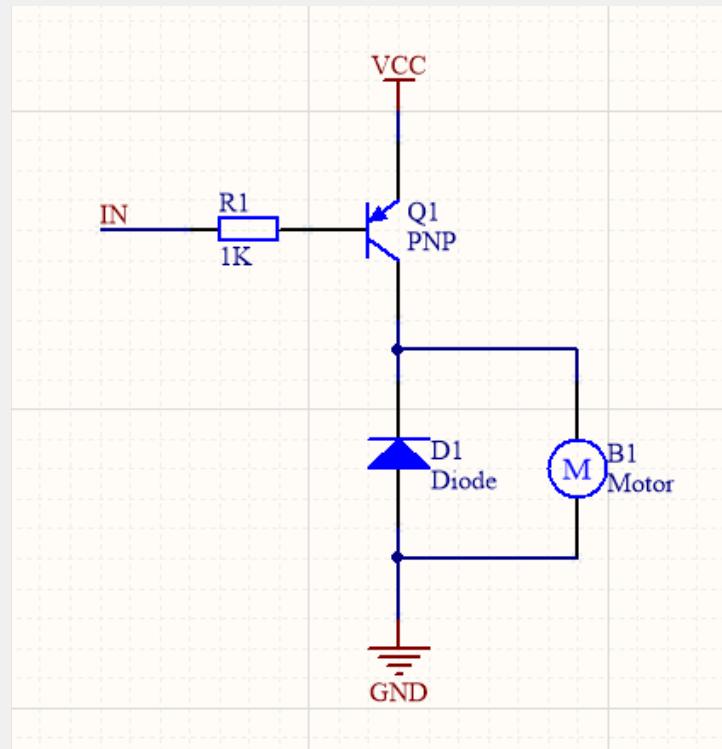
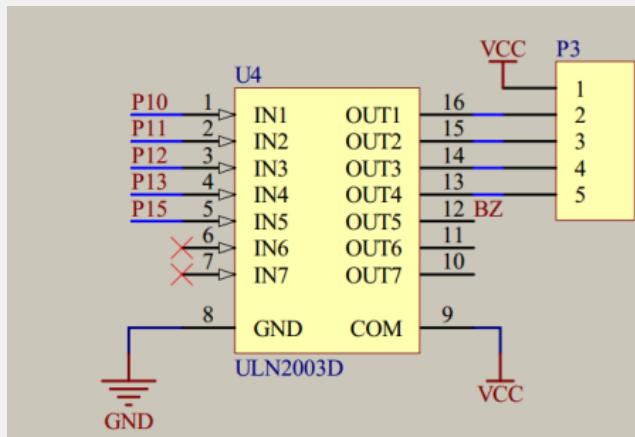
直流电机介绍

- 直流电机是一种将电能转换为机械能的装置。一般的直流电机有两个电极，当电极正接时，电机正转，当电极反接时，电机反转
- 直流电机主要由永磁体（定子）、线圈（转子）和换向器组成
- 除直流电机外，常见的电机还有步进电机、舵机、无刷电机、空心杯电机等



电机驱动电路

- 大功率器件直接驱动

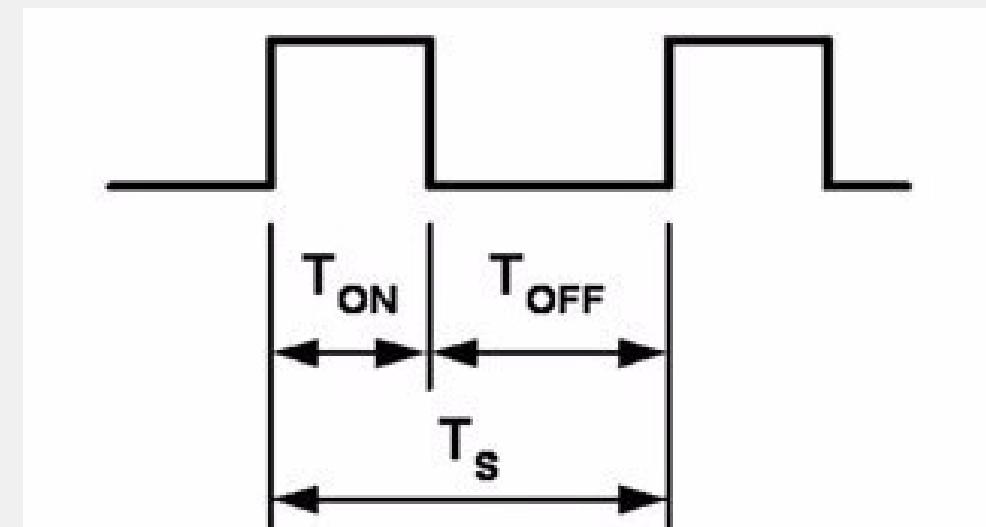
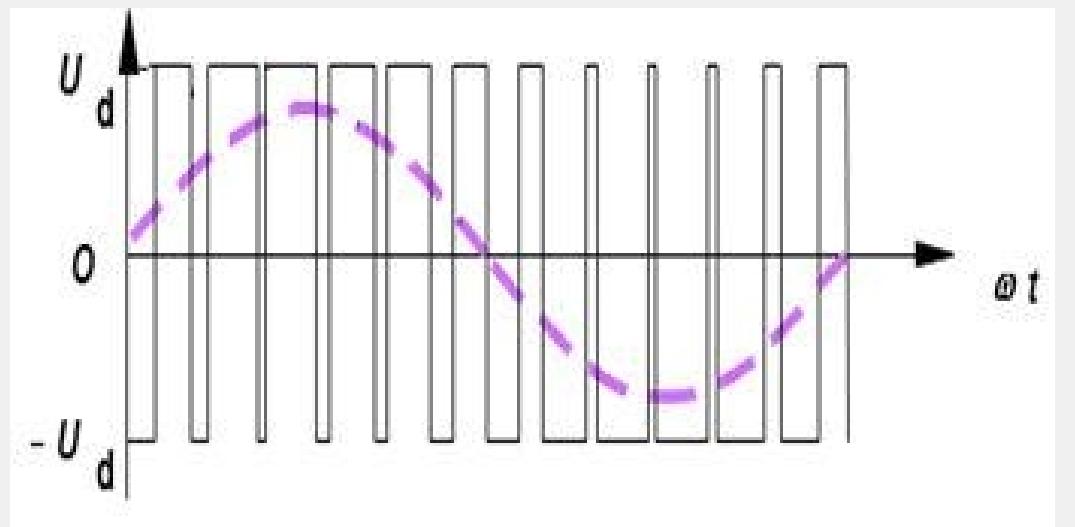


- H桥驱动

PWM介绍

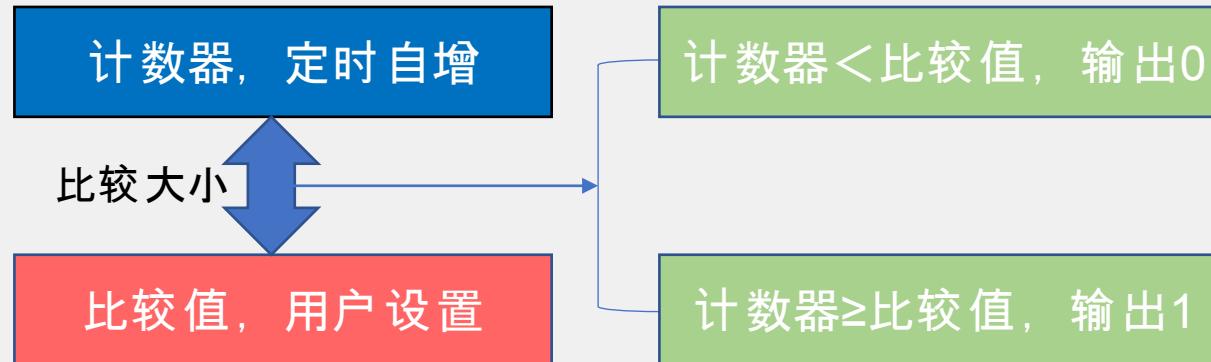
- PWM (Pulse Width Modulation) 即脉冲宽度调制，在具有惯性的系统中，通过对一系列脉冲的宽度进行调制，来等效地获得所需要的模拟参量，常应用于电机控速、开关电源等领域
- PWM重要参数：

$$\text{频率} = 1 / T_S \quad \text{占空比} = T_{ON} / T_S \quad \text{精度} = \text{占空比变化步距}$$

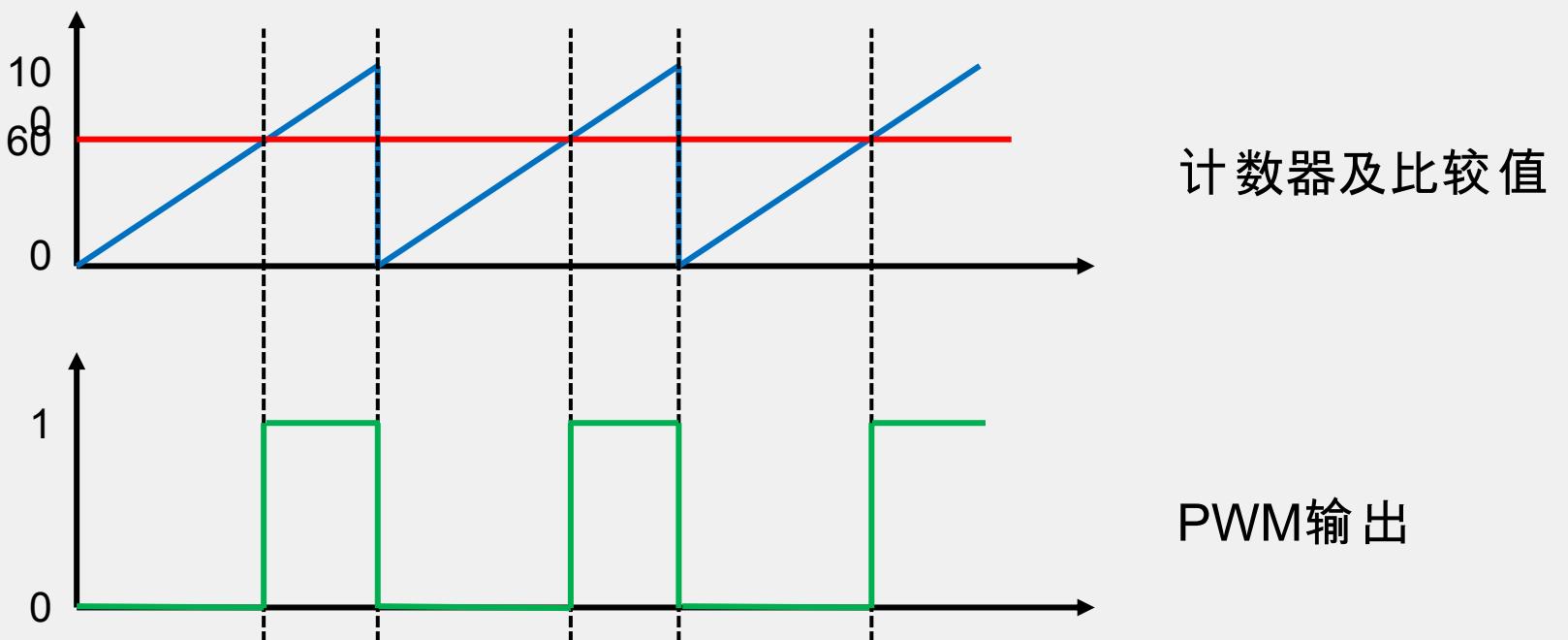


产生PWM方法

- 模型结构

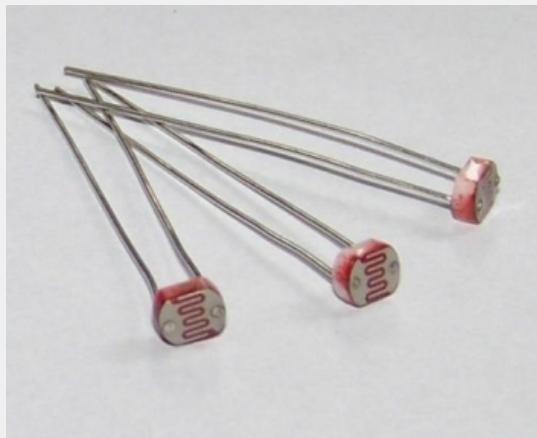


- 波形



AD/DA介绍

- AD (Analog to Digital) : 模拟-数字转换，将模拟信号转换为计算机可操作的数字信号
- DA (Digital to Analog) : 数字-模拟转换，将计算机输出的数字信号转换为模拟信号
- AD/DA转换打开了计算机与模拟信号的大门，极大的提高了计算机系统的应用范围，也为模拟信号数字化处理提供了可能

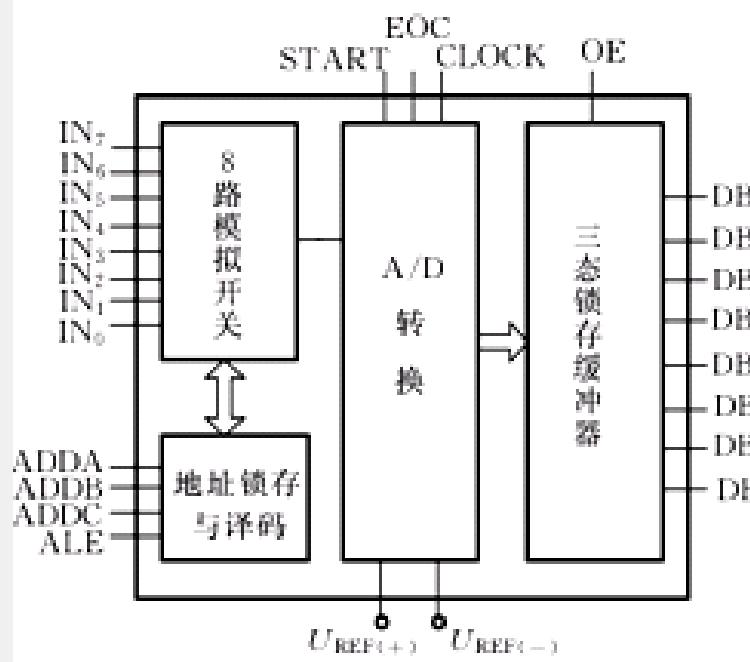
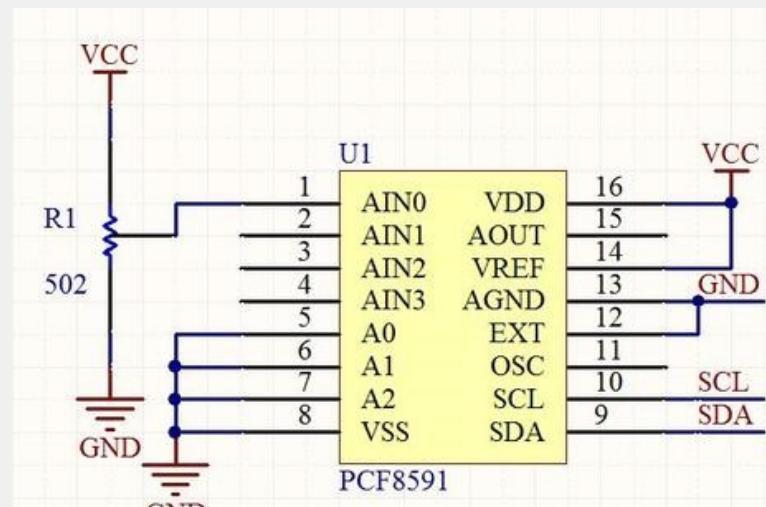


硬件电路模型

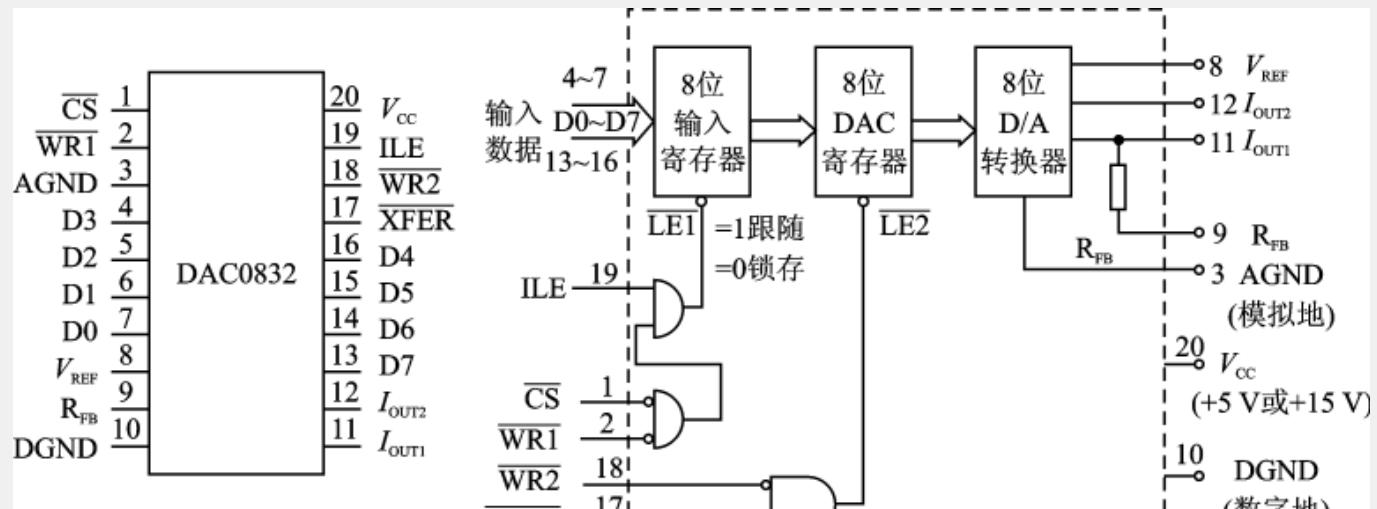
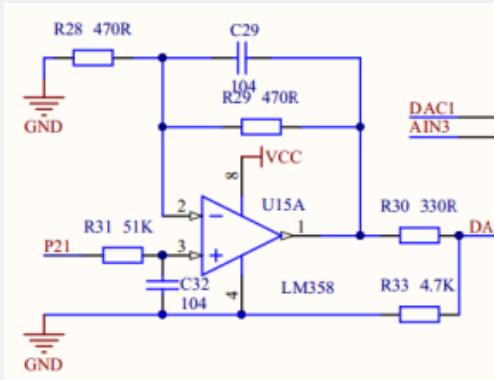
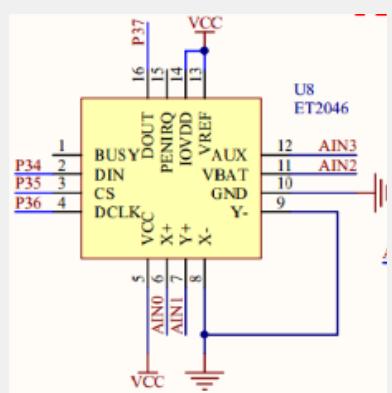


- AD转换通常有多个输入通道，用多路选择开关连接至AD转换器，以实现AD多路复用的目的，提高硬件利用率
- AD/DA与单片机数据传送可使用并口（速度快、原理简单），也可使用串口（接线少、使用方便）
- 可将AD/DA模块直接集成在单片机内，这样直接写入/读出寄存器就可进行AD/DA转换，单片机的IO口可直接复用为AD/DA的通道

硬件电路

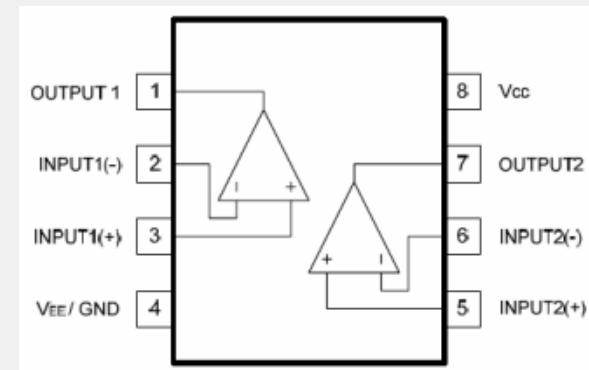
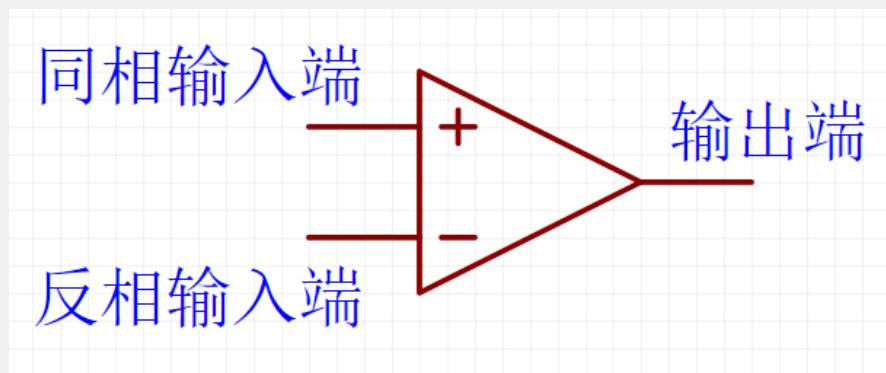


IN ₃	1	IN ₂	28
IN ₄	2	IN ₁	27
IN ₅	3	IN ₀	26
IN ₆	4	ADD _A	25
IN ₇	5	ADD _B	24
START	6	ADD _C	23
EOC	7	ADC0809	22
D ₃	8	D ₇	21
OE	9	D ₆	20
CLOCK	10	D ₅	19
U_{cc}	11	D ₄	18
$U_{REF(+)}$	12	D ₃	17
GND	13	D ₂	16
D ₁	14	D ₁	15



运算放大器

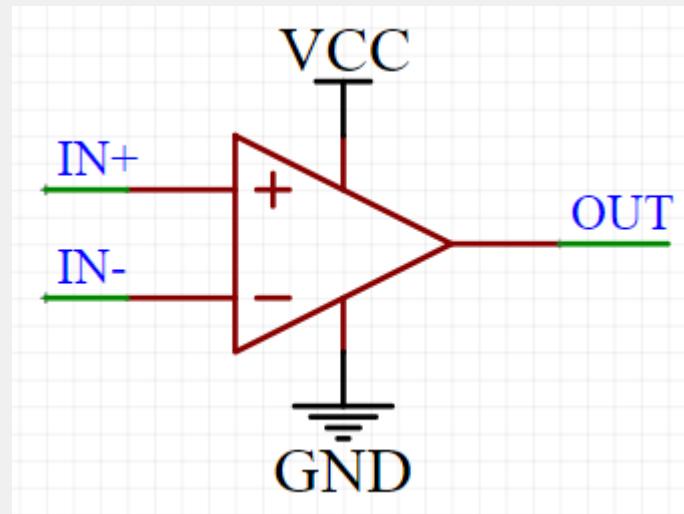
- 运算放大器（简称“运放”）是具有很高放大倍数的放大电路单元。内部集成了差分放大器、电压放大器、功率放大器三级放大电路，是一个性能完备、功能强大的通用放大电路单元，由于其应用十分广泛，现已作为基本的电路元件出现在电路图中
- 运算放大器可构成的电路有：电压比较器、反相放大器、同相放大器、电压跟随器、加法器、积分器、微分器等
- 运算放大器电路的分析方法：虚短、虚断（负反馈条件下）



运放电路

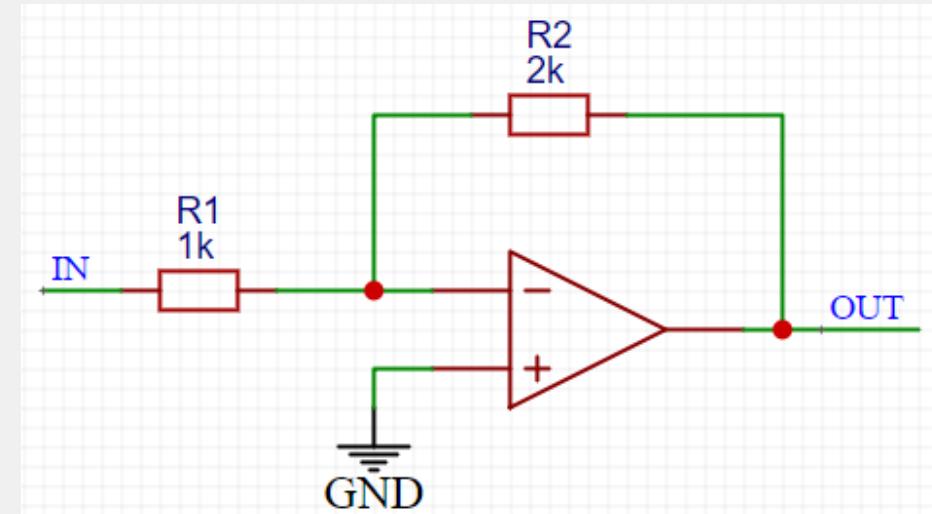
- 电压比较器

$$\bullet V_{OUT} = \begin{cases} VCC & (V_{IN+} > V_{IN-}) \\ GND & (V_{IN+} < V_{IN-}) \end{cases}$$



- 反向放大器

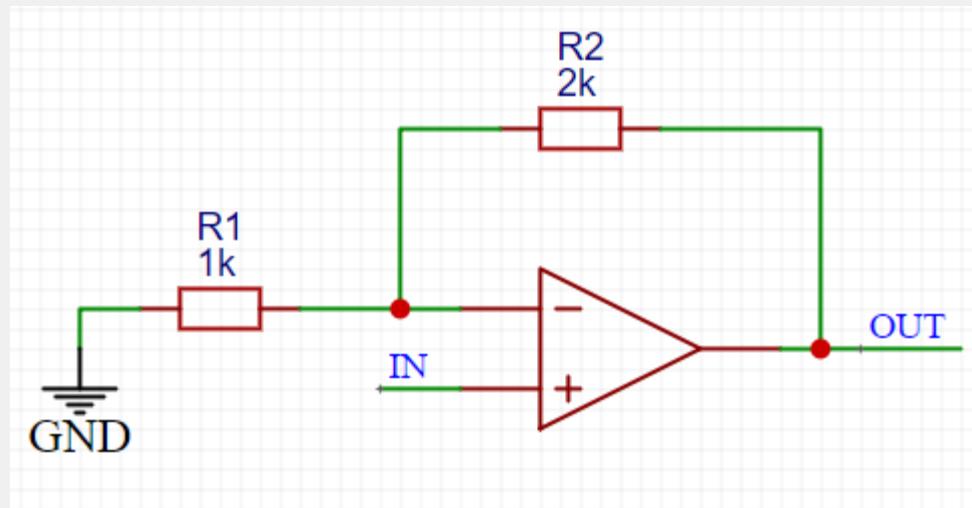
$$\bullet V_{OUT} = -\frac{R2}{R1} \times V_{IN}$$



运放电路

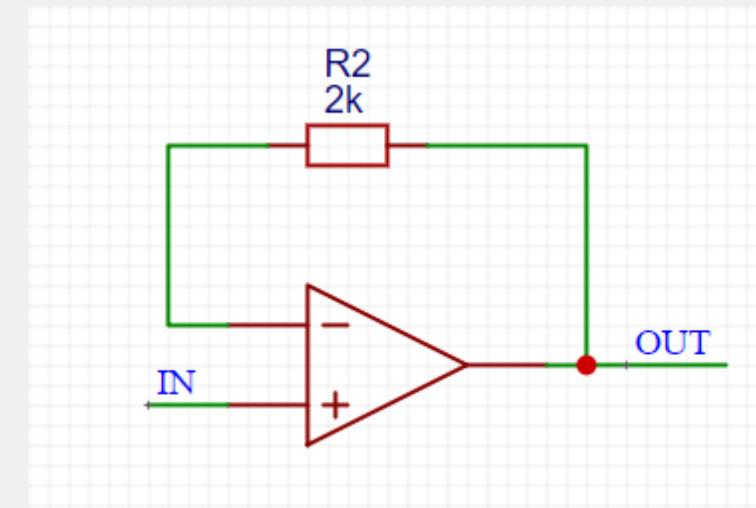
- 同向放大器

- $V_{OUT} = (1 + \frac{R2}{R1}) \times V_{IN}$



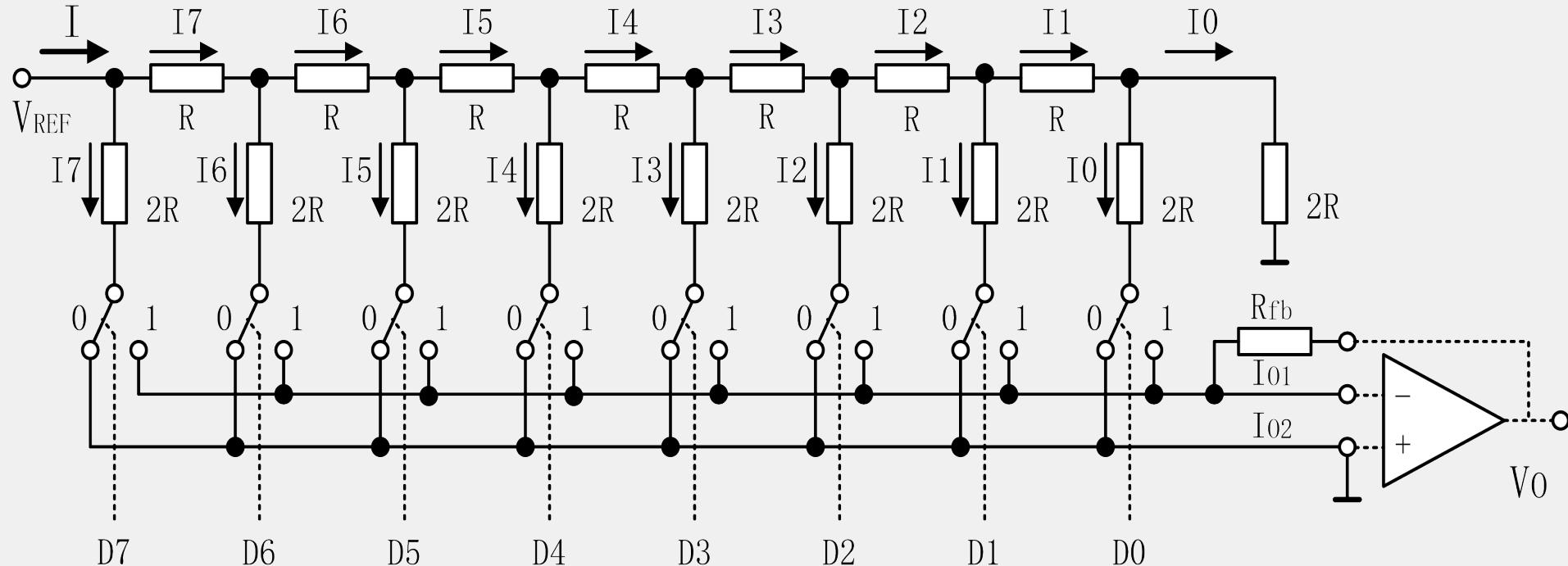
- 电压跟随器

- $V_{OUT} = V_{IN}$



DA原理

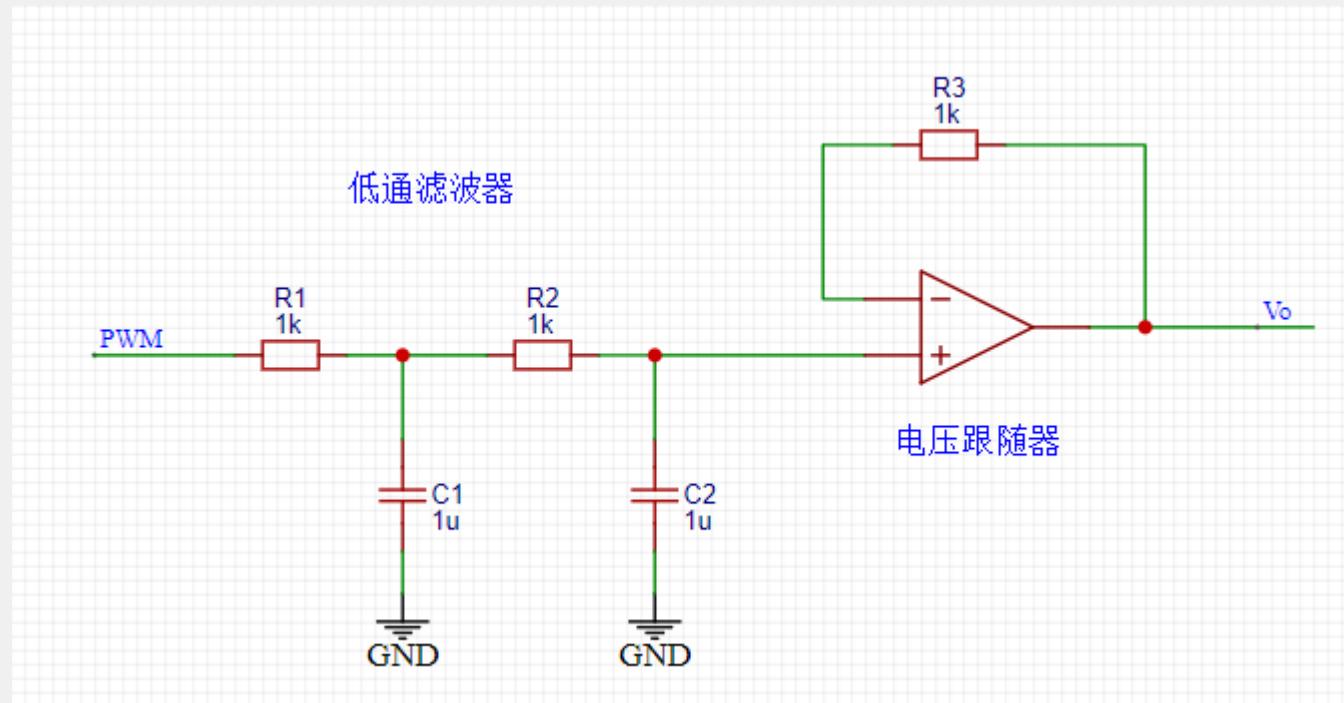
- T型电阻网络DA转换器：



- 输出电压 $V_O = \frac{(D_7 \sim D_0)}{256} \times \frac{V_{REF} \times R_{fb}}{R}$
- 当 $R_{fb} = R$ 时, $V_O = \frac{(D_7 \sim D_0)}{256} \times V_{REF}$

DA原理

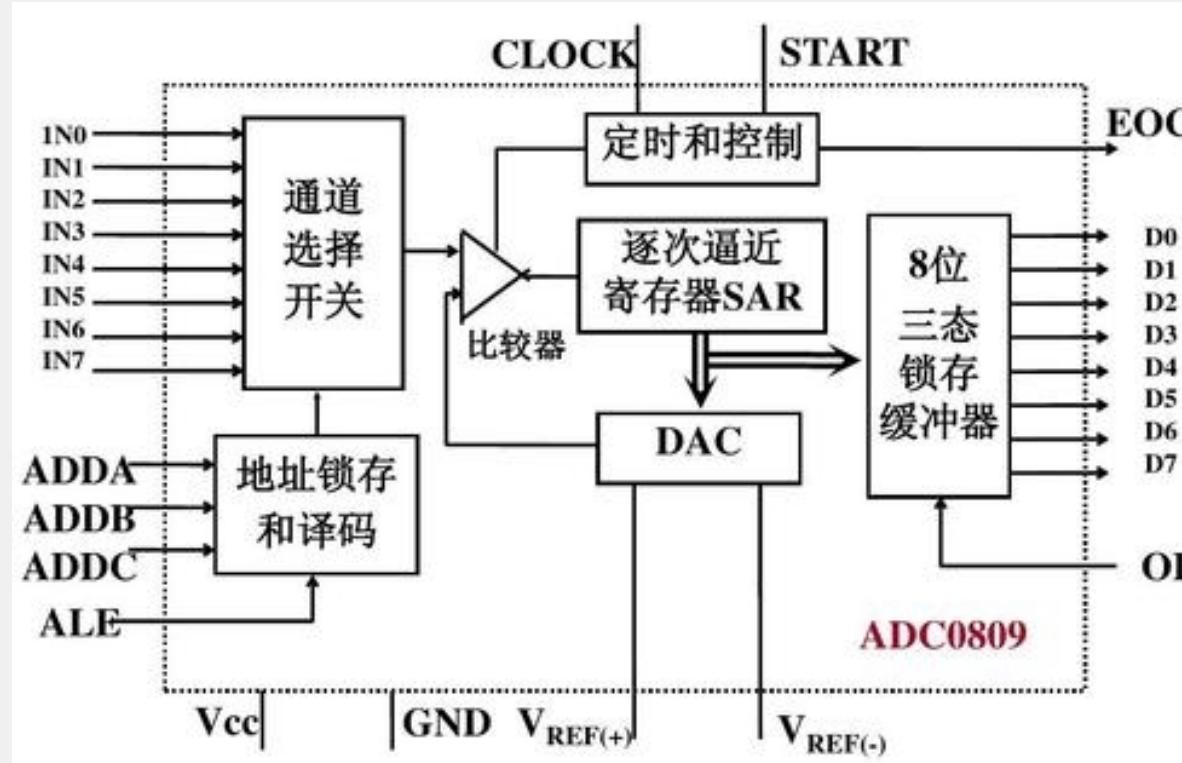
- PWM型DA转换器：



- 输出电压 $V_o = (\text{PWM占空比}) \times V_H$

AD原理

- 逐次逼近型AD转换器：



- 输出数字量 $(D7 \sim D0) = \frac{V_{IN}}{V_{REF}} \times 256 \dots \dots$ 结果取整

AD/DA性能指标

- 分辨率：指AD/DA数字量的精细程度，通常用位数表示。例如，对于5V电源系统来说，8位的AD可将5V等分为256份，即数字量变化最小一个单位时，模拟量变化 $5V/256=0.01953125V$ ，所以，8位AD的电压分辨率为0.01953125V，AD/DA的位数越高，分辨率就越高
- 转换速度：表示AD/DA的最大采样/建立频率，通常用转换频率或者转换时间来表示，对于采样/输出高速信号，应注意AD/DA的转换速度

1 功能说明

XPT2046 是一款 4 线制电阻式触摸屏控制器，内含 12 位分辨率 125KHz 转换速率逐步逼近型 A/D 转换器。XPT2046 支持从 1.5V 到 5.25V 的低电压 I/O 接口。XPT2046 能通过执行两次 A/D 转换查出被按的屏幕位置，除此之外，还可以测量加在触摸屏上的压力。内部自带 2.5V 参考电压，可以作为辅助输入、温度测量和电池监测之用，电池监测的电压范围可以从 0V 到 6V。XPT2046 片内集成有一个温度传感器。在 2.7V 的典型工作状态下，关闭参考电压，功耗可小于 0.75mW。XPT2046 采用微小的封装形式：TSSOP-16, QFN-16 和 VFBGA-48。工作温度范围为 -40°C ~ +85°C。与 ADS7846、TSC2046、AK4182A 完全兼容

XPT2046时序

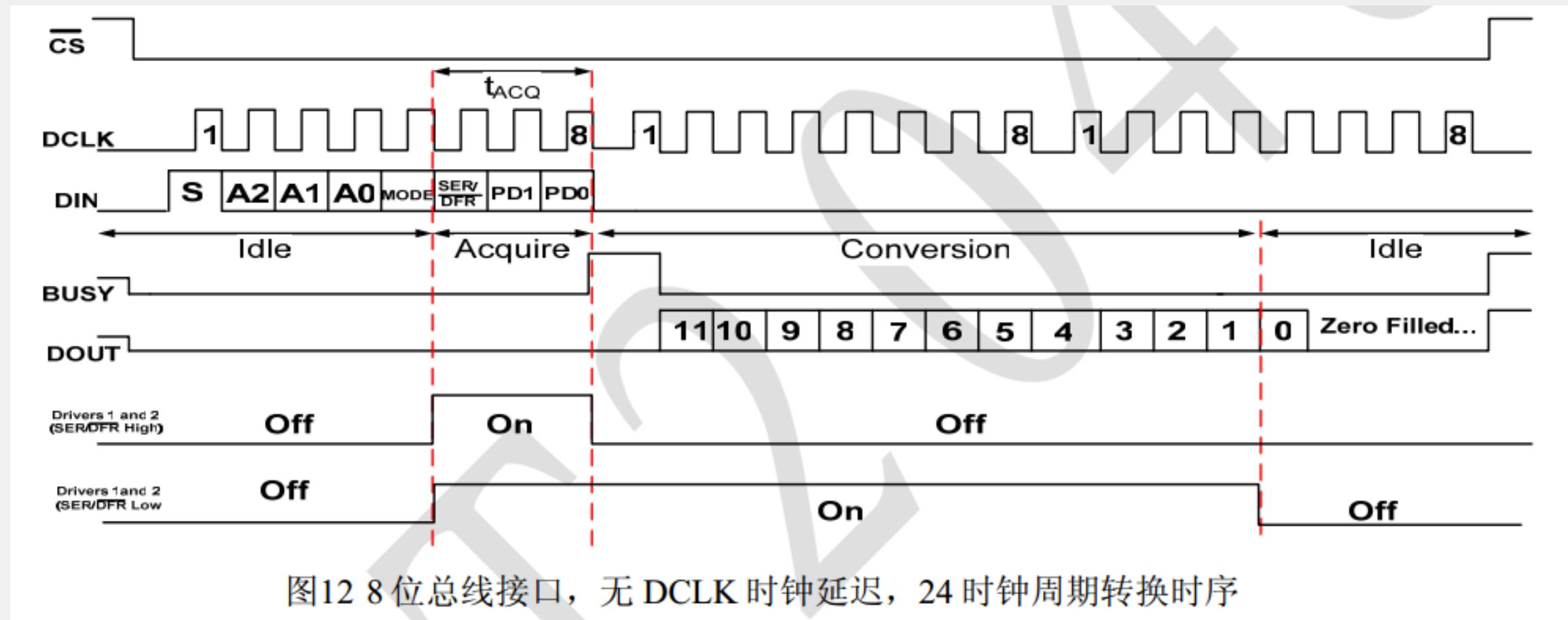


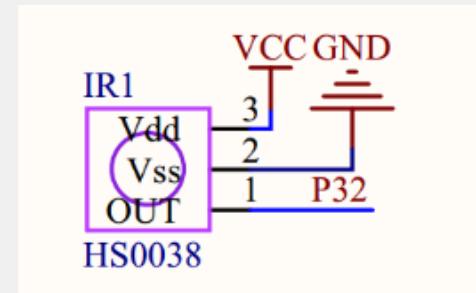
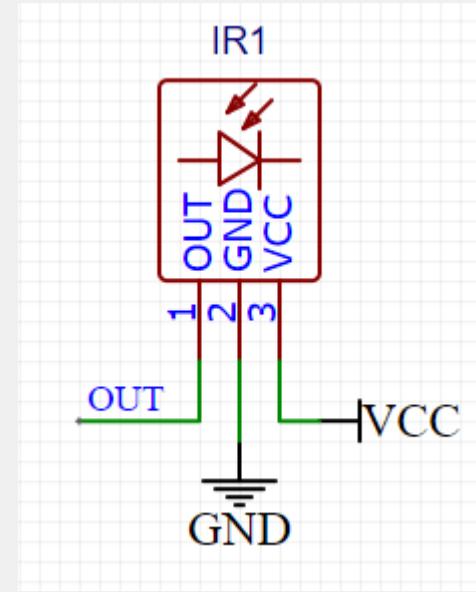
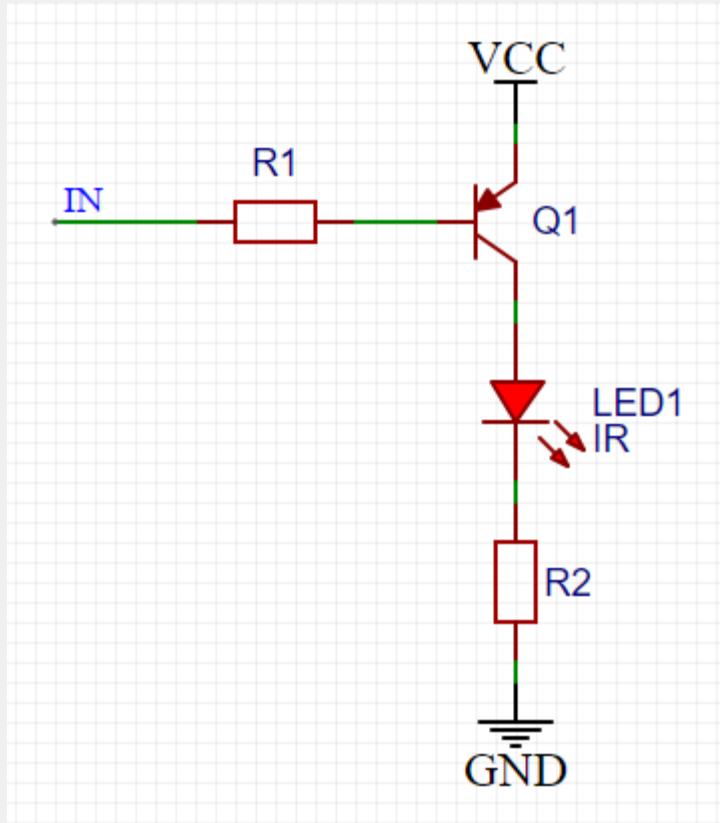
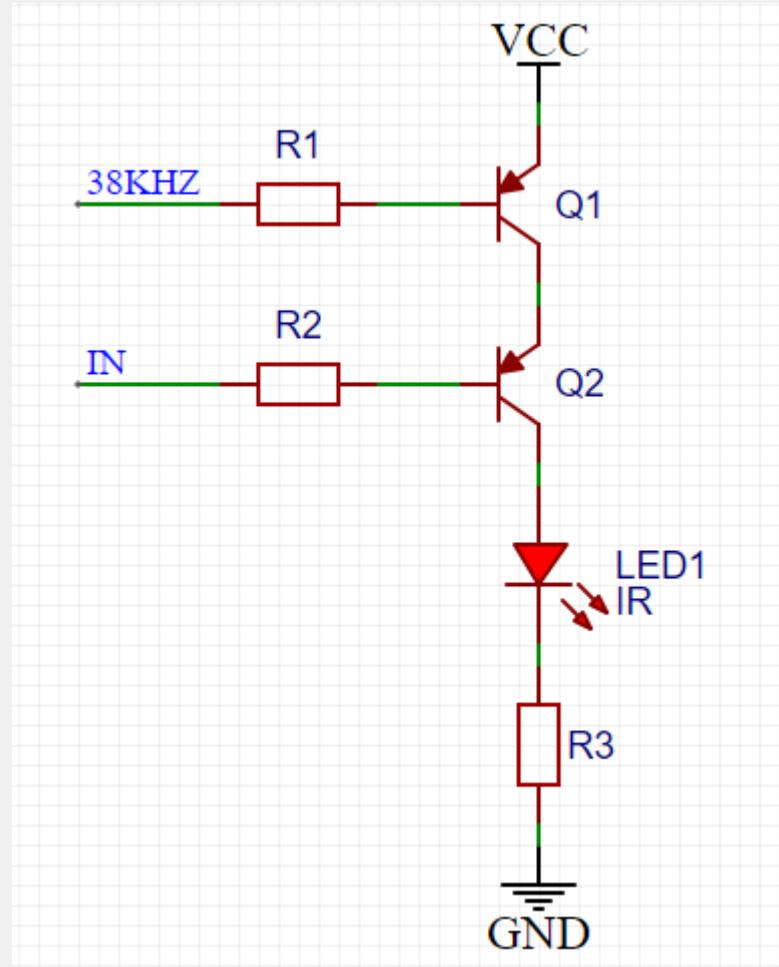
图12 8位总线接口，无 DCLK 时钟延迟，24时钟周期转换时序

红外遥控简介

- 红外遥控是利用红外光进行通信的设备，由红外LED将调制后的信号发出，由专用的红外接收头进行解调输出
- 通信方式：单工，异步
- 红外LED波长：940nm
- 通信协议标准：NEC标准

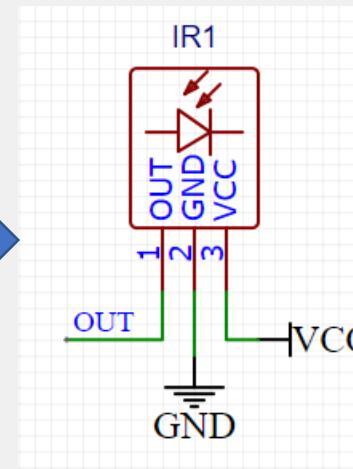
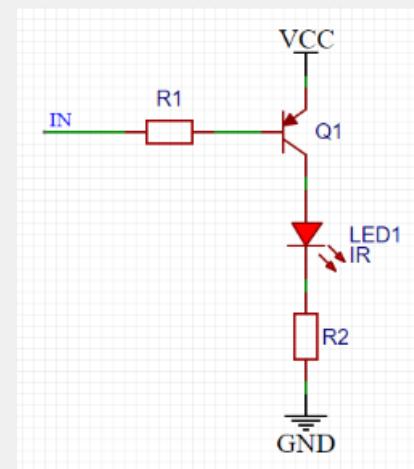
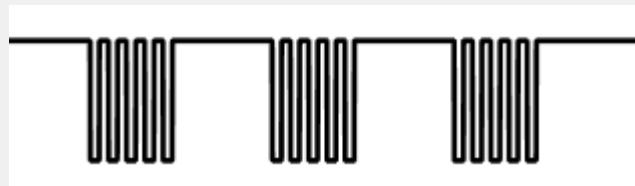


硬件电路

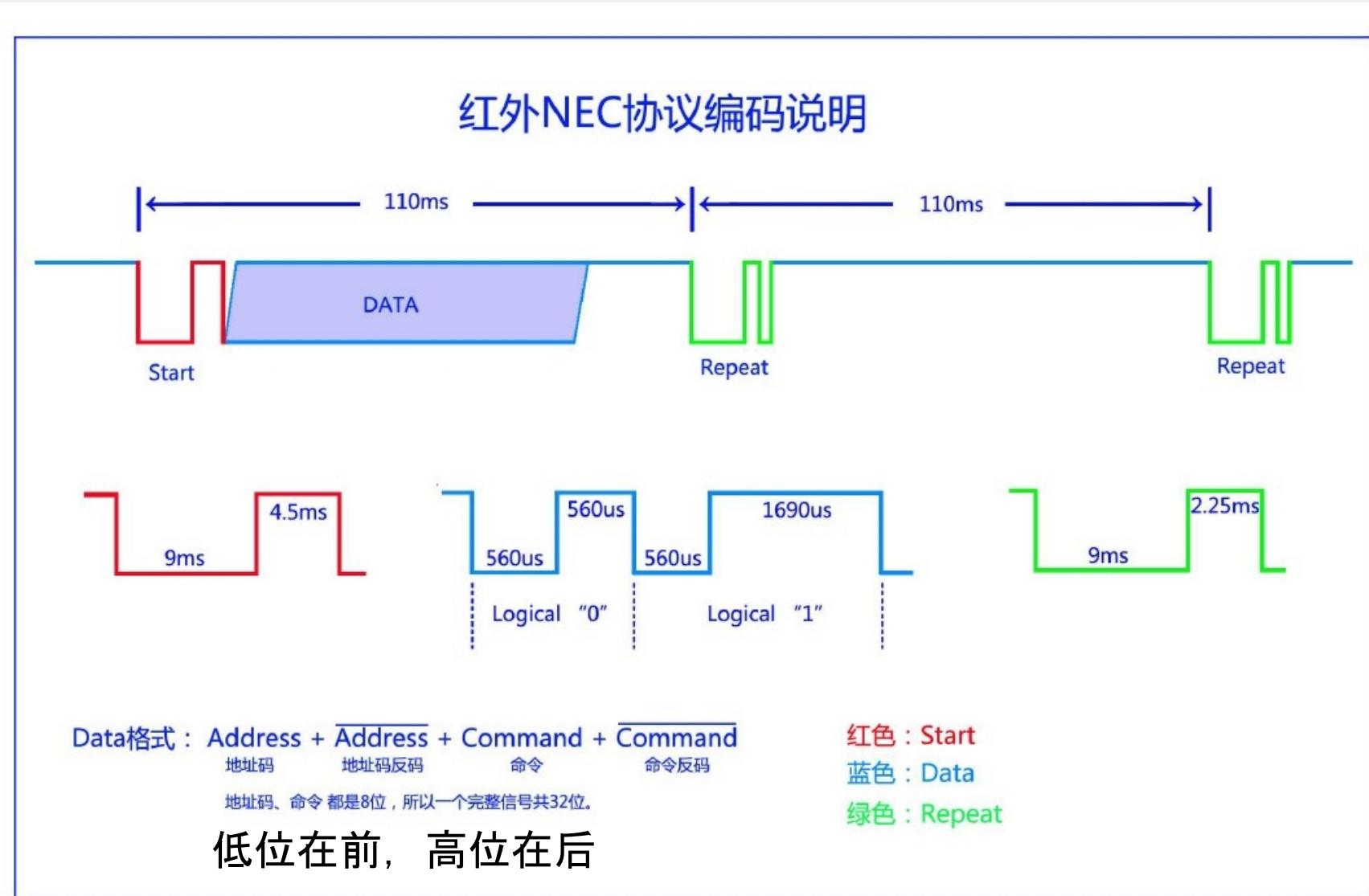


基本发送与接收

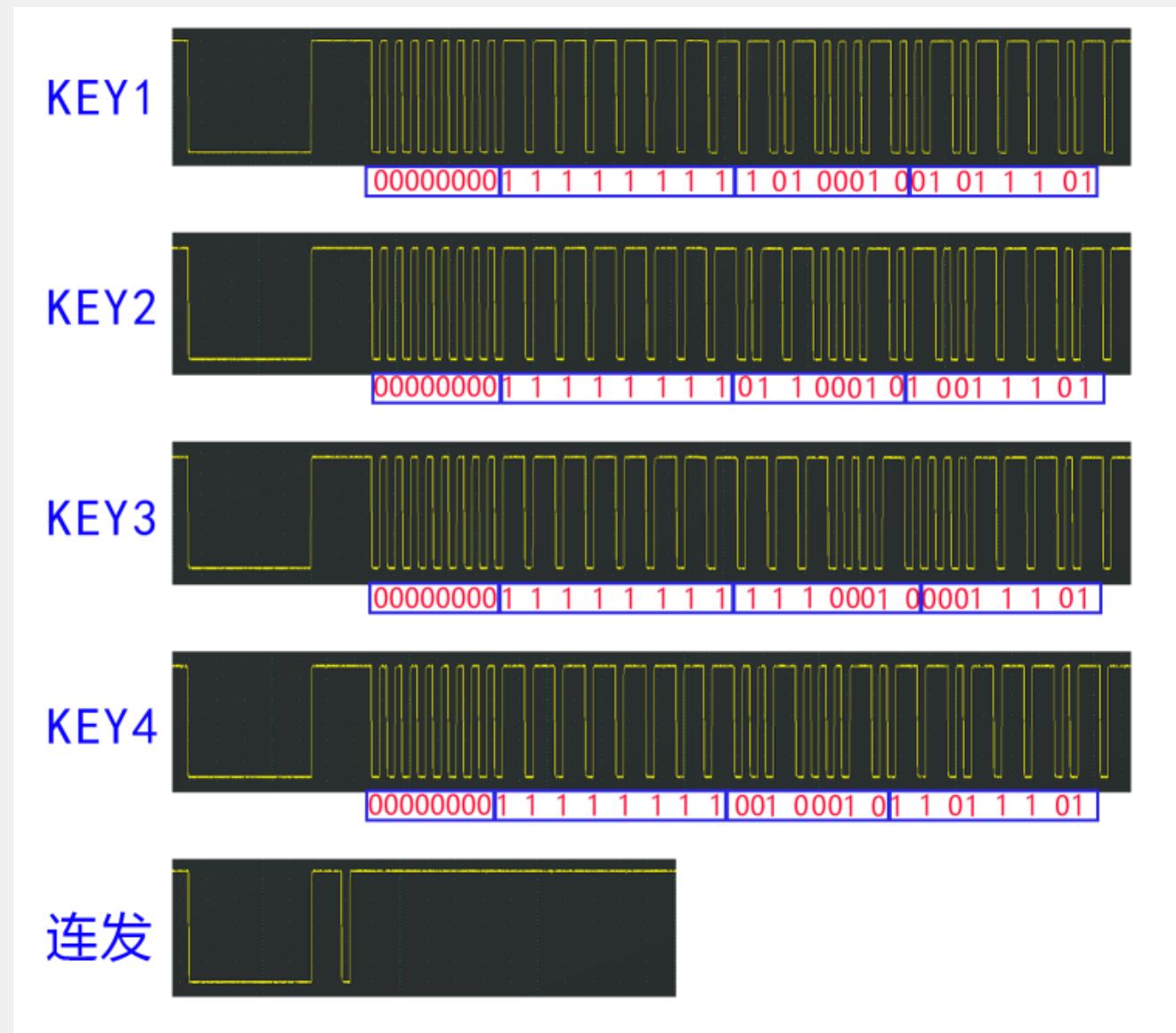
- 空闲状态：红外LED不亮，接收头输出高电平
- 发送低电平：红外LED以38KHz频率闪烁发光，接收头输出低电平
- 发送高电平：红外LED不亮，接收头输出高电平



NEC编码



NEC编码



遥控器键码

客户码：00FF
2102

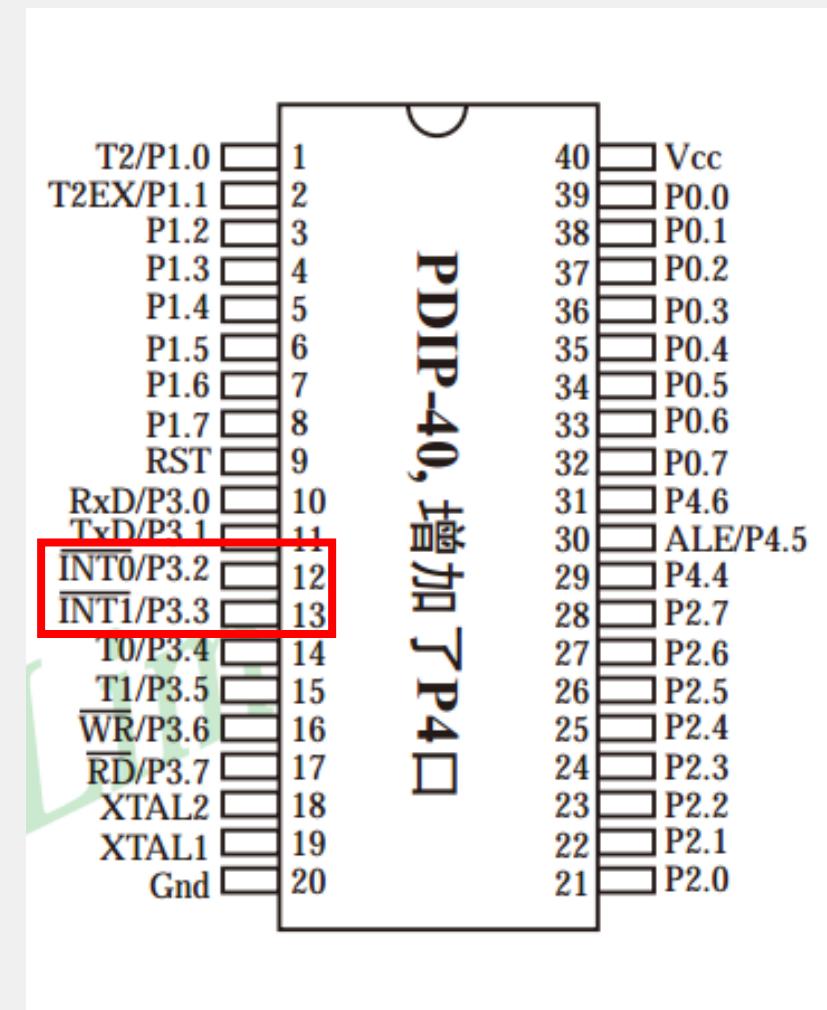


51单片机的外部中断

- STC89C52有4个外部中断
- STC89C52的外部中断有两种触发方式：
下降沿触发和低电平触发
- 中断号：

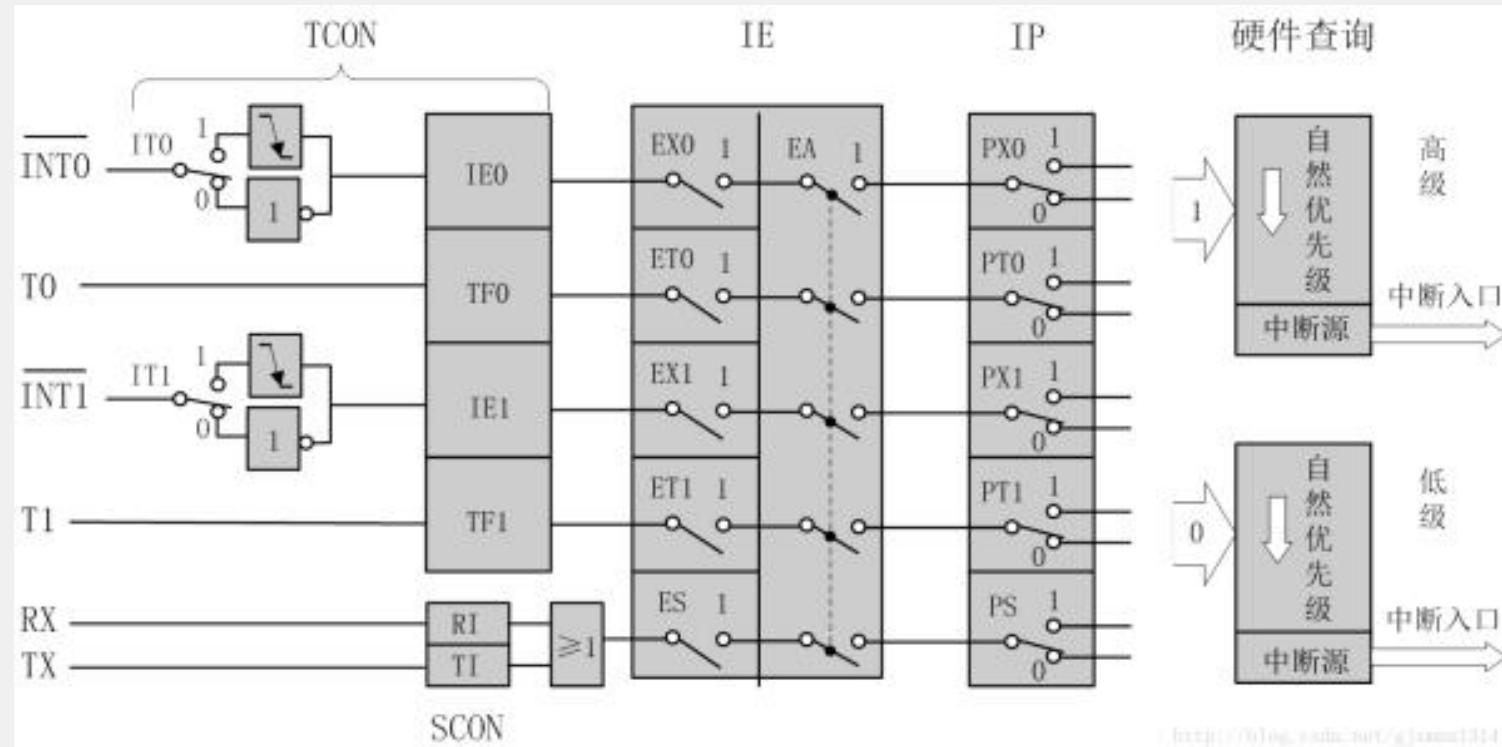
如果使用C语言编程，中断查询次序号就是中断号，例如：

```
void Int0_Routine(void)           interrupt 0;  
void Timer0_Routine(void)         interrupt 1;  
void Int1_Routine(void)           interrupt 2;  
void Timer1_Routine(void)         interrupt 3;  
void UART_Routine(void)           interrupt 4;  
void Timer2_Routine(void)         interrupt 5;  
void Int2_Routine(void)           interrupt 6;  
void Int3_Routine(void)           interrupt 7;
```



外部中断寄存器

符号	描述	地址	位地址及符号								复位值
			MSB				LSB				
IE	Interrupt Enable	A8H	EA	-	ET2	ES	ET1	EX1	ET0	EX0	0x00 0000B
IP	Interrupt Priority Low	B8H	-	-	PT2	PS	PT1	PX1	PT0	PX0	xx00 0000B
IPH	Interrupt Priority High	B7H	PX3H	PX2H	PT2H	PSH	PT1H	PX1H	PT0H	PX0H	0000,0000B
TCON	Timer/Counter 0 and 1 Control	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	0000 0000B



- END