电子系统设计基础

4.单片机基础-C51

张翠翠 西一楼520
西安交通大学 信通学院

CONTENTS

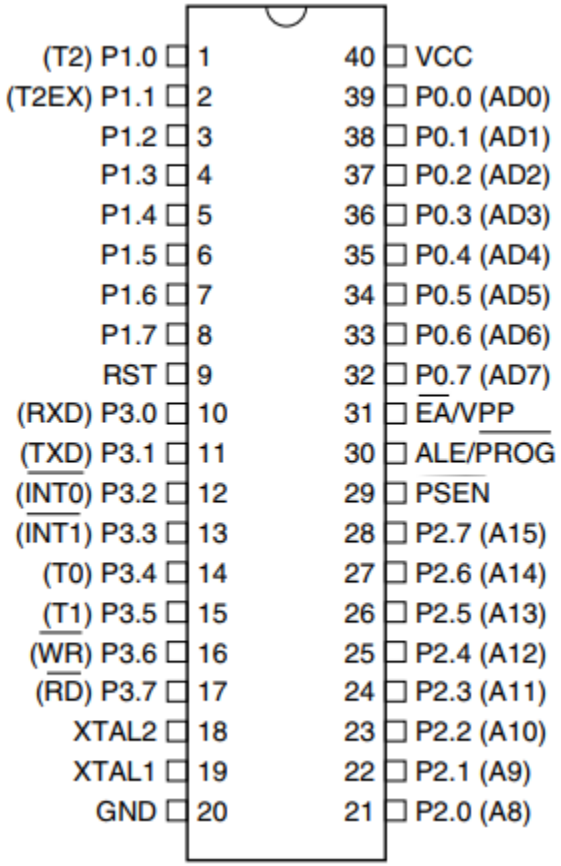
1

51单片机

51单片机

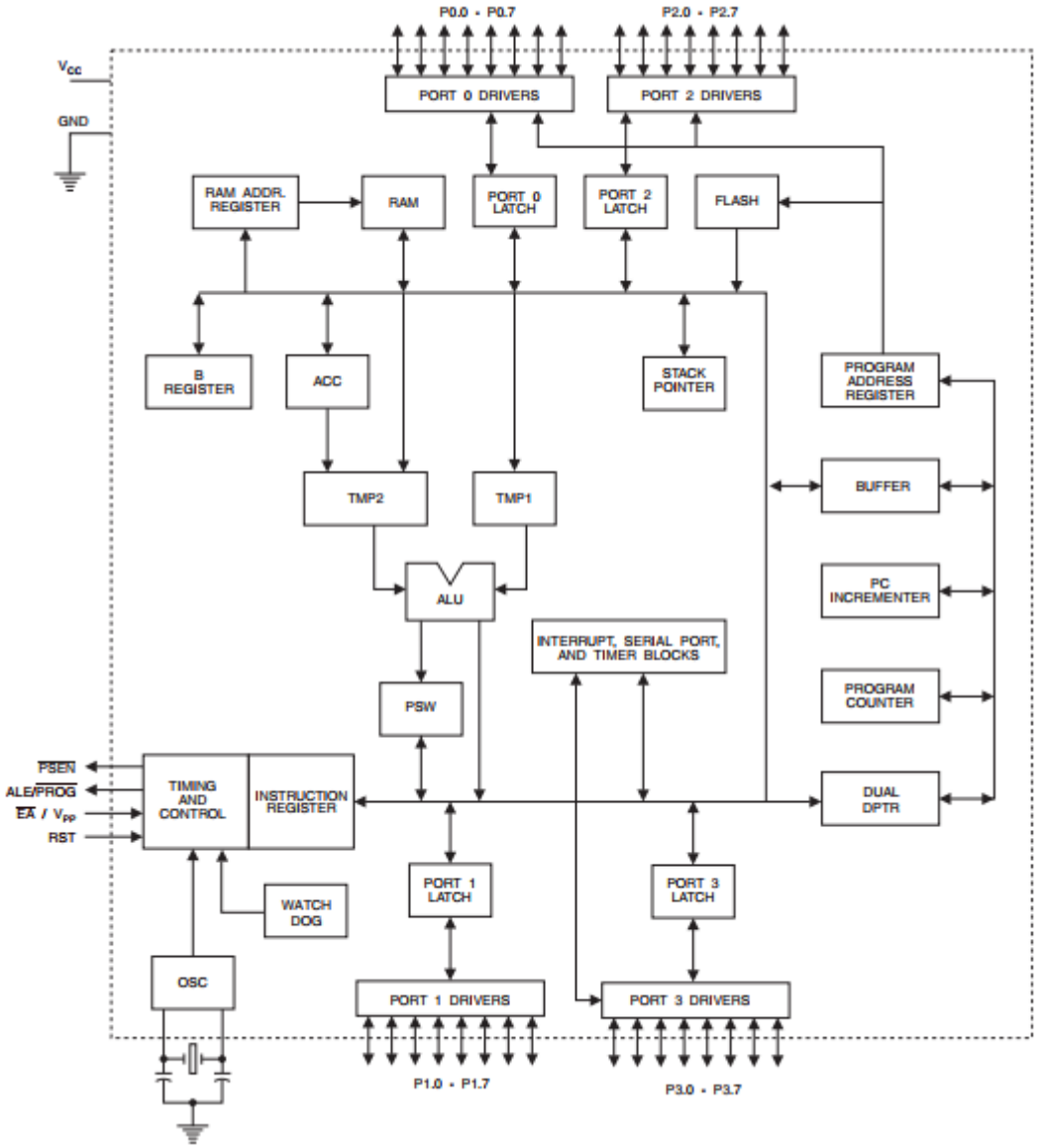
8位微处理器

8-bit Microcontroller Unit



51单片机

程序存储器里的指令
跟着系统时钟的节拍
一条一条执行



51单片机

AT89C51

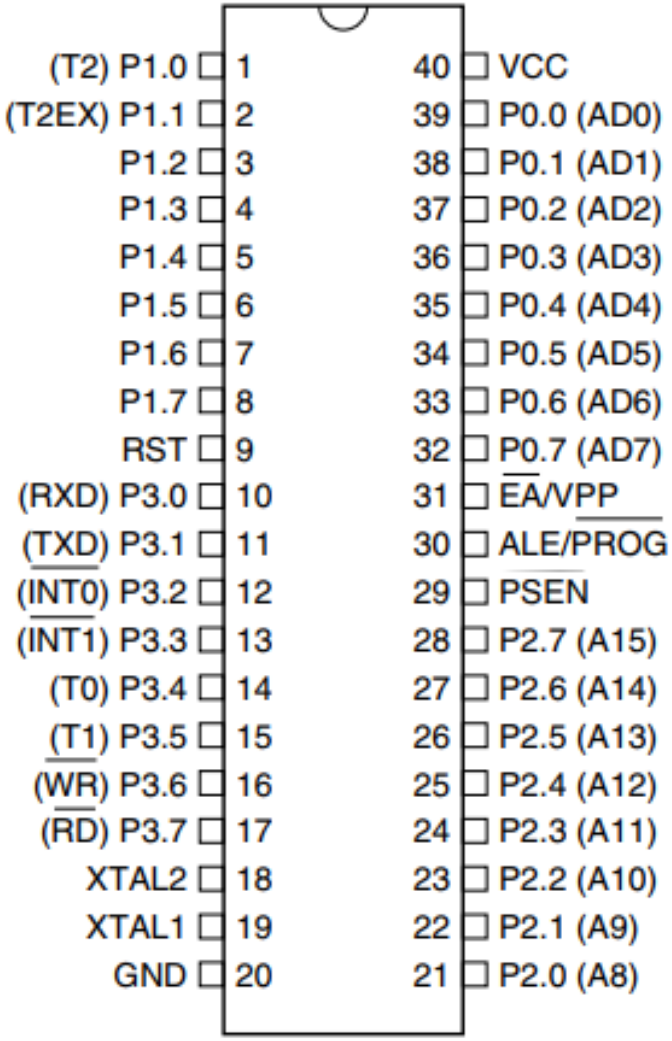
- **主电源引脚**

VCC(40脚)：接+5 V电源正端。

VSS(20脚)：接电源地端。

电源电压范围是4~5.5V，最高电源电压为6.6V。

任何引脚对地的电压范围是－1V~7V



27. Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	15.0 mA

51单片机 AT89C51

- **输入 / 输出引脚**

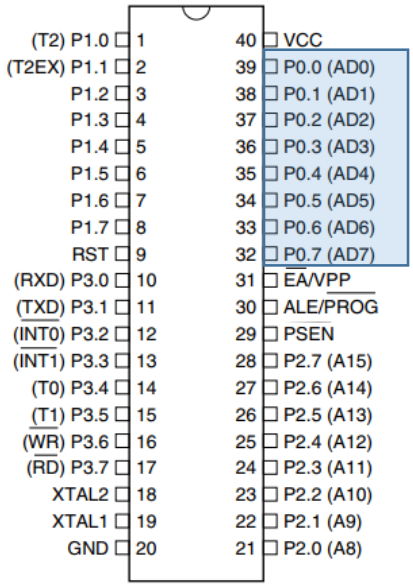
➢ P0口（P0.0~P0.7） 引脚39~32：8位的漏极开路结构，双向IO。

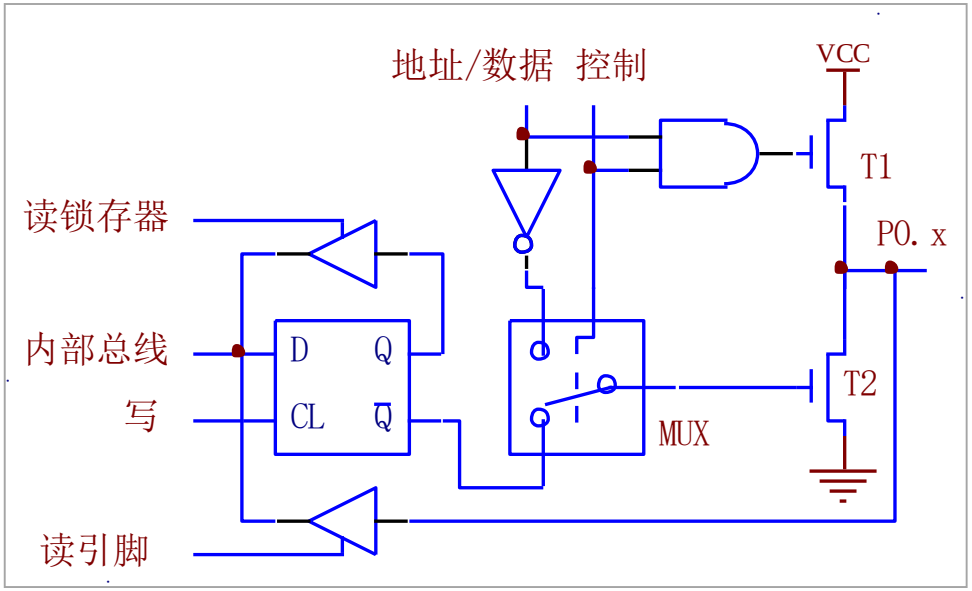
作为输出端使用，需要外接上拉电阻（在作为I/O口使用时，T1管夹断）；

若作为输入端使用，需要先将“1”写入端口（使T2管夹断）；

P0口可作为地址总线（A0-A7），也可作为数据总线（D0-D7）；

P0可驱动8个TTL，其它P口可以驱动4个TLL。（1个TTL负载为0.4mA)。





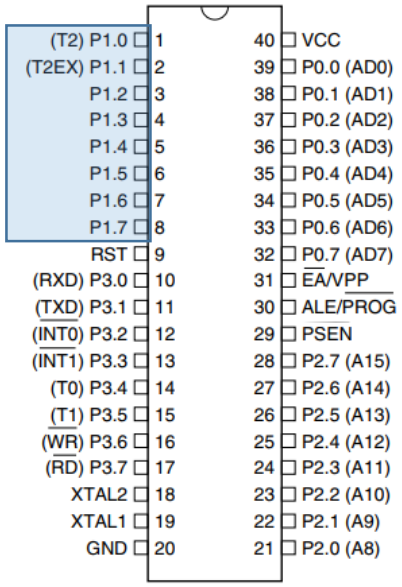
51单片机

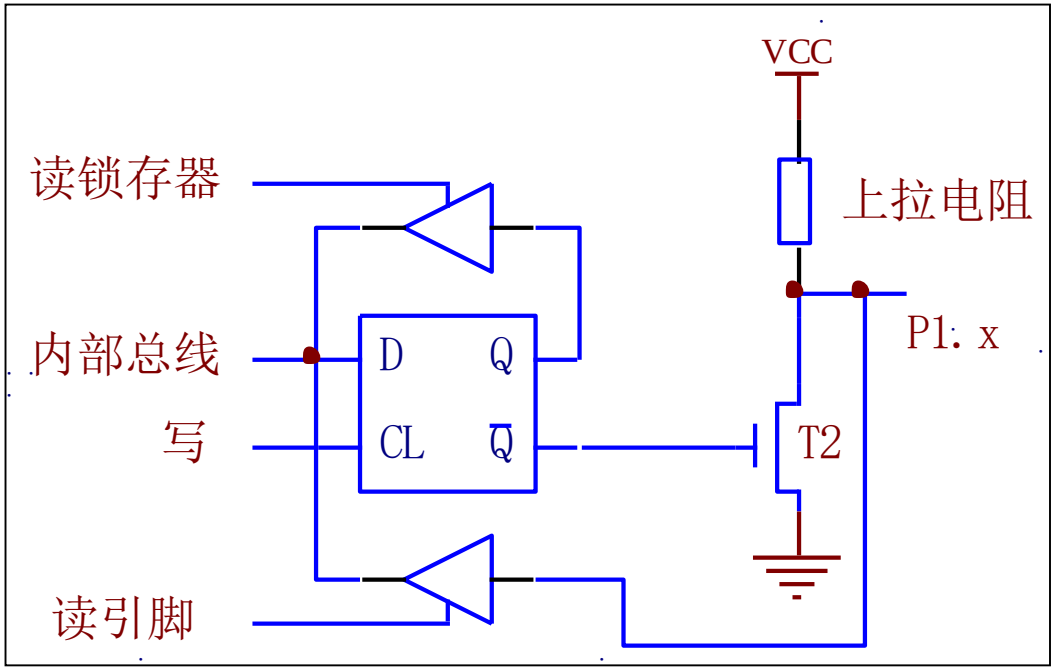
AT89C51

- P1口（P1.0~P1.7）(引脚1~8)：8位的双向IO，有内部上拉

  内部有上拉电阻，因此可以作为准双向I/O口使用。

  作为输入端使用时，需要先将“1”写入端口（使T2管夹断）；





Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)

51单片机

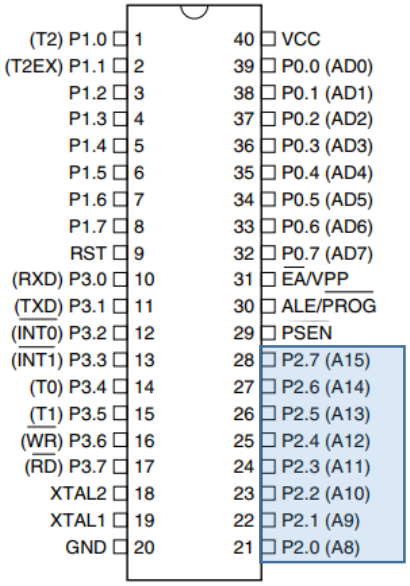
AT89C51

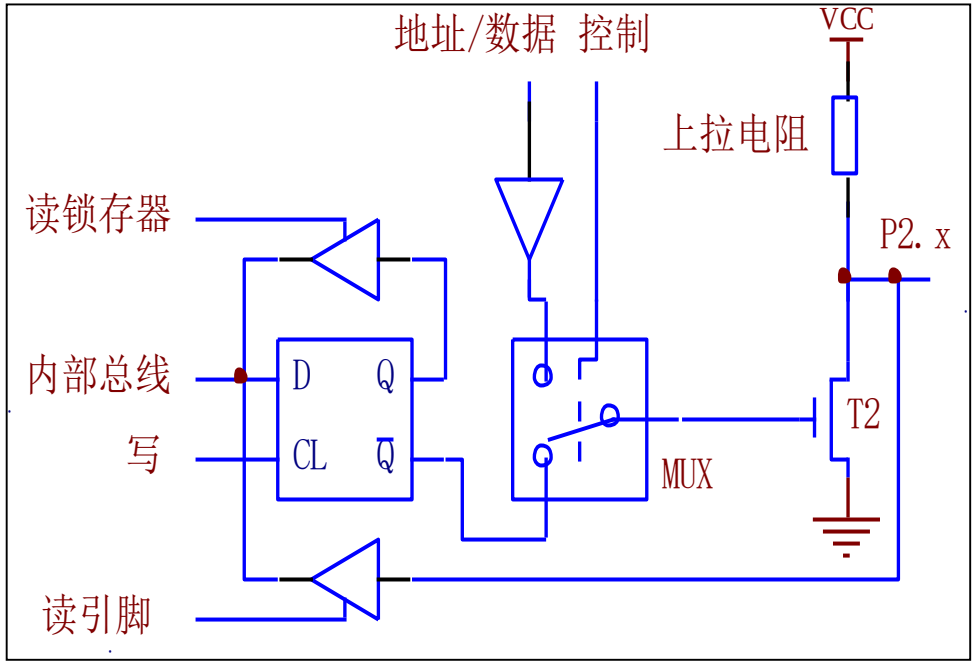
- P2口(P2.0~P2.7)引脚21~28：8位双向IO，内部有上拉。

  内部有上拉电阻，因此可以作为准双向I/O口使用。

  作为输入端使用时，需要先将“1”写入端口（使T2管夹断）；

  在接有片外存储器时，P2口作为高8位地址总线（A8-A15）；





51单片机

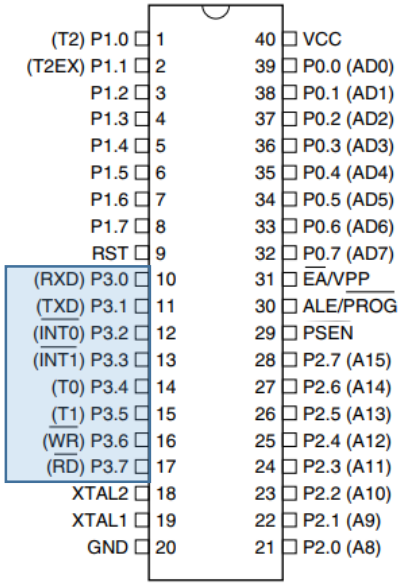
AT89C51

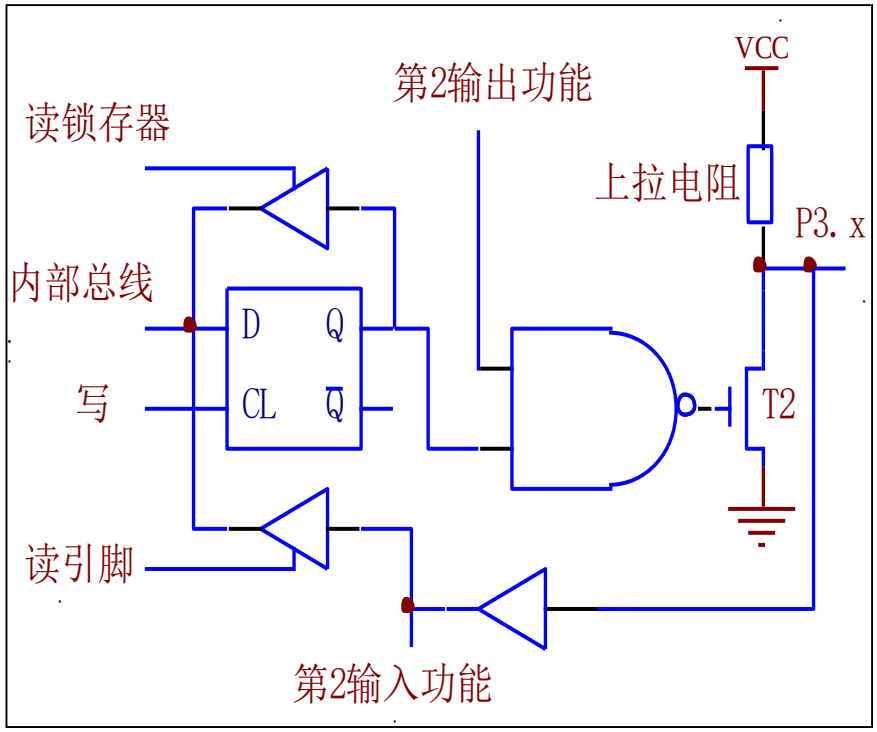
- P3口(P3.0~P3.7) 引脚10~17： 8位双向IO，内部有上拉

  P3口具有上拉电阻，可作为准双向输入输出端口使用；

  作为输入端使用时，需要先将“1”写入端口（使T1管夹断）；

  P3口的每个引脚还有第2功能.





51单片机

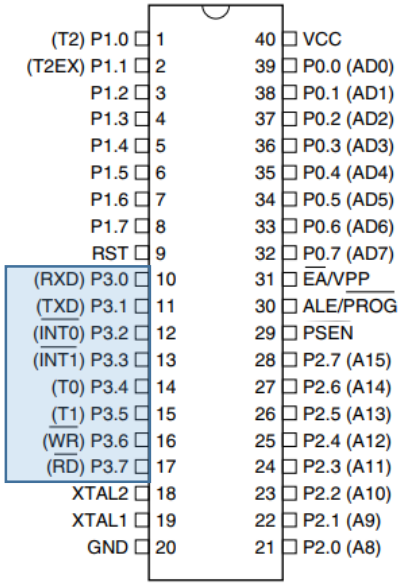
AT89C51

- P3口(P3.0~P3.7) 引脚10~17： 8位双向IO，内部有上拉

  P3口具有上拉电阻，可作为准双向输入输出端口使用；

  作为输入端使用时，需要先将“1”写入端口（使T1管夹断）；

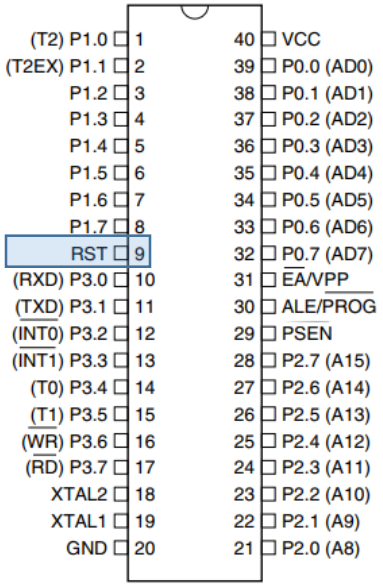
  P3口的每个引脚还有第2功能.

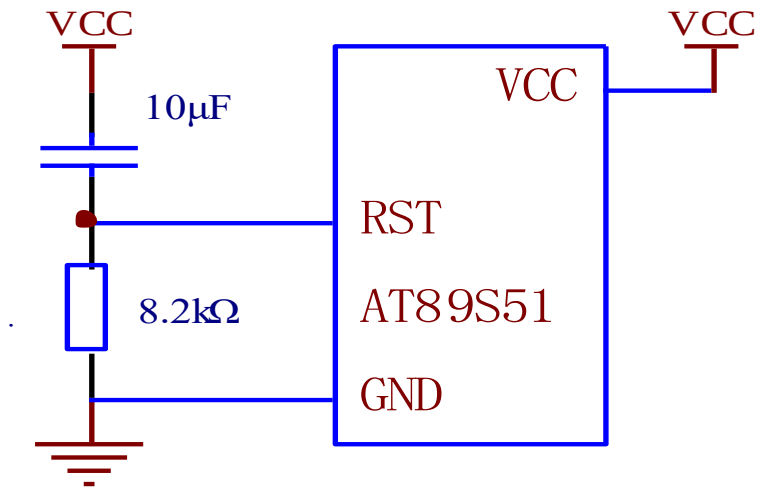


Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{\text{INT0}}$ (external interrupt 0)
P3.3	$\overline{\text{INT1}}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{\text{WR}}$ (external data memory write strobe)
P3.7	$\overline{\text{RD}}$ (external data memory read strobe)

- **RST 引脚：9号引脚**

复位输入。在振荡器运行时，有两个机器周期（24个振荡周期）以上的高电平出现在此引脚时，将使单片机复位，只要这个引脚保持高电平，51芯片便循环复位。复位后P3.0-P3.7口均置1，引脚表现为高电平，程序计数器和特殊功能寄存器SFR全部清零。当复位脚由高电平变为低电平时，芯片为ROM的00H处开始运行程序。复位操作不会对内部RAM有所影响。





51单片机

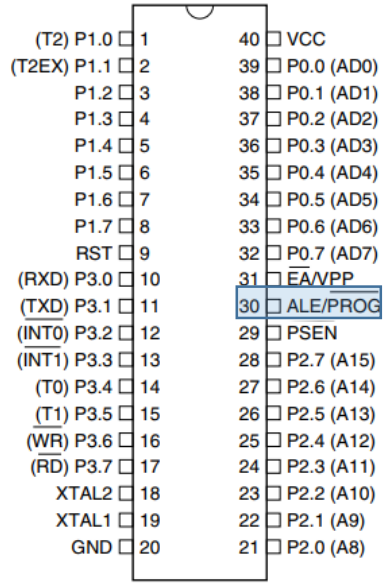
AT89C51

- **ALE/PROG引脚：30号引脚**

ALE：Address Latch Enable 高有效，用于锁存地址的低8位字节。

/PROG：编程引脚，低有效，在FLASH编程期间，此引脚用于输入编程脉冲。

在平时，ALE端以不变的频率周期输出正脉冲信号，此频率为振荡器频率的1/6。因此它可用作对外部输出的脉冲或用于定时目的。



51单片机

AT89C51

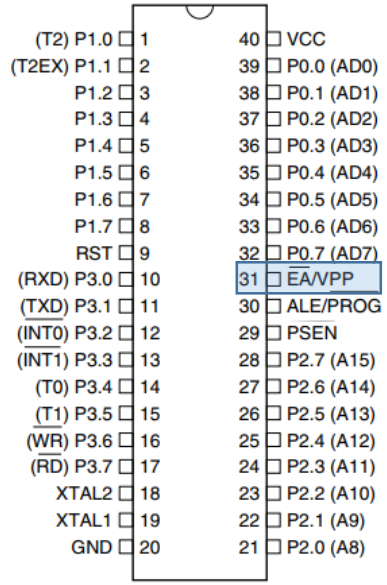
- **EA/VPP引脚：31号引脚**

EA：External Access Enable 外部访问使能

当EA为GND时，访问外部程序存储器（0000H-FFFFH）

当EA为高电平时，访问内部程序存储器

在FLASH编程期间，此引脚用于施加12V的编程电源（VPP）



51单片机

AT89C51

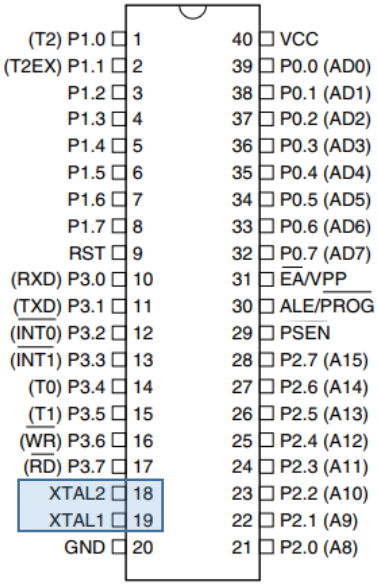
- **外接晶体引脚XTAL1（19）和XTAL2（18）**

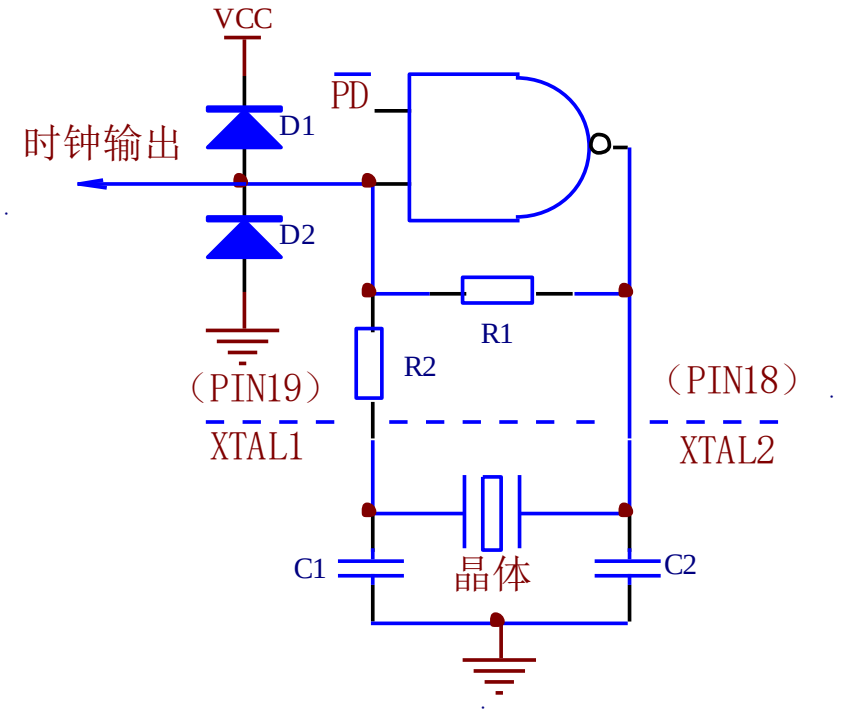
石英晶体与内部的反相器组成Pierce振荡器，接线如图所示。

单片机可使用外部时钟信号也可使用内部时钟

当使用内部时钟时，此二引线用于外接石英晶体和微调电容

当使用外部时钟时，用于接外部时钟信号，NMOS接XTAL2，CMOS接XTAL1





28. DC Characteristics

The values shown in this table are valid for T_A = -40°C to 85°C and V_{CC} = 4.0V to 5.5V, unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC}-0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC}-0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC}+0.9	V_{CC}+0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC}+0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low-voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, V_{CC} = 5V ±10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pull-down Resistor		10	30	kΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode[1]	V_{CC} = 5.5V		100	µA

29.1 External Program and Data Memory Characteristics

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	33	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-25		ns
t_{LLAX}	Address Hold after ALE Low	48		t_{CLCL}-25		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-25		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-45		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-60	ns
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		59		t_{CLCL}-25	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-80	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold after $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float after $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-30		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-130		ns
t_{WHQX}	Data Hold after $\overline{WR}$	33		t_{CLCL}-25		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-25	t_{CLCL}+25	ns

51单片机

AT89C51

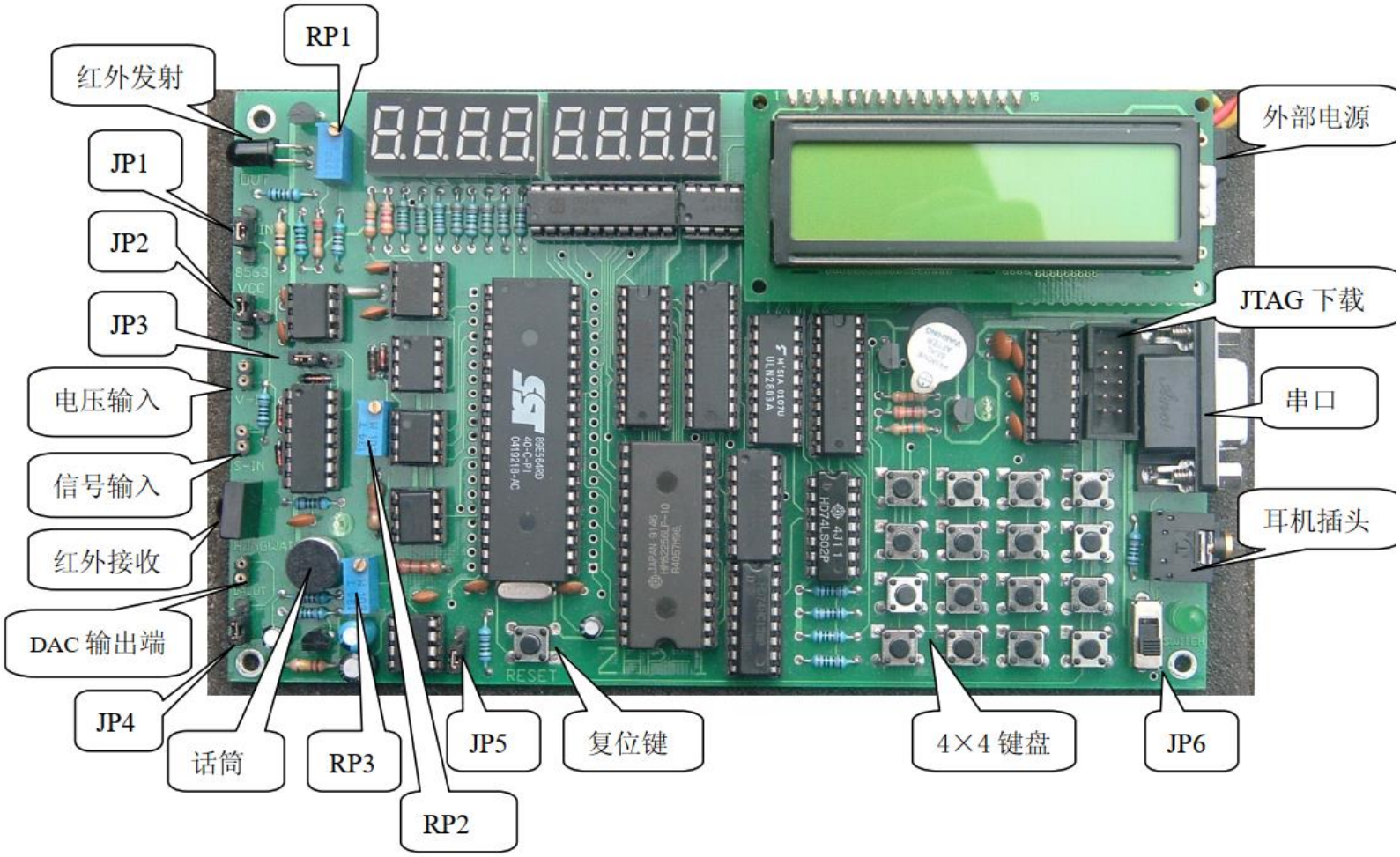
Features

- Compatible with MCS®-51 Products
- 32K Bytes of Reprogrammable Flash Memory
- Endurance: 10,000 Write/Erase Cycles
- 4V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 512 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Hardware Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Green (Pb/Halide-free) Packaging Option

2

51系统板

51系统板


RP1
红外发射
JP1
JP2
JP3
电压输入
信号输入
红外接收
DAC 输出端
JP4
话筒
RP3
RP2
JP5
复位键
4×4 键盘
JP6
外部电源
JTAG 下载
串口
耳机插头

51系统板

- 以SST89E564RD 51单片机为核心

◆ 外设:

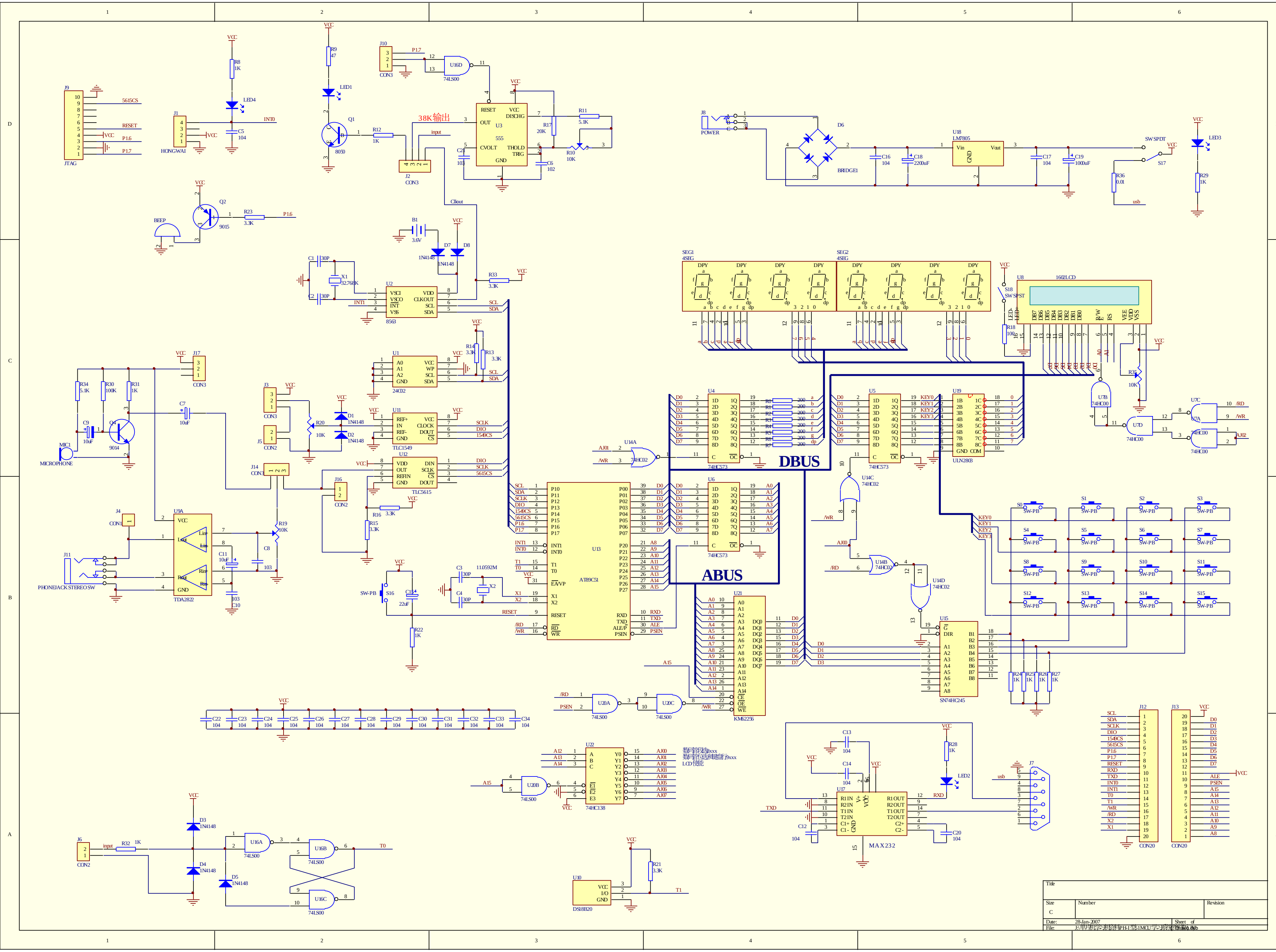
- 并行总线驱动的8位7段数码管
- 4X4 键盘
- 字符型LCD（LC1621）
- 外扩32K RAM（KM62256）
- SPI总线驱动的10位ADC（TLC1549）、10位DAC（TLC5615）
- I2C总线驱动的外扩E2PROM（AT24C02）
- 日历时钟芯片（PCF8563）
- 单总线（1-WIRE）驱动的数字式温度传感器DS18B20
- MAX232的串口通讯
- 基于TDA2822M放大MIC的音频功放
- 红外线发射和接收装置
- 555信号发生和整形电路
- JTAG下载接口
- 蜂鸣器

51系统板

- 以SST89E564RD 51单片机为核心

◆ 仿真调试

- 通过串口与PC机连接，因此仿真时需要设置相应的串口号
- 支持KEIL C51开发仿真环境，64K用户可使用的仿真程序空间，0x0000-0xFFFF，监控程序存储在特殊空间，不占用0x0000-0xFFFF的64K仿真空间
- 全保留单片机特性，避免仿真正常而实际烧录芯片却不正常的问题
- RS-232通讯接口，波特率4.8Kbps-57.6Kbps自适应
- 仿真频率0-40MHz晶振可选，系统配置11.0592MHz
- 程序代码可以重复装载，无需预先擦除用户程序空间
- 监控程序占用用户的资源少，全速运行不占用资源
- 片内64K程序空间可以随时进行在线程序更新，可以调试长达几千行的智能控制、键盘控制汉字液晶显示等大型程序
- 可单步、断点、全速、可查看参考变量、RAM变量
- 支持汇编、C语言、C语言和汇编混合调试
- 支持同时最多10个断点。



JTAG
HONGWAI
INT0
38K输出
555
CON3
POWER
BRIDGE1
LM7805
SW SPDT
SEG1
4SEG
SEG2
4SEG
1602LCD
8563
32.768K
24C02
TLC1549
TLC5615
TDA2822
MICROPHONE
PHONEJACK STEREO SW
74HC573
ULN2803
DBUS
ABUS
AT89C51
KM62256
74HC02
74HC00
74LS00
SN74HC245
74HC138
数码管段选8xxx
数码管位选和键盘行9xxx
LCD使能
MAX232
DS18B20
CON20
Title
Size
C
Number
Revision
Date: 28-Jan-2007
Sheet of

51系统板

■ 外设地址

- **7段数码管**
  段选信号地址0x9000，位选信号地址0x8000，均为高电平有效

- **4x4键盘**
  0x8000H，　**/WR**控制行，　**/RD**控制列

- **字符型LCD（LC1621）：ASII码写入的16字2行的字符型LCD**
  内部显示器第一行地址：0x80~0x8F
  内部显示器第二行地址：0xC0~0xCF

- **外挂存储器（KM62256）**
  8条数据线（D0~D7），15条地址线（A0~A14）
  A15连接到62256的片选信号，低电平时选择62256
  地址范围：0x0000H~0x7FFFH

3

51单片机程序设计

51单片机开发流程

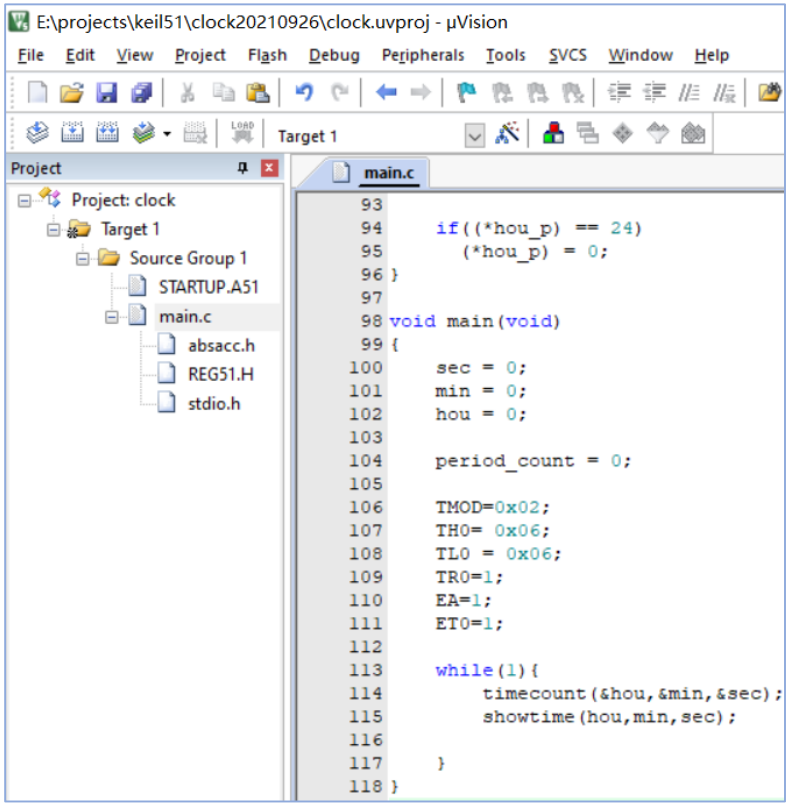
- **工具：Keil-C51版本（51单片机的IDE开发环境）**
  keil-C51的安装过程.pdf
  Keil-C51新建工程.pdf

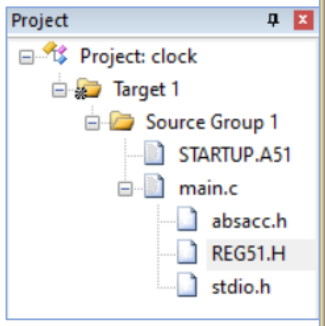
- **语言：C语言、汇编语言、或者两者混合**

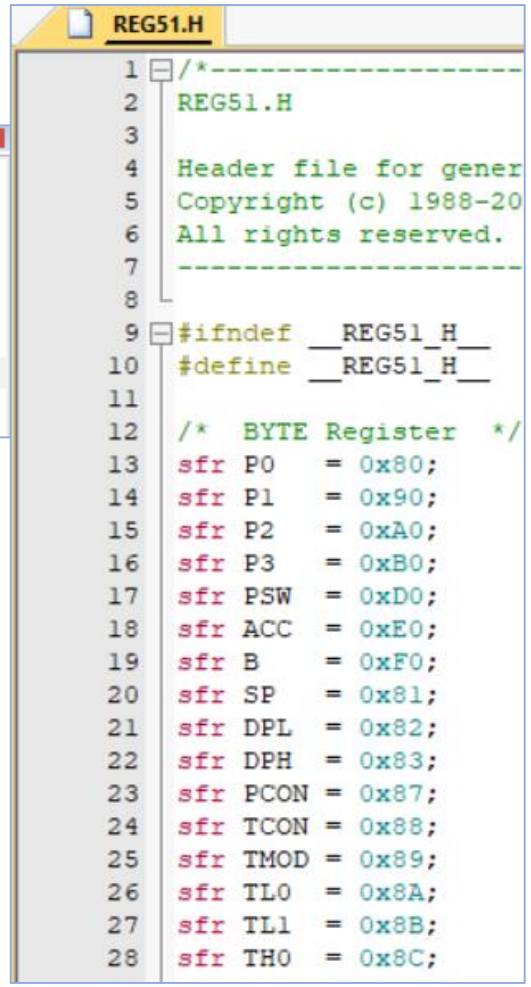
- **开发流程：新建工程、添加设计文件、编译、调试下载**

51单片机程序设计

- 入口为main函数，main函数中以死循环结尾

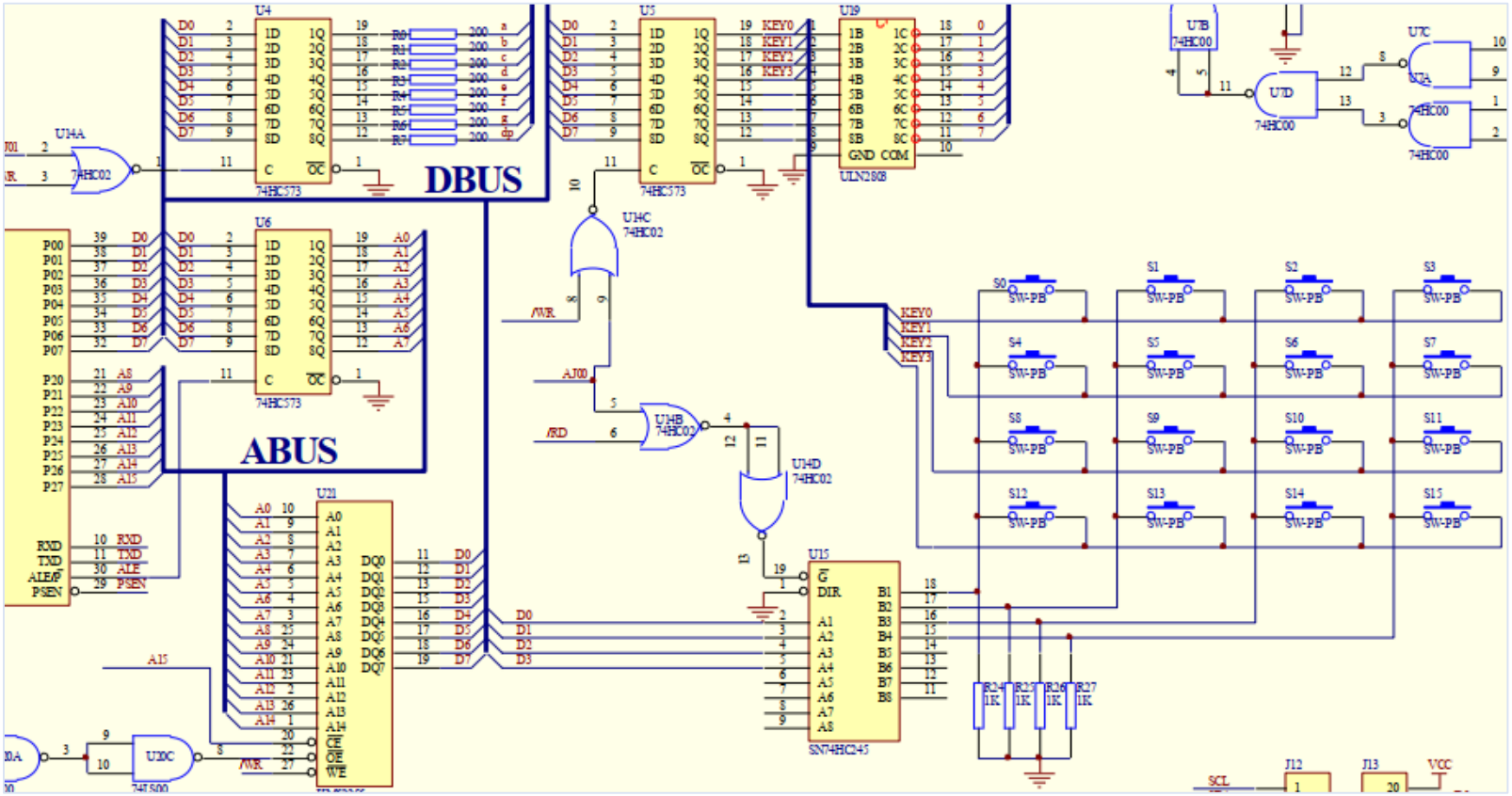






51单片机程序设计

■ 键值获取



4x4键盘地址：0x8000

51单片机程序设计

■ 键值获取

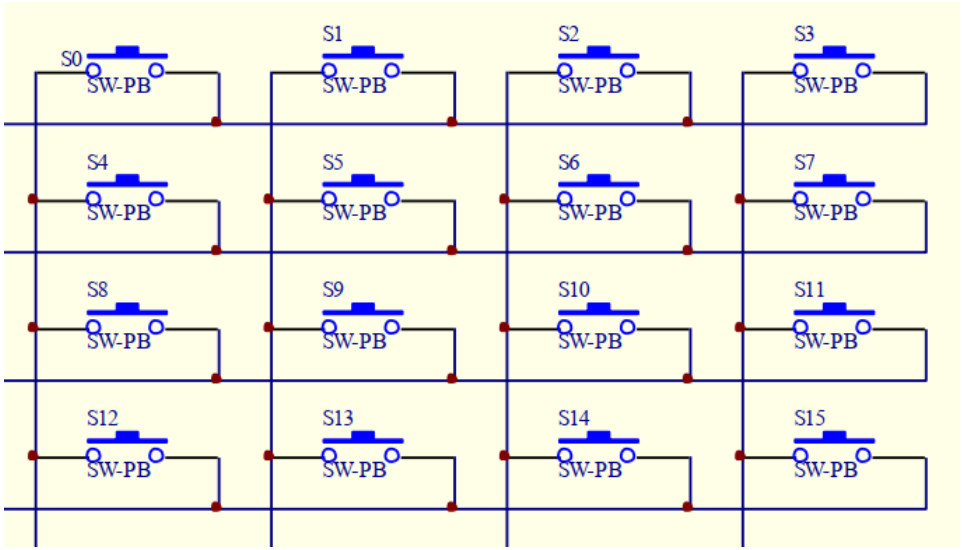
```
/*****************************************************************************/
unsigned char getkeycode(void)          //键盘扫描函数，返回获得键码
{
  unsigned char line=0x00;              /*行码*/
  unsigned char col=0x00;               /*列码*/
  unsigned char scancode=0x01;          /*行扫描码*/
  unsigned char keycode;                /*键号*/

  XBYTE[0x8000]=0xff;                   //和数码管IO复用？
  col=XBYTE[0x8000]&0x0f;               /*从列端口读入四位列码*/
  if (col==0x00) keycode=0x00;
  else
  {
   while((scancode&0x0f)!=0)            /*取scancode的低四位，没变为全0，循环*/
    {
      line=scancode;                    /*行号*/
      XBYTE[0x8000]=scancode;           /*给行赋扫描码，第一行为0x01*/
      if((XBYTE[0x8000]&0x0f)==col)     /*检测按键所在的行跳出循环*/
          break;
      scancode=scancode<<1;             /*行扫描码左移一位，转下一行*/
    }
    col=col<<4;                         /*把列码移到高四位*/
    keycode=col|line;
  }
 return keycode;
}
```

思考:
1）消抖

51单片机程序设计

■ 键值获取



Keycode = 0x11, S0
Keycode = 0x21, S1
Keycode = 0x41, S2
Keycode = 0x81, S3

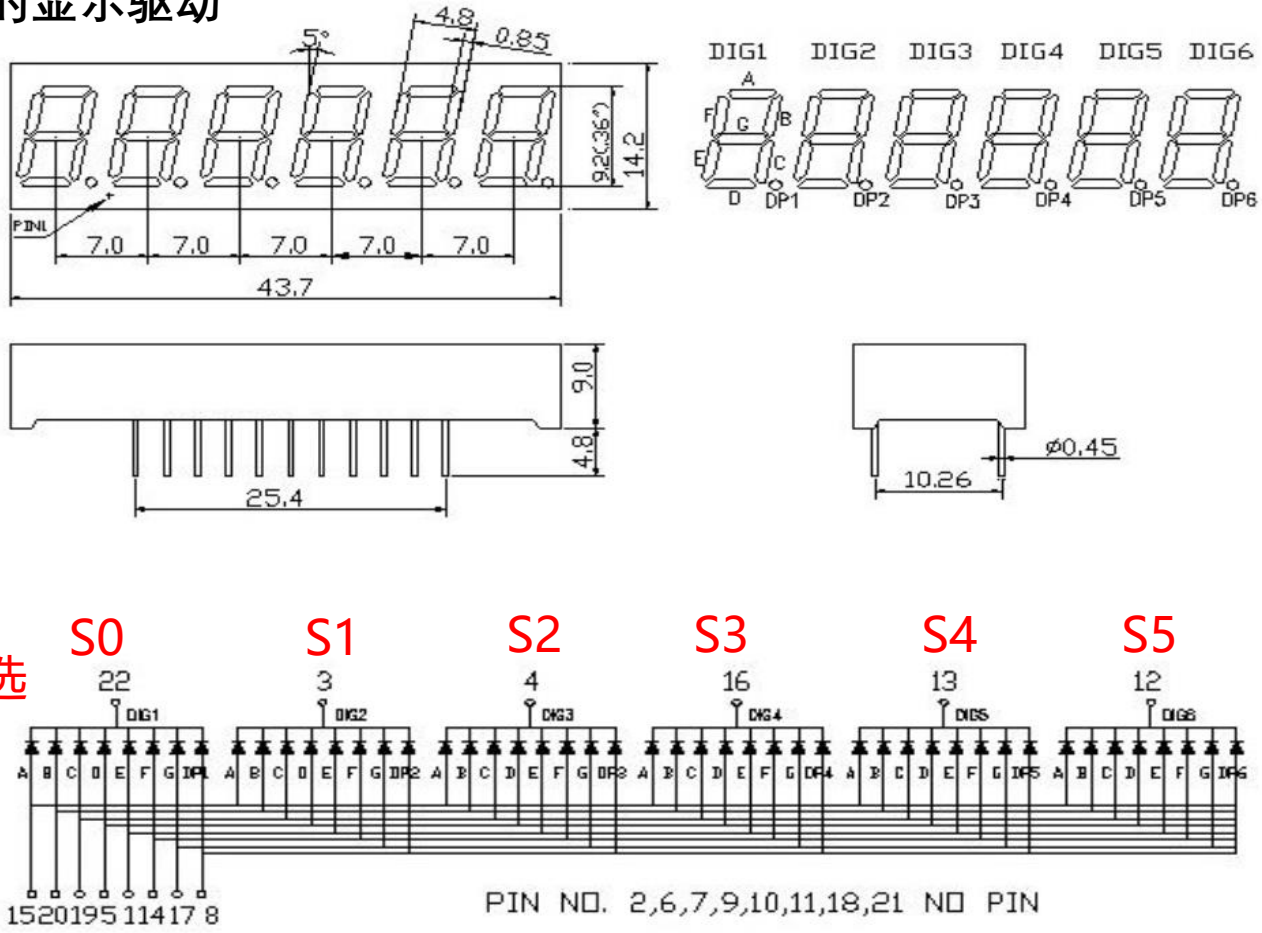
Keycode = 0x12, S4
Keycode = 0x22, S5
Keycode = 0x42, S6
Keycode = 0x82, S7

Keycode = 0x14, S8
Keycode = 0x24, S9
Keycode = 0x44, S10
Keycode = 0x84, S11

Keycode = 0x18, S12
Keycode = 0x28, S13
Keycode = 0x48, S14
Keycode = 0x88, S15

51单片机程序设计

- 7段数码管的显示驱动

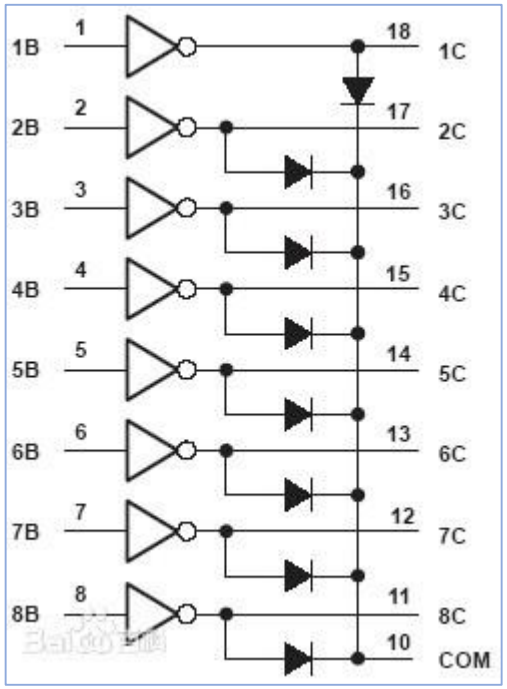



```
unsigned char code led_table[17]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f,0x77,0x7c,0x39,0x5e,0x79,0x71,0x00};//灭(16)
```

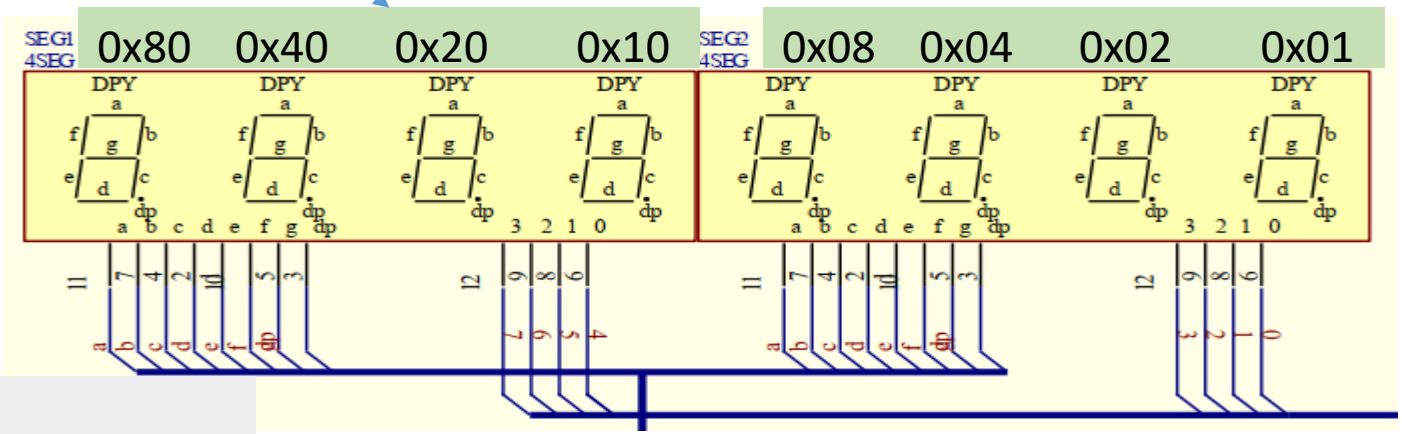
51单片机程序设计

- 7段数码管的显示驱动

```
/*****************************单管显示控制*************************/
void SEG7_show(unsigned char whichseg,unsigned char whichnum)
{
  XBYTE[0x8000]=whichseg;      //选管
  XBYTE[0x9000]=0x00;          //消影
  delay_f(10);
  XBYTE[0x9000]=led_table[whichnum]; //选段显示特定数字
}
```



ULN2803



思考：

1）怎么实现扫描显示?

2）怎么实现同时显示的效果?

4

实验任务

单击添加文本

- **实验名称：流水灯**

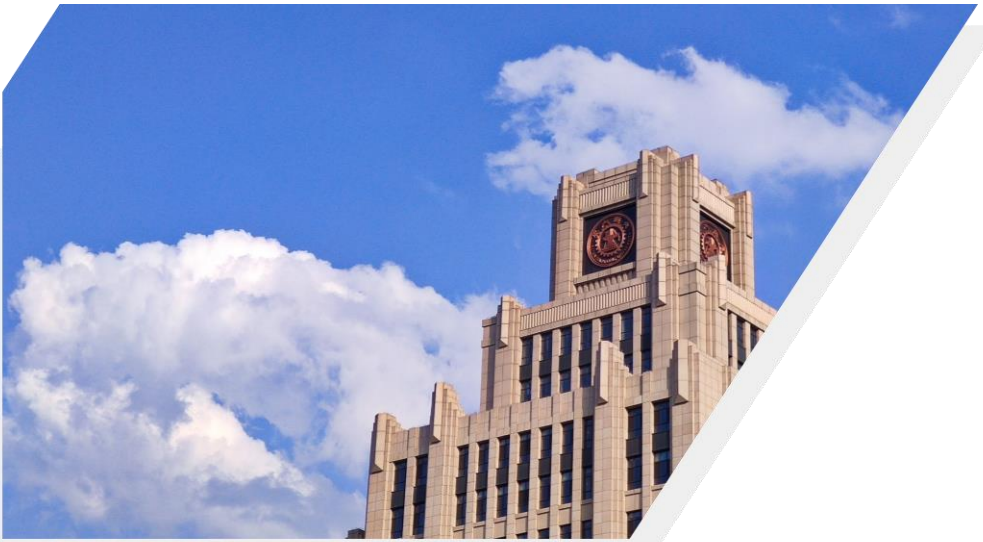
1. 8个7段数码管以流水灯的形式显示当前按键值
2. 8个7段数码管呈现同时显示当前按键值的效果
3. 8个7段数码管滚动显示自己学号后8位
4. 其它

```
void main(void)
{
   //初始化

   while(1)
   {
    //检测按键值

    //将键值扫描显示到7段数码管上
   }
}
```

西安交通大学
XI'AN JIAOTONG UNIVERSITY



谢谢大家