

Quiz 1 - 1 Interrupt Number

1) An interrupt service subroutine shown as following is for which interrupt source?

void InterruptSubroutine(void) _interrupt 3

- a. INT0
- b. INT1
- c. Timer0
- d. Timer1
- e. Serial Port

中断名称(且基本不会被引用→自动启动)

编号

• Interrupt Function in C.

Interrupt Number	Description
0	INT0
1	Timer0
2	INT1
3	Timer1
4	Serial Port

⇒ Interrupt Sources.

- * External Interrupt 0 (INT0)
- * Timer/Counter 0 Overflow (TFO)
- * External Interrupt 1 (INT1)
- * Timer/Counter 1 Overflow (TFO)
- * Serial Port
 - Data Transmited (TI)
 - Data Received (RI)

• External INT.

启动方式: 触发中断引脚 (见 P1-1-1)
(接 P3.4)

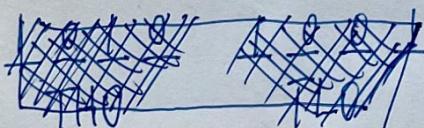
然后进入中断程序。

• Timer.

启动方式: 中断计时器从初始值经过每 $\frac{1}{256}$ 秒
+1, 当加到 $16 = 65536$ 时, 执行
中断程序。

• Summary.

interrupt	0	INT0
	1	Timer0
	2	INT1
	3	Timer1
	4	Serial Port



初值 10110110 00111010 (46650)

TH0 TL0

+1 / $\frac{1}{256}$ s

$$TL0 = (65536 - 46650) \% 256;$$

$$TH0 = (65536 - 46650) / 256;$$

$$TL0 = 256 - 250$$

$$TH0 = 256 - 250$$

11111111 11111111 ⇒ 00



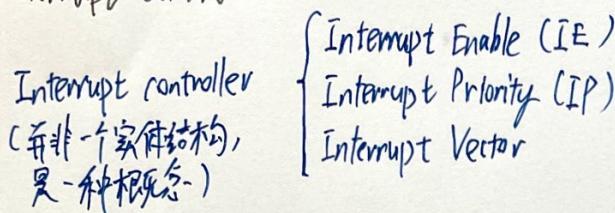
扫描全能王 创建

Quiz 1-2 The function of interrupt controller.

2) Which of the following is the task of an interrupt controller?

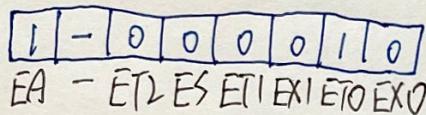
- a. Response the interrupt requests 回应中断请求
- b. Handle the interrupt priority 处理优先级
- c. Provide the interrupt service subroutine 提供中断服务子程序
- d. All of above
- e. None of above

• Interrupt controller



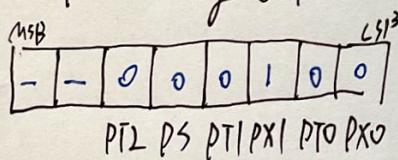
• Interrupt Control Register

Interrupt Enable (IE)



因为:
TMOD = 0x01;
TH0 = 0;
TL0 = 0;
TR0 = 1;
ET0 = 1; EA = 1;

• Interrupt Priority (IP)



将 External Interrupt 1 (INT1) 的优先级提高。

• Interrupt Vector (中断向量是中断处理程度的内存位置——指针)

Interrupt Source Vector Address

External 0 (INT0)	IE0	0003H
Timer 0	TF0	000BH
External 1 (INT01)	IE1	0013H
Timer 1	TF1	001BH
Serial Port	RI or TI	0023H
Timer 2	TF2 or EXF2	002BH < System Reset RGT 0000H

Note: When vectoring to an interrupt the flag that caused the interrupt is automatically cleared by hardware.

中断执行后，中断标志自动消除。等待下次发生

例: RI, TI, TF2, EXF2

because there are two possible sources.

Solution:

① be tested in ISR

② cleared by software

• Summary

The function of interrupt controller is Control.



扫描全能王 创建

Quiz 2-1 Clear the interrupt flag. A serial port interrupt

- 1) You don't need to clear the interrupt flags by software in an 8051 except for the
a. INT0 b. INT1 c. Timer0 d. Timer1 e. Serial Port

cleared by the hardware automatically

and (Timer 2)

ISR - Interrupt Service Routines.

中断过程:

(1) 保持被中断进程现场，然后中断处理结束后能够使进程准确地返回到中断点，系统必须保存当前处理器程序状态字PSW和程序计数器PC的值。

(2) 分析中断原因，转去执行相应的中断处理程序，在新中断请求到来时，处理器先从最高的中断源发出的中断请求。

(3) 恢复被中断进程的现场，CPU继续执行原来被中断的进程。

实现这一功能的系统称为中断系统。

申请CPU中断的请求源称为中断源(interrupt source)

- ① tested in the ISR
② cleared by the software



扫描全能王 创建

Quiz 2-2 serial communication

- 2) Which statement of the following is **incorrect** about the communication between two 8051s through their serial ports?
- The **bit rates** of these two 8051s must be the same.
 - The modes of these two 8051s serial ports must be the same.
 - C** The system clock frequencies of these two 8051s must be the same.
 - The data bit numbers of these two 8051s must be the same.
 - The stop bit lengths of these two 8051s must be the same.

• bit rates.

每秒传送的比特数，bps. bit per second

• SCON: Serial Port Control Register

0	1	0	1	0	0	0	0
SMD0	SMD1	SMD2	REN	TB8	RB8	TI	RI

※

SMD0	SMD1	Mode	Description	Bound Rate
0	0	0	Shift Register	Fosc/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fosc/Fosc/12
1	1	3	9-bit UART	Variable

mode 0 $\frac{Fosc}{12}$ Baud Rate USART (Universal Asynchronous Receiver Transmitter)

mode 1,3 $\frac{2^{SMD0}}{32} \times \frac{Fosc}{12 \times (156-TH1)}$ 异步：双方不需要共同的时钟

mode 2 $\frac{2^{SMD0}}{64} \times Fosc$ 10bit Stop bit

位数 D0 D1 D2 D3 D4 D5 D6 D7 止位

SM2 多机通信控制 (主/从) mode 2,3)

REN 允许串行接收位 REN=1: 接收
REN=0: 不接收

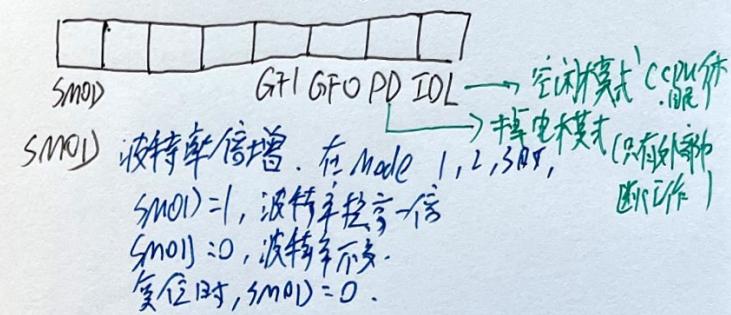
TB8 mode 2,3 中 发送数据第九位(校验位)

mode 0,1 中 未用 (0)

RB8 mode 2,3 中 接收数据第九位(校验位)

mode 1 中 SM2=0 RB8 为接收的停止位。

※ PCON: Power Control Register



波特率与比特率

比特率: 每秒传输的 [信息量] bit \Rightarrow bits/s

波特率: 每秒传输的 [码元] bit \Rightarrow bits/s

$$I = S \cdot \log_2 N$$

比特率 波特率

1 0 1 0 0 1 0 0 0 1

$$10 = 5 \cdot \log_2 4 \rightarrow \boxed{\quad \quad}$$

$$\downarrow \quad \quad \quad \begin{matrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{matrix} \quad 2^2 = 4$$

代表 N 位 M 位

TI 发送中断标志位

发送停止时，由内部硬件将 TI 置 1。

进入中断

在中断服务程序中，必须用软件清 0。

RI 接收中断标志位 功能同上

$TI=0$ 未发送结束
 $TI=1$ 缓待清 0.
 $TI=0$



扫描全能王 创建

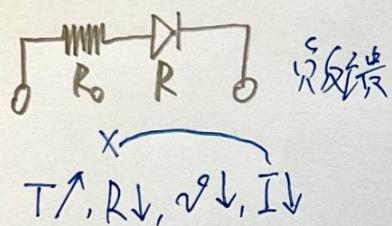
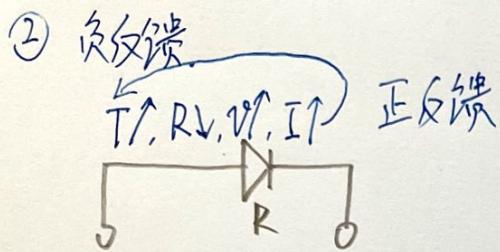
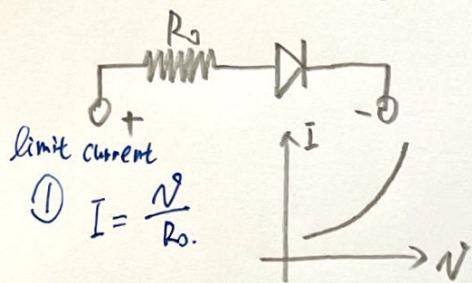
Quiz 3-1 Resistor in the circuit

1) The resistor in serial with a LED is to

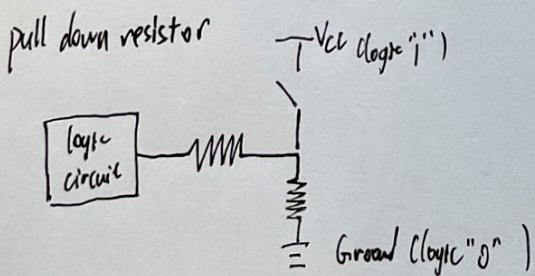
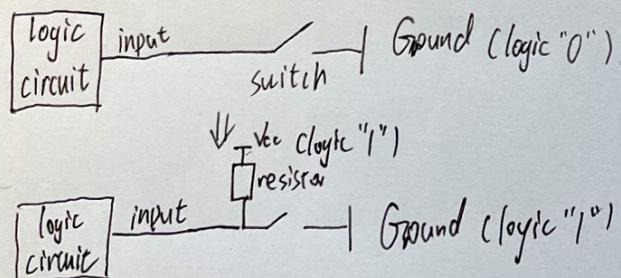
- a. pull up voltage
- d. increase brightness

- b. pull down voltage
- e. reduce brightness

(c) limit current



• pull up resistor



扫描全能王 创建

Quiz 3-2 Debounce process

2) A button connected to a microprocessor is read without any debounce process. What the following may happen normally?

- A. The button work normally.
- B. A single push of the button produces multiple inputs.
- C. The button can't be read.

a) A and B

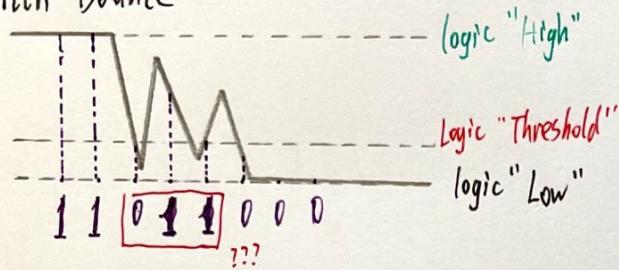
b. A and C

c. B and C

d. all

e. none

Switch Bounce

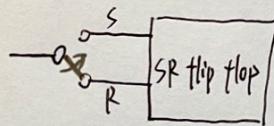


Debounce } hardware
 } software

(1) Hardware debounce

- ① RS flip-flop
- ② D flip-flop

RS flip-flop



Mode	S	R	Q_{out}
No change	0	0	Q
Set	1	0	1
Reset	0	1	0
Forbidden	1	1	-

起始

S	R	Q
0	1	0
0	0	0
0	1	0
0	0	0

抖动

S	R	Q
0	0	0
1	0	1
0	0	0

过渡

S	R	Q
0	0	0
1	0	1
0	0	1

抖动

S	R	Q
1	0	1
0	0	1

最终

D flip-flop

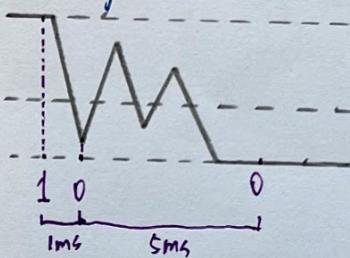
D	CLK	Q	Q_{out}
0	↑	0	1
1	↑	1	0
x	↑	0	0
x	↑	1	1

延时

只有第一次 CLK ↑,
才有效。

(2) software

delay 20 ms



```
void delayms (unit xms) {
    uint i,j;
    for (i=xms;i>0;i--) {
        for (j=10;j>0;j--);
    }
}
```

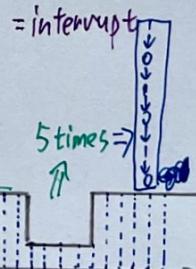
(3) 多触点

if (pressed = 0){

 a++;

 delay(5);

}



(3) 多触点

```
#define task4 S2
void control(void) -> Task4
{
    static_bit pressed = 0;
    if (!stop && pressed)
    {
        run = !run;
        pressed = 1;
    }
}
```

```
if (stop && pressed)
{
    pressed = 0;
}
```



扫描全能王 创建

Quiz 4-1 the advantages and disadvantages of capacitor

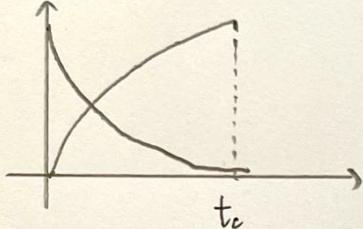
- 1) Using a large capacitance **capacitor** in a sample and hold circuit has **advantage** comparing with which using small capacitance is
- higher precision **更高的精度**
 - shorter acquisition time **数据采集时间**
 - shorter aperture time **快门时间**
 - lower droop rate **下垂率**
 - high resolution **分辨率**

$$V_C = V_{OC} + [V_{(0)} - V_{OC}] e^{-\frac{t}{RC}}$$

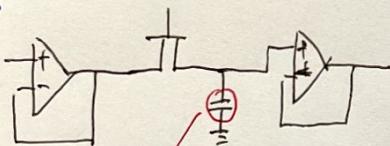
note from circuit analysis.

微小 V_{OC} .

$$t_C = RC$$



① precision



sample hold device (S/H)

当对模拟信号进行A/D转化时需要一定的转换时间，在这个转换时间内，需要保持信号不变。

① precision: ($\pm 5\%$, $\pm 10\%$, $\pm 20\%$) \Rightarrow 样本

② acquisition time: **采集时间** **标准** **电容**.

$$= \text{Sample Time} + \text{Calibration Time} + \text{Charge Distribution}$$

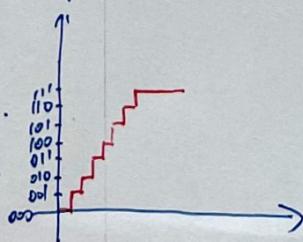
③ aperture time \Rightarrow **快门时间**
is defined as the time difference between the analog propagation delay of the front-end

buffer and the switch driver digital delay

plus half of the aperture time

④ resolution 可以加到的最小增量电压.

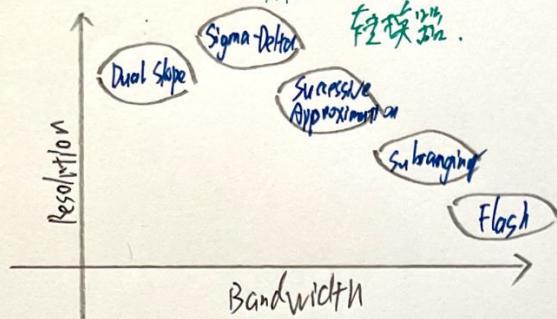
主要位数, 精样波, 精率, 精样带宽.



扫描全能王 创建

Quiz 4-2 The resolution of the digital converter

- 2) Generally, which of the following analog to digital converter types has the **highest resolution**?
- a. Flash 并行型
 - b. Sigma-Delta $\Sigma - \Delta$
 - c. Dual Slope 双斜率型
 - d. Subranging 分段逐次逼近型
 - e. Successive Approximation 逐次比较法



见 P4-2-1



扫描全能王 创建

Quiz 5-1 Real-time OS

1) A **real-time OS** means a system that

- a. has a real-time clock ~~实时钟~~
- c. can response within a known time
- e. none of above

b. can response very fast

d. all of above

单片机 - 慢

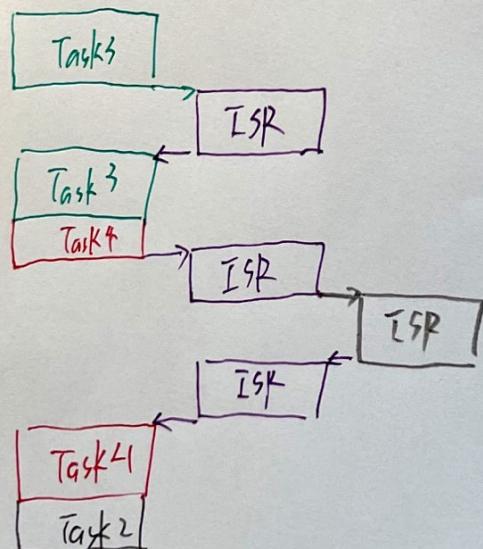
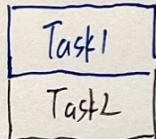
OS a set of programs that manage computer hardware resources and provide common services for application software

Real-time OS $\begin{cases} \text{logical correctness} \\ \text{time} \end{cases}$

实时操作系统保证在一定时间内完成特定功能的执行。

$\begin{cases} \text{硬 (hard)} & \text{fail} \\ \text{软 (soft)} & \text{continue} \end{cases}$

在实时操作系统中，我们可以把要实现的功能分解为许多任务，通过调度实现其中的一部分，而任务本身十分简单，从而提高效率，通常是一个死循环。



扫描全能王 创建

Quiz 5-2 Common Sense

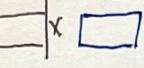
- 2) A cooperative (non-preemptive) multi-tasking is more suitable for an embedded system that
- a. contains many tasks.
 - b. contains only a few tasks.
 - c. developed by many development teams from different companies.
 - d. developed by one individual development team.
 - e. its stabilization is very important.

• Scheduler

a simple operating system that allow tasks to be called periodically or on a one-shot basis

a single timer interrupt service routine that is shared between many different tasks.

{ Co-operative scheduler 

{ Pre-emptive scheduler 

{ Hybrid scheduler 



扫描全能王 创建