

BALOTARIO- ARQUITECTURA DE COMPUTADORAS

El 8051 contiene las siguientes características:

- 1 CPU de 8 bits como parte central.
- 32 líneas bidireccionales de entrada y salida (4 puertos)
- 128 bytes de memoria RAM
- 2 Controladores / Timers de 16 bits
- 1 UART completo
- 5 estructuras de interrupción con dos niveles de prioridad
- 1 circuito de reloj
- 64 Kbytes de espacio para programa.
- 64 Kbytes de espacio para datos.

1) Mencione las diferencias entre el original 8051 fabricado por Intel y el AT89S52 fabricado por Atmel.

el ultimo tiene un contador de mas y una memoria flash de 8k bytes mientras que el 8052 tiene solo 4k bytes.

2) ¿Quiénes son los tres investigadores del microprocesador?. Porque se tardó tanto en reconocer a uno de ellos y quien fue?

Ted Hoff, S.Mazor y F.Fagin desarrollaron el 4004 de Intel. F.faggin fue reconocido tardamente.

3) ¿Cual fue la primera aplicación del microprocesador?. Describa la ley de moore.

gente de la empresa Busicom, una joven empresa japonesa, fue a la compañía Intel (fundada el año anterior) para que hicieran un conjunto de doce chips para el corazón de su nueva calculadora de mesa de bajo costo.

"La complejidad de los circuitos integrados se duplicaría cada año con una reducción de costo conmensurable".

cada año se duplicaría el número de transistores de un circuito integrado

4) Mencionar 5 registros de funciones especiales del 8051.

Sol.

Son aquellos registros que permiten controlar al 8051 en su totalidad. Algunos registros:
ACC - ACUMULADOR, es usado como registro procesador, en torno a él se realizan la mayoría de operaciones.

IE- Habilitador de Interrupciones

TMOD- modo de control del timer/contador

TCON- control del Timer/contador

TH0-Byte alto del Timer/contador

SCON- control del puerto serie

5) Explique la diferencia entre un microprocesador y un microcontrolador.

Sol.

Un microprocesador es usado para fines generales y de alta complejidad de cálculo, en cambio un microcontrolador es una microcomputadora usada para fines específicos. Hablando más en específico, un microcontrolador se comporta como un automata.

Los Microprocesadores tradicionales se basan en la arquitectura de Von Newmann, mientras que los microcontroladores trabajan con arquitectura de Harvard.

6) Mencione la diferencia entre el microcontrolador ATMEL AT89C52 y el ATMEL AT89S52.

Sol.

7) Describa el pin RD y WR del 8051.

RD funciona como control de lectura y WR como control de escritura cuando una RAM externa es conectada.

cuando $[RD \text{ (barra)}] = 0$, luego realiza la operación de lectura desde la RAM externa.

8) ¿Que puertos del microcontrolador 8051 se utilizan como bus de datos y bus de direcciones cuando se conecta una RAM externa?

SOL. Puerto 0 actúa como bus de datos y puerto 2, el puerto 0 actúa también como una dirección de orden inferior, y el puerto 2 como dirección de orden superior

9) Describa el modo de direccionamiento indirecto para el 8051.

SOL1. En este direccionamiento se utiliza un registro en el cual se encuentra la dirección del operando, Se puede decir que este registro trabaja como un puntero. Todas las operaciones se realizarán con el operando al que apunta el registro. Los únicos registros que pueden realizar esta función son R0, R1 y DPTR

SOL2. La Dirección de un operando (registro o algún lugar en la memoria) es colocada en algún registro (R0, R1, DPTR) para luego poder realizar operaciones de forma indirecta. Ejm:

```

org 8000h

mov R0,#40h

mov A,#09h

mov @R0,A

end

```

10) Realice un programa en assembler de un contador decimal de 0 a 99 que muestre la cuenta en las direcciones de memoria 45h y 46h de la memoria de datos. Los valores deben mostrarse en esas direcciones de memoria. Ejemplo: 45h: 00 ,46h:01... 45h: 09, 46h:01

```

org 8000h
mov R1,#0
mov R2,#0

repite:
lcall retraso
mov 45h,R2
mov 46h,R1
inc R1
cjne R1,#10,repite
lcall retraso
mov R1,#0
inc R2
cjne R2,#10,repite
ljmp 2F0H

retraso:
mov R0,#255
lazo:
djnz R0,lazo
ret
end

```

11) Realice un programa en assembler de un contador decimal de 0 a 199 que muestre la cuenta en las direcciones de memoria 50h,51h y 52h de la memoria de datos.

sol.

```

org 8000h
mov R1,#0
mov R2,#0
mov R3,#0
mov R4,#0
repite:
mov 50h,R3
mov 51h,R2
mov 52h,R1

```

```

inc R4
cjne R4,#200,sigue
ljmp 2f0h
sigue:
lcall retraso
inc R1
cjne R1,#10,repite
mov R1,#0
lcall retraso
inc R2
cjne R2,#10,repite
mov R2,#0
lcall retraso
inc R3
cjne R3,#10,repite

```

```

retraso:
mov R0,#255
lazo:
djnz R0,lazo
ret
end

```

12) Describa como funciona el pin EA del 8051.

es el Pin de Habilitación de Acceso Externo.

Si tenemos que usar varias memorias, la aplicación de la lógica 1 a este pin le indica al Microcontrolador que lea los datos de ambas memorias: primero interna y luego externa

13) Describa los modos del timer/contador.

Modo 0. En este modo cualquiera de los 2 timers, 0 y 1, se configuran como registros de **13 bits**, que consisten en los 8 bits del registro de TH (TH1 o TH0) y los 5 bits menos significativos del registro TL (TL1 o TL0). Los 3 bits más significativos de TL no son utilizados en este modo.

Modo 1. Este modo es utilizado por cualquiera de los dos timers, y los configura como Timer/Contador de **16 bits**.

Modo 2. Este modo también puede llegar a ser utilizado por los dos timers, y los configura para un conteo de **8 bits** (Tlx) con recarga automática. Al ser sobrepasada la capacidad de TL, éste es recargado automáticamente, con el contenido de TH y a su vez es activada (TF=1) la bandera de sobreflujo.

Modo 3. El Timer 1, en el modo 3 mantiene su cuenta, es decir, tiene el mismo efecto que cuando se establece la bandera TR1=0.

El Timer 0, en éste modo, establece TL0 y TH0 como de contadores separados.

TL0 utiliza los bits de control (C/T, GATE, TR0, INT0) del Timer 0. TH0 es bloqueado como temporizador "Timer", el cual emplea las señales de control del Timer 1, TR1 y TF1.

14) Explica cuales son las interrupciones del timer y como se habilitan.

Tres interrupciones desde los tres timers, cuando se genera un desbordamiento.

15) Indique para un reloj de sistema de 11.0592 MHz y utilizando el timer 0 en modo 0 como se calcularia un tiempo de un segundo. Este mismo tiempo, como se calcularia para un timer 0 en modo 1 y el mismo reloj del sistema.

$$(\#cuentas) \times (12/11.0592M) = 1 \rightarrow \#cuentas = 921600$$

modo 0, capacidad de conteo = 8192 \Rightarrow se necesitan 112.5 conteos completos

modo 1, capacidad de conteo = 65536 \Rightarrow se necesitan 14 conteos completos

16) Indique para un reloj de sistema de 11.0592 Mhz y utilizando el timer 0 en modo 0 como se calcularia una frecuencia de 20hz.

para generar una frecuencia de 20 hz necesitamos que la onda sea de 1/20 seg de longitud, es decir necesitamos 1/40 seg para la duracion de la señal en alto y 1/40 seg de duracion de la señal en bajo.

Ahora bien, sabemos que $T(\text{timer}) = 12T(\text{reloj})$ con lo que:

$T(\text{timer}) = (12/11.0592M)$ seg por cada cuenta del timer, pero se necesita 1/40 seg, entonces, el numero de cuentas que se necesitara, se dara por la siguiente ecuacion:

$$(\#cuentas) \times T(\text{timer}) = 1/40 \rightarrow \#cuentas = 23.040$$

Ahora como el timer esta en modo 0, la capacidad de este es: 8192

se necesitara 2.8125 conteos completos del timer0.

17) Indique para un reloj de sistema de 11.0592 Mhz y utilizando el timer 0 en modo 1 indique el calculo necesario para obtener una frecuencia de 17 hz. Asi mismo calcule la frecuencia de 23 hz usando un timer 0 en modo 0 y el mismo reloj del sistema.

18) Realice un programa que resuelva la siguiente expresion: $(100+20)/2-30*2+99$ y almacene el resultado en decimal en las direcciones 40h y 41h.

Luego de obtener el resultado:

```
mov A, #99
```

```
mov B, #10
```

```

div AB      ; Divido el resto entre 10
mov 40h,A
mov 41h,B

```

19) Desarrolle un programa que calcule el onceavo y treceavo termino de la serie de fibonacci. Colocar ambos resultados en las direcciones 30h y 31h.

20) Desarrolle un programa de Conversion de un numero decimal de 8 bits a cualquier base. Aplicar el siguiente algoritmo:

```

INICIO
Enteros Num, base, DirMem
DirMem ← 37h
Mientras (Num es distinto a 0)
{
    DirMem ← Resto(Num/base)      ;Almaceno el resto de la división en la
                                ;dirección de memoria(DirMem)
    Num ← Num/base                ; El nuevo "Num" es el cociente de "Num" entre la "base"
    DirMem ← DirMem-1             ;Decremento DirMem
}
FIN

```

Sol.

```

org 8000h
num equ 50h
base equ 51h
mov num,#255
mov base,#8
mov R0,#37h
mov A,num
repite:
mov B,base
DIV AB
mov @R0,B
mov num,A
dec R0
cjne A,#0,repite
ljmp 2f0h
end

```

21) Desarrolle un programa que llene la memoria RAM interna desde la direccion 40h con los numeros: 34, 55, 3, 11, 4, 5, 6, 22, 7, 8,9,33,10,11,12,13,44,18,19,20,21,22,55,24,25,26,27,28,29,30,66.
SUG: usar obligatoriamente un lazo repetitivo,asi como una tabla de numeros para acceder a él de manera indexada.

sol.

```

org 8000h
mov R2,#31h
mov R0,#40h
mov A,#00h

```

```

repite:
inc A
mov R3,A
lcall tabla
mov @R0,A
inc R0
mov A,R3
djnz R2,repite
ljmp 2F0h

tabla:
movc A,@A+PC
ret
db 34, 55, 3, 11, 4, 5, 6, 22, 7,
8,9,33,10,11,12,13,44,18,19,20,21,22,55,24,25,26,27,28,29,30,66
end

```

22) Utilizando el algoritmo en pseudocódigo siguiente:

```

INICIO
para (i=1,i<N,i++){
    j=i
    v=A[i]
    Mientras(j>0){
        if(A[j-1]>v){
            tmp=A[j-1]
            A[j-1]=A[j]
            A[j]=tmp
        }
        j=j-1
    }
}
FIN

```

Desarrolle un programa que ordene el listado de números 22,5,10,90,42,70,33

SUG: Utilize para comparar la instrucción subb

23) Utilizando el siguiente algoritmo. Determine los primos del listado: 17,18,19,20,21,22,23 y almacenarlos a partir de la dirección 55h.

```

1  N: número, X: divisor
2  INICIO
3  Enteros X, N
4  X <-- 2
5  Mientras(X es diferente a N)
6  {
7  ¿Es entero(N/X)?
8  {
9  Si es entero entonces
10 No guarda porque no es primo
11 Termina
12 }
13 {
14 No es entero entonces
15 X <-- X +1
16 }
17 }
18 Almacena el primo en su correspondiente dirección.

```

24) Describa al detalle al menos 4 interrupciones del 8051. Explique.

externas 0 y 1(pines p3.2 y p3.3) y timers 0 y 1.

25) Como se habilitan las interrupciones externas 0 y 1 y la interrupcion interna del timer 1?

Primero habilitamos EL CONTROL Maestro de interrupciones con EA=0, luego

setb EX0; habilita int ext 0

setb EX1; habilita int ext 1

setb ET1;

26) Como se puede fijar para que la interrupcion externa 1 sean en el flanco y no en el estado bajo?

IT1=0 la interrupción se activará al detectar un nivel bajo en el pin correspondiente; IT1=1 la interrupción se activará al detectar un flanco de bajada.

27) ¿En que consiste la prioridad de una interrupcion y como se hace una interrupcion de prioridad alta ,por ejemplo la interrupcion externa 0?

REGISTRO IP:

setb PX0 ; Externa 0 en alta

setb PX1

setb PT0

setb PT1

29) Mencione como se habilita y configura la interrupcion externa 1 y la del timer 0.

30) Indique cuales son las direcciones de memoria de cada fila del display LCD.

0-27 y 40-67

31) Explique la diferencia entre RISC y CISC. Asi, como entre arquitectura HARVARD y VON NEUMANN.

CISC (Complex Instruction Set Computer) Computadoras con un conjunto de instrucciones complejo.

RISC (Reduced Instruction Set Computer) Computadoras con un conjunto de instrucciones reducido.

La arquitectura CISC se refiere a los micorprocesadores tradicionales que operan con grupos grandes de instrucciones de procesador (lenguaje de maquina). Los microprocesadores INTEL 80xxx estan dentro de esta categoria (incluido el PENTIUM). Los procesadores CISC tienen un set de instrucciones complejas por naturaleza que requieren varios a muchos ciclos para completarse.

La arquitectura RISC a diferencia de los CISC tiene un set de instrucciones simples requiriendo uno o pocos ciclos de ejecucion. Estas instrucciones pueden ser utilizadas mas eficientemente que la de los procesadores CISC con el diseño de software apropiado, resultando en operaciones mas rapidas.

NEUMANN:

en la cual la unidad central de proceso (CPU), está conectada a una memoria principal única (casi siempre sólo RAM) donde se guardan las instrucciones del programa y los datos. A dicha memoria se accede a través de un sistema de buses único (control, direcciones y datos).

HARVARD:

Una de las memorias contiene solamente las instrucciones del programa (Memoria de Programa), y la otra

sólo almacena datos (Memoria de Datos).

32) Describa una breve historia de mips.

33) Describa las características del MSP430G2553 de TI y al menos 4 modos de Direccionamiento, explique con ejemplos.

CARACTERISTICAS:

-CPU ortogonal

-timers y registros de 16 bits

-arquitectura von neuman

-16 kbytes de flash room

-convertidor ADC

-SENSOR DE TEMPERATURA

MODO REGISTRO . OPERACIONES EN REGISTROS DIRECTAMENTE `mov R6,R7`

MODO INDEXADO: `x` ES CONSTANTE , luego la posicion de `x+Rn` es direccionada

ejm: `mov F000h(R8),R9` lo que esta en `f000h+R8` se mueve a `R9`

MODO INDIRECTO CON REGISTRO: ejem: `mov.w @R4,R5` igual a 8051

MODO INMEDIATO: `mov.w #E2h,R5`

34) Describa las características internas de MIPS y sus registros.

35) Indique al menos 4 aplicaciones en las que se usa MIPS.

36) Porque la empresa SGI compro a la empresa fabricante de procesadores MIPS?

37) Por que cree UD que MIPS no ha podido superar a la arquitectura ARM?

38) En que consiste la tecnologia de segmentación. DETALLE

39) ¿Que características tiene MIPS que lo hace un procesador sencillo frente a los procesadores INTEL?

ADICIONAL:

USO DE TIMER 0/1:

`org 0000h`

`mov TMOD,#02h ;configura el timer 0 en modo 2`

`; como un temporizador de 8 bits con auto recarga`

`mov TH0,#131 ;valor de recarga`

`setb TR0 ;inicia el timer 0`

`espera_desborde:`

`jnb TF0,$;esto se repite mientras TF0 es cero y no hay`

`;desbordamiento`

`cpl P1.4 ;complementa el pin 5 en el Puerto 1`

`clr TF0 ;este flag de desbordamiento se pone a 1 por hardware,`

`;este flag debe ser puesto a 0 por software`

`sjmp espera_desborde`

`end`

USO DE TIMER 0/1:

```
getbyt equ 11Eh
print equ 136h
org 8000h
;Detiene el temporizador del timer2(clr TR2)
;clr TR2
clr 0CAh ; para el temporizador
; Configure Timer 2 as Clock Out
;CLR T2CON.1
clr 0C9h; usar como temporizador
;----- T2OE DCEN
;T2OE Timer 2 Output Enable bit
;DCEN When set, this bit allows Timer 2 to be configured as an up/down counter
mov 0C9h,#3
; Load Clock values [Crystal : 11.0592MHz]
lcall print
db "Selecione Frecuencia",0dh,0ah
db "00) 42Hz",0dh,0ah
db "01) 100Hz",0dh,0ah
db "02) 500Hz",0dh,0ah
db "03) 1000Hz",0dh,0ah,0;
lcall getbyt
mov 60h,A
lcall selecciona_valor_menos_s; RECAP2L
MOV 61H,A
MOV 0CAh,A;0,0,66,33h
mov A,60h
lcall selecciona_valor_mas_s;RECAP2H
mov 62h,A
MOV 0CBh, A;0,94h,EAh,F5h
;Start Timer 2, señal sale por P1.0
;SETB TR2
setb 0CAh
ljmp 2F0h
selecciona_valor_menos_s:
inc A
movc A,@A+PC
ret
db 0,0,66h,33h
selecciona_valor_mas_s:
inc A
movc A,@A+PC
```

```
ret
db 0,94h,0EAh,0F5h
end
```

USO DE INTERRUPCIONES:

en el timer:

```
Mov TMOD, #0h ;Modo 0 del timer 0
mov a, #1 ; Timer 0 es la fuente de interrupción
mov dptr, #IntTimer0 ; IntTimer0 es la dirección inicial del ISR
lcall setintvec ;Posibilita tener el ISR en RAM
Setb TRO ; inicia el timer 0
Setb ET0 ; habilita interrupción del timer 0
Setb EA ; habilita interrupción global
```

en el boton:

```
clr IT0 ; fija la interrupción externa sensible en el nivel bajo
mov a, #0 ; fuente de interrupción externa 0
mov dptr, #ISR0 ; fijar dptr con la dirección de inicio; de la interrupción externa 0 para
llamar a
; setintvec
lcall setintvec ; fija el ISR para INTO
setb EX0 ; habilita interrupción externa 0 en el registro IE
setb EA ; flag de interrupción maestra del registro IE
setb p3.2 ; fija el pin p3.2 como entrada botón conectado
```