

Ej ①

Un programa recibe por stack la dirección de inicio de un arreglo de 20 elementos, recorre el arreglo y toda vez que ^{encuentra un} elemento no-nulo copia ese valor en un periférico que está mapeado en la dirección 02C10048h.

La escritura del periférico debe ser implementada mediante una subrutina que recibe por ~~stack~~ el valor a escribir y no devuelve argumentos al programa principal y rutina ~~debe~~ deben ser declarados en el mismo módulo.

```
.begin
.org 2048

macro push reg
    add r14, -4, r14
    st reg, r14
```

```
.endmacro
```

```
macro pop reg
    ld r14, reg
    add r14, 4, r14
```

```
.endmacro
```

```
A .eqv 02C10h
```

```
B .eqv 048h
```

```
sethi A, r1
```

```
sll r1, 2, r1
```

```
add r1, B, r1
```

! en r1 está la dir del
periférico

pop %r2 ! en %r2 almaceno la dir del arreglo de 20 elc
add %r0, %r0, %r3 ! guardo el largo del array (4.20 = 80)
call recorrer_array

recorrer_array: addc %r3, -4, %r3

bneg Fin

add %r3, %r2, %r4 ! en r4 almaceno la
posicion del array

ld %r4, %r5 ! obtengo el contenido del
array y lo guardo
en %r5

andc %r0, %r5, %r0

be recorrer_array

push %r5

add %r5, 0, %r2 ! guardo la direccion de mail

call escribir_periferico

ba recorrer_array

escribir_periferico: pop %r4

st %r4, %r1 ! escribo el contenido
no nulo al periferico

Fin: jmp %r2, %r0

.end

$$2^{10} = 1024 \quad 2^9 = 512 \quad 2^8 = 256 \quad 2^7 = 128 \quad 2^6 = 64 \quad 2^5 = 32 \quad 2^4 = 16$$

$$2^3 = 8 \quad 2^2 = 4 \quad 2^1 = 2 \quad 2^0 = 1$$

Ej 2

Un procesador arc ejecuta la siguiente instrucción

Orcc %r10, %r0, %r7

para lo cual utiliza el siguiente microcódigo

1609: R[rd] ← orcc (R[rs1], R[rs2])

GOTO 2047

Se pide indicar

- todos los bits del mir
- todos los bits para el Mux de dir de la memoria de control
- Entradas y salidas para la lógica de control de saltos
- entradas y salidas de de co A
- entrada y salida de la ALU

IR

Decode

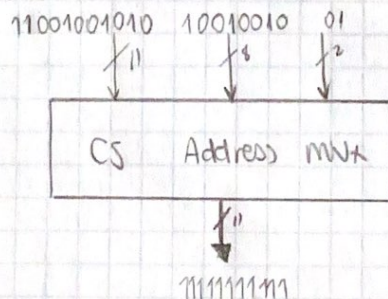
1,10 0,10010,00

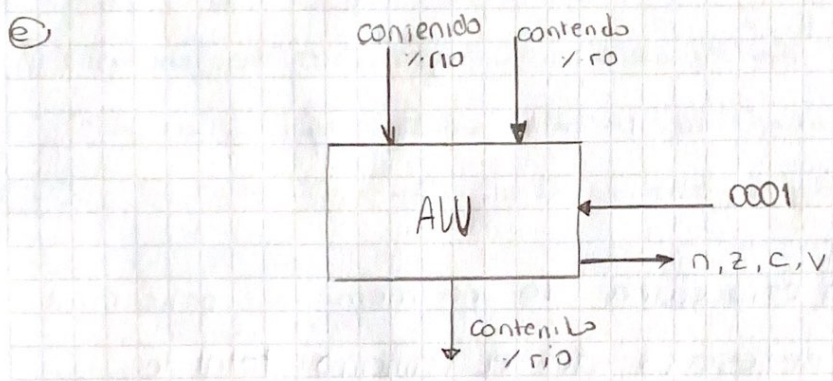
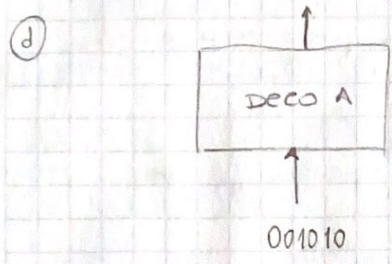
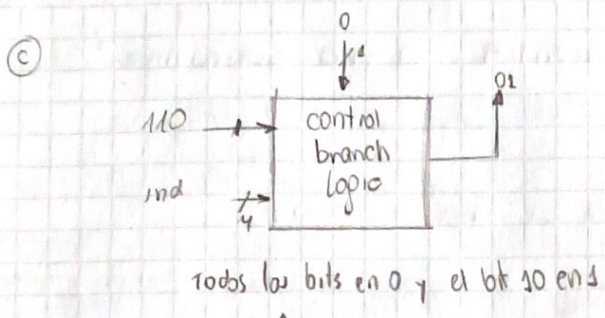
10 00111 010010 01010 0 0000000 00000
op rd op3 rs1 bitB rs2

a) MIR

000000 1 000000 1 000000 1 0 0 000 110 1111111111
A Am B Bm C Cm RD WR AW cond Jmp

b)





Ej ③

① ¿Que significa pasar argumentos a una subrutina?
¿Como? ¿Para qué?

Repetir analisis en el caso de una macro.

El pasaje de argumentos a subrutinas se realiza para poder utilizar informacion cargada anteriormente al programa en la misma.

El pasaje se puede hacer mediante registros, por stack o por area reservada de memoria.

En el caso de la macro los argumentos son definidos al lado del nombre de la misma. Se utilizan para por y evitar la repetición de código.

② Al diseñar la estructura de una memoria caché entre otras cosas es necesario def el tamaño total de memoria caché y también el tamaño de los bloques en que se divide

Analizar ventajas y desventajas de un tamaño relativamente pequeño de los bloques de cache.

Ventaja

Al ser más pequeño el bloque la velocidad aumenta, ya que tiene un tamaño reducido de caché

Desventaja

Si achicamos el tamaño de los bloques el principio espacial se ve reducido pues estamos teniendo menos espacio para almacenar por ejemplo datos juntos en un array dado que el bloque de caché es pequeño.