

Práctica 4



Sección de control

Objetivo

El alumno se familiarizará con la sección de control de un sistema computacional.

Equipo

Computadora personal con el software Logisim.

Desarrollo

1. Instrucciones de salto del procesador de 4 bits

Las **instrucciones de salto** del procesador de 4 bits tienen el siguiente formato:



Las Instrucciones de Salto causan un salto a la dirección **ADDR** de la memoria de instrucciones, esto si **C_i** es igual a 1. El bit **C_i** está definido por **CC**, a menos que CC=0, lo cual indica otra clase de instrucciones (instrucciones de E/S). Los bits CC nos proporcionan tres clases de saltos: incondicional, condicional sobre **C** o **Z** y condicional sobre **C̄** o **Z̄** como puede verse en la Tabla 1.

Tabla 1. Tipos de salto basados en C_i

CC	C _i	Tipo de salto
00	0	-----
01	1	Salto incondicional.
10	C	Salto condicional sobre C o Z .
11	C̄	Salto condicional sobre C̄ o Z̄ .

La Tabla 2 muestra el formato de las cinco instrucciones de salto disponibles en el procesador:

Tabla 2. Instrucciones de salto

INSTRUCCIÓN	MNEMONICO	OPERACIÓN	CÓDIGO			
			OP	CC	F	ADDR (5 bits)
JUMP	JMP addr	PC = addr	11	01	X	addr
JUMP to addr si acarreo	JC addr	if C=1, PC=addr else PC+1	11	10	0	addr
JUMP to addr si no acarreo	JNC addr	if C=0, PC=addr else PC+1	11	11	0	addr
JUMP to addr si es cero	JZ addr	if Z=1, PC=addr else PC+1	11	10	1	addr
JUMP to addr si no es cero	JNZ addr	if Z=0, PC=addr else PC+1	11	11	1	addr

Los pasos para ejecutar una instrucción de salto son:

1. Cargar PC si $C_i=1$ (para JMP/JC y JNC)
En el caso de JZ cargar PC si $Z=1$ y para JNZ cargar PC si $Z=0$
2. Incrementar PC.

- a) Simule en Logisim las instrucciones de salto para el procesador didáctico descritas anteriormente.
 - i) Es necesario decodificar la instrucción para determinar que sea una instrucción de salto.
 - ii) Si **CC=01**, copiar los bits **ADDR** de la instrucción en **PC**. Esta es una instrucción de salto incondicional que no requiere que se decodifiquen más bits.
 - iii) Si **F = 0**, decodificar **CC** para determinar el valor C_i , si $C_i=1$, copiar los bits **ADDR** de la instrucción en **PC**.
 - iv) Si **F = 1**, decodificar **CC** para determinar si el salto es en base a **Z** o \bar{Z} , copiar los bits **ADDR** de la instrucción en **PC** en caso de que la condición se cumpla.

2. Unidad de control del procesador de 4 bits

- a) Simule en Logisim el procesador didáctico de 4 bits.
 - i) Integre la **sección de E/S** realizada en la práctica 2, la **ALU** de la práctica 3 y el módulo de **instrucciones de salto** en un archivo de simulación de Logisim.
 - ii) Agregue una memoria de 32x10bits la cual va a ser la memoria de programa del sistema.
 - iii) Agregue el reloj del sistema.
 - iv) Diseñe y simule la unidad de control (una máquina de estados), la cual se encarga de enviar las señales de control para activar los componentes según el modo de operación e instrucción a ejecutar.
Los modos de operación son **PROG** y **RUN**. Cuando se está en modo **PROG**, el programador ingresa los **10 bits** de la instrucción por medio de un dip-switch conectado a la memoria, la unidad de control está a la espera de una señal **Enter** para enviar la señal de escritura a la memoria para que la almacene. Esto se repite para cada una de las instrucciones del programa (máximo **32**). Cuando se está en modo **RUN**, la unidad de control lee una a una las instrucciones de la memoria, las decodifica y ejecuta. La ejecución implica el envío de señales tales como lectura/escritura de un registro (**R0-R3** y/o **RS** y

RD), enviar la función a la **ALU**, escribir el nuevo valor de **C** y **Z**, incrementar o reemplazar el valor de **PC**, etc.

La Fig. 1 presenta las señales de control del procesador de 4 bits, en su diseño puede agregar u omitir señales si lo considera necesario.

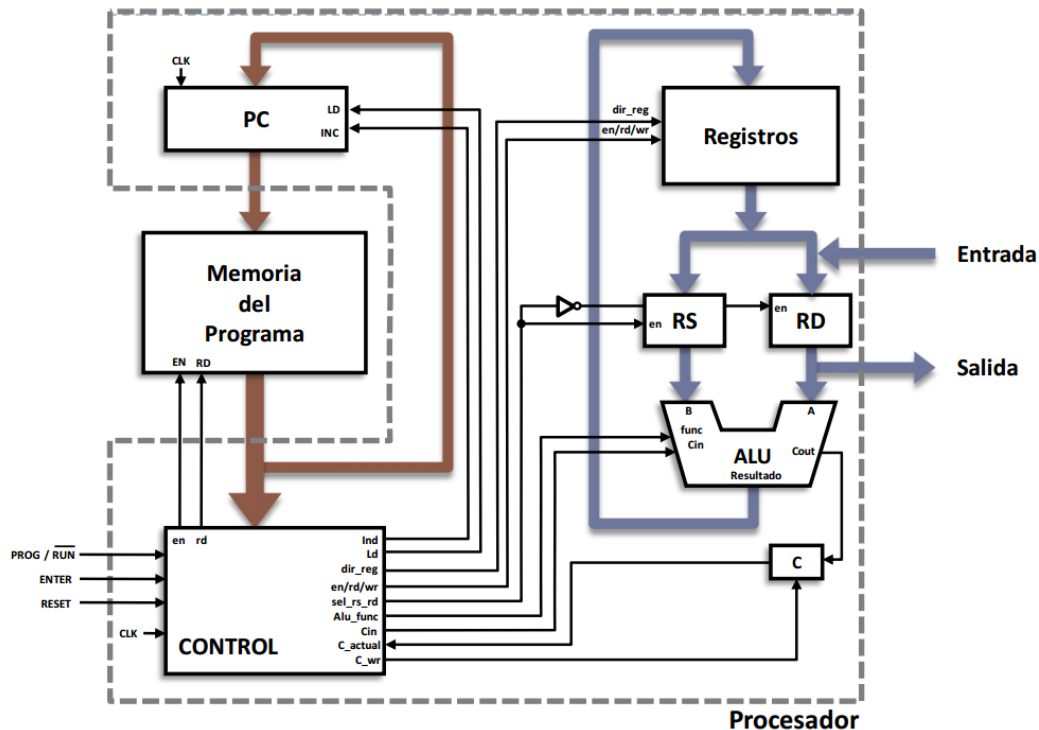


Figura 1. Procesador didáctico de 4 bits. Nota: Falta la bandera **Z** de la ALU y sus señales.

b) Muestre y describa paso a paso en el reporte el procedimiento de:

- i) Almacenamiento de una instrucción en la memoria de programa.
- ii) Ejecución de una instrucción de lectura de puerto de entrada.
- iii) Ejecución de una instrucción de escritura de puerto de salida.
- iv) Ejecución de una instrucción aritmética en la ALU.
- v) Ejecución de una instrucción lógica en la ALU.
- vi) Ejecución de una instrucción de salto incondicional.
- vii) Ejecución de una instrucción de salto condicional sobre **C**.
- viii) Ejecución de una instrucción de salto condicional sobre \bar{Z} .

Incluya impresiones de pantalla de su diseño en Logisim para todos los puntos anteriores.

Conclusiones y comentarios

Dificultades en el desarrollo

Referencias