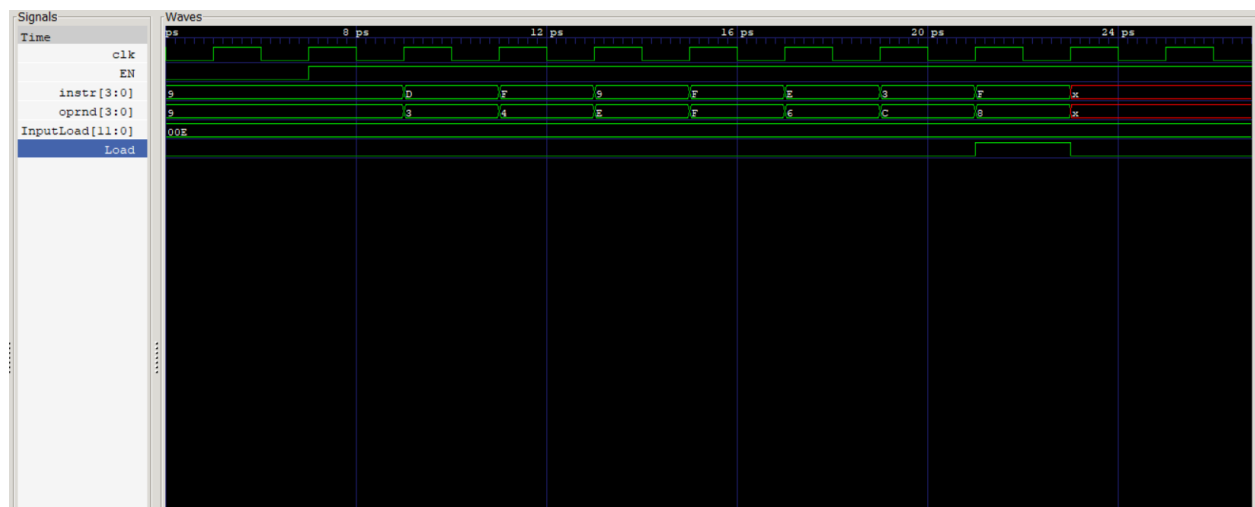


Ejercicio 1

CLK	RST	Enable	Load	InputLoad	Instruction	Operand
1	0	1	0	000000001110	1010	1010
0	0	1	0	000000001110	1010	1010
1	0	0	0	000000001110	1001	1001
0	0	0	0	000000001110	1001	1001
1	0	0	0	000000001110	1001	1001
0	0	0	0	000000001110	1001	1001
1	0	1	0	000000001110	1001	1001
0	0	1	0	000000001110	1001	1001
1	0	1	0	000000001110	1101	0011
0	0	1	0	000000001110	1101	0011
1	0	1	0	000000001110	1111	0100
0	0	1	0	000000001110	1111	0100
1	0	1	0	000000001110	1001	1110
0	0	1	0	000000001110	1001	1110
1	0	1	0	000000001110	1111	1111
0	0	1	0	000000001110	1111	1111
1	0	1	0	000000001110	1110	0110
0	0	1	0	000000001110	1110	0110
1	0	1	0	000000001110	0011	1100
0	0	1	0	000000001110	0011	1100
1	0	1	1	000000001110	1111	1000
0	0	1	1	000000001110	1111	1000
1	0	1	0	000000001110	xxxx	xxxx



Como podemos observar el contador permite la extracción de los datos de la memoria, la cual luego se separa por el Fetch las instrucciones junto con los operandos. El contador es capaz de cargar un valor como se puede observar en el gráfico en los 22ps.

Diego A. Méndez

19673

Lab 10

Ejericio 2

B	Enable	F	Y	Carry	Zero
0000	x , x	xxx	xxxx	0	1
0010	1 , 1	010	0010	0	0
0001	1 , 1	011	0011	0	0
0011	1 , 1	001	0000	0	1
0000	1 , 1	000	0011	0	0
0000	1 , 1	011	0011	0	0
1000	1 , 1	010	1000	0	0
1110	1 , 1	100	0111	1	0
0000	1 , 1	000	1000	0	0
1111	1 , 1	001	1001	1	0
1111	1 , 1	001	1010	1	0

gtkwave Parte2_tb.vcd Parte2_tb.gtkw



En este gráfico se pueden observar los cambios en la Alu y como guarda los valores en el acumulador. Es importante dejar el cambio que se guarde después de un flanco de reloj. Dependiendo del valor de F la función del ALU cambia entre sus 5 posibles opciones. Asimismo se puede observar cuando hay un overflow en el Carry y cuando la salida es zero se enciende la bandera de zero.