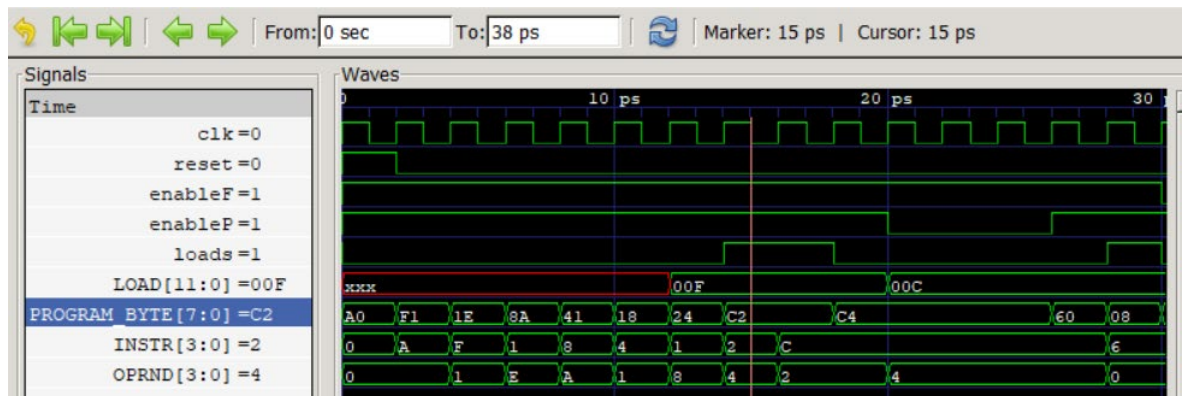


Laboratorio #10

EJERCICIO #1

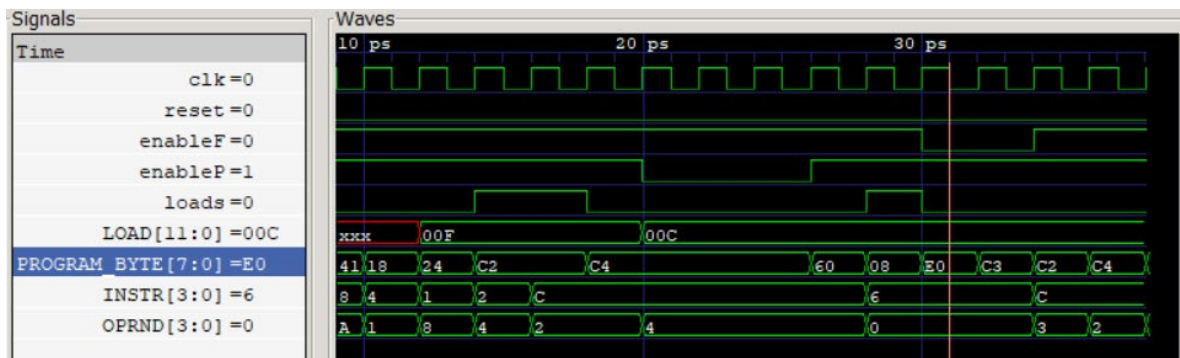
CLK	RESETA	ENABLEB1	ENABLEB2	F	OPRND	OUT	DATA	SALIDA	ACCU	CARRY	ZERO
0	1	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
1	0	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
0	0	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
1	0	1	1	000	1010	0000	1010	0000	0000	0	1
0	0	1	1	000	1010	0000	1010	0000	0000	0	1
1	0	1	1	010	1010	1010	1010	1010	0000	0	0
0	0	1	1	010	1010	1010	1010	1010	0000	0	0
1	0	1	1	001	1010	0000	1010	0000	1010	0	1
0	0	1	1	001	1010	0000	1010	0000	1010	0	1
1	0	1	1	011	1010	1010	1010	1010	0000	0	0
0	0	1	1	011	1010	1010	1010	1010	0000	0	0
1	0	1	0	100	1111	zzzz	1111	0000	1010	1	0
0	0	1	0	100	1111	zzzz	1111	0000	1010	1	0
1	0	1	0	010	1111	zzzz	1111	1111	0000	0	0
0	0	1	0	010	1111	zzzz	1111	1111	0000	0	0
1	0	1	1	010	1111	1111	1111	1111	1111	0	0
0	0	1	1	010	1111	1111	1111	1111	1111	0	0
1	0	1	1	000	1111	1111	1111	1111	1111	0	0
0	0	1	1	000	1111	1111	1111	1111	1111	0	0
1	0	0	1	000	0001	1111	zzzz	1111	1111	0	0
0	0	0	1	000	0001	1111	zzzz	1111	1111	0	0
1	0	0	1	010	0001	zzzz	zzzz	zzzz	1111	0	x
0	0	0	1	010	0001	zzzz	zzzz	zzzz	1111	0	x
1	0	1	1	010	0001	0001	0001	0001	zzzz	0	0
0	0	1	1	010	0001	0001	0001	0001	zzzz	0	0
1	0	1	1	010	0001	0001	0001	0001	0001	0	0



Cuando encendemos la bandera de Loads, el dato LOAD se carga al program Counter y da una dirección para escoger la PC que se dirige hacia la Program ROM. Su salida es la PROGRAM_BYTE, los cuales entran al flipflop D que es el Fetch, de los cuales los 4 bits más significativos se dirigen hacia instr, los 4 bits menos significativos se dirigen hacia oprnd.



Cuando apagamos el enableP deja de contar y se puede observar en el ProgramByte que no cambia su valor.

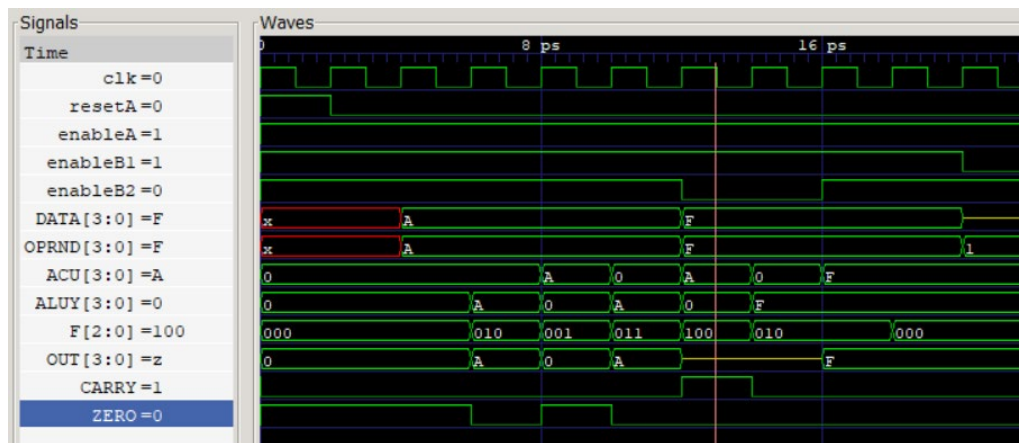


Cuando apagamos el enableF el Fetch deja de recibir valores y las salidas INSTR y OPRND.

```
//Memoria de Fong solo de Fong
1010_0000 //Address i.e 8h00
1111_0001 //Address i.e 8h01
0001_1110 //Address i.e 8h02
1000_1010 //Address i.e 8h03
0100_0001 //Address i.e 8h04
0001_1000 //Address i.e 8h05
0010_0100 //Address i.e 8h06
0000_0001 //Address i.e 8h07
1111_1111 //Address i.e 8h08
1010_1010 //Address i.e 8h09
1000_1111 //Address i.e 8h10
0010_0010 //Address i.e 8h11
0000_1000 //Address i.e 8h12
1110_0000 //Address i.e 8h13
1100_0011 //Address i.e 8h14
1100_0010 //Address i.e 8h15
1100_0100 //Address i.e 8h16
0110_0000
0101_1000
```

Ejercicio 2

CLK	RESETA	ENABLEB1	ENABLEB2	F	OPRND	OUT	DATA	SALIDA	ACCU	CARRY	ZERO
0	1	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
1	0	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
0	0	1	1	000	xxxx	0000	xxxx	0000	0000	0	1
1	0	1	1	000	1010	0000	1010	0000	0000	0	1
0	0	1	1	000	1010	0000	1010	0000	0000	0	1
1	0	1	1	010	1010	1010	1010	1010	0000	0	0
0	0	1	1	010	1010	1010	1010	1010	0000	0	0
1	0	1	1	001	1010	0000	1010	0000	1010	0	1
0	0	1	1	001	1010	0000	1010	0000	1010	0	1
1	0	1	1	011	1010	1010	1010	1010	0000	0	0
0	0	1	1	011	1010	1010	1010	1010	0000	0	0
1	0	1	0	100	1111	zzzz	1111	0000	1010	1	0
0	0	1	0	100	1111	zzzz	1111	0000	1010	1	0
1	0	1	0	010	1111	zzzz	1111	1111	0000	0	0
0	0	1	0	010	1111	zzzz	1111	1111	0000	0	0
1	0	1	1	010	1111	1111	1111	1111	1111	0	0
0	0	1	1	010	1111	1111	1111	1111	1111	0	0
1	0	1	1	000	1111	1111	1111	1111	1111	0	0
0	0	1	1	000	1111	1111	1111	1111	1111	0	0
1	0	0	1	000	0001	1111	zzzz	1111	1111	0	0
0	0	0	1	000	0001	1111	zzzz	1111	1111	0	0
1	0	0	1	010	0001	zzzz	zzzz	zzzz	1111	0	x
0	0	0	1	010	0001	zzzz	zzzz	zzzz	1111	0	x
1	0	1	1	010	0001	0001	0001	0001	zzzz	0	0
0	0	1	1	010	0001	0001	0001	0001	zzzz	0	0
1	0	1	1	010	0001	0001	0001	0001	0001	0	0



- Cuando F = 000 la ALU deja pasar el dato del acumulador. Cuando todos los valores de salida de la ALU son 0, la bandera de 0 se activa.
- Cuando F = 001 realiza la resta entre el valor del ACCU y DATA.
- Cuando F = 010 la ALU deja pasar DATA.
- Cuando F = 011 la ALU realiza la suma. Si tiene un overflow la bandera de Carry se activa.
- Cuando F = 100 la ALU realiza la función NOR entre los valores de ACCU y DATA.
- Cuando enableB1 = 0 apagamos el buffer que nos da el valor de DATA, el cual se dirige hacia B de la ALU
- Cuando enableB2 = 0 apagamos el buffer que nos da los valores de OUT