

**EXAMEN ORDINARIO DE JUNIO DE 2005. I. T. INF. DE GESTIÓN**

En la figura 1 se ilustra un circuito compuesto por una memoria, una unidad aritmético-lógica de 4 bits y los elementos de transferencia necesarios.

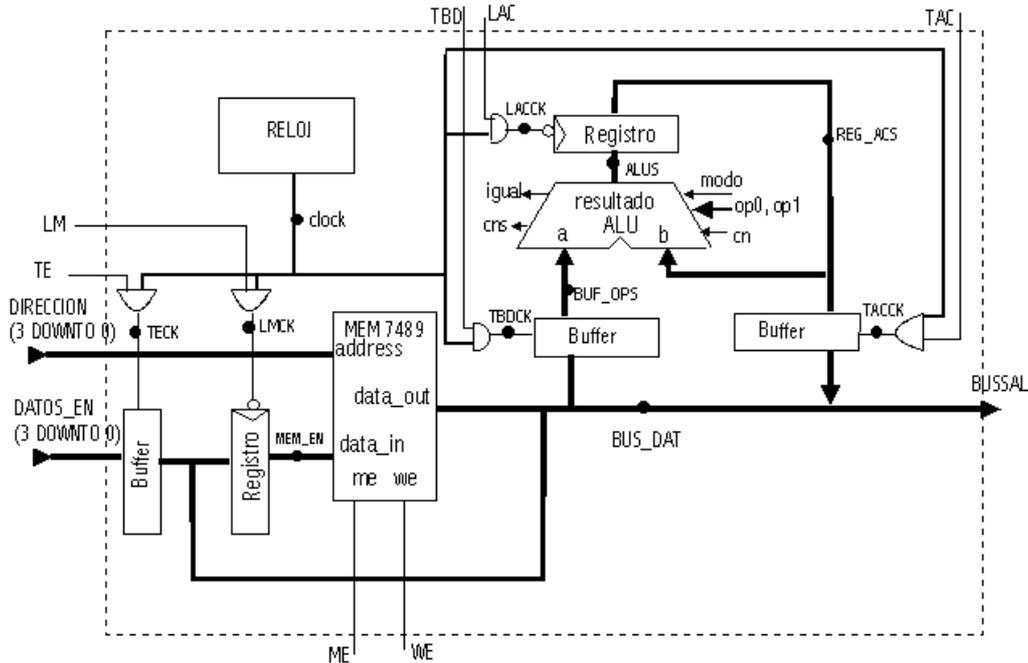


Figura 1.

Los modos de operación de la ALU y de la memoria se resumen en las tablas siguientes:

ME	WE	Operación	Salida
L	L	escritura	Z
L	H	lectura	dato
H	L	No selec.	Z
H	H	No selec.	Z

Tabla 1. Operación Memoria.

op1-op0	modo=0	Modo=1
00	a	a OR b
01	b	a AND b
10	a+b	a XOR b
11	a-b	NOT a

Tabla 2. Operación ALU.

De todos los elementos presentados en las figuras se adjunta su modelo VHDL .

1.- Desarrollar el modelo VHDL estructural del circuito de la figura 1. **(4 Puntos)**

2.- La figura 2 representa una simulación del circuito anterior para un periodo de reloj de 150 ns (Los nombres de señales se corresponden con los indicados en la figura 1). Se pide:

2.1.- Explicar las transferencias/operaciones que se están simulando.**(3 Puntos)**

2.2.- Calcular la frecuencia máxima de funcionamiento que permita, en un solo periodo de reloj, realizar una operación en la ALU entre un operando disponible en el bus de entrada (DATOS\_EN) y otro disponible en la salida del acumulador (REG\_ACS). El resultado válido debe estar presente a la salida de la ALU (ALUS) en la transición de bajada de reloj que finaliza dicho periodo. **(3 Puntos)**

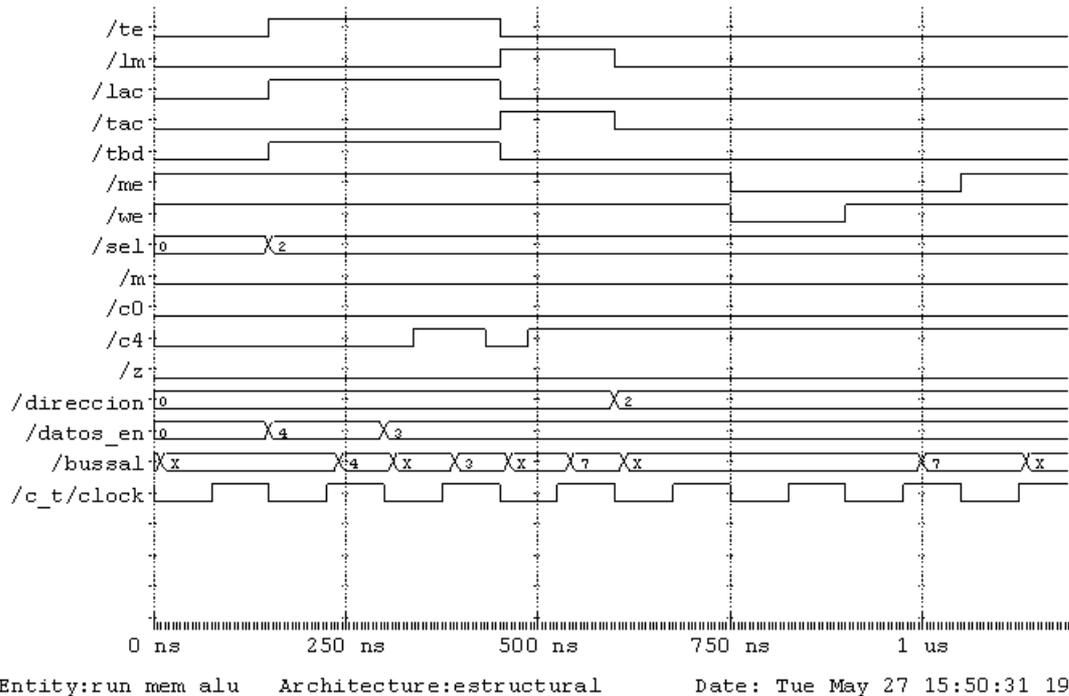


Figura 2.

### MODELOS VHDL DE LOS COMPONENTES:

```
ENTITY and2 IS
    GENERIC (retardo: TIME := 2 ns);
    PORT (e1, e2: IN BIT; sal: OUT BIT);
END and2;

ARCHITECTURE comportamiento OF and2 IS
    ...
END comportamiento;
```

```
USE WORK.arq_pack.ALL;
ENTITY registro4 IS
    GENERIC(retardo_carga:TIME:=10 ns);
    PORT(entrada:IN vector_bus; control:IN BIT:= '0';
        salida:OUT vector_bus:=('0','0','0','0'));
END registro4;

ARCHITECTURE comportamiento OF registro4 IS
    ...
END comportamiento;
```

```
USE WORK.arq_pack.ALL;
ENTITY buffer4 IS
    GENERIC(retardo_transferencia:TIME:=15 ns;retardo_Z:TIME:=10 ns);
    PORT(entrada:IN vector_bus; control:IN BIT:= '0';
        salida:OUT vector_bus);
END buffer4;

ARCHITECTURE comportamiento OF buffer4 IS
    ...
END comportamiento;
```

```
USE WORK.arq_pack.ALL;
ENTITY alu_4bits IS
  GENERIC (retardo:TIME:=25 ns);
  PORT (a, b:IN vector_bus;
        op:IN BIT_VECTOR(1 DOWNT0 0);
        modo,cn:IN BIT;
        resultado:OUT vector_bus ;
        cns,igual:OUT BIT);
END alu_4bits;

ARCHITECTURE comportamiento OF alu_4bits IS
...
END comportamiento;
```

```
ENTITY reloj IS
  GENERIC(periodo:TIME:=?? ns);
  PORT(reloj:OUT BIT:= '0');
END reloj;

ARCHITECTURE comportamiento OF reloj IS
...
END comportamiento;
```

```
USE WORK.arq_pack.ALL;
ENTITY ram_7489 IS
  --retardo_salida: retardo de propagación de las celdas de memoria a la salida
  --retardo_Z: retardo de propagación del estado Z a la salida
  GENERIC(retardo_salida:TIME:=100 ns; retardo_Z:TIME:=85 ns);
  PORT(DATA_IN:IN vector_bus:=('0','0','0','0');
        ADDRESS:IN BIT_VECTOR(3 DOWNT0 0):=X"0";
        ME,WE:IN BIT:= '0';
        DATA_OUT:OUT vector_bus:=('Z','Z','Z','Z'));
END ram_7489;

ARCHITECTURE comportamiento OF ram_7489 IS
.....
END comportamiento;
```

```
PACKAGE arq_pack IS
  TYPE tri_estado IS ('0', '1', 'Z');
  TYPE vector_bus IS ARRAY (3 DOWNT0 0) OF tri_estado;
  TYPE array_vector_bus IS ARRAY (INTEGER RANGE <>) OF vector_bus;
  FUNCTION resolucion (entrada: array_vector_bus) RETURN vector_bus;
  --la señal del bus será la resuelta
  SUBTYPE bus_resuelto IS resolucion vector_bus;
  PROCEDURE logico_entero(VARIABLE vector:IN BIT_VECTOR;
                          VARIABLE entero:OUT INTEGER);
  PROCEDURE entero_logico(VARIABLE entero:IN INTEGER;
                           VARIABLE vector:OUT BIT_VECTOR);
END arq_pack;
PACKAGE BODY arq_pack IS
  .....
END arq_pack;
```