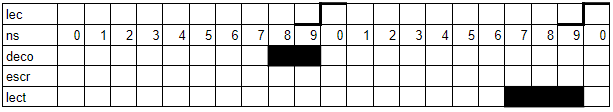
# 

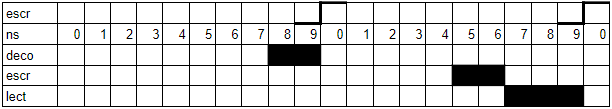
# Multiprocesadores : Práctica 2

Héctor Romero de Blas, Miquel García de Mendoza, Ramon Mateo

**Trabajo 2: Muestre en sendos diagramas temporales de retardos un acceso de lectura y un acceso de escritura. Los retardos deben indicarse en el instante de tiempo más tardío en que pueden producirse para un funcionamiento correcto. Indique los retardos que pueden afectar a un valor, tanto en una posición de almacenamiento como en un puerto de salida. Utilice una señal de reloj cuadrada con periodo de 10 ns como referente. Utilice los valores de retardo especificados en el Apéndice 2.16 (ret\_deco\_dat, ret\_dat\_leer, ret\_dat\_esc)., en la página 5.**

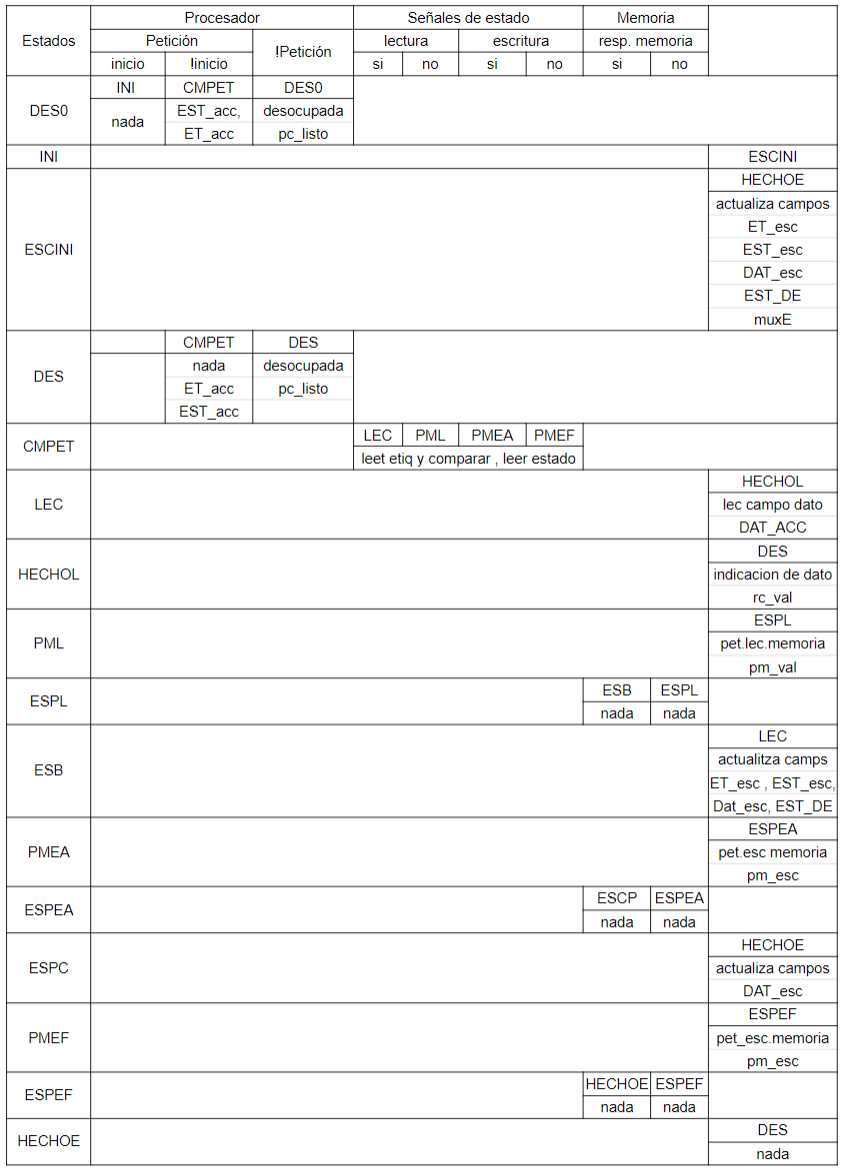
****

*Tabla 1: diagrama temporal del retardo del acceso de lectura.*

**

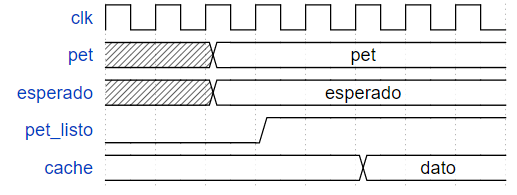
*Tabla 2: diagrama temporal del retardo del acceso de escritura.*

**Trabajo 9: Para el controlador de cache de la Figura 16, construya una tabla de transiciones entre estados (Apéndice 2.3), donde también se especifique la lógica de salida en la segunda subfila de cada estado (activación, desactivación de las señales). En el caso de la lógica de salida indique sólo la activación de la señal (valor 1). Suponga que por defecto las señales no se activan., en la página 15.**

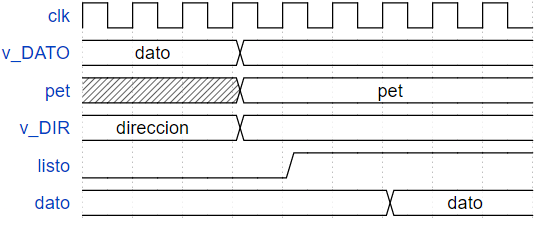
****

*Tabla 3: transacciones entre estados con información de las señales de salida.*

**Trabajo 13: Analice el programa de prueba. Describa los procesos “productor” y “consumidor” mediante diagramas temporales. Así mismo, describa los procedimientos “Plectura” y “Pescritura”. Para ello, utilice la señal reloj y sus flancos como referente. Céntrese en el protocolo de las interfaces: comunicación entre el procesador y la cache y viceversa., en la página 16.**

****

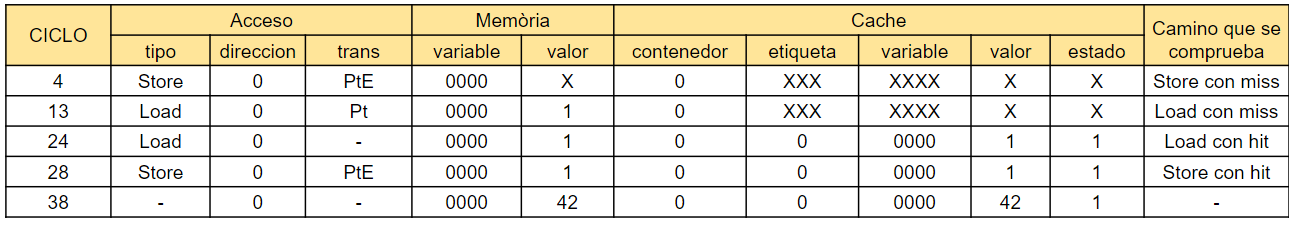
*Figura 1: diagrama temporal del proceso “productor”.*

**

*Figura 2: diagrama temporal del proceso “consumidor”.*

Pescritura empieza generando la petición de escritura en función de las entradas dato y dirección suministrada. Después activa la señal de listo y, tras la latencia de procesamiento, la cache dispone del dato.

**Trabajo 23: Construya una secuencia con el mínimo número de accesos, que muestre una comprobación incremental del diseño. Para este trabajo utilice una única entrada de cache. Muestre en una tabla la información actualizada en cada acceso del programa de prueba., en la página 17.**

****

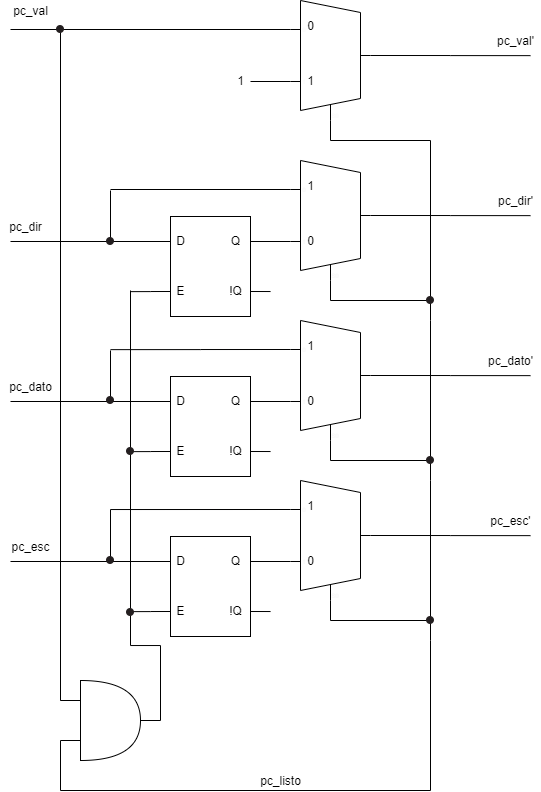
*Tabla 4: accesos a memoria del programa de prueba.*

**Trabajo 24: Una vez esté completamente implementado el controlador de cache, prepare una secuencia de accesos a memoria que compruebe el funcionamiento de forma exhaustiva (comprobar entradas distintas, entrelazar sin accesos consecutivos, accesos a la misma y distinta dirección, conflictos en cache)., en la página 17.**

****

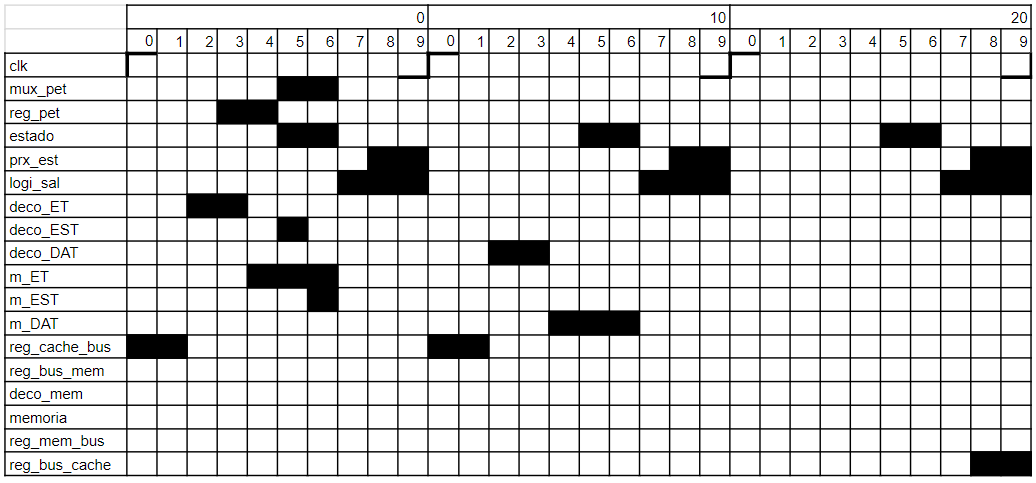
*Tabla 5: secuencia de accesos para comprobar el funcionamento del controlador de cache.*

**Trabajo 25: En la interface procesador/cache mostrada en la Figura 18, utilizada en el diseño RTL (Apéndice 2.4, Figura 45), todas las señales son entrada del multiplexor y del registro. Además, las señales sólo se almacenan en el registro si se cumple la función lógica “pc\_val and pc\_listo”. Esto es, si no hay petición y “pc\_listo = ‘1’” no se actualiza el registro. Desde el punto de vista de la lógica se utilizan tantos multiplexores y registros como señales (pc\_dir, pc\_dato, pc\_esc y pc\_val, Figura 17). Rediseñe la interface, dibujando un esquema de circuito, de forma que se utilice el menor número de multiplexores y registros. Considere cada señal pc\_dir, pc\_dato, pc\_esc y pc\_val como un todo. Esto, si se utiliza un multiplexor o un registro contabilice una unidad., en la página 17.**

****

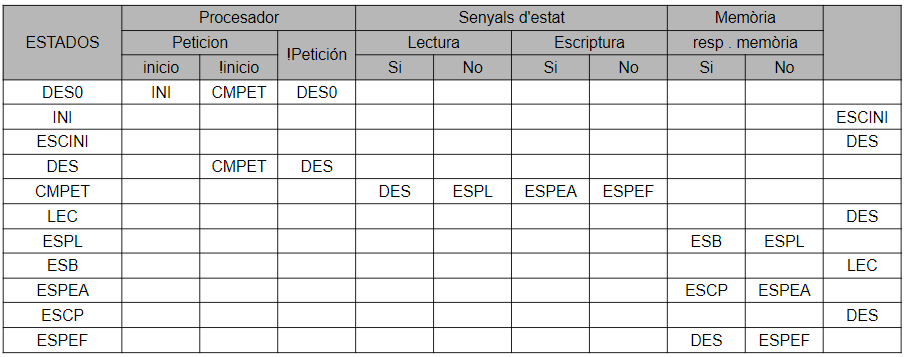
*Figura 3: rediseño del circuito utilizando el mínimo de registros y multiplexores.*

**Trabajo 26: Represente en un diagrama temporal de retardos los retardos de los componentes en un acierto de lectura. Suponemos que el último ciclo del productor se solapa con el estado DES en el controlador de cache. Los retardos correspondientes al productor deben indicarse en el instante más tardío en el cual es factible, teniendo en cuenta el periodo del reloj. El retardo de la comparación de etiquetas y la puerta “and” de este resultado con la lectura del campo estado tiene un retardo de 0 ns. Recuerde que las señales X\_acc y X\_esc están incluidas en la lógica de salida del autómata., en la página 18.**

****

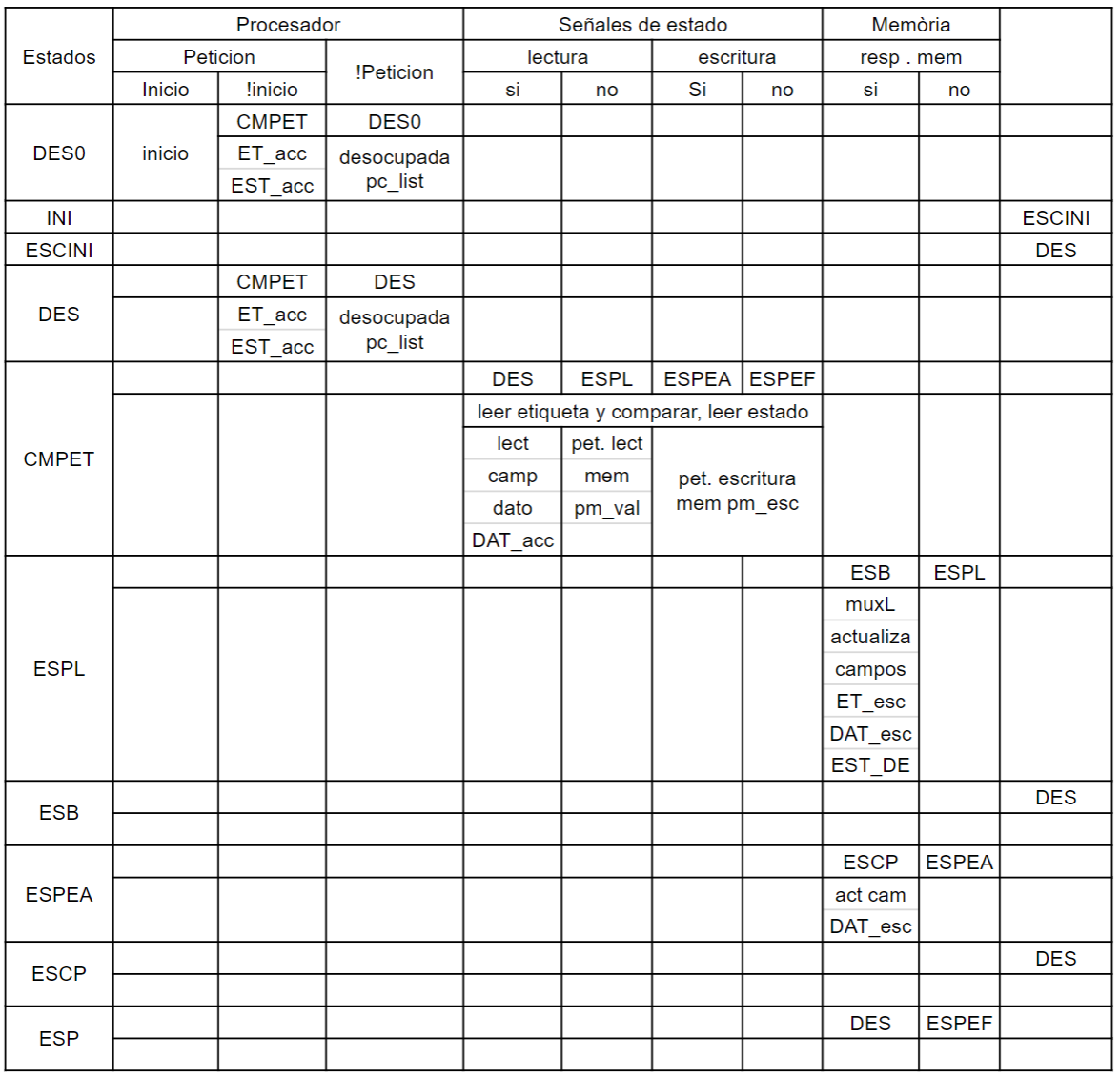
*Tabla 6: diagrama temporal de los retardos de los componentes en acierto de lectura.*

**Trabajo 35: Finalmente construya una tabla de transiciones entre estados (Apéndice 2.7, Figura 68) donde, además, se elimine el estado LEC en una transacción load que acierta en cache. Modifique la descripción VHDL del controlador de cache de forma oportuna. Elabore el controlador de cache con Quartus., en la página 20.**

****

*Tabla 7: transiciones entre estados sin estado LEC en acierto de lectura.*

**Trabajo 40: Construya la tabla de transiciones entre estados del controlador de cache, donde también se especifique la lógica de salida en la segunda subfila de cada estado (activación, desactivación de las señales) (Apéndice 2.10, Figura 72). Denomine mxL a la señal que controla el multiplexor muxL (Figura 26)., en la página 22.**

****

*Tabla 8: transacciones entre estados con información de las señales de salida.*

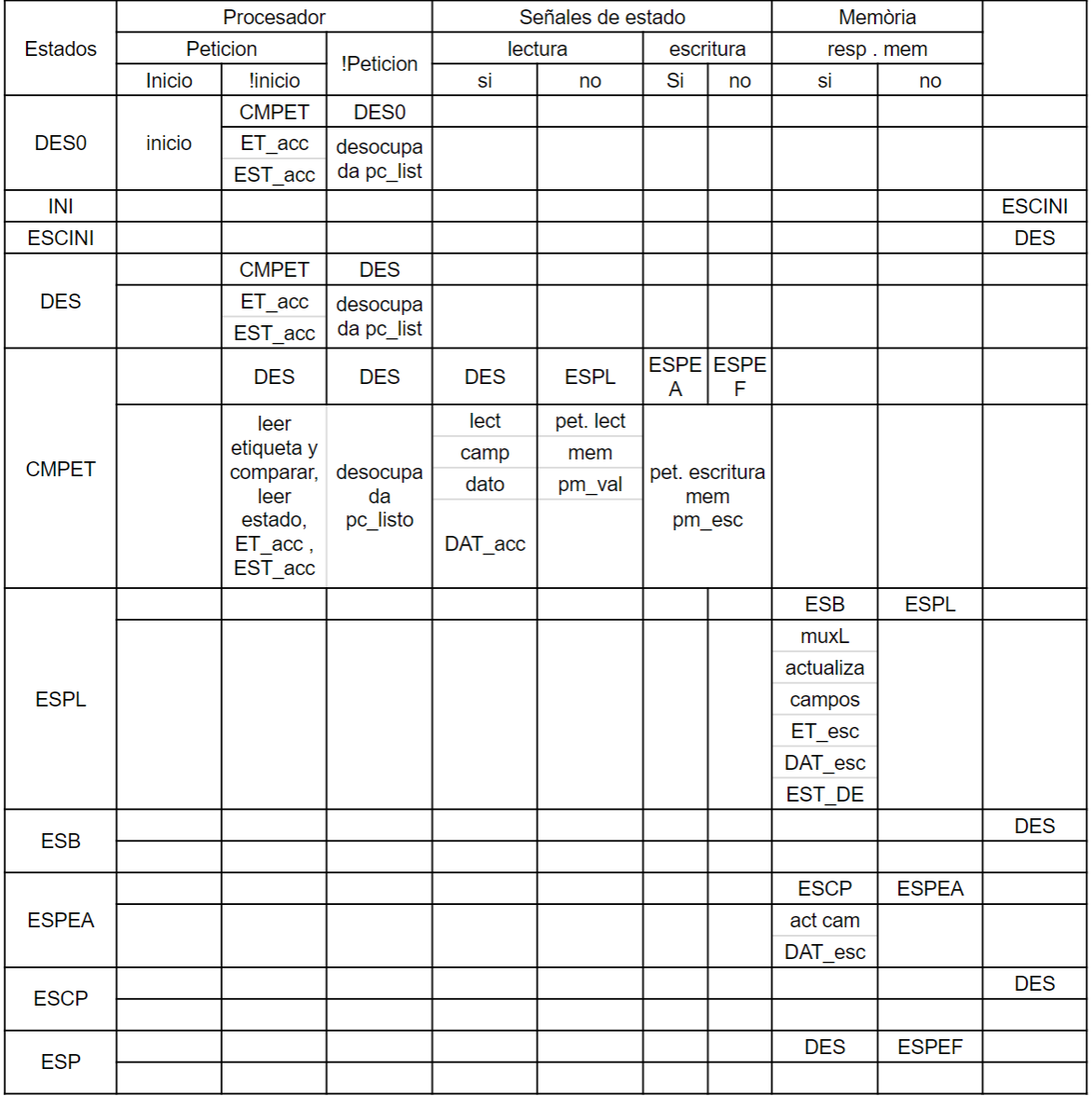
**Trabajo 48: Analice la influencia del multiplexor muxL en el tiempo de ciclo., en la página 23.**

El muxL aumenta ligeramente el tiempo de ciclo dado que depende de las salidas del controlador de cache y de la propia salida de datos (ya sea de la propia cache o memoria). De todos modos, este incremento en el tiempo de ciclo permite enviar el dato al procesador en el mismo ciclo que llega desde la memoria principal.

**Trabajo 53: Para este proyecto, analice la necesidad de disponer de los estados ESB y ESCP., en la página 25.**

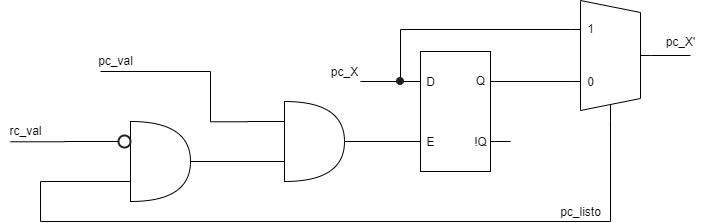
Dado que la lectura con acierto ha sido reducida a un único ciclo se requieren los estados ESB y ESCP para actualizar la memoria cache una vez memoria ha respondido. Para eso se necesita un estado transicional diferente al de esperar a la memoria que exista entre dicho estado y el estado DES..

**Trabajo 56: Construya la tabla de transiciones entre estados del controlador de cache, donde también se especifique la lógica de salida en la segunda subfila de cada estado (activación, desactivación de las señales) (Apéndice 2.13, Figura 76)., en la página 26.**

****

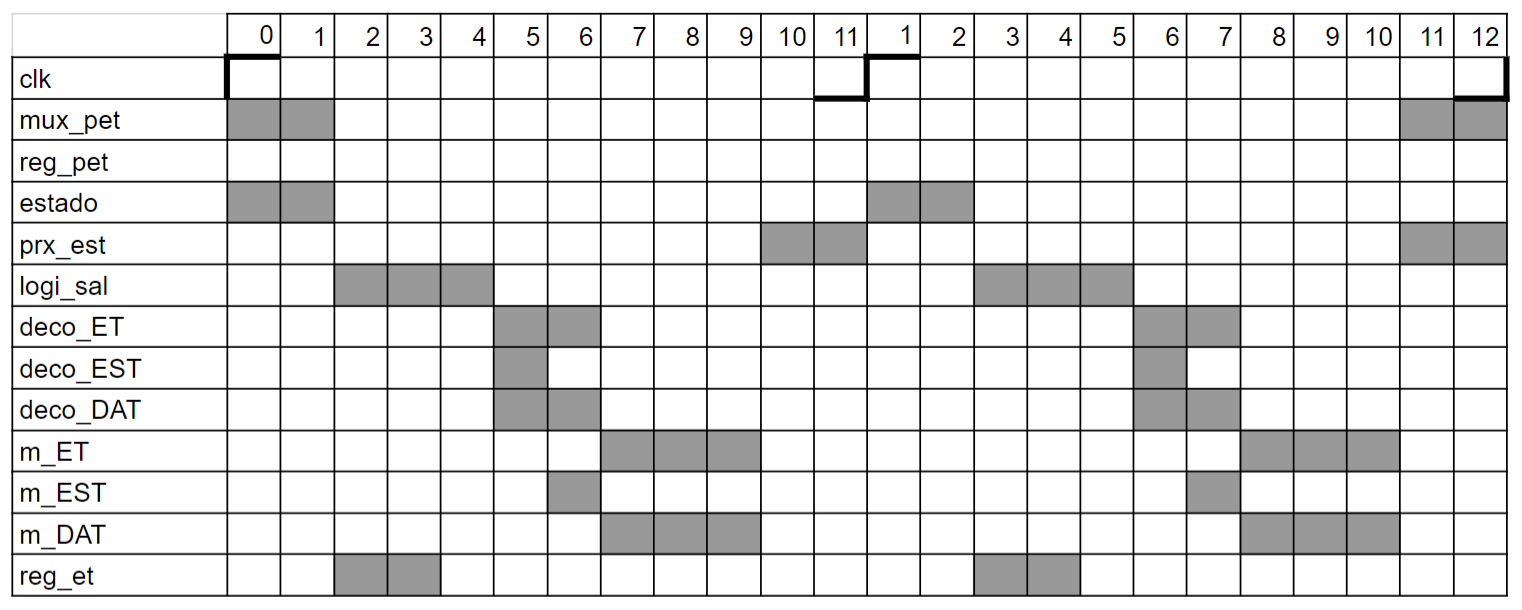
*Tabla 9: transacciones entre estados con información de las señales de salida.*

**Trabajo 63: En la interface procesador/cache que se utiliza, todos los registros se actualizan cuando pc\_listo = ‘1’. Rediseñe la interface, dibujando un esquema de circuito, de forma que las señales de entrada, que sea posible, se almacenen en el registro sólo si hay una petición pendiente (esquema de circuito)., en la página 27.**

****

*Figura 4: rediseño del circuito con almacenaje en registro condicional.*

**Trabajo 64: Represente en un diagrama temporal de retardos los retardos de los componentes en dos acierto de lectura consecutivos a direcciones distintas. El productor tarda 1 ciclo en producir accesos. Los retardos correspondientes al productor deben indicarse en el instante más tardío en el cual es factible, teniendo en cuenta el periodo del reloj. El retardo de la comparación de etiquetas y la puerta “and” de este resultado con la lectura del campo estado tiene un retardo de 0 ns., en la página 27.**

****

*Tabla 10: diagrama temporal de los retardos de los componentes en dos aciertos de lectura consecutivos.*