

Problema 1. Diseñar la máquina de estados algorítmica que controla el funcionamiento de un túnel de lavado de coches según las siguientes especificaciones:

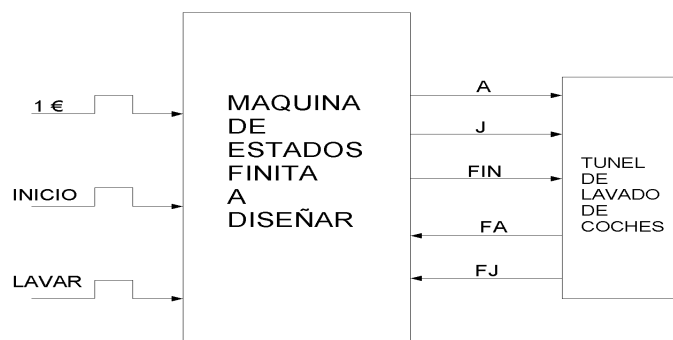
Entradas: INICIO, monedas de 1€, LAVAR, FA (fin agua), FJ (fin jabón)

Salidas: A (agua), J (jabón), FIN (saca el coche del túnel)

El usuario pulsa el botón de INICIO y después introduce 1 ó 2 monedas de 1€ en función del tipo de lavado que desee. Después pulsa el botón LAVAR. El sistema empieza a funcionar y el usuario simplemente espera a que el coche salga del túnel de lavado.

- Si el número de monedas introducidas es 0, la máquina no hace nada y vuelve al estado inicial.
- Si el número de monedas es de 1 €, la máquina activa la señal A ($A=1$) para que caiga agua sobre el coche y cuando se activa la señal de FA ($FA=1$), activa la señal de FIN y vuelve al estado inicial.
- Si el usuario ha introducido 2€ (o 3€ por error), la máquina primero mojará el coche (activa señal A) y cuando se activa la señal de FA lo enjabona activando la señal J ($J=1$) y cuando se activa la señal FJ, lo aclara activando de nuevo la señal A. Al terminar el aclarado ($FA=1$) activa la señal de FIN y vuelve al estado inicial.

Esta máquina no devuelve cambio ni detecta cantidades mayores de 3€.

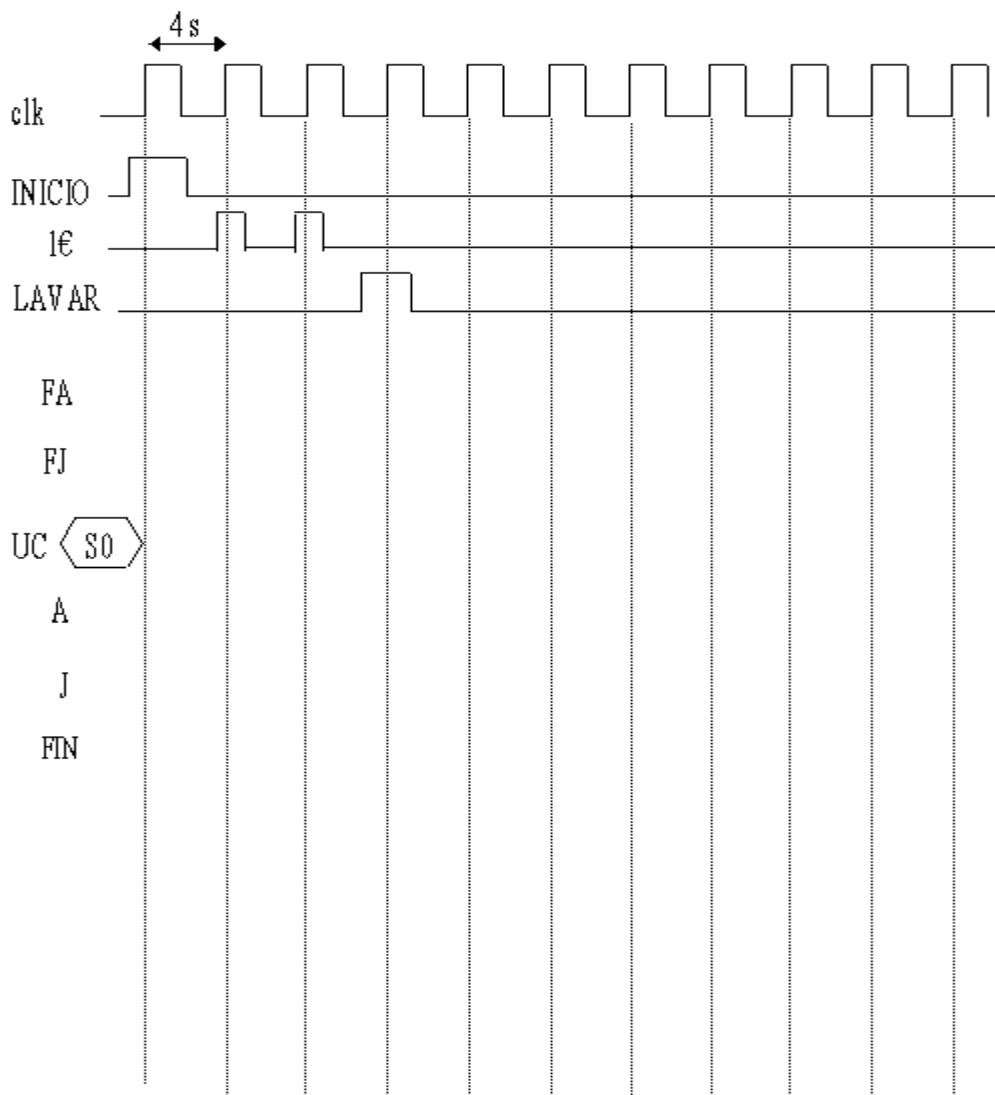


- Escribir el diagrama de flujo del sistema.
- Diseñar la ruta de datos utilizando los módulos combinacionales y secuenciales estudiados en la asignatura, indicando claramente sus entradas, salidas y la función que realizan (en caso de señales de control, función que realizan para cada posible valor o combinación de valores) y en el caso de los registros, y líneas de datos, su anchura en bits. **No se evaluará ningún diseño de ruta de datos que carezca de esta información.**
- Dibujar la Unidad de Control y la Ruta de Datos (o Unidad de Proceso) como dos cajas negras y sobre dicho esquema indicar TODAS las señales que entran y salen de las cajas (incluidas las que van de UC a RD y viceversa) y la función que realiza cada una de ellas y momentos en que se activa en relación al diagrama de flujo.
- Diseñar la UC como máquina de Moore: a) diagrama de estados en que se refleje la relación de los estados con el diagrama de flujo del apartado 1) y

reflejando únicamente las señales (y sus valores) que dan lugar a cambios de estado y b) tabla de estados y salidas

- e) Añadir las señales de control internas del sistema diseñado y completar el cronograma siguiente (incluyendo el estado siguiente y el valor de las señales después de activar la señal de FIN):

Nota: El tiempo programado para que caiga agua sobre el coche son 6 s (contados a partir de la activación de la señal de A), y lo mismo para el enjabonado: son 6 s a partir de la activación de las correspondientes señales.



Problema 2. Diseñar la máquina de estados finita (ASM) que controla un juego electrónico capaz de adivinar números de 0 a 15, con las siguientes especificaciones:

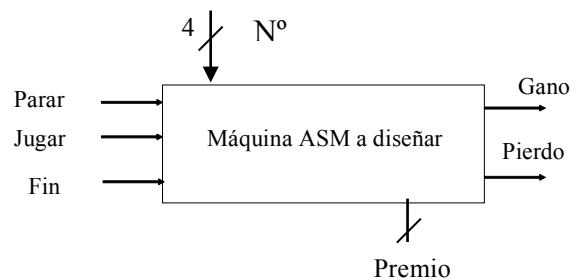
- La máquina proporciona 3 oportunidades para acertar el número, y dispone de 3 premios diferentes codificados en binario según la tabla descrita debajo.
- El juego dispone de un contador funcionando a una frecuencia de 100MHz que genera aleatoriamente números de 0 a 15.

El funcionamiento del juego es el siguiente: En primer lugar el juego está parado hasta que se pulsa el botón de **Jugar**. En ese momento, el usuario introduce el número que

cree va a salir, y el sistema comienza a generar números a una frecuencia de 100MHz, de forma que el usuario desconoce qué salida tiene el contador en cada momento. El sistema continúa generando números hasta que el jugador decide parar el juego y adivinar el número, pulsando el botón **Parar**. Si el número introducido por el usuario no coincide con el generado, el sistema ofrece al usuario otra oportunidad de juego (así hasta un total de 3 oportunidades), de modo que el usuario introduce un nuevo número y el sistema vuelve a generar más números aleatoriamente.

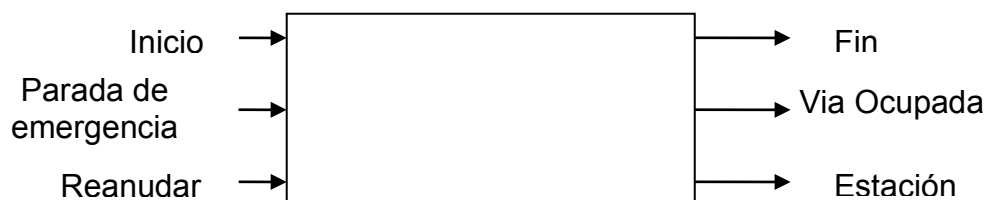
Si tras 3 fallos el usuario no ha acertado el número, se activará la salida **Pierdo** durante un ciclo de reloj, volviendo a continuación el sistema al estado inicial. Si en alguna de las oportunidades permitidas el usuario acierta el número, se activa la señal **Gano** hasta que el sistema ha terminado de dar el premio. Cuando se ha dado el premio, la máquina que suministra las monedas activa la señal de **Fin** (entrada de control de la máquina ASM) y el sistema vuelve al estado inicial.

Oportunidad	Premio (€)
Primera	100
Segunda	50
Tercera	25



- Dibujar el diagrama de estados-código RTL (organigrama) correspondiente a la máquina a diseñar
- Diseñar la ruta de datos (unidad de procesamiento) de la máquina. **Especificar claramente tamaño de registros, entradas y salidas de los módulos, funcionalidad, señales de control y su efecto sobre el sistema. Indicar qué señales van/vienen de UC y cuales a usuario u otros dispositivos**
- Diseñar la unidad de control del sistema (entradas y salidas, diagrama de estados y tabla de salidas; tabla de transición de estados no necesaria)

Problema 3. Diseñar un sistema algorítmico para controlar un tren que viaja desde la estación Origen a la estación Destino, situada a 50 km, deteniéndose en una estación Intermedia, que está a 15 km de Origen. El sistema tiene las siguientes entradas y salidas:



El comportamiento es:

- El tren empieza su viaje cuando el maquinista pulsa *Inicio*. En ese momento el tren se pone en marcha (avanza 1 km por ciclo de reloj) y activa la salida *Vía Ocupada*, que permanece activa hasta que el tren llegue a su destino. Cuando el tren llegue a su destino activa *Fin*.

- El tren sigue su marcha hasta que llega a la estación intermedia. En ella se para 10 ciclos, durante los cuales está activa la salida *Estación*.
- Si en cualquier momento el maquinista pulsa *Parada de emergencia* el tren se detiene hasta que el maquinista pulse *Reanudar*.

Problema 4. Diseñar un sistema algorítmico de reconocimiento de identidad en una sucursal bancaria. El sistema permite reconocer a 256 usuarios, cada uno de ellos con un código de usuario asociado y una clave de 6 bits. Cuando un usuario del banco activa la señal de *Reconocer*, el sistema le pregunta, primero, su código de usuario, y después su clave de acceso. Esta clave de acceso se compara con la que tiene almacenado el sistema, si coinciden la identificación es correcta y se activa la señal *Correcto*. Si no coinciden, el sistema volverá a preguntar por el código de identificación de usuario. A continuación, el usuario tendrá que pulsar el botón de *Entrar* para acceder a la entidad y se activará la señal *Abrir* durante un ciclo de reloj. Durante ese ciclo el usuario puede decidir cambiar la clave, y para ello tiene que pulsar el botón de *Cambio_clave*. Si desea cambiar la clave, el sistema pregunta por la nueva clave y la almacena, activándose la señal de *Fin* cuando el proceso concluye. Si no se desea cambiar la clave, el sistema espera 5 ciclos de reloj, transcurridos los cuales vuelve al estado inicial. Implementar el sistema como una máquina Moore.

- a) Diseñar la unidad de procesamiento con su correcto interconexión.
- b) Diseñar el diagrama de estados de la unidad de control. Especificar la interconexión entre las dos unidades, especificando que señales se activan en cada estado para que el comportamiento temporal del sistema sea correcto.
- c) Dibujar el organigrama que define el comportamiento del sistema.

Problema 5. Un contador cuenta en módulo 3 siempre que la señal de control M esté en 1, y cuenta en módulo 6 cuando la señal de control M está en 0.

Se debe tener en cuenta que el sistema completo dispone de 2 entradas de control: una señal de INICIO para que el sistema comience a funcionar y una señal de FIN que pararía el funcionamiento del sistema.

- a) Describa con un organigrama el sistema secuencial para ser implementado por una Máquina de estados algorítmica.
- b) ¿Podría identificar su Unidad de Control con un contador binario? Si es así, ¿de cuantos bits? ¿Cuál sería la Unidad de Procesamiento usando módulos estándar? ¿Qué señal o señales de control pasa la Unidad de Control a la Unidad de Procesamiento? ¿Qué señales de condición pasa la Unidad de Procesamiento a la Unidad de Control?

Problema 6. Un sistema suma todos los elementos de un vector que se encuentra almacenado en una memoria RAM. El sistema leerá la dirección inicial (*dir*, de 8 bits) del vector y su longitud (*n*, de 4 bits) cuando se active la señal *ini*. A continuación leerá uno a uno los elementos del vector (cada uno de 6 bits) calculando la suma de éstos. Cuando el sistema ha terminado de operar activará la señal de *fin* que indica que la salida *resultado* (de 8 bits) tiene el valor correcto y volverá al estado inicial.

- a) Diseñe la ruta de datos. Para la ruta de datos se pueden utilizar un sumador, dos contadores, un comparador y además el mínimo número de registros del tamaño adecuado.
- b) Diseñe la unidad de control, incluir diagrama de estados y tabla de salidas.

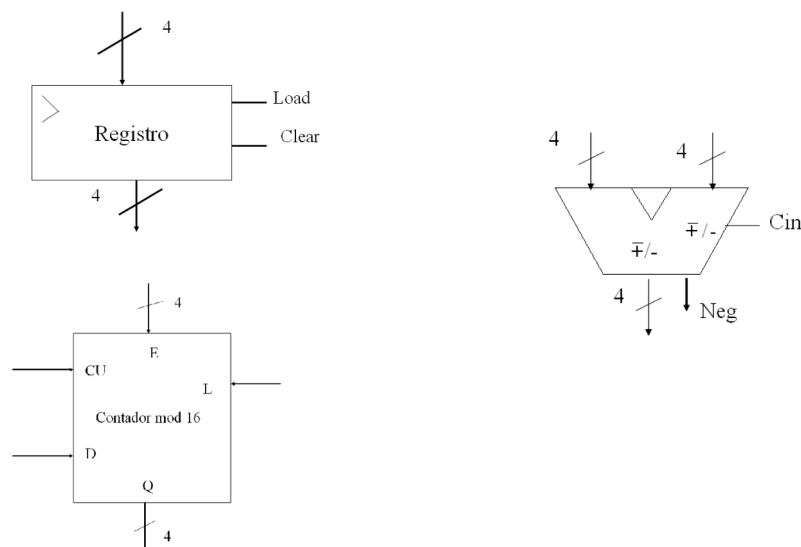
Problema 7. Diseñar un sistema algorítmico que comienza leyendo de la entrada las direcciones de los dos operandos. A continuación, si se activa la entrada empezar, lee el operando A (que es un dígito en BCD) de la memoria RAM, al ciclo siguiente lee el operando B (también en BCD) y realiza la suma. El resultado debe representarse de forma correcta con dos dígitos BCD. Si es menor que 10 no hace falta hacer nada, pero si es mayor se debe sumar 6 para que el resultado se represente correctamente. Cuando se haya realizado la operación la señal Fin se pone a 1 y el sistema vuelve al estado inicial.

Ejemplo1: $A=1000$ y $B=0001$; $A+B=00001001$. Si se interpreta el resultado como dos dígitos BCD es lo mismo que decir que $08+01=09$.

Ejemplo2: $A=1000$ y $B=1001$, $A+B=00010001$, = 11 (en BCD) que es claramente incorrecto. Para obtener el resultado correcto se debe sumar 6: $08+09=11+06=17$.

- La ruta de datos debe realizarse con los siguientes módulos: una memoria RAM, tres registros de 8 bits, un sumador y un módulo combinacional que nos dice si un número de 8 bits es menor que 10. Aparte de estos módulos únicamente se pueden utilizar multiplexores.
- La unidad de control se especificará con el diagrama de estados y la tabla de salidas.

Problema 8. Diseñar un juego electrónico de cartas similar a las 7 y media. El juego dispone de un contador que genera cartas con valor de 1 a 10. Su funcionamiento es el siguiente. En primer lugar el juego está parado hasta que se pulsa el botón de Jugar. A partir de ahí, el juego comienza a barajar las cartas hasta que el jugador decide Pedir_carta. La suma de todas las cartas pedidas se almacena en el registro marcador. Cada vez que se pide una carta se actualiza marcador y se compara con 11. Si $\text{marcador} > 11$, el jugador ha perdido, se activa la señal Pierdo y el sistema vuelve al estado inicial. Si por el contrario, $\text{marcador} \leq 11$, el sistema vuelve a barajar y el jugador decidirá de nuevo si pide o no una nueva carta. Si decide no pedir carta, puede ser debido a que quiere seguir barajando o se quiere plantar. En este último caso, se activa la señal Fin y el sistema vuelve al estado inicial.



Se pide:

- Diseñar la unidad de proceso, para su implementación sólo se pueden usar sumadores, contadores, registros (como los de las figuras) y puertas lógicas.

- b) Diseñar la unidad de control.
- c) Dibujar un cronograma que describa el estado, las salidas y las señales de control para la siguiente secuencia de eventos:

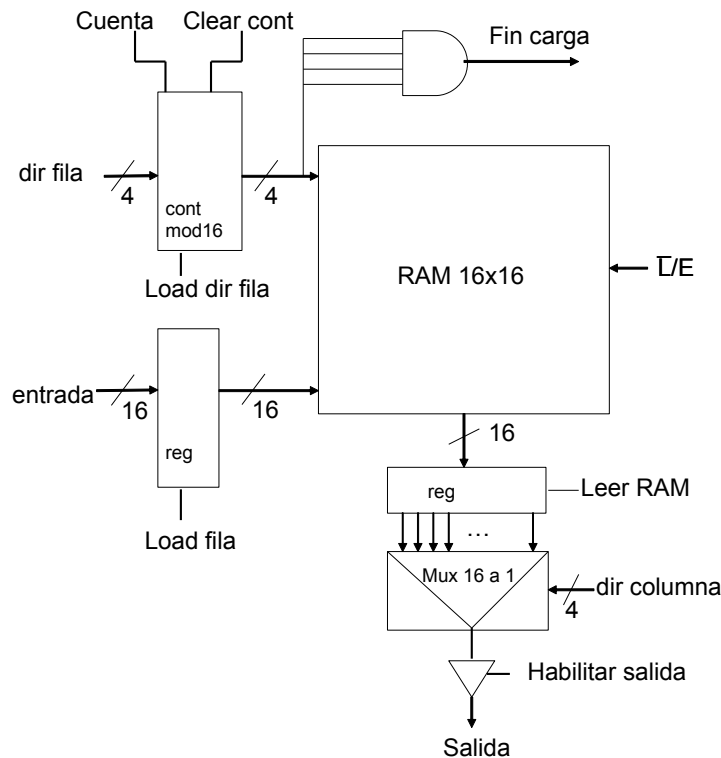
1º ciclo: Jugar=1, 2º ciclo: Pedir_carta=1 y sale un 4, 4º ciclo Pedir_carta=1 y sale un 5, 6º ciclo: Pedir_carta=0, Plantar=1.

Problema 9. Escribir el diagrama de flujo y programa RTL de una máquina expendedora de tickets de aparcamiento. La máquina tiene 2 botones: INICIO (desbloquea la ranura y permite introducir monedas de 1€ solamente) e IMPR_TICKET que imprime 3 tipos de tickets: 20 min. de aparcamiento (1€); 40 min. (2€) y 60 min. (3€). Si se da al botón INICIO y luego a IMPR_TICKET y no se han introducido monedas, la máquina no hace nada y vuelve al estado inicial. El procedimiento para sacar un ticket es: 1) pulsar INICIO; 2) introducir 1, 2 o 3 monedas de 1€ según el tiempo deseado; 3) pulsar IMPR_TICKET. Diseñar la ruta de datos. Indicar claramente las entradas, las salidas y las señales de control del sistema, así como la función que realizan los módulos utilizados para la ruta de datos. Diseñar la unidad de control.

Problema 10. Escribir el diagrama de flujo y programa RTL de un multiplicador de números de 4 bits usando un contador. Diseñar la ruta de datos correspondiente indicando el tamaño de los registros, códigos de operaciones etc. Indicar claramente las entradas, las salidas y las señales de control así como la función que realizan los módulos utilizados para la ruta de datos. Diseñar la unidad de control.

Problema 11. Escribir el diagrama de flujo y programa RTL de un sumador de números binarios de 8 bits en magnitud y signo (1 bit de signo y 7 de magnitud). Diseñar la ruta de datos correspondiente indicando el tamaño de los registros, códigos de operaciones etc. Indicar claramente las entradas, las salidas y las señales de control así como la función que realizan los módulos utilizados para la ruta de datos. Diseñar la unidad de control.

Problema 12. Diseñar un sistema que controle el funcionamiento de un juego de barcos. En la figura se muestra la ruta de datos utilizada para implementar una parte del juego. La señal de reloj no aparece para simplificar la figura. El juego comienza cuando el jugador pulsa el botón de encendido. Si en esos momentos también se ha pulsado el botón Nuevo_mapa, el sistema entra en el modo grabar una configuración nueva. En la memoria RAM se almacenan secuencialmente las 16 palabras de 16 bits que definen el nuevo mapa, en el que se han situado los barcos correspondientes. Si por el contrario, no se pulsa el botón Nuevo_mapa, el jugador suministra las coordenadas (fila, columna) y el circuito debe informarle si en esa posición hay un barco o hay agua (1 quiere decir tocado y 0 agua). Hacer el diagrama de flujo que dirige el funcionamiento del juego. Diseñar la unidad de control, incluyendo el diagrama de estados y la tabla de salidas.



Problema 13. El laboratorio de Rendimiento Deportivo de la Facultad de Medicina cuenta con un equipo digital para la medida de velocidad instantánea en atletas. Para ello, los atletas se disponen al inicio de una plataforma horizontal y, cuando desean comenzar el experimento, activan el pulsador de Inicio. El sistema de medida detecta al atleta en cuestión gracias a que su código ID (de 4 bits) se encuentra codificado en una tarjeta de identificación por radiofrecuencia, la cual porta el atleta en su bolsillo.

Inmediatamente después de activar el pulsador de Inicio, el atleta comienza a correr a lo largo de la plataforma, siendo su movimiento detectado gracias a un sensor de presión. Mientras dura el movimiento del atleta, el tiempo transcurrido (siempre menor de 12 ciclos para los atletas considerados) es medido por un contador de ciclos de reloj. Si a la finalización del movimiento el atleta ha mejorado su marca, la cual está almacenada en la memoria RAM del sistema de medida, ésta se actualiza con la nueva medida y la nueva Marca se presenta por el display del sistema. Al final del ciclo de medida, se activa la señal Fin que informa a los médicos deportivos de la finalización del experimento.

Problema 14. Diseñar un sistema de control de apertura de una puerta. El sistema funcionará del siguiente modo:

Inicialmente, la puerta está cerrada. Cuando la señal externa **Inicio** se pone a 1 se activa un reloj (señal **tiempo**, que cuenta 16 ciclos del reloj del sistema), de forma que el usuario tiene 16 ciclos para introducir la clave correcta. La clave consta de 4 bits que se introducen en serie (señal **clave**) comenzando por el bit más significativo. El sistema comienza a leer los bits de la clave cuando se activa la señal **Inicio**. Cuando los 4 bits se han introducido, esta clave se compara con la almacenada en la puerta, y si las dos claves son iguales la puerta se abrirá, sino el sistema permite al usuario una nueva oportunidad, así hasta 2 veces. Si tras dos intentos, el usuario no acierta la clave la puerta se bloquea (señal **Bloqueo** a '1'). La puerta, por lo tanto, se bloqueará si no se acierta la clave en dos intentos o si una vez introducida la clave correcta se han

superado 16 ciclos de reloj del sistema. Una vez que la puerta se bloquea permanecerá bloqueada hasta que un empleado de seguridad active la señal **Desbloqueo**, volviendo en este caso el sistema al estado inicial. Si la puerta se abre permanecerá abierta un ciclo de reloj y después el sistema vuelve al estado inicial.

