

PRÁCTICA 0. Trabajo final

Introducción al LogicWorks, puertas lógicas, análisis lógico de circuitos combinacionales y encapsulado de un dispositivo sencillo

Objetivos que se deben alcanzar al realizar la práctica

Después de realizar esta práctica, además de haber mejorado el nivel de consecución de los objetivos necesarios para preparar la práctica (citados en la tabla anterior), el alumno deberá saber manejar el programa LogicWorks, con el cuál se realizan las prácticas de esta asignatura, y en particular será capaz de:

- 1) Dibujar un circuito en una hoja de LogicWorks compuesto por las puertas Not-1, And-2 y Or-2 de la librería de dispositivos.
- 2) Simular el funcionamiento del circuito usando el dispositivo binary switch para dar valor a las señales de entrada y binary probe para visualizar el valor de las señales internas y de salida.
- 3) Encapsular un circuito en un dispositivo (símbolo) vacío y salvarlo en la librería de dispositivos de LogicWorks para su posterior utilización en circuitos más complejos.

Directorio de la práctica

Mi PC\Z:\Prac0

0.1 Introducción al diseño modular

En este curso construiremos un computador sencillo, pero completo, utilizando muy pocos elementos distintos. Para ello usaremos solamente cuatro dispositivos muy simples, tres combinacionales, las puertas lógicas Not, And y Or, y uno secuencial, el biestable D activado por flanco, que denominamos Flip-flop. Será como jugar con un mecano que solamente tiene cuatro tipos de piezas, aunque tendremos muchas piezas de cada tipo. El primer procesador que diseñamos hacia el final del curso, que denominamos SISC Harvard unicycle, está compuesto por unos 3000 dispositivos sencillos (aproximadamente 400 puertas Not, 1600 puertas And, 800 puertas Or y 128 Flip-flops) y las interconexiones entre ellos. Construir todo el procesador en un solo nivel, con tantos elementos, sería muy complejo, muy fácil cometer errores y muy difícil de entender. Para evitar estas dificultades seguimos un método de diseño modular (a bloques) multinivel.

Primero diseñamos circuitos sencillos, aunque un poco más complejos que las puertas lógicas básicas. Para cada uno de estos circuitos asociamos un símbolo, con sus patillas de entrada y de salida. Después ya podemos usar estos símbolos (dispositivos) para que, interconectándolos entre sí, podamos construir circuitos un poco más complejos, a los que también asociaremos otro símbolo, que podrá ser usado como un dispositivo más en otros diseños más complejos todavía. Por ejemplo, en la práctica 1 construimos una puerta Xor, que se forma con dos puertas Not, dos And y una Or. Después construimos un Half-adder, que consta de una puerta And y una Xor como la construida antes. Este dispositivo lo usamos en la práctica 2 para construir un Full-adder formado por dos Half-adders y una puerta Or y luego construimos un sumador de números de 16 bits con 16 Full-adders. De esta forma, a lo largo del curso, llegamos a construir el computador completo, de abajo hacia arriba (de lo más sencillo a lo más complejo).


En un diseño modular como el que seguimos en este curso, es fácil detectar y corregir los errores. Cada dispositivo se construye con pocos elementos de los niveles inferiores. La sencillez del circuito hace que no sea muy probable que cometamos errores y si los cometemos, será fácil detectarlos. Además, cada dispositivo nuevo se prueba exhaustivamente antes de ser incorporado a la librería de dispositivos, de forma que cuando es usado para construir otros dispositivos más complejos, es seguro que funciona correctamente. Por último, y lo que es más importante para nosotros, el diseño modular hace que vosotros, los estudiantes, comprendáis mejor el funcionamiento del computador. Sólo tenéis que creer el funcionamiento de las 3 puertas lógicas básicas y el Flip-flop. No entraremos en el interior de estos dispositivos (que están contruidos con transistores, asunto que estudiaréis en asignaturas de Física, de Electrónica, o de diseño VLSI). El funcionamiento interno de estos cuatro dispositivos es lo único mágico de esta asignatura, el resto es lógico y lo podéis comprender perfectamente.

0.1.1 Mirando dentro del Procesador

Con el objetivo de motivaros a estudiar esta asignatura, lo primero que vamos a hacer en esta práctica es mirar cómo está construido por dentro un computador basado en un procesador sencillo muy parecido al SISC Harvard uniciclo, denominado SISP-I-1, cuyo funcionamiento comprenderéis perfectamente estudiando esta asignatura. En la carpeta de esta primera práctica tenéis un circuito realizado con el programa LogicWorks (versión 4.1) que contiene un computador completo que podéis mirar por dentro y ver cómo está construido.

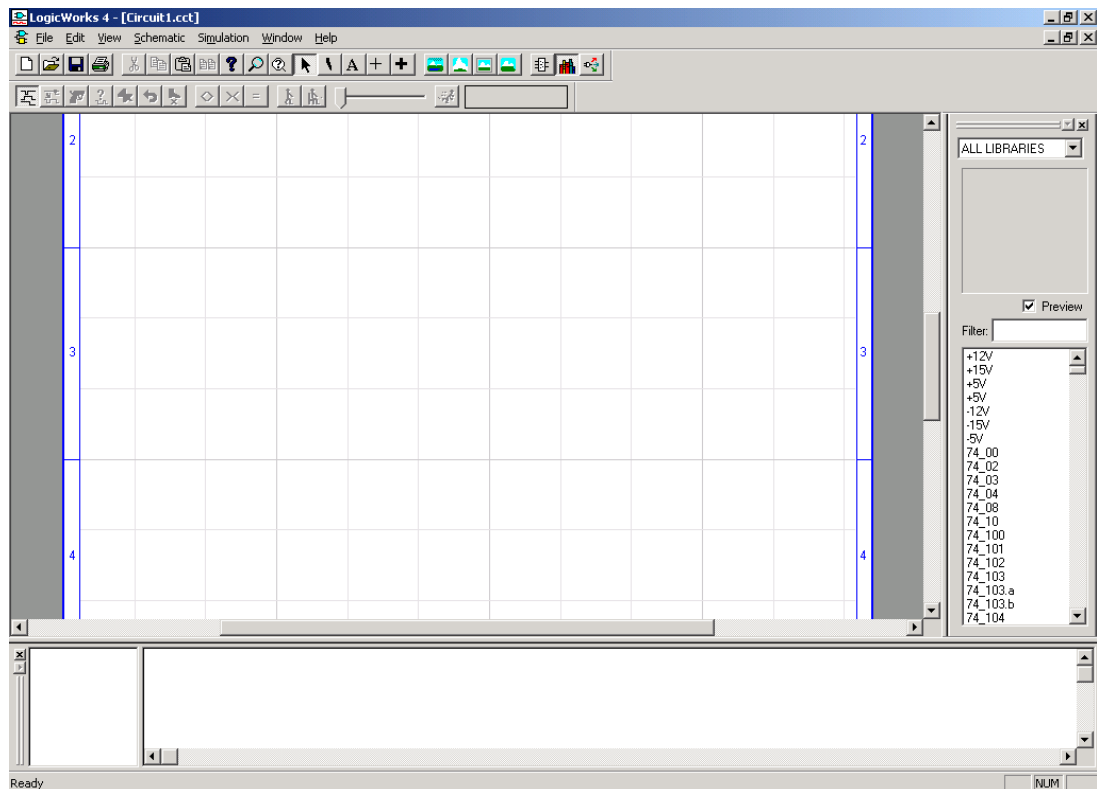
LogicWorks es la aplicación informática que usamos para diseñar los circuitos que componen el computador y simular su funcionamiento, como si los circuitos estuvieran implementados en silicio. De hecho, una vez diseñado un circuito en LogicWorks existe la posibilidad de generar la información necesaria para su fabricación en un chip. ¡Pero su fabricación es un proceso muy caro!


Estos son los pasos que debéis seguir para abrir el LogicWorks y ver el circuito del computador:

- 1 Encended vuestro PC y arrancad con el sistema operativo Windows (si es que no está ya preparado).
- 2 Cuando Windows se haya iniciado, entrad con el mismo usuario y contraseña que usáis para el RACÓ de la FIB.
- 3 Desde el escritorio de Windows, haced doble clic sobre el acceso directo a LogicWorks 4.1 para abrir el programa o id al menú Inicio → Programas → LogicWorks 4 (para los aularios el módulo D6) o menú Inicio → Programas → Software Asignaturas → LogicWorks 4 (para los aularios de los módulos A5 y C6) y haced clic sobre la opción LogicWorks 4.
- 4 Al abrirse la aplicación aparece una pantalla como la que se muestra en la figura de la página siguiente. En la zona central aparece una hoja de edición de circuitos nueva, para poder dibujar un circuito que tiene por defecto el nombre Circuit1.cct. Pero ahora no vamos a dibujar un nuevo circuito sino a ver uno ya creado.
- 5 Seleccionando File → Open se abre una ventana de exploración a través de la cual podéis acceder al fichero SISP-I-1.cct, en el directorio de la práctica, que contiene el circuito del computador.
(Los puntos 4, 5 y 6 se pueden sustituir por: Una vez en el escritorio de Windows abrid el fichero SISP-I-1, que se encuentra en el directorio de esta práctica, haciendo doble clic en el icono).
- 6 Una vez se ha abierto el circuito del computador podéis observar que está formado por solamente 5 dispositivos (símbolos/cajas con patillas) interconectados entre sí: Clk, Input/Output, SISP-I-1, Program Memory y Data Memory. Para mirar cómo está implementado por dentro un dispositivo tenéis que hacer doble clic sobre el símbolo del dispositivo. De esta forma se abre una nueva ventana que muestra el circuito interno del dispositivo. Este circuito a su vez está formado por varios dispositivos interconectados entre sí. Si hacéis doble clic sobre cualquiera de estos dispositivos se abre su circuito interno y así hasta que lleguéis a un dispositivo como son las puertas And y Or de dos entradas, la puerta Not o el Flip-flop, que ya no pueden abrirse. Estos dispositivos básicos no están contruidos con otros dispositivos sino que son simulados directamente por el programa. (Cuando se intenta abrir uno de estos dispositivos que no tienen circuito interno asociado, aparece un mensaje que nos lo indica). Una vez abierto el circuito que implementa un dispositivo, para cerrarlo y subir al nivel donde está el dispositivo, tenéis que hacer Ctrl+Shift+U (o haced clic en el botón  que se encuentra en la parte superior derecha de la ventana para cerrarla —ojo, no cerréis el programa LogicWorks).

Mirad el contenido de los circuitos que forman el procesador SISP-I-1 y veréis que es bastante complejo. Si seguimos la siguiente secuencia de mirar dentro de dispositivos partiendo de la ALU llegamos a un circuito con dos puertas Not, dos And y una Or: ALU → AL → ADD → Fa → Ha → Xor → Puertas básicas.

A pesar de la complejidad del computador, no os asustéis porque vais a entender todo **paso a paso** y cada paso será fácil de comprender. ¡Fijaos en cuántas cosas nuevas vais a aprender en este curso!




Cuando terminéis de mirar el computador no olvidéis cerrar el circuito haciendo clic sobre el botón  de la parte superior derecha de la ventana de edición de circuitos. Al cerrarse el circuito del computador aparecerá la hoja en blanco del Circuito1.cct.

0.2 Las puertas lógicas básicas

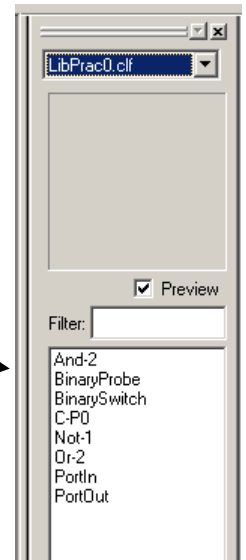
En esta sección nos familiarizamos con el funcionamiento de tres dispositivos que denominamos puertas lógicas básicas, las únicas puertas que usaremos para construir el nuestro computador: *Not*, *And* y *Or*. Ya las conocéis por haber preparado el informe previo de esta práctica. Ahora vamos a trabajar con ellas usando el simulador de circuitos LogicWorks.



Antes de comenzar a usar la librería de dispositivos LogicWorks de la práctica, vamos a copiarla del disco donde se encuentra al escritorio de nuestro PC. Esto lo haremos siempre al principio de cada sesión de laboratorio, ya que normalmente vamos a crear o modificar dispositivos de esta librería para usarlos posteriormente. Para ello necesitamos poder escribir en esta librería y el disco en el que se encuentra es el mismo para todos los usuarios y está protegido contra escrituras. Por eso la copiamos en nuestro escritorio, en el que sí podemos escribir. La librería es un fichero denominado **LibPrac0** (para la práctica 1 se llamará LibPrac1, etc.) y lo copiamos desde el directorio de la práctica al escritorio.

Ahora vamos a comprobar, usando el simulador LogicWorks, que las puertas lógicas de nuestra librería de dispositivos funcionan correctamente. Para ello seguiremos los pasos siguientes:

- 1) En el centro de la pantalla tenemos la ventana de edición de circuitos (ver figura anterior con la pantalla de LogicWorks). Es una hoja blanca cuadrículada, que aparece en la parte central de la pantalla, donde iremos introduciendo los diferentes elementos que tendrá nuestro circuito. En la parte de arriba de la pantalla se visualizan la barra de herramientas de dibujo de circuitos y la barra de herramientas de simulación de circuitos. A la derecha de la pantalla, tenemos la ventana para poder acceder a las librerías de dispositivos. Debajo tenemos la ventana de tiempos, que sirve para visualizar la evolución en el tiempo de las señales del circuito (cronogramas). De momento, como no vamos a utilizar la ventana de tiempos la cerramos, haciendo clic (si no decimos lo contrario siempre nos referimos al botón izquierdo del ratón) sobre el icono de cerrar  que está en la parte superior derecha de la ventana de tiempos.

- 2) En la parte superior de la ventana de las librerías de dispositivos tenemos una lista desplegable donde podemos escoger la librería que contiene los elementos necesarios para realizar la práctica. Si desplegamos la lista no encontramos nuestra librería, ya que por defecto no está abierta. Para abrir nuestra librería tenemos que hacer clic con el botón derecho del ratón sobre la ventana de las librerías y elegir Open Lib... de entre las opciones de menú que aparecen. Se abre una ventana de navegación a través de la cual llegamos a nuestra librería que se encuentra ya copiada en el escritorio, la seleccionamos y la abrimos. Ahora ya podemos desplegar la lista de librerías y elegir la nuestra, LibPrac0. (Imagen de la derecha). Al hacer esto, en la lista de más abajo han aparecido los componentes de la librería por orden alfabético (And-2, Not-1, etc.).



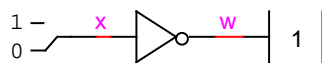
- 3) Empezamos por la puerta Not-1. Para añadirla al circuito, nuestro cursor deberá tener forma de flecha. Si no es así pulsamos sobre el icono con forma de flecha  que se encuentra en la barra de herramientas que aparece en la parte superior de la ventana. Ahora, hacemos doble clic sobre el elemento Not-1 de la lista de elementos nuestra librería.
- 4) En este momento, el cursor habrá tomado la forma de la puerta Not. Para añadirla al circuito tan sólo hemos de hacer clic en el lugar de la hoja de edición donde queramos ponerla. El cursor mantiene la forma de la puerta por si queremos introducir alguna puerta más siguiendo el mismo procedimiento. Como nosotros no queremos introducir ninguna más, pulsamos la barra espaciadora del teclado para que el cursor vuelva a la forma de flecha.
- 5) Lo siguiente es estirar las patillas de la puerta para crear cables (señales binarias). Con el cursor en forma de flecha (a partir de este momento, si no se indica lo contrario, el cursor deberá tener forma de flecha para hacer las acciones que se indiquen), hacemos clic en el extremo libre de la patilla (pin) izquierda de la puerta (la patilla de entrada de la puerta), y sin soltar el botón arrastramos el ratón para estirar el cable. Hacemos lo mismo con la patilla de la derecha, la de salida.
- 6) Ahora damos nombre a los cables. Para ello hacemos clic sobre el icono  de la barra de herramientas, nuestro cursor se convertirá en un lápiz. Seguidamente hacemos clic sobre el extremo del cable izquierdo (señal de entrada). Se abrirá un cuadro de texto donde introducimos su nombre, en este caso, su nombre será: x. El cursor sigue siendo un lápiz. **OJO**, hay que comprobar que lo que hemos hecho realmente ha sido nombrar el cable y no simplemente incluir una nota en la hoja de edición. Para ello, nos fijamos en que las letras que dan nombre al cable son de color rosa (que es el color de los nombres de las señales o cables) y no negro, que es el color de las notas, ni azul, que es el color de los números de las patillas). Otra posible comprobación consiste en hacer clic sobre el nombre o sobre el cable y ambos, el nombre y el cable, se deben seleccionar, el cable de color amarillo y la viñeta del nombre debe ensombrecerse. Si alguna de estas dos cosas no ocurre, volveremos a seleccionar lo escrito mediante la flecha y lo borramos usando la tecla de borrado del teclado e intentamos volver a nombrar el cable de nuevo. Estos fallos se deben a que, a veces, creemos que estamos haciendo clic con el lápiz sobre el cable pero no es así, posiblemente porque estamos utilizando un zoom demasiado bajo.


Para cambiar el zoom de la hoja de edición usaremos los comandos Ctrl+Shift+E para acercar y Ctrl+Shift+R para alejar. O con el ratón en las opciones View → Zoom In (Enlarge) / Zoom Out (Reduce) de la barra de menús del LogicWorks.

Otra forma, más segura, de nombrar el cable es hacer clic con el botón de la derecha del ratón sobre el cable, si hemos afinado bien se ilumina el cable en amarillo y se despliega un menú cerca del cable. Hacemos clic con el botón de la izquierda sobre la opción "Name..." y aparece una ventana donde escribimos el nombre del cable y hacemos clic en OK (en esta ventana normalmente ya está seleccionada la opción "visible", si no es así la seleccionamos antes de hacer OK). Hacemos lo mismo para dar nombre al cable de salida de la puerta que denominamos w.

- 7) Ahora podemos añadir los elementos que nos permitirán poner valores a la entrada y observar el resultado a la salida de la puerta. En primer lugar daremos valor a la entrada de la puerta conectando un Binary Switch, que es el dispositivo que usa el simulador LogicWorks para ello. El Binary Switch sólo tiene una salida que va tomando alternativamente los valores 0 y 1 cada vez que se hace clic sobre el símbolo del dispositivo. Para obtenerlo, hacemos doble clic sobre el elemento Binary Switch de la lista de dispositivos de la librería, y lo ponemos en la hoja de edición de manera que el extremo del Switch quede conectado con el inicio del cable que saquemos de la patilla de entrada x. Si el elemento no está encarado a nuestro gusto, mientras el cursor tenga la forma del elemento y siempre antes de ponerlo en la hoja de edición, podemos girarlo para encararlo usando las cuatro flechas del teclado encargadas del movimiento del cursor (esto se puede hacer con todos los dispositivos que salen de la librería). El efecto de cada flecha sobre el dibujo, no es evidente; es cuestión de probar varias secuencias de flechas hasta obtener el resultado deseado.

- 8) Para visualizar el valor presente a la salida de la puerta usaremos el Binary Probe, que es el dispositivo que usa el simulador de LogicWorks para ello. De forma análoga a la anterior escogeremos de la lista de elementos el Binary Probe y lo conectaremos al extremo libre del cable conectado a la salida de la puerta w. El circuito resultante ha de quedar como el de la figura.



- 9) Una vez montado el circuito, hay que probarlo. Primero aseguraos que la simulación está en marcha. Mirad si en la barra de herramientas de las tres figuras que hay de una persona (sentada, andando y corriendo) está seleccionada la de la persona corriendo (como indica la figura adjunta,  aunque parezca confuso, ya que se marca en negro la opción que podemos seleccionar a continuación. En la figura la simulación está corriendo porque la única opción que hay cuando la simulación está corriendo es pararla (persona sentada). Si hacemos clic a la persona sentada se pondrán en negro las opciones de persona andando y corriendo, ya que después de parar la simulación se puede pasar a simular paso a paso o seguido.). También veréis que a la derecha de esta figura hay un recuadro con números. Estos números representan las unidades de tiempo transcurridas desde que se inició la simulación, y se incrementan a cada acción del simulador. Ahora tan solo hemos de hacer clic sobre el Binary Switch para cambiar el valor de entrada de la puerta y observar cuál es el valor de la salida mirando el Binary Probe y observar si el dispositivo actúa de acuerdo a la tabla de verdad de la puerta Not.
- 10) Siguiendo los mismos pasos que para la Not-1, implementad los circuitos para comprobar el comportamiento de la And-2 y la Or-2. ¡Ojo! si en una misma hoja de diseño (un mismo circuito) colocáis las tres puertas tenéis que dar nombres distintos a los cables de entrada y salida de cada puerta. La razón es que en LogicWorks cuando dos cables tienen el mismo nombre el simulador considera que están conectados (como si hubiera un cable que los conectase). Esto se denomina conexión por nombre y es muy útil en circuitos complejos para ahorrarnos dibujar algunos cables y para que el dibujo quede más claro. Si usáis una hoja (circuito) distinta para cada puerta, podéis usar los mismos nombres sin problema.

□ Informe final

Pregunta 1

Comprobad que para cada una de las posibles combinaciones de las entradas, cada una de las puertas And, Or y Not funciona como especifica su tabla de verdad. Esto es, ¿funcionan correctamente las puertas de nuestra librería?

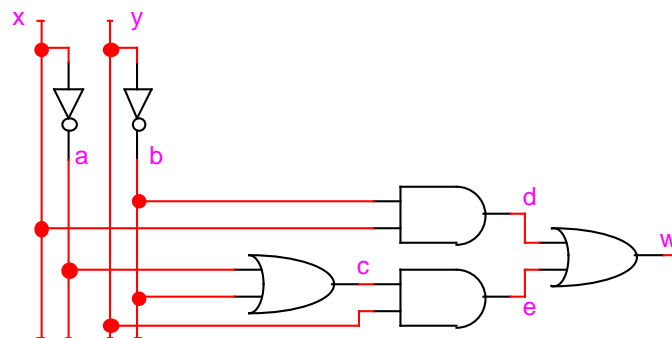
Si alguna puerta no funciona correctamente, pregunta a tu profesor de laboratorio dónde conseguir una puerta correcta, para que puedas continuar haciendo la práctica.

0.3 Implementación de un circuito formado por varias puertas

Ahora vamos a implementar en LogicWorks un circuito formado por varias puertas básicas interconectadas entre sí y comprobar su correcto funcionamiento. Este circuito, que denominamos C-P0 (Circuito de la Práctica 0), es muy sencillo. Se podría decir que es una puerta lógica ya que tiene solamente 1 salida (implementa una sola función lógica) y 2 entradas. Después, este circuito lo vamos a asociar a un símbolo (una caja etiquetada con C-P0 y con dos patillas de entrada y una de salida), para formar un dispositivo de la librería y poderlo usar en la siguiente práctica para formar circuitos o dispositivos más complejos.


0.3.1 Esquema lógico

La siguiente figura muestra un esquema lógico del circuito C-P0.



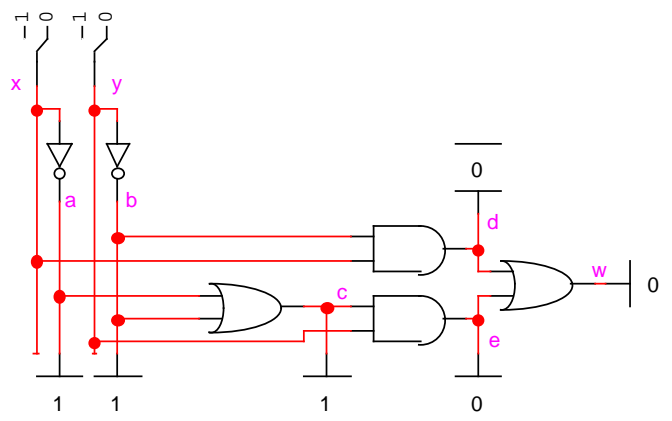
0.3.2 Implementación del circuito en LogicWorks

Vamos ahora a implementar el circuito en LogicWorks y comprobar su correcto funcionamiento. Para implementarlo seguid las indicaciones de edición de circuitos que comentamos anteriormente (Obtención de dispositivos de la librería, conexión de cables, dar nombre a los cables...). Os será útil saber lo siguiente:

- 1) Como podéis ver en el circuito anterior, las señales x e y vienen dadas por dos cables verticales. Para crear estos cables utilizad el icono  de la barra de herramientas. Una vez hayáis hecho clic sobre él, el cursor se convertirá en una cruz. El primer punto donde hagáis clic será el punto de inicio del cable, moviendo el ratón podéis estirar el cable, y para terminarlo haced doble clic sobre el punto donde queráis que acabe.
- 2) La forma de dar nombres a los cables es la misma que ya se ha comentado, y para unir patillas de salida-entrada de dos dispositivos, o para unir una patilla de un dispositivo a un cable, simplemente haced clic en el extremo de la patilla, arrastrad el ratón para estirar el cable y soltad el botón cuando veáis que se unen. Si lo habéis hecho bien, en la unión de dos cables perpendiculares cuando uno de ellos continúa, aparecerá un punto rojo como los que veis en la figura (no aparece punto en un cable que hace un ángulo de 90 grados).
- 3) Seguid el siguiente orden a la hora de implementar el circuito:
 - Primero dibujad los cables verticales para las señales x e y.
 - Después colocad las puertas Not-1 conectadas a los cables verticales.
 - Estirad las salidas de las Not-1 para crear los cables verticales de las señales !x e !y.
 - Seguidamente colocad las puertas And-2 y Or-2 y sus conexiones a los cables verticales para implementar !yx e !y+!x, después la puerta And-2 para (!y+!x)y, y por último, la puerta Or-2, su conexión y la salida.

0.3.3 Comprobación del circuito


Para comprobar el circuito, debéis añadir Switches a las entradas de los cables x e y, y un Binary Probe a la salida w y también en los cables a, b, c, d y e por si queréis ver los resultados intermedios. Comprobad que el circuito construido se comporta como dice la tabla de verdad de la función w que habéis encontrado en el análisis de este mismo circuito al responder a la pregunta 4b del Informe Previo. Para ello, dad valores a los Switches y mirad el resultado que toma la salida w (y las señales intermedias: a, b... e) en los Binary Probe. El aspecto del circuito debe ser como este.

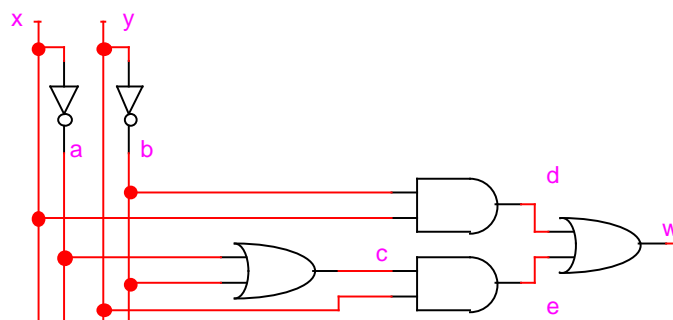


0.3.4 Encapsulado del circuito

Ahora vais a crear un dispositivo en la librería de dispositivos que tenga internamente este circuito que acabáis de implementar. Así podremos usar este dispositivo, al igual que el resto de puertas, para implementar circuitos más complejos. En la librería de dispositivos LibPrac0 encontraréis un dispositivo denominado C-P0. Es un símbolo (una caja con tres patillas y el nombre C-P0 dentro de la caja) para este dispositivo, que los profesores hemos creado. De esta forma, vosotros no tenéis que dibujar símbolos con el editor de dispositivos. Para asociar el circuito al símbolo (encapsular el circuito dentro del símbolo y formar un dispositivo) hay que proceder como sigue (este procedimiento lo tendréis que repetir en todas las prácticas, pero solamente en ésta lo explicamos con el máximo detalle, después supondremos que ya sabéis hacerlo).

Lo primero que debéis hacer es preparar el circuito para poderlo encapsular. Quitad los Binary Switch y los Binary Probe que habéis introducido en el circuito. Para ello, haced clic encima del dispositivo a borrar y una vez que esté seleccionado (lo notaréis porque se resalta en negro) los borráis utilizando la tecla de suprimir del teclado. Para seleccionar un Binary Switch hay que hacer clic sobre él, teniendo la tecla Shift pulsada. No se puede borrar un segmento de un cable seleccionado, pues se borrarían todos los cables conectados al mismo –todos ellos se iluminan al seleccionarlo– y no sólo el segmento deseado. Para borrar segmentos de cable se debe usar el botón Zap (delete). Después de hacer clic en

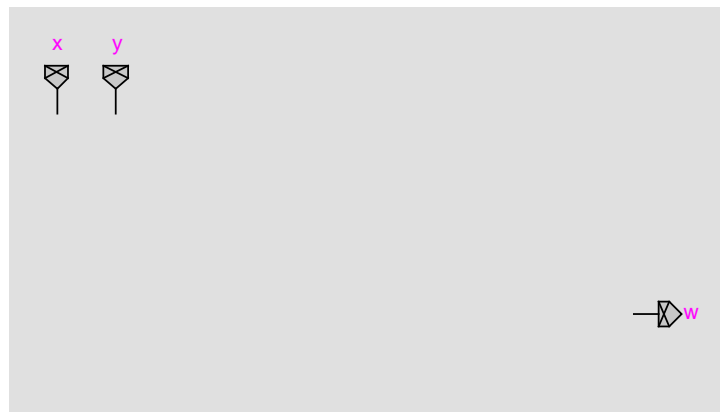
el botón  el cursor se transforma en un rayo, similar al dibujado en el botón. Ahora se puede borrar cualquier segmento de línea haciendo clic con el nuevo cursor. También puede usarse el cursor para borrar dispositivo, nombres,... Para volver al cursor normal hay que pulsar la tecla Esc (Escape). El circuito una vez limpio debe quedar como se muestra aquí (los nombres de las variables también podrían haberse borrado).



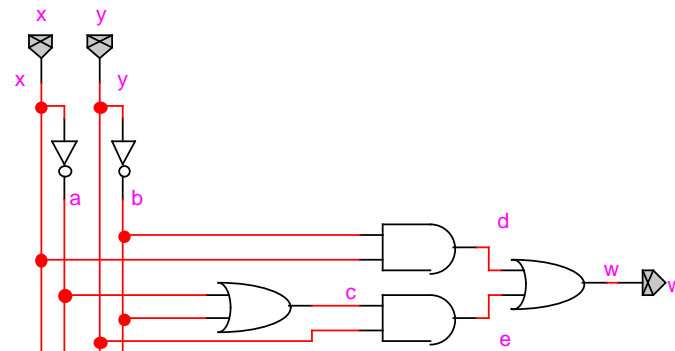
El siguiente paso es asociar el símbolo C-P0, que hemos creado nosotros y que se encuentra en la librería de dispositivos LibPrac0, con el circuito que habéis dibujado vosotros. Para ello seguid los siguientes pasos.

- 1) Seleccionad todo el circuito arrastrando el cursor sobre el circuito hasta cubrirlo completamente y copiadlo (con Edit → Copy o haciendo Ctrl+C), para luego poderlo pegarlo donde interese.

- 2) Ahora añadid un símbolo C-P0 desde la librería de dispositivos a una hoja de circuito. Podéis añadirlo en un trozo en blanco de la hoja del circuito que habéis creado o abrid una nueva hoja en blanco (File → New → Design → Aceptar) y pegadlo allí. Para añadir el símbolo C-P0 a la hoja de circuito, el cursor deberá tener forma de flecha. Si no es así pulsad sobre el icono con forma de flecha de la barra de herramientas que aparece en la parte superior de la ventana. Ahora, haced doble clic sobre C-P0 en la lista de elementos de la librería LibPrac0, que se ve en la ventana de la derecha de la pantalla. En este momento, el cursor habrá tomado la forma del símbolo que hemos diseñado nosotros para este dispositivo (una caja cuadrada con tres patillas). Para añadirlo a la hoja de circuito tan sólo debéis hacer clic en el lugar de la hoja de edición donde deseéis ponerlo. El cursor mantiene la forma del símbolo por si queréis introducir alguna copia más siguiendo el mismo procedimiento. Como no es necesario introducir ninguna más, pulsad la barra espaciadora del teclado para que el cursor vuelva a la forma de flecha.
- 3) El dispositivo C-P0 no es sólo un símbolo vacío, sino que nosotros ya le hemos asociado un circuito al crearlo. Lo que ocurre es que el circuito asociado sólo tiene los conectores (*port connectors*) para asociar los cables que forman las entradas y salidas del circuito con las patillas del símbolo, que son las entradas y salidas del dispositivo. Si hacemos doble clic sobre el símbolo del dispositivo C-P0, para ver su circuito interno, nos aparece un circuito como el siguiente. Este circuito asociado no tiene ninguna puerta ni interconexiones entre ellas. Sólo tiene los conectores de las entradas y salidas con su nombre asociado. Completar este circuito con el que vosotros habéis diseñado es lo que debéis hacer vosotros a continuación.

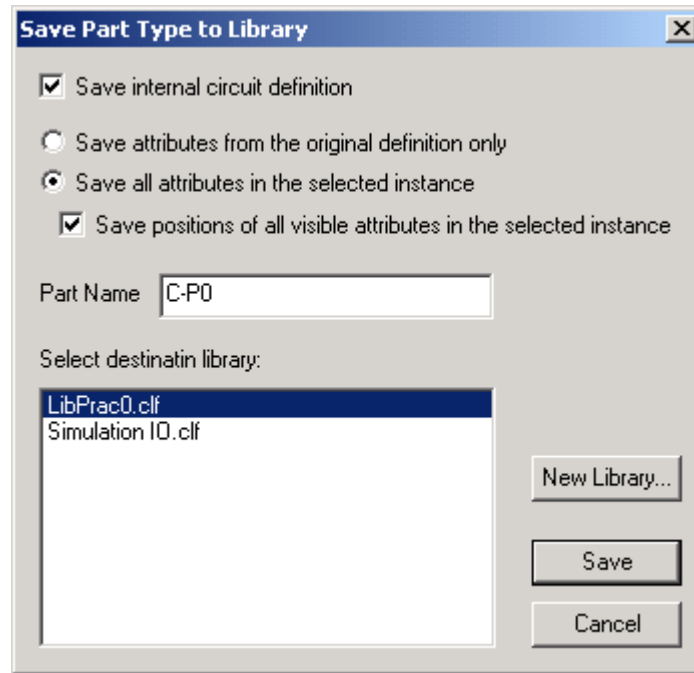


- 4) Para ello, debéis pegar el circuito C-P0 que habéis copiado antes sobre el circuito interno del dispositivo C-P0. Una vez pegado debéis conectar los *port connectors* de entrada x e y que se encuentran en el circuito interno del dispositivo a los cables x e y de entrada de vuestro circuito recién pegado. También debéis conectar la salida w de vuestro circuito con conector de salida w.
- Esto se hace estirando de las patillas de los port conectores y/o usando el botón para crear cables. El resultado debe ser un circuito como el siguiente. Es muy importante que no borréis los conectores ni cambiéis su nombre. Podéis mover los conectores (seleccionando y arrastrando), podéis cortar un conector y pegarlo en otro sitio (seleccionando, haciendo Ctrl+X y luego Ctrl+V), pero finalmente deben quedar conectados a las señales de entrada o salida y con los nombres que nosotros les hemos dado, de lo contrario el dispositivo no funcionará correctamente.



- 5) Cerrad el circuito interno del dispositivo que acabáis de completar para subir de nivel, a la hoja de circuito donde está el dispositivo. Para ello tenéis que hacer Ctrl+Shift+U (o haced clic en el botón que se encuentra en la parte superior derecha de la ventana para cerrarla –ojo no cerréis el programa LogicWorks).


- 6) Ahora, el símbolo que se ve en la hoja de circuitos, la caja con las tres patillas etiquetada con C-P0, ya tiene asociado el circuito correcto que vosotros habéis creado, ya es un dispositivo completo y correcto. Sólo falta salvar este dispositivo en la librería de dispositivos LibPrac0, para poderlo usar en otros diseños. Para ello:
- Seleccionad el dispositivo en la hoja de circuito haciendo un clic sobre la caja.
 - Seleccionad Schematic en menú superior de la hoja de diseño de circuitos. Se despliega un menú sobre el que debéis hacer clic en Save Part to Library... Ahora se abre una ventana de dialogo como la que se muestra a continuación. Debéis seleccionar lo que indica la figura, si es que no viene ya seleccionado por defecto y hacer clic en el botón Save.



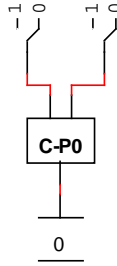
- Ahora ya podéis borrar el dispositivo C-P0 de la hoja de circuitos y cerrar la hoja, pues el proceso ha terminado. En la librería LibPrac0 ya se encuentra el dispositivo C-P0 con el circuito correcto asociado.

0.3.5 Comprobación del circuito encapsulado

Ahora debéis comprobar que el dispositivo C-P0 tiene el comportamiento esperado. Para ello, abrid un nuevo circuito y añadid el dispositivo a la hoja de edición de circuitos como si de una puerta básica se tratara.

Si hacéis doble clic sobre el dispositivo C-P0 que habéis colocado en la hoja de edición de circuitos se os abrirá una nueva ventana donde veréis su implementación interna, la que tiene asociada, que es la que vosotros habéis construido en esta práctica. Para volver a la ventana anterior utilizad Ctrl+Shift+U. o haced clic en el botón  de cerrar ventana (ojo no hacedlo en la ventana que contiene toda la aplicación).

Añadid los Binary Switch y Binary Probe necesarios y simulad su comportamiento para cada una de los posibles valores de las entradas, como ya sabemos hacer de secciones anteriores. Comprobad que el circuito se comporta como dice la tabla de verdad del C-P0. Si es así es que el proceso de encapsular vuestro circuito en el dispositivo de la librería LibPrac0 ha sido correcto. El circuito de prueba resultante para estas pruebas tiene que ser algo parecido a este.



□ **Informe final**

Pregunta 2

Cuando estéis seguros del correcto funcionamiento del dispositivo C-P0, avisad al profesor de laboratorio y pedidle que revise vuestro trabajo.

El proceso que acabáis de hacer, crear un circuito y asociarle un símbolo con su misma funcionalidad, de forma que luego se pueda usar el símbolo como dispositivo en diseños más complejos, es la base del diseño modular multinivel que seguiremos para construir el computador completo. ¡Enhorabuena, por todo lo que habéis aprendido hasta ahora, que no es poco! ☺

Informe final Práctica-0

Apellidos y nombre:Grupo.....

Apellidos y nombre:Grupo.....

(Por orden alfabético)

Pregunta 1 (2 puntos)

Pregunta 2 (8 puntos)

Comentario del profesor:

Firma del profesor: