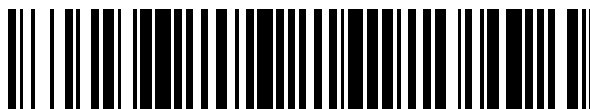


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 691 028**

51 Int. Cl.:

G06F 3/147 (2006.01)

G06F 1/16 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **19.10.2007 PCT/US2007/082001**

87 Fecha y número de publicación internacional: **22.05.2008 WO08060819**

96 Fecha de presentación y número de la solicitud europea: **19.10.2007 E 07868521 (1)**

97 Fecha y número de publicación de la concesión europea: **18.07.2018 EP 2080189**

54 Título: **Aparato y método para permitir que módulos de pantalla comuniquen información acerca de sí mismos a otros módulos de pantalla en el mismo panel de pantalla**

30 Prioridad:

09.11.2006 US 558014

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

23.11.2018

73 Titular/es:

**D3 LED, LLC (100.0%)
380 Mountain Road, Suite 1412
Union City, NJ 07087, US**

72 Inventor/es:

**KAY, JASON E.;
ROBERTUS, BRYAN L.;
SWIMLEY, BRETT D.;
DAUGHENBAUGH, JASON A.;
KOON, JON D. y
LUKES, ANTHONY J.**

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 691 028 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Aparato y método para permitir que módulos de pantalla comuniquen información acerca de sí mismos a otros módulos de pantalla en el mismo panel de pantalla

Antecedentes de la invención

5 1. Campo de la invención

La presente invención está relacionada generalmente con el campo de paneles de pantallas, y más específicamente, con un aparato y un método para permitir que módulos de pantalla comuniquen información acerca de sí mismos a otros módulos de pantalla en el mismo panel de pantalla.

2. Descripción de la técnica relacionada

10 La mayoría de carteles electrónicos o paneles de pantallas se basan hoy en una construcción modular, esto es, comprenden una pluralidad de módulos de pantalla. La led es la tecnología dominante usada en paneles de pantallas de exteriores a pleno color; sin embargo, hay otros tipos de tecnologías de pantalla — tales como plasma, LCD (pantalla de cristal líquido) y CRT (tubo de rayos catódicos)— que también se basan en un diseño modular. El desafío con cualquiera de estos paneles de pantallas modulares es cerciorarse de dónde está cada módulo en relación a los
15 otros módulos de modo que el controlador pueda enviar los datos de vídeo apropiados al módulo correcto.

Los métodos existentes para tratar este problema incluyen el uso de: (1) cableado específico; (2) cableado en cadena en margarita; (3) conmutador de dirección de módulo; (4) programación de dirección de módulo; y (5) un archivo de configuración de software. Cada uno de estos métodos, y sus desventajas, se tratan a continuación.

20 El problema con cableado específico es que cada módulo debe tener su propio cable especial. Por lo tanto supone una carga sobre el instalador para que conecte cada módulo a su cable específico. En grandes paneles de pantallas con cientos de cables, los errores de cableado son comunes y difíciles de encontrar y corregir.

Un ejemplo de un sistema conocido pantalla de multi-panel de pantalla se describe en la solicitud de patente de EE. UU. publicada US2005/0093768 [Devos. J, et al. 5 de mayo de 2005], en la que cada uno de los módulos de pantalla tiene conectores de canto complementarios que proporcionan tanto una conexión física entre módulos
25 adyacentes como una conexión eléctrica, a través de la que se distribuye alimentación y señales de vídeo entre los módulos de pantalla según la configuración aportada por el usuario de los módulos de pantalla individuales relativamente entre sí.

Con el cableado en cadena en margarita, el controlador se conecta al primer módulo, que se conecta al segundo módulo, que se conecta al tercer módulo, etc. El cableado en cadena en margarita no es muy flexible y requiere
30 distribuciones específicas de armarios. A menudo los diseños de armario o configuraciones de panel de pantalla no son óptimos para el número de módulos en una cadena, lo que requiere controladores adicionales. Otro problema con el cableado en cadena en margarita es que si falla un módulo, todos los otros módulos conectados a él aguas abajo también perderán datos, provocando que una gran parte del panel de pantalla falle cuando únicamente ha fallado un módulo. Además, los módulos en cadena en margarita crean problemas de integridad de señal eléctrica dado que se
35 debe pasar una gran cantidad de datos a través de cada módulo para llegar al siguiente.

Otro método para resolver el problema de determinar dónde está ubicado cada módulo de pantalla respecto a otros módulos de pantalla es asociar un conmutador (preferiblemente un DIP (encapsulado dual en línea) o conmutador rotatorio) con cada módulo. Un técnico establece cada conmutador de manera única para cada módulo. Los ajustes
40 de conmutador se deben establecer correctamente a fin de identificar la posición del módulo en el panel de pantalla, y el establecimiento típicamente constituye una asignación de fila y columna. El problema con este método es que la carga para establecer correctamente los conmutadores recae sobre el instalador. Con paneles de pantallas grandes que consisten en cientos de módulos, este puede ser un método tedioso y propenso a error para configurar el panel de pantalla. Además, lleva mucho tiempo.

Incluso otro método para resolver el problema de mapear módulos de pantalla en un panel de pantalla es usar
45 programación de dirección de módulo. La programación de dirección de módulo implica programar cada módulo con un dispositivo externo o por medio de un panel de control ubicado físicamente en el propio módulo. Este método es obviamente intensivo en mano de obra y propenso a error humano.

Finalmente, se puede usar un archivo de configuración de software para mapear la ubicación de módulos de pantalla sobre un panel de pantalla. Este método asigna a cada módulo su propia dirección única, que usualmente se programa
50 en la fábrica. Un técnico introduce entonces estas direcciones en un archivo de configuración software, y un controlador principal lee el archivo de configuración de software para generar una tabla de búsqueda para determinar la ubicación de coordenadas de cada módulo.

Aunque la creación manual del archivo de configuración de software de controlador funciona, un único técnico puede tardar de dos a tres horas en rellenar totalmente el archivo de configuración con coordenadas y las correspondientes

5 direcciones de módulo de pantalla antes de poder probar e instalar el panel de pantalla. Conforme aumentan las dimensiones globales del panel de pantalla, el tamaño del archivo de configuración también debe aumentar; algunos paneles de pantallas más grandes requieren más de entradas 2000 separadas en el archivo de configuración. Debido al simple número de entradas y la potencial complejidad del esquema de direcciones de módulos de pantalla, la introducción manual de direcciones y coordenadas de módulos de pantalla es propensa a error. Si se traslada o teclea incorrectamente un único carácter en un módulo dirección o coordenada "x"-y, el panel de pantalla expondrán incorrectamente información de imagen. La depuración de estos errores es difícil y lleva tiempo, requiriendo a menudo que un técnico repita el mismo proceso que ha usado para crear el archivo de configuración en primer lugar.

10 Si el proceso de crear el archivo de configuración es engorroso, el mantenimiento del archivo de configuración después de haber instalado el panel de pantalla se puede complicar igualmente. Cuando falla un módulo de pantalla de ledes, un electricista u otro técnico de carteles debe sustituirlo manualmente. Como cada módulo de pantalla tiene su propia dirección única, el técnico debe actualizar el archivo de configuración para hacer coincidir coordenadas de panel de pantalla existente con la dirección del nuevo módulo de pantalla. Esto significa que el técnico debe registrar manualmente la dirección de los tableros de circuitos antiguos y nuevos mientras está en el campo, y luego obtener acceso al archivo de configuración del software de controlador para sustituir la dirección del módulo de pantalla antiguo por la dirección del nuevo. Este proceso manual es susceptible de registrar o introducir error y es complicado por el hecho de que la sustitución individual del módulo de pantalla en el campo puede no tener la experiencia informática o el acceso físico al software de controlador para actualizar el archivo de configuración con éxito.

20 Lo que se necesita es un método automatizado para rellenar el archivo de configuración del software de controlador del panel de pantalla con los identificadores y coordenadas x-y asociadas de cada módulo de pantalla en el panel de pantalla. Una solución de este tipo eliminaría las etapas manuales requeridas para crear el archivo de configuración de software de controlador, tanto cuando se rellena antes de la instalación del panel de pantalla como cuando se actualiza durante el mantenimiento tras el despliegue.

Breve compendio de la invención

25 Aspectos de la invención se presentan en las reivindicaciones independientes adjuntas. Rasgos preferidos u opcionales de la invención se presentan en las reivindicaciones dependientes adjuntas.

30 En un ejemplo preferido de la invención, los transmisores, receptores y/o transeceptores comunican información usando circuitos de resonador LC para generar campos magnéticos alternos entre módulos de pantalla adyacentes. Preferiblemente, la información que es transmitida y recibida entre módulos de pantalla adyacentes es información de identificador. La información de identificador desde los módulos de pantalla se transmite preferiblemente a un programa de software de mapeo de cartel que asigna la ubicación de los módulos de pantalla sobre el panel de pantalla. El programa de software de mapeo de cartel descubre preferiblemente todas las caras de cartel separadas y asigna coordenadas x-y a todos módulos de pantalla en cada cara de cartel.

35 En un ejemplo preferido de la invención, las coordenadas x-y que son asignados a cada módulo de pantalla por el programa de software de mapeo de cartel son convertidas a coordenadas x-y reales, y las coordenadas x-y reales se convierten a coordenadas de vídeo.

40 En un ejemplo preferido de la invención, el método comprende además usar circuitos de resonador LC para comunicar información entre los transmisores, receptores y/o transeceptores al generar campos magnéticos alternos entre módulos de pantalla adyacentes. Preferiblemente, la información que es transmitida y recibida entre módulos de pantalla adyacentes es información de identificador. La información de identificador desde los módulos de pantalla se transmite preferiblemente a un programa de software de mapeo de cartel que asigna la ubicación de los módulos de pantalla sobre el panel de pantalla. El programa de software de mapeo de cartel descubre preferiblemente todas las caras de cartel separadas y asigna coordenadas x-y a todos módulos de pantalla en cada cara de cartel.

45 En un ejemplo preferido de la invención, las coordenadas x-y que son asignados a cada módulo de pantalla por el programa de software de mapeo de cartel son convertidas a coordenadas x-y reales, y las coordenadas x-y reales se convierten a coordenadas de vídeo.

Breve descripción de los dibujos

La figura 1 es un diagrama de una realización de la presente invención en la que un módulo de pantalla tiene cuatro bordes y está equipado con dos transmisores y dos receptores.

50 La figura 2 es un diagrama del circuito de transmisor de la presente invención.

La figura 3 es un diagrama del circuito de receptor de la presente invención.

La figura 4 es un diagrama de un panel de pantalla con dos caras.

La figura 5 es un diagrama de flujo de un aspecto de la funcionalidad de software de la presente invención.

La figura 6 es un diagrama de flujo de incluso otro aspecto de la funcionalidad de software de la presente invención.

La figura 7 es un diagrama de flujo de incluso otro aspecto de la funcionalidad de software de la presente invención.

Números de referencia

1. Transmisor superior de módulo de pantalla
2. Receptor derecho de módulo de pantalla
- 5 3. Receptor inferior de módulo de pantalla
4. Transmisor izquierdo de módulo de pantalla

Descripción detallada de la invención

La presente invención permite que los módulos de pantalla individuales de un panel de pantalla basado en módulos se comuniquen con módulos de pantalla adyacentes para compartir información. En el ejemplo preferido de la invención, cada módulo de pantalla comparte su información de identificador con módulos de pantalla adyacentes a sus bordes izquierda y superior.

En el ejemplo preferido de la invención, conforme cada módulo de pantalla recibe información de identificador de módulos vecinos ubicados a su derecha y parte inferior, usa mecanismos de comunicación conocidos para enviar esta información de identificador — más su propio identificador — al módulo de control principal del panel de pantalla, que a su vez envía los datos al software de controlador del panel de pantalla.

La presente invención incluye código de software que interpreta los datos de identificador recogidos de los módulos de pantalla individuales para calcular la posición de cada módulo de pantalla en el panel de pantalla y asociar su identificador con las coordenadas x-y apropiadas. Una vez el software calcula esta información, un técnico de cartel puede guardar la información en el archivo de configuración del software de controlador, eliminando la necesidad para recoger y actualizar esta información manualmente.

La presente invención comprende un componente de hardware y un componente de software, ambos se describen más completamente a continuación.

A. Hardware

Para permitir que los módulos de pantalla comuniquen información entre sí, cada módulo de pantalla se equipa preferiblemente con dos transmisores y dos receptores. En un ejemplo alternativo de la invención, el módulo de pantalla se equipa con uno o más transceptores en lugar de uno o más de los transmisores y/o receptores. A modo de ilustración, la siguiente descripción se referirá a transmisores y receptores; sin embargo, se podría usar un transceptor en lugar de cualquiera o de todos los transmisores y receptores.

Como se muestra en la figura 1, en el borde superior del módulo de pantalla (1) se ubica un transmisor, mientras que el segundo transmisor se ubica en el borde izquierdo (4) del módulo de pantalla. El primer receptor se ubica en el borde derecho (2) del módulo de pantalla, mientras que el segundo receptor se ubica en el borde inferior (3) del módulo de pantalla. Con esta configuración, los transmisores del módulo de pantalla están adyacentes a los receptores del módulo de pantalla próximos a él. Esto permite a un módulo de pantalla comunicar información acerca de sí mismo al módulo de pantalla por encima de él y a su izquierda; aprende acerca de módulos de pantalla ubicados a su derecha y por debajo de él.

En un ejemplo preferido de la invención de la presente invención, los transmisores y receptores comunican información usando circuitos de resonador LC (inductor/condensador) para generar campos magnéticos alternos entre módulos de pantalla adyacentes. Este planteamiento es ventajoso debido a su bajo coste, su baja Interferencia Electromagnética (EMI), y el hecho de que no requiere cambiar al proceso de instalación de módulo de pantalla. Otros ejemplos preferidos de la invención para permitir comunicación entre módulos de pantalla podrían incluir conexiones cableadas o mecanismos de comunicación inalámbrica a corta distancia tales como Infrarrojos (IR), Bluetooth, o Identificación por Radiofrecuencia (RFID).

En un ejemplo preferido de la invención, cada módulo de pantalla transmite un paquete de identificación a sus vecinos cada diez segundos usando Modulación digital de amplitud (OOK On-Off Keying). Lógica de placa de circuitos convierte información a transmitir en este paquete de información en pulsos electromagnéticos que se envían por medio del circuito de resonador LC al módulo de pantalla que recibe. La lógica en el módulo de pantalla que recibe convierte las señales recibidas a un formato binario que se puede usar para "leer" la información enviada por la conexión. En el ejemplo preferido de la invención, esta información comunicada es el identificador del módulo de pantalla que transmite. En otros ejemplos preferidos de la invención, el mecanismo de comunicación de la presente invención se puede usar para enviar datos de diagnóstico u otra información de interés desde un módulo de pantalla a otro.

La Tabla 1 muestra el formato de los paquetes de comunicación en el presente ejemplo de la invención.

Tabla 1 - Estructura de Paquete de Comunicación

Nombre	Identificador	Bits	Descripción
CÓDIGO_INICIO	0x00-0x01	15-0	0xA5C3: usado para verificar inicio de paquete.
MAC	0x02-0x07	47-0	Identificador de módulo
MAC_XSUM	0x08-09	15-0	suma de verificación de 16 bits sobre el campo de identificador de módulo. Algoritmo de suma de verificación IP — RFC 791/768

El ejemplo preferido de la invención usa comunicación en serie entre los módulos de pantalla con una tasa de baudios de 1000 bps y un bit de inicio, 8 bits de datos y un bit de parada.

Los circuitos LC tienen una frecuencia de resonancia establecida por la inductancia del inductor y la capacitancia del condensador. La figura 2 muestra el circuito de transmisor de la presente invención. El circuito de transmisor impulsa ambos extremos del circuito LC con niveles opuestos de lógica CMOS de 3,3 V para generar eficazmente una onda cuadrada de 6,6 V impulsada en la frecuencia resonante. El circuito puede almacenar esta energía porque la onda es impulsada a la frecuencia resonante. La amplitud aumenta hasta que las pérdidas de energía igualan a la energía añadida, dando como resultado gran intensidad de señal de transmisión.

La figura 3 muestra el circuito de receptor de la presente invención. El circuito de receptor incluye un circuito LC que absorbe energía del campo magnético alterno generado por el transmisor. Un lado del resonador se predispone a aproximadamente 1,6 V, y un conjunto de diodos Schottky limita la amplitud para que no se haga muy grande. Un comparador compara la tensión de resonador con una tensión umbral de aproximadamente 82 mV. Si no hay presente nada o hay muy poca señal (menos de la tensión umbral), el comparador no se activará y la salida permanece baja. Si la señal es de suficiente amplitud para activar el comparador, el comparador tendrá como salida una onda cuadrada a la frecuencia de resonancia (portadora). La lógica de placa de circuitos procesa la salida del circuito de receptor para detectar la presencia de la onda cuadrada y convertirla en formato binario. Una onda cuadrada activa indica un valor binario de "0," mientras que una señal pasiva indica un valor binario de "1." Usando este método, el módulo de pantalla que recibe convierte las ondas cuadradas enviadas por el módulo de pantalla que transmite en datos binarios que contienen información: en este caso, el identificador del módulo de pantalla que transmite.

Tras usar el mecanismo de comunicación de la presente invención para aprender los identificadores de los módulos de pantalla ubicados en sus bordes derecho e inferior, cada módulo de pantalla almacena los identificadores de sus vecinos en memoria. Cuando es interrogado, cada módulo de pantalla envía esta información de identificador almacenada — junto con su propio identificador — al controlador principal del panel de pantalla. El controlador principal puede pasar entonces esta información recogida al algoritmo de software que mapea la ubicación de los módulos de pantalla en el panel de pantalla. Como alternativa, cada módulo podría enviar inmediatamente información de identificador al controlador sin almacenar en memoria, ya sea autónomamente o en respuesta a una interrogación. Otra alternativa sería que cada módulo difundiera la información de identificador a todos los otros módulos en la pantalla. Cada módulo podría entonces determinar su propia posición en el panel de pantalla usando la información recibida sin intervención del controlador.

La explicación anterior asume que un módulo de pantalla tiene cuatro bordes (superior, derecho, izquierdo e inferior); sin embargo, la presente invención también trabaja con módulos de pantalla que tienen tres bordes. En caso de un módulo de pantalla con tres bordes, se ubicaría un transceptor en cada borde del módulo de pantalla que está adyacente a otro módulo de pantalla. La presente invención trabajaría como se ha descrito anteriormente en todos los demás aspectos.

B. Software

El software de mapeo de cartel de la presente invención acepta información de identificador recogida y la procesa para determinar las coordenadas x-y de cada módulo de pantalla en una o más 'caras' de cartel. Un panel de pantalla puede tener múltiples monitores, o caras, como se muestra en la figura 4. Una cara es un grupo de módulos de pantalla que están físicamente contiguos; las caras no están físicamente contiguas entre sí y pueden o no exponer los mismos datos de vídeo que otras caras en el panel de pantalla.

La presente invención trabajará con cualquier programa de software de mapeo de cartel que pueda tomar la información de los receptores o transceptores ubicados en los módulos de pantalla y usar esa información para mapear la ubicación de los módulos de pantalla sobre el panel de pantalla. Así, la presente invención no se limita a ningún tipo particular de programa de software de mapeo de cartel, siempre que cumpla esta función. La explicación siguiente pretende describir una posible versión del programa de software de mapeo de cartel.

En un ejemplo preferido de la invención del software, los módulos de pantalla reportan su información de identificador nuevamente al software. Esta información de identificación incluye la propia ID de módulo ("yo"), así como información de ID para un módulo a la derecha del módulo (si se detecta) e información de ID para un módulo por debajo del módulo (si se detecta). Entonces se construye una lista de esta información de módulo, y todos los elementos en la lista ("elementos de lista de módulo") se inicializan para indicar que no se ha asignado información de coordenadas al

módulo asociado con el elemento de lista (p. ej., coordenadas Válido = falso). El elemento de lista de módulos incluye información de coordenadas (x e y), información de ID de cara, e información de validez de coordenadas (que indica si se han asignado coordenadas a ese módulo).

El procesamiento comienza inicializando el primer identificador de cara a cero (es decir, asignando un índice o identificador de cero a la primera cara). Preferiblemente a subsiguientes caras se asignan identificadores de cara en incrementos de uno (1, 2, 3, etc.). El software empieza entonces al comienzo de la lista y examina el primer elemento de lista de módulos para determinar si se han asignado coordenadas a ese módulo. Si se encuentra un elemento de lista de módulos con coordenadas inválidas (es decir, no hay coordenadas asignadas), entonces comienza el procesamiento de coordenadas de módulo para ese módulo. Si un elemento de lista de módulos tiene coordenadas válidas (es decir, se han asignado coordenadas al módulo), entonces se examina el siguiente elemento de lista de módulos en la lista por turnos hasta que se encuentra el final de la lista.

Una vez se han examinado todos los elementos de lista de módulos contenidos dentro de la lista para determinar validez de coordenadas, se pueden "corregir" las coordenadas y traducir desde coordenadas de módulo a coordenadas de vídeo. Específicamente, las coordenadas de módulo se normalizan de manera que el módulo superior izquierdo tiene coordenadas (0,0), y las coordenadas de módulo se convierten en coordenadas de vídeo (píxel), como se explica más completamente a continuación.

Si el software encuentra un módulo con coordenadas inválidas (es decir, no asignadas), el algoritmo de software asume que este es el inicio de una nueva cara e inicializa los valores iniciales de coordenadas de cara a (0,0). Estas coordenadas iniciales se asignan al primer módulo que comprende la cara (que es el primer módulo a examinar y no necesariamente el módulo que en última instancia finaliza con coordenadas (0,0) tras la corrección).

El algoritmo intenta entonces ver, a su vez, si hay un módulo a la derecha, debajo, a la izquierda, o encima del módulo de pantalla actual que no tiene coordenadas asignadas. La primera de estas situaciones que se encuentra (por ejemplo, si hay un módulo a la derecha del módulo de pantalla actual) provoca que el procesamiento de módulos cambie al módulo recientemente descubierto, con un cambio en las coordenadas del módulo basado en la posición del nuevo módulo. En otras palabras, un módulo a la derecha tiene la coordenada "x" aumentada en uno, por debajo tiene "y" aumentada en uno, a la izquierda tiene "x" disminuida en uno, y encima tiene "y" disminuida en uno.

Antes de que el procesamiento cambie a este nuevo módulo, sin embargo, el algoritmo de software guarda el estado de la siguiente ruta de procesamiento que se puede realizar en el elemento actual de lista de módulos a una lista LIFO (último en entrar primero en salir) de registros de información de "estado" de módulo (también llamado "pila"). Esta información de estado permite al algoritmo volver y seguir rutas de procesamiento alternativas de configuración de módulo en una cara, en el caso de que una ruta de procesamiento llegue a un extremo muerto. Por ejemplo, si se encuentra un módulo a la derecha de un módulo de pantalla, el algoritmo "empujaría" el estado del procesamiento para indicar que el siguiente procesamiento a realizar sea por debajo del módulo de pantalla actual. Se guardan las coordenadas de módulo actuales (x e y), así como referencia al elemento actual de lista de módulos de pantalla.

Una vez se ha agotado una ruta de procesamiento (p. ej., no hay módulo a la derecha, debajo, a la izquierda, o encima del módulo de pantalla actual), se examina la pila de procesamiento para la cara. Si la pila no está vacía, la última entrada empujada (o añadida a la pila) es "sacada" fuera (se retira de) la pila, esa información de registro se usa para restituir la información de estado de procesamiento para el módulo referenciado por el registro de pila, y comienza una nueva ruta de procesamiento de módulos. En otras palabras, la información de estado es usada por el software para captar donde cesó en el procesamiento de módulos. Por ejemplo, si se estaba procesando un módulo, y se encontró un módulo a la derecha, la ruta de procesamiento seguiría el módulo a la derecha. El software tiene que recordar, sin embargo, que todavía hay disponibles otras rutas de procesamiento (encima, debajo y a la izquierda del módulo actual) para procesamiento. En última instancia, si no se dejan elementos de lista de módulos en la pila, entonces el algoritmo asume que todo el procesamiento que se puede realizar en esa cara está completado, y posteriormente vuelve a examinar la lista de módulos completa para ver si subsiguientes caras necesitan ser procesadas.

Cuando se han procesado todas las caras en el panel de pantalla, entonces el software realiza una función de corrección de coordenadas. Cada módulo tiene "coordenadas de módulo," que son relativas al primer módulo en la cara que se encontró (que tiene asignadas coordenadas de módulo 0,0 por defecto). Obsérvese que las coordenadas de un módulo pueden ser negativas en ambos ejes de coordenadas.

El software "normaliza" entonces las coordenadas para una cara, de manera que el módulo con las coordenadas más negativas (preferiblemente el módulo en la esquina superior izquierda de la cara) se normaliza a coordenadas (0,0), y a otros módulos que comprende la cara se les reasignan coordenadas de módulo relativas a este módulo particular. De esta manera, se eliminan todas las coordenadas negativas.

Tras conseguir esto, las coordenadas de módulo se convierten a coordenadas de vídeo multiplicando la coordenada "x" de cada módulo por la anchura del módulo (en píxeles de vídeo) y multiplicando la coordenada "y" de cada módulo por la altura del módulo (en píxeles de vídeo).

Este proceso se explica más completamente a continuación en conexión con las figuras 5-7. Haciendo referencia a la figura 5, al ejecutar el algoritmo RealizarConfiguraciónAutomáticaInterna (PerformInternalAutoConfiguration), se

asume que se ha creado la lista de la información de elementos de módulo, y que la información de asignación de coordenadas de módulo se ha inicializado a "no asignado". Empezando en la parte superior de la figura 5, el software inicializa la ID de cara a cero. La primera caja de decisión evalúa si se han añadido elementos de módulo a la lista. Si no, no se tiene que realizar procesamiento adicional. De otro modo, el elemento actual de lista de módulos se establece al primer elemento de la lista de módulos.

El elemento se examina para ver si se han asignado coordenadas. Si es así, el algoritmo se mueve al siguiente elemento en la lista de elementos de módulo y comprueba este elemento para ver si se han asignado coordenadas. Esta etapa se repite a su vez hasta que se encuentra un módulo que tiene coordenadas no asignadas, o hasta que se llega al final de la lista. Cuando se llega al final de la lista, se han identificado todas las caras, y se han asignado todas las coordenadas relativas de los módulos que comprenden la cara. Se llama al algoritmo CorregirCoordenadas (CorrectCoordinates) para que normalice coordenadas de módulo y convierta las coordenadas de módulo en coordenadas de píxeles de vídeo. De otro modo, si se encuentra un elemento con coordenadas no asignadas, se asume que este módulo es un miembro de una nueva cara.

Para la nueva cara, se inicializan a cero variables de coordenadas "x" e "y" para proporcionar una referencia para la asignación de coordenada inicial. La lista de pila de información de estado "pila_módulo" se inicializa a vacía (p. ej., no se ha añadido información de estado a la lista de pila). La variable HECHO se inicializa entonces a FALSO, indicando que no se han completado todas las rutas de procesamiento para esta cara. Las coordenadas "x" e "y" actuales se asignan entonces a este elemento de lista de módulos.

Entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene el mismo identificador de módulo "yo" que el identificador "derecho" de módulo actual. Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la información de estado concerniente al módulo actual se guarda empujando información de estado que comprende la referencia de elemento de lista de módulos, los valores actuales de coordenadas "x" e "y" y la siguiente ruta de procesamiento que tiene que ser evaluada para este módulo (en este caso el siguiente módulo DEBAJO del módulo actual). Entonces se incrementa en uno la coordenada "x", indicando que el siguiente módulo a evaluar está a la derecha del módulo actual. El elemento actual de lista de módulos es sustituido entonces por una referencia al nuevo módulo de pantalla que se ha encontrado, y el algoritmo salta atrás al punto donde se asignan coordenadas "x" e "y" al elemento actual de lista de módulos (denominador conector 1 dentro de la figura 5).

Si no se ha encontrado módulo de pantalla a la derecha del módulo de pantalla actual, o si se encuentra un módulo de pantalla pero sus coordenadas ya se habían asignado, el algoritmo se mueve al denominador conector 2, en la parte superior de la figura 6.

Comenzando en el denominador conector 2 en la parte superior de la figura 6, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene el mismo identificador de módulo "yo" que el identificador "debajo" del módulo actual. Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la información de estado concerniente al módulo actual se guarda empujando información de estado que comprende la referencia de elemento de lista de módulos, los valores actuales de coordenadas "x" e "y" y la siguiente ruta de procesamiento que tiene que ser evaluada para este módulo (en este caso el siguiente módulo A LA IZQUIERDA del módulo actual). Entonces se incrementa en uno la coordenada "y", indicando que el siguiente módulo a evaluar está debajo del módulo actual. El elemento actual de lista de módulos es sustituido entonces por una referencia al nuevo módulo de pantalla que se ha encontrado, y el algoritmo salta atrás al punto donde se asignan coordenadas "x" e "y" al elemento actual de lista de módulos (denominador conector 1 dentro de la figura 5).

Si no se encuentra módulo de pantalla debajo del módulo de pantalla actual, o si se encuentra un módulo de pantalla pero sus coordenadas ya han sido asignadas, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene un identificador de módulo "derecho" que coincide con el identificador "yo" del módulo actual (una coincidencia indica que el módulo encontrado está a la izquierda del módulo actual). Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la información de estado concerniente al módulo actual se guarda empujando información de estado que comprende la referencia de elemento de lista de módulos, los valores actuales de coordenadas "x" e "y" y la siguiente ruta de procesamiento que tiene que ser evaluada para este módulo (en este caso el SIGUIENTE MÓDULO SUPERIOR del módulo actual). Entonces se reduce en uno la coordenada "x", indicando que el siguiente módulo a evaluar está a la izquierda del módulo actual. El elemento actual de lista de módulos es sustituido entonces por una referencia al nuevo módulo de pantalla que se ha encontrado, y el algoritmo salta atrás al punto donde se asignan coordenadas "x" e "y" al elemento actual de lista de módulos (denominador conector 1 dentro de la figura 5).

Si no se encuentra módulo de pantalla a la izquierda del módulo de pantalla actual, o si se encuentra un módulo de pantalla pero sus coordenadas ya habían sido asignadas, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene un identificador de módulo "debajo" que coincide con el identificador "yo" del módulo actual (una coincidencia indica que el módulo encontrado está encima del módulo actual). Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, entonces se reduce en uno la coordenada "y", indicando que el siguiente módulo a evaluar está encima del módulo actual (aquí no

se tiene que guardar información de estado de ruta de procesamiento porque esta ruta de procesamiento es la última que tiene que ser evaluada para el módulo actual). El elemento actual de lista de módulos es sustituido entonces por una referencia al nuevo módulo de pantalla que se ha encontrado, y el algoritmo salta atrás al punto donde se asignan coordenadas "x" e "y" al elemento actual de lista de módulos (denominador conector 1 dentro de la figura 5).

- 5 Si no se encuentra módulo de pantalla encima del módulo de pantalla actual, o si se encuentra un módulo de pantalla pero sus coordenadas ya han sido asignadas, el control pasa a denominador conector 3 en la parte superior izquierda de la figura 7.

Comenzando en el denominador conector 3 en la parte superior izquierda de la figura 7, en este punto en el algoritmo, se ha agotado una única ruta de procesamiento. Sin embargo, se pueden haber guardado rutas de procesamiento alternativas (empujadas a la pila) para cada módulo procesado. Estas rutas de procesamiento representan rutas de procesamiento que todavía no se han seguido.

- 10 Una referencia a un "nuevoMóduloPantalla" se establece inicialmente a una referencia de INVÁLIDO, indicando que no se ha encontrado nueva ruta de procesamiento adecuada. Obsérvese que la referencia "nuevoMóduloPantalla" se establecerá a una referencia válida de módulo de pantalla en el algoritmo como se describe más adelante, si se encontrara un módulo de pantalla con una ruta de procesamiento válida.

- 15 Se examina la lista de información de estado "pila_módulo", y si está vacía, se ejecuta una comprobación para determinar si se ha encontrado una referencia "nuevoMóduloPantalla", indicando que se puede seguir una nueva ruta de procesamiento. Si es así, el elemento actual de lista de módulos se asigna a esta "nuevoMóduloPantalla," y el algoritmo salta atrás al punto donde se asignan coordenadas "x" e "y" al elemento actual de lista de módulos (denominador conector 1 dentro de la figura 5). Obsérvese que estas coordenadas "x" e "y" se basan en la información de estado recuperada de la "pila_módulo," descrita más en detalle a continuación.

- 20 Si se encuentra que la "pila_módulo" está vacía (no se dejan elementos de información de estado en la lista) y la referencia "nuevoMóduloPantalla" es INVÁLIDO, el algoritmo asume que todas las rutas de procesamiento que se podrían haber seguido para la cara actual están completas. El algoritmo incrementa en 1 entonces la ID de cara (como anticipo a identificar otra cara) y establece el marcador HECHO a VERDADERO (que indica que el procesamiento de cara actual está completo). El algoritmo salta entonces al denominador conector 1 dentro de la figura 5.

- 25 Si la "pila_módulo" no está vacía, se recupera información de estado desde el primer elemento de la pila. Como la lista de pila es una pila LIFO (último en entrar primero en salir), se recuperará la información de estado del último módulo procesado. Esta información de estado representa una ruta de procesamiento alternativa para un módulo que aún no ha sido explorado, y puede estar disponible como camino para asignación adicional de coordenadas de módulo.

- 30 Las coordenadas "x" e "y" para el módulo de pantalla se restauran de la información de estado recuperada de la pila. Estas son las coordenadas "x" e "y" que han sido asignadas al módulo de pantalla recuperado cuya información se acaba de recuperar de la pila. El elemento actual de lista de módulos se establece entonces al elemento de lista referencia contenido dentro de la información de estado recuperada de la pila. A continuación, se examina la variable guardada "estado". Esta representa la siguiente ruta de procesamiento que se debe examinar para el módulo de pantalla recuperado.

- 35 Si el estado es SIGUIENTE MÓDULO DEBAJO, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene el mismo identificador de módulo "yo" que el identificador "debajo" del módulo actual. Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la coordenada "y" se incrementa entonces en uno, indicando que el siguiente módulo a evaluar está debajo del módulo actual, la "nuevoMóduloPantalla" se establece a la referencia al nuevo módulo de pantalla que ha sido encontrado, el registro actual de información de estado es "sacado" (retirado) de la "pila_módulo," y el algoritmo salta atrás al punto donde se examina la pila_módulo para determinar si está "vacía" (como se ha descrito anteriormente).

- 40 Si el estado es SIGUIENTE MÓDULO IZQUIERDO, o los resultados del examen SIGUIENTE MÓDULO DEBAJO no encontraron un módulo de pantalla debajo del elemento actual de lista de módulos o se encontró un módulo de pantalla pero ya tenía coordenadas asignadas, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene el mismo identificador de módulo "derecho" que el identificador "yo" del módulo actual. Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la coordenada "y" se reduce entonces en uno, indicando que el siguiente módulo a evaluar está a la izquierda del módulo actual, la "nuevoMóduloPantalla" se establece a la referencia al nuevo módulo de pantalla que ha sido encontrado, el registro actual de información de estado es "sacado" (retirado) de la "pila_módulo," y el algoritmo salta atrás al punto donde se examina la pila_módulo para determinar si está "vacía" (como se ha descrito anteriormente).

- 45 Si el estado es SIGUIENTE MÓDULO SUPERIO, o los resultados del examen SIGUIENTE MÓDULO IZQUIERDO no encontraron un módulo de pantalla a la izquierda del elemento actual de lista de módulos o se encontró un módulo de pantalla pero ya tenía coordenadas asignadas, entonces se examina la lista de elementos de módulo para ver si un módulo dentro de la lista tiene el mismo identificador de módulo "debajo" que el identificador "yo" del módulo actual. Si se encuentra un módulo, se examinan estas nuevas coordenadas de un módulo para ver si han sido asignadas. Si no, la coordenada "y" se reduce entonces en uno, indicando que el siguiente módulo a evaluar está encima del módulo

actual, la "nuevoMóduloPantalla" se establece a la referencia al nuevo módulo de pantalla que ha sido encontrado, el registro actual de información de estado es "sacado" (retirado) de la "pila_módulo," y el algoritmo salta atrás al punto donde se examina la pila_módulo para determinar si está "vacía" (como se ha descrito anteriormente).

- 5 Si el estado no es ninguno de SIGUIENTE MÓDULO DEBAJO, SIGUIENTE MÓDULO IZQUIERDO, SIGUIENTE MÓDULO SUPERIOR, o los resultados del examen de SIGUIENTE MÓDULO IZQUIERDO no encontraron un módulo de pantalla a la izquierda del elemento actual de lista de módulos o se encontró un módulo de pantalla pero ya tenía coordenadas asignadas, la referencia "nuevoMóduloPantalla" se establece a INVÁLIDO, la información de estado actual es "sacada" (retirado) de la "pila_módulo," y el algoritmo salta atrás al punto donde se examina el estado de módulo para determinar si está "vacío" (como se ha descrito anteriormente).
- 10 En última instancia, si se encuentra un "nuevoMóduloPantalla", la información de procesamiento de módulos (elemento actual de lista de módulos, coordenadas "x" e "y") ha sido restaurada al punto de que se pueden asignar coordenadas al nuevo módulo, y el procesamiento puede continuar por medio de la ruta de procesamiento recientemente determinada.
- 15 Aunque se ha mostrado y descrito el ejemplo preferido de la invención de la presente invención, para los expertos en la técnica será evidente que se pueden hacer muchos cambios y modificaciones sin apartarse de la invención en sus aspectos más amplios. Las reivindicaciones adjuntas por lo tanto pretenden cubrir todos esos cambios y modificaciones que caigan dentro del alcance de la invención.

REIVINDICACIONES

1. Un aparato para permitir que módulos de pantalla comuniquen información acerca de sí mismos a otros módulos de pantalla en el mismo panel de pantalla, que comprende un panel de pantalla basado en módulos,
en donde el panel de pantalla comprende una o más caras,
5 en donde cada cara comprende una pluralidad de módulos de pantalla,
en donde cada módulo de pantalla comprende al menos tres bordes y tiene información de identificador de módulo,
en donde al menos uno de los bordes está adyacente a otro módulo de pantalla, en donde un transmisor, receptor o transceptor se ubica en cada borde que está adyacente a otro módulo de pantalla,
10 en donde los transmisores, receptores y/o transceptores se configuran de modo que cada transmisor o transceptor en un módulo de pantalla dado está adyacente a un receptor o transceptor en un módulo de pantalla adyacente,
en donde los transmisores o transceptores en un módulo de pantalla dado transmiten información acerca de ese módulo de pantalla a los receptores o transceptores adyacentes, la información transmitida que comprende información de identificador de módulo;
15 en donde los receptores o transceptores en un módulo de pantalla dado reciben la información de identificador de módulo acerca del módulo de pantalla adyacente de los transmisores o transceptores adyacentes, y
en donde la información de identificador de módulo recibida y el identificador de módulo del módulo de pantalla dado son usados por los módulos de pantalla y/o un controlador principal para determinar la ubicación del módulo de pantalla dado dentro de un panel de pantalla.
2. El aparato de la reivindicación 1, en donde los transmisores, receptores y/o transceptores comunican información
20 usando circuitos de resonador LC para generar campos magnéticos alternos entre módulos de pantalla adyacentes.
3. El aparato de la reivindicación 1 o la reivindicación 2, en donde la información de identificador desde los módulos de pantalla se transmite a un programa de software de mapeo de cartel que asigna la ubicación de los módulos de pantalla sobre el panel de pantalla.
4. El aparato de la reivindicación 3, en donde el programa de software de mapeo de cartel descubre todas las caras
25 de cartel separadas y asigna coordenadas x-y a todo módulo de pantalla en cada cara de cartel.
5. El aparato de la reivindicación 4, en donde las coordenadas x-y que son asignados a cada módulo de pantalla por el programa de software de mapeo de cartel son convertidas a coordenadas x-y reales.
6. El aparato de la reivindicación 5, en donde las coordenadas x-y reales se convierten a coordenadas de vídeo.
7. Un método para permitir que módulos de pantalla comuniquen información acerca de sí mismos a otros módulos
30 de pantalla en el mismo panel de pantalla, que comprende:
(a) en donde un panel de pantalla comprende una o más caras, en donde cada cara comprende una pluralidad de módulos de pantalla, en donde cada módulo de pantalla comprende al menos tres bordes, en donde al menos uno de los bordes en cada módulo de pantalla está adyacente a otro módulo de pantalla, localizar un transmisor, receptor o transceptor en cada borde que está adyacente a otro módulo de pantalla; y
35 (b) configurar los transmisores, receptores y/o transceptores de modo que cada transmisor o transceptor en un módulo de pantalla dado está adyacente a un receptor o transceptor en un módulo de pantalla adyacente;
en donde los transmisores o transceptores en un módulo de pantalla dado transmiten información de identificador de módulo alrededor de ese módulo de pantalla a los receptores o transceptores adyacentes,
40 en donde los receptores o transceptores en un módulo de pantalla dado reciben información acerca del módulo de pantalla adyacente desde los transmisores o transceptores adyacentes, y
en donde la información es usada por los módulos de pantalla y/o un controlador principal para determinar la ubicación del módulo de pantalla dentro de un panel de pantalla.
8. El método de la reivindicación 7, que comprende además usar circuitos de resonador LC para comunicar
45 información entre los transmisores, receptores y/o transceptores al generar campos magnéticos alternos entre módulos de pantalla adyacentes.
9. El método de la reivindicación 7, en donde la información que es transmitida y recibida entre módulos de pantalla adyacentes es información de identificador.

10. El método de la reivindicación 9, en donde la información de identificador desde los módulos de pantalla se transmite a un programa de software de mapeo de cartel que asigna la ubicación de los módulos de pantalla sobre el panel de pantalla.
- 5 11. El método de la reivindicación 10, en donde el programa de software de mapeo de cartel descubre todas las caras de cartel separadas y asigna coordenadas x-y a todo módulo de pantalla en cada cara de cartel.
12. El método de la reivindicación 11, en donde las coordenadas x-y que son asignadas a cada módulo de pantalla por el programa de software de mapeo de cartel son convertidas a coordenadas x-y reales.
13. El método de la reivindicación 12, en donde las coordenadas x-y reales se convierten a coordenadas de vídeo.

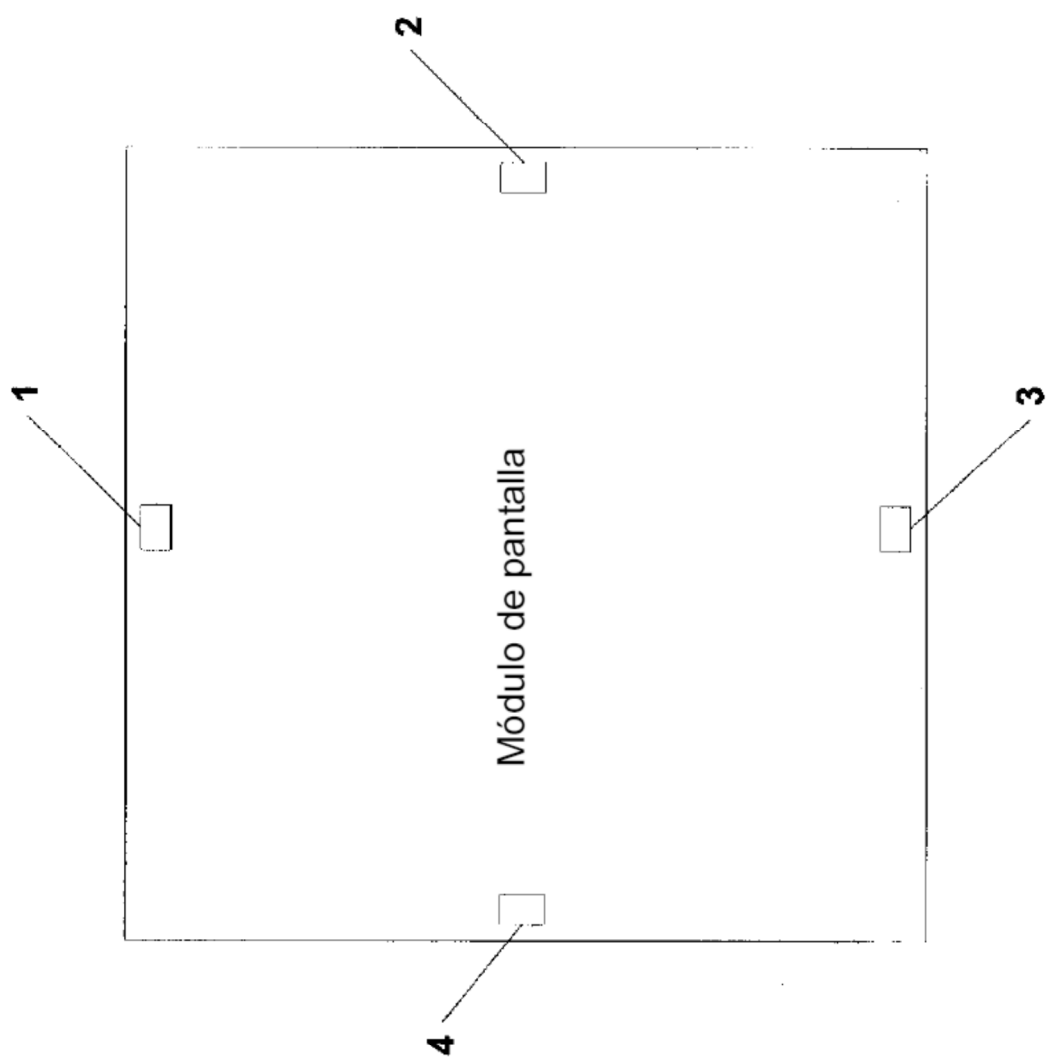


Figura 1 - Ubicaciones de Transmisor y Receptor en el módulo de

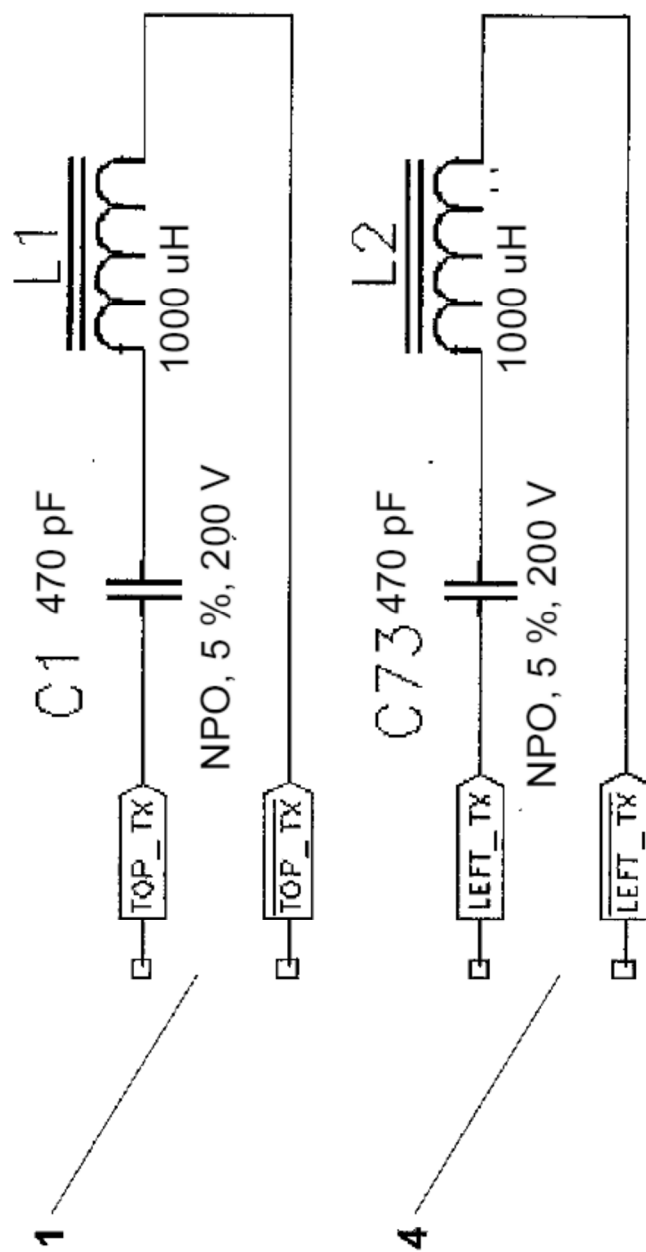


Figura 2 - Diagrama de Circuito de Trasmisor

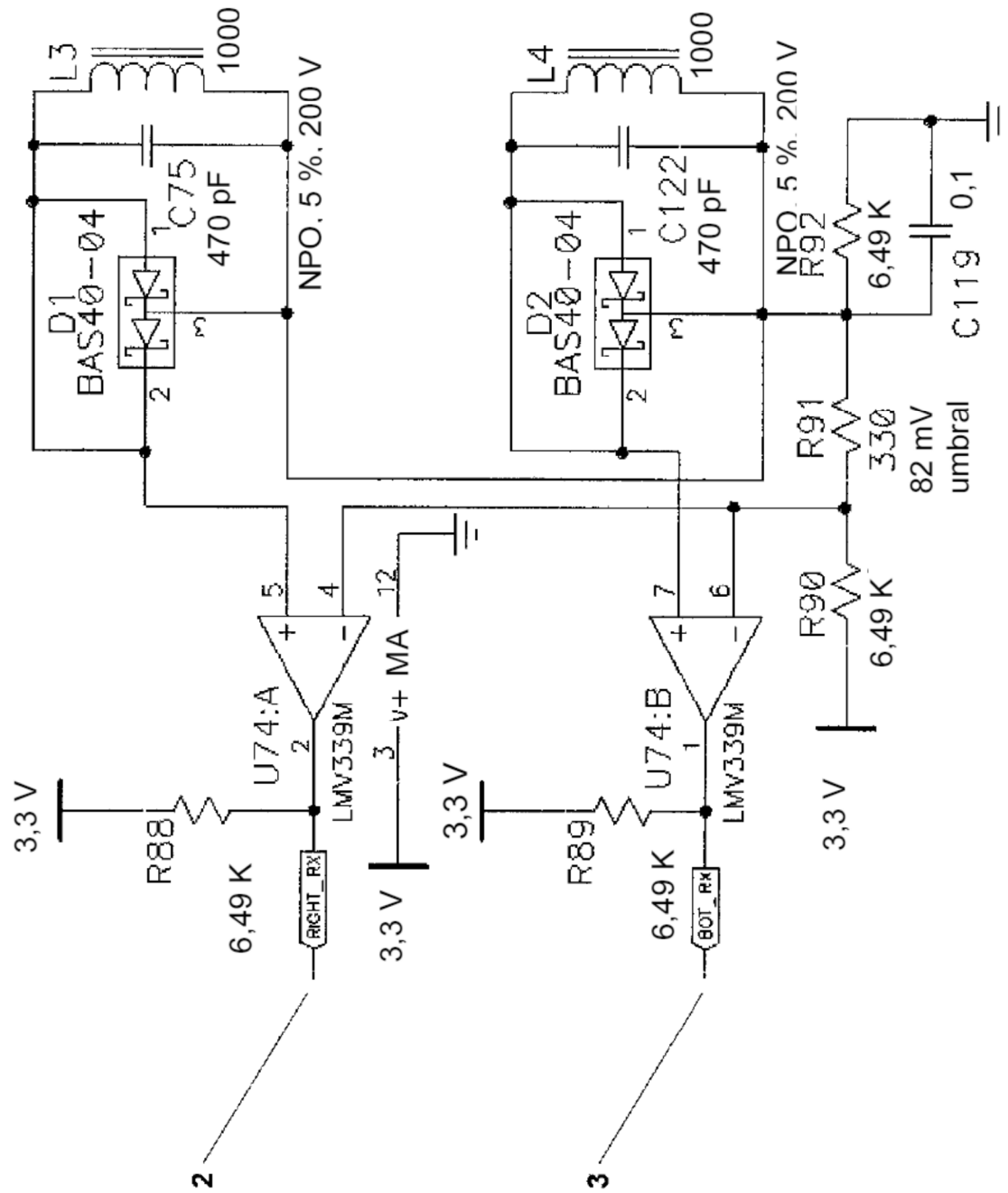


Figura 3 - Diagrama de Circuito de Receptor

Cara 1			Cara 2		
A006	A007	A008	B009	B001	B002
A009	A001	A002	B003	B004	B005
A003	A004	A005	B006	B007	B008

Figura 4 - Caras de cartel

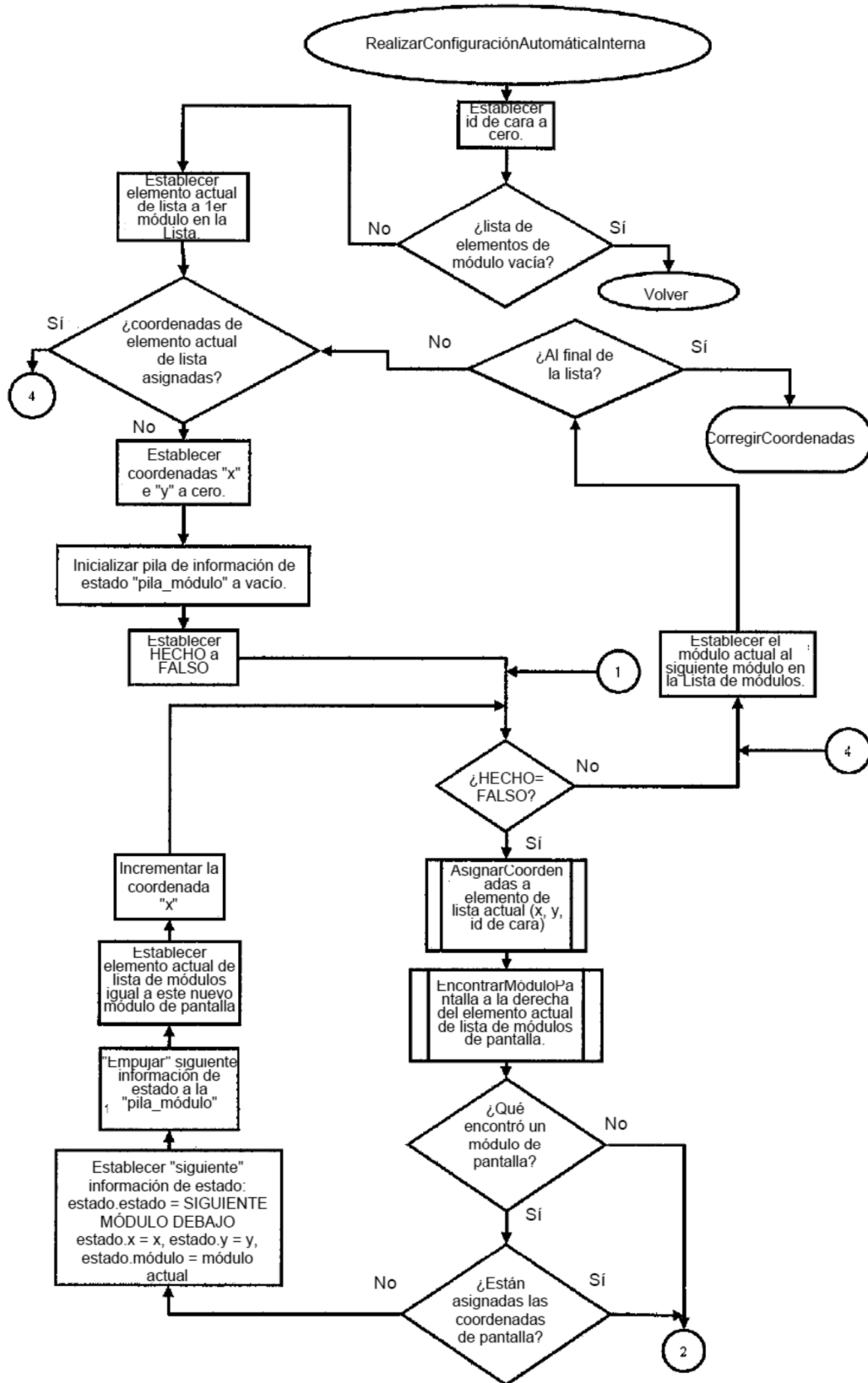


Figura 5

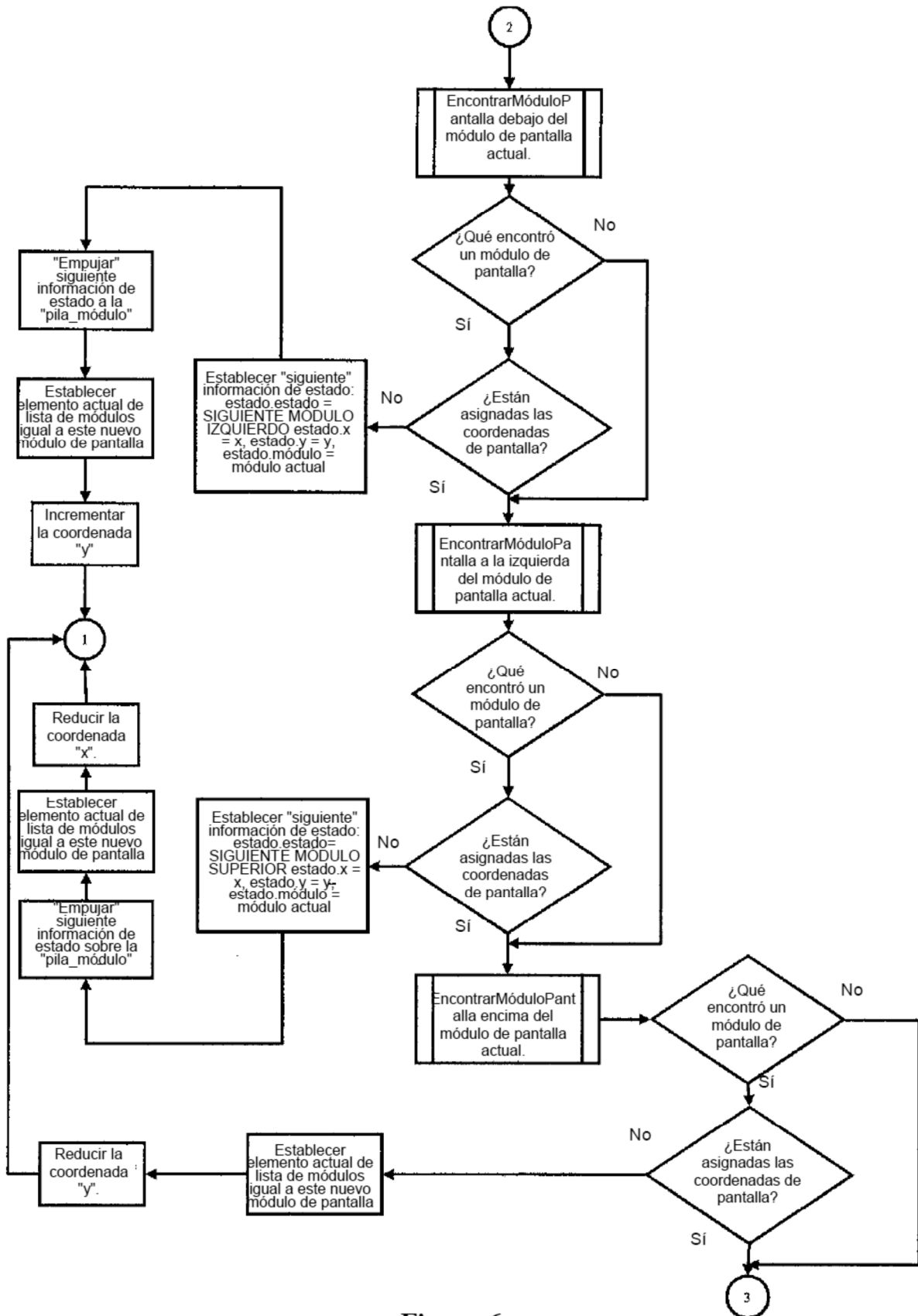


Figura 6

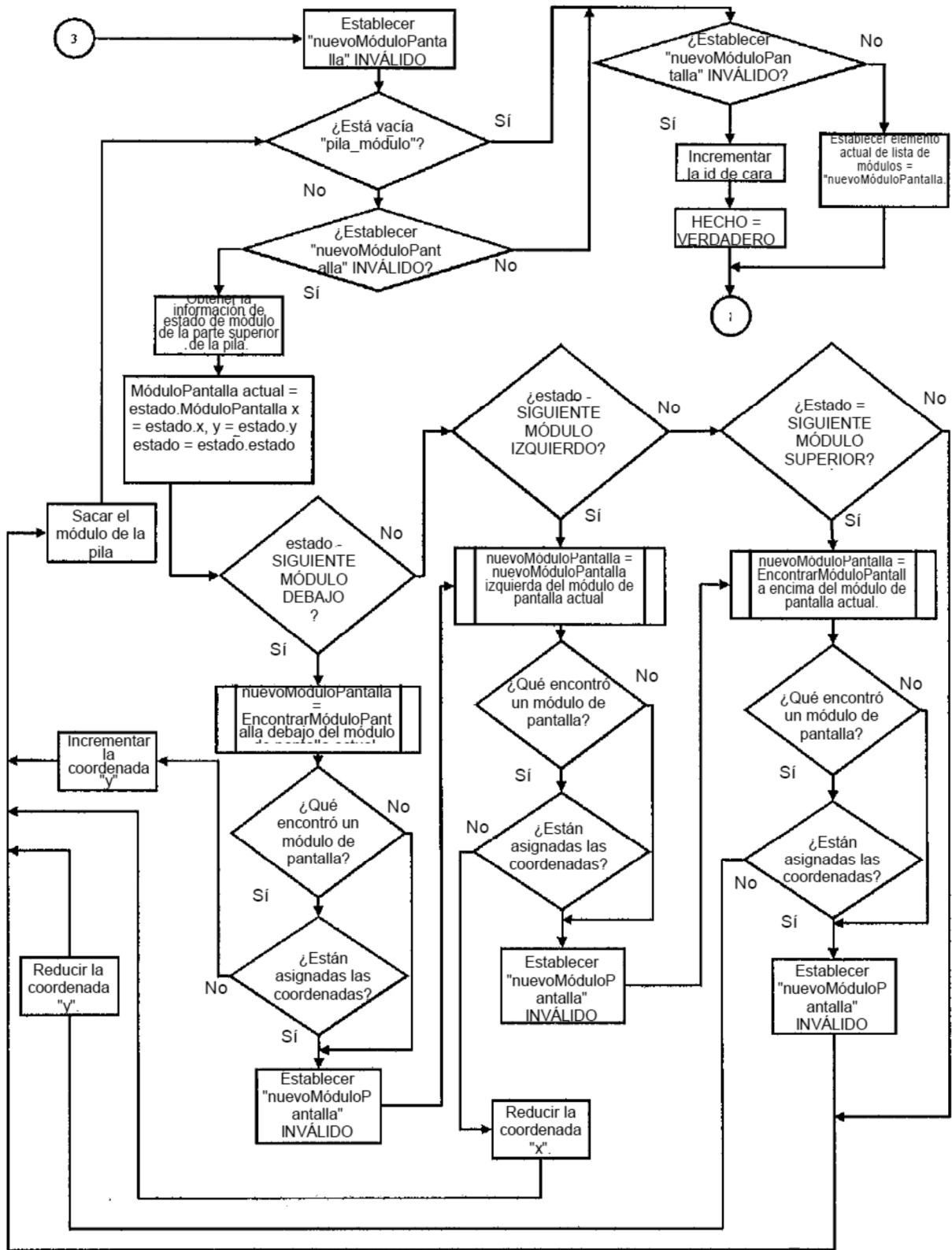


Figura 7