



19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 333 697**

51 Int. Cl.:  
**G05B 19/418** (2006.01)  
**G05B 19/042** (2006.01)  
**G01N 35/00** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **01830237 .2**  
96 Fecha de presentación : **05.04.2001**  
97 Número de publicación de la solicitud: **1248170**  
97 Fecha de publicación de la solicitud: **09.10.2002**

54 Título: **Método para la gestión de sistemas de célula de trabajo basado en un sistema de gestión de la automatización.**

45 Fecha de publicación de la mención BOPI:  
**26.02.2010**

45 Fecha de la publicación del folleto de la patente:  
**26.02.2010**

73 Titular/es: **INPECO IP Ltd.**  
**259, St. Paul Street**  
**VLT 1213 Valletta, MT**

72 Inventor/es: **Fava, Daniele;**  
**Garghentino, Elisabetta y**  
**Pedrazzani, Gianandrea**

74 Agente: **Elzaburu Márquez, Alberto**

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

Método para la gestión de sistemas de célula de trabajo basado en un sistema de gestión de la automatización.

5 La presente invención se refiere a un método para la gestión de Sistemas de Célula de Trabajo capaces de controlar procesos concurrentes que usan piezas de equipos, denominados en el presente documento Recursos, que interactúan entre sí según las necesidades y los cambios que no se han previsto por adelantado en sus ciclos de proceso, por medio de una reprogramación en tiempo real continua de tareas automáticas. Además, la invención se refiere a un Sistema de Gestión de Automatización.

10 Ocurre a menudo que se emplean sistemas de robótica para aumentar la productividad, reducir costes de producción, eliminar errores humanos, retirar Operadores humanos de un ambiente peligroso o simplemente encargarse de tareas aburridas y frustrantes.

15 Este es el caso de la automatización de procesos de producción industriales en donde se ha explotado el uso de dispositivos electromecánicos para la ejecución de tareas repetitivas y, en ocasiones, complejas. El concepto de automatización se ha limitado a ciclos que podrían interrumpirse al esperar la coincidencia con otras tareas, pero nunca con una reprogramación continua de tareas dependiendo de circunstancias que ni se prevean ni se puedan prever por adelantado.

20 Además, el concepto de automatización se ha limitado, hasta ahora, a procesos de producción en los que se dedican ciertas piezas de equipos a tareas específicas, con independencia de si estaban saturadas o no, y sin intentar usarlas para realizar procesos concurrentes o tomar cualquier decisión automática, basada en reglas de sistema experto, para volver a hacer etapas de procesos o para programar nuevos procesos según reglas aplicadas a los resultados logrados por el Sistema y/o ante peticiones inesperadas del Operador.

25 De esta manera, no pueden aplicarse conceptos de automatización de la técnica anterior a ambientes en los que pueden cambiarse tareas muy a menudo como, por ejemplo, las de Laboratorios de Investigación y Medicina de Laboratorio, que se distinguen por: un ambiente impredecible, uso de instrumentos de propósito general que pueden usarse para ensayos diferentes, automatización a pequeña escala, necesidad de una automatización flexible, necesidad de recargarse de equipos complejos y caros que se usan para protocolos diferentes que necesitan ejecutarse casi al mismo tiempo, necesidad de dar al sistema automatizado la posibilidad de modificar automáticamente los protocolos de muestra ante una petición de Usuario y/o reglas de un sistema experto basadas en los resultados del proceso de ensayo.

30 La solicitud de patente EP 596 205 describe un sistema analítico que comprende un sistema Supervisor de Banco, que es responsable de coordinar las operaciones de diversos instrumentos de hardware al ejecutar un método de banco o una secuencia de banco, lo cual requiere la intervención del usuario para verificar si todas las entradas están fijadas correctamente antes de que se ejecute la secuencia y programar la secuencia; para suspender y reprogramar una secuencia en curso; y para abortar una secuencia.

35 La solicitud de patente WO 01/09618 describe un sistema de software de laboratorio clínico integrado para ensayar un espécimen. Se proporciona un módulo de procesamiento de espécimen para realizar ensayos predeterminados particulares sobre el espécimen. Una programación de automatización de flujo de trabajo integrada se comunica con cualquiera de la pluralidad de módulos de procesamiento de especímenes. Los módulos de procesamiento de especímenes pueden incluir hardware de instrumentos y software de control de procesos embutidos. La programación de automatización de flujo de trabajo incluye una programación de procesamiento de peticiones para procesar una petición de usuario para cualquier de las pruebas que estén disponibles para su ejecución por los módulos de procesamiento de espécimen y también incluye una programación de control funcional que proporciona control funcional de uno cualquiera de entre la pluralidad de módulos de procesamiento de especímenes para realizar cualquiera de las pruebas, y la cual además incluye una programación de gestión de datos de resultado que proporciona un procesamiento de datos de resultado de prueba de cualquiera de las pruebas. Una programación de interfaz de usuario integrada se comunica con la programación de automatización de flujo de trabajo para permitir que un usuario controle y vigile todos los aspectos del funcionamiento del sistema de ordenador, incluyendo tareas preanalíticas, analíticas y postanalíticas.

40 Es un objeto de la presente invención superar la limitación anteriormente citada y hacer posible la ejecución contemporánea de procesos concurrentes que hagan uso de piezas de equipos comunes con el fin de obtener su mejor saturación y dar la posibilidad de automatizar procesos con entrada dinámica de datos dinámica y/o recepción de peticiones continuas.

Según la presente invención, se logran estos y otros objetos por medio de un Método para la gestión de Sistemas de Célula de Trabajo según la reivindicación 1.

65 Se considera que una Muestra es un espécimen de material en una forma que permite su tratamiento o ensayo en los Recursos adecuados, y un Protocolo de Muestra es una colección de datos que indican las tareas que han de ser ejecutadas en la Muestra asociada en dicho Sistema de Célula de Trabajo.

Por tales razones la invención se ha dedicado a la automatización de Protocolos de Laboratorio, pero que puedan aplicarse a automatización industrial, en el momento en el que la complejidad de procesos industriales alcance el mismo nivel de complejidad que los entornos de Laboratorio.

Las características y ventajas de la presente invención serán evidentes a partir de la siguiente descripción detallada de realizaciones de la misma, ilustradas como ejemplos no limitativos en los dibujos anexos, en los que:

La figura 1 muestra una estructura general de un Sistema de Célula de Trabajo diseñado según la presente invención;

La figura 2 muestra una disposición de un Sistema de Célula de Trabajo de Automatización de Laboratorio de Diagnóstico *In Vitro*;

La figura 3 muestra la disposición de un Sistema de Célula de Trabajo de Automatización de Laboratorio de Cribado de Alto Rendimiento o de Ácido Nucleico; y

La figura 4 muestra la disposición de un Sistema de Célula de Trabajo de Biorrepositorio.

Haciendo ahora referencia a la figura 1, en donde se representa una estructura general de un Sistema de Célula de Trabajo, R1, R2,...Rn representan los diversos Recursos, D1, D2,...Dn los Excitadores de Recursos asociado, S1 una Muestra genérica y SP1 el Protocolo de Muestra asociado, PC represente el Controlador de Procesos, DM el Gestor de Datos, LIS el Sistema de Información de Laboratorio, LD el Excitador del Sistema de Información de Laboratorio, U el Usuario u Operador humano, GUI la Interfaz Gráfica de Usuario, AMS el Sistema de Gestión de Automatización y WS el propio Sistema de Célula de Trabajo.

El funcionamiento del Sistema de Célula de Trabajo de la figura 1 se explicará en las siguientes realizaciones ejemplificantes.

Se muestra en la figura 2 un ejemplo de Sistema de Célula de Trabajo de Automatización de Laboratorio de Diagnóstico *In Vitro* (IVD LAWS), que integra como Recipientes de Muestra: tubos de muestra que contienen un fluido biológico (por ejemplo, sangre o cualquier otro material), soportes para muestras que portan, por ejemplo, 12, 20 0 48 tubos de muestra cada uno de ellos, y las siguientes piezas de equipo como Recursos: Dispositivos Portamuestras: un Módulo de Transporte y Posicionamiento Inpeco (TPM) 1, es decir, un sistema transportador por correa que transporta Palets (es decir, portamuestras individuales) conteniendo Tubos de Muestra (cada Palet es identificado por un transpondedor de radiofrecuencia, leído adecuadamente de vez en cuando por dispositivos de radiofrecuencia integrados en un Sistema de Lectura de Etiquetas de Transpondedor), un robot polar 2 de Módulo de Entrada/Salida (IOM) montado sobre una pista de cojinete neumático Inpeco para mover muestras desde/hasta el área de Entrada/Salida hasta/desde el TPM, un robot polar 3 de Módulo Centrífugo (CM) para mover muestras desde/hasta el área Centrífuga hacia/desde el TPM, uno o más sistemas robóticos 4 para interconectar el Analizador de Dispositivo de Ensayo de Muestra con el TPM.

El mismo ejemplo integra como Dispositivos de Manipulación de Recipiente de Muestra: dos lectores de código de barras Datalogic 5 para identificación de Recipiente de Muestra, un área de Entrada/Salida 6 equipada con sensores y accionadores para detectar y bloquear soportes de muestra, un Sistema de Visión 7 para reconocer soportes de muestra y tubos de muestra, dos Dispositivos de Detección de Recipiente de Muestra 8 para identificar altura y diámetro de tubo de muestra, un Sistema de Descoronación 9 para retirar cierras (tapas) de muestra de tubos de muestra según sea necesario, Dispositivos de Tratamiento de Muestra: una Centrífugadora Hettich Rotanta 10 para centrifugación de muestra y como Dispositivos de Ensayo de Muestra: hasta 10 diferentes Analizadores de Diagnóstico 11 Químico-Clinico, Inmunoquímico u otros, y un Platillo de Balanza 12 Sartorius BP6100 para medir pesos de muestra.

El estado de la técnica se ocupa de sistemas que realizan análisis de Diagnóstico de Laboratorio, prevé la ejecución manual de tareas preanalíticas (es decir, identificación de muestra, centrifugación, descoronación, carga de muestras en los Analizadores) y de tareas postanalíticas (es decir, descarga de muestras de los analizadores, clasificación de muestras, reejecución de ensayo en caso de errores, nuevas peticiones de análisis debido a reglas particulares), o la ejecución automática de sólo la fase preanalítica, pero de un modo secuencial, sin la manipulaciones de prioridades y concurrencia.

Por el contrario, gracias a la presente invención, el Sistema de Célula de Trabajo de Automatización de Laboratorio de Diagnóstico *In Vitro* que usa el Sistema de Gestión de Automatización de Laboratorio, automatiza todas las fases preanalítica, analítica y postanalítica del ensayo de Muestra de un modo homogéneo, concurrente, reactivo, dinámico y basado en reglas de sistema experto.

Las únicas tareas manuales que se han de realizar por el Operador son la carga y la descarga de Soportes de Muestra, que contienen tubos de muestra etiquetados con códigos de barras, que llegan desde el área de extracción de sangre del Hospital o desde otras localizaciones clínicas. Después de que el Operador carga los soportes en el área 6 de Entrada/Salida LAWS, el sistema de Célula de Trabajo percibe automáticamente y bloquea los soportes y el Robot IOM toma una fotografía de cada soporte con un cámara digital 7, identificando los tubos de muestra contenidos en el

## ES 2 333 697 T3

soporte y el ID de soporte (y, en consecuencia, el área en donde la muestra fue recogida o finalmente el flebotomista que realizó la recogida).

5 A continuación, el robot 2 IOM recoge automáticamente los tubos de los soportes y los pone en el TPM 1, en donde el Lector 5 de Código de Barras y el Dispositivo de Detección de Recipiente de Muestra 8 detecta el ID del Recipiente de Muestra y su dimensión (altura y diámetro). El ID del Recipiente de Muestra está asociado con el Protocolo de Muestra adecuado (es decir, las órdenes de ensayo) descargado por el Sistema de Información de Laboratorio (o insertado manualmente por el operador usando la Interfaz Gráfica de Usuario según sea necesario) y el ID de palet adecuado con el fin de seguir el Recipiente de Muestra a lo largo del TPM 1. Dependiendo del Protocolo de Muestra, 10 la Muestra se centrifuga según sea necesario (en esta fase, la cantidad de muestra se mide con el Platillo de Balanza 12 con el fin de conocer si la muestra es suficiente para realizar los análisis, y para cargar los Recipientes de Muestra en la Centrifugadora 10 de una manera equilibrada: se usa un algoritmo de equilibrio a este respecto), y se descorona según sea necesario por el Sistema Descoronador 9 (es decir, el cierre del Recipiente de Muestra se retira - también es posible cargar Recipientes de Muestra centrifugados previamente o descoronados previamente en localizaciones de 15 Entrada particulares, y estos tubos se tratarán en consecuencia por el Sistema).

Posteriormente, los Recipientes de Muestra se encaminan automáticamente hacia los Analizadores apropiados 11, en donde se realizan los ensayos apropiados. Cada Analizador 11 es manipulado por dos Excitadores de Recurso, gestionando el primero de ellos la comunicación de datos del Sistema de Información de Laboratorio (LIS), es decir, 20 la transmisión de las órdenes de ensayo y la recepción de resultados de muestra y/o de errores analíticos después de realizados los ensayos, gestionando el segundo la comunicación del Sistema de Automatización de Laboratorio (LAS), es decir, la gestión de las órdenes de automatización robóticas que cargan realmente la Muestra en el Analizador.

Después de haber realizado todos los análisis, el Recipiente de Muestra se mueve hacia el soporte de clasificación adecuado (dependiendo de la petición de ensayo que se ha de ejecutar finalmente en otros Analizadores fuera de 25 línea) y, cuando se reciben todos los resultados de prueba (y finalmente se calculan matemáticamente algunos resultados derivados), el Controlador de Procesos LAWS (el coordinador de excitadores de Recursos) toma una decisión, dependiendo de los resultados y las reglas configuradas del sistema experto, para liberar los resultados al Sistema de Información de Laboratorio o para pedir al Operador una intervención para validar los resultados. Finalmente, el 30 Gestor de Datos (el manipulador de la base de datos del Protocolo de Muestra) puede pedir automáticamente que se añadan nuevas órdenes de prueba al Protocolo de Muestra, o que se vuelvan a ejecutar automáticamente algunas pruebas y/o se añadan algunas nuevas pruebas (denominadas pruebas reflejas) al Protocolo de Prueba para una investigación analítica posterior (en estos casos, el Recipiente de Muestra se recoge automáticamente de nuevo por el Robot 2 IOM y se encamina, en consecuencia, hacia los Analizadores adecuados 11 - siempre que la cantidad de Muestra se 35 estime suficiente para finalizar los análisis recientemente solicitados).

Gracias a la presente invención es posible coordinar todos los Recursos del Sistema de un modo multitarea/multienfilado y, por tanto, permitir que cada Recipiente de Muestra tenga un tratamiento diferente basado en una prioridad de Protocolo de Muestra, características de Muestra y Recipiente de Muestra (es decir, si la muestra debe 40 o no centrifugarse, o si el recipiente de muestra tiene o no el cierre, o la altura, diámetro y peso del recipiente de muestra), en ensayos del Protocolo de Muestra que se han de ejecutar (pueden ejecutarse ensayos en Analizadores diferentes y pueden clasificarse Recipientes de Muestra en diferentes soportes de salida según los ensayos que se dejan a Analizadores que están situados fuera del Sistema de Célula de Trabajo de Automatización de Laboratorio), y en resultados de ensayo que puedan provocar nuevas peticiones de ensayo.

Algunos recursos críticos, como el área 6 de Entrada/Salida, la Centrifugadora 10, o algunas zonas intermedias de Analizador 11 (es decir, que en colas esperan Palets conteniendo Muestras a ensayar en esos Analizadores), que pueden 45 provocar cuellos de botella en el sistema, son manejados con reglas basadas en un sistema experto configurables por el Usuario, que permiten que el Controlador de Proceso tenga un control inteligente del flujo de equilibrio de carga de trabajo, adoptando las decisiones adecuadas sobre el encaminamiento del Recipiente de Muestra.

Por ejemplo, el sistema estima automáticamente el tiempo antes de la finalización de cada ensayo de Muestra, y posteriormente el sistema distribuye Recipientes de Muestra a lo largo del Sistema con el fin de optimizar el flujo de trabajo, evitando colas y cuellos de botella. Asimismo, si un Recurso está fuera de línea u ocupado (por ejemplo, 55 un Analizador), algunas tareas que estaban programadas para ese Recurso se reprograman automáticamente para un recurso diferente (por ejemplo, otro Analizador equivalente), maximizando la eficiencia y rendimiento. Asimismo, dependiendo de una petición de Usuario o de algoritmos de reprogramación accionados por eventos, el proceso de ensayo es distribuido a lo largo de los Analizadores equivalentes con el fin de minimizar la división de peticiones de prueba (es decir, la carga de Muestras en Analizadores equivalentes diferentes), o para obtener resultados más rápidos (disminuyendo así el tiempo medio de vuelta de resultados), o para minimizar instrumentos (por ejemplo, durante 60 el período nocturno, en donde únicamente deben ejecutarse unas pocas muestras) o para obtener el mejor ajuste de ensayo en Analizadores (con el fin de tener el mejor rendimiento), o según otras reglas.

El Usuario puede acceder a la interfaz Gráfica de Usuario para modificar los parámetros de configuración y las 65 reglas para manipular la reejecución de ensayo automática, la experimentación refleja y el cálculo matemático de resultados derivados o para ver resultados de ensayo y finalmente validar o programar una reejecución de las pruebas que el sistema experto del Gestor de Datos LAWS interno no manipula automáticamente o por motivos de diagnóstico o resolución de problemas.

## ES 2 333 697 T3

La comunicación entre el Controlador de Procesos y cada Excitador de Recurso se logra mediante un conjunto de variables denominadas RLS (Estado Lógico de Recurso) y RPS (Estado Físico de Recurso).

La variable RLS indica el estado de la ejecución de tarea actual del Recurso relevante, es decir, puede ser Listo si el Recurso está listo para ejecutar nuevas tareas, Ocupado si el recurso está ocupado ejecutando una tarea y, por tanto, no puede actualmente recibir nuevas órdenes, Error si está en un estado de error, Fin de Tarea si ha terminado su tarea actual y está esperando que el Controlador de Proceso actualice las variables RPS, etc.

Las variables RPS indican que el estado físico de un Recurso, por ejemplo, para un Analizador, el RPS es un conjunto de variables que indican el menú actual del Analizador mismo (es decir, la lista de todas los reactivos calibrados disponibles para realizar las pruebas), los consumibles disponibles (es decir, puntas, alícuotas, tubos, etc.) y la carga de trabajo actual del Analizador (es decir, el número de ensayos que se están ejecutando actualmente, el tiempo medio estimado para la finalización, el número de pruebas programadas que se han de ejecutar, etc.)

El Controlador de Procesos, dependiendo de las variables RLS y RPS de los diversos Recursos y de los Protocolos de Muestra de las Muestras que actualmente están en proceso en el Sistema de Célula de Trabajo, decide automáticamente, paso a paso, dónde y cuándo ejecutar un tratamiento de muestra (por ejemplo, centrifugar), manipular un recipiente de muestra (por ejemplo, descoronar) y ensayar una muestra (es decir, ejecutar algunas de todas las pruebas en las muestras relevantes).

Un ejemplo de Sistema de Célula de Trabajo de Automatización de Laboratorio de Cribado de Alto Rendimiento (HTS LAWS) o un Sistema de Célula de Trabajo de Automatización de Laboratorio de Tecnología de Ácido Nucleico (NAT LAWS), que usa el Método de la presente invención, mostrado en la figura 3, integra como Recipientes de Muestra: Placas de Microvaloración (placas que contienen un número múltiple de Muestras, situadas en porciones de la Placa denominadas pocillos), Soportes de Punta (recipientes de puntas, que se usan por algunos Recursos para mover parte de un Muestra hacia y desde el Recipiente de Muestra y Tubos de Muestra) y las siguientes piezas de equipo como Recursos: Dispositivos Portamuestras: robot articulado 13 CRS Robotics A255 montado sobre una pista de cojinete neumático Inpeco TR-10/4 (un doble par de pinzas está diseñado para transferir placas de Microvaloración y Soportes de Punta y Tubos de Muestra hacia y desde todos los Recursos en el LAWS), y un plato giratorio 14 (para giro de placa).

El mismo ejemplo integra Dispositivos de Manipulación de Recipiente de Muestra: un lector 16 de código de barras MicroScan Raster para identificación de Recipiente de Muestra, dos Carruseles de Almacenamiento de Placa 16 que puedan cada una acomodar 144 Placas de Microvaloración, un Carrusel de Almacenamiento 17 de Soporte de Punta, que pueda acomodar 72 soportes de punta o placas de pocillo profundo, un Sellador 18 de Placa Sagian, dos Patines 19 de Placa para eliminación de placas de desechos y soportes de punta, como Dispositivos de Tratamiento de Muestra: tres Dispensadores 20 de 8 Canales LabSystems Multidrop, un Robot 21 de Manipulación de Líquido Beckman Biomek 2000, un Dispensador 22 de 96 canales Beckman Multimek, un Agitador 23 de Placa de 20 posiciones verticales Inpeco, una Incubadora 24 Heraeus BB 6220 M3 con una puerta trasera deslizante, una Centrifugadora 25 de Placa Beckman GS 6R modificada por Inpeco para carga automática, un Lavaplaques 26 SLT Columbus, y como Dispositivos de Ensayo de Muestra: un Contador de Centelleo 27 Wallac TRilux 1450-025 con opción ParaLux, un Detector Multietiqueta 28 Wallac Victor2, un Lector 29 de Placa Colorimétrico SLT SpectraIII.

En el estado de la técnica, existen sistemas de cribado de robot múltiple o de módulos múltiples en los que Recipientes de Muestras podrían manipularse fácilmente a una velocidad de 1 placa por minuto. Sin embargo, leer las placas, por ejemplo mediante conteo radiométrico o fluorescente, lleva típicamente de 8-12 minutos por placa. En consecuencia, esta clase de sistemas tienen serios problemas de cuello de botella y las placas se cuentan normalmente (es decir, se ensayan) fuera de línea. Realmente, el conteo de placas es parte del protocolo mismo y su capacidad debe considerarse e incluirse al definir el rendimiento de un programa de cribado.

No sólo la lectura de placas, sino también otros pasos pueden provocar cuellos de botella en un sistema, aunque en menor medida. Además, pueden acontecer pequeños problemas técnicos que pueden bajar temporalmente la velocidad o detener el sistema. Si no se pueden evitar cuellos de botella en un protocolo, podría ser posible usar un agente de bloqueo para detener, por ejemplo, un ensayo de enzima. De esta manera, el cronometraje del resto del protocolo ya no es crítico y la placa puede leerse después de la finalización de los pasos que tienen limitaciones temporales o que finalmente son puestos fuera de línea.

Por tanto, para aumentar el rendimiento de un cierto formato de cribado y para impedir cuellos de botella más allá de un cierto límite, será necesario usar robots múltiples y, en algunos casos, instrumentos o piezas de equipo (por ejemplo, 2 o 3 contadores de centelleos) y/o pasos de procesamiento fuera de línea.

La ejecución de una gran operación de cribado con decenas de millones de muestras al año en placas puede justificar fácilmente la inversión en una serie de sistemas de cribado, cada uno de los cuales puede especializarse para realizar un tipo particular de ensayo, tales como celular (radiométrico o fluorescente), enzimático (que incluye también colorimétrico) u otros. Los instrumentos o equipos que son parte de cada sistema no necesitan ser particularmente diferentes entre sí, dado que normalmente se realiza rutinariamente el mismo tipo de experimento. Por este motivo, la capacidad de incubadoras, apiladoras, dispensadores, lavadoras e incluso lectores puede ajustarse al tipo particular

de cribado, lo cual debe permitir un gran nivel de optimización. alternativamente, varios sistemas de cribado podrían realizar esencialmente el mismo tipo de trabajo que un sistema de filtrado especializado de este tipo.

El caso se hace algo diferente cuando ha de ejecutarse un proceso de cribado menor, quizás no para decenas de millones de muestras por año, sino para 5-10 millones de muestras, con un componentes significativo de curvas de concentración-respuesta para optimización de aciertos y fallos. Esto puede convertirse ahora en un asunto de inversión el tener varios robots de pipetado, carruseles de placa incubadora y más lectores de placa costosos para un rendimiento relativamente bajo por célula de trabajo de Laboratorio, al tiempo que se mantiene la flexibilidad para responder a las necesidades variables de los objetivos y los formatos de ensayo correspondientes. En este caso una sola célula de trabajo de cribado robótica, pero relativamente completa, equipada para una amplia gama de tipos de ensayo, que pueda conmutarse fácilmente entre uno y otro tipo de ensayo, puede ser una buena solución. Dado que todas las piezas de equipo y operaciones (excluyendo quizá el cultivo de células en placas y la preparación de placas de compuestos, que a menudo se ejecutan mejor y más fácilmente en laboratorios especializados independientes) forman parte de la célula de trabajo, el ensayo mismo debe ejecutarse sin intervención humana hasta el transporte de placa a los recipientes de desechos.

Dado que cada tarea robótica puede llevar hasta un minuto, resulta claro que ensayos sencillos tienen un rendimiento más alto de estado estable que ensayos complejos. El tiempo de robot real puede variar de 2 a 7 minutos en la mayoría de ensayos de cribado típicos, lo cual permite un rendimiento teórico de 8 a 30 placas por hora usando un sistema de robot único flexible. Si el sistema se opera 8 horas al día, 4 días a la semana, 40 semanas al año, esto permite capacidades de cribado de 1,0 a 3,9 millones de muestras por año usando placas de 96 pocillos e incluso de 4 a 15 millones de muestras usando el formato de 384 pocillos.

La programación estática de protocolos de cribado proporciona una historia de muestra idéntica a todas las placas, aunque la optimización apropiada del rendimiento y la manera en la que los protocolos son ejecutados (el orden de operaciones) están basadas en el conocimiento y la experiencia del científico. Unos sistemas de programación dinámicos tienen varias ventajas interesantes, como la posibilidad de añadir placas u otros protocolos una vez que la célula de trabajo ya ha comenzado la ejecución, o la posibilidad de reanudar la ejecución original de manera automática después de una intervención del Usuario u otras clases de interrupciones. Los sistemas de programación dinámica usualmente no son capaces de mantener el cronometraje solicitado y, por ello, únicamente son útiles para protocolos con limitaciones de cronometraje relajados.

Gracias a la presente invención, es posible proporcionar un Sistema de Célula de Trabajo de Automatización de Laboratorio particularmente interesante para situaciones de rendimiento medio/alto en donde se requieren una flexibilidad y adaptabilidad muy altas. En particular, la reactividad del sistema otorga la posibilidad de soportar muchos formatos de ensayo diferentes de manera secuencial o en paralelo y una adaptabilidad inmediata para programar cambios y circunstancias no previstas alrededor y durante la ejecución de un programa.

Además, el Sistema de Célula de Trabajo de Automatización de Laboratorio según la presente invención, es flexible y, por tanto, capaz de soportar cualquier clase de Protocolo; la implementación del ensayo es fácil y rápida de realizar; los resultados de Muestra son accesibles inmediata y fácilmente en un formato estándar; y es rigurosamente modular, de modo que cualquier subsistema podría sustituirse siempre que se requiera por un módulo modernizado o completamente nuevo.

El proceso de programación permite al Usuario del Sistema de Célula de Trabajo de Automatización de Laboratorio definir un conjunto de pasos de operaciones usando los Recursos del sistema, establecer el número de Placas de Muestras que se han de ensayar, generar un programa factible para procesar todas las Muestras y ejecutar el programa.

El Sistema de Gestión de Célula de Trabajo de Laboratorio incluye una Interfaz Gráfica de Usuario (GUI), capaz de crear los Protocolos que se han de almacenar en la Base de Datos de Protocolo y un Módulo de Diagnóstico para solucionar problemas finales. El Controlador de Proceso adquiere programas que se han de ejecutar y controla el cronometraje con la base de datos interna que mantiene la historia y el estado de cada placa procesada a través del sistema. Éste se comunica con los Excitadores de Recursos, al tiempo que crea un fichero de registro y unos archivos de exportación de datos para el Sistema de Información de Laboratorio. Finalmente, los Excitadores de Recursos se comunican con los Recursos de una manera modular y concurrente.

Cada placa tiene una copia local de su Protocolo de Muestra con su estado de ejecución asociado almacenado en la Base de Datos de Protocolo. Esto permite ejecutar de una manera sencilla múltiples Protocolos de Muestra bien secuencialmente o en competición.

Gracias a las variables RLS y RPS, el Controlador de Proceso conoce el estado de cada Recurso en cualquier momento. Así, cada Recurso puede funcionar como un módulo independiente de un modo paralelo (multitenfilado) con respecto a los otros Recursos del Sistema. Por tanto, se definen una serie de estados lógicos y físicos mediante los cuales el Controlador de Proceso pueda controlar el sistema por medio de dichos Recursos.

El RLS tiene un número finito de valores lógicos que permiten la comunicación entre cada Excitador de Recurso y el Controlador de Proceso. El Controlador de Proceso inicia actividades específicas únicamente cuando lee ciertos valores del RLS y puede establecer otros valores a comunicar al Excitador. El Excitador puede leer y escribir valores

## ES 2 333 697 T3

diferentes y actuar sobre los mismos en consecuencia. Usando el RLS como un medio de comunicación, el Controlador de Proceso y los Excitadores conjuntamente pueden ejecutar Protocolos de Muestra.

Pueden considerarse como un ejemplo los siguientes valores del RLS.

Apagado: el Controlador de Proceso comprueba si es necesario el Recurso para unos Protocolos que se hayan previsto y, en ese caso, lanza la rutina de inicialización y establece el RLS como Ocupado.

Listo: el Controlador de Proceso, dependiendo del Estado Físico de Recurso (RPS), explicado en el presente documento a continuación, puede lanzar una orden para ejecución y establecer el RLS como Ocupado, o descargar la placa del equipo y ajustar el RLS a Transporte.

Transporte: en este caso, es el Excitador de Recurso, en comunicación con el Excitador de Dispositivo Portamuestras, el que descarga la placa y establece el RLS como Listo.

Ocupado: también en este caso el Controlador de Proceso no reacciona mientras el Excitador de Recurso, junto con el Excitador de Dispositivo Portamuestras, está cargando la placa y activando la orden ejecutable solicitada para ese Recurso. Una vez que el Excitador de Recurso indica la finalización de la tarea, el Excitador de Recurso ajusta el RLS a Fin de Tarea. En el caso de que el Excitador de Recurso indique un error que no pueda recuperarse automáticamente por el Excitador de Recurso mismo, éste ajustará el RLS a Error. Un caso particular tiene lugar en un ensayo multiplaca, en el que más de una placa necesita cargarse en un Recurso antes de que pueda ejecutar su propio protocolo. En este caso, después de la llegada de una placa, el Excitador de Recurso, en sincronía con el Excitador de Dispositivo Portamuestras, pone inmediatamente su RLS en Listo para permitir la carga de una placa siguiente.

Fin de Tarea: este valor del RLS induce al Controlador de Proceso a actualizar el protocolo de la placa analizada y a ejecutar la gestión de los datos producidos por el lector de placa. Después de esto, ajusta el Excitador de Recurso a Listo.

Error: el valor de error del RLS se indica al Operador cuando el Excitador de Recurso no es capaz de resolver un mensaje de error, enviando la orden apropiada a la Interfaz Gráfica de Usuario (GUI). El Operador tiene que introducir el Módulo de Diagnóstico y, sobre la base del código de error específico, tiene que decidir entre volver a ejecutar la tarea (quizá después de una intervención manual o de software a bajo nivel) o considerarla finalizada y proseguir con la siguiente tarea. En el caso de un error irrecuperable, el Operador puede decidir finalmente abortar el programa. El Módulo de Diagnóstico mismo ejecutará entonces la orden adecuada, ajustará los valores de RLS y/o RPS apropiados y devolverá el control al Controlador de Proceso para continuar o abortar el guión del protocolo relevante.

El Excitador de Dispositivo Portamuestras y el Controlador de Proceso necesitan los parámetros de Estado Físico de Recurso (RPS) para definir un protocolo de transferencia de placa fiable entre el Dispositivo Portamuestras (es decir, el robot, en este caso) y el Recurso adecuado. En particular, los Recursos que necesitan abrir o cerrar una puerta o hacer girar un carrusel de placa antes de que puedan ser cargados, descargados u ordenados que ejecuten un protocolo, necesitan parámetros adicionales para permitir una comunicación inequívoca. Estos parámetros son Espacio Libre, Lleno, Abierto, Cerrado, Descarga y Carga, dependiendo del hecho de que dicho Recurso tenga espacio libre para placas nuevas o esté lleno, o esté en su estado abierto, cerrado, de descarga o de carga, respectivamente. El proceso que necesita ser controlado en detalle es siempre la descarga de un Recurso de origen y la carga de un Recurso de destino. Esto ocurre con la siguiente secuencia.

Descarga: el Controlador de Proceso, sobre la base del protocolo de placa y el RLS de Recurso, necesita mover una placa. Este comprobará si el Recurso RLS de destino está Listo y el RPS tiene Espacio Libre y si el Recurso de origen de RLS está Listo y el RPS tiene Espacio Libre (para capacidad multiplaca) o Lleno (para capacidad de una sola placa). Si estas condiciones son satisfechas, el Controlador de Proceso envía sus órdenes al Excitador de Recurso de Destino y al Excitador de Dispositivo Portamuestras, al tiempo que ajusta el RLS de ambos Excitadores a Ocupado. El Excitador de Dispositivo Portamuestras ajusta el RPS del Excitador de Recurso de origen a Descarga, de lo cual se acusa recibo por el Controlador de Proceso al cambiar el RLS a Transporte. En este punto, el Excitador de Recurso, encontrándose lógicamente en Transporte y físicamente en Descarga establecerá el Recurso en la condición de recibir un placa haciendo girar el carrusel de placa, abriendo la puerta de acceso y ajustando el RPS a Abierto. El robot puede ahora tomar la placa después de lo cual el Excitador de Dispositivo Portamuestras ajusta el RPS a Cerrado. La operación de descarga concluye ahora.

Transporte: Si una placa se mueve por primera vez, el código de barras aún no es conocido y se leerá en primer lugar. La placa también girará según sea necesario, mirando el lado corto o largo hacia la posición derecha del Recurso de destino.

Carga: el Excitador de Dispositivo Portamuestras ajusta ahora el RPS del Recurso de Destino a Carga de modo que pueda ejecutar la tarea para preparar el Recurso para recepción y para ajustar el RPS a Abierto. El robot puede poner ahora la placa en su lugar y después de Excitador de Dispositivo Portamuestras ajustará el RPS a Cerrado.

Esta manera de gestionar el transporte de placas permite la posibilidad de usar los Recursos de la mejor manera posible. Por ejemplo, a una puerta de incubadora de placa, que debe permanecer cerrada tanto como sea posible, se le

dará la orden de abrir su puerta únicamente cuando el robot ya tenga la placa que se ha de cargar en sus pinzas. Además, este mecanismo permite una gestión muy sencilla de ciertos errores. De hecho, cada una de las modificaciones de RPS o RLS sólo sucede cuando la tarea anterior se ha finalizado exitosamente, impidiendo la posibilidad de que el brazo de robot golpee una puerta cerrada o ponga una placa en una posición en la que no esté lista para recibirle.

Una vez que el Controlador de Proceso conoce los estados lógico y físico de cada Recurso, es posible ejecutar un protocolo. Sin embargo, dado que pueden tener lugar en paralelo muchas tareas y que más de una placa estará normal y simultáneamente en el espacio de trabajo del Laboratorio, es muy probable que dos o más tareas exijan servicio de robot al mismo tiempo y que el Controlador de Proceso tenga que decidir qué acción ejecutar. La ejecución en tiempo real de tareas basándose en una ejecución inmediata de interrupciones procedentes de Recursos múltiples es, por tanto, un objetivo poco realista, dado que serían necesarios un robot independiente y una capacidad ilimitada en cada paso.

En este punto, se define un conjunto de heurística, reglas y parámetros para establecer la prioridad de cada placa en cualquier momento en cualquier tipo de protocolo. Estas reglas de prioridad dependen en gran medida del protocolo mismo y siguen en su mayor parte la lógica de los requisitos biológicos que el experimento plantea en el sistema. Por tanto, los estados lógicos y las reglas de prioridad permiten que el Controlador de Proceso y el Excitador de Dispositivo Portamuestras funcionen continuamente sobre la base de un protocolo sin un programa preplanificado.

En primer lugar, el sistema debe seguir el principio de que el primero que entra es el primero que sale para impedir que una placa invada otra en un Recurso multiplaca (típicamente incubadoras) o en un protocolo multiplaca. Una consecuencia particular de esta regla es que también placas que hayan avanzado adicionalmente en el protocolo tienen prioridad sobre placas que acaben de comenzar su protocolo. Esta regla favorece la descarga de Recurso sobre la carga y ayuda a impedir colas y a liberar posiciones del sistema.

Puede introducirse por el usuario un parámetro de Espera por medio de la GUI y asociarse a cualquier Recurso que no tenga comunicación bidireccional para indicar el final de su proceso (como agitadores de múltiples placas e incubadoras virtuales, es decir, placas que esperan a temperatura ambiente el siguiente paso de procesamiento). El Controlador de Proceso no moverá placas antes de que haya transcurrido el periodo de Espera. El parámetro de Espera también permite enlazar dos pasos no consecutivos. Si una placa necesita ser leída 10 minutos antes de la adición de un reactivo, pero aún existe un paso de agitación y un paso de sellado entremedias, es posible colocar la placa en un carrusel después del paso de sellado y especificar un periodo de incubación de 10 minutos con respecto al paso de adición de reactivo. Esto permite un control muy preciso de una reacción y puede impedir el uso de reactivos de detención.

Un parámetro de Cronometraje Crítico (un retardo) puede ser introducido por el Usuario por medio de la GUI y asociarse a cualquier Recurso. Este es puramente un parámetro impulsado por biología. Si, por ejemplo, un periodo de incubación de células o membranas en una incubadora de CO<sub>2</sub> debe llevar 45 minutos, pero no más de 50 minutos, el robot debe finalizar su actividad actual y descargar la placa de la incubadora dentro de los 50 minutos. Al final del periodo de 50 minutos esta operación debe tener prioridad absoluta y el robot deberá abandonar cualquier otra actividad para permitir la operación de Descarga antes de continuar el programa. El Cronometraje Crítico puede imponerse a cada paso de incubación, incluyendo incubaciones virtuales definidas por el parámetro Esperar.

Estas tres reglas por sí mismas son suficientes para definir prioridades entre placas. Realmente, éstas se usan para definir una puntuación de prioridad única, basada en tiempos de Espera expirados y el Cronometraje Crítico de cada placa en cada movimiento de placa, el cual se usa como una herramienta sencilla para establecer la prioridad de cada placa en cualquier momento en el Sistema de Célula de Trabajo de Automatización de Laboratorio.

Existen casos en los que existe un paso lento en el protocolo con capacidad limitada, por ejemplo lectura de las placas con un betacontador, lo cual puede necesitar en ocasiones de 10-15 minutos por placa. El Controlador de Procesos siempre tenderá a maximizar el rendimiento del sistema y a iniciar la ejecución de una nueva placa si no hay otras actividades con una prioridad más alta y el robot está libre. Como el tiempo de robot total de un protocolo es a menudo sólo de 5 a 7 minutos, esto provocará una sobrecarga y una lista de espera creciente justo antes del betacontador relativamente lento.

Existen dos enfoques radicalmente diferentes de este problema. El primero es tener fuera de línea las placas para contarlas o, si el cronometraje no es importante, dejar las placas sobre el LAWS para contarlas cuando el betacontador esté libre. El segundo es crear algún mecanismo de realimentación para contrarrestar la sobrecarga del LAWS. Se introduce un nuevo concepto, el Cuello de Botella variable, para afrontar los problemas de tráfico. El Cuello de Botella es causado por un Recurso lento (como el betacontador de nuestro ejemplo), pero más generalmente representa el tiempo mínimo que se necesita entre el procesamiento de una placa y la siguiente sin causar un atasco de tráfico. El valor de Cuello de Botella se convierte, por tanto, en un parámetro crucial que permite el control del tráfico de placas en un LAWS. Su valor inicial puede fijarse (en la GUI) según la longitud de un paso lento conocido (derivado durante la optimización del ensayo) o tomando sencillamente 30-60 segundos para cada paso de protocolo, dado que este el tiempo de robot mínimo requerido en estado estable para procesar cada placa única. Pueden distinguirse tres casos para controlar totalmente el rendimiento.

Los Recursos solicitan tareas de robot de transporte aproximadamente en el mismo momento y se acumulan algunos retardos, sin provocar una violación de los parámetros de Retardo Crítico. Esencialmente, esto significa que



el robot de transporte no tiene suficiente tiempo libre para dar servicio a todas las peticiones. Sobre la base del retardo acumulado, el valor de Cuello de Botella puede aumentarse gradualmente hasta que no se acumulen retardos significativos. El único efecto sobre el LAWS es que entren nuevas placas con una frecuencia inferior y que decaerán gradualmente los retardos.

En caso de violación de los parámetros de Retardo Crítico, el sistema está en un estado de sobrecarga incluso si éste no se reconoce inmediatamente de manera normal como tal. En este punto, un simple aumento del parámetro de Cuello de Botella no es suficiente y se bloquea una nueva entrada de placa hasta que se haya gestionado la violación del Retardo Crítico. En este punto, el Cuello de Botella se recalculará como en el caso anterior y las placas que esperan su admisión al LAWS se reprogramarán en consecuencia.

El tercer caso consiste en la ausencia absoluta de retardos y en que el robot de transporte no tenga realmente tiempo desocupado. En este caso, el cuello de Botella puede acortarse gradualmente, basándose en el nivel de tiempo desocupado del robot mismo.

El sistema resultante provee realmente un bucle de realimentación negativa y positiva que puede optimizar efectivamente el rendimiento de la placa en el LAWS. Al dar al Cuello de Botella un valor de arranque cercano al valor experimental real, el Controlador de Proceso puede arrancar cerca del rendimiento óptimo y únicamente son necesarias correcciones menores. El resultado será un proceso muy regular, sin ningún problema que sea consecuencia de tener operando al robot de transporte a la capacidad máxima, con independencia de los cuellos de botella del protocolo.

Se puede comprender intuitivamente que al constreñir el Sistema de Célula de Trabajo de Laboratorio a un cronometraje muy preciso en múltiples pasos de un Protocolo se inducirá al sistema a reducir la frecuencia con la que placas nuevas arrancan su procesamiento dado que el robot de transporte dará servicio continuamente a pasos críticos. Por tanto, resulta necesario que restricciones precisas de Espera y otras junto con sus parámetros de Cronometraje Crítico se empleen con precaución y con sentido común. Cuanto menos crítico sea el cronometraje de un protocolo, mayor será la flexibilidad que tiene el LAWS para la optimización y mayor será el rendimiento.

Como el rendimiento total de un ensayo incluye pasos de conteo y análisis de datos, es imposible procesar 200 placas por día, ya que llevará al menos 2 días y 2 noches de conteo ininterrumpido completar el protocolo de ensayo. En ese caso, puede resultar bastante útil procesar 70 placas por día incluyendo conteo y análisis de datos, usando el tiempo de procesamiento libre del LAWS para hacer algo más. La única alternativa para ejecutar este ensayo con un mayor rendimiento sería integrando contadores adicionales de placas.

La Interfaz Gráfica de Usuario (GUI) tiene tres funciones diferentes. La primera es la creación de Protocolos de Muestra, es decir, la definición de un camino de procesamiento que cada placa de un lote puede experimentar durante un recorrido experimental. La segunda es la creación y ejecución de un programa de ejecución en lotes en el que también se definen todas las localizaciones de las placas al arranque de un recorrido, así como el Protocolo al que se someterán. Finalmente, existe un módulo de diagnóstico que permite acceso a la mayor parte de Recursos para solucionar problemas y para intervenir manualmente en caso de que ocurra un error irreparable.

La creación o modificación de cualquier protocolo desde la GUI es bastante intuitiva y rápida y se puede aprender con un entrenamiento mínimo. Esta consiste básicamente en señalar y hacer “click”, sobre un mapa del LAWS, en los Recursos que es necesario usar en el Protocolo. Para cada Recursos aparece una ventana flotante y es posible establecer una serie de parámetros específicos para ese Recurso en ese Protocolo. Estos parámetros incluyen elementos como adición de un cierto volumen de un reactivo a todos los pocillos o a pocillos específicos de una placa para un dispensador, velocidad y tiempo de agitación para una agitador de placas, un ID de protocolo de lectura para un lector de placa, el tiempo que una placa necesita permanecer en una incubadora o cualquier otra instrucción que especifique el Usuario.

En este punto, el Usuario también puede establecer las reglas de prioridad (véase en este documento a continuación) definiendo el parámetro de Cronometraje Crítico, por ejemplo, un tiempo máximo de incubación o un parámetro de Cuello de Botella opcional. Asimismo, es posible crear protocolos ramificados estableciendo un punto de Nudo en el protocolo. Esto permite, por ejemplo, la transferencia de una placa a otra posición, la introducción de nuevas placas (denominadas placas hijas) y la diferenciación de procesamiento de las placas originales y las placas hijas.

Mientras se crea el Protocolo de esta manera, se muestra una ventana de texto en la que se visualiza el guión de órdenes de bajo nivel. Es posible en cualquier momento de la definición del Protocolo (o cuando se recupera un Protocolo definido anteriormente) modificar, añadir o borrar cualquier línea de este guión. Se salvan protocolos en la Base de Datos de protocolo, con una identificación específica y un número de versión de modo que puedan usarse en cualquier momento futuro.

Para hacer funcionar el Sistema de Célula de Trabajo de Automatización de Laboratorio, el Operador tendrá que cargar uno o más protocolos e indicar tipos y posiciones de placa para cada Protocolo. Al igual que en la fase de definición de Protocolo, haciendo sencillamente click en los carruseles de placa (también dentro de una incubadora para ensayos celulares), aparece una ventana flotante de posición de placa, sobre la que el Usuario puede indicar qué posiciones de placas están llenas y qué tipo de placas o soportes de punta se usan (eligiendo de entre una lista apropiada de placas y soportes de punta disponibles). Realmente, cada Excitador de Recurso tiene su propia base de datos

pequeña en la que hace seguimiento de todas la placas que está conteniendo o procesando en ese momento temporal. Esto permite que el Controlador de Proceso y el Excitador de Dispositivo Portamuestras controlen y comprueben totalmente el estado de cada placa según sea necesario.

5 En cualquier momento, también durante la ejecución de un Protocolo, el Usuario puede definir recorridos adicionales y añadirlos sencillamente al lote en ejecución. El sistema permite un procesamiento secuencial, es decir se pondrán en marcha placas adicionales al final del lote de las placas ya en marcha (como si se hubieran añadido algunas placas al protocolo original), o un procesamiento en paralelo, es decir, se ejecutará el segundo protocolo tan pronto como el LAWS tenga tiempo de procesamiento libre disponible. Resulta evidente que ejecutar en paralelo dos protocolos  
10 complejos, que podrían saturar cada uno de ellos el LAWS, desacelerará significativamente a cada uno de los dos protocolos. La ejecución de un protocolo fácil en dos pasos en los tiempos de espera de un protocolo complejo con tiempos de incubación largos o un paso lento de lectura de placa puede ser muy exitoso y permitirá un uso óptimo del LAWS.

15 La GUI tiene algunas características adicionales que ayudan al Usuario en la resolución de estados de error que puedan ocurrir.

Cuando no se está ejecutando protocolo alguno, puede activarse una función de Mantenimiento para acceder al control a bajo nivel de cada Recurso, incluyendo el robot de transporte. De esta manera, cada recurso puede establecerse  
20 en cualquier estado deseado por el operador para fines de mantenimiento o control de calidad.

Durante una ejecución, cuando el Controlador de Proceso indique que ha ocurrido un error, que el sistema no podría recuperar automáticamente, el Operador podría activar la función de Diagnóstico, interrumpir la ejecución y acceder a las opciones de control de bajo nivel. De esta manera, se puede restablecer un Recurso mediante software o  
25 permitir la intervención manual o restablecer un Recurso (por ejemplo, para liberar una placa bloqueada en un lector o para ajustar las propiedades de un sellador de placa que ha pasado por alta una placa). Una vez que el estado de error ha sido cancelado, la operación fallida puede reintentarse o saltarse y puede abandonarse el modo de diagnóstico, permitiendo que el Controlador de Proceso continúe procesando el lote. El Operador tiene también la posibilidad durante una ejecución de intervenir sobre Recursos que no estén en un estado de error. Esto se realiza mediante la  
30 función de Mantenimiento. Finalmente, el Operador puede seleccionar en cualquier momento la función de Reserva para intervenir manualmente en el LAWS. El sistema detendrá la siguiente orden y continuará únicamente cuando el estado de Reserva sea liberado por el Operador.

En estos casos, la extrema flexibilidad de este control dinámico y reactivo del LAWS recuperará muy rápidamente  
35 el procesamiento de estado estable después de cualquier intervención de la ejecución por lotes. Basándose en su mecanismo de realimentación automático, la entrada de placas será desacelerada hasta que el sistema haya recuperado su estado estable.

El control del flujo de datos en el LAWS está basado completamente en la gestión de código de barras. Cada placa  
40 de cribado, teniendo un formato de densidad de pocillos de 96, 386 o más, debe tener un código de barras fijado en la parte delantera izquierda (lado corto) o en la parte delantera derecha (lado largo) de la placa. El código de barras define el contenido de la placa, el cual se puede recuperar de la Base de Datos. Durante el procedimiento de cribado no existe la necesidad de recuperar esta información. El código de barras se convierte en un mero ID de placa, el cual se asigna a un archivo de exportación de datos estandarizado. En el caso de que los compuestos puedan dispensarse  
45 previamente dentro de las placas, sino que tengan que ser dispensados durante el ensayo (por ejemplo, ciertos ensayos activados por agonista), el código de barras heredará el mapa de placa de la placa de compuesto madre. El archivo de exportación estandarizado contiene los datos que se derivan de cualquier lector de placa y una serie de metadatos que describen el experimento (como fecha, hora, nombre del Operador, nombre del protocolo, tipo de lector, ajustes de lectura para control positivo de, por ejemplo, ajustes correctos de filtros y unidades de medición). Los datos se guardan  
50 localmente, pero son accesibles continuamente a través de la red por el Sistema de Información de Laboratorio.

Se realiza un análisis de datos fuera de línea y éste se puede hacer justo desde el momento que ha finalizado la ejecución del cribado. Dado que el LAWS y el software de análisis de datos están ambos basados en una gestión centralizada del código de barras y del mapa de placas, no existe la necesidad de mantener un orden de placa particular  
55 mientras se carga el LAWS. La carga de 100 o más placas sobre los carruseles de placa puede realizarse en consecuencia sin el temor de invertir algún orden o localización de placa. Asimismo, resultará imposible asignar datos a la placa equivocada, debido a la identificación mixta de placa entre un lector y el sistema de cribado.

Un Ejemplo de un Sistema de Trabajo de Biorrepositorio que sigue el Método de la presente invención, mostrado en  
60 la figura 4, integra como Recipientes de Muestra: unos tubos de muestra que contienen un fluido biológico, que ha de mantenerse en frigoríficos durante un largo tiempo y que se entrega bajo petición para análisis de investigación, cribado genérico u otros motivos, y las siguientes piezas de equipo como Recursos: Dispositivos Portamuestras: un módulo de Transporte y Posicionamiento (TPM) Inpeco 30, es decir, un sistema transportador por correa que transporta Palets (es decir, portamuestras sencillos) que contienen Tubos de Muestra (cada Palet está identificado por un transpondedor de radiofrecuencia, leído adecuadamente de vez en cuando por dispositivos de radiofrecuencia integrados en un Sistema de Lectura de Etiquetas de Transpondedor), un robot 31 de Módulo de Entrada/Salida (IOM) para mover muestras desde/hasta el área de Entrada/Salida hasta/desde el TPM y uno o más sistemas 32 de robot de transporte capaces de mover muestras hacia dentro y hacia fuera de sistemas frigoríficos en donde se cargan las muestras que se han de  
65

almacenar o se descargan las que se van a ensayar, como Dispositivos de Manipulación de Recipientes de Muestra: un lector 33 de código de barras para identificación de muestras, uno o más sistemas 34 de refrigeración (u otros sistemas de almacenamiento) para almacenar muestras, y un lector 35 de código de barras dentro de cada sistema de refrigeración para lectura de doble comprobación de Recipientes de Muestra.

La técnica anterior en Biorrepositorios prevé la carga y descarga manual de muestras en y desde unos frigoríficos, creando así problemas cuando el número de muestras que se ha de manipular es alto e introduciendo riesgos para los operadores, especialmente a temperaturas ambiente bajas en cámaras frías o debido a frigoríficos basados en nitrógeno líquido.

Asimismo, la manipulación manual de muestras introduce algunos problemas en la gestión y consistencia de la base de datos de muestras, la posibilidad de errores humanos y una gestión ineficiente del espacio dentro de los frigoríficos (de hecho, es difícil y lleva tiempo optimizar manualmente la distribución de muestras dentro de estos sistemas).

Por el contrario, gracias a la presente invención, el Sistema de Célula de Trabajo de Biorrepositorio que usa el Sistema de Gestión de Automatización de Laboratorio, automatiza todas las tareas del Biorrepositorio, es decir, la carga, descarga y compactado de muestras.

Este objetivo se logra centralizando la base de datos de muestras en el Gestor de Datos del Sistema de Célula de Trabajo y manipulando la carga y la descarga de muestra hacia dentro y hacia fuera de frigoríficos de una manera totalmente automática, permitiendo que el Operador cargue muestras en un área de Entrada/Salida, en donde, automáticamente, se reconocen las muestras y se transportan y cargan éstas en el frigorífico adecuado, y se solicitan localmente muestras desde la Interfaz Gráfica de Usuario o remotamente desde el Sistema de Información de Laboratorio, permitiendo al sistema descargar automáticamente las muestras solicitadas desde los frigoríficos adecuados y transportarlas hacia el área de Entrada/Salida, recompactando según sea necesario el espacio libre generado dentro de los frigoríficos.

La gestión de esta clase de Sistema de Célula de Trabajo es similar a la descrita en los ejemplos anteriores y puede integrarse con cualquier Dispositivo de Ensayo para realizar automáticamente los análisis de cribado o ensayo solicitados en las muestras adecuadas.

En particular, la variable RLS indica el estado de la ejecución actual de una tarea del Recurso relevante, es decir, puede ser Listo si está lista para ejecutar nuevas tareas, Ocupado si está ocupada ejecutando una tarea y no puede actualmente recibir nuevas órdenes, Error si está en un estado de error, Fin de Tarea si ha terminado su tarea actual y está esperando que el Controlador de Proceso actualice las variables RPS, etc.

Las variables RPS indican el estado físico de un recurso, por ejemplo, para un sistema de refrigeración, el RPS es un conjunto de variables que indican el número de muestras contenidas en ese Recurso, su estado de fragmentación (para opciones de recompactación, su accesibilidad y disponibilidad, etc.

## REIVINDICACIONES

5 1. Método para la gestión de Sistemas de Célula de Trabajo (WS) usando un Sistema de Gestión de Automatización (AMS) para controlar una pluralidad de Recursos (R1-Rn) designados para manipular una serie de Muestras (S1) a lo largo de dichos Sistemas de Célula de Trabajo (WS) a fin de realizar operaciones sobre dichas Muestras (S1), comprendiendo los siguientes pasos:

asociar un Protocolo de Muestra (SP1) a cada Muestra (S1),  
10 asociar a cada Recurso (R1-Rn) un Excitador de Recurso (D1-Dn), capaz de excitar dicho Recurso (R1-Rn),

controlar cada uno de dichos Excitadores de Recurso (D1-Dn) mediante un Controlador de Procesos (PC), el cual se comunica con los otros Excitadores de Recurso (D1-Dn) por medio de unas variables (RLS, RPS) asociadas a cada Recurso (R1-Rn), definiendo y modificando dichas variables (RLS, RPS) el estado de dichos Recursos (R1-Rn),  
15

ejecutar dichos Protocolos de Muestra (SP1) por dicho Controlador de Proceso (PC) de una manera concurrente para lograr multienfilado, evitación de cuellos de botella, asignación dinámica de Recursos (R1-Rn) en exclusión mutua, manipulación prioritaria de muestras, equilibrio de carga de Recursos y recuperación de errores automática,  
20

**caracterizado por**

modificar automáticamente dichos Protocolos de Muestra (SP1) reprogramando tareas automáticas en diferentes Recursos (R1-Rn) según los resultados de la etapa de ejecución de dichos Protocolos de Muestra (SP1) y el estado de dichos Recursos (R1-Rn), indicado por un estado de dichas variables (RLS, RPS).  
25

2. Método según la reivindicación 1, en el que el Sistema de Célula de Trabajo (WS) es un Sistema de Célula de Trabajo de Automatización de Laboratorio (LAWS), y cada Muestra (S1) es un espécimen biológico recogido en un Recipiente de Muestra (ID), que incluye una o más Muestras (S1).  
30

3. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo (WS) incluye un Gestor de Datos (DM) que manipula datos de Muestra y Protocolos de Muestra (SP1); operando dicho Gestor de Datos (DM) en coordinación con los otros Recursos de Sistema de Célula de Trabajo (R1-Rn) por medio de un conjunto de variables (RLS, RPS) usadas para comunicación.  
35

4. Método según la reivindicación 1, en el que se asocia una Base de Datos a dicho Gestor de Datos (DM); dicha Base de Datos representa un colección de Protocolos de Muestra (SP1), Resultados de Muestra, mientras que un Resultado de Muestra es la recogida de datos medidos por uno o más dispositivos de Ensayo (11) o calculados por el Gestor de Datos (DM) mismo usando fórmulas matemáticas definidas aplicadas sobre otros datos incluidos en los Resultados de Muestra.  
40

5. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo (WS) incluye una Interfaz Gráfica de Usuario (GUI) que se interconecta con el Usuario (U), funcionando dicha Interfaz Gráfica de Usuario (GUI) en coordinación con los otros Recursos de Célula de Trabajo (R1-Rn) por medio de un conjunto de variables que indican el estado del Recurso mismo.  
45

6. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo (WS) incluye un Excitador de Sistema de Información de Laboratorio (LD) para manejar la comunicación con el Sistema de Información de Laboratorio (LIS); dicho Excitador de Sistema de Información de Laboratorio (LD) funciona en coordinación con los otros Recursos de Sistema de Célula de Trabajo (R1-Rn) por medio de conjunto de variables (RLS, RPS) que indican el estado del Excitador (D) mismo.  
50

7. Método según la reivindicación 2, en el que dicho Sistema de Célula de Trabajo (WS) incluye como Recursos (R1-Rn) uno o más Dispositivos Portamuestras, diseñados para mover Muestras (S1) a lo largo del Sistema de Célula de Trabajo de Automatización de Laboratorio (LAWS).  
55

8. Método según la reivindicación 1, en el que dicho Sistema de Célula de incluye como Recursos uno o más Dispositivos de Tratamiento de Muestras para modificar las Muestras (S1) para los ensayos solicitados.  
60

9. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo (WS) incluye como Recursos (R1-Rn) uno o más dispositivos de Manipulación de Recipiente de Muestra, dedicados a dispensar, almacenar, desechar o alterar las propiedades de un Recipiente de Muestra (ID).  
65

10. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo incluye como Recursos (R1-Rn) uno o más Dispositivos (11) de Ensayo de Muestra dedicados a determinar la cantidad de uno o más componentes en una Muestra (S1).

11. Método según la reivindicación 2, en el que dicho Sistema de Gestión de Automatización (AMS) controla e integra dichos Recursos (R1-Rn) con el fin de obtener un tratamiento homogéneo y concurrente de Muestras (S1) a lo largo del Sistema de Célula de Trabajo (WS) asociando a cada Recipiente de Muestra (ID) un Protocolo de Muestra (SP1) y ejecutando todos los Protocolos de Muestra (SM1) de una manera concurrente.

12. Método según la reivindicación 1, en el que cada Recurso (R1-Rn) está asociado a un Excitador de Recurso (D1-Dn) designado para controlar la comunicación y la ejecución de tarea de las piezas de equipo relevantes en coordinación con los otros Recursos de Sistema de Célula de Trabajo por medio de un conjunto de variables (RLS, RPS) que representan estados de Recursos.

13. Método según la reivindicación 2, en el que cada Recipiente de Muestra (ID) es asociado a un identificador único y cada identificador es asociado a un Protocolo de Muestra (SP1), mientras que dicho Protocolo de Muestra (SP1) tiene que ejecutarse en la Muestra (S1) o en el conjunto de Muestras (S1) contenidas en el Recipiente de Muestra asociado (ID).

14. Método según la reivindicación 1, en el que el Sistema de Gestión de Automatización (AMS) prevé la modificación automática de los Protocolos (SP1), mientras se están ejecutando, dependiendo de una petición de Usuario, una petición del Sistema de Información de Laboratorio (LIS) o una petición automática.

15. Método según la reivindicación 1, en el que cada Protocolo de Muestra (SP1) es asociado a un Nivel de Prioridad actualizable durante la ejecución de dicho Protocolo (SP1).

16. Método según la reivindicación 1, en el que cada Recurso (R1-Rn) usado en un Protocolo de Muestra (SP1) es asociado a un parámetro de Espera para indicar el tiempo en que el Recurso (R1-Rn) completa su tarea en ese Protocolo de Muestra (SP1) y a un Parámetro de Tiempo Crítico al final del cual el Recurso (R1-Rn) debe tener una prioridad absoluta con respecto a cualquier otro Protocolo (SP1).

17. Método según la reivindicación 1, en el que cada Protocolo de Muestra (SP1) es asociado a un Parámetro de Cuello de Botella que puede ajustarse de una manera que proporcione un bucle de realimentación positiva y negativa que pueda optimizar el rendimiento del sistema.

18. Método según la reivindicación 1, en el que dicho Sistema de Célula de Trabajo de Automatización de Laboratorio (LAWS) es un sistema de cribado o ensayo de Muestra automatizado y en el que al menos un Dispositivo Portamuestras es capaz de transferir una pluralidad de Recipientes de Muestra (ID), incluyendo cada Recipiente al menos una Muestra (SP1) entre dicha pluralidad de Recursos (R1-Rn).

19. Método según la reivindicación 1, en el que dichas variables (RLS, RPS) usadas en dicho paso de controlar dichos Excitadores de Recurso (D1-Dn) por dicho Controlador de Proceso (PC) incluyen el Estado Lógico de Recurso (RLS), que expresa las diversas condiciones lógicas del Recurso asociado (R1-Rn), y el Estado Físico de Recurso (RPS) que expresa las diversas condiciones físicas del Recurso asociado (R1-Rn).

20. Método según la reivindicación 1, en el que el Controlador de Procesos (PC), según el estado de los diversos Recursos (R1-Rn), puede decidir automáticamente reprogramar en cualquier momento las actividades del Protocolo de Muestra (SP1).

21. Método según la reivindicación 3, en el que el Gestor de Datos (DM) según el estado de Resultados de Muestra y reglas de sistema experto enseñadas al Sistema, es capaz de generar peticiones automáticas para modificar Protocolos de Muestra (SP1) o pedir la intervención de Usuario para tratar casos especiales e imprevistos.

22. Método según la reivindicación 1, dedicado al control de una Célula de Trabajo de Diagnóstico *In Vitro* (IVD) que realiza tratamiento de muestras y análisis de prueba sobre muestras biológicas usando Instrumentos de Diagnóstico como Dispositivos de Ensayo de Muestra (11).

23. Método según la reivindicación 1 dedicado al control de un Sistema de Célula de Trabajo de Automatización de Laboratorio de Cribado de Alto Rendimiento (HTS) que realiza una gran cantidad de análisis de cribado de muestras biológicas (S1) en Dispositivos de Ensayo de Muestra (11) siguiendo protocolos diferentes con el fin de generar datos que sean usados para investigación farmacéutica.

24. Método según la reivindicación 1 dedicado al control de una Célula de Trabajo de Tecnología Ácido Nucleico (NAT), en tanto que Sistema de Célula de Trabajo de Automatización de Laboratorio NAT, que realiza tratamiento de muestras y análisis de prueba sobre ácidos nucleicos de muestras biológicas (S1) usando Instrumentos adecuados como Dispositivos de Ensayo de Muestra (11).

25. Método según la reivindicación 1 dedicado al control de un Sistema de Célula de Trabajo de Automatización de Laboratorio Biorrepositorio que se usa como un almacenamiento permanente de muestras biológicas (S1) que puedan mantenerse refrigeradas durante años y usarse bajo petición.

## ES 2 333 697 T3

26. Método según la reivindicación 1 dedicado al control de un Sistema de Control Industrial con la misma metodología que la descrita anteriormente, siempre que el Sistema de Control Industrial requiera una gestión de la operación multitarea, reactiva y basada en un sistema experto, y las Muestras (S1), los Recipientes de Muestra (ID), los Protocolos de Muestra (SP1) y los Resultados de Muestra estén representados por objetos y datos adecuados.

5

10

15

20

25

30

35

40

45

50

55

60

65

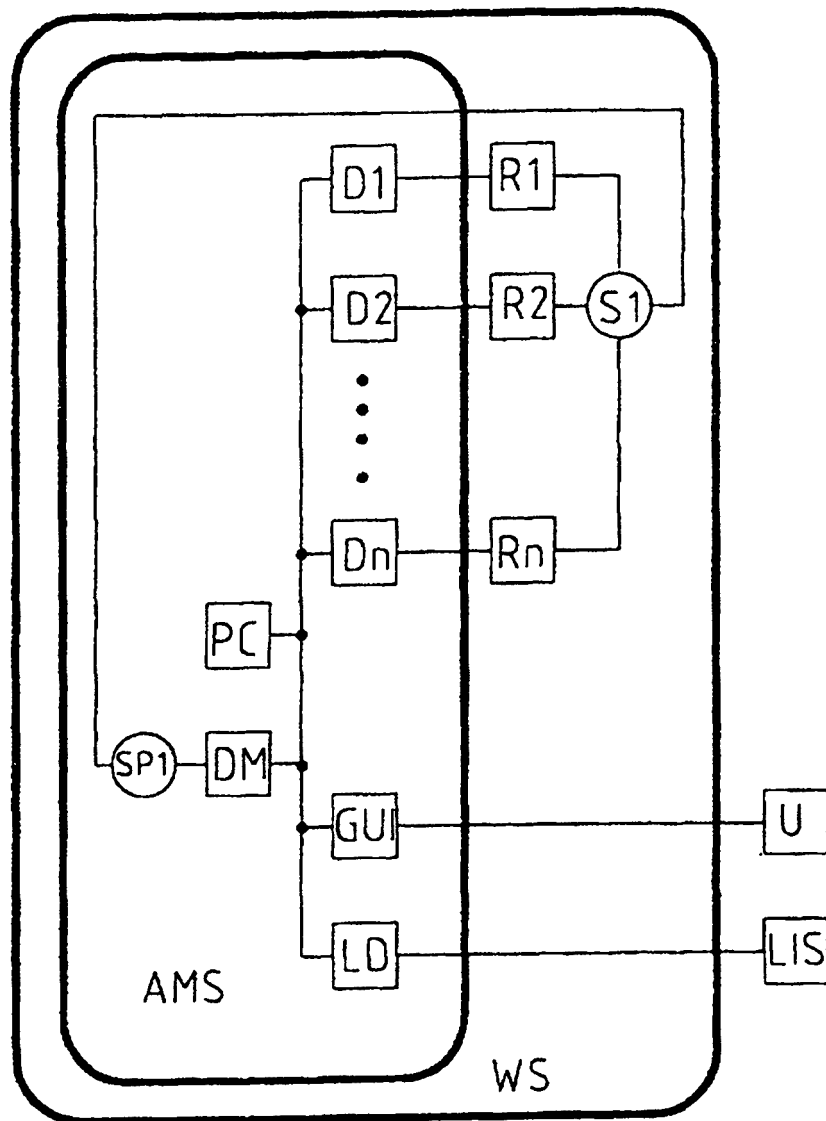
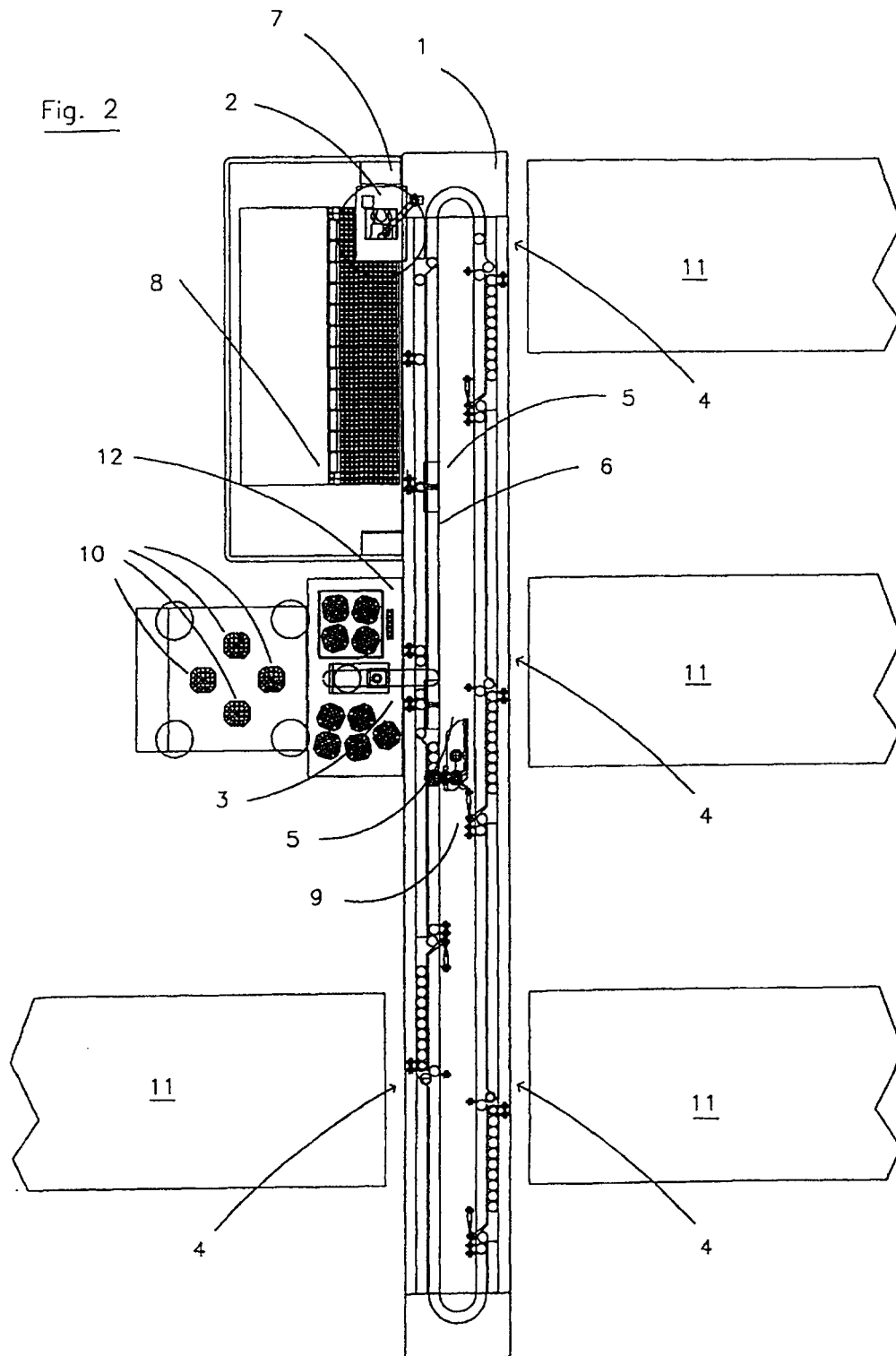


Fig.1

Fig. 2





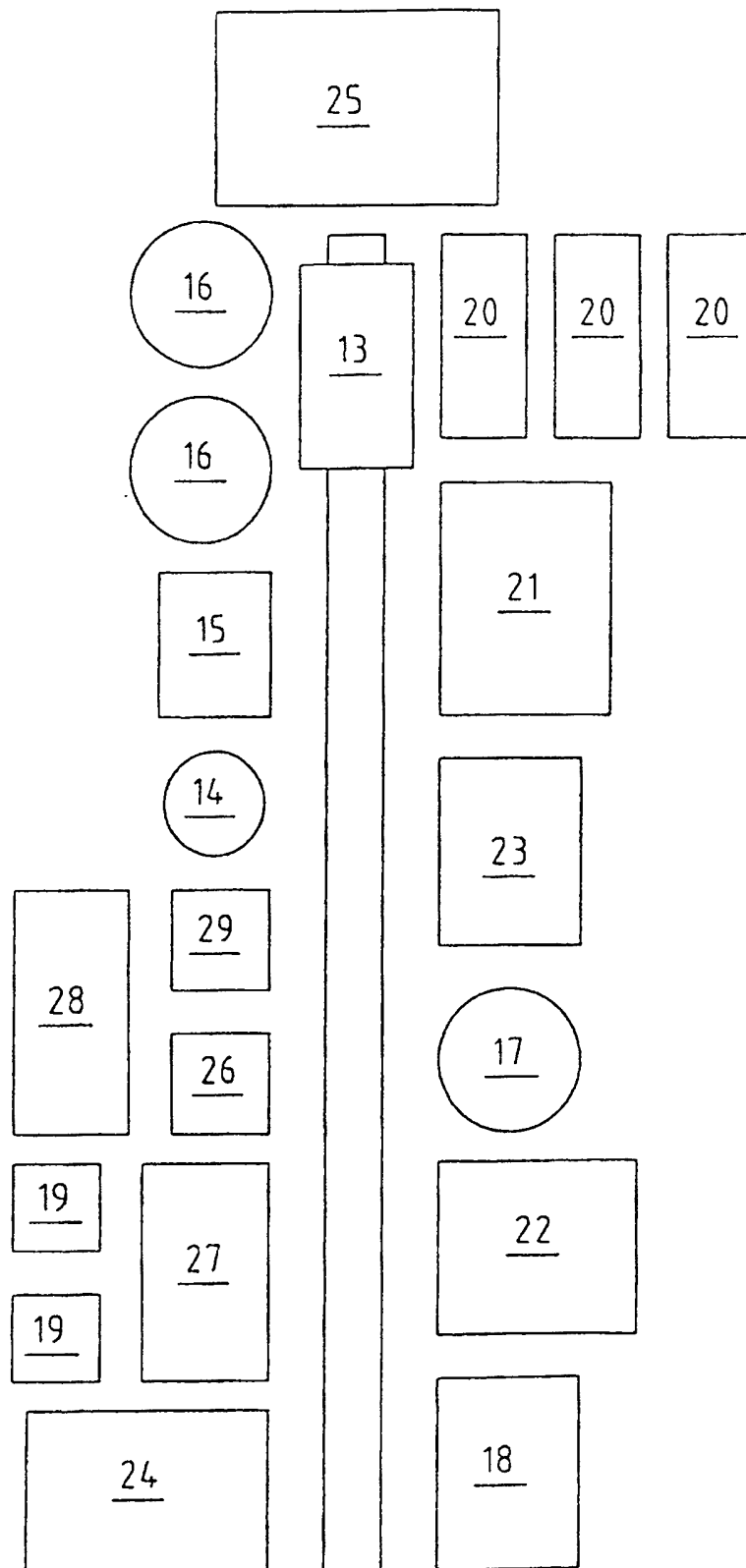


Fig.3

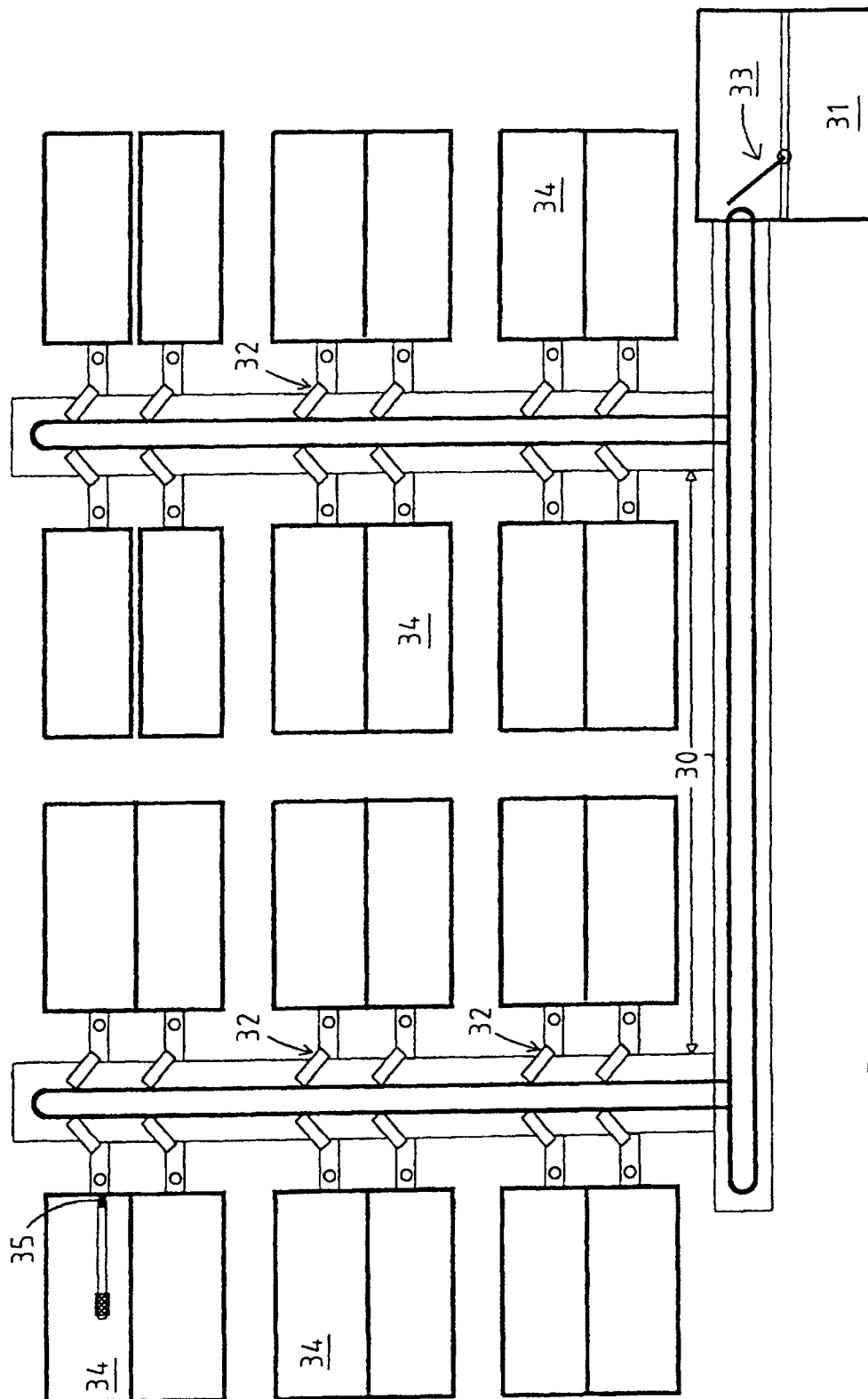


Fig. 4