

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 616 309**

51 Int. Cl.:

**G06F 9/54** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **17.02.2005 PCT/FR2005/000364**

87 Fecha y número de publicación internacional: **06.10.2005 WO05093570**

96 Fecha de presentación y número de la solicitud europea: **17.02.2005 E 05729304 (5)**

97 Fecha y número de publicación de la concesión europea: **07.12.2016 EP 1719054**

54 Título: **Dispositivo de memoria virtual compartida auto-administrada para administrar al menos un flujo de datos multi-pista**

30 Prioridad:

**25.02.2004 FR 0401889**  
**27.02.2004 US 548100 P**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:  
**12.06.2017**

73 Titular/es:

**EVS TOULOUSE (100.0%)**  
**6 rue Brindejonc des Moulinais, Bâtiment A, Parc de la Plaine**  
**31500 Toulouse, FR**

72 Inventor/es:

**FEVRIER, BENOÎT y**  
**MONESTIE, CHRISTOPHE**

74 Agente/Representante:

**IZQUIERDO BLANCO, María Alicia**

**ES 2 616 309 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).

**Dispositivo de memoria virtual compartida auto-administrada para administrar al menos un flujo de datos multi-pista****Descripción**

- 5 **[0001]** La invención se refiere a un dispositivo de memoria virtual compartida autoadministrada.
- 10 **[0002]** Una memoria virtual está formada por un área de memoria de acceso aleatorio electrónico asociado con el (los) microprocesador(es) en una placa base de un dispositivo. Esta memoria virtual se organiza en bloques de memoria que se pueden utilizar de forma dinámica por al menos un programa de aplicación y/o tareas o procesos en complemento de la área de memoria RAM electrónico utilizado por el sistema operativo y el BIOS.
- 15 **[0003]** Un programa de aplicación es un programa destinado a un usuario humano para llevar a cabo funciones (informáticas o de otro tipo) elegidas por él. Algunos programas aplicativos, sobre todo en el ámbito audiovisual, utilizan o generan flujos de datos, llamados flujos de pistas múltiples, que comprende una pluralidad de pistas de lectura y/o escritas y/o editadas en paralelo. Estos flujos de datos pueden requerir para su tratamiento un volumen de memoria grande, ser de muy diferentes tamaños, dependiendo de la aplicación, y requiere la sincronización entre las pistas.
- 20 **[0004]** Sin embargo, se siente la necesidad de tener un dispositivo capaz de gestionar dichos flujos de pistas múltiples en un número sustancial, en paralelo, sin saturación de la memoria RAM del dispositivo.
- 25 **[0005]** Por ejemplo, sería útil leer y/o transferir de un dispositivo a otro y/o modificar (filtrar, convertir, transcodificar, encapsular, extraer...) flujos audiovisuales de pistas múltiples, de cualquier formato como MPEG, MPEG2, DV, de audio, sin que sea necesario una configuración específica de la memoria virtual necesaria y sin riesgo de saturación o de bloqueo de esta memoria virtual. Otras aplicaciones también necesitan del tratamiento de flujo de datos en volúmenes de pistas múltiples paralelos. Por ejemplo, en la industria petrolífera se lleva a cabo muchos estudios sísmicos que forman flujos de datos que han de compararse a los archivos de plantilla.
- 30 **[0006]** Se conoce ya dispositivos informáticos que utilizan una memoria virtual compartida.
- 35 **[0007]** Para asignar memoria virtual a un programa de aplicación, se utiliza por ejemplo una función de modo que "malloc (tamaño)" en lenguaje C o "nuevo[tamaño]" en lenguaje C++, lo que crea un bloque de memoria virtual accesible sólo a través del proceso utilizando esta función. Bibliotecas de sistemas conocidos tales como "shmat" o "shmcreat" en el lenguaje C permiten compartir la misma área de memoria virtual por varios procesos, los cuales alimentan una dirección inicial conocida de memoria virtual. La sincronización se realiza entonces por semáforos o mecanismos de exclusión mutua ("mutex"), de modo que los diferentes procesos acceden a la memoria compartida con una dirección de desplazamiento de la dirección inicial. Pero estos mecanismos de sincronización asumen, para cada proceso, conocer y tener en cuenta las tareas realizadas por los diferentes procesos y formatos de datos utilizados en la memoria virtual. Además, la asignación de bloques de memoria es estática y rígida.
- 40 **[0008]** El documento EP-1 031 927 describe un método para la asignación de bloques de tareas de memoria compartida (o procesos), de identificar y asignar bloques de memoria disponibles para cada tarea a través de fichas y una tabla de asignación única. Este proceso, sin embargo, es independiente del programa de aplicación que no toma en cuenta las limitaciones específicas. Por lo tanto, no es especialmente adecuado para el tratamiento de múltiples flujos de datos, tales como flujos audiovisuales.
- 45 **[0009]** US 5.487.167 describe un sistema de gestión de flujo de datos en multitarea de tiempo real. En este dispositivo, deberá comunicar programas de aplicación multimedia a través de una interfaz de control de multimedia MCI mediante el envío de comandos para controlar dispositivos materiales en el entorno multimedia. Los controladores de dispositivos materiales pasan comandos a un subsistema de sincronización de flujo S/SS a través de una interfaz de flujo SPI programable a un gestor de flujo SSM.
- 50 **[0010]** Este dispositivo que sólo emplea una línea de búferes, no permite manejar flujos de pistas múltiples de cualquier formato de flujo con un importante número de pistas al mismo tiempo.
- 55 **[0011]** Por otra parte, el documento EP 0.694.847 describe un sistema de gestión de datos permitiendo que los dispositivos materiales realicen operaciones de flujo a través de un sistema de bus de forma independiente. El sistema central prepara de antemano todos los dispositivos periféricos o virtuales para el tratamiento de todo el flujo de datos, y después lanza el primer dispositivo periférico. La aplicación de los dispositivos periféricos sucesivos diferentes se lleva a cabo bajo el control de cada uno de estos dispositivos periféricos. En consecuencia, en este documento, la sincronización se lleva a cabo en en los propios dispositivos periféricos materiales. Una vez más, este dispositivo no se ocupa de flujos de pista múltiple de cualesquiera formatos con un gran número de pistas. La sincronización debe llevarse a cabo de manera previa, y no permite hacer cambios en tiempo real.
- 60 **[0012]** Además, el capítulo "3 Interprocess Communication", el libro "UNIX NETWORK PROGRAMMING" de W.
- 65

Richard Stevens, publicado en 1990, describe varias técnicas de comunicación entre procesos, existentes en el sistema operativo Unix. Específicamente se describen técnicas bien conocidas "Pipes" (sección 3.4), "Streams and Messages" (sección 3.6) y "Shared Memory" (sección 3.11).

5 **[0013]** No hay ningún dispositivo conocido que puede manejar una zona de memoria virtual compartida auto-administrada, y por tanto auto-adaptativa, lo que permite su uso dinámico para el tratamiento de datos de pistas múltiples de cualesquiera formatos no necesariamente previamente especificados, y en cualquier número.

10 **[0014]** La invención pretende, por tanto, resolver este problema general.

**[0015]** La invención tiene por objeto más concretamente proporcionar un dispositivo especialmente adecuado para el procesamiento de flujos de pistas múltiples audiovisuales.

15 **[0016]** La invención también se refiere más particularmente a la proporción de un dispositivo en el que se proporciona la gestión del área de memoria virtual auto-administrada directamente y de forma totalmente automática, no procesando los programas de aplicación los problemas de direccionamiento de memoria, la sincronización entre los procesos funcionales, y el procesamiento paralelo en varias partes del área de memoria virtual autoadministrada.

20 **[0017]** La invención también se refiere más particularmente a la proporción de un dispositivo para la realización de tratamientos en paralelo y simultáneamente diferentes (leer, escribir, transferencia, transformación, filtrado, transcodificación, compresión, descompresión, encapsulación, extracción...).

25 **[0018]** La invención también tiene por objeto proponer un dispositivo en el que la gestión del área de memoria virtual autoadministrada permite absorber las diferencias en la escritura o en la lectura de medios o periféricos.

**[0019]** Para ello, la invención se refiere a un dispositivo que comprende:

- 30
- medios de microprocesador(es) y memoria(s) RAM(s) capaz de realizar al menos un sistema operativo y al menos un programa de aplicación de procesamiento de datos,
  - al menos una memoria virtual, adaptada para utilizarse como una RAM de trabajo para al menos un programa de aplicación, que al menos está adaptado para procesar al menos un flujo de datos digital,
  - una pluralidad de procesos funcionales, estando adaptados dichos procesos de gestión de flujo para estar cargado en la memoria RAM y para ejecutar al menos una tarea en los datos de un flujo de pistas múltiples, a través de búferes de memoria RAM,
  - 35 - medios adaptados para conseguir la sincronización de la utilización sucesiva de búferes por el proceso de gestión del flujo,

40 Caracterizado porque:

a) comprende medios, estando dichos medios de configuración adaptados para configurar el dispositivo con:

- 45
- un área de memoria virtual, estando dicha memoria autoadministrada reservada y dedicada al tratamiento de flujos, comprendiendo dicho flujo de pistas múltiples una pluralidad de pistas leídas y/o escritas y/o procesadas en paralelo, comprendiendo esta memoria autoadministrada un área de administración dedicada a la administración de la memoria auto-administrada, y una zona útil para el tratamiento de datos,
  - un proceso funcional, estando dicho proceso controlador adaptado para poder cargarse en la memoria RAM, y definir y registrar en el área de administración, una pluralidad de líneas de memoria destinadas cada una a contener una lista de búferes, estando dichos búferes sincronizados de dicha zona útil de la memoria auto-administrada,
  - 50 • los procesos de gestión de flujo son adaptados para poder cargarse en memoria RAM y, con al menos una línea de memoria, crear y utilizar al menos un búfer sincronizado en esta línea de memoria, para ejecutar al menos una tarea en los datos de un flujo de pistas múltiples, y después liberar este búfer sincronizado,

55 b) el procesador de conmutación está adaptado para:

- \* Determinar, en base a las limitaciones de tratamiento predefinidas para cada flujo de pistas múltiples a tratar, una secuencia de uso de búferes sincronizados de líneas de la memoria de cada proceso de gestión de flujo, incluyendo dicho proceso activo el procesamiento de dicho flujo de pistas múltiples,
  - \* transmitir a cada proceso activo, la(s) línea(s) de memoria(s) en la que debe crear y/o utilizar búferes sincronizados,
- 60

c) comprende un módulo de gestión adaptado para realizar la sincronización de la utilización sucesiva de búferes sincronizados de cada línea de memoria por el proceso activo en función de la secuencia de uso determinada por el proceso de conmutación.

65

**[0020]** La invención permite tratar simultáneamente en paralelo (en la misma línea de tiempo) flujos de pistas

múltiples completamente diferentes, considerados hasta ahora como totalmente incompatibles entre sí, por ejemplo, una pista de formato de vídeo de alta definición sin compresión, y una pista de vídeo con un formato de alta relación de compresión, tal como MPEG2. Esto se logra a través de la memoria autoadministrada, los procesos de control, los procesos de gestión de flujo, y las diferentes líneas de la memoria que permiten la sincronización de datos y la programación de los diversos procesos de flujo de trabajo que pueden ejecutar muchas tareas diferentes en los datos.

**[0021]** Ventajosamente y según la invención, el módulo de administración está vinculado (a la compilación y ejecución) para el proceso de conmutación y cada proceso de gestión de flujo, y reúne las funciones comunes de gestión de la memoria autoadministrada. Según la invención, el módulo de administración se forma ventajosamente de una biblioteca de funciones comunes relacionadas (en el sentido informático) por los procesos de programación orientada a objetos, tales como una biblioteca enlazada dinámicamente en lenguaje C ++.

**[0022]** Además de permitir a los procesos de gestión de flujos para el uso de búferes sincronizados de una línea de memoria, el módulo de administración incluye otras funciones comunes. En particular, ventajosamente y según la invención, el módulo de administración está adaptado para determinar, después de la liberación por un proceso de gestión de flujo de un tampón sincronizado, los procesos de gestión de flujo subsiguientes definidos en secuencia de uso, y, en su defecto, para eliminar el tampón sincronizado. La eliminación del búfer sincronizado ayuda a rendir el espacio de memoria correspondiente disponible de nuevo para otros tratamientos. Tenga en cuenta que los búferes sincronizados creados en la misma línea de memoria no se corresponden necesariamente con fragmentos contiguos de la superficie útil de la memoria autoadministrada.

**[0023]** Ventajosamente y según la invención, cada proceso de gestión de flujo está diseñado para procesar los datos en cada momento con un tampón sincronizado único de una línea de la memoria y para liberar este búfer sincronizado de tratamiento, el uso de diferentes búferes sincronizados a una memoria de línea por cada proceso de gestión de flujo efectuándose sucesivamente, uno tras otro. Por lo tanto, el uso del espacio de la memoria útil se optimiza, varios procesos de flujo de trabajo pueden estar activos al mismo tiempo en diferentes búferes sincronizados de forma perfectamente sincronizada.

**[0024]** Además, ventajosamente y según la invención, el módulo de administración incluye las siguientes características:

- creación del área de administración y de la superficie útil de la memoria autoadministrada,
- inicialización de una línea de memoria con un nivel máximo de llenado de la superficie útil para esta línea de memoria
- creación de una memoria intermedia sincronizada en una línea de memoria,
- liberación de un búfer sincronizado,
- acceso a una memoria intermedia sincronizada por un proceso activo,
- determinación posterior de los procesos activos en secuencia utilizando un búfer sincronizado de una línea de la memoria, después de la liberación de esta última por el proceso activo anterior.

**[0025]** Por otra parte, ventajosamente y según la invención el proceso de control está adaptado para definir, para cada pista de cada flujo de pistas múltiples a tratar, al menos una línea de memoria dedicada al tratamiento de esta pista. Además, ventajosamente y según la invención, el proceso de control está adaptado para definir, para cada pista de cada flujo de pistas múltiples a tratar y para cada uno de los datos del proceso de procesamiento de flujo de trabajo de esta pista, al menos una línea de memoria proporcionando datos a procesar por el proceso de gestión de flujo y/o al menos una línea de memoria de destino que recibe los datos procesados por el proceso de gestión de flujo.

**[0026]** Por otra parte, ventajosamente y según la invención, el proceso de conmutación está adaptado para definir una y sólo una secuencia de uso para todos los búferes sincronizados de una misma línea de memoria.

**[0027]** Por otra parte, ventajosamente y según la invención, el proceso de conmutación está adaptado para transmitir la secuencia de uso de cada línea de memoria a los primeros procesos de gestión de flujo para ser activo en una línea de memoria intermedia sincronizada, este proceso de gestión de flujo siendo el creador de este búfer sincronizado y definiendo y registrando en la zona de administración de datos identificando este búfer sincronizado y asociándolo a la línea de memoria y la secuencia de utilización.

**[0028]** Por otra parte, ventajosamente y según la invención, el proceso de conmutación está adaptado para calcular, en función de la naturaleza de cada flujo de pistas múltiples a ser tratado, un tamaño máximo de la superficie útil de la memoria autoadministrada que puede darse a cada línea de memoria. Por lo tanto, el uso de la superficie útil de la memoria autoadministrada se optimiza mediante el proceso de conmutación de acuerdo con las necesidades de cada pista de cada flujo de pistas múltiples sin riesgo de bloqueo. Este tamaño máximo se define ventajosamente como un factor de llenado, por ejemplo, un porcentaje de la superficie útil de la memoria autoadministrada. Se registra en el área de administración.

**[0029]** Por otra parte, ventajosamente y según la invención, los procesos de gestión de flujo están separados, cada

uno con al menos una tarea propia. Al contrario nada impide prever que múltiples versiones idénticas de uno o varios procesos de gestión de flujo estén activas simultáneamente, notablemente para realizar tareas similares en paralelo. Los procesos de gestión de flujo no comunican directamente entre sí.

5 **[0030]** En una realización de la invención ventajosa para muchas aplicaciones del dispositivo, este último comprende por lo menos un programa aplicativo, dicho módulo de lanzador, adaptado para cargar en la memoria RAM los diversos procesos y módulos permitiendo que la configuración y el funcionamiento de la memoria autoadministrada incluyendo:

- 10
- el proceso de conmutación
  - cada proceso de gestión de flujo susceptible de utilizarse para el tratamiento de flujo de pistas múltiples,
  - el módulo de administración,
  - un módulo de ventanas dinámico en una pantalla de visualización del dispositivo, adaptado para ser capaz de
- 15 formar una interfaz de hombre/máquina que permite a un usuario definir cada flujo de pistas múltiples para hacer frente a las fuentes de datos de diversos orígenes. En esta variante, se carga en la memoria del módulo lanzador bajo las órdenes de un usuario humano, lo cual permite configurar el dispositivo con la memoria autoadministrada, los procesos de gestión de flujo, y los proceso de conmutación de acuerdo con la invención. El módulo de ventanas es una aplicación de programa que permite al usuario utilizar el dispositivo de la invención configurado de este modo fácil y cómodamente, especialmente por órdenes de tipo cliquear/deslizar para
- 20 archivos de datos fuente destinados a constituir flujos de pista múltiple.

**[0031]** En otra variante, que también se puede combinar con la anterior, el dispositivo de la invención puede configurarse por adelantado, por ejemplo cuando se inicia, y se opera de manera más o menos automática, incluyendo dentro de un complejo arquitectónico (por ejemplo, en una red), o el mando de un programa de aplicación

25 de control automático para llevar a cabo determinadas tareas de acuerdo con los eventos predeterminados (por ejemplo, montar, convertir y registrar una unidad de registro de varias pistas leídas de diversas fuentes externas). En esta variante, el módulo de ventanas dinámico no es esencial.

**[0032]** Ventajosamente y según la invención, el dispositivo incluye:

- 30
- al menos un proceso de gestión de flujo, siendo dicho proceso de carga, capaz de escribir datos, especialmente de datos de una unidad de lectura, en la superficie útil de la memoria autoadministrada,
  - al menos un proceso de gestión de flujo, siendo dicho proceso adaptado para leer datos, en particular datos destinados a una unidad de recepción, tal como un dispositivo de registro, una pantalla de visualización...-
- 35 desde la superficie útil de la memoria autoadministrada.

**[0033]** Una unidad de lectura puede ser un dispositivo periférico según la invención, otro dispositivo o cualquier dispositivo capaz de transmitir datos de destino para el dispositivo de la invención.

40 **[0034]** De manera similar, una unidad de recepción puede ser un periférico del dispositivo de acuerdo con la invención, por ejemplo, un dispositivo de registro, otro dispositivo, o cualquier dispositivo capaz de recibir datos transmitidos por el dispositivo de acuerdo con la invención.

45 **[0035]** Por otra parte, ventajosamente y según la invención, los procesos de gestión de flujo están adaptados para cargarse en una zona de memoria viva distinta de la memoria autoadministrada. Del mismo modo, el proceso de conmutación está adaptado para cargarse en un área de memoria separada de la memoria autoadministrada.

**[0036]** Además, ventajosamente y según la invención, el proceso de conmutación está adaptado para, en una primera etapa de análisis, analizar las características de cada flujo de pistas múltiples a tratarse y de las limitaciones de tratamiento de cada flujo de pistas múltiples, a fin de definir los datos que representan las líneas de memoria, y los datos que representan cada secuencia de utilización de búferes sincronizados de cada línea de memoria para el tratamiento de este flujo de pistas múltiples, y en una segunda etapa posterior de tratamiento, iniciar el procesamiento de flujos de pistas múltiples de acuerdo con dichos datos previamente definidos durante la etapa de análisis.

50

**[0037]** Las restricciones predefinidas específicas a cada corriente pueden estar predefinidas mediante la programación de los procesos de conmutación y/o registro de parámetros en una memoria de masa de dispositivo y/o datos, llamados metadatos, asociados al flujo -notablemente en un encabezamiento y leídos por el proceso de conmutación y/o proporcionados por el programa de aplicación. Estas restricciones incluyen, por ejemplo, el número de pistas; los datos de sincronización entre las pistas; la duración de cada pista; la velocidad de transferencia de datos desde/hacia una unidad de lectura/una unidad receptora; el formato de los datos en cada pista; el método de compresión de datos; la naturaleza del tratamiento a llevarse a cabo en cada pista

60

**[0038]** Ventajosamente, en un dispositivo según la invención, el tamaño de la memoria autoadministrada se define por los medios de configuración a un valor fijo predeterminado -en particular entre 20% y 80% de la de la memoria virtual, típicamente del orden de 128 megabytes a 15 gigabytes con memorias actuales. Del mismo modo,

65

5 ventajosamente y según la invención, el tamaño de la zona de administración se define por los medios de configuración a un valor fijo predeterminado - notablemente, un valor fijo absoluto, por ejemplo, aproximadamente 1 megabyte. El tamaño del área de administración es mucho menor que la de la superficie útil de la memoria autoadministrada. Además, ventajosamente y según la invención, la memoria auto-administrada se define por el proceso de conmutación cuando se carga en la memoria RAM.

10 **[0039]** Por otra parte, ventajosamente y según la invención, en la administración de cada área componente, dicho elemento de administración contiene una dirección de un elemento anterior y una dirección de un elemento siguiente.

15 **[0040]** Ventajosamente y de acuerdo con la invención, el área de administración incluye en la etapa de procesamiento:

- una lista de fragmentos disponibles de memoria autoadministrada,
- una lista de líneas de memoria activas (es decir definidas por el proceso de conmutación)
- una lista de búferes sincronizados con la superficie útil de la memoria autoadministrada para utilizarse por las líneas de la memoria activa, y una lista de los diferentes procesos activos que deben utilizar estos búferes sincronizados de memoria autoadministrada.

20 **[0041]** Ventajosamente y según la invención, los medios de configuración están adaptados para permitir el procesamiento de los flujos de pistas múltiples que son flujos audiovisuales, incluyendo pistas de las que el formato se seleccionado de:

- formatos de televisión de alta definición (TVHD)
- formatos de televisión de definición estándar (TVSD)
- formatos de cine digital
- formatos de vídeo comprimidos (MPEG2, MPEG4, DV ...),
- formatos de audio sin comprimir,
- formatos de audio comprimido,
- formatos de encapsulación de pistas múltiples (Quicktime®, AVI® ...)
- formatos de imágenes,
- formatos de datos audiovisuales en bruto.

35 **[0042]** La invención se extiende a un medio de registro capaz de ser leído por una unidad de lectura asociada con un dispositivo de procesamiento digital, comprendiendo este soporte registro un programa de ordenador adaptado para ser capaz de formar medios para la configuración de un dispositivo de acuerdo con la invención, cuando se instala y se ejecuta en dicho dispositivo.

40 **[0043]** La invención se extiende a un método de procesamiento de flujo de pistas múltiples usando un dispositivo de acuerdo con la invención.

**[0044]** La invención también se extiende a un dispositivo; a un medio de registro, y un método caracterizado en combinación por la totalidad o parte de las características mencionadas anteriormente o a continuación.

45 **[0045]** Otros objetos, características y ventajas de la invención resultarán evidentes de la lectura de la siguiente descripción dada a modo de ejemplo, que se refiere a las figuras adjuntas en las que:

- La figura 1 es un diagrama que muestra la organización de una memoria de un dispositivo de acuerdo con la invención,
- La figura 2 es un diagrama de un entorno ejemplar de dispositivos que pueden explotarse ventajosamente con un dispositivo de acuerdo con la invención,
- La figura 3 es un diagrama de una ventana ejemplar de una interfaz hombre/máquina activada por un módulo lanzador de un dispositivo de acuerdo con la invención,
- La figura 4 es un diagrama que ilustra un ejemplo de la arquitectura informática funcional de un dispositivo de acuerdo con la invención,
- La figura 5 es un diagrama de flujo que ilustra un ejemplo de algoritmo de un proceso de conmutación de un dispositivo de acuerdo con la invención,
- La figura 6 es un diagrama de flujo que ilustra un ejemplo de algoritmo de un proceso de gestión de flujo de un dispositivo de acuerdo con la invención,
- La figura 7 es un diagrama que ilustra la arquitectura general de consultas y declaraciones de los procesos de gestión de flujo de un dispositivo de acuerdo con la invención,
- La figura 8 es un diagrama que ilustra un diagrama de temporización de ejemplo de dos flujos de pistas múltiples, debiendo tratarse secuencialmente,
- La figura 9 es un diagrama que ilustra la organización del funcionamiento de la memoria autoadministrada de un dispositivo de la invención para el procesamiento de los flujos de la figura 8.

**[0046]** Un dispositivo 1 de la invención es un dispositivo de procesamiento de datos digitales que puede, del punto de vista de su arquitectura estructural, presentar todas las posibles formas de realización conocidas. Puede ser un microordenador que incluye una placa base de microprocesador y la memoria (RAM) asociada; uno o más bus de conexión de bandas de memoria y/o dispositivos (incluyendo una interfaz de hombre/máquina que comprende un teclado, un dispositivo de puntuación, una pantalla de visualización); y dispositivos de almacenamiento masivo, como un disco duro y/o lectores/registradores de medios de almacenamiento masivo extraíbles. También puede tratarse de una arquitectura de red que comprende varias máquinas y/o partes de máquinas relacionadas entre sí. En cualquier caso, el dispositivo según la invención está adaptado para formar al menos una CPU para ejecutar al menos un sistema operativo (incluyendo el tipo de LINUX®, UNIX®, WINDOWS®, ) y uno o más programas de aplicación de procesamiento de datos.

**[0047]** El dispositivo de la invención comprende además al menos una memoria virtual 2 adaptada para ser utilizada como memoria de trabajo para programas de aplicación.

**[0048]** Una memoria virtual es una zona RAM de gestión centralizada por al menos uno de los módulos del sistema operativo y se puede poner a disposición de al menos un programa de aplicación para que pueda desempeñar tareas específicas.

**[0049]** En la Figura 1, se muestra un ejemplo de la memoria virtual 2. Esta memoria virtual 2 puede ser una parte de memoria de acceso aleatorio electrónico asociado con un microprocesador en una placa base del ordenador. Cabe señalar que la invención se aplica igualmente bien a la aplicación de un intercambio de este tipo con otras formas de RAM, por ejemplo, una RAM asociada a través de un bus a un microprocesador. La tecnología de realización de esta RAM es en realidad de importancia en el contexto de la invención, ya que las capacidades y velocidades de acceso y otras características de la configuración de la memoria material que implementan esta RAM es compatible con las aplicaciones que se realizan, en particular en términos de tiempo de procesamiento. En particular, se observa que para el tratamiento de flujos de pistas múltiples audiovisuales, en particular en vista de su lectura, los tiempos de procesamiento en RAM deben ser lo suficientemente débiles para evitar la interrupción de la lectura de flujos audiovisuales o todo fenómeno de cambio abrupto.

**[0050]** En un dispositivo de acuerdo con la invención, una parte predeterminada de la memoria virtual 2 puede reservarse y dedicarse al tratamiento de flujos de pistas múltiples. Esta zona específica, dicha memoria autoadministrada 3, puede definirse de antemano, por ejemplo mediante la configuración del usuario, ya sea bajo una forma de valor fijo, ya sea como un valor correspondiente a un porcentaje de la memoria virtual 2 o RAM 1 total.

**[0051]** En el ejemplo mostrado en la figura 1, la memoria virtual 2 tiene una capacidad de 512 megabytes, y la memoria autoadministrada 3 tiene una capacidad de 256 megabytes.

**[0052]** Además, la memoria autoadministrada 3 comprende dos áreas separadas, a saber, una zona llamada zona de administración 4, dedicada a la administración de la memoria autoadministrada 3 y en la que los datos permiten la administración (organización, sincronización, desfragmentación...) de la memoria autoadministrada 3 se pueden registrar; y un área llamada la zona útil 5, sirviendo como memoria de trabajo para el procesamiento de flujo de datos digitales, dichos flujos de pistas múltiples, comprendiendo una pluralidad de pistas leídas y/o escritas y/o procesadas en paralelo. El tamaño de la zona útil 5 es mucho mayor que el de la zona de administración. Los datos de administración no son datos a procesar por el dispositivo, tales como datos de flujos de pistas múltiples).

**[0053]** Las pistas de los flujos de pistas múltiples, tales como una pista de vídeo, una pista de audio, ... En un flujo de pistas múltiples en la entrada y/o salida, las pistas pueden transmitirse en un formato multiplexado en una línea única y/o comprimido, por ejemplo, MPEG2, DV .... Sin embargo, el tratamiento de un flujo de pistas múltiples puede comprender al menos una tarea o una serie de tareas (lectura, registro, conversión, transcodificación, filtrado, compresión, descompresión, encapsulación, extracción de un formato encapsulado,...) para llevarse a cabo por separado en varias pistas (de las que el número puede ser muy importante).

**[0054]** Como se muestra en la figura 2, el dispositivo 1 de la invención se puede utilizar para el tratamiento de flujo de datos hacia y/o desde diversos dispositivos de formatos normalmente incompatibles entre ellos. En el ejemplo mostrado en la figura 2, se proporcionan cámaras (por ejemplo, typecinema digital, vídeo digital o videocámara digital,...) que puede proporcionar datos de vídeo a través de interfaces de tipo HDSOI, SDI, "Firewire" (denominado también "I-Link" o IEEE1394), o Ethernet. En el ejemplo, se ha representado una cámara de cine digital 6a y una cámara de vídeo digital 6b. También se espera una cámara de vídeo 7 u otro dispositivo de lector/escritor que puede adquirir y/o proporcionar datos de vídeo a través de interfaces de tipo HDSOI, SDI, "Firewire" o una red local, por ejemplo de Ethernet. También se puede proporcionar un dispositivo de almacenamiento masivo, como una unidad de disco 9, por ejemplo de RAID proporcionando y/o recibiendo datos de vídeo, una pantalla de visualización VGA o un monitor de vídeo 10 que recibe datos de vídeo por una interfaz de tipo HDSOI, SDI o analógica, y una conexión a una red 11 a través de una interfaz de tipo Ethernet o red de almacenamiento compartido "SAN". En este ejemplo, el dispositivo 1 según la invención es un servidor de vídeo.

**[0055]** Por supuesto, esta ilustración es sólo un ejemplo y cualquier otro enlace que proporciona o recibe datos de

pistas múltiples puede proporcionar, por ejemplo, un receptor de emisión de televisión (por ondas hertzianas, satélite o cable, ...).

5 **[0056]** El dispositivo de la invención incluye al menos un programa de aplicación, estando dicho módulo lanzador adaptado para cargar en memoria RAM y lanzar la ejecución de medios de configuración del dispositivo de acuerdo con la invención. Este módulo lanzador lanza en particular un módulo de ventanas dinámico que realiza, en una pantalla de visualización del dispositivo, una ventana 26 tal como la mostrada en la figura 3, adaptado para ser capaz de formar una interfaz hombre/máquina 26 que permite que un usuario defina cada flujo de pistas múltiples para hacer frente a las fuentes de datos de diversos orígenes. En el ejemplo mostrado en la figura 3, la ventana 26 incluye una barra de título 12, una barra de menú 13, una ventana de visualización de vídeo 14 asociada a una zona 15 de mando/reproducción de visualización de información de lectura/registro (retroceso, rebobinar, reproducir, pausar, detener, avance rápido, avance, contador, ...). También se prevé una ventana 16 de navegación con un área 17 de archivos del árbol de archivos y un área 18 de visualización de miniaturas o iconos que representan archivos.

15 **[0057]** La ventana 26 incluye además una ventana de montaje 19 que comprende una zona de comandos o acciones 20, una zona 21 de representación cronográfica de flujos de pistas múltiples a tratar utilizada en el caso de una edición, una zona 22 de herramientas de filtrado que pueden accionarse por el usuario y una zona 23 suplementaria de visualización/entrada de comandos específicos. También se puede proporcionar ventajosamente una zona (que no se muestra en el ejemplo) para la gestión de la adquisición de flujos de pistas múltiples.

20 **[0058]** Con tal ventana 26, el usuario puede, por ejemplo, simplemente seleccionar un archivo en la zona de navegación 16 y moverlo a la zona de visualización del cronógrafo 21, que tendrá el efecto de tener en cuenta el flujo de pistas múltiples asociado a este archivo en su tratamiento por la memoria autoadministrada 3. Gracias a la invención, se puede particularmente asociar en el cronograma y montar simultáneamente de modo sincrónico diferentes flujos de pistas múltiples de formatos completamente diferentes, normalmente incompatibles, e incluyendo los formatos de alta definición, formatos de definición estándar, formatos comprimidos o no comprimidos, formatos de encapsulación (Quicktime®, AVI®, ...).

25 **[0059]** La figura 4 muestra un ejemplo de la arquitectura informática correspondiente a los medios de configuración de la memoria autoadministrada 3 en un dispositivo según la invención. Esta arquitectura incluye la interfaz hombre/máquina 26 que se muestra en la figura 3.

30 **[0060]** Esta interfaz hombre/máquina 26 comunica con un proceso funcional, estando cargado dicho proceso de conmutación 27 en la RAM 1 y, preferiblemente, se lleva a cabo en la misma máquina en la que la memoria autoadministrada 3 se gestiona. Este proceso de conmutación 27 es un proceso funcional, es decir, un proceso de tipo servidor de bajo nivel en el sistema operativo que no es directamente visible ni accesible para el usuario.

35 **[0061]** Los medios de configuración de la invención incluyen además otros procesos funcionales, llamados procesos de gestión de flujos, de los que el número no está limitado, estando cada uno de los cuales adaptado para cargarse en la memoria RAM 1 y realizar al menos una tarea en datos de flujo de pistas múltiples. Muchos procesos de gestión de flujo pueden desarrollarse de acuerdo con las funciones que se llevan a cabo para examinar la aplicación en el dispositivo de la invención. Preferiblemente, cada proceso de gestión de flujo está diseñado para realizar una sola tarea específica o una serie de tareas que corresponden a una función de procesamiento único en una pista de un flujo de pistas múltiples, por ejemplo, una lectura, un registro, una transferencia hacia un dispositivo como una pantalla de visualización, un filtrado, una transcodificación, una compresión, una descompresión, una encapsulación, una extracción de un formato encapsulado.

40 **[0062]** Es de notar, sin embargo, que la interfaz/máquina 26 comunica directamente y únicamente con el proceso de conmutación 27, y de alguna manera con el proceso de gestión de flujo. Por lo tanto, independientemente de la función requerida por el programa de aplicación controlado por la interfaz hombre/máquina 26, esta función se dirige necesariamente al proceso de conmutación 27 y se procesa y se analiza por este último.

45 **[0063]** En el ejemplo no limitativo mostrado en la Figura 4, se proporciona como proceso de gestión de flujo, un proceso 28 de carga de datos en la zona útil 5 de la memoria autoadministrada 3, un proceso 29 de leídos en la zona útil 5 de la memoria autoadministrada 3, un proceso 30 de filtro de datos leídos en la zona útil 5 de la memoria autoadministrada 3, y un proceso 31 de control de dispositivos de visualización. Es de notar que los diferentes procesos de gestión de flujo no comunican directamente entre sí. Comunican directamente sólo con el proceso de conmutación 27. La comunicación entre la ventana dinámica de interfaz hombre/máquina 26 y el proceso de conmutación 27 se realiza por dos enlaces dedicados de comunicación (por ejemplo, de tipo "SOCKET"), es decir, un enlace de comunicación de tipo de comando/absolución CMD/ACK 24 y un enlace de supervisión 25 que permite transmitir los artículos, los códigos temporales y los errores eventuales entre el proceso de conmutación 27 y el módulo de ventana dinámica 26.

50 **[0064]** Cada proceso de gestión de flujo 28, 29, 30, 31 está configurado por el proceso de conmutación 27. Por otra parte, los distintos procesos de gestión de flujo 28 a 31 sólo intercambiarán datos correspondientes al contenido de los flujos de pistas múltiples, a través de la zona útil 5 de la memoria autoadministrada 3.



**[0065]** Cada proceso de gestión de flujo 28 al 31 se conecta con el proceso de conmutación 27 mediante dos enlaces de comunicación (por ejemplo, de tipo "SOCKET"), es decir, un enlace de comando/adquisición (CMD/ACK) y un enlace de supervisión de errores eventuales relevados por el proceso de gestión de flujo 28 al 31 correspondiente.

5 **[0066]** También se prevé una biblioteca 32 común de operaciones informáticas que los procesos de gestión de flujo 28 a 31, y el proceso de conmutación 27 utilizan para ejecutar comandos o tareas habituales que utilizan los scripts/lecturas en la memoria. Esta biblioteca 32 forma un módulo, dicho módulo de administración 32, vinculado mediante programación a cada proceso 27, 28 a 31. Se observa que la memoria autoadministrada 3 con los  
10 procesos de gestión de flujo 28 a 31 y el proceso de conmutación 27 y el módulo de administración 32 pueden funcionar de modo completamente autónomo, sin requerir la ejecución de una ventana dinámica, o más generalmente una interfaz gráfica de usuario, tal como 26.

15 **[0067]** Los diversos procesos de gestión de flujo 28 a 31 son preferentemente similares en su funcionamiento y su arquitectura. Esta arquitectura general común se muestra por ejemplo en la figura 7.

**[0068]** El servicio REQ\_ISALIVE permite al módulo de interfaz hombre/máquina 26 saber si los diversos procesos de gestión de flujo se cargan y se activan o no.

20 **[0069]** El servicio REQ\_INIT realiza la inicialización del proceso de gestión de flujo y de lugar al espacio "INIT" que se muestra en la figura 7. Es a la recepción de este servicio que todos los procesos de gestión de flujo se configurarían antes de lanzar una acción en los datos a procesar. Cada proceso de gestión de flujo también tiene igualmente un servicio REQ\_CHANGECONF que permite al proceso de conmutación 27 cambiar la configuración específica del proceso de gestión de flujo.

25 **[0070]** REQ\_PROCESS se refiere genéricamente a todas las acciones realizadas en cada flujo de pistas múltiples a través de un proceso de gestión de flujo que se encuentra a continuación en el estado "PROCESS" que se muestra en la figura 7.

30 **[0071]** La solicitud REQ\_STOP sitúa el proceso de gestión de flujo en el estado inicializado. La solicitud REQ\_RESET permite al proceso de gestión de flujo pasar a un estado estable "READY".

**[0072]** La biblioteca 32 que forma el módulo de administración incluye diferentes funciones comunes que pueden utilizarse por el proceso, incluyendo:

- 35
- una función de creación de la zona de administración 4 y de la zona útil 5 de la memoria auto-administrada, consistiendo en reservar las zonas de RAM correspondientes con sus correspondientes direcciones de memoria,
  - 40 - una función de inicialización en la zona de administración 4 de una línea de memoria (dependiendo del proceso de conmutación 27) con una tasa máxima de relleno (calculada por el proceso de conmutación 27) de la zona útil 5 por esta línea de memoria,
  - 45 - una función de creación de un búfer sincronizado en una línea de memoria (a través de un proceso de gestión de flujo creativo de un búfer sincronizado) con su tamaño de memoria, número, dirección, secuencia de utilización (lista de los diferentes procesos de gestión de flujo activo debiendo acceder sucesivamente a este búfer, como se determina por el proceso de conmutación 27) y un campo de memorización de su estado actual,
  - una función de acceso a un búfer sincronizado de una línea de memoria por un proceso de gestión de flujo activo,
  - 50 - una función de liberación un búfer sincronizado después de la utilización por un proceso de gestión de flujo, permitiendo rendir este búfer sincronizado disponible para el proceso de gestión de flujo posterior, o eliminar este búfer sincronizado, si se trata del último proceso de gestión de flujo activo, debiendo actuar en este búfer sincronizado,
  - 55 - una función de sincronización que consiste en determinar, después de la liberación de un búfer sincronizado por un proceso activo, cuál es el proceso activo subsiguiente, debiendo intervenir en este búfer sincronizado, a partir de la secuencia de uso de este búfer sincronizado.

**[0073]** Esta biblioteca 32 puede formar clases programadas por ejemplo en lenguaje C ++.

60 **[0074]** El módulo lanzador carga en memoria RAM, además de la interfaz hombre/máquina 26, el proceso de conmutación 27, cada proceso de gestión de flujo 28 a 31, y la biblioteca 32 (módulo de administración), estos procesos 27 a 31, entre ellos y la biblioteca 32 estando vinculados.

**[0075]** La figura 5 muestra un diagrama de flujo de la operación del proceso de conmutación 27.

65 **[0076]** El paso 50 representa la inicialización del proceso de conmutación 27 y su carga en la memoria, por ejemplo, como consecuencia de la puesta en marcha del módulo lanzador. En la etapa posterior 51, el proceso de

conmutación 27 creó la memoria autoadministrada 3. Se crea una conexión con cada proceso de gestión de flujo de 28 a 31. Cuando el usuario ejecuta un comando en la ventana 26, el proceso de conmutación 27 recibe, de hecho, una secuencia de flujo de pistas múltiples bajo una forma de una cronograma convencionalmente designado "lista de ediciones". Esta recepción está representada por el paso 52 en la figura 5.

5 **[0077]** A continuación, se ejecuta, en la etapa 53, el análisis de esta lista de ediciones y se abre un bucle en las diferentes ediciones, es decir, en las diferentes pistas que tratar.

10 **[0078]** El proceso de conmutación 27 registra (etapa 54), en la zona de administración de la memoria autoadministrada 3, una o más líneas de la memoria, de las que generalmente al menos una línea de memoria fuente y/o al menos una línea de memoria de destino. El proceso de conmutación 27 crea al menos una línea de memoria para cada pista (fuente o destino) que va a tratarse. Es de notar que la misma línea de memoria puede actuar como fuente y como destino para los datos a procesar.

15 **[0079]** Por ejemplo, se cree una memoria de línea para recibir pista de vídeo de origen y una línea de memoria de fuente para recibir una pista de audio que debe tratarse en paralelo y/o una o más línea(s) de memoria de destino para recibir el resultado del tratamiento con proceso de gestión de flujo activo.

20 **[0080]** Esta etapa de análisis 53 permite al proceso de conmutación 27 establecer el número de línea de memoria, el tamaño máximo de la memoria en cada línea de memoria, y la secuencia de uso de memoria intermedia sincronizada en cada línea de memoria, y esto en función de restricciones predefinidas por el programa de aplicación que provee el flujo de pistas múltiples a tratar al proceso de conmutación 27. En el ejemplo, este programa de aplicación consiste en la interfaz hombre/máquina 26. El paso 53 de análisis se lanza especialmente cuando el usuario sitúa un archivo en la zona de visualización de cronógrafo 21 utilizando el dispositivo señalador, que tiene el efecto de proporcionar el proceso de conmutación 27 de las consultas y parámetros correspondientes al flujo de pistas múltiples a tratar. Tras la recepción de dicha solicitud para la edición de un flujo de pistas múltiples, el proceso de conmutación 27 determina en primer lugar si el tratamiento a realizarse consiste en una edición o, por el contrario, en una adquisición.

30 Caso de una edición:

**[0081]** En función de los parámetros pasados por el programa de aplicación 26, el proceso de conmutación 27 determina si la edición se debe filtrar y/o se visualiza y/o se registra y determina la secuencia de utilización de búferes sincronizados para cada línea de memoria a crearse, lo que corresponde en realidad a la secuencia de intervención de los diversos procesos de gestión de flujo que deben estar activos en cada pista de flujo de pistas múltiples.

**[0082]** Los parámetros contemplados son:

- 40
- Un archivo de "lista de ediciones" (edl)
  - Un archivo "lista de archivos" (archivo edl)
  - Un archivo "lista de filtro" (filtro edl).

45 **[0083]** Para cada edición, es decir, cada pista de la lista de ediciones, el proceso de conmutación 27 determina el formato de los datos, es decir, notablemente el audio/vídeo estándar en el que se registra, y determina el tamaño de edición (es decir, de la pista) en comparación con el tamaño máximo de las diferentes ediciones de la secuencia, para determinar el porcentaje de la zona útil 5 de RAM que puede afectar a cada línea de memoria correspondiente a cada pista.

50 **[0084]** El proceso de conmutación 27 crea a continuación una línea de memoria fuente para cada pista de audio y una línea de memoria fuente para cada pista de vídeo, y calcula y dispone los parámetros destinados al proceso de carga, es decir, la identificación de las diferentes líneas de memoria utilizadas y las secuencias de utilización correspondientes.

55 **[0085]** El proceso de conmutación 27 determina a continuación si cada edición, es decir, cada pista debe filtrarse o no. Si es así, el proceso de conmutación 27 extrae del archivo de filtros enviados en parámetros el o los filtros relativos a la pista a filtrar, y comprueba el número de entrada/salida, y crea líneas de memoria que hay salida para el (los) filtro(s) a aplicar. El proceso de conmutación 27 prepara a continuación, prepara y dispone los parámetros destinados al proceso de filtrado, incluyendo la identificación de las diferentes líneas de memoria de audio y de vídeo fuente y diferentes líneas de memoria de destino de la pista.

60 **[0086]** El proceso de conmutación 27 a continuación, examina si la pista se debe mostrar o no. Si es así, y si también se filtró la pista, el proceso de conmutación 27 utiliza las líneas de memoria de destino identificadas en el proceso de selección. Si la pista tiene que visualizarse pero sin filtrado, el proceso de conmutación 27 envía las líneas de memoria fuente de audio y vídeo creadas previamente al proceso de visualización. Es de notar que en este caso, las líneas de la memoria fuente también sirven como líneas de memoria de destino. El proceso de

conmutación 27 calcula y dispone los parámetros para el proceso de visualización (líneas de memoria y secuencias de uso).

5 **[0087]** El proceso de conmutación 27 determina entonces que el flujo de la edición debe registrarse o no. En caso afirmativo, y si el flujo ha sido objeto de un filtro, el proceso de conmutación 27 utiliza las líneas de memoria de destino del proceso de filtrado. Si el flujo debe registrarse sin filtrar, el proceso de conmutación 27 envía las líneas de memoria fuente audio y vídeo al proceso de registro. Una vez más, a continuación, calcula y dispone los parámetros destinados al proceso de registro (líneas de memoria y secuencias de uso).

10 Caso de una adquisición:

**[0088]** En función de los parámetros recibidos desde el programa de aplicación, el proceso de conmutación 27 determina si la adquisición se muestra y se registra y se calcula la secuencia de uso de búferes sincronizados correspondientes.

15 **[0089]** Los parámetros son:

- Un archivo "lista de ediciones de adquisición" (adquisición edl)
- Un archivo de "lista de archivos" (archivo edl).

20 **[0090]** Para cada edición de la lista de edición de transmisión para el proceso de conmutación 27, este último determina el formato de datos (audio/vídeo estándar), y calcula el tamaño de la edición de adquisición en relación con el tamaño máximo de pistas diferentes de cada flujo de pistas múltiples de la secuencia a adquirir, con el fin de determinar el porcentaje de zonas útiles de la memoria autoadministrada utilizada por cada línea.

25 **[0091]** En el caso en el que el proceso de conmutación 27 detecta la presencia de pistas de audio en la lista de ediciones de adquisición, se crea una línea de memoria de adquisición para cada pista de audio correspondiente. Del mismo modo, cuando el proceso de conmutación 27 detecta la presencia de pistas de vídeo en la lista de las ediciones de adquisición, se crea una línea de memoria de adquisición para cada pista de vídeo correspondiente.

30 **[0092]** El proceso de conmutación 27 determina y dispone a continuación los parámetros para el proceso de adquisición, incluyendo la identificación de las diferentes líneas de la memoria y su secuencia de uso.

35 **[0093]** El proceso de conmutación 27 determina entonces si la edición se debe mostrar o no. Si es así, se prepara los parámetros correspondientes (líneas de memoria de adquisición) para el proceso de visualización. A continuación, el proceso de conmutación 27 determina y dispone los parámetros para el proceso de registro.

40 **[0094]** El paso de análisis 53 y el paso 54 para crear las líneas de la memoria descritas anteriormente son solamente ejemplos no limitativos y muchas otras formas de análisis pueden proporcionarse, dependiendo de las aplicaciones del dispositivo la invención.

45 **[0095]** El siguiente paso 55 consiste en abrir un bucle a través de los distintos procesos de gestión del flujo 28 a 31 cargados en la memoria. Para cada proceso, se realiza en el paso 56 una prueba para determinar si el proceso de gestión de flujo del 28 al 31 puede tratarse por la pista a procesar. Si es así, el proceso de conmutación 27 envía las líneas de memoria y la información de sincronización (secuencia de uso) correspondientes al proceso de gestión de flujo implicado, en el paso 57. Si no es así, el proceso vuelve de nuevo al siguiente proceso de gestión de flujo. Después de la etapa 57, se ejecuta una prueba 58 para terminar el bucle, es decir para saber si se trata del último proceso de gestión de flujo de la secuencia de uso. Si no es así, se pasa al siguiente proceso de gestión de flujo. Si esto es el caso, se realiza una prueba 59 para saber si la pista procesada era la última. Si no era el caso, se vuelve al paso 53 para volver a ejecutar los pasos 54 a 58 en la siguiente pista. Si se trata de la última pista, las primeras fases de análisis es la secuencia de flujo de pistas múltiples a tratarse ha terminado, y se pasa a una fase posterior de ejecución compendiando primero una etapa 60 de inicialización del proceso de conmutador 27, y recibir en el paso 61 un comando de parte del usuario, es decir el programa de aplicación controlado por el usuario (interfaz hombre/máquina 26), y se envía a continuación, en la etapa 62, una acción a cada proceso de gestión de flujo 28 a 31 para activar la operación de colocación de estos procesos de gestión de flujo, y de este modo sincronizar los unos con los otros.

60 **[0096]** La figura 6 muestra el diagrama de flujo del funcionamiento de un proceso de gestión de flujo controlado por el proceso de conmutación 27. En las figuras 5 y 6, los vínculos entre los dos gráficos están representados por las letras A y B.

65 **[0097]** El paso 63 corresponde al inicio del proceso de gestión del flujo, seguido del paso 64 del compromiso de este proceso de gestión de flujos a la memoria autoadministrada 3, es decir, el estado "READY" representado en la figura 7.

**[0098]** En la etapa 65 posterior, el proceso de gestión de flujo puede recibir una lista de edición (una secuencia flujos

de pistas múltiples) que se transmitió por el proceso de conmutación 27 al final de la etapa 57 de este proceso de conmutación 27. Si el proceso de gestión de flujo recibe entonces una acción durante la etapa 66 de la parte de proceso de conmutación 27 (después de la etapa 62 del envío de la acción de este proceso de conmutación 27), el proceso de gestión de flujos abre una etapa 67 de abertura de bucle que permite navegar por cada pista de la lista correspondiente a una línea de memoria. Después de esta etapa 67, se ejecuta una prueba 68 para determinar si la acción requerida y la función que realiza corresponde o no a la creación de uno o más búfer(es) sincronizado(s) en la zona útil 5 de la memoria autoadministrada 3. En caso negativo, el proceso de gestión de flujo ejecuta una etapa 69 de espera sincronizada en un búfer sincronizado de la memoria autoadministrada 3. A continuación, se ejecuta una prueba 70 para determinar si este búfer sincronizado está disponible o no en la línea de memoria fuente.

**[0099]** Hay que señalar que el búfer sincronizado en el que el proceso de gestión de flujo se posiciona se determina por adelantado en la línea de memoria por el proceso de conmutación 27, y este dato es conocido en el proceso de gestión de flujo.

**[0100]** Mientras que el búfer sincronizado no está disponible según lo determinado por la prueba 70, el proceso de gestión de flujo vuelve a la etapa 69 de espera. Cuando el búfer sincronizado se hace disponible, el proceso de gestión de flujo ejecuta la etapa 71 posterior de tratamiento de los datos en este búfer sincronizado.

**[0101]** Además, si la prueba 68 determina que el proceso de gestión de flujo debe crear un búfer sincronizado, se ejecuta a continuación en la etapa 72 de creación de este búfer sincronizado, entonces el proceso pasa a la etapa 71 de procesamiento de datos en el búfer sincronizado así creado.

**[0102]** El proceso de gestión de flujo crea un búfer sincronizado cuando es el primer proceso de gestión de flujo a intervenir en una línea de memoria a procesarse. Después de realizar el procesamiento de datos de la etapa 71, el proceso de gestión de flujo libera el búfer sincronizado en la etapa 73 para que esté disponible para el proceso de gestión de flujo, debiendo intervenir en este búfer sincronizado. Después de esta etapa 73 de la liberación del búfer sincronizado, el proceso de gestión de flujo termina el bucle de los diversos flujos de la lista gracias a la prueba 74 que, después de procesar todas las pistas en la lista, se ejecuta una etapa 75 para el procesamiento de esta lista de ediciones.

**[0103]** La biblioteca común 32 permite definir diferentes elementos de administración que en realidad son listas, al contener cada elemento de administración una referencia a los elementos siguientes y anteriores. **[0104]** Un elemento de administración de tipo fragmento de memoria se define por su primer desajuste con respecto a la dirección de base de la zona útil 5 de la memoria autoadministrada 3, por su desajuste final con respecto a la dirección de base de esta zona útil 5 de la memoria autoadministrada 3 y por su tamaño.

**[0105]** Un elemento de tipo línea de memoria o "TRACK" se define por un identificador, una lista de búferes sincronizados asociados a él y por su tamaño.

**[0106]** Un elemento de administración de tipo de búfer de memoria o "BUFFER" se define más por su identificador, su dirección de memoria (desajuste con respecto a la dirección de inicio de la zona útil 5 de la memoria), su tamaño, una secuencia de uso (o tabla de transición), y una variable que representa el estado.

**[0107]** La zona de administración 4 se divide en búferes de administración o se definirán los elementos de administración (líneas de memoria, búferes de memoria o fragmentos de memoria libres).

**[0108]** En la solicitud de creación de una línea de la memoria o un búfer de memoria, el módulo de administración 32 convierte un elemento de administración libre en un elemento de tipo línea de memoria o de tipo búfer de memoria.

**[0109]** En el caso de una conversión en elemento de tipo línea de memoria, el proceso de conmutación 27 define la secuencia de utilización en función de las limitaciones de procesamiento de flujos de pistas múltiples, incluso por los diferentes procesos de gestión de flujo que serán necesarios para este tratamiento.

**[0110]** En el caso de una conversión de un elemento en el búfer de memoria, el módulo de administración 32 recupera un fragmento de la zona útil 5 libre a partir de la lista de fragmentos de memoria libre, y esto en función del tamaño deseado por este búfer. Tres casos son posibles:

1) Si el fragmento de memoria libre tiene un tamaño inferior al tamaño del búfer de memoria deseada, el módulo de administración recuperará el siguiente fragmento de memoria libre y repetirá la prueba. Si no tiene ningún fragmento de memoria libre, se informa de un error en el proceso de gestión de flujo creador (el proceso realiza varias solicitudes sucesivas en espera de la liberación de una zona por otro proceso).

2) Si el fragmento de memoria libre tiene el mismo tamaño que el tamaño de búfer deseado se le asigna el valor de inicio del fragmento al valor de dirección del búfer y elimina el fragmento de búfer de la lista de fragmentos de memoria libre en la zona de administración 4.

3) Si, al contrario, el fragmento de memoria tiene un tamaño superior al tamaño de búfer deseado, afecta al valor de inicio del fragmento en el valor de dirección en el búfer y se reduce el fragmento de búfer del tamaño

asignado al búfer.

**[0111]** A continuación, se inicializa el estado del búfer de memoria para el valor correspondiente a un número que identifica el proceso de gestión de flujo activo en este búfer.

**[0112]** Tras la liberación de un búfer sincronizado por un proceso de gestión de flujo, el estado evolucionará dependiendo de la secuencia de uso definida para este búfer sincronizado (ya sea el retorno al estado inicial, ya sea el paso al siguiente estado, este último provocando la disponibilidad del búfer sincronizado para el proceso de gestión de flujo posterior o la eliminación del búfer en el caso en el que se trataría del último proceso de gestión de flujo en la secuencia de uso).

**[0113]** Si la liberación del búfer sincronizado implica su eliminación, el módulo de administración 32 convierte este último en fragmento de búfer, y se añade al final de la lista de fragmentos de búfer en la zona de administración 4. A continuación, se desfragmenta la zona útil 5 de memoria mediante la comprobación de si el búfer sincronizado liberado es o no adyacente a uno o dos fragmentos de memoria libre.

**[0114]** Cuando la solicitud de acceso de un búfer sincronizado por un proceso de gestión de flujo, las siguientes verificaciones se llevan a cabo antes de dar acceso al búfer sincronizado:

- ¿Existe la línea de memoria?
- ¿Existe el búfer sincronizado?
- ¿Está el búfer sincronizado disponible para el proceso de gestión de flujo (verificación de su estado actual)?

**[0115]** Si el búfer sincronizado está disponible para este proceso de gestión de flujo, la dirección del búfer sincronizado se devuelve al proceso de gestión de flujo que puede utilizarlo. De lo contrario, un código indica el estado del búfer sincronizado.

**[0116]** El proceso de gestión de flujo tiene la oportunidad de solicitar el acceso a un búfer de memoria de manera asíncrona (útil para un proceso de verificación), es decir, se puede recuperar un búfer de memoria independientemente del estado de este último.

**[0117]** Hay que señalar que un proceso de gestión de memoria autoadministrada implementado en un dispositivo según la invención se puede implementar en cualquier lenguaje informático que apoya la asignación dinámica de memoria y la gestión de memoria compartida. También hay que señalar que los datos descriptivos contenidos en un búfer de memoria pueden variar. Por ejemplo, los identificadores (búfer de memoria o línea de memoria) que consisten en números enteros pueden ser sustituidos con cadenas de caracteres. Las figuras 8 y 9 ilustran un ejemplo de aplicación particular. Es de notar que este ejemplo no corresponde estrictamente al ejemplo de implementación mostrado en la figura 4. En la figura 8 se muestra un ejemplo de un cronograma de dos flujos audiovisuales sucesivos, a saber, una secuencia en formato MPEG2 que dura 3 segundos y que comprende una pista de vídeo V1 y una pista de audio A1, seguida de una secuencia DV, que dura 3 segundos, comprendiendo además una pista de vídeo V1 y una pista de audio A1.

**[0118]** La figura 9 muestra esquemáticamente el tratamiento de estos flujos por la memoria autoadministrada 3 según la invención. La lista de edición 90 se proporciona al proceso de conmutación 27. En el ejemplo representado, se prevén los siguientes procesos de gestión de flujo: un proceso PGF1 de carga, un proceso PGF4 de visualización de vídeo, un proceso PGF5 de audio, un proceso PGF2 de descompresión MPEG 2, y un proceso PFG3 de descompresión DV.

**[0119]** En el ejemplo mostrado, cada proceso de gestión de flujo utiliza al menos una línea de fuente de memoria y al menos una línea de memoria de destino. La línea de memoria LM0 como fuente significa que el proceso de gestión de flujo es creador de búfer sincronizado (y primer interviniente de la secuencia de utilización) y no recibe datos de una línea de memoria (por ejemplo, proceso de carga). Del mismo modo, la línea de memoria LM0 en cuanto destino significa que el proceso de gestión de flujo es el último interviniente de la secuencia de uso de un búfer sincronizado.

**[0120]** El proceso de conmutación 27 define en la zona de administración 4 utilizando la biblioteca 32, seis líneas de memoria LM1, LM2, LM3, LM4, LM5, LM6 con, para cada una de ellas, su tamaño máximo y su dirección inicial. Por ejemplo, el tamaño máximo para la línea de memoria LM1 y para la línea de memoria LM4 es de 10% de la zona útil 5 de la memoria autoadministrada 3. Y el tamaño máximo de las líneas de memoria LM2, LM3, LM5 y LM6 es de 20% de la zona útil 4 de la memoria autoadministrada 3.

**[0121]** Cada proceso de gestión de flujo creador de búferes sincronizados en una línea de memoria de destino también define búferes sincronizados como TM1, TM2..., y registra en la zona de administración 4 su tamaño, su número, su dirección, su secuencia de uso, y su estado actual, todo ello utilizando la biblioteca 32.

**[0122]** Se muestra en la zona útil 5 de la memoria autoadministrada 3 las diferentes líneas de memoria LM1 a LM6 y la naturaleza de los datos que se tratan. También se muestra los datos de sincronización que se dirigen por el proceso de conmutación 27 a cada uno de los procesos de gestión de flujo que identifican las líneas de memoria y de destino fuentes y los datos de sincronización (secuencias de uso y/o identificación de búferes sincronizados).

**[0123]** En el ejemplo mostrado, el proceso de conmutación 27 proporciona el primer proceso de gestión de flujo (PGF1), que es el proceso de carga, los datos 95 que incluyen el número de la línea de memoria fuente, que, en el ejemplo, es LM0, y el número de la línea de memoria de destino, así como la secuencia de utilización de búferes sincronizados de esta línea de memoria destino. Para la primera lista de ediciones (secuencia MPEG2), la línea de memoria de destino es LM1 y la secuencia de uso es 1,2, lo que significa que los procesos de gestión de flujo PGF1 y PGF2, sucesivamente, se intervendrán sucesivamente sobre los datos de la línea de memoria LM1 para el tratamiento de este flujo. Para la segunda lista de edición (secuencia DV), la línea de memoria fuente es LM0, línea de memoria de destino es LM4 y la secuencia de uso es 1,3, lo que significa que los procesos PGF1 y PG F3 intervendrán sucesivamente.

**[0124]** El proceso PGF1 de carga es la creación de búferes sincronizados ya que debe cargar los datos de la línea de memoria fuente en la zona útil 5 de la memoria. Así pues, el proceso PGF1 de carga crea los búferes sincronizados necesarios. Típicamente, en el ejemplo mostrado, treinta búferes sincronizados por segundo de flujo se deben crear. De este modo, el proceso PGF1 de carga utiliza sucesivamente noventa búferes sincronizados TM1, TM2, TM90 para la secuencia MPEG2 en la línea de memoria de destino LM1. Para ello, se crea un primer búfer sincronizado TM1, se cargan los datos de la secuencia MPEG2, liberando a continuación este búfer TM1. El módulo de administración 32, permite al proceso de gestión de flujo activo subsiguiente de intervenir, es decir, el proceso PGF2 que realiza la descompresión MPEG2. El proceso de carga repite sucesivamente estas operaciones en noventa búferes sincronizados TM1 a TM90.

**[0125]** Anteriormente, el proceso de conmutación 27 había proporcionado datos 96 al proceso PGF2 de gestión de flujo posterior. Por lo tanto, este proceso PGF2 sabe que debe intervenir en una línea de memoria fuente LM1 y en las líneas de la memoria de destino LM2 con la secuencia de uso 2, 4 (PGF2 a continuación PGF4) y LM3 con la secuencia de uso 2, 5 (PGF2 entonces PGF5), para los búferes sincronizados a partir de búfer sincronizado TM1 a búfer sincronizado TM90. Por lo tanto, este ejemplo muestra que, gracias a la invención, varios procesos de gestión de flujo se ejecutan simultáneamente en diferentes búferes sincronizados, y, de este modo, se sincronizan perfectamente.

**[0126]** El proceso PGF2 puede utilizar la línea de memoria LM2 para la pista de vídeo de flujo MPEG2 y la línea de memoria LM3 para la pista de audio de flujo MPEG2. Se cargan y se liberan sucesivamente los búferes sincronizados de estas dos líneas de memoria con los datos apropiados, como se describe anteriormente para el proceso PGF1. El módulo de administración 32 permite a continuación el uso de búferes sincronizados por el proceso de gestión de flujo PGF4 de visualización de vídeo de la línea de memoria LM2 a medida de la liberalización sucesiva de estos búferes sincronizados. Esto hace que sea posible, de la misma manera, el uso sucesivo de los búferes de la línea de memoria LM3 por el proceso PGF5 de audio.

**[0127]** De nuevo, el proceso de conmutación 27 había proporcionado previamente los datos 97 y, respectivamente, 98, a estos procesos PGF4, PGF5. De esta manera, estos dos procesos saben que las líneas de memoria fuente LM2 y LM3 deben procesarse, y que la línea de memoria de destino es LM0, lo que significa que estos procesos PGF4, PGF5 son los últimos intervinientes en la secuencia de utilización correspondiente al flujo MPEG2.

**[0128]** El dispositivo de la invención permite así en el ejemplo dado de leer simultáneamente, a partir de un flujo MPEG2, una pista de audio y una pista de vídeo de un modo perfectamente sincronizado.

**[0129]** Por otra parte, el proceso PFG1 carga datos en memoria mientras los procesos PGF2 y PGF3 leen y descomprimen los datos de la memoria. De hecho, una ventaja de la invención es permitir que varios procesos puedan utilizar la escritura y/o lectura de memoria simultáneamente.

**[0130]** El mismo tipo de operación se lleva a cabo para el tratamiento del segundo flujo en el formato DV. Progresivamente, y en función de su carga en los búferes sincronizados de la línea de memoria LM4 sucesivamente, la secuencia de uso permite la activación del proceso de gestión de flujo PGF3 de descompresión DV. Este último recibe los datos de cada tampón sincronizado de la línea de memoria LM4 al formato DV y descomprime y carga los datos descomprimidos en búferes sincronizados sucesivamente de dos líneas de memoria LM5 con la secuencia de uso 3, 4 (PGF3 a continuación PGF4) y LM6 con la secuencia de uso 3, 5 (PGF3 a continuación PGF5) vídeo, respectivamente audio. El proceso de conmutación 27 había comunicado los datos 99 al proceso de gestión de flujo PGF3 para indicar la identificación de las líneas de memoria y destino fuente, las secuencias de uso, así como los números de búferes sincronizados de inicio a fin (TM91 y TM180).

**[0131]** Hay que señalar que, con un dispositivo de acuerdo con la invención, los búferes sincronizados TM1 a TM90 o TM91 a TM180 creados en la misma línea de memoria no se corresponden necesariamente con espacios contiguos en la zona útil 5 de la memoria autoadministrada. Resulta una gestión mucho más flexible y eficaz de esta

zona útil 5, cuyas capacidades se mejoran en relación a los dispositivos anteriores, en los que dicho uso de búferes disjuntos para el mismo proceso y/o para el tratamiento de un mismo flujo de datos no es posible. Sin embargo, las líneas de memoria representan una abstracción de compromiso y posteriormente búferes sincronizados.

5 **[0132]** Como se muestra, los diferentes procesos de gestión de flujo reciben del proceso de conmutación 27, la integridad de los datos de sincronización y no comunican entre sí. Se sincronizan a partir de estos datos, y por el módulo de administración 32. Sólo los procesos de gestión de flujo PGF1, PGF2, PGF3 creadores de búferes sincronizados en las líneas de la memoria reciben las secuencias de uso correspondientes, y las registra en la zona de administración 4 con las líneas de memoria y el identificador de búferes sincronizados correspondientes.

10 **[0133]** Evidentemente, el ejemplo que representa figuras 8 y 9 no es limitativo y se ofrecen muchas otras opciones. Por ejemplo, dos secuencias de vídeo en el formato sin comprimir, se pueden leer simultáneamente en la misma pantalla, divididas en dos partes distintas por el dispositivo según la invención. Por lo tanto, es suficiente para proporcionar una pista de tipo por ejemplo F1 que realiza un filtrado que consiste en insertar una imagen en la otra y en aplicar un proceso de gestión de flujo para esta inserción simultáneamente en las dos secuencias leídas en paralelo. El proceso de conmutación 27 puede definir tres líneas de memoria, es decir, una línea de memoria para cada una de las secuencias de vídeo a leer, y una tercera línea de memoria de destino para recibir el resultado de la inserción que se proporcionará al proceso de visualización de vídeo. Muchos otros ejemplos son posibles.

20

**Reivindicaciones**

5           1. Dispositivo que comprende:

- medios con microprocesador(es) y RAM(s), aptos para ejecutar al menos un sistema operativo y al menos un programa de aplicación de procesamiento de datos,
- al menos una RAM virtual, que es adecuada para ser utilizada como RAM de trabajo para al menos un programa de aplicación, de la cual al menos una es adecuada para procesar al menos un flujo de datos digitales,
- múltiples procesos funcionales, llamados procesos de gestión de flujo (28-31; PGF1 - PGF5), que son adecuados para cargarse en RAM y ejecutar al menos una tarea en los datos de un flujo de pistas múltiples a través de memorias intermedias RAM,
- medios adecuados para sincronizar el uso sucesivo de amortiguadores por los procesos de gestión del flujo (28-31, PGF1-PGF5),

**caracterizado porque:**

20 a) este dispositivo es adecuado para ser configurado con:

- una zona de memoria virtual, denominada memoria autoadministrada (3), reservada y dedicada al tratamiento de flujos, denominada flujos de pistas múltiples, que comprenden pistas múltiples que se leen y/o se escriben y/o se procesan en paralelo, comprendiendo esta memoria autoadministrada una zona de administración (4) que está dedicada a la administración de la memoria autoadministrada, y una zona útil (5) para procesar los datos,
- un proceso funcional denominado proceso de conmutación (27), que es adecuado para ser cargado en la RAM, y para definir y registrar en la zona de administración (4) múltiples líneas de memoria que están destinadas a contener una lista de búferes, llamados búferes sincronizados, de dicha zona útil (5) de la memoria autoadministrada,
- los procesos de gestión de flujo (28-31; PGF1- PGF5) son adecuados para cargarse en RAM y, con al menos una línea de memoria, para crear y/o utilizar al menos un búfer sincronizado en esta línea de memoria, para ejecutar al menos una tarea en los datos de un flujo de pistas múltiples, y después para liberar este búfer sincronizado,

35 b) siendo adecuado el proceso de conmutación (27) para:

- \* determinar, en función de las restricciones de procesamiento predefinidas para cada flujo de pistas múltiples a procesar, una secuencia para utilizar los búferes sincronizados de las líneas de memoria por cada proceso de gestión de flujo, denominado proceso activo, que está implicado en el procesamiento de dicho flujo de pistas múltiples,
- \* transmitir a cada proceso activo la(s) línea(s) de memoria en la que debe crear y/o utilizar memorias intermedias sincronizadas,

45 c) el dispositivo que incluye un módulo de administración (32), que es adecuado para sincronizar el uso sucesivo de las memorias intermedias sincronizadas de cada línea de memoria por los procesos activos, en función de la secuencia de uso que determina el proceso de conmutación (27).

50           2. Un dispositivo de acuerdo con la reivindicación 1, **caracterizado porque** el módulo de administración (32) está unido al proceso de conmutación (27) y a cada proceso de gestión de flujo, y combina las funciones de gestión comunes de la memoria de autoadministración (3).

55           3. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 ó 2, **caracterizado porque** el módulo de administración (32) es adecuado para determinar, cuando un búfer sincronizado es liberado por un proceso de gestión de flujo, el proceso subsiguiente de gestión de flujo se define en la secuencia de uso, y si no se define para deshacer el búfer sincronizado.

60           4. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 3, **caracterizado porque** cada proceso de gestión de flujo (28-31; PGF1-PGF5) es adecuado para procesar los datos en cada instante con un solo búfer sincronizado de una línea de memoria, y después para liberar este búfer sincronizado al final del procesamiento, utilizándose los diversos búferes sincronizados de una línea de memoria sucesivamente por cada proceso de gestión de flujo, uno tras otro, de tal manera que varios procesos de gestión de flujo pueden estar activos simultáneamente en diferentes búferes sincronizados.

65           5. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 4, **caracterizado porque** el módulo de administración (32) es una biblioteca de funciones que incluye las siguientes funciones:

- crear la zona de administración (4) y la zona útil (5) de la memoria autoadministrada (3),



- inicializar una línea de memoria con una velocidad máxima de llenado dla zona útil (5) para esta línea de memoria,
  - crear un búfer sincronizado en una línea de memoria,
  - liberar un búfer sincronizado,
  - 5 - acceso a un búfer sincronizado mediante un proceso activo,
  - determinar el proceso activo subsiguiente en la secuencia de uso de un búfer sincronizado de una línea de memoria, después de que el búfer sincronizado se haya liberado por el proceso activo anterior.
6. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 5, caracterizado porque el proceso de conmutación (27) es adecuado para definir, para cada pista de cada flujo de pistas múltiples a procesarse, al menos una línea de memoria que se dedica al procesamiento de esta pista.
7. Un dispositivo de acuerdo con la reivindicación 6, **caracterizado porque** el proceso de conmutación (27) es adecuado para definir, para cada pista de cada flujo de pistas múltiples a procesarse y para cada proceso de gestión de flujo que procesa los datos de esta pista, al menos una línea de memoria fuente que suministra datos a procesarse por el proceso de gestión de flujo y/o al menos una línea de memoria de destino que recibe los datos que ha procesado el proceso de gestión de flujo.
8. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 7, **caracterizado porque** el proceso de conmutación (27) es adecuado para definir una y sólo una secuencia de uso para todos los búferes sincronizados de la misma línea de memoria.
9. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 8, **caracterizado porque** el proceso de conmutación (27) es adecuado para transmitir la secuencia de uso de cada línea de memoria al primer proceso de gestión de flujo que debe ser activo en un búfer sincronizado de esta línea de memoria, siendo este proceso de gestión de flujo el creador de este búfer sincronizado, y definiendo y registrando, en la zona de administración (4), datos que identifican este búfer sincronizado y lo asocian con la línea de memoria y secuencia de uso.
10. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 9, **caracterizado porque** el proceso de conmutación (27) es adecuado para calcular, en función de la naturaleza de cada flujo de pistas múltiples a procesarse, un tamaño máximo dla zona útil (5) de la memoria autoadministrada (3) que se puede asignar a cada línea de memoria.
11. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 10, **caracterizado porque** los procesos de gestión de flujo (28-31, PGF1-PGF5) son distintos y cada uno de ellos lleva a cabo al menos una tarea que le pertenece.
12. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 11, **caracterizado porque** incluye al menos un programa de aplicación, denominado módulo de lanzador, que es adecuado para cargar en RAM los diversos procesos y módulos que realizan la configuración y funcionamiento de la memoria autoadministrada, incluyendo:
- el proceso de conmutación (27),
  - cada proceso de gestión de flujo que es susceptible de utilizarse para procesar flujos de múltiples pistas,
  - el módulo de administración (32),
  - un módulo (26) de ventanilla dinámica en una pantalla de visualización del dispositivo, apto para formar una interfaz hombre/máquina que permite al usuario definir cada flujo de pistas múltiples para procesarse a partir de datos de origen de varios orígenes.
13. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 12, **caracterizado porque** comprende:
- al menos un proceso de gestión de flujo, denominado proceso de carga, adecuado para escribir datos en la zona útil (5) de la memoria autoadministrada,
  - al menos un proceso de gestión de flujo, denominado proceso de descarga, es adecuado para leer datos dla zona útil (5) de la memoria auto-administrada.
14. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 13, **caracterizado porque** los procesos de gestión de flujo son adecuados para cargarse en una zona RAM que se diferencia de la memoria autoadministrada (3).
15. Un dispositivo según cualquiera de las reivindicaciones 1 a 14, **caracterizado porque** el proceso de conmutación (27) es adecuado para cargarse en una zona de RAM que es distinta de la memoria autoadministrada (3).

16. Un dispositivo según cualquiera de las reivindicaciones 1 a 15, **caracterizado porque** el proceso de conmutación (27) es adecuado para:

- en una primera fase de análisis, analizando las características de cada flujo de pistas múltiples a procesarse y las restricciones de procesamiento de cada flujo de pistas múltiples, de tal manera que se definan los datos que representan las líneas de memoria y los datos que representan cada secuencia de uso de los búferes sincronizados de cada línea de memoria para procesar este flujo de pistas múltiples,
- después, en una segunda etapa de procesamiento subsiguiente, iniciando el procesamiento del flujo de pistas múltiples de acuerdo con dichos datos, que se definió previamente en la fase de análisis.

17. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 16, **caracterizado porque** el tamaño de la memoria auto-administrada (3) está definido por los medios de configuración a un valor fijo predeterminado.

18. Un dispositivo de acuerdo con la reivindicación 17, **caracterizado porque** el tamaño de la memoria autoadministrada (3) está entre 20% y 80% del de la memoria virtual.

19. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 18, **caracterizado porque** el tamaño de la zona de administración (4) está definido por los medios de configuración a un valor fijo predeterminado.

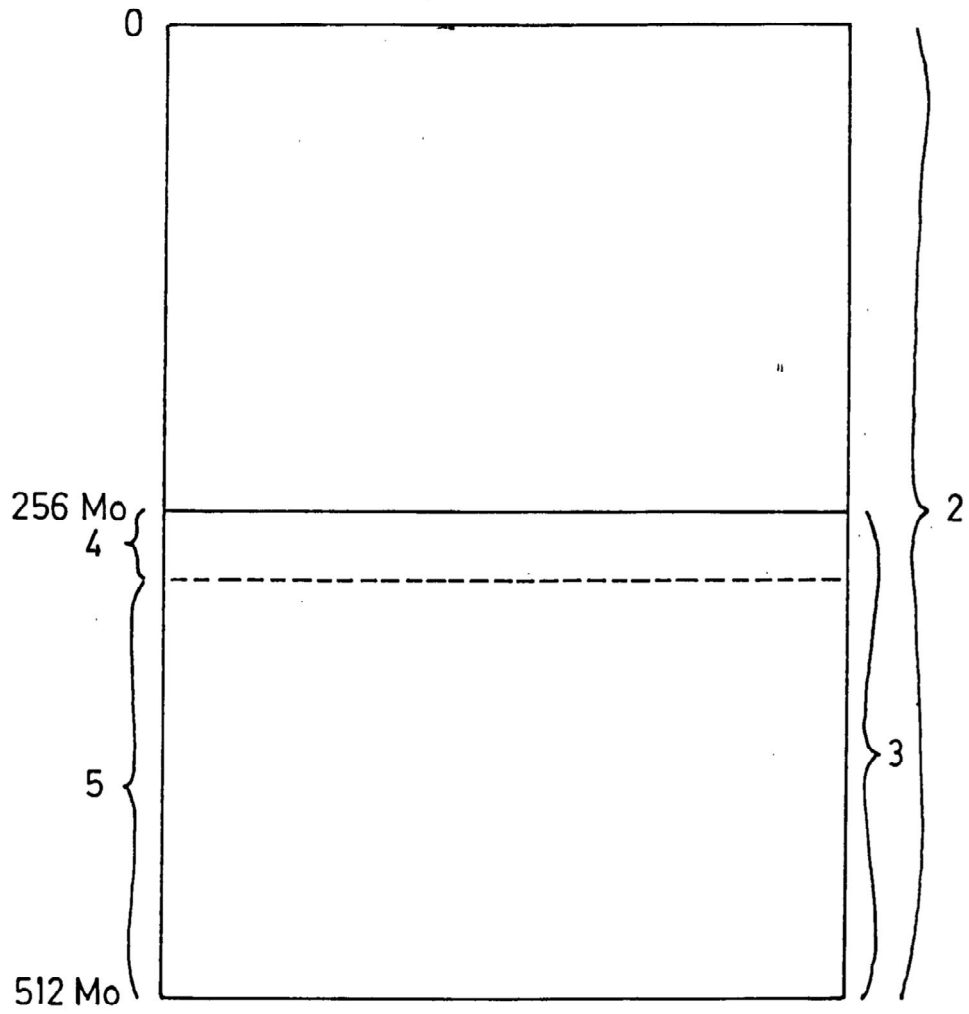
20. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 19, **caracterizado porque** la memoria (3) autoadministrada es definida por el proceso de conmutación (27) cuando se carga en RAM.

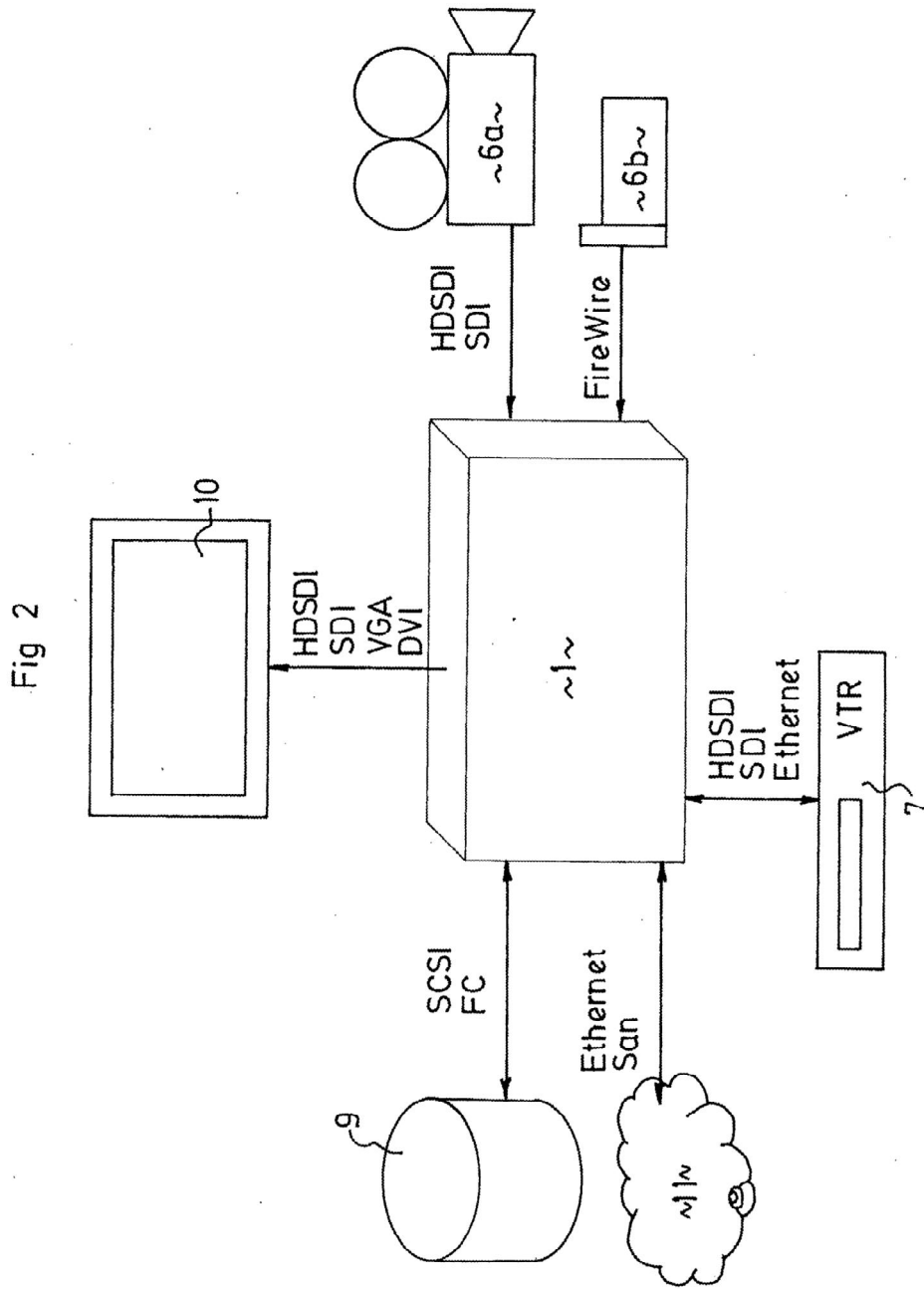
21. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 20, **caracterizado porque** los medios de configuración son adecuados para permitir el procesamiento de flujos de múltiples pistas que son flujos audiovisuales.

22. Un dispositivo de acuerdo con una cualquiera de las reivindicaciones 1 a 21, **caracterizado porque** los medios de configuración son adecuados para permitir el procesamiento de flujos de pistas múltiples con pistas cuyo formato se elige entre:

- formatos de televisión de alta definición (TVHD),
- formatos de televisión de definición estándar (SVSD),
- formatos de cine digital,
- formatos de video comprimidos (MPEG2, MPEG4, DV, etc.),
- formatos de audio no comprimidos,
- formatos de audio comprimidos,
- formatos de encapsulación pistas múltiples,
- formatos de imagen,
- formatos de datos audiovisuales sin procesar.

Fig 1





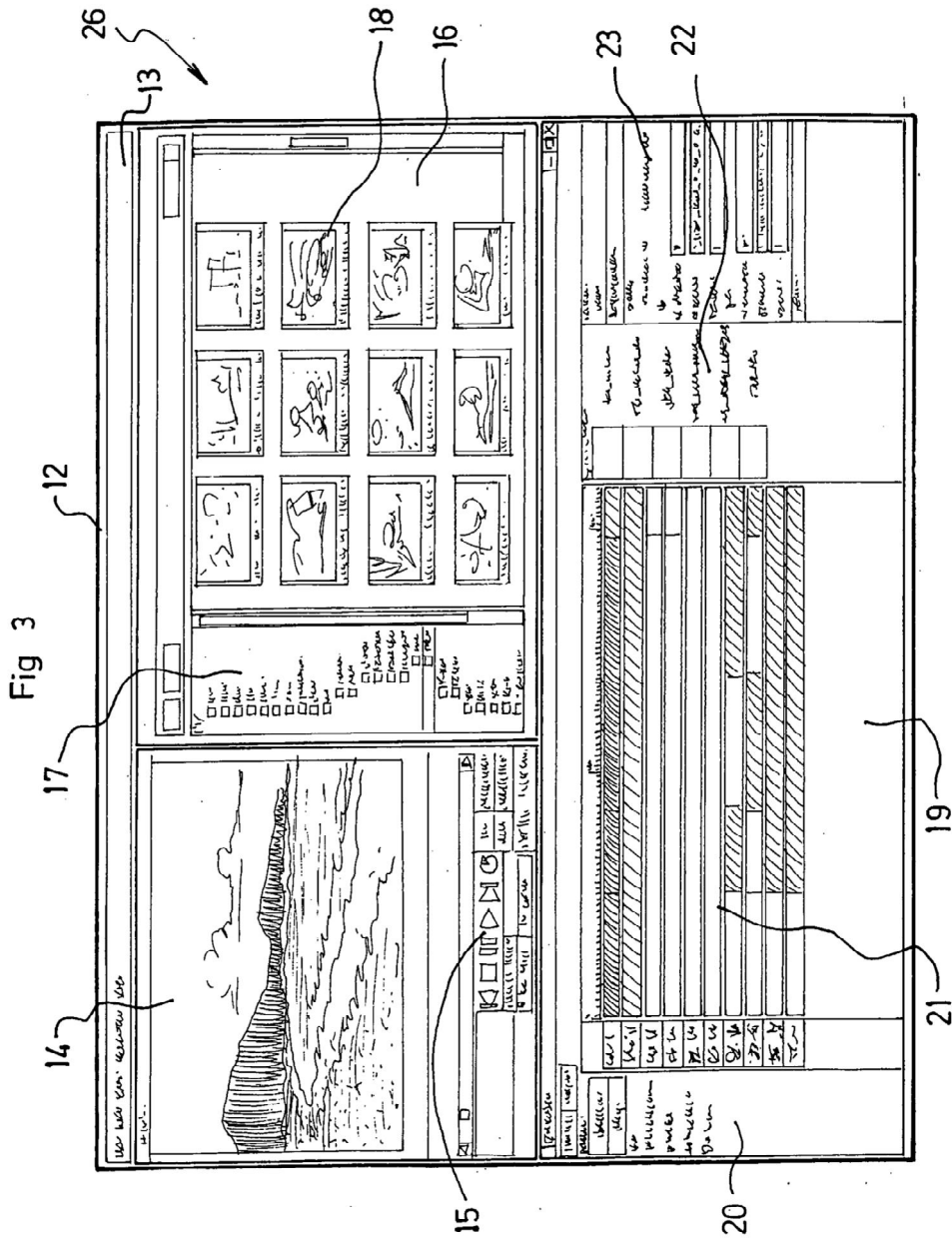


Fig 4

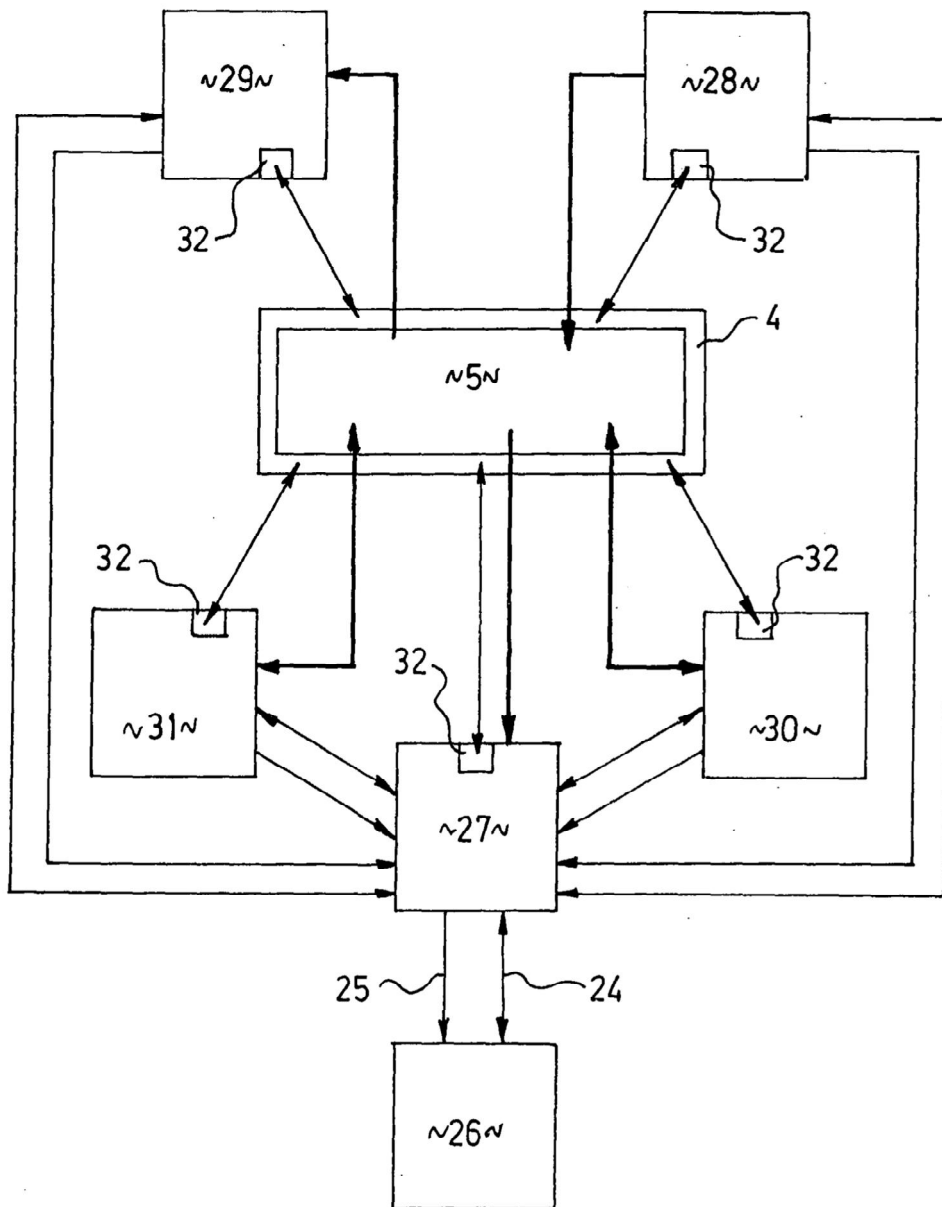


Fig 5

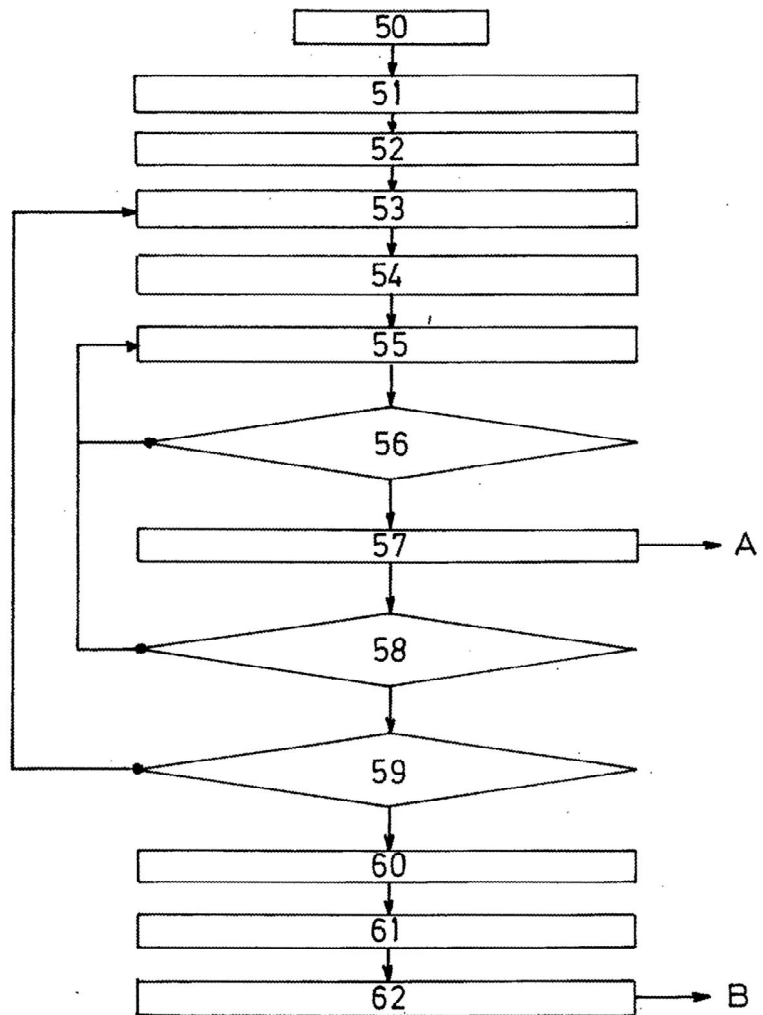


Fig 6

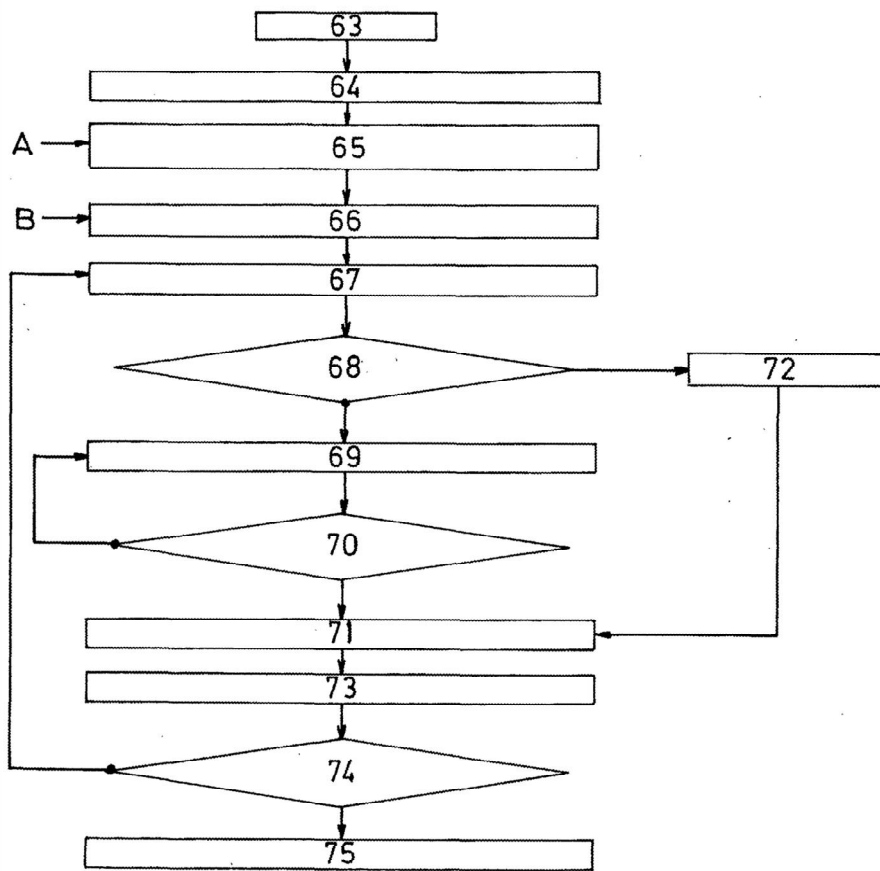




Fig 7

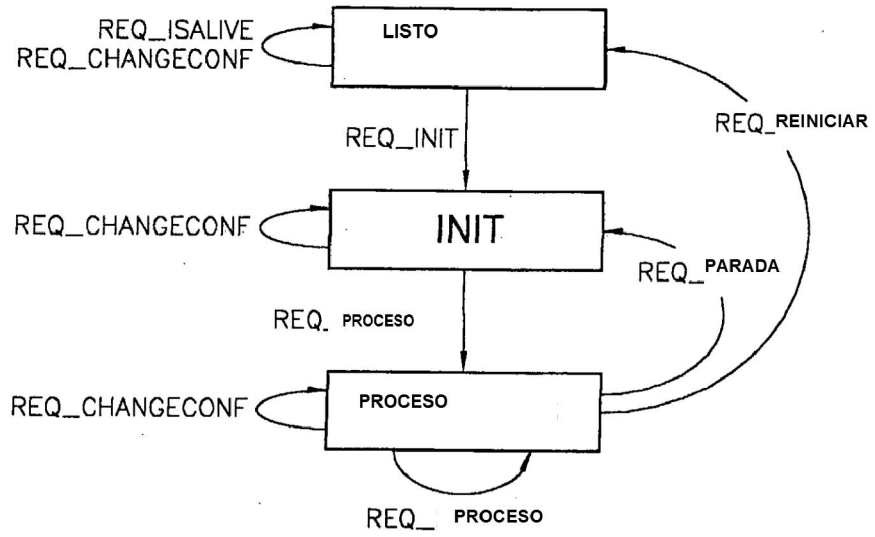


Fig 8

	00.00.00.00	00.00.03.00	00.00.03.01	00.00.06.02
V1	Mpeg2		DV	
V2				
A1	Mpeg2		Dv	
A2				
F1				

Fig 9

