

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 371 995**

51 Int. Cl.:
G06F 9/445 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **08860602 .5**
96 Fecha de presentación: **21.11.2008**
97 Número de publicación de la solicitud: **2229625**
97 Fecha de publicación de la solicitud: **22.09.2010**

54 Título: **ACTUALIZACIÓN DEL FIRMWARE DE UN DISPOSITIVO ELECTRÓNICO.**

30 Prioridad:
13.12.2007 US 13365 P

45 Fecha de publicación de la mención BOPI:
12.01.2012

45 Fecha de la publicación del folleto de la patente:
12.01.2012

73 Titular/es:
**Telefonaktiebolaget LM Ericsson (publ)
164 83 Stockholm, SE**

72 Inventor/es:
MORTON, Derek

74 Agente: **de Elzaburu Márquez, Alberto**

ES 2 371 995 T3

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

DESCRIPCIÓN

Actualización del firmware de un dispositivo electrónico.

CAMPO TÉCNICO

5 Esta invención se refiere en general a la actualización de software instalado en una memoria persistente de un dispositivo electrónico.

ANTECEDENTES Y DESCRIPCIÓN DE LA TÉCNICA RELACIONADA

10 Muchos dispositivos electrónicos modernos, como por ejemplo dispositivos manuales como dispositivos de comunicación manuales/portátiles, por ejemplo teléfonos móviles, comunicadores, buscas, u otros dispositivos portátiles/manuales como asistentes personales digitales, dispositivos de entretenimiento personal, como reproductores de mp3, etc., están controlados por software almacenado en una memoria flash. En la presente descripción, al software almacenado en una memoria persistente de un dispositivo electrónico que controla el funcionamiento o hace funcionar el dispositivo electrónico de tal modo que durante el funcionamiento normal del dispositivo electrónico no puede ser detenido y/o iniciado por separado del dispositivo electrónico como un todo, se hará referencia como firmware. Por tanto, se puede considerar que el firmware forma el software principal del sistema sobre el que corre. Además del firmware, se puede almacenar y ejecutar otro software en el dispositivo electrónico; dicho otro software generalmente puede iniciarse y detenerse por un usuario del dispositivo electrónico durante el funcionamiento normal del dispositivo electrónico, separadamente del arranque/apagado del dispositivo electrónico y el firmware. A un dispositivo electrónico controlado por firmware almacenado en una memoria persistente de dicho dispositivo electrónico también se hace referencia como un dispositivo integrado o un sistema integrado, y al firmware se hace referencia como integrado en el dispositivo electrónico. Aunque el firmware de un dispositivo electrónico típicamente no cambia durante el funcionamiento normal del dispositivo, pueden ser deseables o incluso necesarias actualizaciones del firmware durante la vida útil de un dispositivo electrónico, por ejemplo para instalar una nueva función, actualizar una funcionalidad existente, corregir errores en el firmware, y/o similares. Se apreciará que un dispositivo electrónico puede tener almacenado en el mismo más componentes de software, como software de aplicación, que proporciona funcionalidades adicionales a un usuario del dispositivo electrónico pero que no son necesarias para el funcionamiento del dispositivo electrónico.

Una actualización del firmware de un dispositivo electrónico implica varios problemas específicos, por ejemplo en comparación con una actualización de software de un ordenador personal:

30 Para actualizar una aplicación de un ordenador personal, esa aplicación normalmente debe detenerse si está activada. Se lleva a cabo entonces la actualización sobrescribiendo los archivos de la aplicación instalada con nuevas versiones. El sistema operativo (OS) generalmente no contiene software de actualización dedicado. Las actualizaciones se llevan a cabo ejecutando una aplicación que lleva a cabo los cambios necesarios en el sistema. Para actualizar las partes activas del software, por ejemplo el propio OS, se necesita reiniciar para completar el proceso. Cuando se necesita llevar a cabo acciones después del reinicio, la aplicación está preparada para arrancar de nuevo después del reinicio. Este proceso no está inherentemente libre de errores. Si la actualización se interrumpe, la aplicación queda dañada y el usuario tiene que repararlo.

40 Por otro lado, los dispositivos integrados frecuentemente utilizan una memoria flash para el almacenamiento persistente, ya que permite reescrituras múltiples. Una memoria flash generalmente incluye sectores de memoria, y las operaciones de lectura y/o escritura de la memoria flash típicamente se llevan a cabo en múltiplos de dichos sectores de memoria. Cuando el firmware almacenado en una memoria flash de un dispositivo electrónico es actualizado, por ejemplo para añadir nuevas características al software y/o para corregir errores de la versión actual del firmware, algunos o todos los sectores de memoria de la memoria flash son reescritos/reprogramados (se suele hacer referencia a este proceso como "re-flashear"). Cualquier espacio de datos de usuario típicamente será particionado separadamente del firmware, y normalmente no hay espacio disponible para copias completas de la imagen de firmware, de modo que las actualizaciones típicamente se llevan a cabo "en sitio". La nueva versión del firmware típicamente se escribe encima de la versión antigua. Durante el proceso de actualización, el firmware almacenado en la memoria persistente será parcialmente el firmware antiguo y parcialmente el firmware nuevo. Intentar ejecutar el firmware en este estado producirá resultados impredecibles, en el mejor de los casos, y más probablemente una caída del sistema.

50 Un sistema integrado generalmente ejecutará un sistema operativo en tiempo real. Los sistemas simples pueden no ejecutar ningún sistema operativo. A diferencia de lo que ocurre en un ordenador personal, todo el firmware está activo durante el funcionamiento del sistema integrado y no es posible cerrar una aplicación que es parte del firmware para actualizarlo. Si cualquier parte de tal sistema integrado falla, la totalidad del sistema falla. Debido a estas limitaciones, el proceso de actualización debería diseñarse para ser insensible a los fallos, o una interrupción de la actualización dará como resultado la avería del dispositivo.

55 En particular, una aplicación donde la fiabilidad del proceso de actualización es de gran importancia es la actualización de dispositivos de comunicación a-través-del-aire (OTA), como terminales móviles, es decir, una

actualización donde el software actualizado es recibido desde un sistema de actualización remoto a través de una red de comunicaciones celular. La actualización a-través-del-aire del firmware de tal dispositivo de comunicación también recibe el nombre de actualización de Firmware-A-Través-Del-Aire (FOTA).

5 El documento WO 03/625742 describe un método de actualizar software en un dispositivo remoto, donde el software almacenado en el dispositivo está particionado en una partición de arranque, una partición principal de firmware, y una partición de software auxiliar. Durante una actualización de software, la partición de software auxiliar es reescrita inicialmente por la versión actualizada del firmware principal. En un paso siguiente, el dispositivo es reiniciado con el nuevo firmware principal, y el software auxiliar actualizado es descargado y escrito encima del firmware principal antiguo. Aunque este proceso permite una actualización del software de un dispositivo electrónico que no tiene memoria suficiente para una copia adicional completa del software, este proceso requiere una actualización completa de todo el software para alcanzar de nuevo un estado completamente operativo. Sería, por tanto, deseable proporcionar un proceso de actualización más eficiente que requiere un menor tiempo de descarga y/o ancho de banda. Además, el proceso de la técnica anterior descrito depende de su capacidad para reconectarse al servidor desde el cual se descarga la actualización de software después del reinicio, o al menos después de una pérdida de potencia u otra interrupción del proceso de actualización, para descargar el software auxiliar. Si el proceso falla en la reconexión al servidor, el dispositivo queda en un estado no operativo. Por tanto, continúa siendo un problema el proporcionar un proceso de actualización que tenga un alto grado de resistencia a los fallos.

SUMARIO

20 En el presente documento se describe un método para actualizar el software de un dispositivo electrónico desde una versión actual del software hasta una versión actualizada del software, comprendiendo el dispositivo electrónico un medio de almacenamiento que tiene almacenado en el mismo la versión actual del software. Las realizaciones del método descritas en el presente documento comprenden las siguientes operaciones a llevar a cabo por el dispositivo electrónico:

25 - recibir instrucciones de actualización incrementales para provocar que el dispositivo electrónico transforme la versión actual almacenada del software en la versión actualizada del software; donde el software comprende un primer y un segundo conjunto de módulos de carga, comprendiendo las instrucciones de actualización incrementales recibidas un primer conjunto de instrucciones de actualización para provocar que el dispositivo electrónico genere una versión actualizada del primer conjunto de módulos de carga, comprendiendo además las instrucciones de actualización incrementales recibidas un segundo conjunto de instrucciones de actualización para provocar que el dispositivo electrónico genere una versión actualizada del segundo conjunto de módulos de carga;

30 - almacenar al menos el segundo conjunto de instrucciones actualizadas en el medio de almacenamiento;

35 - ejecutar el primer conjunto de instrucciones de actualización para actualizar la versión actual almacenada del primer conjunto de módulos de carga por la versión actualizada generada del primer conjunto de módulos de carga;

- arrancar el dispositivo electrónico en un modo de actualización en el que se ejecuta la versión actualizada del primer conjunto de módulos de carga;

40 - ejecutar el segundo conjunto de instrucciones de actualización almacenadas para llevar a cabo una actualización en-sitio del segundo conjunto de módulos de carga, comprendiendo la actualización en-sitio reemplazar la versión actual almacenada del segundo conjunto de módulos de carga con la versión actualizada generada del segundo conjunto de módulos de carga.

45 En consecuencia, las realizaciones del método descritas en el presente documento comprenden llevar a cabo una actualización incremental, requiriendo así sólo la descarga de un paquete de actualización relativamente pequeño y su almacenamiento en el medio de almacenamiento del dispositivo electrónico. Tal actualización incremental también es denominada actualización delta, ya que el paquete de actualización únicamente contiene las diferencias entre las versiones actual y actualizada del software y por tanto típicamente se considera más pequeña que el software que se va a actualizar.

50 Como el primer conjunto de módulos de carga sólo necesita incluir una pequeña parte del firmware suficiente para llevar a cabo una actualización, el segundo conjunto de módulos de carga incluye la mayoría del software. Por ejemplo, cuando el dispositivo electrónico es un dispositivo de comunicaciones de radio portátil para la comunicación a través de una red de comunicaciones por radio, el software puede incluir funciones de comunicación para establecer comunicaciones por radio entre el dispositivo de comunicaciones y otro dispositivo de comunicaciones de la red de comunicaciones por radio, por ejemplo una estación base. Como se descarga el conjunto completo de instrucciones de actualización y al menos el segundo conjunto de instrucciones de actualización se almacena en la memoria persistente antes de la actualización, una vez las instrucciones de actualización han sido recibidas y verificadas, el dispositivo electrónico no necesita contactar con una entidad remota, como una estación base, durante el resto del proceso de actualización, ni siquiera en el caso de que el proceso de actualización sea

interrumpido y tenga que reanudarse. Por tanto, toda la funcionalidad de las comunicaciones, incluidas funciones requeridas para establecer contacto con una estación base de una red celular, puede incluirse en el segundo conjunto de módulos de carga, manteniendo así pequeño el tamaño del primer conjunto de módulos de carga, y por tanto la memoria necesaria para la actualización resistente a fallos del primer conjunto de módulos de carga.

5 Cuando el primer conjunto de módulos de carga comprende un módulo de carga principal adaptado para proporcionar un conjunto mínimo de servicios de sistema básicos requeridos para el funcionamiento del dispositivo electrónico, y un segundo módulo de carga adaptado para proporcionar un conjunto mínimo de servicios básicos requeridos por el proceso de actualización, el primer conjunto de módulos de carga se mantiene pequeño, reduciéndose así la capacidad de almacenamiento requerida para una actualización resistente a fallos del primer conjunto de módulos de carga. El módulo de carga principal puede comprender un sistema operativo principal del dispositivo electrónico y uno o más controladores físicos. El segundo módulo de carga puede comprender código de programación requerido para controlar el dispositivo electrónico para indicar el progreso del proceso de actualización a un usuario del dispositivo electrónico.

10 Además, como al menos el segundo conjunto de módulos de carga se cambian en-sitio, es decir, la versión actualizada del segundo conjunto de módulos de carga se escribe sustancialmente en la misma parte del medio de almacenamiento que está ocupado por la versión actual del segundo conjunto de módulos de carga, no es necesario sobrescribir una parte del software actual antes de su actualización, haciendo que el proceso sea más resistente a fallos. Se apreciará que las versiones actual y actualizada pueden tener un tamaño diferente, provocando así que la versión actualizada después de la actualización en-sitio ocupe un espacio de memoria adicional en comparación con la versión actual y/o deje sin ocupar parte del espacio de memoria anteriormente ocupado por la versión actual. Adicionalmente, como el primer conjunto de módulos de carga es pequeño, se puede conseguir sin necesidad de un gran espacio de memoria libre una actualización resistente a fallos que no requiere sobrescribir el segundo conjunto de módulos.

15 Además, cuando sólo requiere una actualización una porción de todo el firmware, sólo esa porción realmente debe ser reescrita, proporcionándose así un proceso de actualización rápido, en particular con relación a tipos de memoria que requieren un proceso de escritura múltipaso, por ejemplo, cuando se debe primer eliminar contenido antiguo antes de que se pueda escribir contenido nuevo en la misma porción de memoria. También, como algunas tecnologías de memoria sólo permiten la reescritura de una memoria un número limitado de veces, no se reduce innecesariamente la vida útil de la memoria del dispositivo electrónico.

20 Además, como la actualización delta es recibida y almacenada en el medio de almacenamiento antes de la actualización, no es necesario descargar o re-descargar partes de la actualización una vez que el proceso de actualización actual ha sido iniciado. Incluso en el evento de que se produzca una interrupción del proceso de actualización, por ejemplo debido a una caída/corte de tensión, las realizaciones del proceso de actualización descritas en el presente documento pueden reanudarse y completarse sin tener que re-descargar ninguna parte del software actualizado. Esto puede ser particularmente ventajoso en situaciones en las que la versión antigua de alguno o todos los módulos de carga del segundo conjunto de módulos de carga no pueden funcionar junto con las versiones actualizadas de los módulos de carga del primer conjunto.

Por tanto, las realizaciones del método descritas en el presente documento proporcionan un alto grado de seguridad ante fallos sin la necesidad de un gran espacio de memoria persistente libre.

25 Además, las realizaciones del método descritas en el presente documento proporcionan una arquitectura modular del software donde no hay duplicación de código o funcionalidades. En algunas realizaciones de esta arquitectura, se trata un agente de actualización como un módulo de carga dentro del sistema, en particular un módulo de carga de entre el primer conjunto de módulos de carga. Además, el primer conjunto de módulos de carga que comprende las funciones de software requeridas para llevar a cabo la actualización se puede utilizar tanto durante el proceso de actualización como durante el funcionamiento normal, minimizándose así la duplicación de código con relación a las funciones de actualización y proporcionando un mecanismo de actualización que utiliza eficientemente los recursos. El segundo conjunto de módulos de carga no es necesario para el funcionamiento del dispositivo en el modo de actualización, es decir, el dispositivo electrónico puede funcionar en el modo de actualización sin ejecutar el segundo conjunto de módulos de carga.

30 Para la presente descripción, los términos memoria persistente y memoria no-volátil se refieren ambos a cualquier forma de memoria legible por ordenador que pueda retener la información almacenada incluso cuando no reciben alimentación.

35 En la presente descripción, el término módulo de carga pretende comprender una unidad de software identificable que se pueda identificar en una memoria persistente y sea adecuada para la ejecución bien en-sitio, es decir, directamente de la memoria persistente, o bien cargando el módulo de carga desde la memoria persistente a una memoria principal (por ejemplo, una RAM) o una unidad de procesamiento. Se apreciará que durante la actualización del primer y/o segundo conjunto de módulos de carga, uno o más de los módulos de carga del correspondiente primer y/o segundo conjuntos de módulos de carga pueden permanecer sin modificar, es decir, una actualización de un conjunto de módulos de carga no implica necesariamente un cambio de todos los módulos de

carga del conjunto.

En algunas realizaciones, ejecutar el primer conjunto de instrucciones de actualización para actualizar la versión actual almacenada del primer conjunto de módulos de carga con la versión actualizada generada del primer conjunto de módulos de carga comprende:

- 5
- crear una copia de seguridad del primer conjunto de módulos de carga y ejecutar el primer conjunto de instrucciones de actualización para sustituir la copia de seguridad almacenada del primer conjunto de módulos de carga con la versión actualizada generada del primer conjunto de módulos de carga;
 - intercambiar las versiones de copia de seguridad y original del primer conjunto de módulos de carga.

10 En consecuencia, al aplicar las instrucciones de actualización recibidas a una copia de seguridad de la versión actual del primer conjunto de módulos de carga, el dispositivo electrónico puede hacerse funcionar todavía incluso si la generación de la versión actualizada del primer conjunto de módulos de carga falla. Por ejemplo, la operación de intercambio se puede llevar a cabo modificando un puntero u otra referencia que indique – por ejemplo, a un programa de inicio – la posición del primer conjunto de módulos de carga, de modo que el puntero modificado u otra referencia apunte a la versión actualizada del primer conjunto de módulos de carga.

15 En general, en algunas realizaciones, la actualización del primer conjunto de módulos de carga se lleva a cabo de un modo resistente a fallos, es decir, de modo que el dispositivo electrónico se puede arrancar al menos en un estado que permite la continuación/reanudación o reinicio del proceso de actualización, incluso si un intento inicial de llevar a cabo el proceso de actualización ha sido interrumpido o ha fallado de otro modo. Se apreciará que la secuencia exacta de las operaciones para llevar a cabo una actualización resistente a fallos del primer conjunto de módulos de carga puede diferir. Por ejemplo, en algunas realizaciones la actualización puede llevarse a cabo en la copia de seguridad del primer conjunto de módulos de carga y luego intercambiarlos, mientras que en otras realizaciones la actualización puede llevarse a cabo en la versión original del primer conjunto de módulos de carga. En esta última realización, la copia de seguridad puede cargarse al iniciar si se detecta que la actualización se ha interrumpido. La elección de la secuencia exacta puede depender de la implementación específica, por ejemplo, dependiendo de si el código se ejecuta en-sitio.

20

25

En algunas realizaciones, todo el conjunto de instrucciones de actualización recibidas se almacena en el medio de almacenamiento para evitar la necesidad de recibir el primer conjunto de instrucciones de actualización de nuevo en caso de que la actualización del primer conjunto de módulos de carga falle, proporcionándose así una actualización resistente a fallos del primer conjunto de módulos de carga.

- 30 En algunas realizaciones, el dispositivo electrónico tiene además almacenado en el mismo un programa de carga de reinicio, y arrancar el dispositivo electrónico en el modo de actualización comprende:
- ejecutar el programa de carga de reinicio,
 - iniciar la ejecución de al menos uno del primer conjunto de módulos de carga bajo el control del programa de carga de reinicio,
 - 35 - opcionalmente terminar la ejecución del programa de carga de reinicio,
 - iniciar la ejecución del agente de actualización bajo el control del al menos uno de entre el primer conjunto de módulos de carga.

40 En algunas realizaciones, el método comprende además ejecutar las versiones actualizadas del segundo conjunto de módulos de carga después de finalizar con éxito la actualización del segundo conjunto de módulos de carga. En particular, la versión actualizada del segundo conjunto de módulos de carga se puede ejecutar directamente después de completar la actualización del segundo conjunto de módulos de carga, es decir, sin ninguna necesidad de volver a arrancar el dispositivo electrónico.

45 Se debe remarcar que las características de los métodos descritos anteriormente y más adelante pueden ser implementadas mediante software y llevadas a cabo utilizando medios de procesamiento de un dispositivo electrónico controlado por la ejecución de medios de código de programa, como instrucciones ejecutables por ordenador. Aquí, y en lo que sigue, el término medios de procesamiento comprende cualquier circuito y/o dispositivo adecuadamente adaptado para llevar a cabo las funciones anteriores. En particular, el término medios de procesamiento comprende microprocesadores programables de propósito general – o específico –, Procesadores de Señales Digitales (DSP), Circuitos Integrados de Aplicación Específica (ASIC), Matrices Lógicas Programables (PLA), Matrices de Puertas Programables de Campo (FPGA), circuitos electrónicos de propósito específico, etc., o combinaciones de los mismos.

50

La presente invención se refiere a diferentes aspectos, que incluyen el método descrito anteriormente, un dispositivo electrónico correspondiente, u otros medios de producto, cada uno de los cuales proporciona uno o más de los beneficios y ventajas que se describen con relación con el método descrito anteriormente, y cada uno de los cuales

tiene una o más realizaciones que corresponden a las realizaciones descritas con relación al método descrito anteriormente y/o a las reivindicaciones dependientes.

5 En particular, se describen en el presente documento realizaciones de un dispositivo electrónico controlable por software, comprendiendo el dispositivo electrónico un medio de almacenamiento que tiene almacenado en el mismo una versión actual de software, donde el dispositivo electrónico está configurado para llevar a cabo las operaciones del método descrito anteriormente y más adelante.

10 A los efectos de la presente descripción, el término dispositivo electrónico comprende cualquier dispositivo que comprende un medio de almacenamiento como una memoria flash u otra forma de memoria persistente para almacenar código de programa. Ejemplos de tales dispositivos incluyen equipos portátiles de comunicaciones por radio y otros dispositivos portátiles o manuales. El término equipo portátil de comunicaciones por radio incluye todo tipo de equipos como teléfonos móviles, buscas, comunicadores, es decir, organizadores electrónicos, teléfonos inteligentes, asistentes personales digitales (PDAs), ordenadores portátiles, reproductores, como por ejemplo reproductores de mp3, cámaras digitales u otros medios de grabación, dispositivos integrados de la industria de la automoción, dispositivos médicos, o similares.

15 El medio de almacenamiento puede ser un medio de almacenamiento no volátil, como una memoria flash. Se apreciará que el método y aparato descritos en el presente documento también puede ser implementado con relación a otros tipos de medios de almacenamiento.

20 En un aspecto, se describe en el presente documento un producto de programa de ordenador que comprende un programa de carga de reinicio y software, comprendiendo el producto de programa de ordenador medios de código de programa adaptados para provocar que un dispositivo electrónico lleve a cabo las operaciones del método descrito anteriormente y más adelante cuando dichos medios de código de programa son ejecutados por el dispositivo electrónico.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

25 Los anteriores y otros aspectos serán evidentes a partir de las realizaciones descritas a continuación con referencia a las figuras, en las que:

La Figura 1 muestra esquemáticamente un diagrama de bloques de una realización de un sistema para actualizar software en un dispositivo electrónico.

La Figura 2 muestra esquemáticamente un diagrama de bloques de un dispositivo electrónico como un terminal móvil.

30 La Figura 3 ilustra un ejemplo de una secuencia de inicio convencional.

La Figura 4 ilustra un ejemplo de una secuencia de inicio de un sistema modular.

La Figura 5 ilustra esquemáticamente una distribución de un ejemplo de memoria persistente de un dispositivo electrónico durante un proceso de actualización según se describe en el presente documento.

La Figura 6 muestra un diagrama de flujo de un ejemplo de un proceso de actualización.

35 DESCRIPCIÓN DETALLADA

La Figura 1 muestra esquemáticamente un diagrama de bloques de una realización de un sistema para actualizar software en un dispositivo electrónico, como un terminal móvil. El sistema comprende un dispositivo 101 electrónico, por ejemplo un teléfono móvil o similar, un sistema 102 de actualización de software, y una interfaz 103 de comunicaciones.

40 El sistema 102 de actualización de software puede comprender un ordenador servidor con acceso a la red de comunicaciones. En algunas realizaciones, la funcionalidad del ordenador servidor puede ser distribuida entre una pluralidad de ordenadores, por ejemplo ordenadores conectados por medio de una red de ordenadores, por ejemplo, una red de área local (LAN), una red de área amplia (WAN), Internet o similar. El sistema 102 de actualización de software comprende un circuito 104 interfaz que permite que el sistema de actualización de software comunique datos a través de la red 103 de comunicaciones. Por ejemplo, el circuito de interfaz puede comprender un puerto serie, un puerto paralelo, una interfaz de comunicaciones inalámbricas de corto alcance, por ejemplo un puerto de infrarrojos, un transceptor Bluetooth, o similares. Otros ejemplos de circuitos de interfaz incluyen una tarjeta de red, un módem DSL, un ordenador pasarela, o similares.

50 El sistema de actualización de software comprende además una unidad 105 de procesamiento, por ejemplo la CPU de un ordenador servidor, programada adecuadamente para controlar y llevar a cabo el proceso de actualización, incluyendo la generación del código de programa actualizado según se describe en el presente documento. La unidad de procesamiento puede comprender además una base 106 de datos de versiones que tiene almacenadas

en la misma imágenes de memoria de al menos una versión base y una versión actualizada del software a actualizar. En algunas realizaciones, la base de datos de versiones puede comprender además información adicional, por ejemplo una pluralidad de versiones base y/o versiones actualizadas, por ejemplo, para diferentes modelos o dispositivos electrónicos, para diferentes grupos de consumidores, y/o similar.

- 5 La interfaz 103 de comunicaciones puede ser cualquier interfaz de comunicaciones adecuada, por cable o inalámbrica, para comunicar datos entre el sistema 102 de actualización de software y el dispositivo 101 electrónico. Por ejemplo, en el caso de un teléfono móvil adaptado para la comunicación a través de una red de comunicaciones celular, por ejemplo una red GSM, una red UMTS, una red GPRS, o similares, la comunicación entre el sistema de actualización de software y el dispositivo electrónico para una actualización de software se puede llevar a cabo por
- 10 medio de dicha red de comunicaciones celular, evitándose así la necesidad de interfaces de comunicación adicionales en el dispositivo electrónico. Se entiende también que la comunicación entre el dispositivo electrónico y el sistema de actualización de software puede implicar más de una red de comunicaciones. Por ejemplo, el teléfono móvil puede comunicarse a través de una estación base y una red de telecomunicaciones celular con un sistema pasarela que, a su vez, proporciona comunicación con el sistema de actualización de software a través de Internet.
- 15 Por tanto, para actualizar el software del dispositivo 101 electrónico, por ejemplo el firmware del dispositivo electrónico o partes del mismo, el dispositivo electrónico puede conectarse al sistema 102 de actualización de software. Alternativamente, el sistema de actualización de software puede conectarse al dispositivo electrónico una vez que esté disponible una versión de software actualizada. Una vez conectado al sistema de actualización de software, el dispositivo electrónico puede enviar información al sistema de actualización de software acerca de su
- 20 versión de software actual. La comunicación se puede llevar a cabo mediante un protocolo de actualización adecuado, por ejemplo un protocolo construido sobre un protocolo TCI/IP. Basándose en la información recibida del dispositivo electrónico, el sistema de actualización de software puede generar un mensaje/paquete de actualización dedicado que comprende instrucciones de actualización para el dispositivo electrónico. En algunas realizaciones, las instrucciones de actualización incluyen las imágenes de los sectores de memoria a reescribir. En algunas realizaciones, se puede utilizar el mismo paquete de actualizaciones para una pluralidad de dispositivos electrónicos. En un sistema de actualización diferencial que utiliza archivos delta, las instrucciones de actualización generadas de modo que habilitan al dispositivo electrónico para que genera la versión de software actualizada a partir de la versión existente ya almacenada en el dispositivo electrónico y a partir de información adicional incluida en las instrucciones de actualización. Son conocidas técnicas adecuadas para generar archivos delta, por ejemplo según se describe en
- 30 WO 05/08996.

En una realización, el proceso de actualización es iniciado por un agente de actualización que se ejecuta en el dispositivo electrónico. El agente de actualización controla la recepción y verificación del archivo delta. A continuación, el agente de actualización provoca que el dispositivo electrónico se desconecte de la red y que se lleve a cabo el proceso de actualización. Una realización de un proceso de actualización se describe con mayor detalle

35 más adelante.

La generación de un archivo delta puede ilustrarse esquemáticamente mediante las siguientes operaciones:

$$\text{archivo}_{\text{nuevo}} - \text{archivo}_{\text{base}} \text{ ---} \Delta \text{archivo},$$

donde $\text{archivo}_{\text{nuevo}}$ denota la versión actualizada del software, $\text{archivo}_{\text{base}}$ denota la versión actual del software, y $\Delta \text{archivo}$ denota las instrucciones de actualización delta.

- 40 Correspondientemente, la generación de la nueva versión puede ser llevada a cabo por el dispositivo electrónico de acuerdo con la siguiente operación:

$$\text{archivo}_{\text{base}} + \Delta \text{archivo} \text{ ---} \text{archivo}_{\text{nuevo}},$$

- Se entenderá que las operaciones anteriores para generar el archivo delta (denotadas como “-“ en la notación anterior) y para generar la nueva versión del dispositivo electrónico (denotadas como “+“ en la notación anterior) pueden comprender operaciones más o menos complicadas. El término aplicar instrucciones de actualización al software, por ejemplo un módulo de carga, se refiere en general a las operaciones para generar una versión actualizada del software a partir de la versión actual del software y de las instrucciones de actualización.
- 45

- Como se describirá con mayor detalle más adelante, el software a actualizar puede comprender una pluralidad de módulos de carga. En consecuencia, el archivo delta puede incluir una correspondiente pluralidad de secciones de archivo-delta, cada una de ellas para generar una versión actualizada de uno de los módulos de carga. Se apreciará que, alternativamente, un paquete de actualización puede inducir una pluralidad de archivos delta separados, cada uno de ellos para actualizar uno de los módulos de carga. El archivo delta puede aplicarse en-sitio, es decir, los cambios son realizados por el dispositivo electrónico sobre la imagen existente, requiriéndose así poco espacio de almacenamiento adicional. Además, como sólo es necesario cargar el archivo delta y como el archivo delta típicamente es considerablemente más pequeño que la nueva versión, mediante el método anterior se reduce el
- 50 tiempo de carga.
- 55

La Figura 2 muestra esquemáticamente un diagrama de bloques de un dispositivo electrónico como un terminal móvil. El dispositivo 101 electrónico comprende un bloque 210 de comunicaciones, una unidad 211 de procesamiento, y una unidad 212 de memoria.

5 El bloque 210 de comunicaciones comprende circuitos y/o dispositivos que permiten la comunicación por radio de datos a través de una red de comunicaciones celular. Por tanto, el bloque 210 de comunicaciones puede comprender circuitos de receptor y circuitos de transmisor para recibir y transmitir señales de datos. El bloque de comunicaciones puede comprender además circuitos para procesar adecuadamente – por ejemplo, modular, codificar, amplificar, etc. – las señales por medio de técnicas adecuadas bien conocidas como tales en el campo de las comunicaciones por radio.

10 El dispositivo electrónico comprende además una unidad 211 de procesamiento, por ejemplo un microprocesador adecuadamente programado. La unidad de procesamiento puede estar adaptada para determinar la versión del software almacenado en el dispositivo electrónico, para calcular sumas de control del software almacenado, y para generar una versión actualizada del software cuando recibe las correspondientes instrucciones de actualización.

15 La unidad 212 de memoria comprende memoria no volátil que tiene almacenado en la misma el software en una versión predeterminada. Por ejemplo, la memoria 121 puede comprender el firmware del dispositivo electrónico que implementa funciones del dispositivo electrónico cuando se carga en el mismo y se ejecuta por la unidad 210 de procesamiento. El firmware puede comprender además un sistema operativo que permita la ejecución de software de aplicación. En consecuencia, la memoria 212 puede tener además almacenada en la misma software de aplicación para proporcionar funciones adicionales. A la memoria 212 se accede utilizando un espacio de memoria adecuado, permitiendo así a la unidad de procesamiento acceder a partes seleccionadas de la memoria. En algunas realizaciones, la memoria 212 puede estar dividida lógicamente o físicamente en una pluralidad de sectores de memoria. Por ejemplo, la memoria 212 puede comprender una memoria flash que permite que los datos se escriban en sectores de un tamaño predeterminado.

25 También se entiende que el proceso de actualización descrito en el presente documento se puede aplicar a toda la memoria 212, por ejemplo si se tiene que actualizar toda la imagen de la memoria flash de un teléfono móvil, o sólo partes predeterminadas de la memoria, por ejemplo si se deben actualizar una o más aplicaciones de software.

También se apreciará que el dispositivo electrónico puede comprender además una memoria volátil, como una RAM.

30 La Figura 3 ilustra un ejemplo de una secuencia de inicio convencional durante una actualización del firmware de un dispositivo electrónico. En este ejemplo, cuando el dispositivo electrónico es encendido, el sistema ejecuta inicialmente un código de inicio 321. El código de inicio provoca que un agente 322 de actualización de firmware se cargue, es decir, una pieza de código autónoma que provoca cambios en el cuerpo principal del código firmware. Durante su ejecución, el agente de actualización puede provocar que el dispositivo electrónico muestre una indicación adecuada del progreso del proceso de actualización. Cuando la actualización haya sido completada, el agente de actualización termina y se ejecuta el firmware 323 actualizado.

35 Incluso si es posible llevar a cabo actualizaciones del firmware principal en segundo plano mientras el firmware se está ejecutando, el agente 322 de actualización autónomo también es necesario para continuar la actualización en caso de que ésta haya quedado interrumpida, por ejemplo, si el dispositivo se desconecta durante la actualización. Para cualquier sistema que lleve a cabo una actualización en-sitio, es necesario un Agente de Actualización de Firmware 322 para permitir que el sistema se recupere de una interrupción de alimentación durante la actualización. El Agente de Actualización de Firmware 322 provoca la duplicación del código. Si el Agente de Actualización de Firmware contiene un núcleo de sistema operativo (OS), puede ocupar unos recursos de memoria persistente adicional significativa en el dispositivo integrado. Si el Agente de Actualización de Firmware está escrito para no necesitar OS, sería necesario mantener los controladores que requiera tanto en la versión con OS (para el firmware principal) como en la versión sin OS (para el Agente de Actualización de Firmware). Esto aumenta el coste de desarrollo y mantenimiento del sistema. Si el sistema es capaz de llevar a cabo actualizaciones en segundo plano, todas las funciones requeridas del Agente de Actualización de Firmware 322 también estarán duplicadas en el firmware principal. El Agente de Actualización de Firmware autónomo 322 sigue siendo necesario, ya que una pérdida de alimentación durante una actualización en-sitio del firmware principal dejará el firmware principal en un estado en que no se puede ejecutar para continuar la actualización.

40 En lo que sigue, se describirá una realización del proceso de actualización con referencia a las Figuras 4-6. La Figura 4 ilustra un ejemplo de una secuencia de inicio de una arquitectura modular para actualizar un dispositivo electrónico, mientras que la Figura 5 ilustra esquemáticamente un ejemplo de configuración de la memoria 212 persistente de un dispositivo electrónico durante un proceso de actualización según se describe en el presente documento, por ejemplo, el dispositivo 101 electrónico que se describe con relación a las Figuras 1 y 2. En particular, la Figura 5a muestra un ejemplo de una configuración de memoria antes de una actualización. La Figura 6 muestra un diagrama de flujo de un ejemplo de un proceso de actualización.

En la realización ilustrada en las Figuras 4 y 5a, el dispositivo electrónico tiene almacenados en el mismo código de inicio 421, y firmware 531 para controlar el funcionamiento del dispositivo electrónico. El código de inicio 421

5 generalmente incluye las primeras instrucciones que se deben ejecutar en el arranque (inicio) del dispositivo electrónico. El firmware 531 está dispuesto como un sistema modular que comprende varios módulos de carga designados módulo de carga A (432), módulo de carga B (433), agente de actualización UA (434), módulo de carga C (435) y módulo de carga D (436). El módulo de carga A es el módulo de carga principal que proporciona los servicios de sistema básicos, por ejemplo, el sistema operativo, y es iniciado por el código de inicio 421. Por ejemplo, el módulo de inicio A principal puede contener el sistema operativo principal y otros controladores físicos requeridos, como controladores flash, controladores de control de alimentación, E/S, controladores de archivos de sistema, etc. El módulo de carga B es el módulo de carga que proporciona los servicios básicos requeridos por el agente 434 de actualización además de los servicios proporcionados por el módulo de carga A. Por tanto, el módulo de carga B puede contener código que no es parte del OS principal pero que se utiliza durante una actualización. Por ejemplo, en un dispositivo dotado de pantalla, el módulo de carga B puede incluir los controladores de la pantalla, codecs, etc. necesarios para reportar el progreso de la actualización. En dispositivos alimentados por baterías, el módulo de carga B puede incluir los algoritmos para controlar la carga de la batería. El agente de actualización UA es un módulo de carga que incluye el código que realmente lleva a cabo cualquier actualización 15 440 si es necesaria o deseada una actualización. Los módulos de carga C y D son otros módulos de carga que comprenden el resto del firmware. Por ejemplo, los módulos de carga C y/o D pueden contener código que no es necesario ejecutar durante una actualización, por ejemplo aplicaciones, listas de redes, controladores de audio, codecs de vídeo, etc.

20 En la secuencia de inicio de la Figura 4, el código de inicio 421 es ejecutado cuando se enciende el dispositivo electrónico. El código de inicio 421 inicia el módulo de carga A principal que proporciona servicios de sistema básicos. Después del arranque con éxito del módulo de carga A, puede terminar el código de inicio. El módulo de carga A principal arranca entonces el resto de módulos de carga en un orden predeterminado que corresponde a sus interdependencias. En particular, el módulo de carga A principal arranca el módulo de carga B antes de arrancar el agente de actualización UA, ya que el agente de actualización requiere servicios proporcionados por el módulo de carga B, como se ilustra por medio de las flechas 441 y 442, que ilustran que el agente de actualización puede utilizar servicios de visualización proporcionados por el módulo de carga B para mostrar una indicación del progreso a través de una pantalla del dispositivo electrónico durante una actualización 440.

30 Como se indica en la Figura 4, el agente de actualización puede llevar a cabo una actualización 440 cuando es activado para ello, por ejemplo en el inicio. Cuando se ha completado la actualización (o si no es necesaria), el agente de actualización UA puede enviar una señal de "listo" 443 al módulo de carga A principal, provocando que el módulo de carga A inicie los módulos de carga restantes C y D. Como se describirá a continuación con mayor detalle, la secuencia de inicio mostrada en la Figura 4 se puede aplicar durante una segunda etapa de un proceso de actualización donde los módulos de carga C y D son actualizados. Durante una etapa inicial de una actualización, el agente de actualización puede llevar a cabo una actualización de los módulos A, B y/o UA a la vez que los módulos C y D están activos, es decir, en un momento posterior de la secuencia de inicio, por ejemplo después del inicio 444 del módulo de carga D.

40 Durante una actualización 440, el agente de actualización realiza los cambios requeridos en el sistema. En algunas realizaciones, puede ser posible la ejecución del agente de actualización a la vez que el resto del sistema, por ejemplo los módulos de carga C y D, se están ejecutando. Sin embargo, el agente de actualización no depende de que los módulos de carga C y D se estén ejecutando. La tecnología de actualización particular implementada en el agente de actualización puede dictar si la mayoría del sistema debe ser apagada con anterioridad o si puede permitirse su ejecución durante al menos una parte del proceso de actualización.

45 Incluso aunque las Figuras 4 y 5 muestran un ejemplo donde el firmware está estructurado en cinco módulos de carga, se apreciará que en realizaciones alternativas el firmware puede dividirse en un número diferente de módulos de carga y/o las funciones del firmware pueden estar divididas entre los módulos de carga de un modo diferente. Por ejemplo, los módulos de carga A y B pueden combinarse en un único módulo de carga y/o las funciones del módulo B pueden dividirse entre una pluralidad de módulos de carga, y/o las funciones que no son requeridas por el agente de actualización, es decir, las funciones proporcionadas por los módulos de carga C y D en el ejemplo de las Figuras 4 y 5 pueden estar proporcionadas por un único módulo de carga o por un número mayor de módulos de carga.

50 La memoria persistente 212 del dispositivo electrónico puede tener además almacenada en la misma componentes de software adicional, por ejemplo software adicional, y/o datos como datos de configuración, datos generados y/o utilizados por el firmware y/o por el software de aplicación, etc. Por tanto, además del firmware 531, una porción 538 de la memoria 212 persistente puede estar ocupada por dichos componentes de software y/o datos. Se apreciará que el proceso de actualización descrito en el presente documento también puede actualizar partes o todo el software de aplicación y/o incluso partes o todos los datos. Por último, se muestra cómo la memoria 212 persistente de la Figura 5a tiene una porción 537 de memoria libre disponible que puede utilizarse para almacenar datos y/o software adicional.

60 El firmware y cualesquiera componentes de software adicionales se pueden almacenar en cualquier formato de archivo de sistema adecuado, que puede ser el mismo o diferente para el firmware y para los componentes de software adicionales. Por ejemplo, el firmware puede almacenarse en una estructura monolítica o no monolítica.

La Figura 5a muestra cómo los módulos de carga ocupan un espacio de memoria consecutivo. Sin embargo, se apreciará que la asignación de los módulos de memoria al espacio de memoria se puede realizar de un modo diferente. Es más, incluso aunque la Figura 5 muestra el código de inicio 421 almacenado en la misma memoria persistente que el firmware, se apreciará que el código de inicio 421 puede alternativamente almacenarse en una memoria diferente, por ejemplo, en una Memoria de Sólo Lectura (ROM). Similarmente, el software de aplicación/datos 538 y el espacio 537 de memoria libre pueden disponerse de un modo diferente del mostrado en el ejemplo de la Figura 5a, por ejemplo, se pueden distribuir por varias partes diferentes de la memoria.

Haciendo referencia ahora a la Figura 6 y continuando la referencia a las Figuras 4 y 5, se describirá un ejemplo de un proceso de actualización. En una operación inicial S601, se inicia el proceso de actualización. Por ejemplo, el proceso de actualización puede ser iniciado mediante una entrada adecuada del usuario, por ejemplo, cuando el usuario conecta el dispositivo electrónico a un sistema externo de actualización, cuando el usuario invoca la interfaz de usuario del agente de actualización, etc. Alternativamente, el proceso de actualización puede iniciarse de manera automática, por ejemplo, puede ser activado por un temporizador que provoca que el agente de actualización contacte un sistema de actualización remoto (por ejemplo, el sistema 102 descrito con relación a la Figura 1) para comprobar si hay disponible alguna actualización relevante, o puede ser activado por un mensaje recibido desde un sistema de actualización remoto indicando que hay una actualización disponible, o de cualquier otro modo.

En la operación S602, el dispositivo electrónico descarga bajo el control del agente de actualización un paquete 539 de actualización delta desde un sistema de actualización externo, por ejemplo el sistema 102 mostrado en la Figura 1. El proceso de actualización se puede llevar a cabo de cualquier modo conocido como tal en la técnica. En particular, el proceso de descarga puede comprender un paso de verificación durante el cual el dispositivo electrónico verifica la autenticidad y/o integridad del paquete delta recibido, por ejemplo según se describe en WO 2005/050441. Otros ejemplos de procesos de verificación adecuados incluyen cualquier método adecuado para verificar la integridad y/o autenticidad de los datos recibidos, por ejemplo por medio de sumas de control, CRCs, sumas de control criptográficas, firmas digitales, códigos de autenticación de mensajes, etc. El paquete 539 de actualización descargado se almacena en la memoria persistente 212.

La Figura 5b muestra la memoria persistente después de la descarga del paquete 539 de actualización delta. El paquete 539 de actualización delta incluye instrucciones de actualización para actualizar los módulos de carga respectivos del firmware 431. El paquete 539 de actualización delta comprende un primer conjunto de instrucciones de actualización para actualizar los módulos de carga A, B, y el agente de actualización UA, y un segundo conjunto de instrucciones de actualización para actualizar los módulos de carga C y D. Como el paquete de actualización delta sólo incluye instrucciones de actualización incrementales, su tamaño puede mantenerse relativamente pequeño, requiriendo así sólo un pequeño espacio de memoria libre 537.

En la operación 603, los módulos de carga requeridos por el agente de actualización, es decir, en este ejemplo los módulos A, B y el propio agente de actualización UA, son actualizados de un modo resistente a fallos utilizando el primer conjunto de instrucciones de actualización del paquete de actualización delta. Por ejemplo, el agente de actualización puede crear las versiones actualizadas 432', 433', 434' de los módulos de carga A, B y del propio agente de actualización UA, respectivamente. Por ejemplo, el agente de actualización puede copiar los módulos de carga A, B y UA en un espacio 537 disponible de la memoria persistente, y luego aplicar el primer conjunto de instrucciones de actualización delta a la versión actual de los módulos de carga para realizar una actualización en-sitio de las versiones actuales, por ejemplo mientras las versiones actuales de los módulos de carga A, B y UA se ejecutan desde una RAM del dispositivo electrónico. La generación de las versiones actualizadas 432'-434'' se ilustra esquemáticamente en la Figura 5c. Una vez las versiones actualizadas 432''-434'' se han generado y verificado con éxito, el agente de actualización puede borrar las versiones copiadas 432'-434', dando como resultado el esquema de memoria que se ilustra en la Figura 5f. Si la generación de las versiones actualizadas 432''-434'' se interrumpe o falla, el sistema puede arrancarse desde las versiones 432'-434' de copia de seguridad.

Se apreciará que la actualización resistente a fallos de los módulos A, B y UA se puede llevar a cabo de un modo diferente. En particular, la actualización se puede llevar a cabo de modo que la dirección de ejecución del código ejecutado durante la actualización no sea el mismo que la dirección de almacenamiento para la versión del primer conjunto de módulos de carga que se están actualizando. Esto se puede conseguir de diferentes maneras, por ejemplo se pueden copiar en una RAM y ejecutar, o se puede utilizar una unidad de gestión de memoria (MMU) para establecer una relación entre las direcciones de almacenamiento y una dirección de ejecución virtual, o bien el código puede ser reubicable de modo inherente, de modo que se pueda ejecutar desde cualquier dirección. Por ejemplo, el agente de actualización puede realizar una copia de seguridad de los módulos A, B y UA y llevar a cabo una actualización en-sitio sobre la versión de la copia de seguridad, permitiendo así que todo el sistema se ejecute incluso en un dispositivo electrónico donde el firmware no es almacenado en la RAM para su ejecución, sino que se ejecuta en-sitio. En tal realización, después de terminar la actualización con éxito, se puede modificar un registro maestro de inicio u otra referencia adecuada para que apunte a la versión actualizada antes de reiniciar, para provocar que el cargador de inicio ejecute la versión actualizada. En cualquier caso, la actualización de los módulos A, B, UA se lleva a cabo de tal modo que las versiones originales de A, B y UA están todavía disponibles hasta que la generación de las versiones actualizadas se ha completado con éxito. De este modo, si la generación de la versión actualizada falla o es interrumpida prematuramente, el dispositivo electrónico puede ser reiniciado

basándose en las versiones originales, y el proceso de actualización puede reanudarse o reiniciarse basándose en el paquete de actualización delta almacenado.

5 Incluso aunque en el presente ejemplo se indica que los tres módulos de carga A, B y UA son actualizados, se apreciará que algunas actualizaciones pueden afectar sólo a uno o algunos de los módulos de carga, dejando el resto inalterados. Se apreciará también que en dicha situación, sólo es necesario realizar copias de seguridad de aquellos módulos de carga que están realmente afectados por la actualización. Por ejemplo, en algunas realizaciones sólo los módulos de carga A, B, C y D puede estar afectados, quedándose el agente de actualización inalterado. En otra actualización, puede estar afectados sólo los módulos de carga C y/o D, etc. Es más, si una actualización afecta solamente a los módulos de carga C y D, se pueden saltar las operaciones S603-S605. A efectos de la presente descripción, se hace referencia a las versiones de todos los módulos de carga después de una actualización como versiones actualizadas, independientemente de si cada módulo de carga ha sido realmente modificado por la actualización. Por tanto, en algunas situaciones la versión actualizada de uno o más de los módulos de carga puede ser idéntica a la versión actual del módulo de carga correspondiente.

15 Cuando se ha completado con éxito la actualización resistente a fallos de los módulos de carga A, B y UA, el proceso continúa en la operación S604 y envía una señal para provocar que el dispositivo cargue los nuevos módulos de carga A', B' y UA' durante la siguiente secuencia de inicio. Las copias de seguridad realizadas durante el proceso de actualización se pueden eliminar. En algunas realizaciones, incluso el primer conjunto de instrucciones de actualización se puede quitar de la memoria en este momento.

20 En la siguiente operación S605, el dispositivo electrónico es reiniciado, por ejemplo utilizando el proceso de inicio mostrado en la Figura 4 pero basándose en las versiones actualizadas A', B' y UA' de los módulos de carga requeridos para el proceso de actualización.

25 En la siguiente operación S606, se actualizan los módulos de carga restantes (es decir, los módulos de carga C y D en el presente ejemplo) que requieren ser actualizados mientras se están ejecutando sólo las versiones actualizadas del agente de actualización UA' y los módulos de carga A' y B'. Esto se puede llevar a cabo aplicando el segundo conjunto de instrucciones de actualización del paquete 539 de actualización delta para llevar a cabo una actualización en-sitio de los módulos de carga C y D, por ejemplo como se ilustra esquemáticamente en la Figura 5e, dando como resultado un esquema de memoria actualizada según se ilustra en la Figura 5f. En general, el término actualización en-sitio hace referencia a una actualización donde las versiones actualizadas se generan encima de la versión original, dando como resultado que la versión actualizada ocupa al menos parte del espacio que anteriormente ocupada la versión antigua. Por tanto, la operación S606 da como resultado la sustitución de los módulos de carga C y D por sus respectivas versiones actualizadas C' (435') y D' (436'). Si se interrumpe la actualización durante esta etapa, por ejemplo si la batería del dispositivo integrado se extrae, la actualización continuará cuando el dispositivo integrado sea reiniciado sin la necesidad de un agente de actualización autónomo y sin la necesidad de reconectar ningún sistema externo.

35 La aplicación del archivo delta para crear versiones actualizadas de los módulos de carga en las operaciones S603 y S606 puede llevarse a cabo de cualquier modo adecuado, por ejemplo como se describe en WO 2007/071324, que describe un ejemplo de un proceso para generar actualizaciones delta para módulos de carga, y un ejemplo de un agente de actualización para aplicar dichas actualizaciones delta.

40 Después de completar esta operación con éxito, el proceso continúa en la operación S607, donde el agente de actualización UA' puede eliminar cualquier copia de seguridad y el paquete de actualización delta almacenado, y enviar una señal de listo al módulo de carga A' principal, provocando que el módulo de carga A' principal inicie los módulos de carga actualizados C' y D', como se ilustra en la Figura 4. Después de haber terminado con éxito el proceso de actualización, la versión actualizada del agente de actualización puede cerrarse o continuar ejecutándose, por ejemplo para poder iniciar una actualización subsiguiente de los módulos de carga A, B y/o UA como un proceso en segundo plano.

45 Por tanto, en la realización de un proceso de actualización según se ha descrito anteriormente, después de la recepción y almacenamiento de las instrucciones de actualización, el proceso de actualización se lleva a cabo en dos etapas: en una primera etapa se efectúa cualquier actualización de los módulos de carga que sea requerida por el agente de actualización, así como cualquier actualización en el propio agente de actualización. Estas actualizaciones se llevan a cabo de un modo resistente a fallos, de modo que siempre está presente en el dispositivo electrónico una versión operacional de los módulos de carga requeridos por el agente de actualización. Luego, el dispositivo electrónico es reiniciado, por ejemplo utilizando la secuencia de inicio de la Figura 5, y se actualizan los módulos restantes.

55 Aunque algunas de las realizaciones se han descrito y mostrado en detalle, la invención no se restringe a los mismos, sino que también puede estar configurada de otros modos dentro del ámbito definido por las siguientes reivindicaciones. En particular, incluso aunque el proceso de actualización descrito en el presente documento se ha descrito principalmente con la actualización de firmware, se apreciará que el proceso de actualización también se puede aplicar a actualizaciones que incluyen otros componentes de software como software de aplicación. Además, es necesario remarcar que las realizaciones anteriores se han descrito principalmente con referencia a una memoria

flash. Sin embargo, se debe entender que el método descrito en el presente documento también se puede implementar utilizando otros tipos de medios de almacenamiento, como discos ópticos, discos duros, discos flexibles, cintas, y/o cualquier otro tipo de medio de almacenamiento magnético y/o óptico.

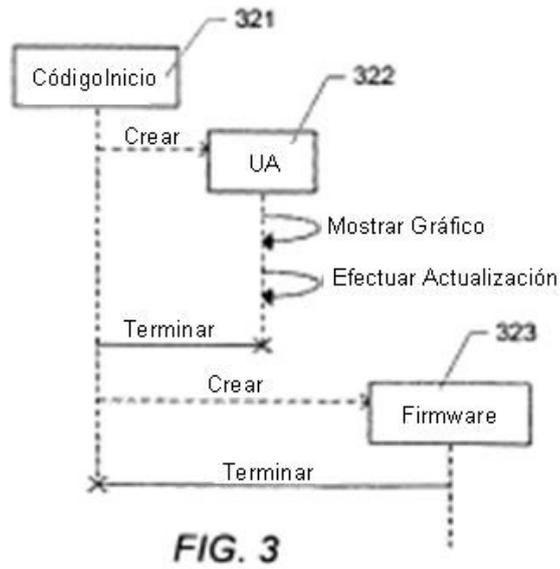
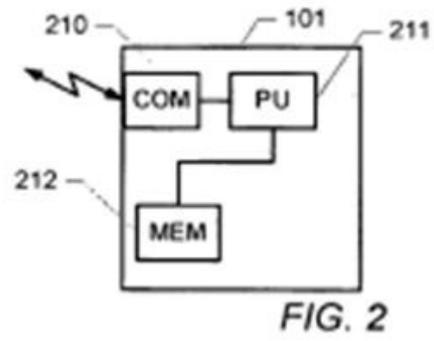
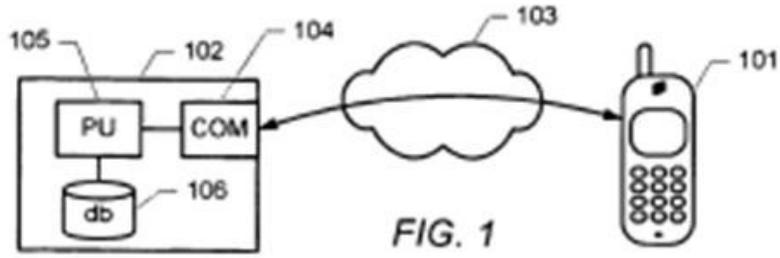
- 5 La invención se puede implementar por medio de hardware que comprende varios elementos diferentes, y por medio de un ordenador u otro dispositivo de procesamiento adecuadamente programado. En las reivindicaciones de dispositivo que enumeran múltiples medios, varios de estos medios pueden estar integrados en una única pieza de hardware, por ejemplo un microprocesador u ordenador adecuadamente programado, y/o una o más interfaces de comunicación, según se describe en el presente documento. El mero hecho de que ciertas medidas estén descritas en reivindicaciones dependientes diferentes o descritas en realizaciones diferentes no indica que no sea posible utilizar ventajosamente una combinación de tales medidas.
- 10

Se debe también remarcar que el término "comprende/que comprende", cuando se utiliza en esta memoria, pretende especificar la presencia de las características, partes, operaciones o componentes, pero no impide la presencia o adición de una o más características, partes, operaciones, componentes, o grupos de los mismos.

REIVINDICACIONES

- 5 1. Un método para actualizar el software de un dispositivo electrónico desde una versión actual del software a una versión actualizada del software, comprendiendo el dispositivo electrónico un medio de almacenamiento que tiene almacenado en el mismo la versión actual del software; comprendiendo el método que el dispositivo electrónico lleve a cabo las siguientes operaciones:
- recibir instrucciones de actualización incrementales para provocar que el dispositivo electrónico transforme la versión actual almacenada del software en la versión actualizada del software;
- 10 donde el software comprende un primer y un segundo conjunto de módulos de carga, comprendiendo el primer conjunto de módulos de carga código de programa para proporcionar un mínimo conjunto de servicios de sistema básicos requeridos para hacer funcionar el dispositivo electrónico en un modo de actualización y código de programa para proporcionar funciones para llevar a cabo la actualización, comprendiendo las instrucciones de actualización incrementales un primer conjunto de instrucciones de actualización para provocar que el dispositivo electrónico genera una versión actualizada del primer conjunto de módulos de carga, comprendiendo además las instrucciones de actualización incremental recibidas un segundo conjunto de instrucciones de actualización para provocar que el dispositivo electrónico genere una versión actualizada del segundo conjunto de módulos de carga;
- 15 - almacenar al menos el segundo conjunto de instrucciones de actualización recibidas en el medio de almacenamiento;
- 20 - ejecutar el primer conjunto de instrucciones de actualización para actualizar la versión actual almacenada del primer conjunto de módulos de carga con la versión actualizada generada del primer conjunto de módulos de carga
- reiniciar el dispositivo electrónico en el modo de actualización en el que se ejecuta la versión actualizada del primer conjunto de módulos de carga;
- 25 - ejecutar el segundo conjunto de instrucciones de actualización almacenado de modo que se lleva a cabo una actualización en-sitio del segundo conjunto de módulos de carga, comprendiendo la actualización en-sitio reemplazar la versión actual almacenada del segundo conjunto de módulos de carga con la versión actualizada generada del segundo conjunto de módulos de carga
- 30 en donde el dispositivo electrónico es un dispositivo portátil de comunicaciones por radio para la comunicación a través de una red de comunicaciones por radio, donde el software incluye funciones de comunicación para establecer comunicaciones por radio entre el dispositivo de comunicaciones y otros dispositivos de comunicaciones de la red de comunicaciones por radio; y donde dichas funciones de comunicación están incluidas en el segundo conjunto de módulos de carga.
- 35 2. Un método de acuerdo con la reivindicación 1, donde el primer conjunto de módulos de carga comprende un módulo de carga principal adaptado para proporcionar un conjunto mínimo de servicios de sistema básicos requeridos para el funcionamiento del dispositivo electrónico, y un segundo módulo de carga adaptado para proporcionar un conjunto mínimo de servicios básicos requeridos para llevar a cabo la actualización.
- 40 3. Un método de acuerdo con la reivindicación 2, donde el módulo de carga principal comprende un sistema operativo principal del dispositivo electrónico y uno o más controladores físicos requeridos.
4. Un método de acuerdo con la reivindicación 2 ó 3, donde el segundo módulo de carga comprende código de programa requerido para controlar el dispositivo electrónico para indicar un progreso del proceso de actualización a un usuario del dispositivo electrónico.
- 45 5. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde el primer conjunto de módulos de carga comprende un módulo de carga de agente de actualización para controlar el proceso de actualización; y donde ejecutar el segundo conjunto de instrucciones de actualización almacenadas comprende ejecutar el segundo conjunto de instrucciones de actualización almacenadas bajo el control del módulo de carga de agente de actualización de la versión actualizada del primer conjunto de módulos de carga.
- 50 6. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde reiniciar el dispositivo electrónico en el modo de actualización comprende ejecutar sólo el primer conjunto de módulos de carga, y donde ejecutar el primer conjunto de módulos de carga comprende ejecutar la versión actualizada del primer conjunto de módulos de carga.
7. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde el dispositivo electrónico tiene además almacenado en el mismo un programa de carga de inicio, y donde reiniciar el dispositivo electrónico en el modo de actualización comprende:

- ejecutar el programa de carga de inicio,
 - iniciar la ejecución de al menos uno de entre el primer conjunto de módulos de carga bajo el control del programa de carga de inicio,
 - 5 - iniciar la ejecución del módulo de carga de agente de actualización bajo el control de al menos uno de entre el primer conjunto de módulos de carga.
8. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, que además comprende enviar una señal después de haber completado con éxito la operación de ejecutar el primer conjunto de instrucciones de actualización, indicando dicha señal al dispositivo electrónico que durante el siguiente procedimiento de reinicio el dispositivo electrónico se debe reiniciar en el modo de actualización.
- 10 9. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde el software incluye firmware.
10. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde el dispositivo electrónico es operacional en el modo de actualización sin la ejecución del segundo conjunto de módulos de carga.
11. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde el medio de almacenamiento es una memoria persistente.
- 15 12. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde ejecutar el primer conjunto de instrucciones de actualización para actualizar la versión actual almacenada del primer conjunto de módulos de carga con la versión actualizada generada del primer conjunto de módulos de carga comprende:
- 20 - crear una copia de seguridad del primer conjunto de módulos de carga y ejecutar el primer conjunto de instrucciones de actualización para reemplazar la copia de seguridad almacenada del primer conjunto de módulos de carga con la versión actualizada generada del primer conjunto de módulos de carga;
 - intercambiar las versiones de copia de seguridad y original del primer conjunto de módulos de carga.
13. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, que además comprende ejecutar las versiones actualizadas del segundo conjunto de módulos de carga.
- 25 14. Un método de acuerdo con cualquiera de las reivindicaciones precedentes, donde almacenar al menos el segundo conjunto de instrucciones de actualización en el medio de almacenamiento comprende almacenar el primer y segundo conjuntos de instrucciones de actualización en el medio de almacenamiento.
15. Un dispositivo electrónico controlable por software, comprendiendo el dispositivo electrónico un medio de almacenamiento que tiene almacenada en el mismo una versión actual del software, donde el dispositivo electrónico está configurado para llevar a cabo los pasos del método de acuerdo con cualquiera de las reivindicaciones 1 a 15.
- 30 16. Un dispositivo de acuerdo con la reivindicación 15, donde el dispositivo electrónico es un dispositivo portátil de comunicaciones por radio.
- 35 17. Un producto de programa de ordenador que comprende un programa de carga de inicio y software, comprendiendo el producto de programa de ordenador medios de código de programa adaptado para provocar que un dispositivo electrónico lleve a cabo las operaciones del método de acuerdo con cualquiera de las reivindicaciones 1 a 14 cuando dichos medios de código de programa son ejecutados por el dispositivo electrónico.



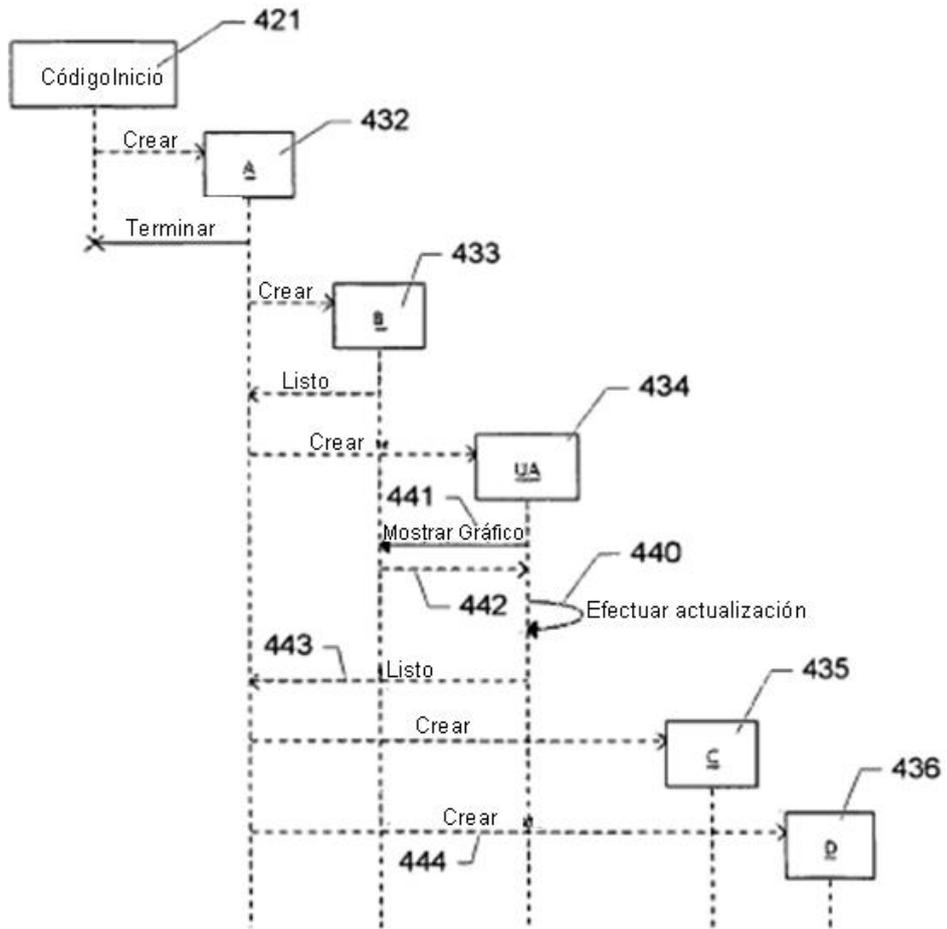


FIG. 4

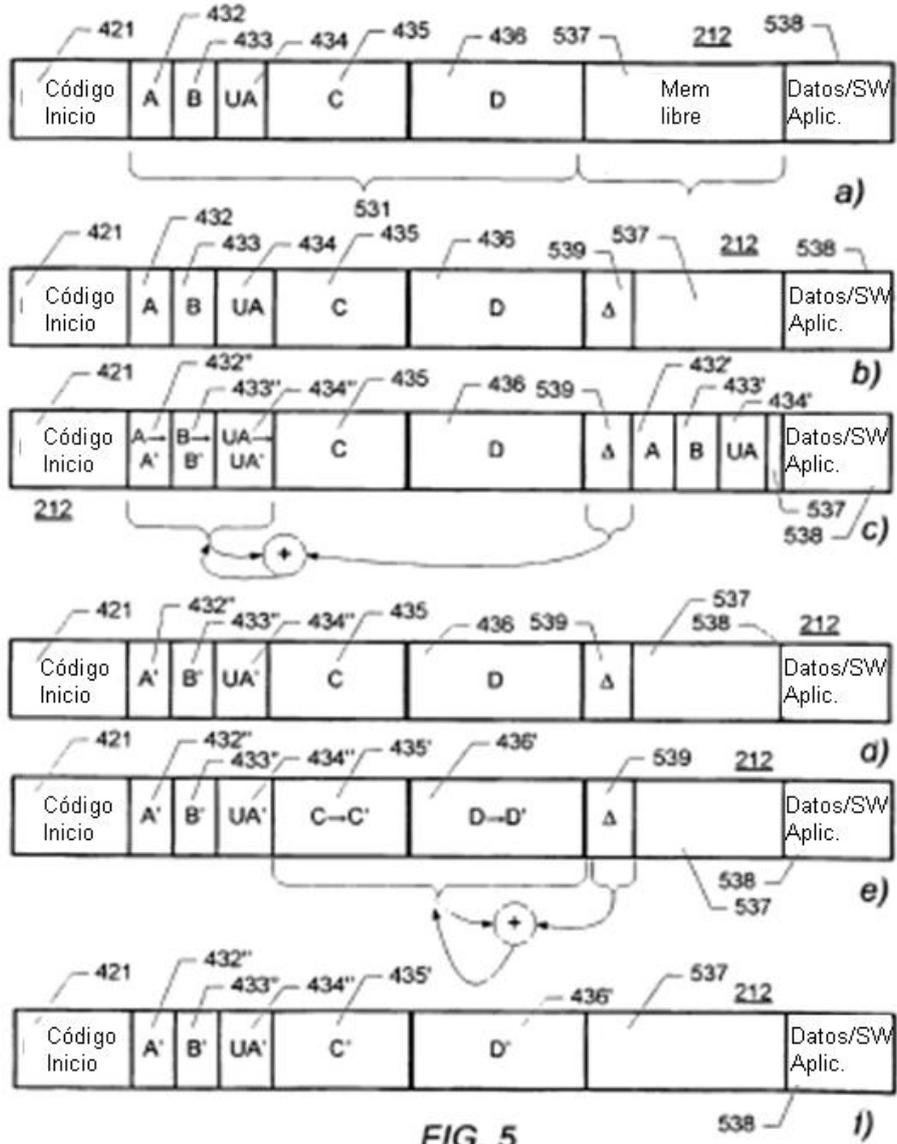


FIG. 5

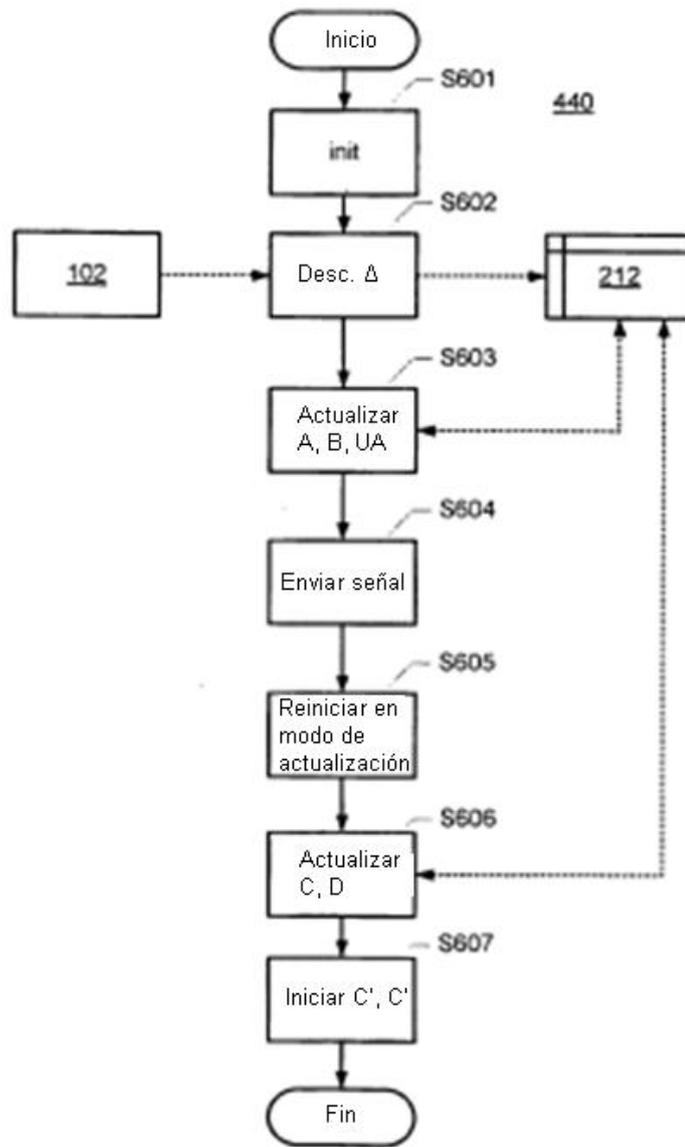


FIG. 6