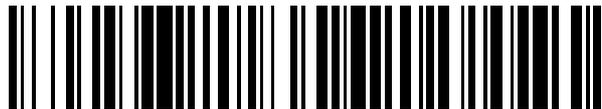


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 666 472**

51 Int. Cl.:

H04L 29/06 (2006.01)

H04M 7/00 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 86 Fecha de presentación y número de la solicitud internacional: **11.10.2001 PCT/US2001/31809**
- 87 Fecha y número de publicación internacional: **03.10.2017 WO02031669**
- 96 Fecha de presentación y número de la solicitud europea: **11.10.2001 E 01979709 (1)**
- 97 Fecha y número de publicación de la concesión europea: **07.03.2018 EP 1332435**

54 Título: **Aparato y sistema de telefonía de red que soportan telefonía inalámbrica por Internet**

30 Prioridad:

11.10.2000 US 240869 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
04.05.2018

73 Titular/es:

**THE TRUSTEES OF COLUMBIA UNIVERSITY IN
THE CITY OF NEW YORK (100.0%)
West 116th Street and Broadway New York
New York 10027, US**

72 Inventor/es:

SCHULZRINNE, HENNING

74 Agente/Representante:

LINAGE GONZÁLEZ, Rafael

ES 2 666 472 T3

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

DESCRIPCIÓN

Aparato y sistema de telefonía de red que soportan telefonía inalámbrica por Internet

5 **Campo de la invención**

El presente invento se refiere en general al campo de la telefonía de Internet e Intranet y más particularmente se refiere a un dispositivo y sistema de telecomunicaciones de red que soportan servicios de telefonía de red tanto alámbricos como inalámbricos.

10

Antecedentes de la invención

Internet ha evolucionado desde un medio adicional conveniente de comunicaciones a una herramienta de comunicación esencial. A este respecto, un segmento creciente de Internet se refiere a la telefonía por Internet que proporciona una serie de ventajas sobre la red conmutada por circuitos convencional controlada por una red de señalización separada. Por un lado, las partes pueden seleccionar y usar más fácilmente la codificación y otras técnicas de compresión de datos que sean más apropiadas para sus necesidades de calidad. Las partes pueden, por ejemplo, decidir que, para llamadas internacionales, intercambien un menor coste por la calidad interurbana total, mientras que un periodista que cuente su historia a una estación de radio puede optar por la calidad total de FM sin tener en cuenta el precio. Incluso sin degradación de la calidad, 5,3 kb/s (G. 723.1) a 8 kb/s (G. 729) son suficientes para soportar una calidad interurbana frente a 64 kb/s para las redes telefónicas convencionales. Esta flexibilidad también tiene la ventaja de que durante una sobrecarga de red severa, por ejemplo después de una catástrofe natural, los clientes de telefonía aún pueden comunicarse a aproximadamente 3 kb/s, aumentando así la capacidad de la red en veinte veces.

25

Junto con el crecimiento de la telefonía por Internet, ha habido un crecimiento en la telefonía inalámbrica y el acceso inalámbrico a Internet. Estas tecnologías han liberado a millones de usuarios al proporcionar capacidades de comunicaciones de voz y datos que ya no están atadas al escritorio de la oficina o al hogar. Estos servicios han evolucionado desde simples conexiones telefónicas de voz hasta servicios de mensajería, contenido transmitido, acceso inalámbrico a Internet y similares. Tales servicios se conocen como servicios inalámbricos de segunda generación (2G), segunda generación extendida (2.5G) y tercera generación (3G), dependiendo de la naturaleza y la sofisticación del servicio que se ofrece. A medida que evolucionan las tecnologías inalámbricas, se han propuesto varios estándares para tales dispositivos.

30

Uno de estos protocolos es el protocolo de aplicaciones inalámbricas (WAP), que es una especificación abierta que ofrece un método estándar para acceder a contenido y servicios basados en Internet desde dispositivos inalámbricos, tales como teléfonos móviles y PDA (asistentes digitales personales). El modelo WAP intenta proporcionar una interfaz que es similar al ordenador de escritorio tradicional de Internet, pero formatea el contenido de una manera adecuada para las pantallas más pequeñas y más limitadas generalmente disponibles en dispositivos portátiles. Los dispositivos WAP portátiles generalmente incluyen un software de micronavegador que proporciona visualización de contenido y funcionalidad de navegación de red. Para realizar esta funcionalidad, el contenido WAP está escrito en un lenguaje de marcado llamado WML (lenguaje de marcado inalámbrico). Una extensión de WML, WMLScript, habilita aún más la inteligencia del lado del cliente. El protocolo WAP está destinado a ser independiente de la red y el sistema operativo.

35

40

45

Una característica de los sistemas actuales de telefonía inalámbrica es la provisión para intercambiar mensajes cortos de texto. Por ejemplo, el estándar de mensaje corto (SMS) es una adición al estándar GSM que permite enviar mensajes de texto de hasta 160 caracteres en redes GSM y 190 caracteres en otras redes entre teléfonos móviles. El SMS ha ido ganando popularidad debido a su bajo coste y rápida transferencia de mensajes. Además de los mensajes entre teléfonos móviles, el SMS se puede incorporar a las aplicaciones para alertar a los usuarios de los eventos, como la llegada de un nuevo correo electrónico o un movimiento de los precios de las acciones. Los mensajes SMS se transfieren entre teléfonos móviles a través de un centro de servicio de mensajes cortos (SMSC). El SMSC es un software que reside en la red de operadores y gestiona los procesos, incluida la puesta en cola de los mensajes, la facturación al remitente y la devolución de los recibos, si es necesario.

50

55

Además, Bluetooth es un estándar abierto para comunicaciones de radio bidireccionales de onda corta entre diferentes dispositivos, como teléfonos móviles, ordenadores de mano, PC portátiles e impresoras. El protocolo Bluetooth permite que la información entre dichos dispositivos se sincronice. Por ejemplo, la información de la agenda contenida en un PDA se puede actualizar automáticamente cuando se encuentra dentro del alcance de un PC con Bluetooth.

60

Si bien existen numerosos protocolos para diversas características inalámbricas, sigue existiendo la necesidad de integrar dichos protocolos en un dispositivo o sistema de telefonía de red que pueda integrar fácilmente las características de Internet con aquellas características de los sistemas de telefonía convencionales e inalámbricos.

65

El documento WO 00/02403 A divulga un método y disposición para dirigir información en una red de comunicación

móvil. Este sistema dirige cualquier mensaje entrante recibido en un aparato y convertido por una pasarela con un servidor de dirección. Las direcciones son URL o identificadores que se correlacionan con URL para páginas web que se han actualizado recientemente. El usuario puede entonces acceder a la página web a través de un explorador WWW en base al URL o el identificador proporcionados.

5 Además, el documento WO 99/35595 A divulga un método y sistema para el navegador de páginas de hipertexto con hiperenlaces en los que la página web se convierte en texto de la página web y una lista de los hiperenlaces en la página web, que se envían al teléfono GSM de forma separada. De este modo, cualquier mensaje presuntamente recibido en el aparato de red y convertido por un servidor proxy no se refiere a contenido publicitario y en cambio son texto de página web (excluidos los gráficos) e hiperenlaces desde la página web.

15 El documento WO 99/66747 A divulga un servicio y sistema de emisión de hiperenlace de titulares que usa recursos de radiocomunicación para proporcionar información a un usuario de estación móvil. Se describe un método para emitir mensajes cortos o titulares (por ejemplo noticias, cotizaciones bursátiles, deportes, tiempo, y viajes) con hiperenlaces seleccionables para recibir información adicional. Cualquier mensaje presuntamente recibido en un aparato de red y trasladado por un servidor de servicio no se relaciona con contenido publicitario. El servicio puede actuar como un traductor que pasa solicitudes de información al equipo del proveedor de servicios.

20 **Sumario de la invención**

La presente invención es definida por el tema de las reivindicaciones independientes. Un sistema de comunicación de acuerdo con la presente invención incluye al menos un dispositivo de aparato de red que está acoplado a una red. El dispositivo de aparato de red incluye un software para detectar llamadas entrantes e iniciar sesiones de llamadas de acuerdo con un protocolo de señalización. El sistema incluye además una pasarela de comunicación inalámbrica acoplada a la red para permitir que el dispositivo de aparato de red se comunique de acuerdo con un protocolo de red inalámbrica. La pasarela de comunicación inalámbrica incluye, o está acoplada a, un proxy de comunicación para trasladar mensajes entre el protocolo de señalización y el protocolo de red inalámbrica. El dispositivo de aparato de red comprende una pantalla, y el dispositivo de aparato de red recibe mensajes trasladados por el proxy de comunicación inalámbrica que se relaciona con contenido publicitario para salida en la pantalla.

En una realización, el protocolo de señalización toma la forma del protocolo de inicio de sesión (SIP) y el protocolo de red inalámbrica toma la forma del protocolo WAP. Por supuesto, se pueden emplear otros protocolos de red inalámbrica, como los usados para habilitar los servicios de telefonía inalámbrica 2G, 2.5G y 3G.

35 El sistema puede incluir además un servidor de señalización, como un servidor proxy SIP. El servidor proxy SIP generalmente funciona como un intermediario entre los dispositivos de red individuales y la pasarela de comunicación.

40 También de acuerdo con la presente invención es un método de comunicación con un dispositivo de aparato de red en la red. El método incluye accionar un dispositivo de red para detectar llamadas entrantes e iniciar sesiones de llamadas de acuerdo con un primer protocolo. El método incluye además trasladar mensajes entre el primer protocolo y un segundo protocolo, usando un proxy de comunicación, para permitir que el dispositivo de aparato de red se comunique de acuerdo con un segundo protocolo. Generalmente, el primer protocolo es un protocolo de señalización, tal como SIP y el segundo protocolo es un protocolo de comunicaciones inalámbricas, tal como WAP.

Otros objetos, características y ventajas de la invención se harán evidentes a partir de la siguiente descripción detallada tomada junto con las figuras adjuntas que muestran realizaciones ilustrativas de la invención.

50 **Breve descripción de los dibujos**

Para una comprensión completa de la presente invención y sus ventajas, ahora se hace referencia a la siguiente descripción tomada junto con los dibujos que se acompañan en los que los mismos números de referencia indican características similares y en los que:

55 la figura 1 es un diagrama ilustrativo de un sistema de telecomunicaciones que presenta una red de voz conmutada por circuito convencional operativamente acoplada a una red de paquetes de voz;

60 la figura 2 es un diagrama de bloques de un sistema de telefonía de red de datos por paquetes;

la figura 3 es un diagrama que muestra una pila de protocolos para dispositivos telefónicos que funcionan en el sistema de telefonía de la red de datos por paquetes de la figura 2;

65 la figura 4 es un diagrama de bloques de una arquitectura de hardware preferida de un aparato de telefonía de red de acuerdo con la presente invención;

ES 2 666 472 T3

- la figura 5 es un diagrama de bloques que ilustra adicionalmente el aparato de telefonía de red de la figura 4;
- la figura 6 es un mapa de memoria de ejemplo para el DSP del aparato de telefonía de red de la figura 5;
- 5 la figura 7 es un diagrama de bloques de una interfaz de memoria para el DSP del aparato de telefonía de red de la figura 5;
- la figura 8 es un diagrama de bloques de una interfaz de controlador de red para el DSP del aparato de telefonía de red. 5;
- 10 la figura 9 es un diagrama de bloques de una interfaz de códec para el DSP del aparato de telefonía de red de la figura 5;
- 15 la figura 10 es un mapa de memoria de ejemplo para el DSP de la figura 5 que muestra un mapeo de la interfaz de control de LCD a las direcciones de memoria de DSP;
- la figura 11 es un diagrama de bloques que muestra la arquitectura de software para el aparato de telefonía de red de la figura 4;
- 20 la figura 12 es un diagrama de bloques que muestra los mecanismos de planificación del software de nivel de proceso de la figura 11;
- las figuras 13A-F son tablas que ilustran definiciones de tareas de ejemplo para operaciones de software de un método preferido para accionar el teléfono de red de datos por paquetes de acuerdo con las arquitecturas de hardware y software de las figuras 4 y 11;
- 25 la figura 14 es un diagrama de flujo de un procedimiento de salida de solicitud ARP de acuerdo con las arquitecturas de hardware y software de las figuras 4,11 y 13;
- 30 la figura 15 es un diagrama de flujo de un procedimiento de entrada de solicitud de ARP de acuerdo con las arquitecturas de hardware y software de las figuras 4,11 y 13;
- la figura 16 es un diagrama que muestra los pasos de procesamiento de IP de acuerdo con las arquitecturas de hardware y software de las figuras 4,11 y 13;
- 35 la figura 17 es una lista de estructuras de datos de transmisión de Ethernet de ejemplo de acuerdo con la arquitectura de software de la figura 11;
- la figura 18 es un diagrama de flujo de datos de un procedimiento de envío de paquetes de acuerdo con las arquitecturas de hardware y software de las figuras 4, 11 y 13;
- 40 la figura 19 es un diagrama de flujo de datos de un procedimiento de recepción de paquetes de acuerdo con las arquitecturas de hardware y software de las figuras 4,11 y 13;
- 45 las figuras 20A y 20B muestran el esquema de búfer de "ping-pong" A/D y D/A usado por el software del presente aparato de telefonía de red;
- la figura 21 es un diagrama de transición de estado del proceso de Call_task del presente aparato de telefonía de red;
- 50 la figura 22 es un cuadro que define los valores de teclado para la realización preferida del teléfono de red de datos por paquetes de la figura 5;
- la figura 23 es una estructura de datos que ilustra las definiciones de estado clave para la realización preferida del presente aparato de telefonía de red de la figura 5;
- 55 la figura 24 es un mapeo del puerto paralelo de E/S del dispositivo de telefonía de red de la figura 5;
- la figura 25 es una estructura de datos que define los estados del controlador de Ethernet del aparato de telefonía de red de la figura 5;
- 60 la figura 26 es una estructura de encabezado RTP de ejemplo para el procesamiento de paquetes RTP usado en el aparato de telefonía de red de la figura 5;
- 65 la figura 27 es una estructura de datos para usar con una función de generación de tono del teléfono de red de datos por paquetes de la figura 5;

la figura 28 es un diagrama de temporización para la función de generación de tono del aparato de telefonía de red de la figura 5;

5 la figura 29 es una lista de estructuras de datos usadas para procesar las solicitudes SIP_task o respuestas de acuerdo con el aparato de telefonía de red de la figura 5;

la figura 30 es un diagrama de transición de estado que ilustra el aparato de telefonía de red que funciona como un cliente (iniciando una llamada) de acuerdo con la figura 5;

10 la figura 31 es una lista de respuestas de SIP_task de acuerdo con el aparato de telefonía de red de la figura 5;

la figura 32 es un diagrama de estado que ilustra el diagrama de transición de estado de un UAS de SIP de acuerdo con el aparato de telefonía de red de la figura 5; y

15 la figura 33 es un diagrama de bloques que ilustra parte de un sistema de telefonía de red de datos por paquetes que incluye uno o más aparatos de telefonía de red de acuerdo con la presente invención; y

20 la figura 34 es un diagrama de bloques que ilustra una arquitectura de red que soporta protocolos de comunicaciones inalámbricas en un sistema de telefonía que tiene al menos un aparato de red.

Descripción detallada de la invención

25 La figura 1 es un diagrama de bloques que ilustra un sistema de telecomunicaciones que tiene componentes de telefonía convencional y telefonía por paquetes. Como se muestra en la figura 1, el sistema incluye una red 20 de voz conmutada por circuito acoplada a una red 30 de paquetes a través de una primera pasarela 12. La figura muestra al menos tres posibles interacciones entre los servicios de telefonía de Internet y un sistema convencional de "servicio de telefonía antiguo simple" (POTS): entrega de paquetes "de extremo a extremo"; entrega de "salida de extremo a extremo"; y entrega local de paquetes. Con la entrega de paquetes de "extremo a extremo", los sistemas de extremo como los ordenadores en red, los teléfonos dedicados de Internet o los ordenadores personales (PC) se usan para empaquetar audio y entregar paquetes de audio a uno o más sistemas de extremo similares para su reproducción. Con la entrega "salida de extremo a extremo", las redes de paquetes se usan para la transmisión de voz de larga distancia, mientras que los circuitos de voz estándar conmutados por circuito se usan para conectar equipos locales del cliente (CPE), es decir, teléfonos analógicos estándar, a las pasarelas de telefonía por paquetes. La "salida de extremo a extremo" se puede usar tanto para circuitos de voz individuales como para interconexiones PBX, y permite eludir los servicios de larga distancia convencionales, así como la interconexión de equipos POTS a sistemas de audio por paquetes. Con la entrega local de paquetes, los datos de voz se generan mediante sistemas de extremo de audio por paquetes, pero se transportan como voz conmutada por circuito sobre instalaciones públicas o alquiladas.

40 La figura 2 ilustra una realización de un sistema 50 de telefonía de red de datos por paquetes de acuerdo con la presente invención. El sistema de telefonía de red de datos por paquetes incluye: una LAN 52 de Ethernet, teléfonos 54, 56 y 58 de Ethernet, una estación 60 de trabajo, un servidor 62 y una pasarela 64 de Ethernet. Los teléfonos de Ethernet son dispositivos de red, que pueden tomar la forma de dispositivos independientes, como un aparato de red o un sistema de ordenador personal con periféricos de entrada y salida de audio y que funcionan bajo el control de un programa informático apropiado. Con tal enfoque de red de datos por paquetes, el tráfico de datos de voz se empaqueta cerca del usuario final. El sistema de telefonía de red de datos por paquetes de la figura 2, por ejemplo, puede incluir varias docenas de hogares, oficinas o apartamentos que están conectados a una pluralidad de pasarelas de Ethernet (solo una se muestra en la figura 2), cada una de las cuales se ubica dentro del límite de distancia de cableado CAT-3S de 328 pies (99,97 m) desde la unidad de terminación de red. Las pasarelas pueden, a su vez, conectarse a través de fibra óptica al conmutador del vecindario (no mostrado), o conectarse directamente a la red telefónica pública conmutada (PSTN) a través de las líneas 66 como se muestra en la figura 2. Esta arquitectura tiene la ventaja de que se puede acomodar una combinación de clientes con ancho de banda bajo y ancho de banda alto sin necesidad de cables adicionales. Como los costes de los conmutadores están dominados por recuentos de interfaz en lugar de ancho de banda, este mecanismo ofrece un ancho de banda por usuario mucho mayor (en particular, el ancho de banda máximo), pero los costes de conmutación son similares a las redes telefónicas actuales.

60 En la arquitectura de la figura 2, cada dispositivo de red incluye una dirección de red y cada dispositivo puede acceder directamente a cualquier otro dispositivo de red a través de la dirección de red. Si bien un servidor especializado, como un servidor proxy o un servidor de redirección, puede ser deseable para implementar ciertas características, no es necesario establecer una sesión de llamada, es decir, comunicaciones de datos punto a punto entre dos o más dispositivos de red.

65 El uso de una LAN 52 de red de datos por paquetes es ventajoso en el sentido de que es una solución relativamente económica en la que se pueden usar interfaces de PC y hardware de red convencionales. La LAN 52 de red de

datos por paquetes puede funcionar en una variedad de medios y permite añadir fácilmente más dispositivos en una LAN de acceso múltiple. La pasarela 64 puede ser un único DSP que actúa como un simple módulo de voz por paquetes y que implementa el reconocimiento DTMF para la señalización de usuario a red.

5 La figura 3 es un diagrama de bloques que ilustra un diagrama de pila de protocolos de red de datos por paquetes para proporcionar servicios de telefonía por Internet y otros medios continuos ("medios de transmisión") tales como "radio por Internet" y "TV por Internet". Como saben y entienden los expertos en la técnica, un "protocolo" generalmente es un conjunto de reglas para comunicarse entre ordenadores. Como tal, los protocolos rigen el formato, el tiempo, la secuencia y el control de errores. El término "pila" se refiere al software real que procesa los
10 protocolos y, por lo tanto, permite el uso de un conjunto o conjuntos de protocolos específicos. El diagrama mostrado en la figura 3 ilustra una jerarquía en capas entre los diversos protocolos usados.

La pila 80 de protocolos de la figura 3 incorpora una serie de protocolos en capas que incluyen: un protocolo base 82 para proporcionar formato de mensaje de Ethernet básico e información de temporización; un protocolo 84 de resolución de dirección (ARP) para interconectarse con el protocolo base 82 y para trasladar direcciones IP en direcciones de control de acceso al medio (MAC); una capa 86 de red de protocolo de Internet (IP) para interactuar con el protocolo base 82; un protocolo 88 de configuración dinámica de host (DHCP) opcional para interconectar con el protocolo base 82; y un protocolo 90 de datagrama de usuario (UDP) para interconectarse con los protocolos ARP 84, IP 86 y DHCP 88 y para el transporte en tiempo real de los datos y controles de la aplicación. La pila 80 de
20 protocolos incluye además los siguientes protocolos específicos de aplicación para codificar información de voz: un protocolo 92 de protocolo de transporte en tiempo real (RTP) para el transporte de datos de audio en tiempo real, en el que el protocolo RTP 92 generalmente interactúa con el UDP 90 y la modulación, las aplicaciones 94, 96 y 98 de control y códec de voz, respectivamente. Los protocolos 94 y 96 de aplicación pueden tomar varias formas, tales como los protocolos de modulación de código de pulso G. 711 y de códec de voz G. 723, respectivamente. Además,
25 se puede incluir la capa 97 del protocolo de transmisión en tiempo real (RTSP) para proporcionar un rendimiento mejorado en aplicaciones de medios de transmisión. El protocolo 98 de control se usa para la iniciación y señalización de la sesión y preferiblemente toma la forma del protocolo de inicio de sesión (SIP). La pila de protocolos puede incluir además una capa para proporcionar funcionalidad inalámbrica, tal como una capa 99 de protocolo inalámbrico, para recibir y procesar datos que se ajustan a un protocolo inalámbrico, tal como el protocolo de datos WAP. Por supuesto, también se pueden usar otros protocolos inalámbricos, como Bluetooth.

Como se muestra en la figura 3, RTP es un protocolo para transportar datos en tiempo real a través de Internet. Véase H. Schulzrinne, S. Casner, R. Frederick y V. Jacobson, "RTP: un protocolo de transporte para aplicaciones en tiempo real", solicitud de comentarios (estándar propuesto, RFC 1889, equipo de trabajo de ingeniería de Internet (enero de 1996). RTP es un protocolo "delgado" que brinda soporte para aplicaciones con propiedades en tiempo real, incluida la reconstrucción del tiempo, la detección de pérdidas, la seguridad y la identificación del contenido. Además, RTP proporciona soporte para conferencias en tiempo real para grandes grupos dentro de una intranet, incluida la identificación de origen y soporte para pasarelas, como puentes de audio y video, y traductores de multidifusión a unidifusión. RTP ofrece retroalimentación de calidad de servicio de los receptores al grupo de multidifusión, así como soporte con la sincronización de diferentes flujos de medios.
40

En la figura 3, la pila combinada de los protocolos IP, UDP y RTP 88, 90 y 92 añade 40 bytes a cada paquete para enlaces de baja velocidad y audio altamente comprimido, y 20 bytes para 20 ms de 8 kb/seg. audio. Por lo tanto, la compresión del encabezado es deseable.
45

Como se indicó anteriormente, la pila 80 de protocolos de la figura 3 preferiblemente emplea el protocolo de inicio de sesión (SIP) para establecer intercambios multimedia con una o más partes. En lugar de usar números de teléfono, el SIP usa direcciones en la forma usuario@dominio o usuario@host. Esta dirección, por ejemplo, puede ser idéntica a la dirección de correo electrónico de una persona.
50

El SIP proporciona la funcionalidad PBX o CLASS estándar, como desvío de llamadas, llamada en espera, llamador M, transferencia de llamadas, "llamada en espera", "estacionamiento de llamada" y "captura de llamada". "Llamada en espera" permite que una llamada con extensión u originada por el operador a una estación de voz ocupada de línea única espere automáticamente en la estación llamada hasta que se libere mientras el operador puede gestionar otras llamadas. "Estacionamiento de llamada" permite a un usuario poner una llamada en espera y luego recuperar la llamada desde otra estación dentro del sistema. "Captura de llamada" permite a las estaciones responder llamadas en otros números de extensión dentro de un grupo de captura de llamadas especificado por el usuario. Muchas de estas características en realidad no requieren soporte de señalización en absoluto, pero pueden ser implementadas por el software del sistema de extremo. El SIP está diseñado como una variante de HTTP/1. 1, que permite una fácil reutilización de las características de seguridad y autenticación HTTP, etiquetado de contenido y negociación de pagos.
55
60

El SIP emplea además un gestor de llamadas basado en calendario. El software de procesamiento de llamadas accede al calendario de citas personales del usuario y responde al teléfono en consecuencia. El usuario puede definir categorías de emisores de llamadas y preajustes, basándose en la entrada del calendario, si y dónde se reenvían sus llamadas. La información entregada a la persona que llama si las llamadas no se reenvían puede
65

variar, por ejemplo, de "actualmente no está disponible" a "John Smith está en una reunión hasta las 3 p.m. en la habitación 5621 con Jane Doe", dependiendo de la identidad de la persona que llama. El gestor de llamadas también se puede integrar con el lenguaje de procesamiento de llamadas (CPL), que es un lenguaje de programación basado en estado que permite al usuario construir sistemas de correo de voz o sistemas automáticos de manejo de llamadas en unas pocas líneas de código. El gestor de llamadas también gestiona la traducción entre llamadas RDSI y llamadas de telefonía por Internet.

La figura 4 es un diagrama de bloques de hardware de alto nivel que muestra una realización de un teléfono de red de datos por paquetes o aparato 100 de red. Como será evidente a lo largo de esta divulgación, el dispositivo 100 es un producto de interfaz de coste relativamente bajo para colocar voz y datos en una red de datos por paquetes, tal como LAN de Ethernet, intranets e Internet. Por lo tanto, el dispositivo 100 generalmente se denominará aparato de red para reflejar la amplia aplicabilidad de este dispositivo independiente.

El aparato 100 de red proporciona comunicaciones de audio y video a través de una red de área local (LAN), Internet u otra red Ethernet, y generalmente incluye: un subsistema 110 de controlador de red (por ejemplo, Ethernet); un subsistema 120 de procesamiento de señal digital; un subsistema 130 de conversión de señal; y un subsistema 160 de interfaz de usuario acoplado tanto al subsistema 130 de conversión de señal como al subsistema 120 de procesamiento de señal digital. El teléfono 100 incluye además un suministro de potencia, ROM 142 y RAM 152. El subsistema de interfaz de usuario puede incluir un altavoz 161, un micrófono 162 y otros controles 169 de usuario como se describe a continuación y con referencia a la figura 5. La circuitería 135 de interfaz para la adquisición de datos y funciones de control también se puede acoplar al subsistema 130 de conversión de señal. Alternativamente, tal circuitería de E/S se puede acoplar directamente a DSP 120.

El subsistema 110 de controlador de red está interpuesto entre el DSP 120 y la red de datos externa y, como tal, proporciona y recibe paquetes de datos hacia y desde la red de datos (Ethernet). El subsistema 110 de controlador de Ethernet también instruye al subsistema 120 de procesamiento digital para que acepte los datos recibidos desde o para proporcionar datos a la red Ethernet. Además, el subsistema de controlador de red puede actuar como un controlador de acceso inicial rechazando y descartando paquetes de datos dañados o no deseados recibidos de la red Ethernet.

La figura 5 es un diagrama de bloques que ilustra el presente aparato de red con más detalle. Como se muestra en la figura 5, una realización preferida del subsistema 110 de controlador de red incluye un controlador 112 de Ethernet, un filtro 114 de servicio (transformador 10Base-T) y al menos un enchufe RJ-45 116. Entre otras cosas, el subsistema 110 de controlador de red realiza las siguientes funciones: interconectar el dispositivo de red a la red Ethernet; enviar y recibir paquetes de Ethernet; informar al subsistema DSP 120 que acepte los datos cuando los datos estén disponibles desde Ethernet; recibir los paquetes del subsistema DSP 120 y enviarlos a la Ethernet; y rechazar y descartar paquetes no deseados de Ethernet.

Como se muestra en la figura 5, el controlador 112 de Ethernet puede tomar la forma del controlador de acceso al medio AM79C940 para Ethernet (MACE) disponible desde el microdispositivo avanzado (AMD). El dispositivo de MACE es un periférico basado en el registro esclavo. Todas las transferencias hacia y desde el sistema se realizan usando memoria simple o comandos de lectura y escritura de E/S. Junto con un motor de DMA definido por el usuario, el chip de MACE proporciona una interfaz IEEE 802.3 adaptada a una aplicación específica.

Las FIFO individuales de transmisión y recepción disminuyen la latencia del sistema y admiten las siguientes características: retransmisión automática sin recarga FIFO; acople automático de recepción y transmisión; rechazo automático de paquetes runt; eliminación automática de tramas de colisión; acceso directo de lectura/escritura FIFO para una interfaz simple con controladores de DMA o procesadores de E/S; alineación arbitraria de bytes y compatibilidad con la interfaz de memoria pequeña/grande/mediana; y velocidad de reloj del sistema de 5 MHz-25 MHz.

Con referencia de nuevo a la figura 5, el subsistema 120 de procesamiento de señal digital incluye un procesador 122 de señal digital (DSP) y circuitos lógicos relacionados, que incluyen una memoria 142 de solo lectura (ROM), una memoria 52 de acceso aleatorio (RAM) y un dispositivo lógico programable borrable (EPLD) 124. El subsistema 120 de procesamiento de señal digital proporciona las siguientes funciones: procesamiento de señal digital, tal como compresión de voz; generación de tono de progreso de llamada y generación de señal de llamada; lógica general de "pegamento" para interconectar DSP, memoria y dispositivos de E/S; procesamiento de protocolo de red; control de flujo de llamada e implementación de máquina de estado finito; detección y decodificación de la actividad del teclado; y control de pantalla.

Como se muestra en la figura 5, el DSP 122 usado en el dispositivo de red puede ser cualquier DSP comercialmente disponible adecuado, tal como el TMS320C32 de Texas Instruments. El DSP TMS320C32 tiene las siguientes características: operaciones de multiplexación paralela y unidad lógica aritmética (ALU) en datos enteros o de coma flotante en un solo ciclo; archivo de registro de propósito general; caché del programa; unidades aritméticas de registro auxiliar dedicado (ARAU); memorias internas de doble acceso (512 palabras dobles); dos canales de acceso directo a memoria (DMA); un puerto en serie; dos temporizadores; un puerto de memoria externo; y una estructura

de interrupción múltiple. Por supuesto, también se pueden usar otros productos DSP que tengan características similares o mejoradas.

Además, el DSP TMS320C32 incluye recursos de cuatro interrupciones externas y seis interrupciones internas. La interrupción externa puede ser activada directamente por los pines externos. La interrupción interna puede activarse programando los periféricos individuales, como el puerto en serie, el controlador de DMA y los temporizadores. Además, todas estas fuentes de interrupción se pueden programar como la interrupción del canal de DMA mediante el registro habilitado de CPU/DMA, IE. El DSP TMS320C32 también incluye un cargador de arranque flexible que permite que el programa principal de control del aparato de red se cargue automáticamente desde uno de los tres espacios de memoria externos o el puerto en serie, de acuerdo con lo que determine la actividad de las interrupciones externas de INTO a INT3 cuando el DSP 122 se inicializa, como cuando el dispositivo está encendido.

El DSP 122 generalmente está configurado para incluir las siguientes asignaciones de recursos. Las interrupciones externas incluyen: INTO: Indicación de "arranque del sistema desde 0x1000". Cuando el sistema es encendido e INTO está activo, el DSP arrancará el programa desde el espacio de la memoria externa 0x1000; INT1: señal de interrupción externa de DMA0, usada para recibir paquetes del controlador 112 de red; INT2: señal de interrupción externa de DMA1, usada para enviar paquetes al controlador 112 de red; INT3: estado de paquete AM79C940 e interrupción de mensaje de error. En la figura 6 se muestra un mapa de memoria de DSP de muestra para usar en una realización del presente aparato de red.

Con referencia de nuevo a la figura 5, el presente aparato de red tiene el subsistema 160 de interfaz de usuario que incluye: un codificador 166 de clave, una pantalla 164 de cristal líquido (LCD) y un conjunto manual 163, que incluye un teclado 165, un micrófono 162 y un altavoz 161. Los componentes del subsistema 160 de interfaz de usuario permiten la interacción del usuario con el aparato de red proporcionando las siguientes funciones: interfaz de usuario para entrada (teclado) y salida (LCD); interfaz de voz; salida de alerta de timbre a través del altavoz; y una alternativa de comunicación con auricular o manos libres (micrófono y altavoz). A través de esta interfaz 160, se introducen comandos de usuario y el audio se envía y se recibe en el usuario. La LCD 164 y el teclado 165 pueden cooperar para formar una interfaz gráfica de usuario (GUI) limitada para visualizar y manipular contenido en la pantalla y tomar acciones en respuesta al mismo.

Además, la pantalla LCD puede tener botones adyacentes a la pantalla, como en el lateral y debajo. La función de estos botones puede funcionar como "teclas programables" cuya función depende del estado actual del sistema. Por ejemplo, cuando no se responden llamadas, la pantalla puede mostrar una lista de marcado rápido y la hora del día. Además, después de que las llamadas no se respondieron o se reenviaron al correo de voz, la pantalla muestra una lista de llamadas recibidas. Durante la llamada, se muestran todas las demás llamadas entrantes, lo que permite que el suscriptor cambie de una llamada a otra o conecte la llamada a la llamada existente.

Alternativamente, la interfaz 160 de usuario del presente aparato 100 de red puede configurarse con una pequeña pantalla táctil (no mostrada) para reemplazar o complementar la pantalla LCD y los botones. La pantalla táctil, que muestra gráficamente las funciones y operaciones disponibles y responde al contacto del usuario en la pantalla, proporciona una interfaz de usuario mejorada, como para la entrada de direcciones de red alfanuméricas y otras operaciones de telefonía.

La figura 5 también muestra el sistema 130 de procesamiento de señal, que incluye un codificador y decodificador PCM que realiza conversión de analógico a digital (A/D) y de digital a analógico (D/A), y un amplificador 134 de audio acoplado al teléfono y el altavoz 161 correspondiente y el micrófono 162. También se proporciona un suministro de potencia para proporcionar niveles de tensión positiva y negativa de 5 V a partir de un adaptador de suministro de potencia de CA o CC único ("verruga de pared"). En la realización preferida de la figura 5, los niveles de voltaje negativos son requeridos por la LCD 164 y el códec PCM 132.

La figura 7 es un diagrama de bloques que ilustra una interfaz 700 de memoria adecuada para su uso en el aparato de red de la figura 5. La interfaz 700 de memoria incluye módulos 142 y 152 de memoria externa, que incluyen 128 Kb de memoria de solo lectura (ROM) 142 para almacenamiento de programas y al menos 32 Kbytes de memoria 702, 704, 706 y 708 de acceso aleatorio (RAM) estática de doble palabra (32 bit). Debido a la velocidad relativamente lenta de la ROM 142, es preferible que el aparato de red inicialice el programa principal desde la ROM y almacene este programa en la RAM relativamente rápida para la ejecución del tiempo de ejecución.

La figura 8 es un diagrama de bloques que muestra una interfaz de ejemplo entre el DSP 122 y el controlador 124 de Ethernet de acuerdo con una realización de un aparato de red. Los 32 registros del controlador 124 de Ethernet están mapeados por memoria en el espacio de memoria 0x810000 del DSP 122 como se muestra en la figura 6. Preferiblemente, los primeros dos registros están recibiendo y transmitiendo colas de "primero en entrar, primero en salir" (FIFO). El DSP 122 intercambia los datos con el controlador 124 de Ethernet a través de un bus 802 de datos de 16 bits.

La figura 9 es un diagrama esquemático que ilustra una interfaz entre el DSP 122 y el códec PCM 132 de acuerdo con una realización preferida de la presente invención. Como se muestra en la figura 9, el DSP 122 se conecta al

códec PCM 132 a través de un puerto 902 en serie interno. El puerto en serie en el DSP 122 es un puerto en serie bidireccional independiente.

5 Como se muestra en la figura 5, el DSP 122 también está acoplado operativamente a la LCD 164. La interfaz de control LCD se mapea en las direcciones DSP que se muestran en la figura 10. En una realización, la LCD 164 es una pantalla LCD de 120 x 32 píxeles tal como la LCD MGLS-12032AD, fabricada por Vazitronics. Como la velocidad de acceso de la pantalla LCD es generalmente lenta, los datos que se muestran en la pantalla LCD se pueden mapear en el espacio de memoria STRB0 (1X1000) del DSP 122, que es el mismo espacio de memoria que el espacio de la memoria ROM. Preferiblemente, la lógica de temporización de LCD es la misma que la lógica de temporización para el DSP 122. Sin embargo, cuando la LCD se compone de una mitad izquierda y una derecha, como en la MGLS-12032, es necesario controlar y programar ambas mitades de la LCD cuando se muestra un mensaje de línea completo.

15 La figura 11 es un diagrama de bloques que muestra un ejemplo de la arquitectura de software para el presente aparato de red. Como se muestra en la figura 11, la arquitectura de procesamiento para el presente aparato de red se organiza generalmente en tres niveles: el nivel 1110 de ISR (rutina de servicio de interrupción); el sistema operativo o el nivel 1120 de proceso; y la aplicación o el nivel 1130 de tarea. En la figura 13 se proporciona una lista de ejemplo de funciones y tareas que se pueden realizar en cada uno de los niveles de software.

20 El nivel más bajo, el nivel 1110 de ISR, incluye controladores de interrupción y funciones de interfaz de E/S. El nivel 1110 de ISR sirve como interfaz entre el nivel 1120 de proceso y el hardware del aparato de red mostrado en las figuras 4 y 5.

25 Por encima del nivel 1110 de ISR se encuentra el nivel 1120 de proceso, o sistema operativo, que es preferiblemente un micronúcleo multitarea en tiempo real, como el micronúcleo en tiempo real integrado en CRTX de StarCom. Generalmente, el software 1120 de nivel de proceso (micronúcleo) realiza la gestión de memoria, la gestión de procesos y tareas y las funciones de gestión de disco. En una realización preferida de la presente invención como se muestra en la figura 12, el micronúcleo admite tres mecanismos de planificación: un gestor 1222 de indicador de eventos en tiempo real; un gestor 1224 de tareas retrasadas; y un gestor 1226 de planificación. El micronúcleo tiene tres colas separadas para los tres mecanismos diferentes anteriores, respectivamente.

30 El gestor 1222 de indicador de eventos en tiempo real se usa para activar la ejecución de eventos en tiempo real mediante el establecimiento de indicadores. Si un indicador se establece en una condición "de encendido", la tarea asociada con el indicador se ejecuta inmediatamente. Por ejemplo, una rutina de servicio de interrupción establecería un indicador en particular cuando ocurra un determinado evento. Los eventos de indicador se introducen en una cola de indicador con una dirección de tarea asociada.

35 El gestor 1224 de tareas retrasadas es responsable de los eventos temporizados. Una tarea temporizada, como una tarea a prueba de fallos o "vigilancia", se puede ejecutar después de un cierto tiempo de retraso. Si un determinado evento no ocurre dentro de un cierto período de tiempo, el temporizador activa la tarea que hace que se ejecute. Otro ejemplo es la ejecución repetida de una tarea controlada por un temporizador periódico. En una realización de ejemplo, hay 10 entradas de temporizador. Cada temporizador se carga con un conteo de tics y luego se disminuye en cada tic del temporizador de intervalos del hardware. Cuando el conteo llega a cero, la tarea asociada con el temporizador es planificada en la cola de tareas. El gestor 1226 de planificación escanea la cola de planificación de tareas buscando tareas planificadas. Al descubrir una entrada en la cola, el control pasa a una tarea planificada.

40 Las figuras 13a-f son tablas cuyas tareas y funciones de software de ejemplo de lista pueden ser parte del software de nivel de tareas (figuras 13a-c), software de nivel de proceso (figura 13d) y software de nivel de ISR (figuras 13e-f). Para los fines de la presente invención, los términos "tarea" y "función" como se refieren con respecto a la arquitectura de software se consideran sinónimos. Sin embargo, las "tareas" son generalmente ejecutadas por el gestor 1226 de planificación, mientras que las "funciones" generalmente son llamadas por tareas u otras funciones. Las tareas de la aplicación, como las tareas de procesamiento de llamadas (Call_task) y de procesamiento de IP (IP_Send_task y Ercv_task, etc.), están planificadas por el software 1120 de nivel de proceso. La ejecución de tales tareas es el resultado de una planificación previa por parte de una ISR, otra tarea o la propia tarea actual.

45 Las figuras 13 A-F ilustran definiciones de funciones y procedimientos de ejemplo llamadas en una operación accionada de eventos realizada por el presente software telefónico de red de datos por paquetes de la figura 11. Las funciones, que son llamadas en la aparición de varios eventos, permiten el funcionamiento del teléfono/sistema de la red de datos por paquetes e incluyen operaciones brutas tales como: inicialización del teléfono/sistema de la red de datos por paquetes; procesamiento de datos ARP; codificación de datos de voz; procesamiento de datos de mensajes; procesamiento de datos de IP; descifrado de datos de voz; transferencia de datos analógicos y digitales y formar búferes correspondientes; y realización de funciones de "vigilancia".

50 La inicialización del aparato telefónico de red de datos por paquetes incluye los pasos de inicialización de hardware y planificación de tareas. Después del encendido, el DSP 122 transferirá automáticamente el programa principal desde la ROM 142 a la RAM 152 (operación de arranque). La inicialización de hardware ocurre de manera

tradicional, incluyendo los pasos de: inicialización del puntero de pila, registro de control de interfaz de bus externo, registro de control global del DSP, vector de interrupción para la ISR, y similares.

5 Una vez completada la inicialización del hardware y la planificación de tareas preliminares, el control de procesamiento se devuelve al nivel 1120 de proceso (micronúcleo). El micronúcleo CRTX 1120 y las tareas planificadas luego controlan el procesamiento posterior.

10 Con referencia de nuevo a la figura 13A, el software de nivel de tareas del presente aparato de red incluye el procesamiento de protocolo de resolución de direcciones (ARP). ARP es un protocolo TCP/IP conocido que se usa para convertir una dirección IP en una dirección física (llamada una dirección de control de enlace de datos (DLC)), como una dirección Ethernet. Un ordenador host que desea obtener una dirección física transmite una solicitud ARP a la red TCP/IP. El ordenador host en la red que tiene la dirección IP en la solicitud responde después con su dirección física de hardware.

15 La figura 14 es un diagrama de flujo que ilustra un procedimiento 1400 de salida de solicitud ARP, ARP_Out(). Como se ilustra en la figura 13B, ARP_Out() es un componente del software de nivel de tareas que recibe una dirección IP a resolver, y emite una dirección de MAC correspondiente. Cuando comienza una solicitud ARP (paso 1402), la función ARP_Out() comprueba primero la dirección IP solicitada desde una tabla de memoria caché ARP local, arptable (paso 1404). Si la entrada correspondiente es RESUELTA en el paso 1406, entonces ARP-Out() copia la dirección de MAC desde arptable al parámetro solicitado y devuelve un indicador de estado ARPOK (paso 20 1408). De lo contrario, el procedimiento asignará una entrada en el arptable y planificará una solicitud ARP (paso 1410). Como se muestra adicionalmente en el paso 1410, una dirección de MAC, es decir, "manejo", del arptable se devuelve al programa principal (c_int00()). De acuerdo con el manejo, el software luego verifica el ae_state de la entrada correspondiente.

25 La figura 15 es un diagrama de flujo de un procedimiento de entrada de solicitud ARP de ejemplo 1500, ARP_In_task(), que es un componente del software de nivel de tareas enumerado en la figura 13 A. La ARP_In_task recibe un paquete ARP y modifica el arptable o pone en cola una respuesta ARP si el paquete entrante es una solicitud ARP. Cuando se recibe un paquete ARP (paso 1502), el software verificará si el hardware ARP del paquete o los tipos de protocolo coinciden (paso 1504). Si los tipos no coinciden, el control se devuelve al programa principal (paso 1506). Si uno o ambos tipos coinciden, el software verifica si el host de destino es el host actual (paso 1510). Si el host de destino no es el host actual, entonces el control se devuelve al programa principal (paso 1508).

35 Como se muestra adicionalmente en la figura 15, si el host de destino es el host actual, entonces la ARP_In_task verifica a continuación la tabla ARP para determinar si hay una entrada ARP correspondiente para el paquete entrante (paso 1512). Si se encuentra una entrada (paso 1514), entonces la nueva dirección de MAC se copia en la entrada existente y modifica el "tiempo de vida" (TTL) de la entrada a un nuevo valor (paso 1516). Los expertos en la técnica entienden que un TTL es un campo en el protocolo de Internet (IP) que especifica cuántos saltos más puede 40 recorrer un paquete antes de descartarse o devolverse al remitente. Sin embargo, si no se encuentra dicha entrada de MAC de acuerdo con el paso 1513, entonces la ARP_In_task añade una nueva entrada de MAC en la tabla ARP (paso 1518). Si la entrada de MAC está en estado PENDIENTE (paso 1520), se cambia a un estado RESUELTO y la dirección de MAC se copia en la entrada de destino (paso 1522). Si el paquete ARP entrante es una solicitud ARP de otro host, se envía un paquete ARP de respuesta poniendo en cola la IP_Send_task, pasos 1524 y 1526. El control se devuelve al programa principal (paso 1528).

Además de los procesos de entrada y salida ARP, el procesamiento ARP en el nivel de tareas incluye una tarea ARPTimer_task(), que es una tarea de bucle retrasado usada para mantener la tabla de entrada ARP arpentry. 50 Nominalmente, la ARPTimer_task() se genera una vez por segundo. El objetivo principal de la tarea ARPTimer_task() es disminuir el "tiempo de vida" (TTL) de la entrada ARP y reenviar la solicitud ARP durante el estado pendiente en caso de que se pierda la solicitud ARP anterior.

El procesamiento de nivel de tareas también puede incluir operaciones de procesamiento asociadas con la codificación y decodificación de paquetes de audio. La Codec_task generalmente incluye una función 55 SpeechEncode(), que codifica los datos de voz del búfer de ADBuf al EncodeBuf de acuerdo con el algoritmo indicado por el parámetro "tipo". Los datos codificados se envían a través de la cola IP_Send_task, con el conjunto de parámetros "RTP".

60 Las operaciones de nivel de tareas también pueden incluir el procesamiento del protocolo de Internet (IP). Las operaciones generales de procesamiento de IP se ilustran en el diagrama de bloques de la figura 16. Como se muestra en la figura 16, el procesamiento de IP incluye los pasos de: transmitir y recibir paquetes de Ethernet, paso 1602; paquetes IP de multiplexación y demultiplexación, paso 1604; y empaquetar y desempaquetar Ethernet, protocolo de internet (IP), protocolo de datagrama de usuario (UDP), protocolo de transporte en tiempo real (RTP) y paquetes de protocolo de resolución de direcciones (ARP), paso 1606.

65 De acuerdo con el paso 1602 de la figura 16, la transmisión de paquetes puede realizarse usando canales de

acceso directo a memoria (DMA) del controlador 112 de Ethernet. El DMA es una técnica para transferir datos desde la memoria principal a un dispositivo sin pasarlo a través de la CPU. Dado que los canales de DMA pueden transferir datos hacia y desde dispositivos mucho más rápidamente que con medios convencionales, el uso de canales de DMA es especialmente útil en aplicaciones en tiempo real, como el actual sistema de telefonía de red.

5 El controlador 110 de red soporta preferiblemente una pluralidad de canales de DMA, tales como el canal de DMA1 del controlador 112 de Ethernet que se puede usar para la transmisión de paquetes. Cuando un paquete de Ethernet está listo para la transmisión, se llama a la función DMA1(), una función de nivel de ISR, configurando la dirección de origen (búfer de paquetes Ethernet, ESend), la dirección de destino (FIFO de transmisión del controlador de Ethernet) y un contador (la longitud del paquete) . Ejemplos de estructuras de datos de transmisión Ethernet se proporcionan en la figura 17. La función DMA1() luego inicia el canal DMA1. Cuando el contador llega a cero, el DMA1 se detiene y espera la siguiente llamada.

15 La figura 18 es un diagrama de bloques que muestra el flujo de datos entre un búfer 1802 de entrada de audio, un búfer UDP 1804 y una tabla ARP 1806 a la interfaz Ethernet (FIFO de transmisión de Ethernet) del controlador 112 de red Ethernet. Como se muestra adicionalmente en la figura 18, los datos del búfer 1802 de entrada de audio, el búfer UDP 1804 y la tabla ARP 1806 se envían a una cola 1810 de salida IP, y están dispuestos para indicar el tipo de protocolo, el puntero fuente y la longitud de datos. En lugar de poner en cola los datos de envío, la IP_Send_task se pone en cola por el software 1120 de nivel de proceso (micronúcleo). Los tipos de protocolo admitidos por la IP_Send_task generalmente incluyen UDP, RTP, ARP_REQUEST, ARP_REPLY. La IP_Send_task se usa para la transmisión de paquetes y el empaquetado de Ethernet. Preferiblemente, la IP_Send_task está planificada por otras tareas o funciones tales como SIP_task, ARP_Out (), SpeechEncode(), etc. Una vez que se ejecuta la IP_Send_task, verifica el tipo de protocolo de los datos. Esta tarea encapsula los datos de salida en el paquete de Ethernet correspondiente en el búfer ESend. Finalmente, el paquete se envía a través del canal de DMA asignado (DMA1).

30 La figura 19 es un diagrama de flujo de datos que ilustra adicionalmente operaciones de recepción y demultiplexación de paquetes. La demultiplexación se realiza planificando diferentes tareas para diferentes protocolos en la tarea Ercv_task. De acuerdo con el paso 1602 de la figura 16, los paquetes se reciben en la memoria FIFO de datos de recepción (paso 1902) y se procesan adicionalmente mediante un controlador de canal DMA0 (paso 1904). Como el DSP 122 no sabe cuándo llegarán los paquetes, el canal DMA0 está activo todo el tiempo (es decir, no se detiene incluso cuando el contador llega a cero). Cuando llega un paquete, el canal DMA0 lo copiará automáticamente desde la FIFO receptora del controlador de Ethernet al búfer receptor de Ethernet, Ercv (paso 1906). El canal DMA0 se detiene cuando no hay datos disponibles en la FIFO.

35 ERcv_task es una tarea de activador de indicador para el desempaqueado de paquetes de Ethernet y la demultiplexación de paquetes IP (paso 1908). La Ercv_task funciona de la siguiente manera: primero, una función PacketCheck() es llamada para verificar el paquete entrante. PacketCheck() devolverá el tipo de protocolo del paquete, o NULO si el paquete no es válido. En segundo lugar, dependiendo del tipo de protocolo devuelto, la ERcv_task activará las diferentes tareas para procesar el paquete recibido, RTP_In_task para paquete "RTP" (paso 1910), ARP_In_task para paquete "ARP" (paso 1912) o tareas de procesamiento UDP (paso 1912) para paquetes UDP, por ejemplo.

45 Con referencia a la figura 13C, SpeechDecode() es una función de decodificación de voz asociada con el procesamiento RTP del paso 1910. En primer lugar, una tarea de SpeechDecode() comprueba si hay datos disponibles en el búfer de decodificación, DecodeBuf. Si los datos están disponibles, por ejemplo, RcvFlag es ESTABLECIDO, luego SpeechDecode() lo decodifica de acuerdo con el tipo de datos de datos de recepción, PCM (G. 711), G. 723, G. 729, por ejemplo. Los datos decodificados se envían al búfer D/A, DABuf.

50 La rutina de interrupción A/D y D/A puede ser activada por una fuente de interrupción interna, e. g., Rint0(). Preferiblemente, la rutina de interrupción A/D y D/A es activada por una frecuencia de muestreo de 8 kHz proporcionada por el DSP. Como esta rutina se llama con frecuencia, Rint0() se escribe preferiblemente en lenguaje ensamblador. Los pasos realizados por Rint0() incluyen los pasos de: leer una muestra D/A desde el búfer D/A, DABuf; enviar la muestra al puerto en serie D/A; obtener una muestra del puerto en serie A/D; guardar la muestra A/D en un búfer A/D, ADBuf; e incrementar los punteros de búfer A/D y D/A, ADPnt y DAPnt, en uno.

60 Las figuras 20A y 20B son diagramas de bloques que muestran un esquema de búfer de "ping-pong" A/D y D/A usado por el software de la presente invención. Además, si el valor del puntero A/D actual (ADPnt) excede un umbral de búfer predeterminado (ADTh), entonces se establece un indicador en la cola de tareas de indicador que indica que se requiere el servicio.

65 Los búferes A/D y D/A se pueden dividir en dos partes, el búfer superior 2002a y el búfer inferior 2002b, respectivamente. Ambos búferes se pueden diseñar como búferes circulares. De esta forma, cuando el puntero actual alcanza el fondo del búfer, se ajusta a su comienzo. Sin embargo, desde el punto de vista del codificador y del decodificador, se usa como un esquema de búfer de ping-pong de dos tramas (definido como la trama superior y la trama inferior). El funcionamiento de este proceso se muestra en las figuras 20A y 20B. Para la conversión A/D,

cuando la parte superior (o inferior) está llena, los datos en el búfer superior (o inferior) pasarán a través del conmutador 2004 de ping pong y se copiarán al búfer 2006 de codificación de voz, EncodeBuf. Para la conversión D/A, si se completa el búfer superior (o inferior), se copiará una nueva trama de datos del búfer 2010 de decodificación de voz, DecodeBuf, al búfer superior 2008a (o inferior 2008b). Este mecanismo asegura que mientras el algoritmo de codificación (o decodificación) lee (escribe) desde una parte del búfer, la ISR de muestreo A/D (o D/A) puede escribir (leer) la otra parte del búfer sin conflicto.

La figura 21 es un diagrama de transición de estado de una subrutina de Call_task usada en una realización de ejemplo del presente dispositivo de red. La Call_task es una tarea en bucle que gestiona el procedimiento de llamada. Como se muestra en la figura 21, el estado "inactivo" 2102 se produce cuando no se realiza ninguna llamada y no hay llamadas entrantes. Cuando existe esta condición, la Call_task se enlaza en el estado "inactivo" 2102. El estado "Tono de marcado" 2104 existe cuando el estado del receptor es DESCOLGADO, o el estado del auricular indica MANOS LIBRES, y así el estado de la Call_task cambiará del estado "inactivo" 2102 al estado "Tono de marcado" 2104 cuando exista una condición DESCOLGADO o MANOS LIBRES. Estos estados generalmente son introducidos por una entrada de un usuario a través de los controles 160 de usuario que indican que se debe iniciar una llamada. Cuando el estado de Call_task está en el estado "Tono de marcado" 2104, la Codec_task se configurará como "Modo de tono, Tono de marcado" y se enviará un tono de marcado a los componentes del auricular de la interfaz 160 de usuario.

Con referencia de nuevo a la figura 21, mientras se encuentra en el estado "Tono de marcado" 2104, si se presiona cualquier tecla de dígito ('0'...'9', '*' y '#') o el botón de rellamada, el estado de la llamada cambia del estado "Tono de marcado" 2104 al estado "Consigue número" 2106. En el estado "Consigue número" 2106, el tono de marcado se detiene en el auricular.

Una vez que el usuario ha introducido el número de llamada y el usuario ha presionado un botón INTRODUCCIÓN para indicar que se ha completado la marcación, la Call_task verificará si la entrada es válida. Si el número es válido, se crea una entrada de llamada mediante una función CreateSipCall() y la Call_task entrará en un estado "SIP" 2108. De lo contrario, si el número de entrada no es válido, el número se solicita nuevamente y el estado permanece en el estado "Consigue número" 2106.

Mientras se espera el procesamiento de SIP_task, se pueden tomar varias decisiones dependiendo del estado "SIP" 2108. El estado "SIP" 2108 es una variable global, SIP_status, que se modifica mediante la SIP_task de acuerdo con su transición de estado. Si el estado "SIP" 2108 cambia a SIP_Ring, la Call_task cambiará al estado "Llamar de nuevo" 2114 y la Codec_task se configurará como modo "Modo de tono, RingBack". Cuando la Codec_task está en el modo "Modo de tono, Llamar de nuevo", se envía un tono de devolución de llamada al auricular.

Desde el estado "SIP" 2108, si el estado "SIP" 2108 cambia a SIP_busy, la Call_task y por lo tanto la llamada cambiará al estado "Tono ocupado" 2120 y el tono de ocupado se reproducirá en el teléfono. Si el estado "SIP" 2108 cambia a SIP_Refused, se mostrarán mensajes apropiados en la pantalla LCD relacionados con el estado SIP_Refused.

Desde el estado "Llamar de nuevo" 2118, si el estado "SIP" se convierte en SIP_Connected, el estado de Call_task cambia al estado "Hablar" 2116. Cuando el estado de Call_task está en el estado "Hablar" 2116, la Codec_task se configurará como modo SpeechEncode y SpeechDecode.

Para las llamadas entrantes, mientras se encuentra en el estado "Inactivo" 2102, si el estado "SIP" 2108 es SIP_Invite, el estado Call_task cambia al estado "Ring" 2114 y la Codec_task se configurará como "Modo de tono, Tono de llamada". Cuando la Codec_task se configura como "Modo de tono, RingTone" se reproducirá un tono de llamada en el altavoz. Después de que el estado SIP pase a SIP_Connected, el estado Call_task cambiará al estado "Hablar" 2116. De lo contrario, si el estado SIP pasa a SIP_Cancel, lo que sucede si la persona que llama deja la llamada, el estado Call_task vuelve al estado "inactivo" 2108.

Mientras está en el estado "Inactivo" 2102, si se presiona el botón INTRODUCCIÓN, la Call_task llama a la Setting_task. Cuando el programa de configuración de parámetros finalice, volverá a Call_task.

Durante la ejecución de Call_task, si el estado de enlace indica que el receptor está en COLGADO o se encuentra un error del sistema, la Call_task cambia al estado "inactivo" 2102, independientemente de cuál sea el estado anterior (excepto el estado "Llamada" 2114).

En la realización preferida del aparato de red como se muestra en la figura 5, el teclado del teléfono tiene 17 teclas para proporcionar entradas y comandos al usuario. El teclado del teléfono incluye teclas de 10 dígitos, dos teclas especiales y cinco teclas de función se definen como se muestra en la figura 22.

La Key_task es una tarea retrasada de bucle que se ejecuta periódicamente, como cada 0,1 segundos. Cuando se inicia, Key_task primero llama a la función key(). Si el valor de retorno no es "-1", significa que se ha presionado una tecla. Luego, la función KeyMap() mapea la palabra clave binaria de entrada a la palabra clave ASCII. La Key_task

luego establece el miembro correspondiente de la estructura FuncKey. Si el sistema está listo para aceptar la entrada de clave (se indica KeyRegEnable), la palabra clave de entrada se almacena en KeyBuf.

Además, la Key_task preferiblemente admite cuatro modos de entrada diferentes: modo de entrada de dígito, modo de entrada de dirección IP, modo de entrada alfabética y modo de entrada de dirección de lista. El cambio entre los cuatro modos se puede hacer presionando el botón ENTER antes de marcar cualquier número o alfabeto cuando se coge el auricular y se escucha un tono de marcado. Una vez que se completa la entrada y se presiona el botón ENTER, los números de entrada se transferirán a la tarea actual (Call_task o Setting_task) mediante un canal de mensajes. Si se presiona la tecla Volver a marcar, la tarea copiará la entrada anterior desde el búfer de respaldo, KeyBackup, al KeyBuf. Luego, los datos se transferirán a la Call_task.

El sistema operativo del presente aparato de red preferiblemente soporta un esquema de planificación de tareas retrasadas. La tarea retrasada es similar a la función sleep() en UNIX. Sin embargo, una tarea retrasada también puede ser una ejecución de tarea persistente desde un temporizador periódico cuando se establece el indicador de repetición de la tarea. Para las tareas retrasadas, el software 1120 de nivel de proceso requiere un temporizador de intervalo para proporcionar un tic del sistema. El sistema de la figura 5 usa el temporizador1 TMS320C32, TCLK1, como la base del temporizador del sistema.

La Clock_task es una tarea retrasada de bucle que realiza funciones de reloj y calendario en tiempo real. Sirve como el reloj general para calcular y mostrar la hora actual, incluida la hora, el minuto y el segundo. Cuando se conecta una llamada, puede mostrar la duración de la llamada. Cuando el teléfono está colgado, el año, mes y fecha actuales también se pueden mostrar en la pantalla LCD.

Con referencia de nuevo a la figura 11, el software de teléfono de red de la presente invención incluye varias funciones de bajo nivel que se incluyen como parte del nivel de ISR de software. Algunas de estas funciones de bajo nivel son funciones relacionadas con E/S, que se usan con el puerto paralelo de E/S de 8 bits del teléfono definido en la figura 24. Las funciones relacionadas de E/S de bajo nivel incluyen: el monitor de estado "colgado", Hooks(); la verificación de disponibilidad de entrada clave y lectura, Key(); control de auricular y manos libres, HandSet(); restablecimiento del controlador de Ethernet, ENET_reset(); control de volumen, AmpControl(); y restablecimiento del software del sistema.

El chip 136 de interfaz de audio, que preferiblemente toma la forma de un LM4830, se puede usar para controlar el cambio entre el modo auricular y manos libres. Por ejemplo, la función HandSet() puede escribir un '0' en el puerto de E/S cuando se requiere el modo "manos libres" o escribir un '1' en el puerto apropiado cuando se requiere el modo "auricular".

Las funciones de bajo nivel de la presente invención también incluyen la ISR de interrupción del controlador de Ethernet, c_int03(). La estructura de mensaje global para su uso con c_int03() se define para el estado del controlador de Ethernet como se muestra en la figura 25. Cada vez que se envía un paquete, o se completa un paquete recibido, el controlador de Ethernet interrumpirá el DSP 122 para indicar la interrupción. El DSP 122 leerá los estados de transmisión y recepción del registro del controlador de Ethernet y luego almacenará el estado en la estructura de estado anterior. Esta información puede ser verificada por otras tareas. Además, estos mensajes se leen después de cada transmisión de paquetes. De lo contrario, el controlador de Ethernet estará bloqueado.

Como se indicó anteriormente, se prefiere que el presente aparato de red de la presente invención use el protocolo RTP para transmitir y recibir paquetes de voz en tiempo real. El paquete RTP está encapsulado en un paquete UDP. Los módulos de IP_Send_task y RTP_In_task funcionan para crear y analizar paquetes RTP. La figura 26 muestra una estructura de encabezado RTP para el procesamiento de paquetes RTP. Cuando la IP_Send_task obtiene una solicitud para enviar un paquete RTP, primero genera un encabezado Ethernet y UDP. A continuación, añade el encabezado RTP en el búfer de transmisión de paquetes de Ethernet. Finalmente, los datos RTP se copian en el área de datos RTP y se envían a través de la red de datos.

La figura 27 muestra una estructura de datos para su uso con una función de generación de tonos, Tone_task(). Los parámetros descritos en la figura 27 se ilustran en el diagrama de temporización de generación de tonos de la figura 28. Tone_task es una tarea retrasada que se puede ejecutar aproximadamente cada 0,1 segundos. Se usa para contar el tono activo y detener la duración definida en la estructura ToneType. Tone_task establece ToneState en ACTIVO durante la ráfaga y STOP durante el silencio. La duración activa y de parada diferente genera diferentes tonos. Estos son: tono de marcado, tono continuo (sin parada); tono de ocupado, ráfaga 0,5 s y silencio 0,5 seg; Suena el tono de fondo, ráfaga 2 segundos y silencio 4 segundos; señal de timbre, ráfaga 0,8 seg dos veces en dos segundos, luego silencio 4 seg. El módulo ToneGenerate() genera un tono de 400 Hz de una trama o una señal de timbre de 2400 Hz definida por el parámetro "modo" cuando ToneState está ACTIVO. De lo contrario, se proporciona una señal de silencio de una trama.

El aparato de red de la presente invención generalmente usa UDP como su protocolo de transporte para SIP. La SIP_task es una tarea en bucle que gestiona la señalización SIP. Dado que el aparato de red actual se puede usar como persona que llama o llamada, la SIP_task funciona como un UAC (cliente de agente de usuario) y un UAS

(servidor de agente de usuario).

La figura 29 es un ejemplo de código fuente que muestra las estructuras de datos usadas para procesar las solicitudes SIP o respuestas de acuerdo con el protocolo SIP. Tstate es la estructura de transición de estado usada en la SIP_In_task y SIP_task para la transición de estado SIP. Los mensajes SIP analizados están en la estructura de datos message_t. La llamada de estructura se define para cada llamada y las entradas totales de llamada se definen mediante msg [MaxSipEntry].

La figura 30 es un diagrama de transición de estado de la SIP_task que funciona como un cliente (por ejemplo, un parte que llama). Cuando el teléfono SIP inicia una llamada, funciona como un cliente. Se creará una llamada a través de los siguientes pasos: se asigna un mensaje de entrada de llamada msg[CurrentIndex] cuando se coge el teléfono y el indicador de la llamada es ESTABLECIDO; CreateSipCall() crea un paquete SIP de acuerdo con la configuración actual y las entradas de marcación, en donde el paquete SIP se usa como referencia de la llamada y el us_state se establece en UAC; SIPParse() genera la estructura de mensaje (msg [CurrentIndex].m) para la llamada desde el paquete anterior; la SIP_task verificará si hay llamadas activas; si hay una llamada (msg[i].flag es ESTABLECIDO), la SIP_task creará la solicitud correspondiente de acuerdo con la especificación SIP y los estados SIP se actualizarán en la SIP_task como se muestra en la figura 30.

La figura 30 muestra un diagrama de estado de ejemplo para operaciones de cliente (llamador), denominado diagrama de transición de estado UAC de SIP_task. Desde un estado inicial (paso 3002) se introduce un estado de llamada y una SIP_task retransmite una solicitud SIP INVITAR periódicamente (T1) hasta que se recibe una respuesta (paso 3004). Nominalmente, T1 es 500 ms inicialmente y se duplica después de cada transmisión de paquetes. (Paso 3006) T2 es nominalmente 32 segundos. Si el cliente no recibe respuesta, la SIP_task finaliza la retransmisión cuando expira el temporizador T2 y el estado SIP cambiará a Cancelar (paso 3008). Si la respuesta es provisional, el cliente continúa retransmitiendo la solicitud hasta siete veces. Cuando se recibe una respuesta final, el estado cambiará a Completado y se generará un ACK (paso 3010). Cuando la persona que llama se rinde, el estado cambiará al estado Adiós (paso 3012). Las solicitudes ADIÓS también se retransmiten durante el intervalo de T1 hasta que T2 expira con el propósito de una transmisión confiable. La variable, SIP_Status, se cambiará de acuerdo con la respuesta recibida como se muestra en la figura 31. Por ejemplo, si se recibe una respuesta 3xx, la SIP_task iniciará otra llamada a la dirección redirigida. Se pueden mostrar otras respuestas finales en la pantalla LCD.

Cuando el aparato de red recibe una llamada, la SIP_task funciona como UAS de SIP (servidor). Los paquetes entrantes se procesan de la siguiente manera: UDP_In_task acepta el paquete UDP entrante y envía los paquetes a SIP-In-task junto con su dirección IP de origen y número de puerto. SIP_In_task procesa el paquete de acuerdo con la especificación SIP y actualiza los estados en consecuencia. La SIP_task supervisará el estado del receptor, establecerá y disminuirá el temporizador T1 y T2 de cada llamada y actualizará los estados SIP si es necesario.

La figura 32 ilustra un diagrama de transición de estado de ejemplo de un UAS de SIP. Mientras que la SIP_task permanece en un estado inicial (paso 3205), escucha los paquetes SIP entrantes. Si se recibe una solicitud INVITAR, genera una respuesta de llamada (180) y su estado cambia a Invitar y el módulo de SIP_task avanzaría a un paso de Procedimiento (paso 3210). Si un abonado llamado coge el teléfono, el estado cambia a Descolgar y el proceso avanza hacia el Éxito (paso 3215), lo que indica que se ha establecido una sesión de llamada exitosa. Si la parte llamada no contesta, el estado cambia a Fallo y el proceso avanza al estado Fallo (paso 3220). Después del éxito o fracaso, el cliente confirmará el estado actual y avanzará el proceso al estado Confirmado (paso 3225). Cuando la parte que llama finaliza la sesión, el estado cambia a Colgar y el proceso avanza a Adiós (paso 3230) indicando que la sesión actual se ha completado.

Como se establece en este documento, el aparato de red es un dispositivo independiente capaz de iniciar y recibir llamadas telefónicas en una red de datos por paquetes. Aunque la arquitectura independiente descrita en este documento ofrece muchas ventajas concomitantes, tal como su coste de implementación relativamente bajo, la arquitectura de software similar y las definiciones funcionales descritas en relación con el aparato independiente 100 también pueden proporcionarse en un dispositivo telefónico basado en PC. En tal caso, un ordenador personal convencional que tiene un micrófono, altavoces y una tarjeta de interfaz de red adecuada, está provista de un software para operar de manera consistente con la manera descrita anteriormente. Por supuesto, los cambios se efectúan en esta realización, tal como los componentes y funciones de la interfaz de usuario que se realizan por elementos convencionales del PC por ejemplo, el teclado, monitor, ratón y similares. El software proporciona una interfaz GUI para la funcionalidad del teléfono para habilitar las funciones de telefonía deseadas.

El aparato de red de la presente invención, además de realizar funciones de telefonía tradicionales, también puede proporcionar una interfaz rentable entre la red y el entorno. Aunque equipar los sensores con interfaces Ethernet no es factible, debido a la gran cantidad de puertos requeridos y el coste del hardware mínimo requerido, el aparato de red de la presente invención puede convertirse en el punto de reunión para una serie de sensores digitales y analógicos. Esto se lleva a cabo generalmente acoplando el sensor externo al aparato de red a través de la circuitería 135 de E/S convencional que está acoplada al DSP 122. La circuitería de E/S puede tomar la forma de búferes simples, convertidores A/D, registros y similares. Esta característica es particularmente útil en entornos que tienen teléfonos por razones de seguridad, por ejemplo, ascensores, vestíbulos, pisos de tiendas, garajes, etc. Los

ejemplos incluyen: sensor digital infrarrojo pasivo (PIR) para detectar la presencia de personas; esto se puede usar para reenviar llamadas automáticamente si nadie está en la oficina o como parte de un sistema de seguridad o administración de energía; sensor de luz analógico o digital para detectar si la oficina está ocupada; sensor de temperatura analógico; detectores de humo, monóxido de carbono y radiación; y cierres de contacto para sistemas de seguridad. Por lo tanto, el presente aparato de red proporciona un punto de integración del sistema.

Para proporcionar una mayor capacidad de E/S mejorada, la circuitería de E/S puede ser compatible con los protocolos de control local, como los protocolos X10 y CEBus, que son estándares reconocidos para controlar dispositivos alimentados por línea, como iluminación o electrodomésticos. Añadir tal interfaz al teléfono proporciona un control basado en la red de dichos dispositivos.

La figura 33 ilustra un sistema que emplea el presente aparato de red para establecer llamadas entre dos o más partes en la red. El sistema generalmente incluye uno o más aparatos 100 de red independientes, tal como se describió anteriormente. Además, el sistema también puede incluir dispositivos 3320 de telefonía basados en PC, tales como un PC habilitada para red que acciona un software de telefonía de red adecuado que cumple con el protocolo con el aparato 100 de red. Cada punto final de telefonía se puede referir como un nodo y tiene una dirección SIP específica. Al emplear esta dirección específica, cualquier nodo que actúe como una parte que llama (cliente) puede iniciar directamente una sesión de llamada con cualquier otro nodo en la red (servidor).

El sistema preferiblemente también incluye un servidor 3325 de redirección al que pueden acceder los diversos nodos en la red para proporcionar servicios mejorados, tales como un servicio de directorio, reenvío de llamadas, ramificación de llamadas, mensajes de llamadas y similares. Por ejemplo, una parte que llama que desee iniciar una llamada a JOHN SMITH puede introducir la dirección SIP para esa persona si se la conoce, como sip:john.smith@work.com. Si, por otro lado, la parte que llama no conoce la dirección SIP de la parte, la parte que llama puede contactar con el servidor 3325 de redirección con una solicitud para comenzar una sesión con JOHN SMITH. El servidor de redirección incluye bases de datos con información de registro para varias partes y puede devolver la dirección SIP a la parte que llama o reenviar la solicitud de llamada a la dirección SIP adecuada. Además, la parte llamada puede tener múltiples direcciones sip como john.smith@home, john.smith@office, john.smith@lab y similares. El servidor de redirección puede proporcionar una señal de inicio de sesión a cada una de estas direcciones y establecer una conexión entre la parte que llama y el primer nodo contactado que responde a la solicitud de inicio. Del mismo modo, las partes pueden registrarse periódicamente con el servidor de redirección para indicar la dirección SIP actual donde se pueden contactar (función de reenvío de llamadas).

El aparato 3305 de red se puede configurar para interconectarse a uno o más sensores 3310. Las señales de los sensores son recibidas por el aparato de red 3305 y pueden enviarse a lo largo de la red a un nodo de red deseado. Las señales de los sensores pueden detectarse periódicamente mediante un temporizador en el aparato de red y enviarse a una dirección SIP almacenada en la memoria. Alternativamente, las señales del sensor pueden medirse mediante el aparato 100 de red basándose en un comando recibido de otro nodo (sondeado por un nodo de red remoto) o puede medirse basándose en una señal de interrupción recibida que indica un cambio de estado del sensor (interrupción controlada) Por ejemplo, el aparato 100 de red se puede usar como un dispositivo de comunicación de sistema de seguridad que informa del estado de varios puntos del sensor de seguridad a una estación de monitorización central. En tal caso, el aparato puede verificar periódicamente el estado de los sensores conectados, tales como sensores de puerta, sensores de incendio, detectores infrarrojos pasivos y similares, e informar al estado de la estación central el estado actual. En el caso de un cambio de estado que indicaría una condición de alarma, el aparato 100 podría generar una sesión de llamada con la estación central e informar de esta condición también. Por supuesto, el mismo aparato que actúa como un comunicador de alarma también puede proporcionar funciones de telefonía completas. Además, aunque se describió una aplicación de seguridad simple, también se apreciará que varias aplicaciones de recopilación y control de datos generalmente conocidas como SCADA (control de sitio y adquisición de datos) pueden implementarse usando el presente aparato 100 de red.

Para mantener el servicio de línea de vida durante cortes de energía, el aparato de red de la presente invención puede estar equipado con una batería recargable, posiblemente integrada en un transformador de pared.

Como muchas ubicaciones están actualmente equipadas con una sola interfaz Ethernet, el aparato de red de la presente invención debería proporcionar un conector de Ethernet de dos puertos, con una interfaz RJ-45 externa. Esto permite el funcionamiento simultáneo tanto del dispositivo de telefonía como del ordenador habilitado de red.

Además de los datos de audio, el aparato de red actual también puede recibir y transportar datos de video. Por ejemplo, una interfaz de entrada de video, ya sea analógica o a través de un USB (bus universal en serie) puede ser operativamente acoplada al DSP 122 para implementar esta característica.

El presente aparato 100 de red también se puede acoplar a una interfaz Ethernet inalámbrica adecuada para permitir el equivalente de un teléfono inalámbrico. Por ejemplo, se pueden incorporar módulos inalámbricos compatibles Bluetooth o IEEE 802.11 en el aparato para proporcionar intercambio de datos inalámbrico con otros dispositivos. Tal módulo puede incorporarse, por ejemplo, como parte del subsistema 110 de control de Ethernet para proporcionar conectividad inalámbrica a la red de datos.

Los siguientes protocolos se pueden añadir al presente aparato 100 de red para proporcionar funcionalidad expandida: DHCP y RARP para asignación automática de direcciones IP; IGMP para suscribirse a grupos de multidifusión; RTSP para recuperar correo de voz y señales de timbre distintivas; SAP para escuchar anuncios de eventos de "radio" de multidifusión; y DNS para la resolución de nombres (sujeto al espacio disponible en la memoria del programa).

Además de las operaciones básicas de telefonía, el presente aparato de red también puede proporcionar funciones de telefonía de alto nivel. Por ejemplo, se puede proporcionar una característica de "No molestar" que reenvía automáticamente las llamadas durante una duración determinada a una ubicación designada de acuerdo con lo especificado por una entrada de dirección SIP por parte del usuario. Cada vez que se selecciona la función, como al presionar un botón en la interfaz del usuario, el tiempo aumenta en un intervalo predeterminado (por ejemplo, 15 minutos).

El "registro de llamadas" también se puede proporcionar cuando la dirección SIP e información relacionada con las llamadas entrantes se registra almacenando la información en la memoria, con la capacidad de devolver la llamada a la persona que llama desplazándose por la lista y seleccionando la dirección SIP de la parte que llama del registro por interacción del usuario a través del subsistema 160 de interfaz de usuario.

El aparato de red también puede incluir una "libreta de direcciones automática". A través de la entrada del usuario o a través de un servidor conectado a la red, el aparato de red puede adquirir una lista de marcado rápido o una lista de nombres almacenados en su memoria local por la que el usuario puede desplazarse (usando la respuesta de "opciones múltiples" de SIP).

Una característica de "sistema de interfaz a correo de voz" puede mostrar todas las llamadas no contestadas que han llegado, incluida la hora de la llamada, la persona que llama, el tema y la urgencia de la llamada y si la persona que llamó dejó el mensaje de voz. Las llamadas pueden ordenarse cronológicamente o por urgencia. La pantalla de llamada presenta preferiblemente cinco botones programables: para eliminar la entrada, avanzar y retroceder en la lista, devolver la llamada y recuperar el mensaje.

El "timbre distintivo" es una característica en la que el aparato 100 está programado para anunciar ciertas llamadas mediante un clip de sonido distintivo, como un timbre distintivo, una melodía o el nombre de la persona que llama. En este caso, una pequeña base de datos asocia al que llama, o una clase de partes que llaman (por ejemplo, amigo, cliente, urgente) a una respuesta de timbre seleccionada en particular. El clip de sonido se reproduce desde la memoria o desde un servidor.

El "reenvío de llamadas" es una característica adicional que puede implementarse en el aparato 100. Normalmente, las llamadas se reenvían por el servidor de redirección de proxy. Sin embargo, el aparato 100 de red también puede realizar un simple reenvío, como se describió anteriormente para el botón "no molestar". La redirección puede tomar la forma de llamar al teléfono desde otro teléfono con un comando de REGISTRO, para implementar llamadas de seguimiento. Además, el reenvío automático de llamadas desde ciertos dominios o durante ciertas horas se implementa fácilmente sin el uso de un servidor de redirección.

El modo "Intercomunicador" es una función en la que las llamadas entrantes se "recogen" automáticamente, con el micrófono desactivado hasta que se presiona un botón de pulsar para hablar o se levanta el receptor. Esto también se puede usar como parte de un sistema de direcciones públicas de seguridad.

Las funciones de "monitorización de bebés" permiten que el aparato de red actúe como un dispositivo de monitorización de audio remoto. Por ejemplo, al recibir una llamada entrante, el aparato 100 de red se activa con el altavoz desactivado pero con el micrófono habilitado automáticamente de forma que la parte que llama pueda escuchar el entorno en el que se encuentra el dispositivo llamado. Esta característica se puede activar selectivamente, tal como mediante un código predeterminado o identidad de la persona que llama.

Una característica de "radio por Internet" permite que el aparato 100 de red reproduzca automáticamente estaciones de radio suministradas por un servidor de multidifusión RTP local u otra fuente de medios de transmisión cuando no se recibe o inicia una llamada. El aparato 100 puede escuchar anuncios de SAP y puede mostrar la lista de estaciones en la pantalla, con botones programables. Cualquier llamada entrante interrumpe el programa de radio actual.

El presente aparato de red también puede mantener una "Lista de llamados". Si una llamada anterior fue exitosa, la dirección del llamado se introduce automáticamente en una porción de la memoria usada como lista local de marcado de guía. Cuando se deba volver a marcar a esta parte, el destinatario puede ser seleccionado con la tecla hacia arriba o hacia abajo de la lista del destinatario. En general, esta es una estructura de memoria de tipo FIFO que purga automáticamente las entradas antiguas y las reemplaza con más entradas actuales; y

"Volver a marcar", que permite marcar con una sola tecla el último número marcado o el último llamado.

Además, la "mejora de procesamiento de voz", tal como la supresión de silencio, la generación de ruido de confort y la cancelación de eco, también pueden incluirse en el presente aparato de red de una manera que es bien conocida en la técnica de la telefonía. Por lo tanto, se ha divulgado un teléfono basado en red que es un "aparato de red" independiente que permite al usuario hacer llamadas telefónicas dentro de una red de área local (LAN) o a través de Internet. El núcleo del aparato de red es un procesador de señal digital única (DSP) (un microcontrolador optimizado para procesar datos de audio y video). Proporciona servicios que son un superconjunto de los de un teléfono normal, pero se conectan a una red de datos Ethernet en lugar de a la PSTN (red telefónica pública de conmutación). Dado que Ethernet se ejecuta a 10 Mb/s puede usar el mismo cableado de par trenzado usado para teléfonos analógicos y digitales, el teléfono de la red de datos por paquetes no requiere la reconfiguración de las instalaciones del cliente. Un sistema mínimo consiste en dos teléfonos de red de datos por paquetes conectados por un cable cruzado Ethernet. Se puede implementar un PBX básico de varias líneas que consiste en cualquier número de teléfono de red de datos por paquetes conectado a un conector o conmutador Ethernet. Este "PBX" puede escalar cualquier número de teléfonos, simplemente añadiendo capacidad Ethernet y puertos. El teléfono de red de datos por paquetes comparte Ethernet con otros servicios LAN. En casi todos los casos, el tráfico de voz será una pequeña fracción de la capacidad de la red. (Una sola llamada de voz consume aproximadamente 16 kb/s de la capacidad de 10 Mb/s). El teléfono de la red de datos por paquetes ofrece comunicaciones de voz, implementando las funciones habituales de los PBX. Sin embargo, el presente aparato de red puede usar un servidor ubicado en la LAN o Internet para proporcionar una funcionalidad adicional, como la ubicación del usuario y los servicios de directorio, reenvío de llamadas, correo de voz, servicios de asistente.

Un PBX basado en el aparato de red actual puede llegar a los teléfonos tradicionales a través de una pasarela de telefonía por Internet (ITG). Dicha pasarela se conecta a la PSTN usando líneas analógicas, las interfaces ISDN básicas o de velocidad primaria o troncales digitales (como T1/E1). Los ITG se han introducido recientemente como productos comerciales, con capacidades de una a alrededor de 240 líneas.

La figura 34 es un diagrama de bloques de una arquitectura de sistema para habilitar servicios inalámbricos en una red de telefonía por Internet. El sistema incluye uno o más aparatos 100 de telefonía de red que preferiblemente funcionan de acuerdo con el protocolo SIP. Los aparatos 100 de red SIP están acoplados a una red 3402 de datos, tal como una intranet basada en Ethernet, y funcionan generalmente junto con uno o más servidores proxy SIP 3325 que pueden usarse como servidores de redirección y pueden proporcionar una funcionalidad mejorada. También se proporcionan un servidor proxy 3405 de servicios inalámbricos y una pasarela 3410 de servicios inalámbricos. El servidor proxy 3405 de servicios inalámbricos puede formarse como parte de la pasarela 3410 de servicios inalámbricos o puede ser unidades operativas separadas como se muestra. Cuando se forman como unidades separadas, el servidor proxy 3405 de servicios inalámbricos se puede acoplar y estar en comunicación con la pasarela 3410 de servicios inalámbricos a través de una conexión dedicada tal como se muestra o mediante la red 3402 de datos. En una realización, el servidor proxy inalámbrico 3405 y la pasarela 3410 de servicios inalámbricos son comparables con el protocolo WAP. Sin embargo, otros servicios inalámbricos también pueden ser compatibles para proporcionar servicios de telefonía mejorados 2G, 2.5G y 3G.

Se contempla además que uno o más aparatos 3415 de red inalámbricos se pueden desplegar en el sistema. Como se describió anteriormente, esta funcionalidad puede proporcionarse incluyendo un módulo inalámbrico, tal como un módulo compatible con Bluetooth o IEEE 802.11 en el aparato 100 de red y teniendo una interfaz de red inalámbrica adecuada como parte de la red 3402 de datos. Esto se ilustra mediante el enlace de comunicación inalámbrico 3420.

La pasarela WAP 3410 es un sistema informático que admite comunicaciones con una infraestructura de red inalámbrica y comunica mensajes hacia y desde la infraestructura de red inalámbrica en respuesta al servidor proxy WAP 3405. El servidor proxy WAP generalmente se interpone entre la pasarela WAP 3410 y el aparato 100 de red. El servidor proxy WAP 3405 recibe mensajes de acuerdo con el protocolo WAP de la pasarela WAP 3410 y traslada estos mensajes a una forma que puede visualizarse fácilmente en el aparato 100 de red. El servidor proxy WAP 3405 traslada la pila del protocolo WAP, que incluye WML y WMLscript, a una señal de interfaz de terminal simple que es compatible con las propiedades de visualización del aparato de red, tal como la pantalla LCD descrita anteriormente. De manera similar, cada operación de visualización y pulsación desde el aparato 100 de red puede transmitirse al servidor proxy WAP 3405 como una solicitud de mensaje SIP, donde se traslada a un mensaje adecuadamente formateado para el protocolo inalámbrico correspondiente, tal como WAP.

Cuando se usa un servidor proxy WAP, el aparato de red no necesita ser un dispositivo compatible con WAP, ya que el servidor proxy WAP 3405 actúa como un controlador para el protocolo WAP. Esto permite que el aparato de red funcione sin tener que cargar con el procesamiento relacionado con el protocolo WAP. Sin embargo, si se desea, como se ilustra en la figura 3, la arquitectura de pila de software para el aparato de red puede incluir una capa 99 de protocolo inalámbrico.

Durante el funcionamiento normal en la arquitectura de la figura 34, el aparato 100 de red transmitirá mensajes de MENSAJE SIP a un servidor proxy SIP 3325 designado que enruta los mensajes al servidor proxy WAP 3405. Sin embargo, el aparato 100 de red puede intercambiar mensajes de MENSAJE SIP directamente con el servidor proxy WAP 3405, si se prefiere.

5 El presente sistema proporciona un entorno en el que SIP proporciona el protocolo de habilitación para que los aparatos de red realicen numerosas aplicaciones inalámbricas avanzadas, tales como servicios de telefonía de segunda generación (2G), servicios de telefonía de segunda generación extendida (2.5G) y aplicaciones inalámbricas de tercera generación (3G).

10 Una de estas características es la entrega de contenido a un aparato 100 de red cuando el sistema detecta que el aparato está en modo inactivo, como cuando no hay llamadas en curso. El contenido, tal como un anuncio proporcionado por un patrocinador de contenido, puede entregarse al aparato de red, y si el usuario habilita la característica, el contenido puede mostrarse en la pantalla del aparato 100 de red. El contenido puede incluir un URL, una dirección SIP o similar que se puede mostrar y que el usuario del aparato de red puede seleccionar para vincular automáticamente al patrocinador del contenido. El enlace puede ser una llamada de voz iniciada por el usuario, un mensaje SMS o un enlace a una ubicación de Internet especificada por el URL entregado en el contenido. En una realización, el contenido recibido puede configurarse para reconfigurar automáticamente una "tecla programable" en el aparato de red. En este caso, al presionar el botón programado en el aparato, se iniciará el enlace.

20 Aunque la presente invención se ha descrito en conexión con realizaciones particulares de la misma, debe entenderse que los expertos en la técnica pueden realizar diversas modificaciones, alteraciones y adaptaciones sin salir del espíritu y alcance de la invención. Se pretende que la invención esté limitada únicamente por las reivindicaciones adjuntas.

REIVINDICACIONES

- 1.- Un sistema de comunicación que comprende un dispositivo de aparato (100) de red acoplado a una red (3402), comprendiendo el dispositivo (100) de aparato software para detectar llamadas entrantes e iniciar sesiones de llamadas de acuerdo con un protocolo de iniciación de señal, SIP; una pasarela (3410) de comunicación acoplada a la red (3402), proporcionando la pasarela (3410) de comunicación comunicaciones entre el dispositivo (100; 3415) de aparato de red y dispositivos fuera de la red (3402) de acuerdo con un protocolo de comunicaciones inalámbricas; y un servidor proxy (3405) de comunicación inalámbrica acoplado a la red (3402) para trasladar mensajes entre el SIP y el protocolo de comunicación inalámbrica, por lo que el dispositivo (100) de aparato de red comprende una pantalla (164), el dispositivo (100) de aparato de red adaptado para recibir mensajes trasladados por el servidor proxy (3405) de comunicación inalámbrica desde el protocolo de comunicación inalámbrica hasta el SIP cuando el dispositivo (100) de aparato de red está en modo inactivo, en el que los mensajes comprenden contenido y el dispositivo (100) de aparato de red adaptado para mostrar el contenido en la pantalla (164) cuando el dispositivo (100) de aparato de red está en modo inactivo.
- 2.- El sistema de la reivindicación 1, caracterizado porque el protocolo de comunicación inalámbrica es el protocolo de aplicación inalámbrica (WAP).
- 3.- El sistema de la reivindicación 1, caracterizado porque al menos un servidor proxy (3325) de SIP está acoplado a la red (3402) para intercambiar mensajes SIP con el dispositivo (100) de aparato de red con el servidor proxy (3405) de comunicación inalámbrica.
- 4.- El sistema de la reivindicación 1, caracterizado porque el que el dispositivo (100) de aparato de red y el servidor proxy (3405) de comunicación inalámbrica intercambian mensajes SIP directamente.
- 5.- El sistema de la reivindicación 1, caracterizado porque una pulsación de tecla en el dispositivo (100) de aparato de red se proporciona al servidor proxy (3405) de comunicación inalámbrica como una única solicitud SIP.
- 6.- El sistema de la reivindicación 1, caracterizado porque la pasarela (3410) de comunicación y el servidor proxy (3405) de comunicaciones inalámbricas son un sistema integrado.
- 7.- El sistema de comunicación de acuerdo con la reivindicación 1 a 6, caracterizado porque al menos un servidor SIP (3325) está acoplado a la red para intercambiar mensajes SIP con el dispositivo (100) de aparato de red; y el dispositivo (100) de aparato de red recibe mensajes trasladados por la pasarela (3410) de comunicación que comprenden contenido publicitario para la pantalla (164).
- 8.- Un método de comunicación con un dispositivo (100) de aparato de red en una red (3402), que comprende, en el dispositivo (100) de aparato de red, detectar llamadas entrantes e iniciar sesiones de llamadas de acuerdo con un protocolo de inicio de sesión, SIP, usar una pasarela (3410) de comunicación para proporcionar comunicaciones entre el dispositivo (100) de aparato de red y dispositivos fuera de la red (3402) de acuerdo con un protocolo de comunicaciones inalámbricas; recibiendo el dispositivo (100) de aparato de red mensajes trasladados por un servidor proxy (3405) de comunicación inalámbrica desde el protocolo de comunicación inalámbrica al SIP cuando el dispositivo (100) de aparato de red está en modo inactivo, en el que los mensajes comprenden contenido; mostrar el contenido cuando el dispositivo (100) de aparato de red está en modo inactivo; y usar el servidor proxy (3405) de comunicación inalámbrica para trasladar mensajes entre el SIP y el protocolo de comunicación inalámbrica.
- 9.- El método de la reivindicación 8, caracterizado porque el protocolo de comunicación inalámbrica es el protocolo de aplicación inalámbrica, WAP.
- 10.- El método de la reivindicación 8, caracterizado porque uno o más servidores SIP (3325) son acoplados a la red (3402) para intercambiar mensajes SIP con el dispositivo (100) de aparato de red y con el servidor proxy (3405) de comunicación inalámbrica.
- 11.- El método de la reivindicación 8, caracterizado por intercambiar directamente mensajes SIP entre el dispositivo (100) de aparato de red y el servidor proxy (3405) de comunicación inalámbrica.

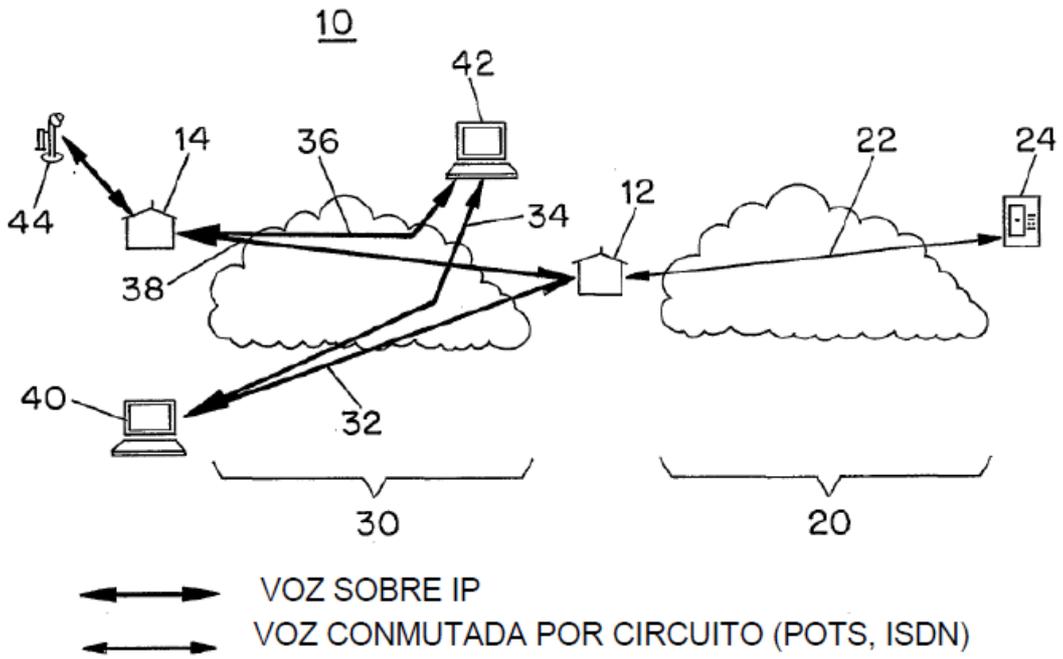


FIG. 1

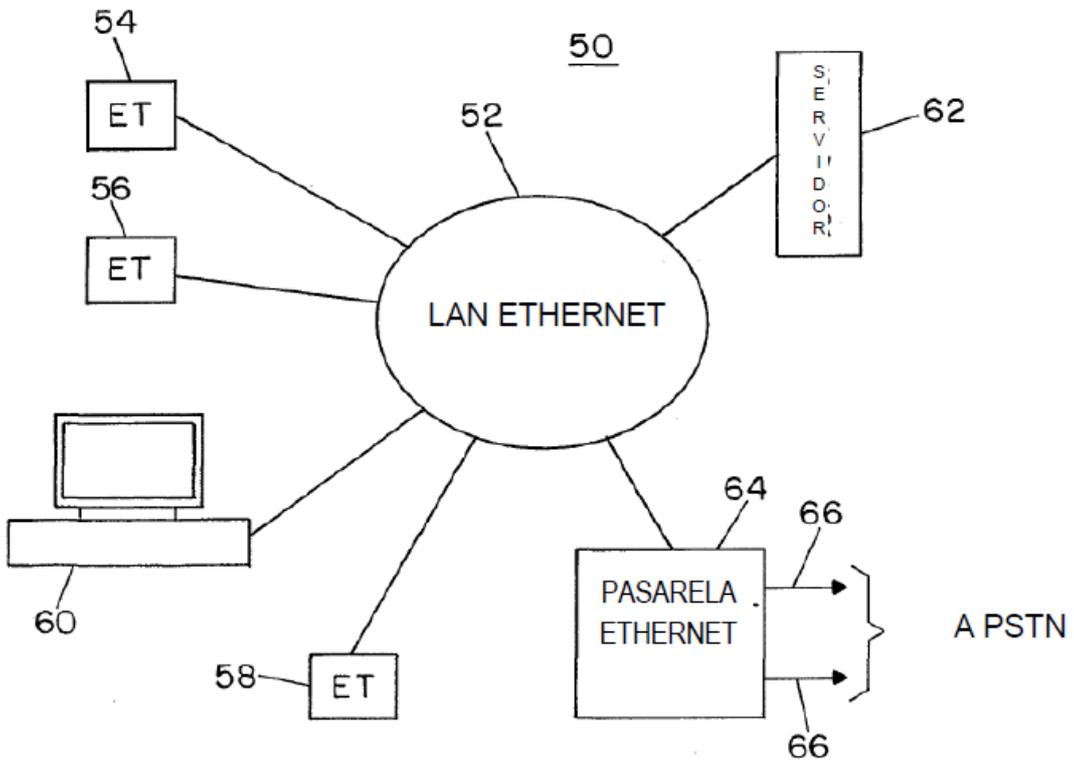


FIG. 2

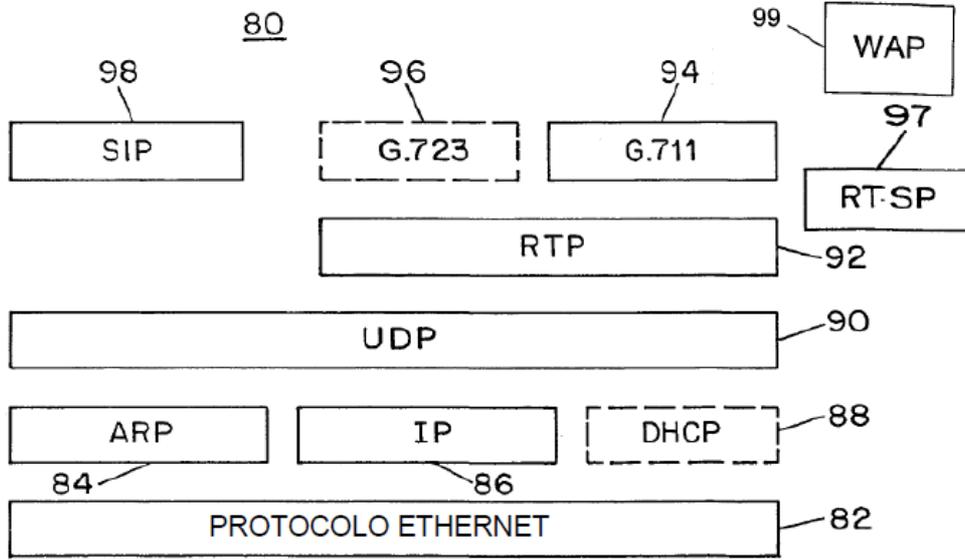


FIG. 3

Dirección DSP	Longitud (hex)	(dec.) Uso
0x1000	0xB000	128K ROM
0x20000	0x10	16 LCD
0x810000	0x20	32 Controlador Ethernet
0x820000	0xf	16 lectura de teclado, amplificador de audio, control, control de colgado, restablecer software del sistema
0x87fe00	0x200	512 RAM interna
0x900000	0x8000	32K Ram externa

FIG. 6

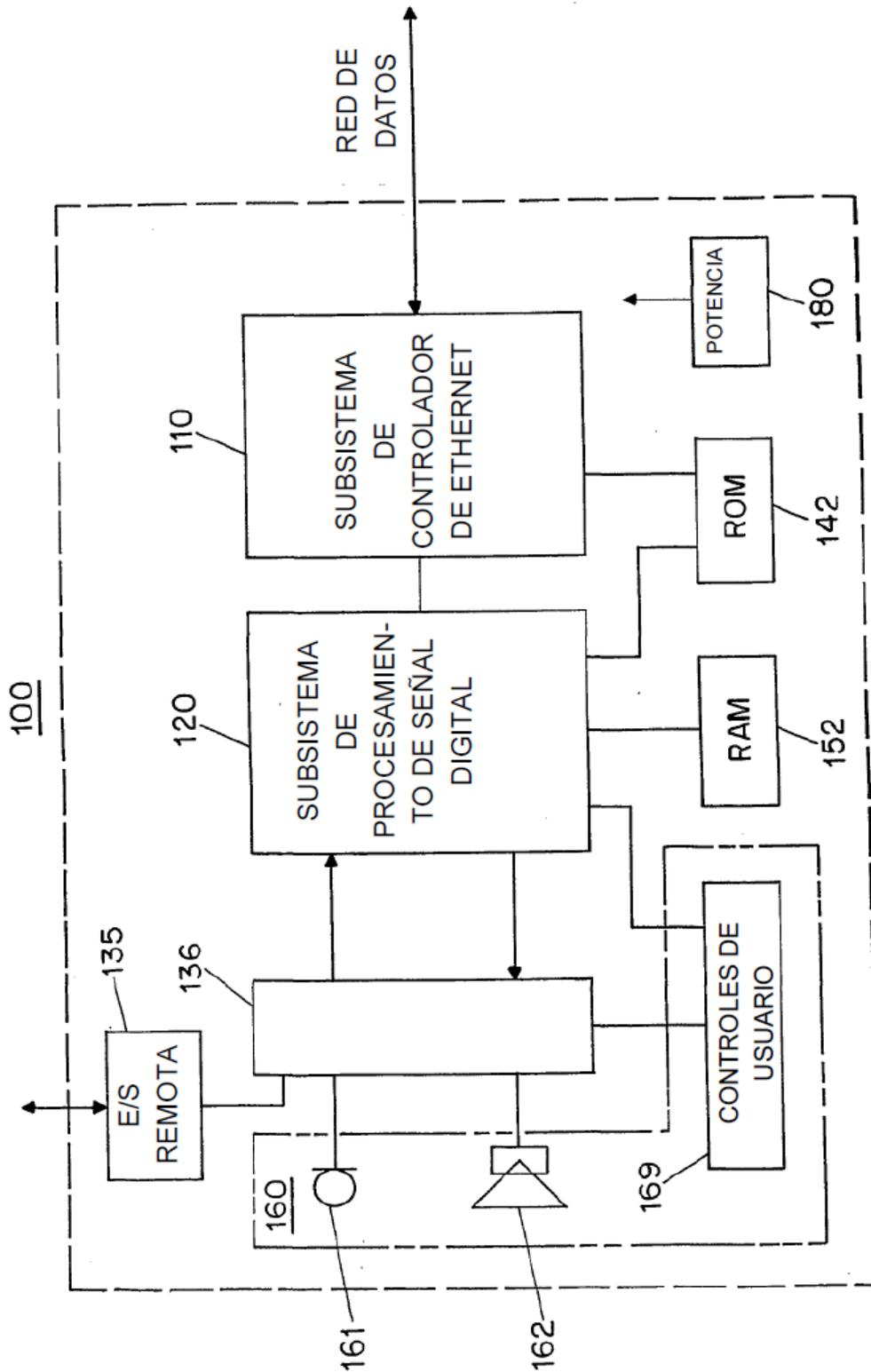


FIG. 4

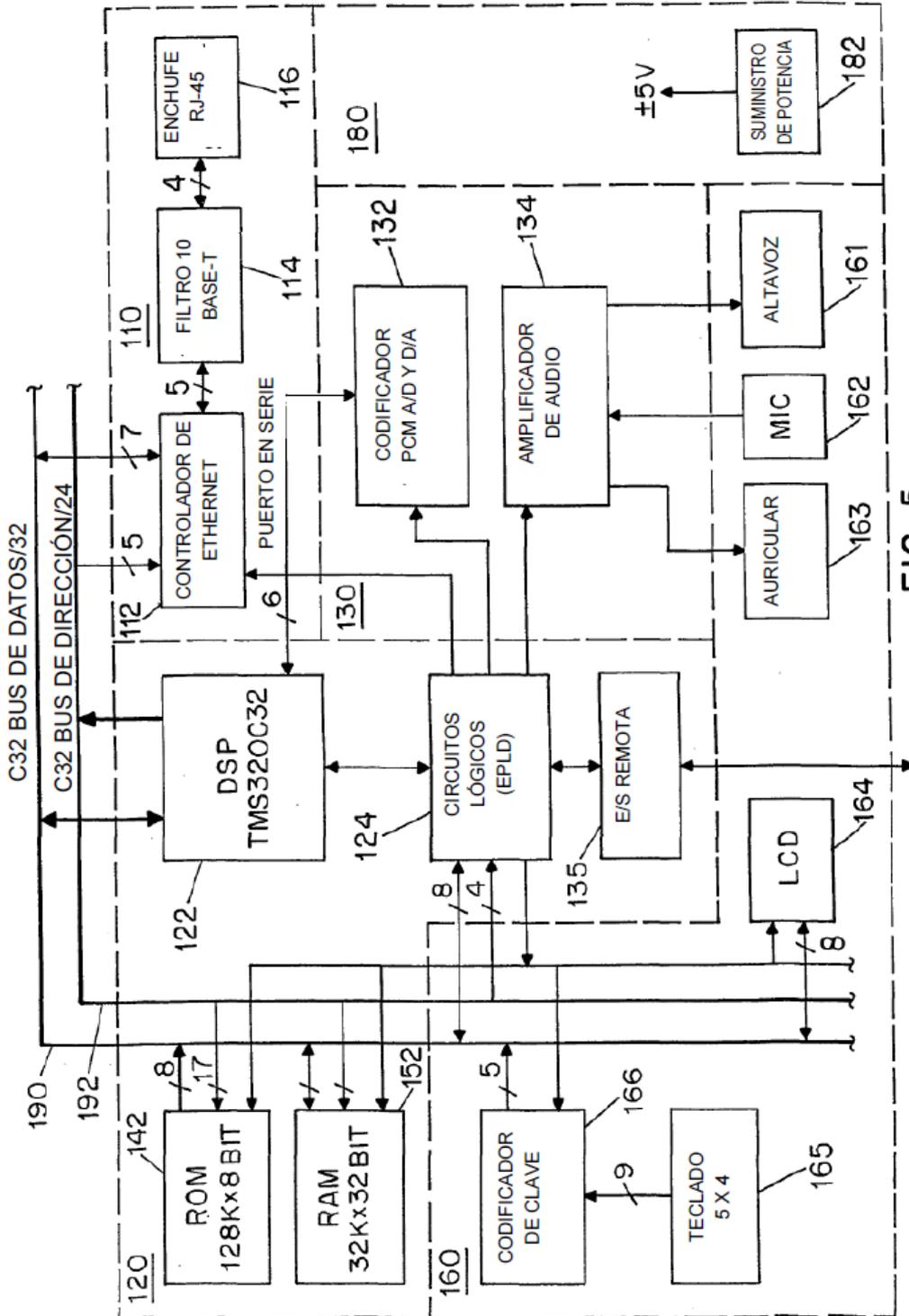


FIG. 5

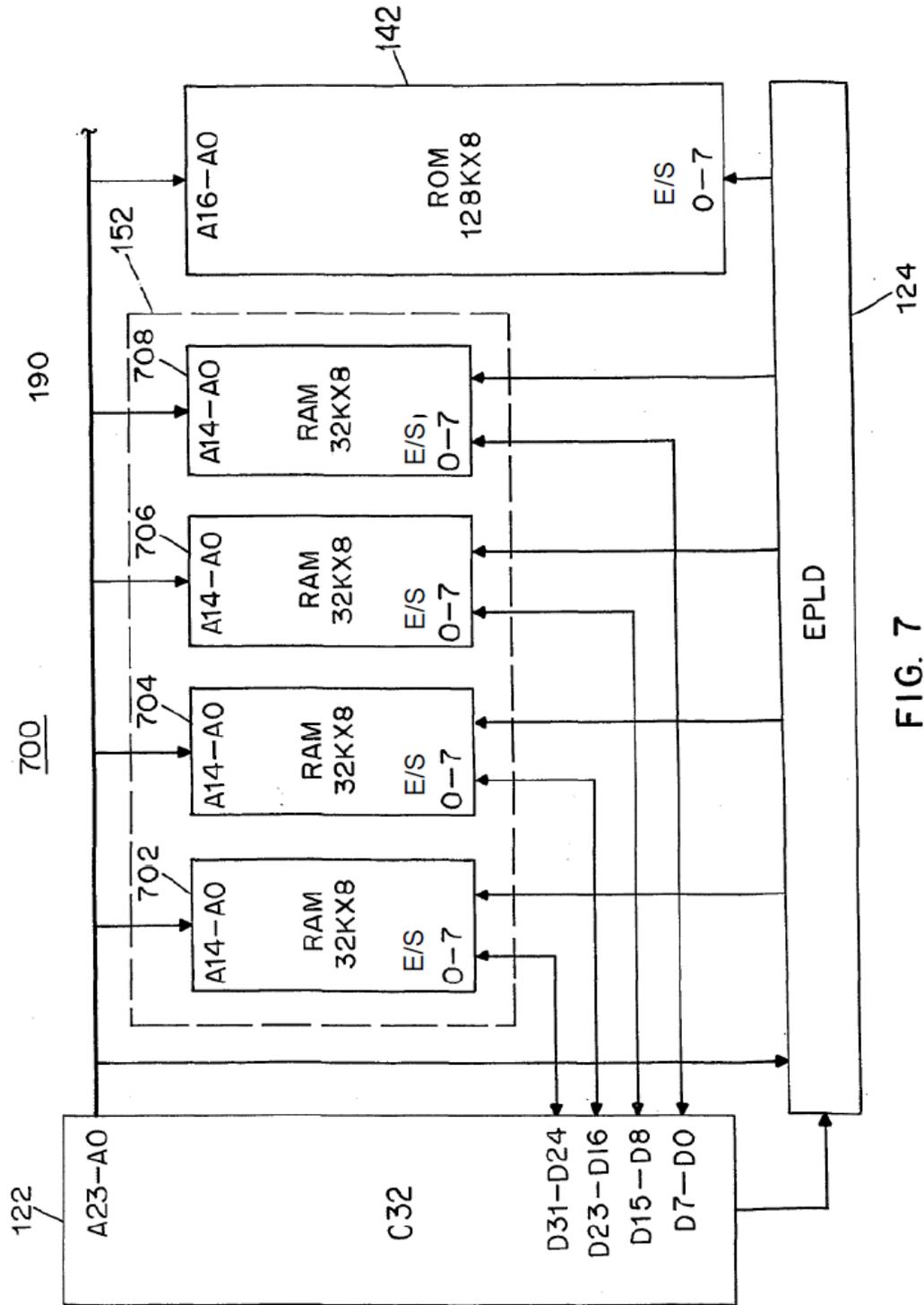


FIG. 7

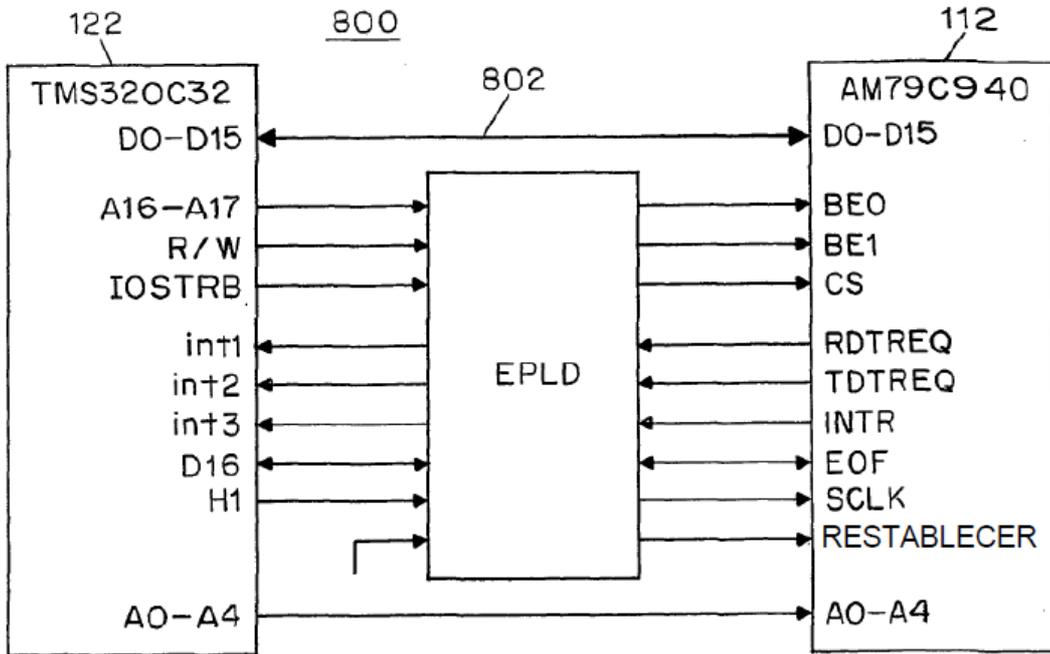


FIG. 8

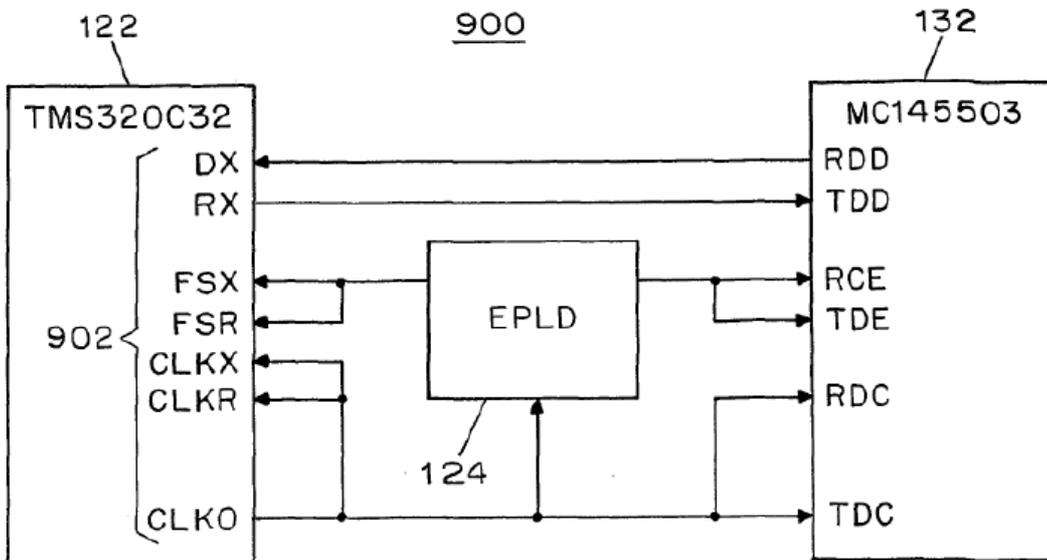


FIG. 9

Dirección DSP	Uso
0x2000	Puerto de comando para mitad izquierda de LCD
0x2001	Puerto de datos para mitad izquierda de LCD
0x2002	Puerto de comando para mitad derecha de LCD
0x2003	Puerto de datos para mitad derecha de LCD

FIG. 10

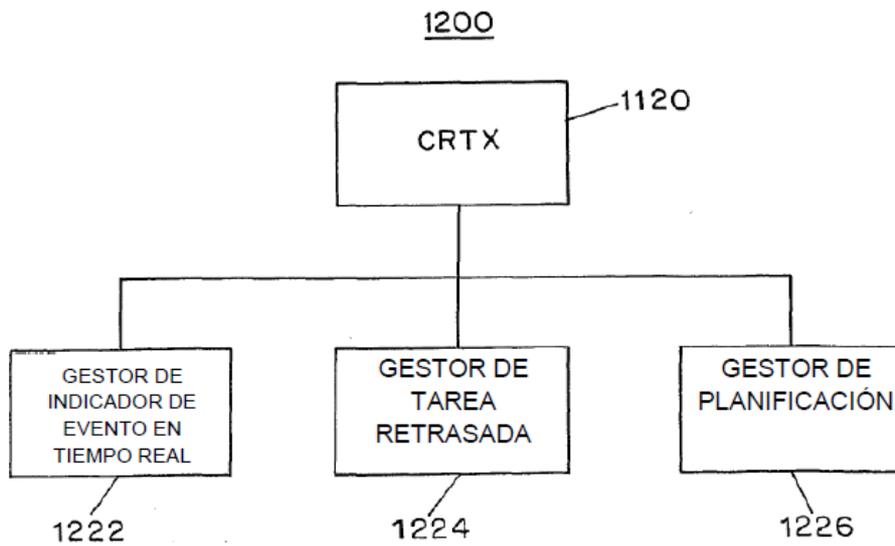


FIG. 12

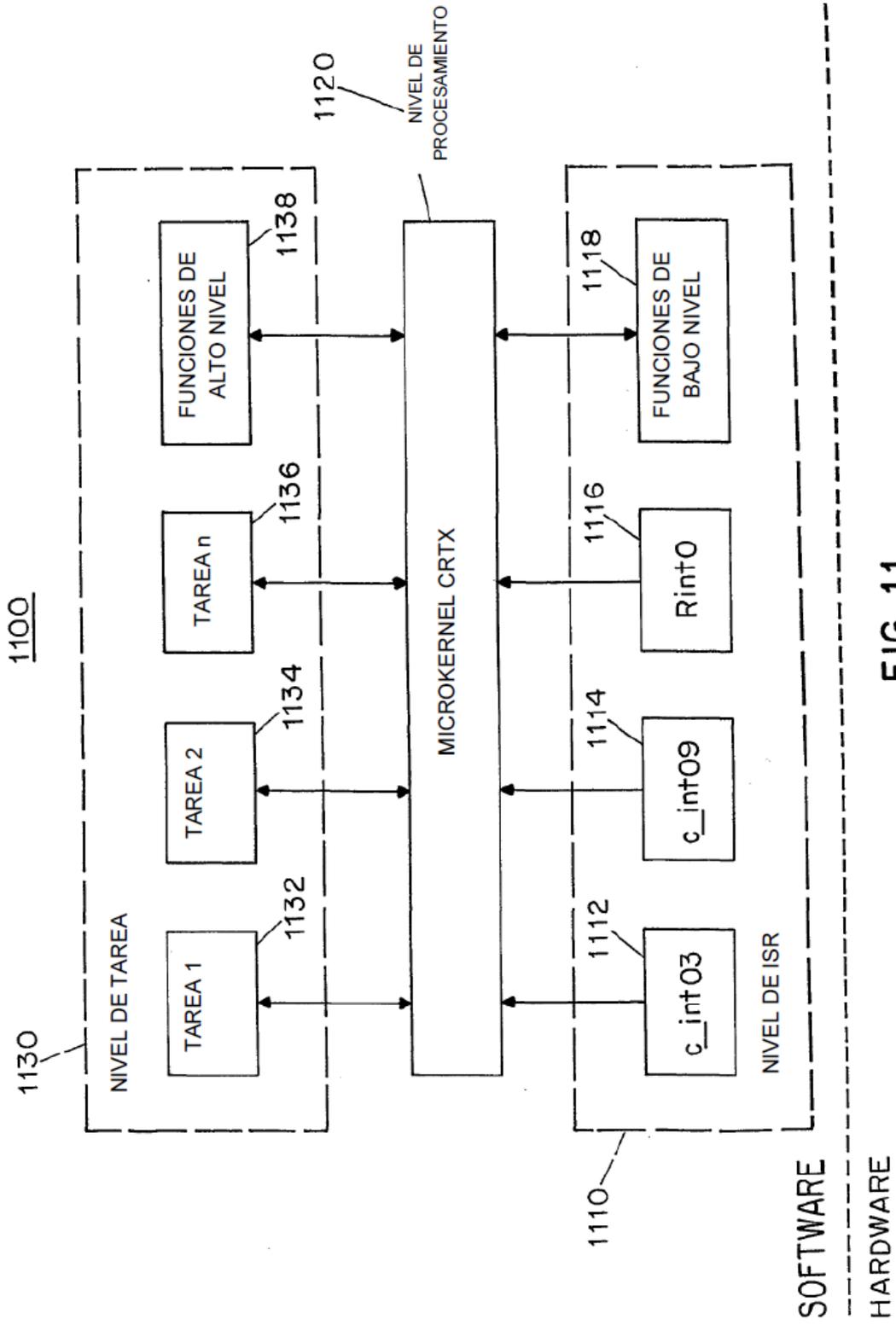


FIG. 11

Nombre de función	Función
<i>ARPinit()</i>	(Función de inicialización) Inicialización de tabla ARP.
<i>c_init00()</i>	(Función de inicialización) Programa principal, inicializar el puntero de pila, interfaz de bus externo, y vector de interrupción para TMS320C32.
<i>DMA_initialize()</i>	(Función de inicialización) Inicializar los canales DMA0 y DMA1.
<i>ENET_initialize()</i>	(Función de inicialización) Inicializar el controlador de Ethernet.
<i>InitHardWare()</i>	(Función de inicialización) Inicializar el Timer0, Timer1 y puerto en serie.
<i>NameInit()</i>	(Función de inicialización) Inicializar algunos encabezados SIP y cuerpo.
<i>SerialPortInit()</i>	(Función de inicialización) Inicializar el puerto en serie.
<i>ARP_In_task()</i>	Analizar los paquetes de entrada ARP.
<i>ARPTimer_task()</i>	Temporizador ARP, mantener la tabla ARP.
<i>Call_task()</i>	Procesamiento de llamada.
<i>Clock_task()</i>	Un reloj genera la hora, minuto y segundo.
<i>Codec_task()</i>	Una tarea para codificación, decodificación, generación de señal de llamada, generación de tono o bucle de memoria de llamada.
<i>CreateSipCall()</i>	Crear un paquete de solicitud SIP para una llamada.
<i>Ercv_task()</i>	Receptor de paquetes de Ethernet y demultiplexado IP.

FIG. 13A

Nombre de función	Función
<i>IP_Send_task()</i>	Multiplexado IP y envío de paquete Ethernet
<i>Key_task()</i>	Monitor de teclado y salida
<i>RTP_In_task()</i>	Procesamiento RTP
<i>Sendto()</i>	Enviar paquetes UDP a dirección IP dada
<i>Setting_task()</i>	Establecer los parámetros de E*teléfono
<i>SIP_In_task()</i>	Aceptar paquetes SIP, y actualizar estatus de llamada y SIP
<i>SIP_task()</i>	Tarea de transición de estatus SIP
<i>Tone_task()</i>	Contar la duración activa y de parada para tono o llamada
<i>UDP_In_task()</i>	Aceptar paquetes UDP
<i>ARP_Out()</i>	(Función de alto nivel) Programa de solicitud ARP
<i>ClearScreen()</i>	(Función de alto nivel) Despejar todas las líneas en la LCD
<i>CodecConfig()</i>	(Función de alto nivel) Planificar una tarea de códec de acuerdo con el parámetro de modo de ejecución
<i>Disp()</i>	(Función de alto nivel) Mostrar una secuencia en la pantalla LCD
<i>LCD()</i>	(Función de alto nivel) Mostrar un carácter en la pantalla LCD
<i>LCDClear()</i>	(Función de alto nivel) Despejar una línea en la pantalla LCD
<i>LinearToUlaw()</i>	(Función de alto nivel) Conversión de datos lineales a datos de ley μ

FIG. 13B

Nombre de función	Función
<i>Initialization()</i>	(Función de alto nivel) Llamar función de inicialización y tareas de preplanificación
<i>RTP_para_init()</i>	(Función de alto nivel) Generar la fecha aleatoria y SSRC para una sesión RTP
<i>ScreenScroll()</i>	(Función de alto nivel) Desplazar la pantalla LCD para una línea hacia arriba o hacia abajo
<i>SDPparse()</i>	(Función de alto nivel) Analizar paquetes SDP
<i>SIPparse()</i>	(Función de alto nivel) Analizar paquetes SIP
<i>SIP_Request()</i>	(Función de alto nivel) Crear mensajes de solicitud SIP
<i>SIP_Response()</i>	(Función de alto nivel) Crear mensajes de respuesta SIP
<i>SpeechDecode()</i>	(Función de alto nivel) Decodificación de voz
<i>SpeechEncode()</i>	(Función de alto nivel) Codificación de voz
<i>ToneGenerate()</i>	(Función de alto nivel) Generar tono de marcado, tono ring back, tono ocupado o tono de alerta

FIG. 13C

Función
Prototipos de función
Archivo de encabezado
Supervisor (núcleo)
Archivo fuente de gestor de trampas

FIG. 13D

Nombre de función	Función
<i>LCDWrite()</i>	(Función de bajo nivel) Escribir datos de visualización a LCD
<i>RintEnable()</i>	(Función de bajo nivel) Habilitar el RINT() para ISR Rint()
<i>RintDisable()</i>	(Función de bajo nivel) Deshabilitar el RINT()
<i>SerialPortRst()</i>	(Función de bajo nivel) Restablecer el puerto en serie
<i>TimerEnable()</i>	(Función de bajo nivel) Habilitar el temporizador del sistema TCLK1
<i>TimerDisable()</i>	(Función de bajo nivel) Deshabilitar el temporizador del sistema TCLK1

FIG. 13F

Nombre de función	Función
<i>c_int03()</i>	ISR de controlador de Ethernet. Activada en INT3 de TMS320C32 por interrupción externa desde AM79C940
<i>c_int09()</i>	ISR de temporizador de sistema. Activada en TINT1 por temporizador 1 interno de TMS320C32
<i>Rint0()</i>	ISR de A/D y D/A. Activada en RINT0 por interrupción de puerto en serie interno de TMS320C32
<i>AmpControl()</i>	(Función de bajo nivel) Controlar el volumen del altavoz
<i>DMA1()</i>	(Función de bajo nivel) Iniciar el canal DMA1
<i>DMA0_Release()</i>	(Función de bajo nivel) Iniciar el canal DMA0
<i>DMA_int_set()</i>	(Función de bajo nivel) Habilitar INT1 e INT2 para DMA0 y DMA1
<i>ENET_reset()</i>	(Función de bajo nivel) Restablecer el controlador de Ethernet
<i>ENET_disable()</i>	(Función de bajo nivel) Deshabilitar el controlador de Ethernet
<i>HandSet()</i>	(Función de bajo nivel) Controlar el cambio de auricular y manos libres
<i>HookState()</i>	(Función de bajo nivel) Verificar el estado enganchado
<i>Key()</i>	(Función de bajo nivel) Verificación y lectura de teclado
<i>KeyMap()</i>	(Función de bajo nivel) Mapear la entrada binaria de clave a formato ASCII
<i>LCDCmd()</i>	(Función de bajo nivel) Comando de control LCD

FIG. 13E

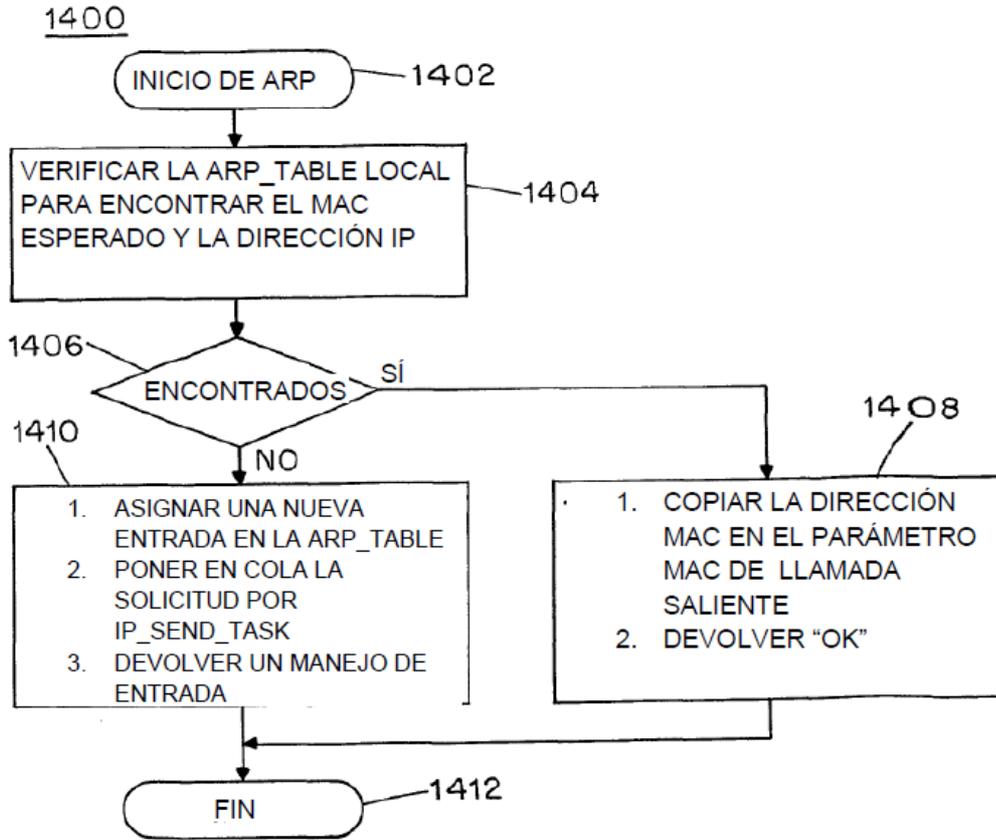


FIG. 14

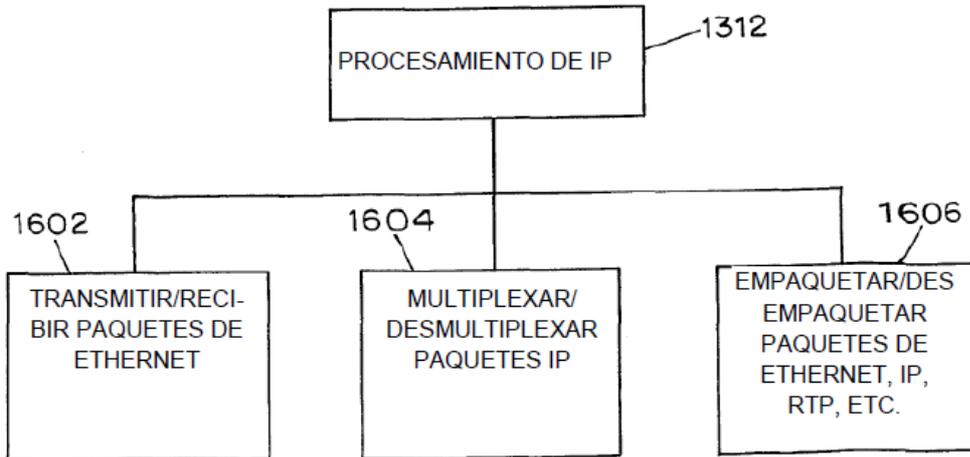


FIG. 16

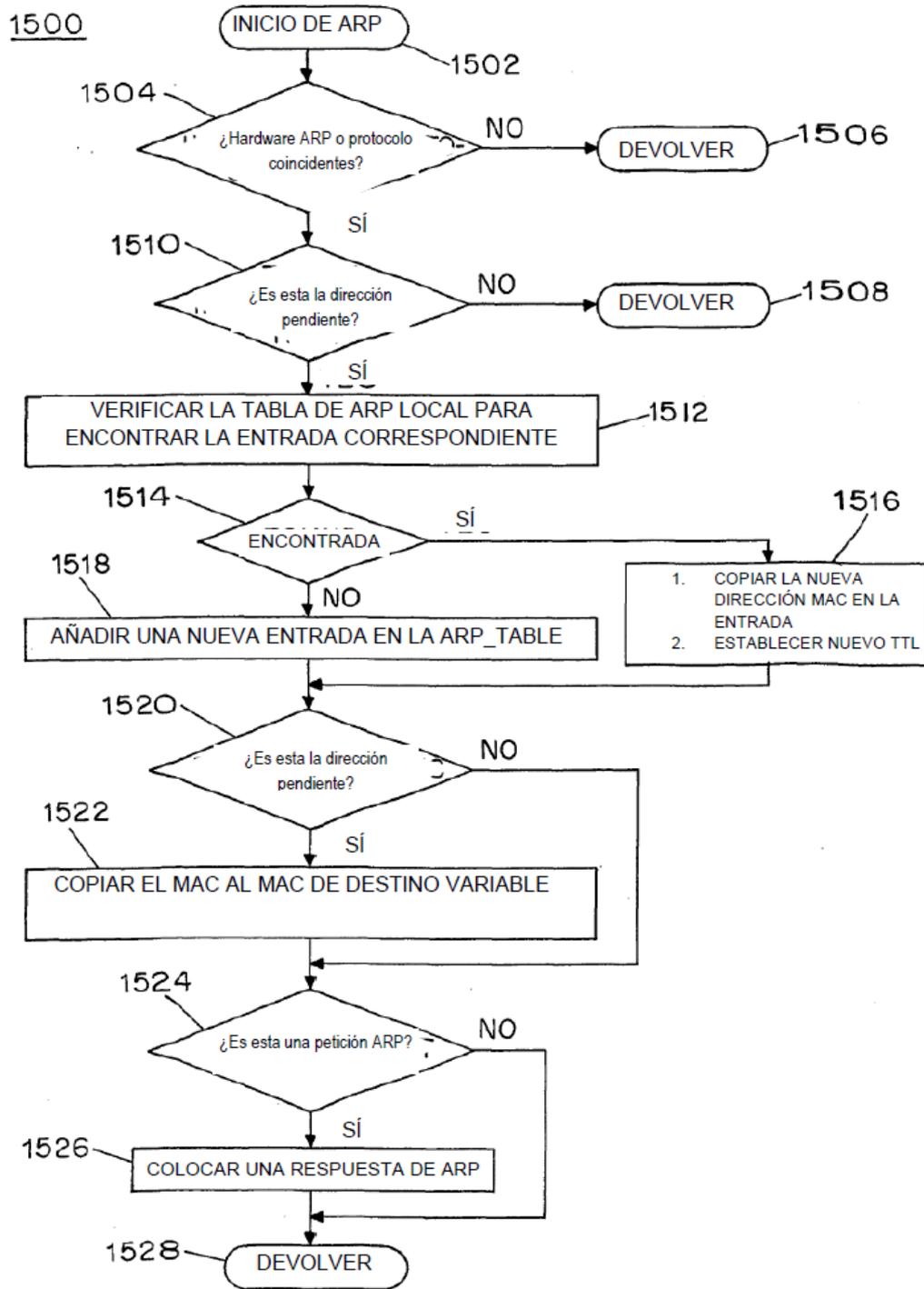


FIG. 15

```

struct ENetHeader {
    ETA Dest;
    ETA Source;
    int Type;
};

struct IPHeader {
    int VI_ToS;
    int Length;
    int Identify;
    int FragDff;
    int TTL_Protocol;
    int ChkSum;
    IPA Source;
    IPA Dest;
};

struct UDPHeader {
    int SPort;
    int DPort;
    int Length;
    int ChkSum;
};

struct EPACKET {
    struct ENetHeader Enh;
    struct IPHeader Iph;
    struct UDPHeader Uh;
    int data[MaxUDPLength];
};

```

/ estructura de encabezado de Ethernet */*
/ dirección de MAC de origen */*
/ dirección de Mac de origen */*
/ tipo de paquete de Ethernet */*

/ estructura de encabezado IP */*
/ versión IP, longitud de encabezado, y tipo de servicio */*
/ longitud total */*
/ identificador del paquete IP */*
/ indicadores y desvío de fragmento */*
/ tiempo de vida, y protocolos */*
/ suma de comprobación */*
/ dirección IP de origen */*
/ dirección IP de destino */*

/ estructura de encabezado UDP */*
/ puerto de origen */*
/ puerto de destino */*
/ longitud de mensaje UDP */*
/ suma de comprobación UDP */*

/ estructura de paquete de recepción Ethernet */*
/ encabezado de Ethernet */*
/ encabezado IP */*
/ encabezado UDP */*
/ campo de datos */*

FIG. 17

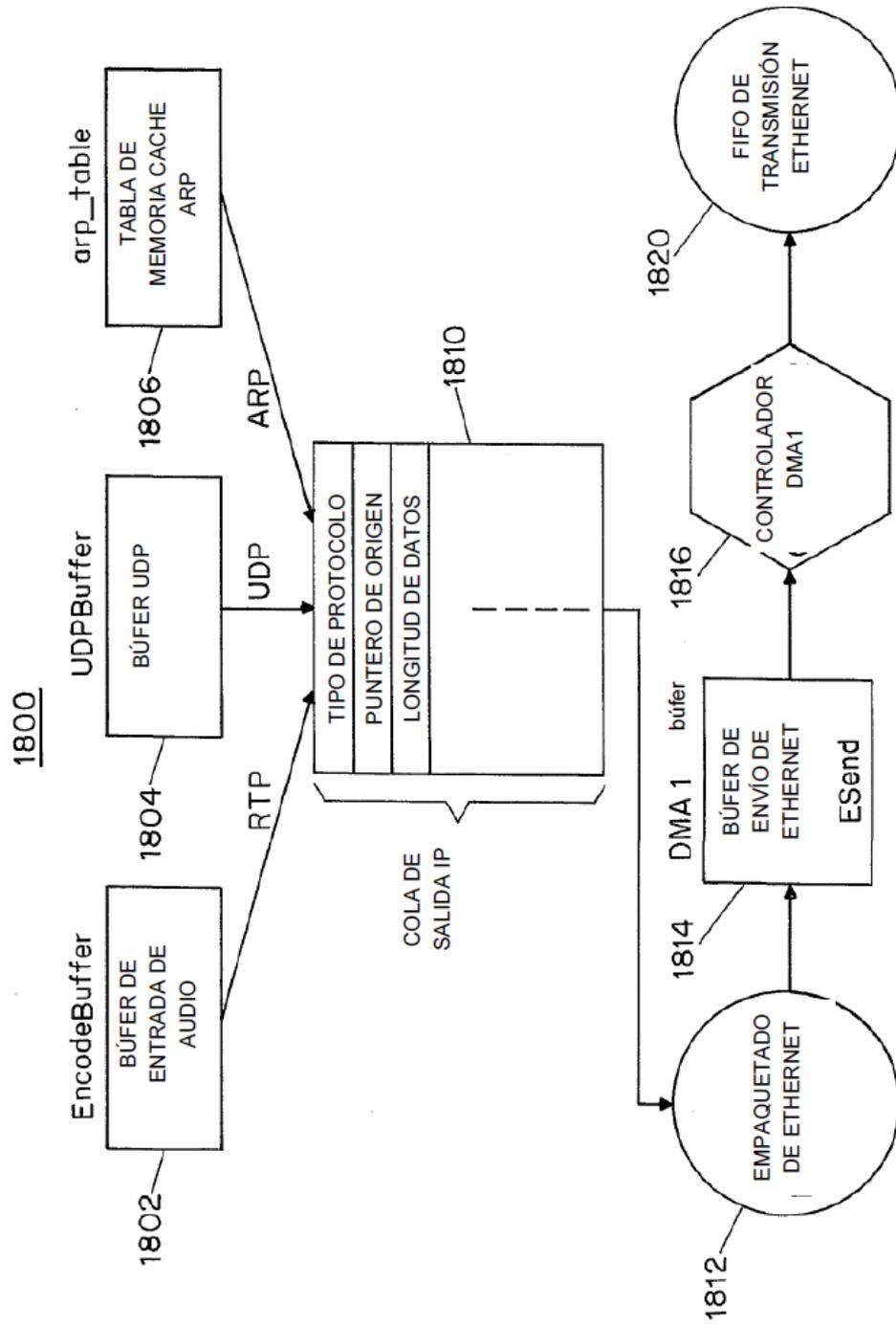


FIG. 18

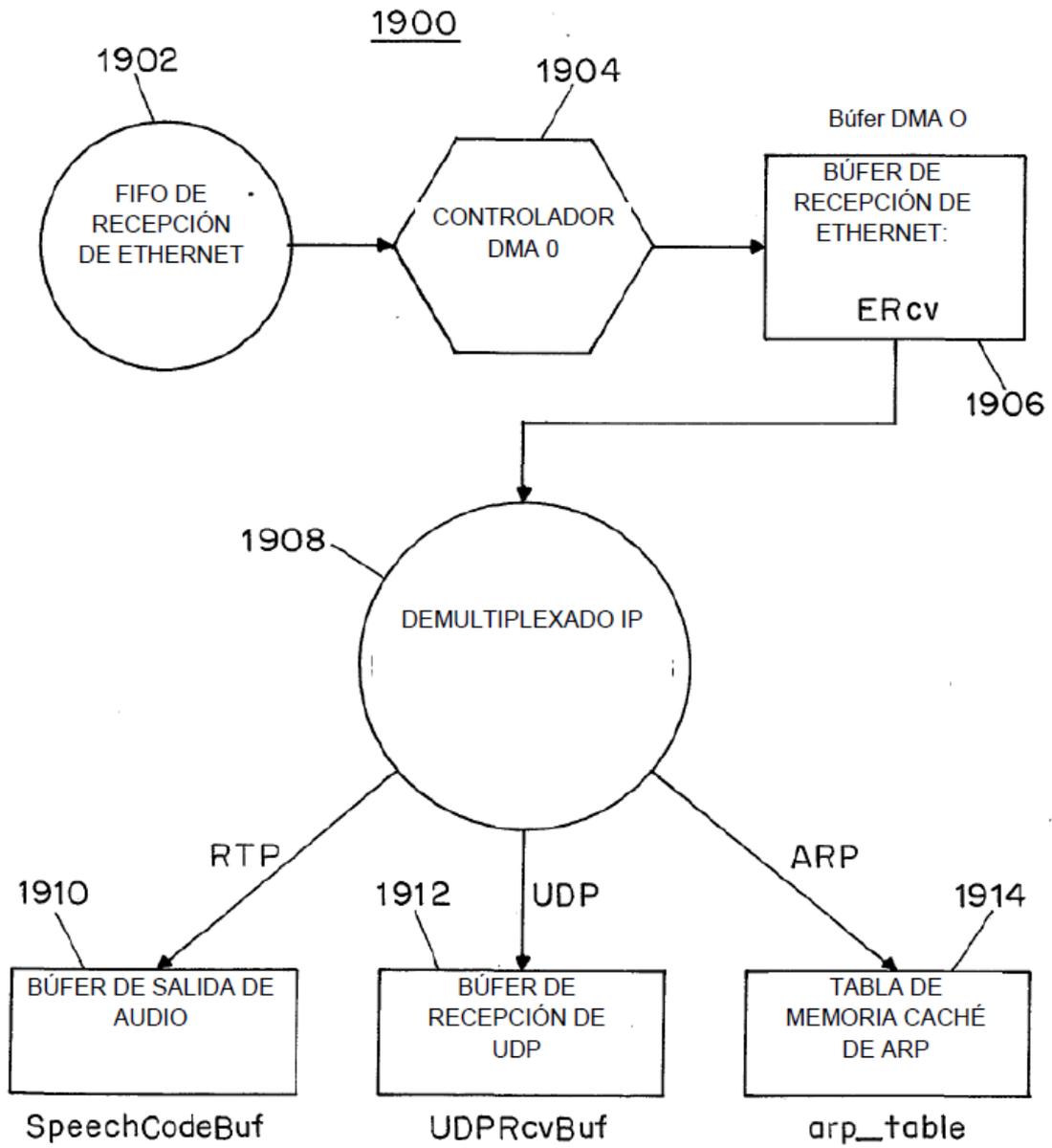


FIG. 19

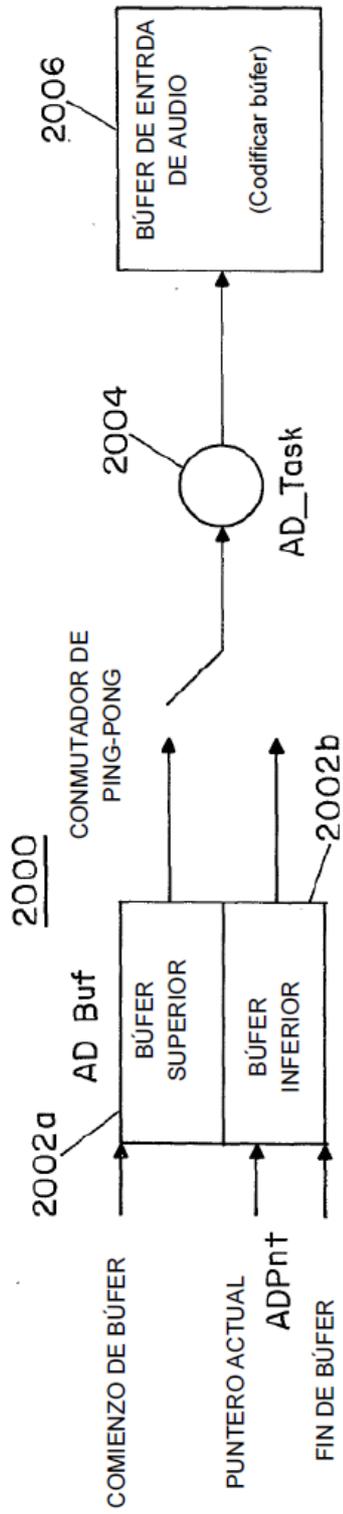


FIG. 20A

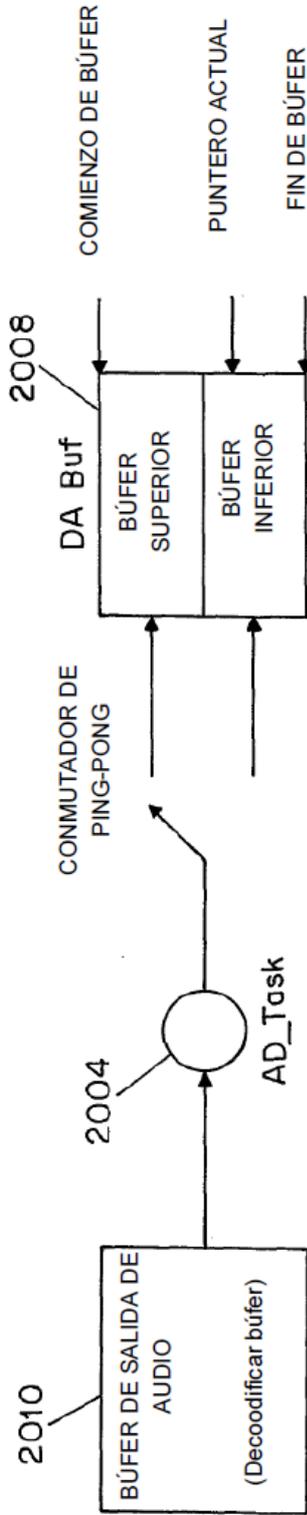


FIG. 20B

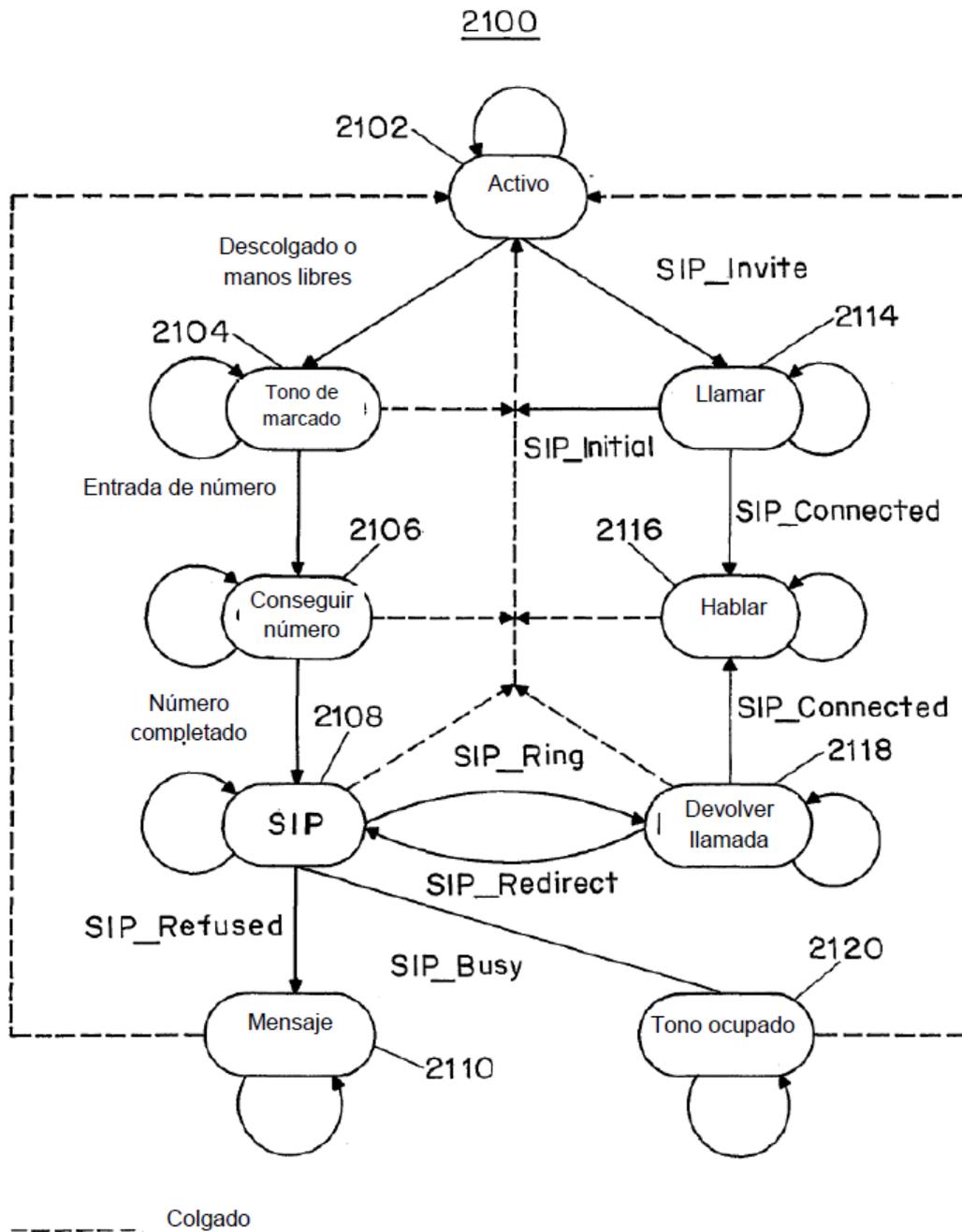


FIG. 21

Tecla	Valor devuelto
Teclas de número	'0' ... '9'
Teclas especiales	'*' y '#'
Tecla de introducción	'E'
Manos libres	'H'
Tecla de rellamada	'R'
Hacia arriba	'U'
Hacia abajo	'D'

FIG. 22

```

struct FuncKey {
    PALABRA Introducción;    tecla de introducción
    PALABRA Rellamada;      tecla de rellamada
    PALABRA Arriba;         tecla de flecha hacia arriba
    PALABRA Abajo;          tecla de flecha hacia abajo
    PALABRA Número;         teclas de número o tecla especial
    PALABRA Completo;       búfer de tecla lleno
    PALABRA Habilitar;      cuando se establece, indica que la entrada de tecla se habilita
    PALABRA Tocar;          se ha presionado cualquier tecla
    PALABRA Alf;            se ha presionado una tecla alfabética
};
    
```

FIG. 23

Dirección de puerto	Leer / Escribir	Valor de bit	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0x820001	W	0	x ⁵	x	x	x	Restablecimiento	Bloqueo de volumen	Liberación ENET	Manos libres
0x820001	W	1	x	x	x	x	x	Desbloqueo	Restablecer ENET	Auricular
0x820001	R	0	x	x	x	x	x	Sin entrada de tecla	X	Descolgado
0x820001	R	1	x	x	x	x	x	Tecla pulsada	X	Colgado

FIG. 24

```

struct Message {      /* una estructura para todos los mensaje en el teléfono SIP */
  int ENetXmtST;      /* estado de paquete de transmisión Ethernet */
  int ENetRcvST0;     /* estado de paquete de recepción Ethernet */
  int ENetRcvST1;     /* estado de paquete de recepción Ethernet */
  int ENetRcvST2;     /* estado de paquete de recepción Ethernet */
  int ENetRcvST3;     /* estado de paquete de recepción Ethernet */
  int RcvFlag;        /* los datos de voz de recepción están disponibles cuando están ESTABLECIDOS */
  int ARPST;         /* reservado */
};

```

FIG. 25

```

typedef struct {
  int pt:7           /* tipo de carga útil */
  int m:1           /* bit de marcador */
  int cc:4          /* cuenta CSRC */
  int x:1          /* indicador de extensión de encabezado */
  int p:1          /* indicador de almohadilla */
  int version:2     /* versión de protocolo */
  int seq          /* número de secuencia */
  int ts1         /* fecha menos significativa 16 bits */
  int ts2         /* fecha más significativa 16 bits */
  int ssrc1       /* fuente de sincronización menos significativa 16 bits */
  int ssrc2       /* fuente de sincronización más significativa 16 bits */
  int csrc[1]     /* dirección de lista CSRC opcional */
} RTPHeader;

```

FIG. 26

```
struct ToneType {  
    int ActiveTime;    El periodo para sonido está activo  
    int ActiveCnt;    El contador para el sonido durante el tiempo activo  
    int StopTime1;    Primer periodo de parada de sonido  
    int StopCnt1;    Primer contador de parada de sonido  
    int StopTime2;    Segundo periodo de parada para sonido  
    int StopCnt2;    Segundo contador de parada de sonido  
}
```

FIG. 27

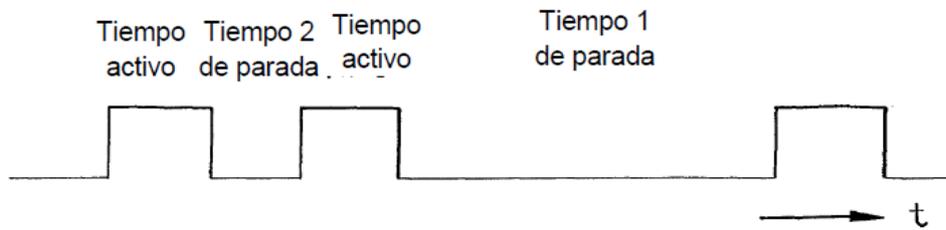


FIG. 28

ES 2 666 472 T3

```

typedef struct { /* tipo de secuencia usada en estructura de mensaje t */
    char *s; /* inicio de secuencia */
    short len; /* longitud de secuencia */
} string;

typedef enum { /* estructura de transición de estado */
    Initial, /* estado inicial SIP. UAC o UAS */
    Proceeding, /* procedimiento de la solicitud, UAS */
    Failure, /* fracaso. UAS */
    Success, /* éxito, UAS */
    Confirmed, /* confirmado, UAS */
    Calling, /* llamando. UAC */
    CallProc, /* procedimiento de llamada, UAC */
    Completed, /* completado, UAC */
    Bye /* estado Adiós, UAC o UAS */
} Tstate;

typedef struct { /* estructura de mensaje SIP */
    method_t method; /* solicitud: métodos; respuesta: 0 */
    short status; /* respuesta: valor de estatus; solicitud: 0 */
    string url; /* solicitud URL */
    string via; /* a través de encabezado */
    string callid; /* identificador de llamada */
    string contact; /* encabezado de contacto */
    string from; /* desde dirección */
    string from_display; /* desde nombre de pantalla */
    string subject; /* tema */
    string to; /* hacia dirección */
    string to_display; /* mostrar nombre */
    string ts; /* fecha */
    string reason; /* frase de razón de respuesta */
    content_t contenttype; /* encabezado de tipo de contacto */
    int contentlength; /* longitud de contacto */
    unsigned cseq; /* número de secuencia */
    string body; /* cuerpo de SDP */
    sdp_t sdp; /* descripción de sesión */
} message_t;

typedef struct { /* estructura de llamada */
    int flag; /* ESTABLECIDO para efectivo, RESTABLECIDO para claro */
    int ua_state; /* no llamada actual; 0; UAC:1; UAS:2 */
    int status; /* estatus de respuesta actual */
    message_t m; /* mensaje SIP */
    char * udp; /* puntero de paquetes SIP de recepción */
    char * local; /* puntero de paquete de solicitud UAC */
    sockaddr peer; /* dirección IP de host par */
    sdp_t sdp; /* copia sdp */
    Tstate state; /* estado de transición SIP */
    int t1; /* temporizador T1 */
    int t2; /* temporizador T2 */
} call;

```

FIG. 29

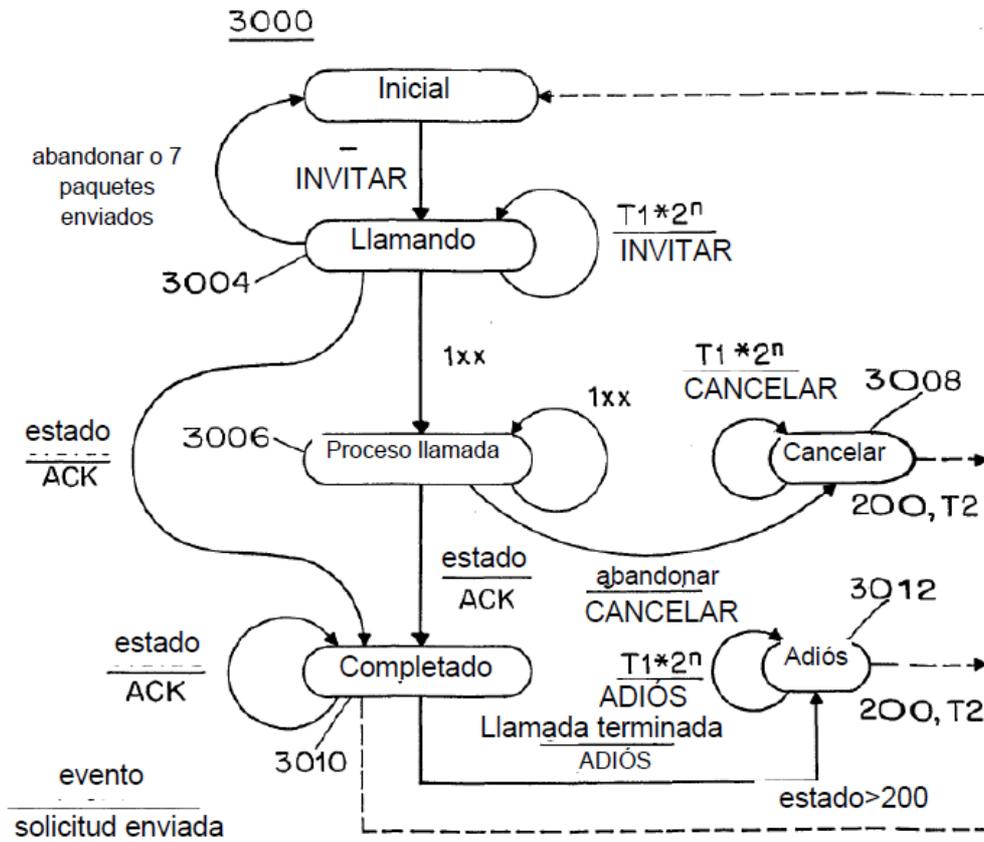


FIG. 30

Mensaje recibido	SIP_Status
100	SIP_Trying
18x	SIP_Ring
200	SIP_Connected
3xx	SIP_Redirect
4xx, 5xx	SIP_Refused
6xx	SIP_Busy

FIG. 31

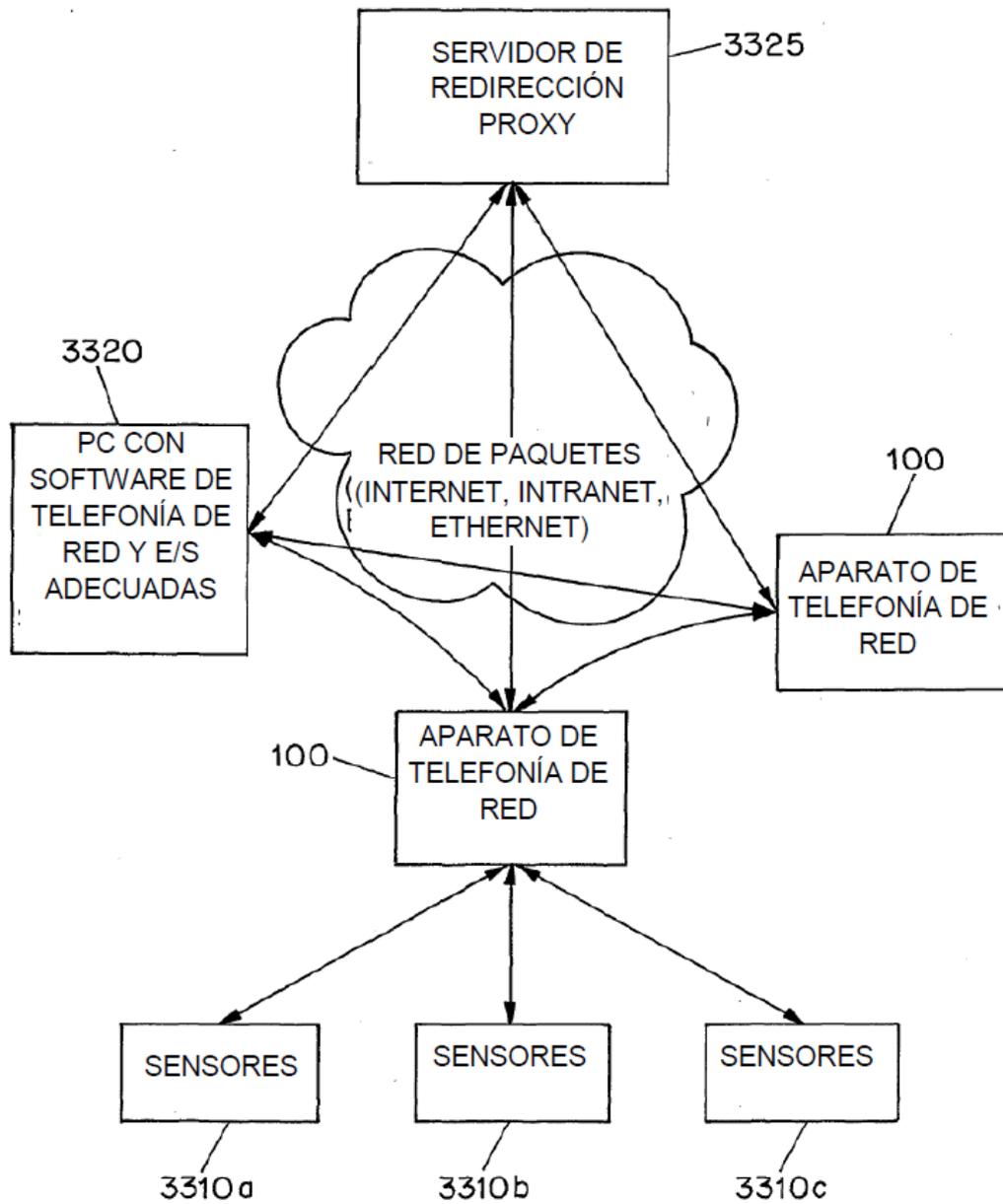


FIG. 33

FIG. 34

A INFRAESTRUCTURA
DE RED INALÁMBRICA

