Universidad Autónoma de Baja California

Facultad de Ciencias Químicas e Ingeniería

Método para la Transmisión y Recepción de Voz a través de Redes Inalámbricas Wi-Fi para un Sistema Embebido

TESIS

que para obtener el grador de

Maestro en Ciencias

Presenta:

José Macías Galarza

Director de Tesis

Dr. Leocundo Aguilar Noriega

Tijuana B.C. **Marzo 2016**

Universidad Autónoma de Baja California

FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA
COORDINACIÓN DE POSGRADO E INVESTIGACIÓN

FOLIO No. 171

Tijuana, B. C., a 10 de febrero de 2016

C. José Macías Galarza
Pasante de: Maestro en Ciencias
Presente

El tema de trabajo y/o tesis para su examen profesional, en la Opción TESIS

Es propuesto, por el C. Dr. Leocundo Aguilar Noriega

quien será el responsable de la calidad de trabajo que usted presente, referido al tema Método para la Transmisión y Recepción de Voz a través de Redes Inalámbricas Wi-Fi para un Sistema Embebido.

el cual deberá usted desarrollar, de acuerdo con el siguiente orden:

I.- INTRODUCCIÓN
II.- ANTECEDENTES Y MARCO TEORICO
III.- MÉTODO PROPUESTO
IV.- IMPLEMENTACIÓN
V.- PRUEBAS Y RESULTADOS
VI.- CONCLUSIONES Y TRABAJO FUTURO

Dr. José Luis González Vázquez
Sub-Director Secretario

Dr. Leocundo Aguilar Noriega
Director de Tesis

Dr. Luis Enrique Palafox Maestre
Director

Resumen

En la actualidad se ha estado incrementando la tendencia de las comunicaciones tecnológicas y el uso de los dispositivos inalámbricos Wi-Fi para transmisión y recepción de audio y voz en los sistemas de cómputo móviles incluyendo los sistemas inteligentes de telefonía. Estos dispositivos están basados en grandes sistemas de cómputo. Por lo cual los sistemas embebidos se han dejado a un lado debido limitaciones de recursos de cómputo que poseen y los protocolos VoIP sobre WiFi que no están adaptados para estos sistemas. Sin embargo esa tecnología pudiera ser aprovechada en dichos sistemas (limitados), adaptando los métodos de codificación, transmisión de una manera eficiente (adecuada).

En la actualidad existen una gran variedad de métodos para codificación, transmisión y recepción de audio en dispositivos que sus aplicaciones son sistemas de cómputo, pero ¿Cuál es el mejor método a utilizar en los sistemas embebidos WiFi para trasmitir y recibir voz con calidad? No existe algún método optimizado para implementar en los sistemas embebidos.

Este proyecto de tesis está enfocado en proponer un método para la trasmisión y recepción de voz en un sistema embebido WiFi, con el propósito de utilizar el protocolo de transmisión y recepción de voz; utilizando tres partes fundamentales para su planteamiento: la definición de lo que es codificación y protocolos de voz, exploración de los métodos que existen para la transmisión y recepción de VoIP y finalmente diseñar un nuevo método.

Índice general

Pag.

Resumen III

Índice General IV

Lista de Figuras VI

Capítulo

1 Introducción 4

1.1 Motivación 4

1.2 Planteamiento del problema 5

1.3 Propuesta de Solución 5

1.4 Objetivos 6

1.4.1 Objetivo General 6

1.4.2 Objetivos particulares 7

1.5 Metodología 7

1.6 Contenido de la Tesis 7

2 Antecedentes y Marco Teórico 9

2.1 Modelo OSI 16

2.1.1 Segmentos de Información 17

2.1.2 Capas del Modelo OSI (Open System Interconnection siglas en Ingles) 19

2.2 Arquitectura de Internet 25

2.2.1 Protocolos de la capa de transporte 26

2.2.2 Protocolo de Control de Transmisión (Transmission Control Protocol siglas en Ingles) 26

2.2.3 Protocolo de paquetes de usario (User Datagram Protocol siglas en Ingles) 27

2.3 Protocolo de iniciacion de session (Session Initiation Protocol) 28

2.3.1 SIP Entities 29
2.3.2 Operación del protocolo SIP 29
2.3.3 Respuestas de protocolo SIP 30
2.3.4 Las solicitudes de SIP 30
2.4 Transporte RTP 31
2.4.1 Paquetes de RTP 32
2.5 VoIP codificadores de Audio 34
2.5.1 Señales de Voz 36
2.5.2 Codificadores y Decodificadores de Audio 39
2.5.3 Estándares de Codificadores 40
3 Método Propuesto 43
3.1 Simulador 46
3.1.1 Herramienta de simulación 47
3.1.2 Componentes del Simulador 47
3.1.3 Canales 50
3.2 Pruebas y Resultados 50
3.2.1 Resultados de la Simulación 52
4 Implementación 58
4.1 Implementación en un sistema de computadoras portátiles 58
4.2 Implementación en el Sistema Embebido 63
5 Pruebas y Resultados 67
6 Conclusiones 72
6.1 Trabajo Futuro 73
Referencias 74
Apéndices
A 76

Lista de Figuras

2.1 Telefonía analógica, desde su invención hasta la actualidad se sigue utilizando 9
2.2 TDM Switching 10
2.3 Transmitiendo voz asíncrona 14
2.4 Transmitiendo de voz asíncrona 15
2.5 VoIP flujo del protocolo 16
2.6 Componentes básicos de un paquete de correo electrónico 18
2.7 Encapsulacion de datos que pasan a traves del modelo 21
2.8 Descripción de la dirección MAC 24
2.9 Las cuatro capas de Internet TCP/IP 25
2.10 Servicios de Usuario de Internet 26
2.11 Servicios de Usuario de Internet 31
2.12 Formato de los paquetes RTP 33
2.13 Rangos de escucha 35
2.14 Formato de los paquetes RTP 36
2.15 Forma de onda de una fracción de un segundo 37
2.16 Mensaje "Hola" 38
2.17 Mensaje "Hola.^{ex}pandido 38

3.1 Clasificación de los encabezados de RTP,UDP y IP 45
3.2 Propuesta del Protocolo 47
3.3 Red llamada WirelessHostWithAP 48
3.4 Red llamada WirelessHostWithAP 49
3.5 Modulo accessPoint con sus modulos compuestos 50
3.6 Transmisor y Receptor 52
3.7 Simulacion de la red cableada 52

3.8 Simulación de la red inalambrica 54
3.9 Resultados de la simulación 56
3.10 Resultados de la simulación 57

4.1 Diagrama a bloques del sistema de computadoras personales 59
4.2 Diagrama a bloques de la aplicación del transmisor 60
4.3 Diagrama a bloques de la aplicación del transmisor 61
4.4 Diagrama de flujo del transmisor 62
4.5 Diagrama de flujo del receptor 63
4.6 Diagrama a bloques de la computadora personal portátil y el sistema embebido 64
4.7 Diagrama a bloques del programa del sistema embebido 66

Abreviaturas

VoIP Voice over Internet Protocolo

WLAN Wireless local-area network

WiFi Wireless Internet

VoWiFi Voice over Wireless

SSID Service Set Identification

PC Personal Computer

SIP Simple Internet Protocol

RTP Real Time Protocol

CELP Celda

TDM Time Division Multiplexing

UC Unify Communication

IP Internet Protocol

PBX Private Branch Exchange

PSTN Public Switched Telephone Network

ATM Asynchronous Tranfer Mode

ISP Internet Service Provider

PDA Personal Digital Assistances

VAD Voice Active Detection

OSI Open System Informacion

FCS Frame Check Sequence

NIC Network Interface Card

HTTP Hypertext Transfer Protocol

FTP File Transfer Protocol

SMTP Simple Mail Transfer Protocol

POP3 Post Office Protocol 3

SQL Structured Query Language

IMAP Internet Message Access Protocol

ASCII American Standard Code for Information Interchange

RPC Remote Procedure Call

LDAP Light Directory Access Protocol

NetBIOS Network Basic Input/Output System

MAC Media Access Control

LAN Local Address Network

CRC Cyclic Redundancy Code

TCP Transmission Control Protocol

UDP User Datagram Protocol

SCTP Stream Control Transmission Protocol

SSRC Synchronization Source

CSRC Computer Security Resource Center

NTP Network Time Protocol

WAN Wide Area Network

ACELP Adaptive Code Excited Linear Prediction

WAV Windows Wave (audio format/file extension)

SoC System on Chip

Capítulo 1

Introducción

1.1 Motivación

En la última década una de las tecnologías que ha estado creciendo significativas, son las redes de acceso inalámbrico. Antes de estas redes inalámbricas, las redes disponibles al consumidor eran las utilizadas en la telefonía celular, pero eran de muy baja velocidad lo cual soportaban solamente datos [1]. La transmisión de voz sobre el protocolo de Internet o comúnmente llamada VoIP (siglas en Ingles Voice over Internet) a través de una red inalámbrica (WLAN) se está convirtiendo en una necesidad; sin embargo hay un problema muy importante aquí, es el hecho de que los punto de acceso utilizados no son ideal para establecer una comunicación eficiente bajo esta infraestructura, ya que fueron diseñadas inicialmente para el transporte de datos. [2]

Voz sobre Protocolo de Internet (VoIP) es una tecnologías de transmisión y recepción de voz a través de redes de conmutación de paquetes, como lo es internet u otras redes que manejen el protocolo IP (Siglas en Ingles Internet Protocolo). Sistemas de red inalámbricos (WiFi) en conformidad con el estándar IEEE 802.11 están creciendo cada año, sobre la base de bajo costo y sencilla estructura. Estos sistemas se han desplegado como una alternativa para dar los servicios de Internet en los países en desarrollo, en esencia, regiones rurales han sido más atraídos por el despliegue de éstas tipo de sistemas en los que las compañías no tienen el interés de satisfacer estos servicios. Por estas razones, la determinación de si varios servicios de Internet, tales como llamadas telefónicas se pueden ejecutar sobre ellos Sistemas WiFi, sería necesario y también atractivo para el futuro investigación, especialmente dedicado a mejorar la capacidad de Wi-Fi sistemas [4].

1.2 Planteamiento del problema

Se han propuesto diversas extensiones al estándar IEEE 802.11 [5] con referencia al protocolo de voz sobre Internet (VoIP; del inglés Voice over Internet Protocol) para mejorar la comunicación en tiempo real en sistemas embebidos, tanto en la calidad de señal, pérdida de paquetes y el ancho de banda requerido para mantener al menos una conexión de voz mediante VoIP. Estas propuestas han sido solamente simuladas, y los resultados demuestran que no es viable implementarlo en sistemas embebidos con recursos limitados de cómputo [1]. El consumo de potencia es de los mayores problemas que enfrenta VoIP en sistemas embebidos, ya que el actual protocolo de Internet (IP; del inglés Internet Protocol) no ha implementado un estándar estratégico para ajustar los intervalos de dormir al sistema durante el envío de paquetes de voz [2]. Teniendo en consideración estas limitaciones, no se ha implementado VoIP en sistemas embebidos debido a las limitaciones de recursos que posee, tanto de software (memoria) como de hardware (consumo de energía).

Este proyecto de tesis está enfocado a la propuesta de un método de recepción/ transmisión de voz en un sistema embebido a través del protocolo de Internet inalámbrico (VoWiFi del inglés VoIP basado en Wi-Fi) [5], para sistemas embebido con recursos limitados de hardware y software. Este método contempla la inclusión de un algoritmo inteligente, con el propósito de mejorar el rendimiento de la codificación, calidad de audio.

1.3 Propuesta de Solución

En las últimas décadas se han venido expandiendo las redes locales basadas en el estándar 802.11x, conocidas como redes WiFi. Inicialmente este tipo de red no fue contemplada para comunicación de voz en tiempo real, pero debido a su bajo costo y gran disponibilidad se han venido utilizando como medio de comunicación complementario a la telefonía celular. Actualmente VoIP está siendo soportado por dispositivos móviles inteligentes con capacidad de conectarse a redes WiFi, sin embargo estos dispositivos son de alto costo y no cumplen

con las características para un sistema ubicuo inteligente, donde exista una integración de los sistemas tecnológicos dentro de tal manera que ayude al humano de manera natural en sus actividades diarias [7]. Los gestos, el habla, utensilios para escritura, y la comunicación del humano son las acciones naturales humanas, lo cual los sistemas ubicuos apoyan a estas formas.

Los dispositivos móviles de voz telefónica por el hecho de utilizar un proveedor de servicio lo convierte en un dispositivo caro [3]. La combinación de voz sobre protocolo de Internet (conocido como VoWiFi) ha generado bastante interés [1] en los últimos años. Los mayores beneficios del uso de esta tecnología han sido las empresas corporativas, instituciones educativas [4], hospitales y en los hogares, donde ya se encuentra instalada y en uso una red local WiFi.

El notable desarrollo de la tecnología de VoIP y las aplicaciones de comunicación de voz en tiempo real, se han expandido a través del uso de Internet. Sin embargo, los sistemas utilizados tienen una excesiva dependencia a gran cantidad de recursos donde se utilizan dichos programas como los son las PC o teléfonos inteligentes. El modelo que se presenta en este documento es básicamente proponer un modelo de que soporte la comunicación de VOIP sin depender de un sistema de grandes recursos de computación. Utilizando como base para su análisis los protocolos de SIP y RTP como protocolo de transmisión y emplea CELP algoritmos de compresión para asegurar la baja latencia y alta calidad la comunicación.

1.4 Objetivos

1.4.1 Objetivo General

Plantear un modelo para transmisión y recepción de voz atreves de WiFi en un sistema de recursos limitados de cómputo

1.4.2 Objetivos particulares

- Estudiar los componentes de un sistema de VoIP
- Investigar los métodos de codificación que se utilizan para transmisión/recepción de VoIP
- Simular un sistema VoIP
- Proponer el modelo

1.5 Metodología

La metodología utilizada para la realización de este trabajo, esta compuesta en la siguiente serie de etapas que a continuación se describe:

- Análisis de los protocolos de VoIP. El objetivo de esta etapa es entender cuales son los protocolos que se utilizan para la trasnmicion y recepción de VoIP.
- Etapa de desarrollo. En esta etapa el objetivo es proponer un protocolo que sea eficiente respecto a los protocolos actuales de tal manera que pueda coexista y soportarlo en un sistema embebido de recursos limitados de computo.
- Para la etapa de validación de la propuesta se utilizara un simulador de computadora para realizar la evaluación respectiva.

1.6 Contenido de la Tesis

El planteamiento de los objetivos anteriormente presentados,han resultado en los siguientes capítulos.

- En capítulo I se describen y se definen las bases para el desarrollo y sustentación del trabajo de tesis. El capítulo se encuentra organizado en los siguientes subtemas:

- Capitulo II se describe el marco teórico, se abordan los protocolos y el estado del arte de VoIP.
- Capitulo III describe el método propuesto
- Capitulo IV detalla la implementación del método propuesto, incluyendo la descripción de los sistemas que se utilizaron
- Capitulo V presenta el análisis y reporte de los resultados obtenidos con el metodo propuesto para transmitir y recibir VoIP
- Capitulo IV conclusiones obtenidas con el estudio, análisis e implementación de la metodología propuesta, ademas de los trabajos futuros.

Capítulo 2

Antecedentes y Marco Teórico

El sistema telefónico más comúnmente utilizado en la actualidad es el sistema analógico. La telefonía analógica (Figura 2.1) utiliza la modulación de señales eléctricas a lo largo de un alambre para el transporte de voz.

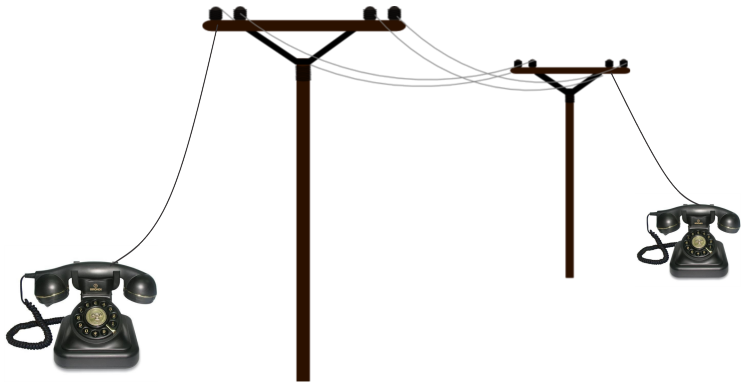


Figura 2.1 Telefonía analógica, desde su invención hasta la actualidad se sigue utilizando

Aunque es una tecnología bastante antigua, la transmisión analógica tiene muchas ventajas: es sencilla, el retardo es muy bajo desde que se trasmite la voz hasta que llega al receptor , porque la señal se propaga a lo largo del hilo, la velocidad con que se trasmite es casi similar a la de la luz. Relativamente el costo es barato cuando hay pocos usuarios hablando al mismo tiempo y cuando la distancia es corta. Pero la tecnología analógica básica requiere un par de hilos para tener una conversación activa, esto causa que se convierta poco práctico y caro. La primera mejora a la tecnología analógica básicamente involucra multiplexar varias conversaciones sobre el mismo cable, utilizando una frecuencia de transporte independiente para cada señal. Pero incluso con esta mejoría, la telefonía analógica tiene muchos inconvenientes:

- Al menos que se utilice un tablero analógico, la cual este tipo de conmutadores requieren de muchos mecanismos electromecánico, lo que los convierte en equipos caros para adquirirlos y un alto costo de mantenimiento.

- ruidos parásitos se suma en todas las etapas de la transmisión, porque no hay manera de diferenciar la señal principal del ruido y no se puede limpiar. Por todas estas razones, la mayoría de los países hoy en día utilizan la tecnología digital para su telefonía básica.

En la mayoría de los casos, los clientes mantienen las líneas analógicas, pero esta senal se convierta en un flujo de datos digitales en el primera central telefónica local. Por lo general, esta señal tiene una tasa de bits de 64 kbit por segundo o 56 kbit por segundo (una muestra cada 125 microsegundos). Con esta tecnología digital, muchos canales de voz fácilmente se pueden multiplexar a lo largo de la misma línea de transmisión usando una tecnología llamada multiplexación por división de tiempo (TDM Time Division Multiplexing siglas en Ingles). En esta tecnología, el flujo de datos digitales que representa una sola conversación se divide en bloques (por lo general un octeto) y bloques de varias conversaciones están intercaladas en turnos rotativos en los intervalos de tiempo de la línea de transmisión, como se muestra en la figura 2.2

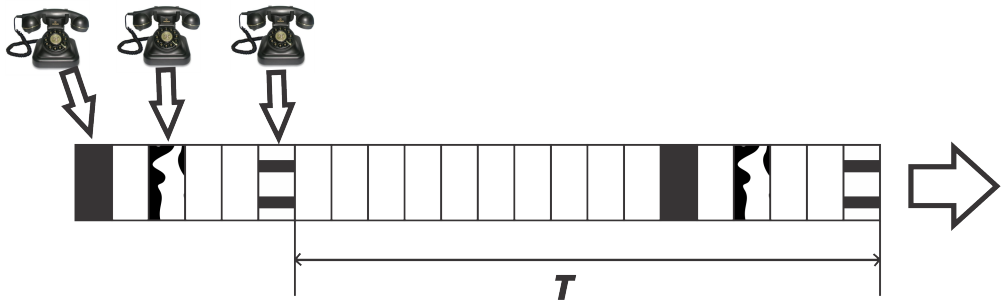


Figura 2.2 TDM Switching

Debido a la tecnología digital, el ruido que se añade en la cadena principal no influye en la calidad de la comunicación, porque la información digital se puede reconstruir exactamente como la original, incluso en la presencia de ruido significativo. Por otra parte, TDM digital

hace posible la conmutación digital. Por lo tanto, esta conmutación puede ser realizada por los ordenadores.

Los paquetes de voz sobre IP y Comunicaciones Unificadas (UC) de tecnologías surgen después de 1980, cuando la compañía Bell introdujo la transmisión digital y empezó el cambio de remplazar la infraestructura del operador conjuntamente con los métodos analógicos. Como la tecnología digital se extendido en las empresas a través de la década de los 1980, una ola de telefonía privada empresarial PBX (Private Branch Exchang siglas en Ingles) digitales empezó a remplazar la mayoría de los sistemas analógicas. Hoy en día el único sistema analógico existente es la red de telefonía pública conmutada (PSTN: Public Switched Telephone Network siglas en Ingles)

Desde 1998 voz sobre IP, definitivamente ha sido la palabra de moda favorita en la industria de las telecomunicaciones. Aunque el protocolo de internet (Internet Protocol siglas en inglés) aún no estaba muy establecida y dominante como lo es al día de hoy, la mayoría de los ingenieros de telecomunicaciones todavía creían que sólo la tecnología ATM (Asynchronous Transfer Mode siglas en Ingles) sería capaz de soportar aplicaciones multimedia. De hecho hubo al inicio de la internet los servicio eran a través de módems dial-up y la mayoría de los proveedores de servicio de Internet (ISP Internet Service Provider siglas en Ingles), estos sistemas eran incapaces de mantener eficientemente las conexiones y servicios de Internet en un buen funcionamiento, con ese infraestructura difícilmente se podría soportart voz sobre IP y servicios de multimedia.

Inicialmente VoIP se presentó como una tecnología que podría permitir a un proveedor de internet el servicios el transporte de voz 'gratis' sobre el protocolo de Internet, porque el transporte IP era "libre", y las llamadas podría ser enviado dentro de un grupo. Las primeras aplicaciones comerciales de VoIP se centraron en la telefonía de prepago, que era un objetivo razonable teniendo en cuenta que los compradores potenciales de los sistemas de tarjetas de prepago se preocupan sobre los costos.

VoIP, es el acrónimo de voz sobre el protocolo de Internet, básicamente significa voz transmitida a través de una red digital. Bueno, eso no es técnicamente preciso debido a

que la Internet no es estrictamente necesario para VoIP, aunque fue como surgió en esos momentos.

Lo que es necesario para la tecnología VoIP es el uso de los protocolos que se utilizan para navegar en la red de Internet. (Un protocolo es un conjunto de reglas que se utilizan para permitir la comunicación ordenada.)

Por lo tanto, la voz sobre protocolo de Internet significa que la voz viaja a través de los protocolos utilizados en Internet.

Servicios de telefonía VoIP son interoperables, lo que significa que funcionan sobre todos los tipos de redes. También son altamente portables, lo que significa que van a trabajar con cualquier dispositivo que tenga las capacidades de IP tales como un teléfono IP, un ordenador, o incluso un sistema inteligente movil o asistente digital personal (Personal Digital Assistance siglas en Ingles)

El sistema de conmutación central de un proveedor de servicios VoIP no se procesa el flujo de voz, sino solamente se procesan los mensajes de señalización: por ejemplo una llamada iniciada desde Tijuana a New York por un controlador de llamada VoIP ubicado en Japón. Sólo los mensajes de señalización hacen de ida y vuelta a través del Atlántico, pero los paquetes de voz sólo se cruzan el canal de comunicación.

Ahora está claro que las razones claves para el éxito de la VoIP son:

- Independencia de la Ubicación .Debido a las características únicas de los controladores de llamadas VoIP, muchas de las funciones que antes requerían múltiples puntos de distribución ahora puede ser centralizada, reduciendo los gastos generales administrativos y acelerar los despliegues.

- La simplificación del transporte en las redes:

- La capacidad de establecer y controlar las comunicaciones multimedia, por ejemplo, audio interactivo y llamado de vídeo, sesiones de intercambio de datos, etc.

- Telefonía residencial, nuevos proveedores de servicios pueden desplegar los centros de

control de los servidores para llamadas VoIP y utilizar cualquier tecnología de redes IP.

- En general, todas las aplicaciones que requieren una red inteligente con una arquitectura compleja con el fin de minimizar la conmutación, puede ser significativamente simplificado utilizando un servidor de IP que controla las llamadas y señalización de voz, a través de la optimización de la ruta de voz .

En condiciones normales de una llamada, las de personas que se están comunicando en general lo normal es hablar sólo la mitad del tiempo durante una conversación. Dado que todos tenemos que pensar un poco antes de responder, cada parte generalmente habla sólo el 35

Si pudieras apretar un botón cada vez que hable, entonces podrías enviar datos a través de la línea del teléfono sólo cuando realmente quieres decir algo, no cuando están en silencio. De hecho, la mayor parte de las técnicas utilizadas para transformar su voz en datos (conocidos como codificadores) tienen la capacidad para detectar cuando hay silencio. Con esta técnica, conocida como detección de actividad de voz (Voice Active Detection siglas en ingles), en lugar de transmitir un pedazo de datos, voz, o el silencio cada 125 milisegundos, como se hace hoy en Redes TDM, sólo se transmiten datos cuando se necesita, de forma asíncrona, como se ilustra en la Figura 2.3

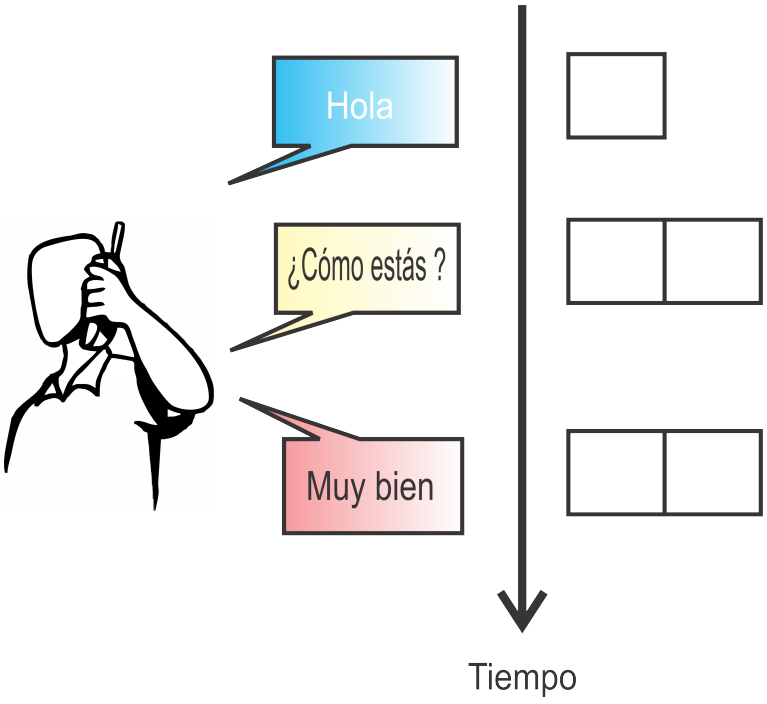


Figura 2.3 Transmitiendo voz asíncrona

Y cuando se trata de multiplexar varias conversaciones en una sola línea de transmisión, en lugar de ocupar una fracción del ancho de banda, entonces el ancho de banda se puede utilizar por otra persona mientras está en silencio. Esto se conoce como multiplexación estadística. La principal ventaja de la multiplexación estadística es que permite que el ancho de banda puede ser usada mas eficientemente, sobre todo cuando hay muchas conversaciones multiplexados en el misma línea.

Multiplexación estadística, como su nombre indica, introduce incertidumbre en la red. Como acabamos de mencionar, en el caso de TDM un retraso de hasta 125 milisegundos podría ser introducido en cada cambio; este retraso es constante a lo largo de la conversación. La situación es totalmente diferente con multiplexación estadística (Figura 2.4),

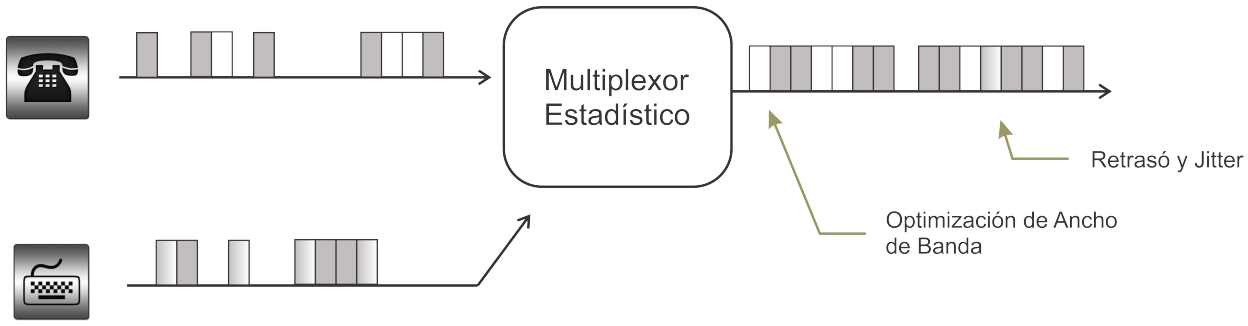


Figura 2.4 Transmitiendo de voz asíncrona

si la línea de transmisión está vacía cuando se necesita enviar un fragmento de datos, esta pasaría de inmediato. Si por otro lado la línea está ocupada o llena, es posible que tenga que esperar hasta que haya espacio para enviarla. Este retraso es variable dependiendo de la fluctuación de llamadas, y necesita ser corregido por el lado receptor. De lo contrario, si los fragmentos de datos se reproducen tan pronto como se reciben, el discurso original puede llegar a ser escuchado correctamente como el mensaje original. Las redes de telefonía de nueva generación utilizarán multiplexación estadística, para mezclar voz y los datos a lo largo de las mismas líneas de transmisión.

Creemos que voz sobre IP es una solución muy flexible, ya que no requiere de la creación de canales virtuales entre los sitios que se desean comunicarse. La escala de redes de VoIP es mucho mejor que las redes ATM (Asynchronous Transfer Mode siglas en Ingles) . VoIP también permite la comunicación directa entre los puntos finales.

Es sorprendente las características nuevas y emocionantes que tiene VoIP. Esto es por el modelo de datos OSI (Open System Information siglas en Ingles)en el que se está apoyando que tiene más de veinticinco años de su creación.

Existen varios protocolos de VoIP , pero sólo entran dos categorías: la señalización y el transporte. Los protocolos de señalización manejan las funciones derivadas de la arquitectura del sistema de teléfono, y los protocolos de transporte llevan los paquetes de voz generados desde el codificador de audio. Móviles utilizan el protocolo de señalización para registrarse en el servidor de llamadas, configurar y terminar las llamadas. Protocolos de señalización

también se utilizan para funciones como directorio servicios y pantallas. Los paquetes de datos de voz una vez que una llamada se ha establecido normalmente se envían directamente entre los teléfonos usando encapsulación de RTP . Las trayectorias de flujo se muestran en la Figura 2.5.

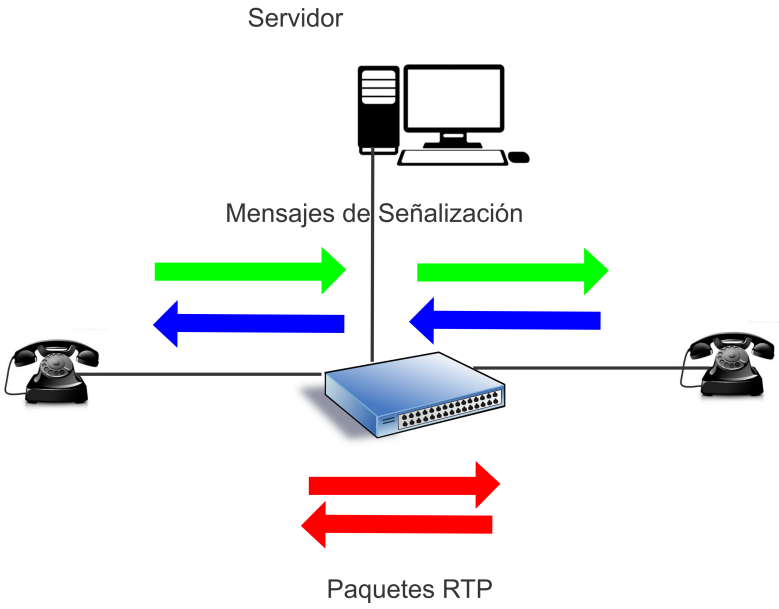


Figura 2.5 VoIP flujo del protocolo

2.1 Modelo OSI

Para explicar los protocolos de VoIP en la cual nos enfocaremos, nos basaremos a el siguiente modelo. El modelo OSI (Open Systems Interconnection),que originalmente fue diseñado para permitir que proveedores independientes participaran en los de protocolos, y eliminar protocolos monolíticos, evitar que el emisor tenga el control en una sola capa de todo el proceso de transferencia. En la actualidad tiene un uso muy importante ya que es una de las mejores herramientas disponibles para describir interacciones tan compleja que se tiene en las redes. Dado que la mayoría de los protocolo que se utilizan ahora (como TCP / IP), su diseño están basado es este un modelo, los protocolos que no estan basados en este

modelo, provocan una gran confusión. Por ejemplo, algunos libros afirman que Protocolo de información de enrutamiento (RIP) reside en la capa de red, mientras que otros afirman que reside en la capa de aplicación. La verdad es que no reside únicamente en una sola capa.

2.1.1 Segmentos de Información

Los términos paquete, datagrama, marco, mensaje, y segmento todos ellos tienen esencialmente el mismo significado, la diferencia existe dependiendo de la capa del modelo OSI en donde reside.. Como ejemplo podemos pensar en un paquete que se envía a través del correo tradicional. Para enviar esta pieza de correo postal, se necesita un número de componentes que a continuación se mencionan:

Carga útil . Este componente es la carta que se está enviando, por ejemplo, una foto de su hijo recién nacido a un ser querido

Dirección de origen .Este componente es la dirección de retorno del correo. Esto indica que el mensaje proviene del originador, por si acaso hay un problema de la entrega de la carta

Dirección de destino . Este componente es la dirección de la persona a la que se le envía, por lo que la carta es entregada a la persona correcta

Sistema de verificación . Este componente es el sello. Esto confirma que a pasado por los canales apropiados lo que hace que la carta sea válida para que sea entregada por el servicio postal que corresponda.

El envió de un paquete en el modelo OSI,es realmente similar al proceso de enviar una carta. Entonces ahora haremos el ejercicio de enviar un mensaje por medio del correo electrónico (ver Figura 2.6. Utilizando la misma información del ejemplo anterior.

Carga útil Este componente son los datos que se están enviando, por ejemplo, un correo electrónico a un ser querido

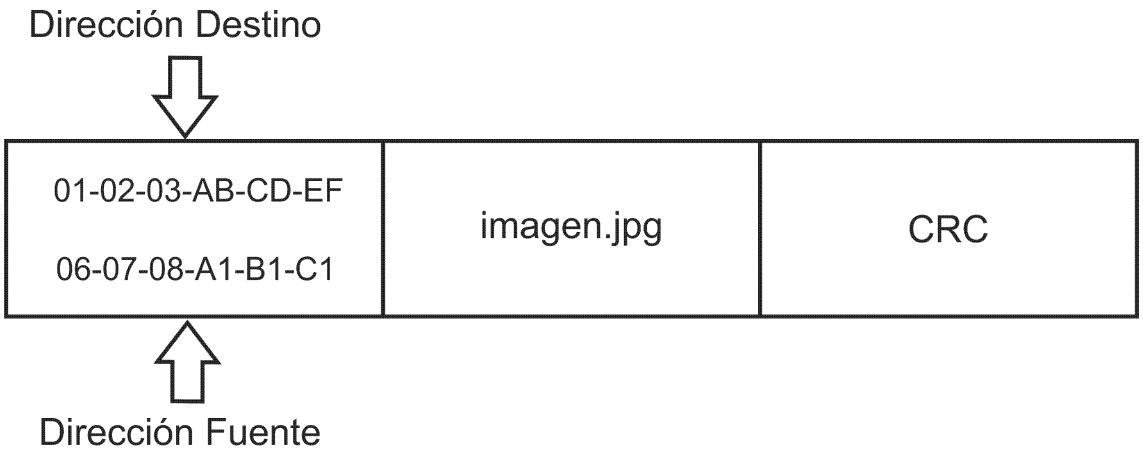


Figura 2.6 Componentes básicos de un paquete de correo electrónico

Dirección de origen Este componente es la dirección del remitente de tu correo electrónico. Indicando que el mensaje proviene de ti, por si acaso hay algún problema la entrega a la dirección de correo

Dirección de destino Este componente es la dirección del correo electrónico para tu ser querido, por lo que el correo electrónico se puede entregar correctamente

Sistema de verificación En el contexto de un paquete, este componente de cierta manera verifica y comprueba los errores del sistema.

En este caso, vamos a utilizar la secuencia de verificación de la trama (FCS, Frame Check Sequence siglas en Ingles). El FCS es poco más que una fórmula matemática que describe la composición de un paquete. Si el calculo del FCS es correcto en el punto final (al llegar a tu ser querido), entonces los datos son válidos y serán aceptados. Si entonces el resultado del calculo computado no es correcto, el mensaje se descarta. Las siguientes secciones utilizan el concepto de un paquete descrito en el ejemplo anterior, para ilustrar los recorridos de los datos realizados por el modelo OSI, a través del cable, y la copia de seguridad del modelo OSI para llegar como un nuevo mensaje en la bandeja de entrada del ser querido que va recibir el mensaje.

2.1.2 Capas del Modelo OSI (Open System Interconnection siglas en Ingles)

El modelo OSI es un enfoque por capas para la creación de redes. Aunque algunas de las capas no necesariamente se incluyan a momento de implementar el protocolo deseado, pero el modelo OSI está dividido de modo que cualquier función dentro de la red puede ser representada por alguna de las 7 capas. Tabla 2-1 describe las capas, comenzando con la capa 7 y terminando con la capa 1, se explica en este orden porque en la mayoría de los casos, la gente tiende a entender el modelo mejor si se introduce en este orden.

Tabla 2.1 Capas del Modelo OSI

Modelo OSI	
*N*ombre de las Capas	*F*uncion
Aplicación (Capa 7)	Esta capa es responsable de comunicarse directamente con la aplicación. Esta capa permite que una aplicación sea desarrollada con muy poco código de red. En cambio, la aplicación de usuario le dice el protocolo de la capa de aplicación lo que necesita, y es responsabilidad de la capa de aplicación de traducir esta petición en algo el conjunto de protocolos puedo entender
Presentación (capa 6)	Esta capa es responsable de todo lo involucrado con el formato de un paquete: compresión, cifrado, decodificación, y asignación de caracteres. Si recibe un correo electrónico, por ejemplo, y el texto es un galimatías, usted tiene un problema de presentación de capa
Sección (Capa 5)	Esta capa es responsable de establecer conexiones, o sesiones, entre dos puntos finales (generalmente aplicaciones). Se asegura de que la aplicación en el otro extremo tiene los parámetros correctos configurados para establecer la comunicación bidireccional con la aplicación de origen
Transporte (Capa 4)	Esta capa proporciona una comunicación entre un programa de aplicación y otra. Dependiendo del protocolo, puede ser responsable de la detección de errores y la recuperación, establecimiento de la sesión la capa de transporte y la terminación, multiplexación, fragmentación y control de flujo
Red (Capa 3)	Esta capa es la principal responsable de direccionamiento lógica y determinación de ruta,o el encaminamiento, entre lógica agrupaciones de direcciones
Enlace de Datos (Capa 2)	Esta capa es responsable del direccionamiento físico y tarjeta de interfaz de la red de control (NIC). Depende de protocolo, esta capa puede realizar el control de flujo también. Esta capa también añade las FCS, dándole cierta capacidad para detectar errores.
Física (Capa 1)	Esta es la más simple de todas las capas, esta capa solamente se ocupa de características físicas de una conexión de red: cableado, conectores, y cualquier otra cosa puramente física. Esta capa También es responsable de la conversión de bits y bytes (Eléctrico (de 1 y 0 de) a una representación física impulsos, ondas o señales ópticas) y de nuevo a los bits de el lado de recepción.

Cuando los datos se envía de una computadora a otra en una red, que solicita un requerimiento; este viaja hacia abajo a través del modelo; atravesando los medios de comunicación

(generalmente cables de cobre) como señal eléctrica u óptica, en representación individual de de 0 y 1; y luego a llega al otro modelo en el otro de la red. Mientras esto sucede, cada capa que tiene un protocolo que le va añadiendo una cabecera al paquete, que identifica cómo se debe procesar ese protocolo específico el paquete desde un extremo al otro . Este proceso se denomina encapsulación. Vea la Figura 2.7 para un diagrama (nota que AH significa encabezado de la aplicación, PH significa presentación cabecera, y así sucesivamente). Al llegar al destino, el paquete se transmitirá una copia de seguridad el modelo, con las cabeceras de protocolo que se retira a lo largo del camino. En el momento en el paquete llega a la aplicación, lo único que queda son los datos, o la carga útil. Ahora vamos a profundizar en los detalles de cada capa y los procesos adicionales para la que cada capa es responsable.

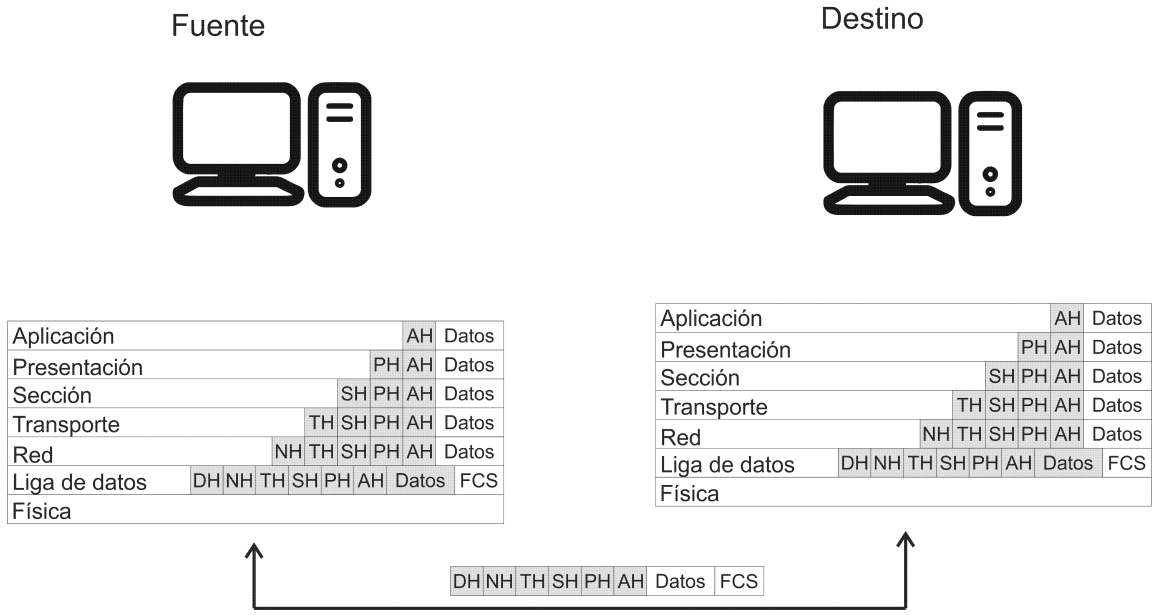


Figura 2.7 Encapsulacion de datos que pasan a traves del modelo

2.1.2.1 Capa de Aplicación

La capa de aplicación es responsable de interactuar con la aplicación real del usuario. Nota que no es (generalmente) la aplicación de usuario en sí, sino, más bien, las aplicaciones de red que son utilizadas por la aplicación de usuario. Por ejemplo, en la navegación web,

la aplicación de usuario es su software del navegador, como Microsoft Internet Explorer. Sin embargo, la aplicación de red siendo utilizado en este caso es http, que también es utilizado por un número de otras aplicaciones de usuario (tales como Chrome Navigator). Por lo general,la capa de aplicación es responsable de la creación inicial de paquetes. Si bien esto no es siempre el caso (algunos protocolos que existen en otras capas crean sus propios paquetes). Algunos protocolos de la capa de aplicaciones comunes son HTTP, FTP, Telnet, TFTP, SMTP, POP3, SQL, e IMAP.

2.1.2.2 Capa de Presentación

La capa de presentación es una de las capas más fáciles de entender, porque usted puede fácilmente ver sus efectos. La capa de presentación modifica el formato de los datos. Por ejemplo, yo podría enviarle un mensaje de correo electrónico que incluye una imagen adjunta. Transporte Simple Mail Protocolo (SMTP) no puede soportar nada más allá de texto sin formato (caracteres ASCII de 7 bits). Para soportar el uso de esa imagen, la aplicación necesita un protocolo para convertir la imagen a texto.

2.1.2.3 Capa de Sesión

Por el contrario, la capa de sesión es una de las capas más difícil de entender. Es responsable de establecer, mantener y terminar sesiones. Esto es un poco mas amplia y ambigua descripción, sin embargo, debido a que varias capas realizan alguna operaciones de establecer, mantener y terminar sesiones en cierta manera. La mejor manera de pensar en la capa de sesión es que se realiza esta función entre dos aplicaciones.Algunos capas comunes de secciones son los de protocolos RPC, LDAP y servicio de sesión NetBIOS.

2.1.2.4 Capa de Transporte

La capa de transporte realiza una serie de funciones, de la más importante de las cuales son la comprobación de errores, recuperación de errores y control de flujo. La capa de transporte es responsable para los servicios de transporte de datos entre redes confiables que

sean transparentes a la capa superior programas. El primer paso en la comprensión es la comprobación de errores de la capa de transporte.

2.1.2.5 Capa de Red

Las ofertas de la capa de red con direccionamiento lógico y determinación de ruta. Mientras que los métodos utilizados para el direccionamiento lógico varían con el conjunto de protocolos utilizado. El direccionamiento de la capa de red se utilizan principalmente para la localización geográficamente del dispositivo. Esta tarea se realiza generalmente dividiendo la dirección en dos partes: el campo de grupo y el campo de host. Estos campos juntos describen qué cual host eres, pero dentro del contexto del grupo que se encuentra. Esta división permite que cada host que se encarge de su grupo; y la división permite dispositivos especializados, llamados enrutadores, para hacer frente a conseguir los paquetes de un grupo a otro.

2.1.2.6 La capa de enlace de datos

La capa de enlace de datos trata con el arbitraje, direccionamiento físico, la detección de errores, y encuadre, tal como se describe en las siguientes secciones. Arbitraje significa simplemente determinar la forma de negociar el acceso a un único canal de datos cuando varios hosts están tratando de utilizar al mismo tiempo. En banda base transmisiones de half-duplex , el arbitraje es requiere porque sólo un dispositivo puede enviar en forma activa una señal eléctrica a la vez. Si dos dispositivos intentan acceder al medio al mismo instante, y luego las señales de cada dispositivo interferirán, causando una colisión.

Todos los dispositivos deben tener una dirección física. En tecnologías de red local (Local Area Network sigla en Ingles), esta es normalmente una Dirección de control de acceso al medio (Media Access Control siglas en Ingles). La dirección física está diseñado para identificar de forma exclusiva el dispositivo a nivel mundial. Una dirección MAC (también conocido como una dirección Ethernet, la dirección LAN, dirección física, dirección de hardware, y muchos otros nombres) es una dirección de 48 bits generalmente escrita como 12 dígitos he-

xadecimales, como 01-02-03-AB-CD-EF. Los primeros seis dígitos hexadecimales identificar el fabricante del dispositivo, y los últimos seis representan el dispositivo individual a partir de ese fabricante. Figura 2.8 presenta el desglose de la dirección MAC. Estas direcciones fueron históricamente "grabadas"haciéndolos permanentes. Sin embargo, en muy raros casos, una dirección MAC pudiera duplicarse. Por lo tanto, un gran número de dispositivos de red de hoy tienen direcciones MAC configurables.

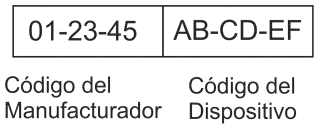


Figura 2.8 Descripción de la dirección MAC

Otra de las funciones de enlace de datos de capa, detección de errores, determina si problemas con un paquete se introdujeron durante la transmisión. Lo hace mediante la introducción de un remolque, el FCS, antes de enviar el paquete a la máquina remota. Este FCS utiliza una cíclico de comprobación redundante (CRC) para generar un valor matemático y colocar este valor en el paquete. Cuando el paquete llega a su destino, el FCS se examina y se aplica el inverso del algoritmo original que creó el FCS. Si la trama fue modificado de alguna ninguna manera, y no se obtiene igual al original, entonces se descarta ese paquete.

2.1.2.7 Capa Fisica

La capa física es responsable de una de las partes sustancial de todas las funciones. Todos los conectores, cableado, especificaciones de frecuencia, las distancias, los requisitos de la propagación de retardo, voltages, todas las cosas físicas-residen en la capa física. Algunos protocolos de capa física comunes son EIA / TIA 568A y B, RS 232, 10BaseT, 10Base2, 10Base5, 100BaseT, y USB.

2.2 Arquitectura de Internet

Como lo hemos comentado anteriormente las fortalezas más importantes de Internet es sus competencias para ser utilizado como una plataforma de creación de servicios. El modelo de Internet consiste en cuatro capas que se encuentran en la parte superior de la capa física (Figura 2.9),

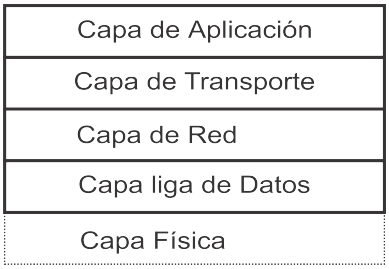


Figura 2.9 Las cuatro capas de Internet TCP/IP

Comúnmente se conoce como el protocolo de control de transmisión / Protocolo de Internet (TCP Transmission Control Protocol / Internet Protocol siglas en Ingles). La capa de transporte esta implementada en la parte superior de la capa de IP y el protocolos de la capa de aplicación hacen uso de los diferentes protocolos de transporte. El objetivo de la aplicación es recoger los servicios necesarios para implementar los servicio requerido por el cliente. Digamos que queremos construir un navegador Web que permite al usuario navegar por Internet y leer un correo electrónico. Para ello, tendremos que hacer uso de dos protocolos de la capa aplicaciones. Primeramente elegimos el protocolo de transferencia de hipertexto (Hypertext Transfer Protocol siglas en Ingles) [RFC 2068] para navegar por Internet y accesar los mensajes de Internet (Internet Message Access Protocol siglas en Ingles) [RFC 2060] para descargar el correo electrónico de un servidor. Ambos se ejecutan en la parte superior de la capa del protocolo de control de Transmisión (Tranmission Control Protocol siglas en Ingles) [RFC 793], que esta en la capa del protocolo de transporte (Figura 2.10).

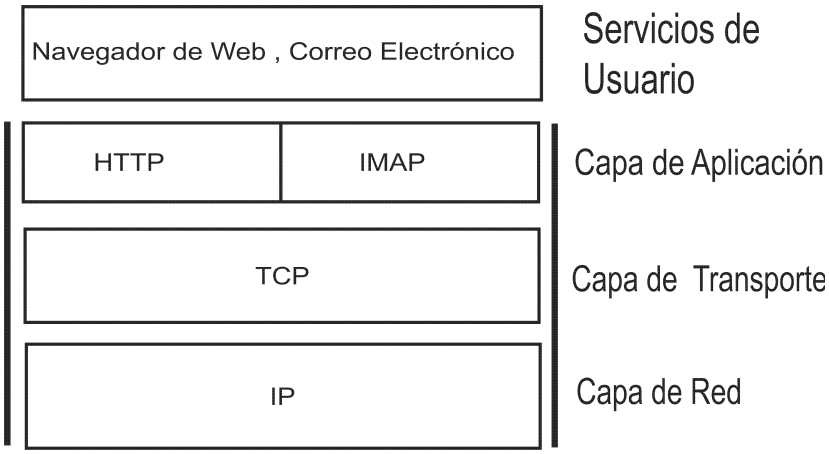


Figura 2.10 Servicios de Usuario de Internet

2.2.1 Protocolos de la capa de transporte

Las aplicaciones deben hacer uso de capa de aplicación con el fin de proporcionar cualquier servicio que sea con referencia a conectividad de la red. La capa de aplicación debe hacer uso del protocolo de capa de transporte, y los protocolos de la capa de transporte se ejecutan en la parte superior de IP. Actualmente, existen dos protocolos de la capa de transporte están ampliamente disponibles: TCP [RFC793] y User Datagram Protocol (UDP) [RFC 768].

2.2.2 Protocolo de Control de Transmisión (Transmission Control Protocol siglas en Ingles)

Protocolo de control de transmisión proporciona transporte confiable entre el receptor y transmisor. Este mecanismos contiene los tiempos de espera, retransmisiones, y los números de secuencia lo que le permiten al protocolo entregar los datos al receptor exactamente como fueron enviados por el transmisor. TCP también realiza el control de flujo y corrección de errores. Por ejemplo, si Roberto envía un mensaje a Laura el mensaje es "Laura, ¿cómo estás?.ª través de una conexión de TCP,entonces TCP se asegurará de que Laura reciba el mensaje "Laura, ¿cómo estás?"Porque TCP es un protocolo de extremo a extremo. El

receptor envían mensaje de reconocimiento a el transmisor proporcionándole la suficiente información para la detección de errores y el control de flujo se pueda realizar. TCP permite al usuario receptor de demultiplexar los paquetes IP entrantes a las diferentes aplicaciones. Debido a que todos los paquetes que llegan al receptor contienen la misma dirección IP de destino, entonces se necesita un identificador adicional para asociar la llegada de cada paquete a la correcta aplicación. Estos identificadores se denominan como números de puerto TCP, y cada aplicación basada en IP utiliza puerto. Por ejemplo, los paquetes con el número de puerto destino TCP 80 se entregan a la aplicación HTTP, mientras que los paquetes enviados al número de puerto TCP 23 son manejados por el Telnet [RFC 854] aplicación (Figura 3-3).

2.2.3 Protocolo de paquetes de usario (User Datagram Protocol siglas en Ingles)

UDP [RFC 768] es un protocolo que no es confiable en la entrega de los paquetes de datos (datagrama termino en Ingles); es decir, que no asegura que un paquete de datos llegue a su destino por cualquier medio. UDP simplemente realiza el tráfico e IP demultiplexación basado en números de puerto UDP. Ambos UDP y TCP protocolos utilizan los puertos de 16 bits. Después de demultiplexación el tráfico, UDP proporciona una suma de comprobación que permite a los sistemas determinar que los paquetes recibidos son validos. Por ejemplo, si Roberto envía el mismo mensaje que antes pero ahora se utiliza el protocolo UDP, él no se puede estar seguro de Laura lo recibirá. Para averiguarlo, él tendrá que consultar con Laura. Si un datagrama de UDP no llega, Roberto tendra que retransmitirlo; pero el protocolo de UDP no cuenta con el protocolo de transmisión de control del flujo de datos (Stream Control Transmission Protocol siglas en Ingles) El protocolo de transmisión de control de flujo de datos (SCTP) [RFC 2960] es un protocolo de transporte nuevo. Se prevé que este protocolo se utilice ampliamente, pero en la actualidad, es menos disponible que TCP y UDP.

2.3 Protocolo de iniciacion de session (Session Initiation Protocol)

SIP, es el resultado de la unión de dos protocolos. SIP tiene lo mejor de cada uno de los protocolos. Las primeras transmisiones de voz a través de redes se realizaron utilizando conmutación de paquetes, esto fue alrededor de 1974, y los primeros sistemas de conferencia multimedia surgieron a principios de 1990. Thierry Turletti desarrolló el Sistema de Videoconferencia (IVS Video Conference System siglas en Ingles), era un sistema de audio y vídeo que realizaba transmisiones a través de la Internet. Un usuario de video conferencia podría llamar a otro usuario y podría establecer una sesión unidireccional. IVS también podría ser utilizado en sesiones de multidifusión. Tomando la ventaja de la codificación de vídeo H.261 a través de Internet, el trabajo sobre IVS fue implementado en el desarrollo del protocolo del transporte del tiempo real (RTP) para el formato de vídeo H.261 [RFC 2032]. Poco después, Eve Schooler desarrolló el control de conferencia multimedia (Multimedia Conference Control). Estos dos sistemas de multimedia provocaron que de alli surgiera el diseño del protocolo de iniciación de sesion (SIP siglas en Ingles) creada por Mark Handley y Eva Schooler. La primera versión de SIP, SIPv1, fue presentado a la IETF como un proyecto de Internet en 22 de febrero de 1996. SIPv1 utilizaba un protocolo de descripción de sesión (SDP Session Description Protocol siglas en Ingles) para describir sesiones y UDP (User Datagram Protocol siglas en Ingles) como transporte. Este estaba basaba solo texto. Una vez que un usuario había registrado su ubicación, el servidor era capaz de direccionar la ruta para el usuario adecuado.

El 22 de febrero de 1996, Henning Schulzrinne presentó un proyecto de Internet a la IETF con las especificaciones del protocolo de invitación a una simple conferencia (Simple Conference Invitation Protocol siglas en Ingles). SCIP era también un mecanismo para invitar a los usuarios de un punto a otro punto y sesiones de multidifusión. Este estaba basaba en el Protocolo de transferencia de hipertexto (Hyperttext Transfer Protocol) y por lo tanto se utiliza el Protocolo de Control de Transmisión (TCP) como protocolo de transporte.

Despues Schooler presentó el protocolo SIP y Schulzrinne presentó el SCIP, después de varias discusiones se decidió fusionar los dos protocolos. El protocolo SIP resultante mantiene como un nombre, pero cambió el significado de las siglas del protocolo de Inicio de Sesión (SIP) y avanzó la versión de número 2. El nuevo SIP se basa en HTTP, pero podría usar tanto UDP como TCP como protocolo de transporte. Este usa el SDP para describir sesiones de multimedia. SIP empezó a ganar importancia en el IETF, dando lugar a la formación de un nuevo grupo de trabajo SIP en septiembre de 1999. A partir de julio de 2001, SIP todavía no está terminado como un producto. Su estado actual se proponía como un estándar funcionalidad proporcionada para SIP [RFC 2543] la cual describe el núcleo del protocolo SIP: es decir, el funcionamiento básico del protocolo. Además de esta especificación.

2.3.1 SIP Entities

El protocolo SIP define varias entidades, y es importante para entender su función dentro de la arquitectura dentro del protocolo de SIP.

Agente de Usuario Es la entidad que interactúa con el usuario. Este tiene una interface de usuario como una ventana de menú y con selección de botones.

Herramientas de Multimedia SIP proporciona una descripción de la sesión a una entidad SIP de UA. Si la sesión es de voz, el UA tendrá que entregar a la herramienta de voz para controlar el audio. Para otros tipos de sesiones, la UA entregar las herramientas adecuadas para la media utilizada.

2.3.2 Operación del protocolo SIP

SIP esta basado en el protocolo Hypertext Transfer Protocol (HTTP Hypertext Transfer Protocol siglas en Ingles), SIP es un protocolo de petición / respuesta. Para entender el mecanismo de solicitud / respuesta utilizado en SIP, tendremos que examinar la siguiente definiciones del cliente y el servidor. Un cliente es una entidad SIP que genera peticiones y un

servidor es una entidad SIP que recibe las solicitudes y devuelve una respuesta. Esta terminología es hereda de HTTP, en el que un navegador Web contiene un cliente HTTP. Cuando se escribe una dirección en mi navegador web, como http://www.uabc.com, se está enviando una solicitud a un servidor de Web en particular. El servidor Web envía una respuesta a la información que le fue solicitada.

2.3.3 Respuestas de protocolo SIP

Después de recibir una solicitud, el servidor emite una o varias respuestas. Cada respuesta que realiza tiene un código indicando el estado de la transacción. El estado de los códigos son números enteros que van desde 100 a 699 y se agrupan en clases, como se muestra en la Tabla 2.2.

Tabla 2.2 Respuestas de SIP

Modelo OSI	
Rango	*Respuestas*
100-199	Informativo
200-299	Exitoso
300-399	Re-dirección
400-499	Error de Cliente
500-599	Error de Servidor
600-699	Error Global

Una respuesta con un código 100 a 199 se considera provisional. Las respuestas de 200 a 699 son respuestas finales. Junto con el código de estado, las respuestas de SIP llevan una razón.Por ejemplo, un código de 180 es una alerta indicando que el usuario es invitado a una sesión. Por lo tanto, la razón es contener un "Zumbido".

2.3.4 Las solicitudes de SIP

La especificación del protocolo SIP define seis tipos de solicitudes SIP, cada uno de ellos con un propósito diferente. Cada petición de SIP contiene un campo, llamado método, que

denota su propósito. La lista muestra los seis métodos.

INVITE Solicitud para invitar a participar a una sesión.

ACK Solicitud usada para darle una respuesta a una invitación

OPTIONS Solicitud al servidor para recabar las capacidades del servidor

BYE Solicitud para abandonar una sesión

CANCEL Solicitud de cancelar una transacción, como lo puede ser una INVITE.

REGISTER El usuario envia una solicitud para registrarse al servidor.

2.4 Transporte RTP

La llamada tiene que haberse establecido para que RTP pueda trasmitir datos de voz y video a través de una red.Sin embargo, RTP también se puede utilizar para transmitir muestras creadas para otros sonidos. RTP se usa en la parte superior de UDP. UDP es el transporte no confiable más utilizado sobre el protocolo IP. UDP sólo puede garantizar la integridad de datos mediante el uso de una suma de comprobación, pero una aplicación que utiliza UDP tiene que hacerse cargo de todas las tareas de recuperación de datos. UDP proporciona también la noción de un "puerto", que es un número entre 0 y 65535 (presente en cada paquete como parte de la dirección de destino) que permite hasta 65.536 objetivos UDP para distinguir en la misma dirección IP de destino. Un puerto también está unido a la dirección de origen y permite hasta 65.536 fuentes de distinguirse de la misma dirección IP. Por ejemplo, un RTP con la secuencia, UDP se pueden enviar desde 192.168.0.11:1234 a 192.168.0.15:5678 (Figura 2.11):

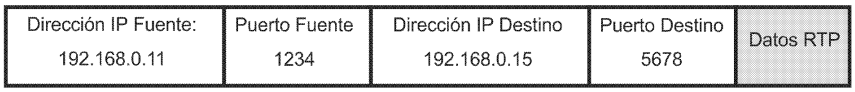


Figura 2.11 Servicios de Usuario de Internet

Sesión RTP : una sesión de RTP es una asociación de los participantes que se comunican a través RTP. Cada participante utiliza al menos dos direcciones de transporte (por ejemplo, dos puertos UDP en elmáquina local) por cada período de sesiones: una para el flujo RTP, uno para los informes de RTCP. Cuando se utiliza una transmisión por multidifusión a todos los participantes utilizan el mismo par de multidifusión direcciones de transporte. Flujo de audio video en la misma sesión comparten un canal común de RTCP. Considerando que H.323 o SIP requieren aplicaciones para definir un puerto para cada canal de audio video. Así que, aunque la mayoría cumplen con los requisitos para RTP y RTCP para los puertos de RTP y el puerto adjayecte para RTPC.

Fuente de sincronización (SSRC) : identifica la fuente del flujo de RTP, identificado por 32 bits en la cabecera de RTP. Todos los paquetes RTP con un SSRC común tienen un tiempo comun y una referencia secuencia. Cada emisor tiene que tener un SSRC; cada receptor también el receptor necesita tener al menos un SSRC ya que esta información se utiliza para los informes de receptor (RR).

Contribuir fuente (CSRC) cuando un flujo de RTP es el resultado de una combinación poner varios flujos y mezclado en un RTP, la lista de los SSRC de cada flujo se añade en la cabecera RTP de la cada flujo. El flujo resultante tiene su propia SSRC.

Formato NTP una forma estándar para el formato a una marca de tiempo, escribiendo el número de segundos desde 1/1/1900 a las 0 h con 32 bits para la parte entera y 32 bits para la parte decimal.Pero tambien existe un compacto con sólo 16 bits para la parte entera y 16 bits para la parte decimal. Los primeros 16 dígitos de la parte entera por lo general se puede derivar a partir del día actual, la parte fraccional es simplemente truncado para los más significativos 16 dígitos.

2.4.1 Paquetes de RTP

En la figura 2.12 se describe los campos de RTP.

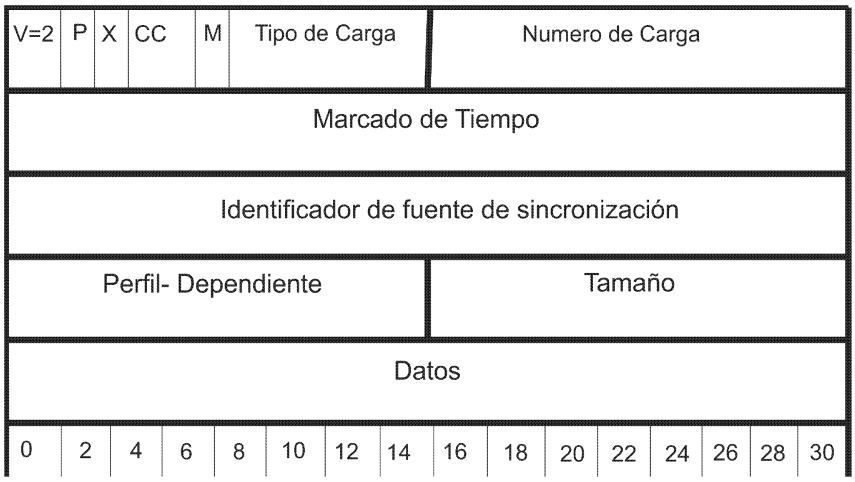


Figura 2.12 Formato de los paquetes RTP

Aquí tenemos una breve explicación de cada campo de RTP:

- Dos bits están reservados para la versión RTP, que ahora es la versión 2 (10). Versión 0 fue utilizado por el VAT y la versión 1 fue un proyecto IETF antes.
- Un bit de relleno P indica si la carga útil se ha rellenado con fines de alineación. Si se ha rellenado (P = 1), entonces el último octeto de la carga útil indica más campo con precisión cuántos octetos de relleno se han añadido a la carga útil originales.
- Un bit de extensión X indica la presencia de extensiones después del eventual CSRCs la cabecera fija.
- 4 bits de conteo CSCR (CC) establece cuantos identificadores CSRC siguen la cabecera fija.
- Marcador (M): 1 bit. Su uso es definido por el perfil de RTP.
- Tipo de carga útil (PT): 7 bits. La carga útil de cada paquete RTP es la información en tiempo real contenida en cada paquete. Su formato es completamente libre y debe ser definido por el aplicación o el perfil de RTP en uso. Permite a las aplicaciones

distinguir un formato particular sin tener que analizar el contenido de la carga útil. Algunos identificadores comunes se enumeran en la Tabla 1.1; que son utilizados Estos son:

Tabla 2.3 Tipos de carga útil

Tipo de Carga Útil	
Tipo de Carga Util	Codificadores
0	PCM,u-law Audio
8	PCM, A-law
9	G.722
4	G.722
15	G.728
18	G.729

- Un número de secuencia y de marca de tiempo: el número de secuencia es de 16 bits y marca de tiempo empieza con una valor aleatorio y se incrementan en cada paquete de RTP. La marca de tiempo utiliza 32 bits y utiliza un reloj una frecuencia que es definido para cada tipo de carga útil (por ejemplo, carga útil H.261 utiliza una Reloj de 90 kHz para la marca de tiempo RTP). Para los codificadores de audio de banda estrecha (G.711, G.723.1,G.729, etc.) la frecuencia de reloj RTP está ajustado a 8000 Hz.Cada paquete RTP lleva un número de secuencia y una marca de tiempo. Marcas de tiempo RTP no tienen un significado absoluto (las marcas de tiempo iniciales de un flujo de RTP puede ser seleccionada al azar).Una aplicación de audio utilizará el número de secuencia y marca de tiempo para gestionar una memoria intermedia de recepción. Por ejemplo, una aplicación puede decidir almacenar de 20 cuadros por segundo de audio antes de comenzar la reproducción.

2.5 VoIP codificadores de Audio

Flujo de Audio y vídeo que son creados cuando hablamos a traves de un micrófono de la computadora o utilizamos una cámara digital y comenzamos con señales analógicas. Hubo

un tiempo cuando la señal voz atravesaba toda la red telefónica en forma analógica. Sin embargo, esta arquitectura estaba muy limitada y engorrosa. Actualmente la conversación de voz o vídeo, si se utiliza en un sistema tradicional de telefonía o uno basado en voz sobre IP, siempre comienzan como una señal analógica, pero luego es convertida a un formato digital. La representación digital de la voz o video se transmite al receptor y aquí nuevamente se vuelve a convierte de nuevo a analógico con el fin de ser entendido por el ser humano. En su mayor parte, este proceso de analógico al formato digital y conversión de digital a analógico es manejado por un codificadores o codificador-decodificador. Hay muchas técnicas diferentes que se utilizan para manejar estos flujos de audio y vídeo. La mayoría de los codificadores populares utilizados en la actualidad están estandarizados en el UIT-T (Telecomunicaciones Internacionales Unión-Telecom siglas en Ingles) aunque hay muchos otros. La gran parte del trabajo hecho con codificadores ha estado en un enfocado a reducir la cantidad de ancho de banda consumido por el flujo de voz a través del uso de la compresión. Voz sobre IP , requieren los mismos procesos de conversión, a pesar de que no siempre tienen el mismo ancho de banda de una topología tradicional. Como seres humanos, tenemos las capacidades limitadas en términos de la audición y de generación sonidos. El oído humano puede oír frecuencias tan bajas como 20 Hz (hercios) y los hasta aproximadamente 20.000, o 20KHz (Figura 2.13).

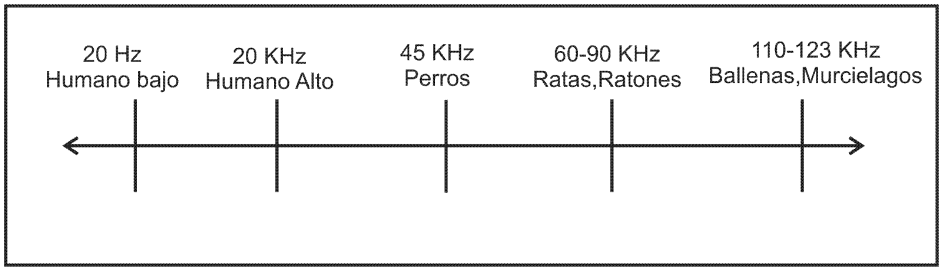


Figura 2.13 Rangos de escucha

También podemos generar sonidos, pero el rango es mucho menos de lo que podemos escuchar. según el libro Guinness de los Récords Mundiales, la frecuencia más baja generada por un humano era justo por encima de 0 Hz, y la más alta fue un poco menos de 5000Hz.

Dado que la mayoría de los sonidos que creamos son de bajas frecuencias, el rango estándar es de 0-4000Hz. Para sistemas de transmisión, este rango en realidad es reducido de 300-3400Hz como se indica en el estándar en el documento UIT-T G.107. Este recomienda el modelo computacional para la planificación de la transmisión.

2.5.1 Señales de Voz

Una señal de audio o voz es una forma de onda analógica. Una forma de onda analógica básica se muestra en la Figura 2.13. Esta imagen ofrece tres segundos de reproducción de un tono generado en 500Hz. Parece como si se trata de un bloque continuo de información analógica. A un ritmo de 500Hz, hay 500 ciclos, o formas de onda completa, por segundo. Una sola onda puede ser transmitida en una de cinco centésimas de segundo, o .002 segundos. Esto también se llama el período de la onda. Dicho de otra forma, en los tres segundos podemos ver en la figura 2.14.

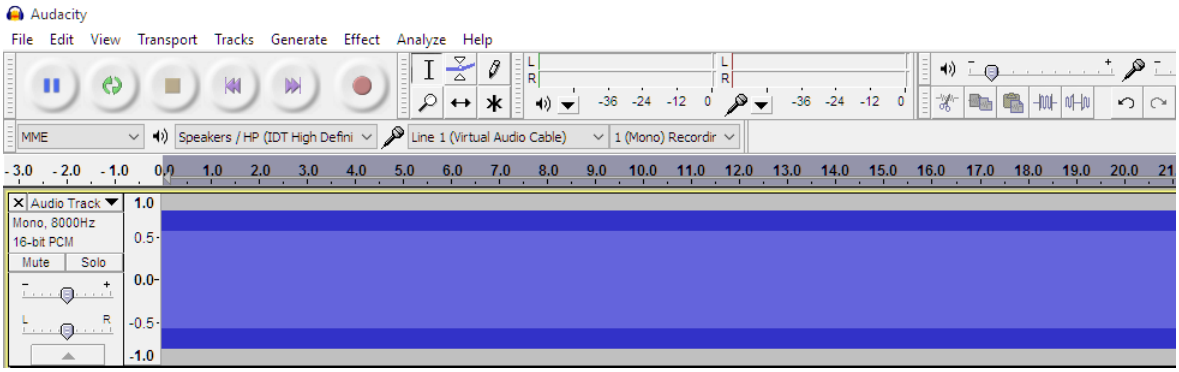


Figura 2.14 Formato de los paquetes RTP

Con la figura 2.15, es un acercamiento para examinar lo que ocurre en una centésima de un segundos (.01 segundos), la forma de onda puede ser visto. De hecho, hay cinco ciclos presente si medimos desde el primero hasta el último cruce por cero.

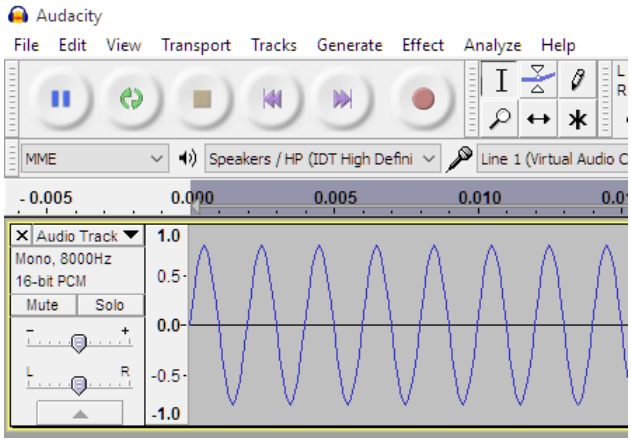


Figura 2.15 Forma de onda de una fracción de un segundo

Las formas de onda analógicas tienen tres características principales: la frecuencia, amplitud y fase. Fase (relación con respecto al tiempo o en otro punto en el tiempo) no juega mucho de un papel en este trabajo, pero la amplitud (fuerza) de la onda y la frecuencia (ciclos por segundo) lo hacen. Debido a que es un sonido único que no cambia el volumen, la frecuencia y amplitud permanecen constantes. Quinientos hertz esta dentro del rango de la audición humana y cae dentro de las frecuencias permitidas en el circuito de teléfono. Pero escuchar este tono sería bastante aburrido, ya que no existen variaciones. El lenguaje humano es una colección de frecuencias que varían. Incluso la más simple de la señal contiene una amplia variedad de estas frecuencias.

Un ejemplo de una señal de voz se muestra en la Figura 2.16. En este caso, representa la palabra "Hola", que se dijo tres veces.

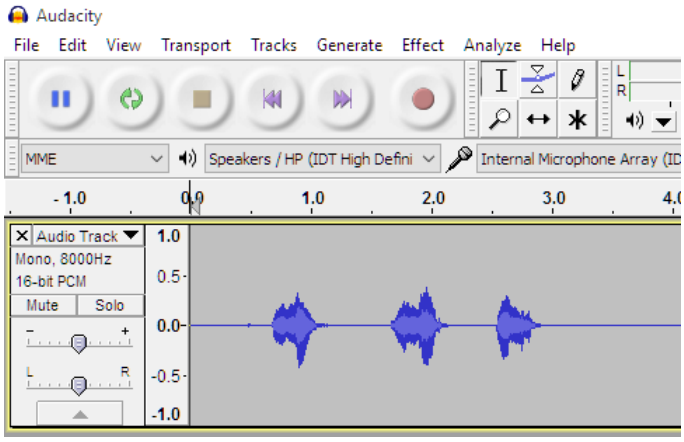


Figura 2.16 Mensaje "Hola"

Aunque podemos ver alguna variación en la señal, incluso cuando se repite la misma palabra, lo que no se claro en la imagen es lo complejo que en realidad podría ser la señal. Ampliando la base de tiempo de la forma de onda y se centrándonos en una parte más pequeña, podemos ver algunos de los detalles como lo vemos en la figura Figura 2.17

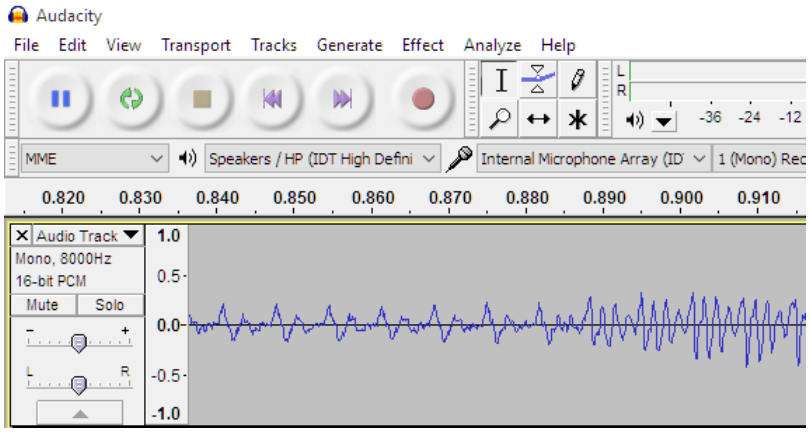


Figura 2.17 Mensaje "Hola.^{ex}pandido

A diferencia de la sección expandida del tono de 500 Hz, hay variaciones en la frecuencia y en la amplitud. Esto es porque los sonidos que se generan, incluso dentro de la misma palabra, varían en fuerza y tono. Señales como estas son enviadas por los cables telefónicos

a la central telefónica. El reto para un codificador de audio es capturar todos esas variación en una serie de muestras y que se puede poner de nuevamente juntos en el otro extremo .

2.5.2 Codificadores y Decodificadores de Audio

Un codificador tiene la tarea de convertir la onda analógica en una serie de muestras y luego proporcionando un valor binario para esa muestra. Esto se hace para que los datos de voz que se pueden transferir través de las partes digitales de la red telefónica pública conmutada, o PSTN. Pero esto es sólo la mitad de la historia. En el receptor, estas muestras deben ser re-ensamblados y convertidas de nuevo en una forma de onda analógica que puede ser interpretado correctamente. Este proceso es llamado código de modulación por impulsos (Pulse Code Modulation Siglas en Ingles), y sirve como el método de base que se utilizada en la PSTN y muchas implementaciones de VoIP para la codificación de datos de voz. Este esta definido por la UIT-T G.711.

Recordemos que ambos extremos de la conexión son típicamente analógica, y la onda analógica debe ser primero muestreada. En el PCM, una muestra es simplemente una evaluación de la intensidad de la señal en tiempo.El truco con las muestras es tomar suficiente muestras para proporcionar una reproducción precisa de la señal original.Una frecuencia de muestreo baja puede causar perdida de detalles de la señal original. Además, una baja frecuencia de muestreo puede provocar distorsión en la señal. Esto ocurre cuando una señal transmitida parece ser de una frecuencia inferior a la señal real. El muestreo en exceso consume un gran ancho de banda innecesario.

Una vez que se toman las muestras, un valor se debe asignar a cada uno de los valores obtenidos en un proceso llamado de cuantificación. Cada codificador ofrece un número limitado de valores para estas muestras, Codificadores como G.711 requerir amplitud granulada en amplitudes más bajas para ser mayor que en los superiores. En otras palabras, hay una mayor precisión en las amplitudes inferiores. En resumen, sabemos que las tasas de muestreo que son demasiado pequeñas o demasiado grandes crean problemas. También sabemos que los niveles de cuantificación impropias pueden afectar la precisión de la señal a la hora de

recrearlos en el receptor.

Entonces el estándar G.711 PCM crea 8000 muestras por segundo. La justificación de este valor se atribuye a Harry Nyquist. La premisa es que para reproducir con precisión una señal, debe muestrear a una tasa igual a dos veces el ancho de banda inicial. Para G.711 PCM, cada muestra se le asigna un valor basado en ocho bits. Esto proporciona 256 valores posibles. La capacidad del canal es el producto de los dos. Para un canal codificado-PCM G.711, la velocidad de datos es igual a la tasa de muestreo por los bits de muestro.

$$8000 \quad samples \quad x \quad 8 \quad bits \quad per \quad sample \quad = 64000 \quad bits \quad per \quad second$$

Los 64Kbps asignados para un canal de voz G.711 es la mayor cantidad de ancho de banda consumido por un codificador de audio en la mayoría de los sistemas. Esto no suena como un valor muy grande pero cuando hay una preocupación de ancho de banda, sobre todo en la red de área extensa (WAN), 64 Kbps puede parecer una gran cantidad para una sola llamada. Por esta razón, otros métodos han sido desarrollados para proporcionar una mayor compresión y una reducción de los recursos consumidos. Pero incluso con varios tipos de compresión, modulación por impulsos codificados proporciona la base para la mayoría comúnmente utilizado codificadores.

2.5.3 Estándares de Codificadores

Los codificadores de audio y vídeo más comunes están estandarizados por el Instituto ITU-T G. En general,estos codificadores se dividen en dos tipos: en forma de onda y de la fuente de la codificación. A continuación veremos el muestreo de la forma de onda analógica, su compresión y cuantificación. La cuantización son los valores que se transmiten al receptor y se decodifican directamente de la forma original de onda. La forma de onda de loas codificadores son fáciles de reconocer, ya que se basan normalmente en PCM. Los codificadores fuente tal como G.729 intentan determinar un valor que describe los datos (metadatos) y enviarlo para enviarlo. Por ejemplo, G.729 analiza muestras de voz para predecir el coefi-

cientes. Esos parámetros son codificados y se transmiten. Los forma de onda más comunes,la fuente del codificador y sus técnicas básicas se describen en el siguiente párrafo..

G.711 Modulacion por codigo de Pulso (PCM) Este surge en el ano de 1972.Como se dijo anteriormente, este define al codificador de 64 Kbps PCM codificador de datos de voz y los pone en el formato adecuado para PSTN. Esto se logra mediante el muestreo de voz 8000 veces por segundo (uno cada 125 microsegundos)y la asignación de 8 bits por muestra.Existen dos versiones diferentes de G.711: G.711 (Mu-Law) y G.711(A-Law) . Ellos son similares pero tienen una pequeña diferencias en el detalle de la señal.Mu-Law es utilizado en en América del Norte y Japón, mientras que A-Law es utilizado en Europa.

G.722 7KHz Codificador de Audio Este utiliza una técnica llamada PCM subbanda diferencial adaptativa (SB-ADPCM) codificación en un rango de 50-7000Hz. Este tiene tres modos de funcionamiento que resultan en las tasas de transmisión de 48, 56, y 64 Kbps. Pero el codificador utiliza una tasa de muestreo más alta de 16 kHz y 14 bits por muestra. Las sub-bandas dividen el canal en la parte baja (0-4000Hz) y bandas superiores (4000-8000Hz). Los resultados de subbanda inferiores en una Señal de 48 Kbps y para la parte superior es de 16 Kbps . Independientemente del modo, la codificación base es de 64 Kbps.

G723.1 Codificador Dual de voz para comunicaciones de multimedia Este codificador utiliza una tecnica de voz y audios a una tasa muy bajas de bits (5.3Kbps y 6.3Kbps) para voz y audio como parte del estandard. A diferencia del tamaño de cuadro de 20 milisegundos utilizado por G.711, G.723 utiliza 30 milisegundos y un 7,5 milisegundos para ver adelante, haciendo un total de retardo de 37,5 milisegundos en la creación del marco. Como la mayoría del flujo de datos de voz se muestrea primero en una manera compatible con G.711 y luego se convierte a la PCM lineal de 16 bits requerido por el codificador, la cual trabaja en bloques de 240 (30 ms) muestras.

G.726 Codificación diferencial adaptativa PCM a los 16, 24, 32, y 40kbps Aprobado en 1990, el codificador convierte el canal de 64 Kbps estándar a uno de los velocidades listadas anteriormente. Las velocidades de datos individuales tienen diferentes aplicaciones que van del uso de los equipos multiplexores o módems.

G.729 Codificación de Discurso a 8Kbps Usando Estructura Conjugada ACELP G.729 y G.729 proporcionan una calidad similar a 32Kbps ADPCM. G.729 es sin duda uno de los codecs más populares junto a el estándar G.711. Por ejemplo, algunos dispositivos Cisco utilizan esto como el codificador por defecto. Desarrollado en 1996, el estándar G.729 utiliza la predicción lineal descrita anteriormente, pero las cuadros de voz utilizados son 10 mili-segundos y por lo tanto el tamaño son de 80 muestras. Se toman muestras cada 125 micro-segundos.

Capítulo 3

Método Propuesto

Como parte de la propuesta, se analizaron los protocolos actuales de VoIP, con la finalidad de retomar aquellos elementos que nos ayuden a proponer un nuevo modelo de tal manera que podamos aplicarlo de forma eficiente en un sistema embebido de recursos limitados de computo. De todo el análisis pudimos detectar que hay varias maneras de comprimir la cabecera, ya que utiliza una redundancia, es esencial para proporcionar un servicio de VoIP en las redes inalámbricas y de una forma eficiente en un sistema embebido. Un paquete VoIP es generalmente es transportado por el protocolo RTP / UDP / IP. La cabecera sin comprimir es de aproximadamente los paquete de voz son de 40 bytes .

- IPv4 Identificación(IP-ID), 16 bits
- UDP Checksum, 16 bits
- RTP Marcador (M-bit), 1 bit
- RTP Numero de Secuencia (SN), 16 bits
- RTP Marca de Tiempo (TS), 32 bits

Sin ninguna compresión de la cabecera, esto causa problemas de sobrecarga, lo cual esto pudiera llegaría a ser dominante y tienen un impacto significativo en la capacidad de densidad causando problemas en el proceso de transmisión/recepción de los paquetes, lo que no contribuye en gran manera a un sistema embebido de recursos limitados. Los encabezados de paquetes de RTP, UDP, e IP a través del flujo de paquetes tiene redundancia significativa. Los campos en una cabecera se clasificaran en las siguientes categorías:

Campos estáticos o conocidos , que no necesitan ser enviados en cada paquete. Por ejemplo, la dirección de origen y destino en la cabecera IP, el puerto de origen y destino en el encabezado UDP

Campos inferidos , que pueden inferirse de otros campos y por lo tanto no necesitan ser enviado con cada paquete; por ejemplo, la marca de tiempo en la cabecera RTP, la cual se incrementa en una cantidad fijada con cada aumento en el número de secuencia. El tiempo de marca por lo tanto no necesita ser enviado durante el envió de paquetes mientras esta la conversación una vez que el número de secuencia se este enviando.

Cambio de los campos que se pueden enviar en forma comprimida para ahorrar ancho de banda.

La Figura 3.1 muestra la clasificación de los encabezados RTP/UDP/IP.

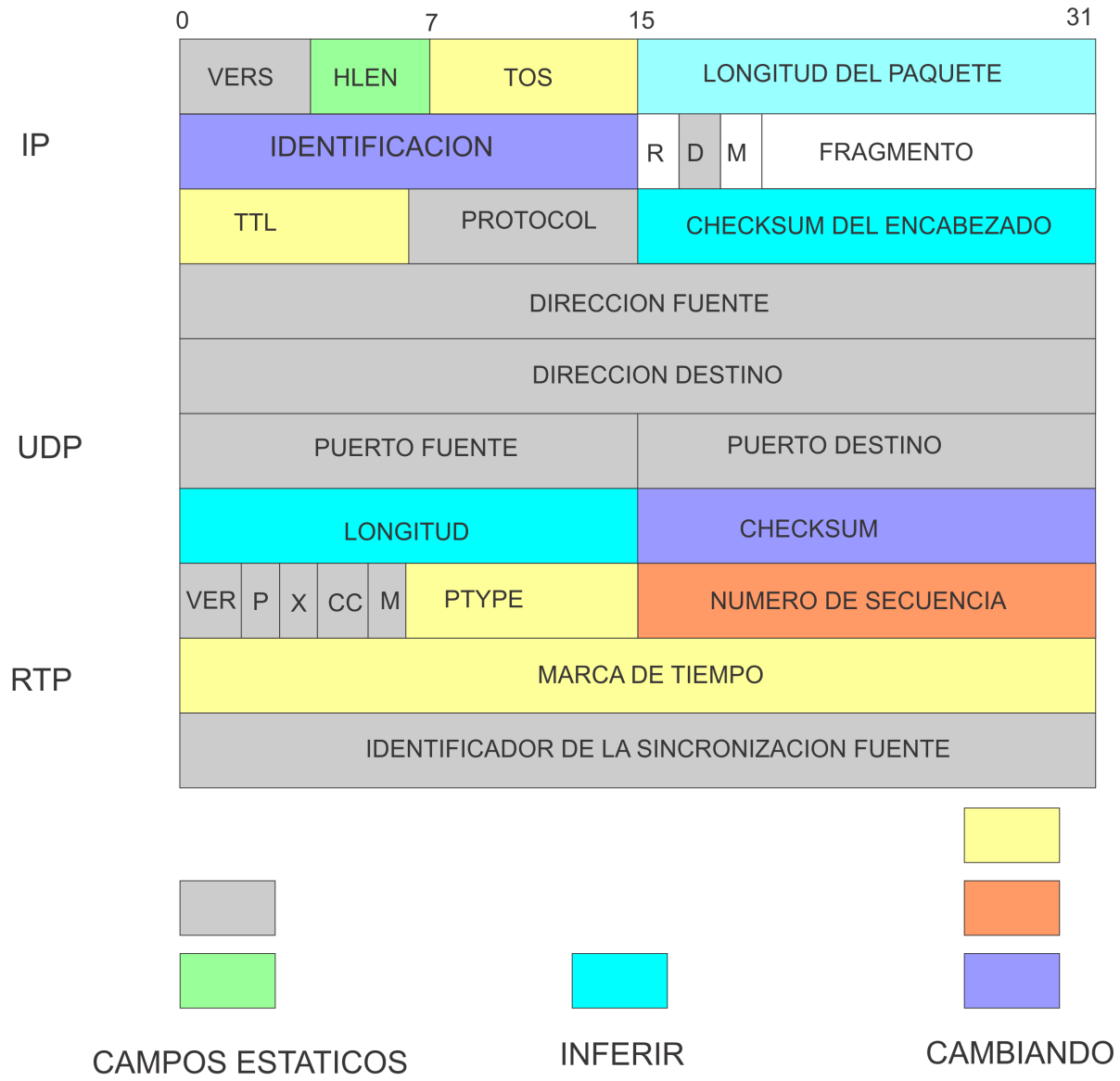


Figura 3.1 Clasificación de los encabezados de RTP,UDP y IP

Para hacer mas viable la transmicion y recepcion de VoIP sobre una infrastructura de red inalámbrico en un sistema embebido, un número de técnicas deben ser abordadas. Uno de los factores más críticos es la eficiencia de ancho de banda y el otro factor es el procesamiento que se requiere para su generación .El ancho de banda y los recursos de computo siguen siendo recursos escaso y de vital importancia.Habiendo dicho eso la eficiencia del ancho de banda

es crítica para VoIP sobre wireless, un inmediato problema inmediato es que los grandes gastos en los encabezados de VoIP. Los datos de voz sobre IP es típicamente realizado por el protocolo RTP. Un paquete de VoIP tendrá una cabecera IP de 20 octetos, un UDP cabecera de 8 octetos, y una cabecera RTP de 12 octetos. Esto hace un total de 40 octetos para la Cabecera de RTP / UDP / IP. Si se utiliza IPv6, la cabecera IP solo será de 40 octetos y el total de Cabecera RTP / UDP / IP será de 60 octetos. Esto es además de la sobrecarga del protocolo de interfaz aérea. Por otro lado, el tamaño de la carga útil de voz es típicamente tan pequeño como 2 a alrededor de 20 octetos.Por tanto, es imprescindible comprimir la cabecera de RTP / UDP / IP.

Típicamente, los incrementos de campo del número de serie (SN) en uno por cada paquete enviado por la fuente de RTP. El TS y campos IP-ID lo general puede predecirse a partir de la SN. El Checksum de UDP no debe ser previsto y se envía tal cual cuando está.

3.1 Simulador

Después del análisis existente de los protocolos de VoIP que se utilizan actualmente, podemos reducir redundancia en la cabecera el protocolo que se utiliza para envíar y recibir paquetes de voz, de esta manera tenemos un protocolo mas eficiente que ayudara a su implementacion eficiente y de esta manera pueda ser soportado por un sistema empotrado de bajos recursos.

La propuesta del nuevo protocolo esta descrita en la siguiente figura 3.2 Un paquete de voz en el nuevo protocolo sera de 24 bytes.El protocolo IP se queda intacto, ya que tiene que ser compatible con el actual protocolo IP para que pueda viajar a través del protocolo de internet. La distribución del nuevo protocolo queda de la siguiente manera:

- IPv4 , 20 bytes
- nuevo, 4 bytes

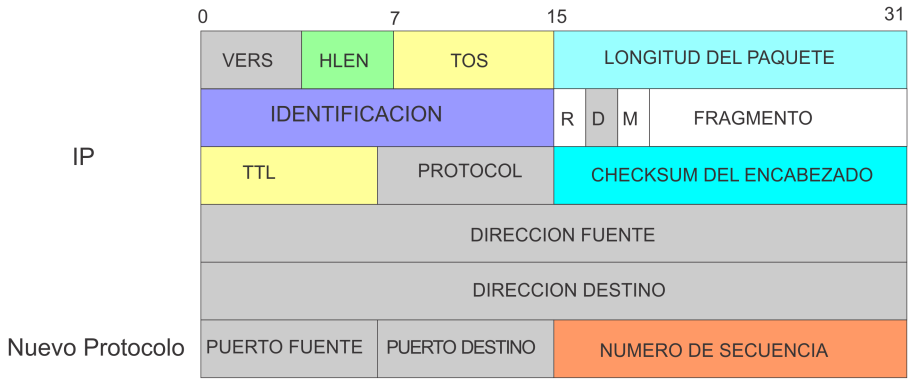


Figura 3.2 Propuesta del Protocolo

3.1.1 Herramienta de simulación

EL simulador utilizado para realizar las pruebas se llama OMNeT ++. Es una aplicación de computadora de código abierto. Ademas es sinónimo de objetos Modulares para pruebas de redes que utiliza un lenguaje C++ para la programación de los módulos . Contiene bibliotecas de simulación basado en componentes escritos en lenguage de programacion C ++, diseñado para simular redes de comunicación. OMNeT ++ no es un simulador de redes, sino un marco que le permite crear sus propias simulaciones de red.

El resultado de la simulación puede ser analizada para determinar cómo el verdadero la red se verá afectada por las condiciones y parámetros utilizados en la simulación.

3.1.2 Componentes del Simulador

A continuación se presenta la descripción de los componentes que utiliza el simulador.

Red En OMNeT ++, es un objeto que define la red y tiene módulos, módulos compuestos y submódulos . El objeto de red permite diferentes tipos de módulos que se comunican entre sí a través de canales. Cuando se ejecuta la simulación de la red se verá como en la siguiente ejemplo en la figura 3.3:

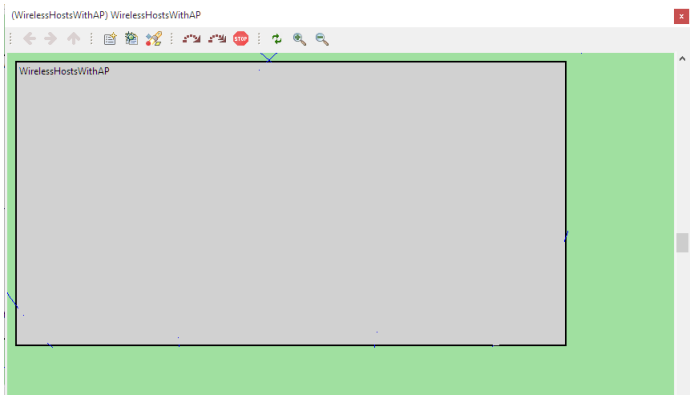


Figura 3.3 Red llamada WirelessHostWithAP

Modulo El módulo es un componente que se encuentra dentro de un objeto de red que tiene la capacidad de enviar mensajes a otros objetos de módulo. En la simulación, este modulo podría ser un ruteador, un servidor de web, una máquina autónoma, o cualquier otro componente que pueda pensar, que sea capaz de hacer la comunicación a través de una red.

Modulo Compuesto Este se compone de varios objetos o módulos. El marco de INET del simulador OMNeT ++ es una biblioteca que contiene bastantes módulos pre-compilados y módulos compuestos . Por ejemplo un modulo podria ser un punto de acceso llamado accessPoint (Figura 3.4) es un módulo compuesto del objeto Wireless-HostWithAP:

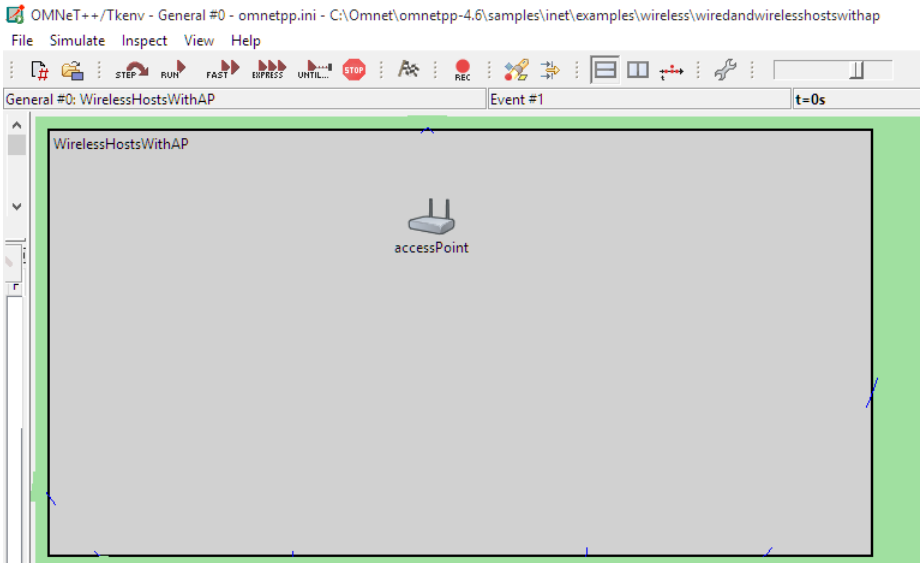


Figura 3.4 Red llamada WirelessHostWithAP

INET proporciona un icono ordenada para el módulo compuesto accesPoint. Echándole un vistazo a lo que hace a este objeto, se puede ver una colección de modulos y módulos compuesto (figura 3.5)que juntos conforman un típico o punto de accesso (accesPoint) que puedes encontrar en una red de la vida real.

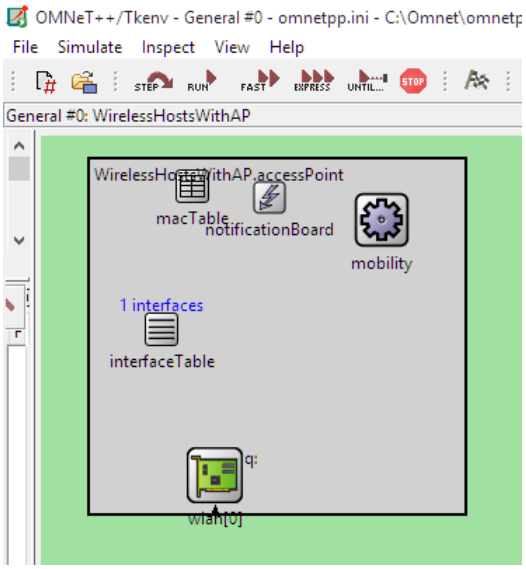


Figura 3.5 Modulo accessPoint con sus modulos compuestos

3.1.3 Canales

Los canales son objetos que se utilizan para conectar un módulo de un módulo compuesto a otro. Los canales permiten a estos módulos enviar mensajes entre sí y conectar la interface de un módulo a la interface de otro. Esto podría ser un punto de accesso(accessAP) hablando con otro punto de accesso(accessPoint) través de wifi.

3.1.3.1 Lenguaje y descripción de la red (NED)

NED es un lenguaje de descripción de la red que se utiliza para crear la topología de redes en OMNeT ++. La interface grafica de OMNeT ++ le permite crear gráficamente cualquiera topología utilizando NED. Si decide crear sus topologías de red utilizando el editor gráfico, automaticamente generará el código fuente de NED .

3.2 Pruebas y Resultados

El plan de pruebas a realizar se dividen en las siguientes etapas:

Prueba al actual protocolo . La simulación del actual protocolo donde el transmisor y el receptor se encuentran conectados a través del cable de red directamente, el objetivo es de evaluar el tiempo de transmicion y recepción como base para realizar el comparativo, como lo podemos ver en la en la figura (3.6)

Estos son los parámetros que se utilizaron para realizar la simulación utilizando un protocolo estándar. Configuración del Protocolo

- Total de bytes que se transmites son 200 bytes compuesto por los siguientes componentes

  IP (20 bytes)

  UDP (8 bytes)

  RTP (12 bytes) encabezados.

  Carga Útil 160 Bytes (Audio no codificado)

- Configuración de red

  Velocidad de Datos = 12Mbps

Protocolo propuesto En estas pruebas se utilizo la topologia anterior como se ve en la fig(3.6), unicamente lo que cambia es que los bytes que se transmiten y se reciben:

- Total de bytes que se transmites son 184 bytes compuesto por los siguientes componentes

  IP (20 bytes)

  Nuevo protocolo (4)

  Carga Útil 160 Bytes (Audio no codificado)

- Configuración de red

  Velocidad de Datos = 12Mbps

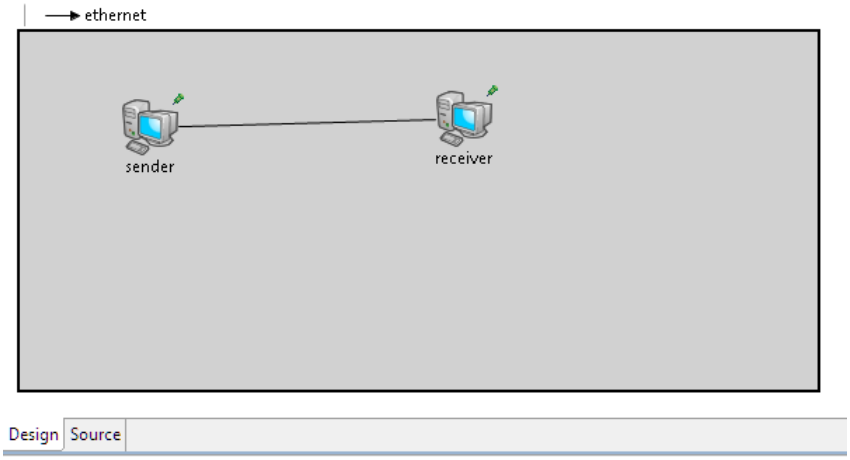


Figura 3.6 Transmisor y Receptor

3.2.1 Resultados de la Simulación

Los resultados que se presentan en la siguiente tabla, muestran el tiempo que tarda desde que envía los paquetes de audio desde son transmitidos hasta son recibidos los paquetes, como se puede observar el flujo de en la figura 3.7

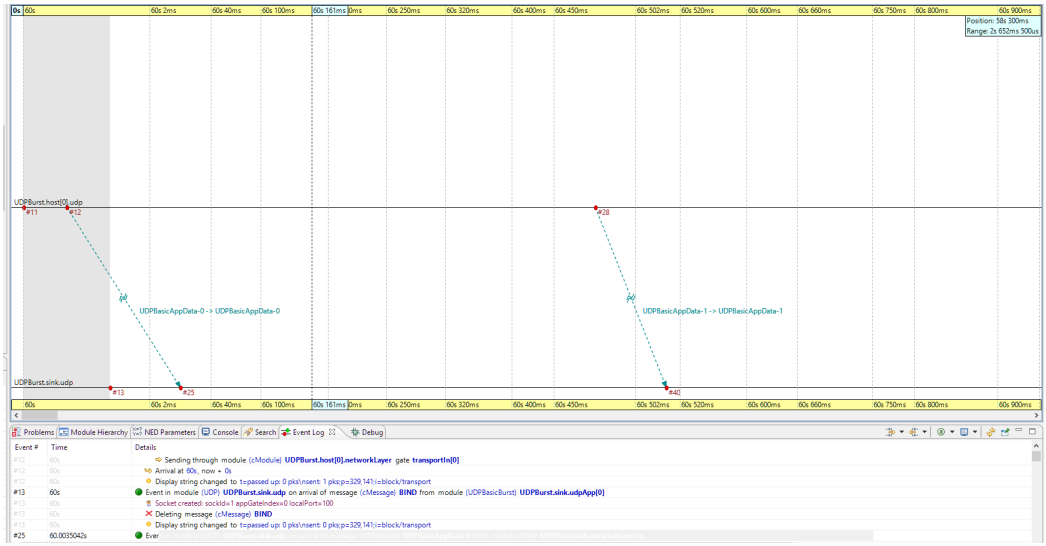


Figura 3.7 Simulacion de la red cableada

Los resultados obtenidos de la simulación están descritos en la siguiente tabla 3.1 Donde el tiempo inicial es en el momento que se envió los paquetes, tiempo final es el tiempo que el receptor recibió los paquetes de voz, y la columna de duración es el tiempo que tardo el paquete desde que se envió hasta que llego al receptor.

Tabla 3.1 Resultados de la simulación

Protocolo	Tiempo de Inicio (mSeg)	Tiempo Final(mSeg)	Duración (msec)
Estandar	60.0000000	60.0037602	3.7602
Propuesto	60.0000000	60.0035000	3.5042
Eficiencia en tiempo (usec)			256.00
Eficiencia en porcentaje			6.80

Podemos observar una reducción de 256 msec utilizando el nuevo protocolo, desde que es los paquetes son transmitidos hasta que llega al receptor, considerando que el protocolo propuesto se pretende utilizar en un sistema embebido de recursos limitados de computo, este desempeño se considera que no seria un porcentaje apropiado para cumplir con el objetivo.

Por lo tanto revisando que se podría realizar con la carga útil, existe actualmente codificadores de voz que se utilizan dentro de la industria de comunicaciones, explorando cual codificadores de voz se podría que pueda adaptarse para un sistema embebido con recursos limitados. De tal manera que el codificador no utilice algoritmos muy complejos que requieran grandes recursos de procesador y memoria para poder realizar la codificación. Esos codificadores son: G.726 y G.729

3.2.1.1 Pruebas con codificador de voz

En la siguiente tabla tenemos a los codificadores de audios que utilizaremos en la simulación indicando respectivamente los datos de carga útil y paquetes por segundo, como otra de las formas de mejorar la eficiencia en la transmicion y recepción de los paquetes de voz.

Tabla 3.2 Codificadores de Voz

Codificador de Voz	Carga Útil (mSeg)	Carga Útil(Bytes)	Paquetes por segundo
PCMU	20	160	50
G.726	20	60	50
G.729	30	20	33.3

Primeramente se realizaran pruebas con la topologia anteriormente mencionada en la Fig 3.6.Posteriormente se utilizara una topologia de red inalambrica, tratando de incluir los elementos que se utilizan en un ambiente real. El modulo de red es "WirelessHostWithAP",los modulos compuestos son: wirelessHost1 es un cliente configurado como transmisor y el cliente wirelessHost2 como receptor. El punto de acceso es .ªccessPointPara"que conecta a ambos clientes a traves de la red inalambrica. a estas pruebas se utilizo la siguiente configuración de red con los parámetros correspondientes. figura 3.8

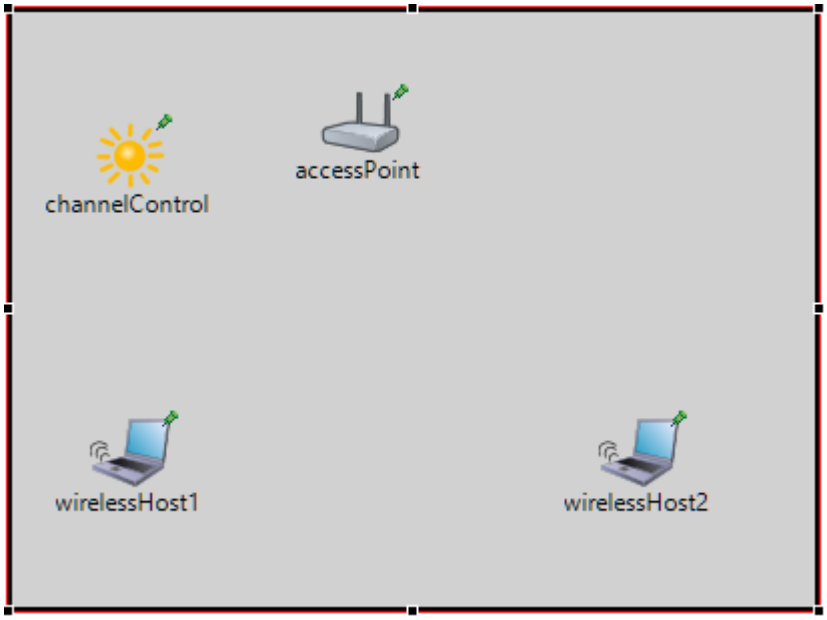


Figura 3.8 Simulación de la red inalambrica

Canal de Control Potencia de transmicion de 20 mW

Frecuencia de la portadora 2.4 GHz,

	Numero de canales 1
Punto de Acceso	Potencia de transmicion de 20 mW
	Frecuencia de la portadora 2.4 GHz,
	Numero de Radios 1
wirelessHost1	Configurado como Transmisor
	Dirección IP:10.0.0.1
	Direccion Fisica (MAC)0A-AA-00-00-00-01
	Potencia de transmicion de 20 mW
	Frecuencia de la portadora 2.4 GHz,
	Numero de Radios 1
wirelessHost2	Configurado como Receptor
	Direccion IP:10.0.0.2
	Direccion Fisica (MAC)0A-AA-00-00-00-02
	Potencia de transmicion de 20 mW
	Frecuencia de la portadora 2.4 GHz,
	Numero de Radios 1

3.2.1.2 Resultados de la simulación modificando la carga útil

En la siguiente gráfica se muestran resultados obtenidos de la configuración y topologia descrita en la figura 3.6 y sus resultados están en la gráfica 3.9. Como podemos observar la utilizando del protocolo propuesto en combinación con el codificador de audio G.729 se pudo obtener un alto porcentaje de eficiencia en la transmicion y recepción de paquetes de voz, como podemos verlo en la gráfica, el valor obtenido es de 45 % de eficiencia.

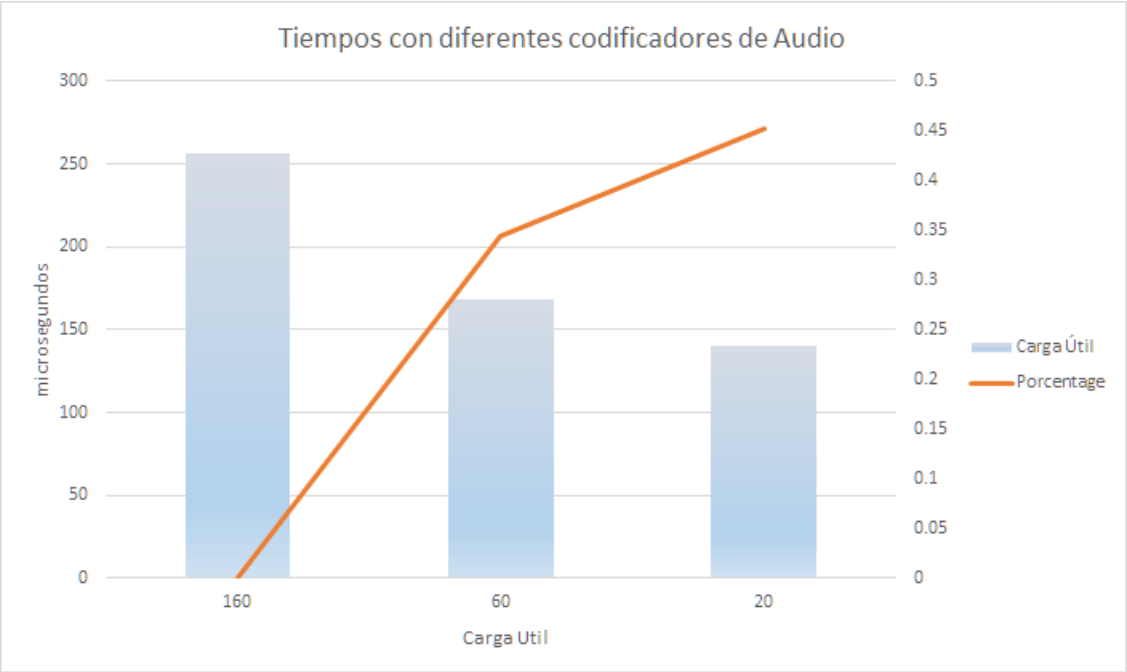


Figura 3.9 Resultados de la simulación

A continuación se tenemos los resultados de la simulación utilizando una red inalambrica, con la configuración descrita anteriormente como en la Figura 3.8 y utilizando el nuevo protocolo propuesto. Como podemos observar en la figura 3.10 que el protocolo propuesto y el codificador de audio G.729 tiene una eficiencia bastante alta. basta los codificadores de voz que existen actualmente se seleccionaron los que están descritos en el siguiere tabla por las características que se adaptan mas a un sistema de recursos limitados.

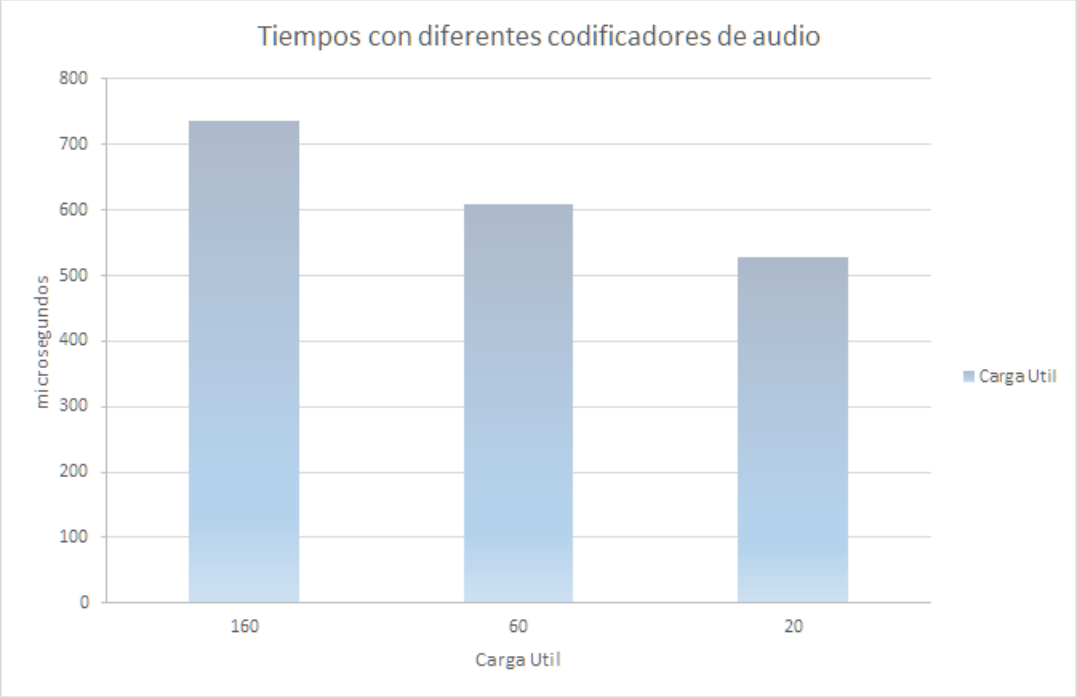


Figura 3.10 Resultados de la simulación

Capítulo 4

Implementación

Una vez propuesto y simulado el método para la transmisión y recepción de voz a través de redes inalambricas procederemos con la implementación.

El método propuesto se implemento utilizando dos configuraciones , la primera configuración fue una computadora personal portátil configurada como transmisor y otra computadora personal portátil configurada como receptor.

La segunda configuración donde se implemento el método propuesto fue en un sistema embebido, cuyo tarjeta de desarrollo fue adquirido de la compañía llamada Econias (www.econias.com) la que fue configurado como receptor.

4.1 Implementación en un sistema de computadoras portátiles

Antes de explicar la implementación del modelo, describiremos el diagrama a bloques del programa y del equipo de los sistemas que se utilizaron en la implementación. En la figura 4.1 tenemos el primer sistema que consiste en dos computadoras personales, una configurada como transmisor y la otra como receptor que se conectan a la red inalambrica a través de un punto de acceso.

Figura 4.1 Diagrama a bloques del sistema de computadoras personales

Una vez presentado el diagrama a bloques del equipo pasamos al diagrama a bloques del programa a alto nivel. Tendremos un diagrama a bloques tanto para el transmisor como el receptor. Además se tiene un archivo en formato .wav (.WAV siglas en ingles Windows wave) que contiene un mensaje de voz grabado, con las siguientes características: la frecuencia de muestreo es de 8 KHz, la cantidad de bit que se utilizan para la captura son de 16 bit y de un solo canal.

El proceso del transmisor contiene dos tarea ejecutándose concurrentemente, una encargada de crear una conexión UDP con el encabezado modificado de acuerdo al modelo propuesto en este trabajo, además de leer los datos del archivo donde se encuentra almacenada los datos de voz, posteriormente esos datos son codificados bajo el formato G.729, y enviados a el receptor a través de la red inalámbrica.

Entonces podemos observar que la implementación de la propuesta dentro del sistema, esta en dos partes, uno momento de generar la conexión UDP modificando el encabezado y la otra parte es la codificación los paquetes de voz, y al momento de prepáralos para de ser enviados, como se puede ver en la siguiente figura 4.2.

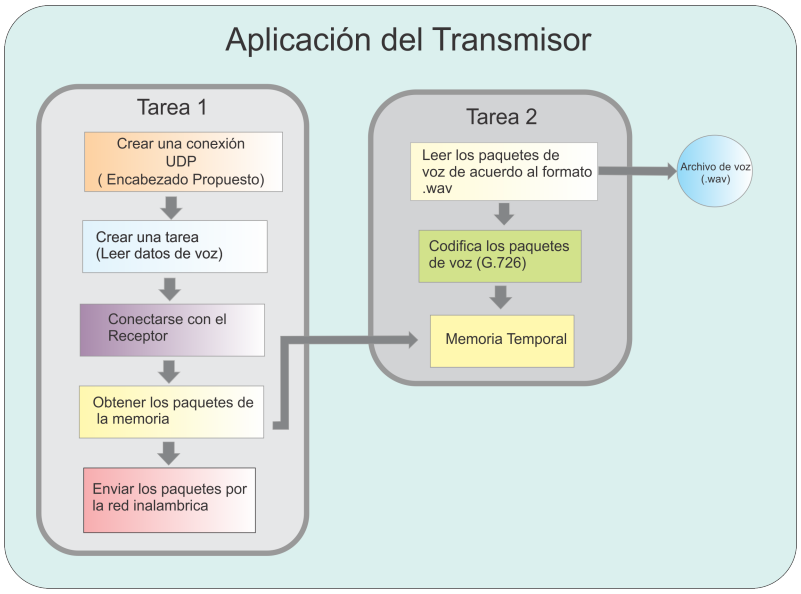


Figura 4.2 Diagrama a bloques de la aplicación del transmisor

De igual manera en el receptor, se tienen dos tareas, una de ellas tiene la función de crear una conexión UDP con el encabezado modificado de acuerdo al modelo anteriormente mencionado ,capturar los paquetes a través de la red inalámbrica después los almacena en memoria temporal. La tarea dos del receptor una vez conectado al transmisor , esta se encarga de estar leyendo los paquetes de voz codificados, posteriormente los decodifica con el formato G.729 y esos paquetes de datos crudos (.wav) son enviados al reproductor de voz que a su vez son enviados al amplificador, y se puedan ser escuchados en la bocina , como se puede ver en la figura 4.3

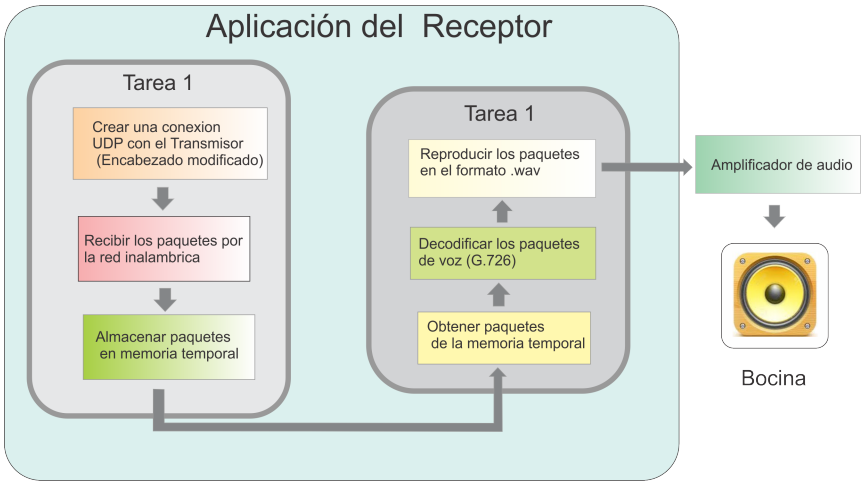


Figura 4.3 Diagrama a bloques de la aplicación del transmisor

Aquí esta el diagrama de flujo de los programas del transmisor y receptor de voz a través de una red inalambrica. En la figura 4.4 tenemos mas a detalle el flujo del programa del transmisor, de lo que explicamos anteriormente.Haciendo énfasis en lo que el modelo esta proponiendo dentro del proceso del sistema. En la figura 4.5 tenemos el diagrama de flujo del programa del sistema de recepción del modelo propuesto.

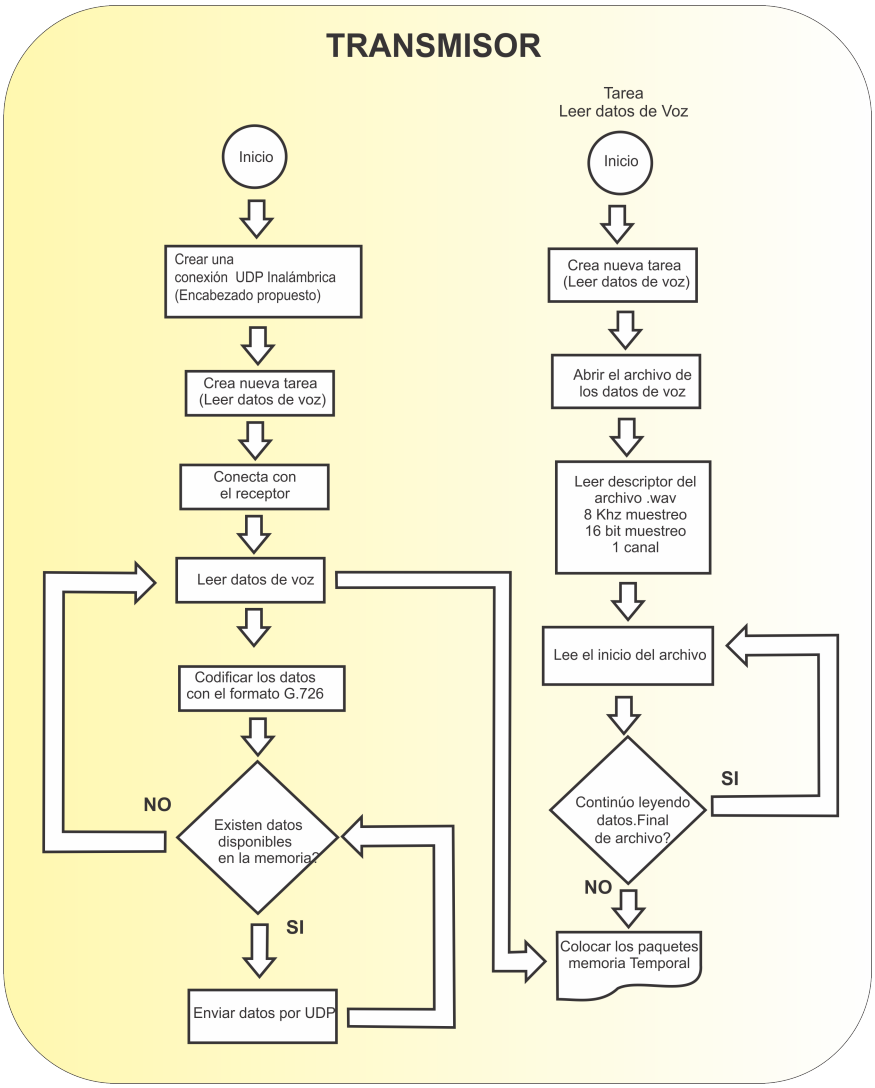


Figura 4.4 Diagrama de flujo del transmisor

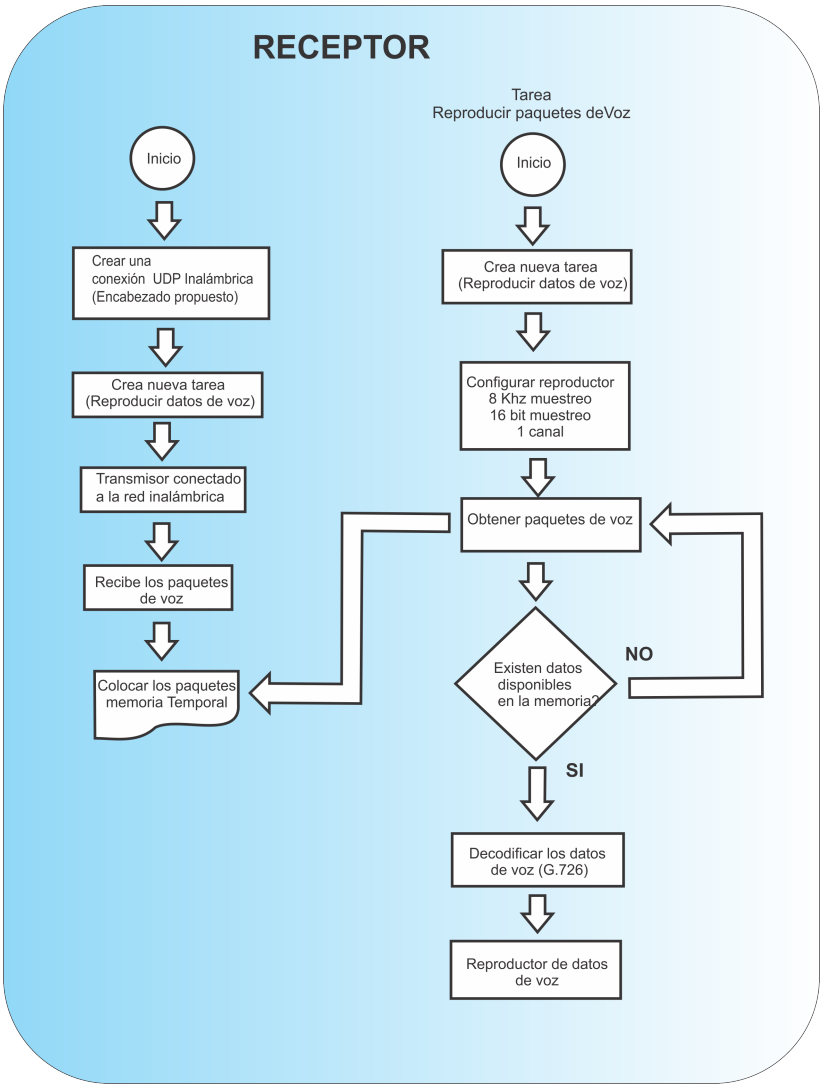


Figura 4.5 Diagrama de flujo del receptor

4.2 Implementación en el Sistema Embebido

El modelo también se implementó en otro sistema, la cual consistía en una computadora personal portátil y un sistema embebido, este sistema embebido que fue adquirido de la compañía Econais, el número de parte del módulo de desarrollo es el EC32L13WADK. En la figura 4.6 tenemos la descripción del sistema. La computadora portátil esta ejecutando

el mismo proceso que esta descrito en la sección anterior 4.1 específicamente descrito en la figura 4.2. Lo que si cambia es el proceso que se tiene en el sistema embebido.Este utiliza un sistema operativo de tiempo real para manejar multitareas y administración de los recursos del sistema.

Figura 4.6 Diagrama a bloques de la computadora personal portátil y el sistema embebido

El sistema de tiempo real es el ChibiOS un sistema operativo en tiempo de código abierto. ChibiOS está diseñado para aplicaciones en tiempo real profundamente arraigadas en la eficiencia y la ejecución de código compacta son requisitos importantes. El núcleo ChibiOS está muy optimizado para el cambio de contexto extremadamente eficiente y ofrece un basto conjunto de comandos primitivos como para manejo de tareas, temporizadores virtuales, semáforos, exclusiones mutuas, las variables de estado, mensajes, buzones de correo, banderas de eventos y colas. Entonce el diagrama de flujo del programa se puede ver en la figura 4.7 en la que la tarea principal inicializa el modulo de la red inalambrica,se crea una tarea encargada de estar capturando los paquetes que esta recibiendo, y esos datos colocarlos en una memoria temporal. Despues inicializa el codificador de audio la cual sera responsable de reproducir los datos de voz, la configuración del codificador de audio es .wav, la frecuencia de muestreo es de 8 KHz, la cantidad de bit que se utilizan para su muestreo es de 16 bit y de un solo canal. Aqui es donde se lleva acabo las inicializaciones como lo viene siendo: la

memoria de audio, la inicializacion del hardware es decir la configuracion de los registros de los puertos de entrada salida, los temporizadores, el tamaño de la memoria de acceso directo, como todas las configuraciones de los registros para darle servicio a la comunicación con el codificador de audio, la comunicación con el módulo de la red inalámbrica.

Después de todas las inicializaciones se crea una conexión de UDP con el encabezado propuesto, y entra al ciclo permanente, en ese ciclo se estará verificando si existen datos disponibles en la memoria temporal que han llegado a través de la red inalambrica, si existen datos recibidos entonces los lee, se decodifican con el formato G.729 , el resultado se coloca en la memoria de acceso directo, la cual se comunicara directamente con el codificador de audio, esto significa que en cuanto existan datos en esta memoria de acceso directo, mandara una interrupción al codificador de audio provocando que el codificador de audio tome los datos los amplifique y los envié a la bocina para que sea audible los datos de audio.

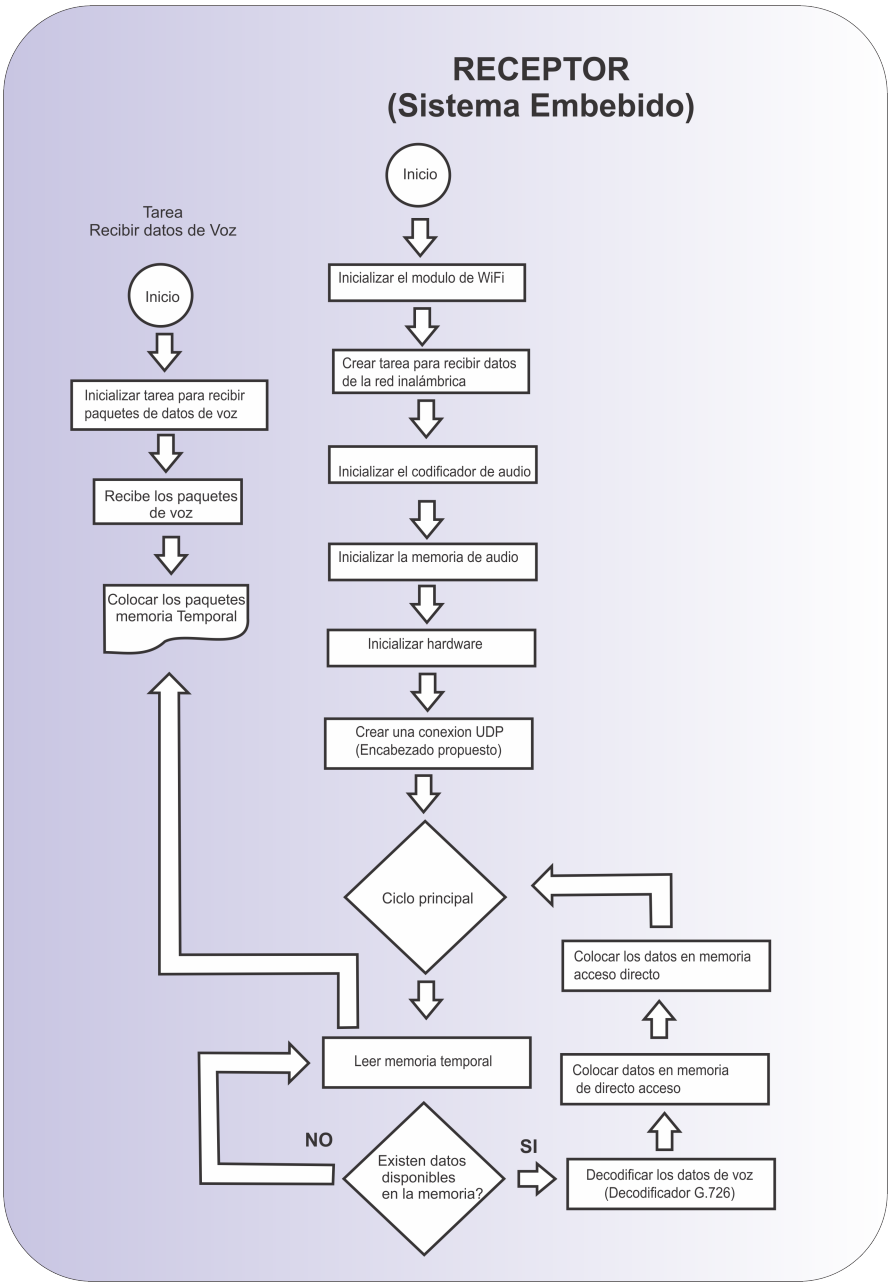


Figura 4.7 Diagrama a bloques del programa del sistema embebido

Capítulo 5

Pruebas y Resultados

En este capítulo se describen las pruebas realizadas a los sistemas que se utilizaron para implementar el modelo propuesto con su respectiva validación. También se presentan los resultados de las pruebas realizadas. Básicamente se realizaron pruebas con dos sistemas como lo hemos venido describiendo, uno de ellos consistía en dos computadoras personales portátiles la cual se instaló un programa en ambas computadoras, no de ellas fue configurada como transmisor y la otra como receptor de voz a través de la red inalámbrica.

El programa que se ejecuta en esta computadoras fue desarrollado en lenguaje de programación C# para ambas funciones, transmisor y receptor. El otro sistema que se utilizó para la implementación fue el de una computadora portátil configurada como transmisor y un dispositivo embebido utilizado como receptor que se comunicaban a través de un punto de acceso.

Enseguida se describe el equipo utiliza en el sistema donde se utilizaron dos computadoras portátiles una como transmisor y la otra como receptor.

- Dos computadoras portátiles,HP Pavilion Sleekbook y Toshiba T215D-SP1004
- Un punto de acceso

Estas son las condiciones de configuración de los equipos de los equipos utilizados:

- Ambas computadoras estaban ejecutando la aplicacion bajo un Sistema Operativo Windows 7. La configuración de la red inalambica es relizada para que se conecten al punto de acceso con al conjunto de servicios de identificación (SSID Siglas en Ingles Service Set Identification (IEEE 802.11 wireless networks) nombrado "Prueba"

- Una aplicacion desarrollada para ejecutarse en Windows, el principal propósito de esta es implementar y evaluar el método propuesto.
- Punto de acceso (AP Siglas en Ingles Access Point) que es configurado con un conjunto de servicios de identificación "Prueba", el nivel de seguridad es abierto, la frecuencia de la portadora en 2.4 GHz, utilizando el canal 6 MHz y a una velocidad de transmisión de 12 Mbps y 1 Mbps.
- La computadora Toshiba Satellite T215D es configurada como recepción de voz.
- Un archivo de voz grabado con una extensión .wav (.WAV siglas en ingles Windows wave) que sera reproducido por la computadora portátil configurada como transmisor, las siguientes características del archivo son: 8 KHz de muestreo, 16 bit de muestreo y solo canal.

Aquí se describe la configuración que se utilizo en el sistema de una computadora portátil y el sistema embebido utilizados para validar la propuesta del modelo.

- Una computadoras portátiles,HP Pavilion Sleekbook
- Un punto de acceso
- Sistema embebido EC32L13WADK (Econias)
- Bocina o unos articulares con conector de 3.5 mm

Estas son las condiciones de configuración de los equipos de los equipos utilizados:

- Computadoras estaban ejecutando la aplicacion bajo un Sistema Operativo Windows 7. La configuración de la red inalambica es relizada para que se conecten al punto de acceso con al conjunto de servicios de identificación (SSID Siglas en Ingles Service Set Identification (IEEE 802.11 wireless networks) nombrado "Prueba"

- Una aplicacion desarrollada para ejecutarse en Windows, el principal propósito de esta es implementar y evaluar el método propuesto.

- Punto de acceso (AP Siglas en Ingles Access Point) que es configurado con un conjunto de servicios de identificación "Prueba", el nivel de seguridad es abierto, la frecuencia de la portadora en 2.4 GHz, utilizando el canal 6 MHz y a una velocidad de transmisión de 12 Mbps y 1 Mbps.

- Un sistema embebido configurado como recepción de voz.

- Un archivo de voz grabado con una extensión .wav (.WAV siglas en ingles Windows wave) que será reproducido por la computadora portátil configurada como transmisor, las siguientes características del archivo son: 8 KHz de muestreo, 16 bit de muestreo y solo canal.

Describiremos el proceso de prueba que se utilizo para validar el modelo. Antes de iniciar se configura el punto de acceso con la identificación como lo comentamos en la sección anterior, el nombre del identificador es "Prueba"sin ninguna contraseñara, la velocidad de transmisión sera primero de 12 Mbps, de 1 Mbps.Haciendo referencia al sistema donde se utilizan las dos computadoras, se les instala la aplicacion encargada de transmitir/recibir los datos de voz.Se les configura la red inalambrica de tal manera que se conecten al punto de acceso "Prueba", sin contraseña. A la computadora Toshiba esta sera configurada como receptor habilitando la opción en la aplicacion. La computadora HP se configura como transmisor, se ejecuta la aplicacion y se habilita la opción para transmitir los datos de voz,se selecciona el archivo donde están almacenados los datos de voz, en la misma aplicacion se coloca la dirección de IP (Internet Protocol siglas en Ingles)del receptor.

En el sistema donde se utiliza una computadora portátil y el sistema embebido este es el proceso que se siguió para realizar las pruebas. El punto de acceso es el mismo que anteriormente se utilizo, con la misma configuración. La computadora HP que se utilizo con la configuración anterior como transmisor también se utilizo, lo que cambia es que la

dirección a la cual se van a enviar los datos de voz,que viene siendo el sistema embebido. Se tiene que configurar el sistema embebido a través de modificar el código fuente para que se conecte al punto de acceso, y no se coloca contraseña, se le conecta la bocina. El criterio de los resultados está basados en la percepción de la calidad de audio del usuario, considerando la evaluación de las características que determinan una recepción y transmisión de voz a través de la red inalámbrica. Los resultados de las pruebas se ven en la tabla 3.1 donde el criterio de aceptación es que todas las pruebas evaluadas sean aceptables.

Tabla 5.1 Resultados de las pruebas

Configuración	Transmicion (Mbps)	Características	Resultado
Computadora (Trasnmisor)-Computadora (Receptor)	12	No se escucha ruido	Aceptable
		Voz se escucha continua	Aceptable
		Se entiende el mensaje	Aceptable
		No hay interrupciones continuas	Aceptable
Computadora (Trasnmisor)-Computadora (Receptor)	1	No se escucha ruido	Aceptable
		Voz se escucha continua	Aceptable
		Se entiende el mensaje	Aceptable
		No hay interrupciones continuas	Aceptable
Computadora (Trasnmisor)-Sistema Embebido(Receptor)	12	No se escucha ruido	Aceptable
		Voz se escucha continua	Aceptable
		Se entiende el mensaje	Aceptable
		No hay interrupciones continuas	Aceptable
Computadora (Trasnmisor)-Sistema Embebido(Receptor)	1	No se escucha ruido	Aceptable
		Voz se escucha continua	Aceptable
		Se entiende el mensaje	Aceptable
		No hay interrupciones continuas	Aceptable

Capítulo 6

Conclusiones

El alto grado de crecimiento de los dispositivos inteligentes móviles y la expansión de las redes de comunicaciones inalámbricas utilizadas actualmente para la transmisión y recepción de voz se convierte atractivo para un explotar la actual infraestructura a través que un sistema de tiempo real de bajos recursos sea capaz de reutilizarla para la transmisión y recepción de voz.

A pesar de que el protocolo de comunicación no esta pensado para los sistemas embebidos de bajos recursos, con el modelo propuesto en esta investigación,que esta basado en modificar el encabezado de los paquetes de transmicion y recepcion conjuntamente con la codificación de los datos de voz, los resultados obtenidos nos indican que es posible transmitir y recibir voz de una manera eficiente, con la simulación. Los resultados de las pruebas donde se implementó el modelo, nos indican que es posible la transmisión y recepción de voz a través de un sistema embebido. Se realizaron a 2 niveles de velocidad de transmisión de los paquetes de voz, y en ambos casos se obtuvieron buenos resultados, esto quiere decir que en el receptor que escucho la reproducción de voz de una manera aceptable, no se escuchó ruido, era entendible el mensaje recibido. Como resultado de la investigación la aplicación que puede tener este modelo conjuntamente con un sistema embebido de recursos limitados de computo puede abarcar varios campos, como podría ser enviar mensajes y salvarlos en memoria, un sistema de voceo en alguna institución educativa, hospital entre otras.

6.1 Trabajo Futuro

El desarrollo tecnológico de los dispositivos electrónicos e integración de los mismos se ha logrado que en la actualidad se pueda encontrar sistemas embebidos que integre el protocolo de la red inalámbrica (TCP/IP) y la unidad de procesamiento en un solo circuito integrado. Por su tamaño tan pequeño se pudiera adaptarse en un sistema ubicuo, ademas de tener un costo bastante atractivos. Tal es el caso de los siguientes dispositivos. El módulo WiFi ESP8266 es un circuito integrado (SoC System on a Chip siglas en Ingles) contiene integrado en hardware el protocolo TCP / IP que puede dar acceso a cualquier micro controlador de una red inalámbrica. El ESP8266 es capaz de tener integrar el código de la aplicación y tener todas las funciones de red inalámbrica en el mismo circuito integrado. El módulo ESP8266 es un costo extremadamente eficaz comparado con circuitos integrados que además son más grandes y requieren de mayor integración para lograr lo que este circuito integrado proporciona. Otro de los circuitos integrados que estaría interesante utilizar para la implementación es el CC3200. Creado para la Internet de las cosas (Internet Of Things Siglas en Ingles), el dispositivo CC3200 es un microcontolador(MCU Microcontroller Unit Siglas en Ingles) conteniendo el protocolo TCP/IP para una red inalámbrico, integra una arquitectura ARM (Advanced RISC Machine Siglas en Ingles) de alto rendimiento para procesamiento de señales (Cortex-M4), permitiendo desarrollar aplicaciones completas con un solo un circuito integrado.

Contiene un subsistema de administración de energía que incluye convertidores DC-DC que soportan una amplia gama de voltajes de alimentación integrada. Este subsistema permite modos de consumo de baja potencia, tales como la hibernación con el modo de control de tiempo real.

Referencias

[1] Ahmad Belhoul Y. Ahmet Şekercioğlu Nallasamy Mani, "A Mechanism for Enhancing VoIP Performance over Wireless Networks using Embedded Mobility-Specific Information in RSVP Objects," *IEEE Journal*,

[2] Lizzie Edmea Narváez Díaz,Jesús Arturo Pérez Díaz A Model for Designing WLAN's 802.11 for VoIP IEEE, 2006.

[3] Wenbo Zhai,Jian Wang An Application of VoIP Communication on Embedded System, ICCASM, 2010.

[4] David Villacıs, Freddy R. Acosta, Student Member, IEEE, and Roman A. Lara Cueva, Member, IEEE Department of Electrical and Electronic, Escuela Politecnica del Ejercito Performance Analysis of VoIP Services over WiFi-based systems, 2013

[5] Arlindo F. da Conceiqdo, Jin Lit, Dinei A. Florenciot and Fabio Kon Department of Computer Science, Institute of Mathematics and Statistics, University of Sao Paulo Communication and Collaboration Systems, Microsoft Research. Is IEEE 802.11 ready for VoIP

[6] GREGORY D. ABOWD and ELIZABETH D. MYNATT Charting Past, Present, and Future Research in Ubiquitous Computing

[7] Olivier Hersent,Jean-Pierre Petit,David Gurle IP Telephony: Deploying Voice-over-IP Protocols John Wiley@ Sons, Ltd 2005

[8] Gonzalo Camarillo SIP Demystified,McGraw-Hill 2002

[9] Syed A. Ahson,Mohammad Ilyas VoIP HANDBOOK Applications, Technologies,Reliability, and Security CRC Press 2009

[10] William A. Flanagan VoIP AND UNIFIED COMMUNICATIONS Internet Telephony and the Future Voice Network Wiley 2012

Apéndice A

life.augmented

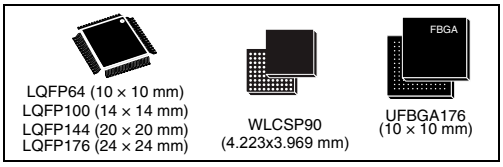
STM32F405xx
STM32F407xx

ARM Cortex-M4 32b MCU+FPU, 210DMIPS, up to 1MB Flash/192+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 15 comm. interfaces & camera

Datasheet - production data

Features

- Core: ARM® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 168 MHz, memory protection unit, 210 DMIPS/ 1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
  - Up to 1 Mbyte of Flash memory
  - Up to 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM
  - Flexible static memory controller supporting Compact Flash, SRAM, PSRAM, NOR and NAND memories
- LCD parallel interface, 8080/6800 modes
- Clock, reset and supply management
  - 1.8 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC (1% accuracy)
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
- Low-power operation
  - Sleep, Stop and Standby modes
  - V_{BAT} supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
- 3×12-bit, 2.4 MSPS A/D converters: up to 24 channels and 7.2 MSPS in triple interleaved mode
- 2×12-bit D/A converters
- General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
- Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 168 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
  - Cortex-M4 Embedded Trace Macrocell™



LQFP64 (10 × 10 mm)
LQFP100 (14 × 14 mm)
LQFP144 (20 × 20 mm)
LQFP176 (24 × 24 mm)

WLCSP90
(4.223x3.969 mm)

UFBGA176
(10 × 10 mm)

- Up to 140 I/O ports with interrupt capability
  - Up to 136 fast I/Os up to 84 MHz
  - Up to 138 5 V-tolerant I/Os
- Up to 15 communication interfaces
  - Up to 3 × I^2C interfaces (SMBus/PMBus)
  - Up to 4 USARTs/2 UARTs (10.5 Mbit/s, ISO 7816 interface, LIN, IrDA, modem control)
  - Up to 3 SPIs (42 Mbits/s), 2 with muxed full-duplex I^2S to achieve audio class accuracy via internal audio PLL or external clock
  - 2 × CAN interfaces (2.0B Active)
  - SDIO interface
- Advanced connectivity
  - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
  - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
  - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
- 8- to 14-bit parallel camera interface up to 54 Mbytes/s
- True random number generator
- CRC calculation unit
- 96-bit unique ID
- RTC: subsecond accuracy, hardware calendar

Table 1. Device summary

Reference	Part number
STM32F405xx	STM32F405RG, STM32F405VG, STM32F405ZG, STM32F405OG, STM32F405OE
STM32F407xx	STM32F407VG, STM32F407IG, STM32F407ZG, STM32F407VE, STM32F407ZE, STM32F407IE

This is information on a product in full production. www.st.com

Contents

1 Introduction 11

2 Description 12

2.1 Full compatibility throughout the family 15

2.2 Device overview 18

2.2.1 ARM® Cortex®-M4 core with FPU and embedded Flash and SRAM . . 19

2.2.2 Adaptive real-time memory accelerator (ART Accelerator™) 19

2.2.3 Memory protection unit 19

2.2.4 Embedded Flash memory 19

2.2.5 CRC (cyclic redundancy check) calculation unit 20

2.2.6 Embedded SRAM 20

2.2.7 Multi-AHB bus matrix 20

2.2.8 DMA controller (DMA) 21

2.2.9 Flexible static memory controller (FSMC) 22

2.2.10 Nested vectored interrupt controller (NVIC) 22

2.2.11 External interrupt/event controller (EXTI) 22

2.2.12 Clocks and startup 22

2.2.13 Boot modes 23

2.2.14 Power supply schemes 23

2.2.15 Power supply supervisor 23

2.2.16 Voltage regulator 25

2.2.17 Regulator ON/OFF and internal reset ON/OFF availability 28

2.2.18 Real-time clock (RTC), backup SRAM and backup registers 28

2.2.19 Low-power modes 29

2.2.20 V_{BAT} operation 30

2.2.21 Timers and watchdogs 30

2.2.22 Inter-integrated circuit interface (I²C) 33

2.2.23 Universal synchronous/asynchronous receiver transmitters (USART) . 33

2.2.24 Serial peripheral interface (SPI) 34

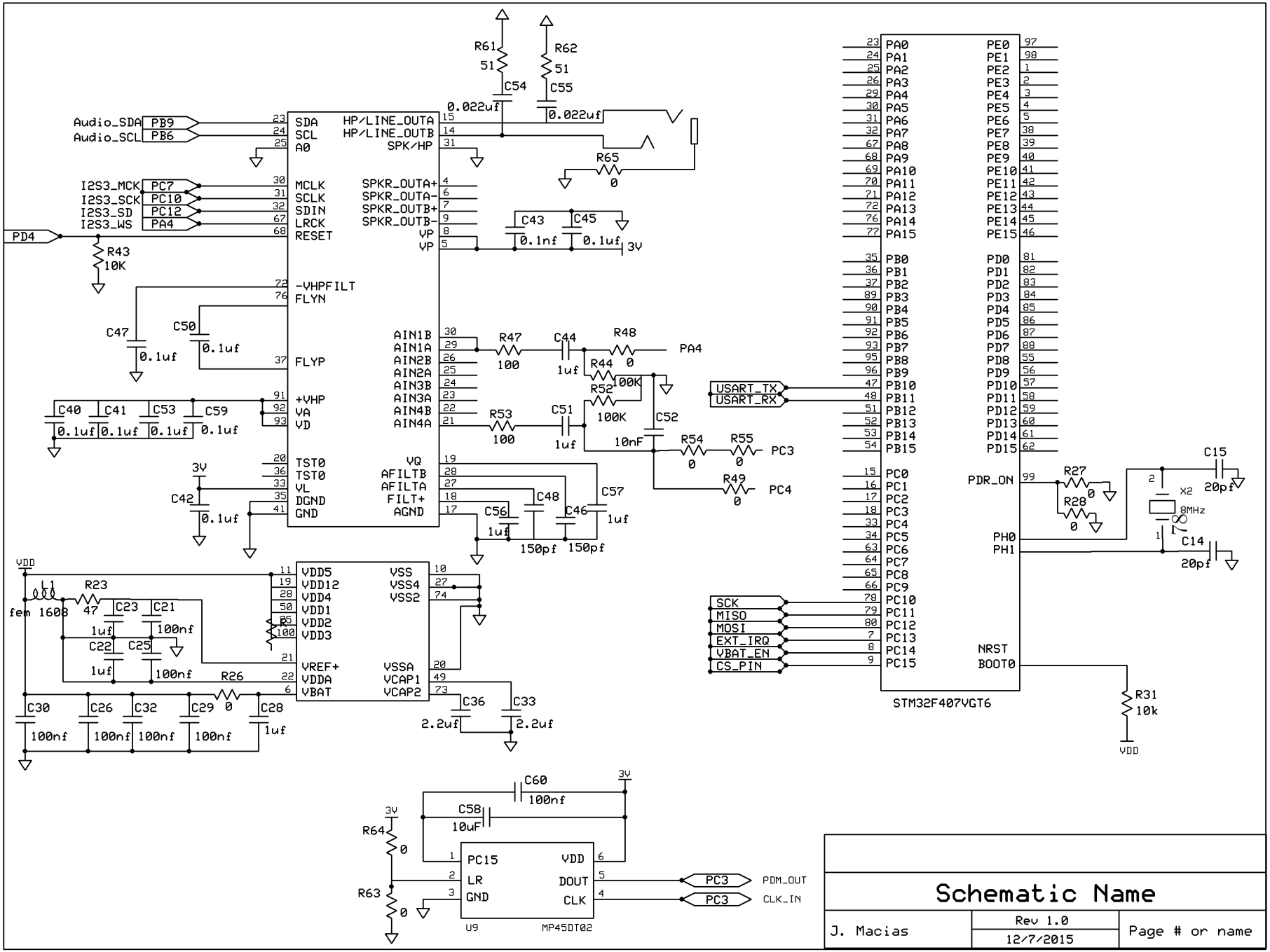
2.2.25 Inter-integrated sound (I2S) 34

2.2.26 Audio PLL (PLLI2S) 35

2.2.27 Secure digital input/output interface (SDIO) 35

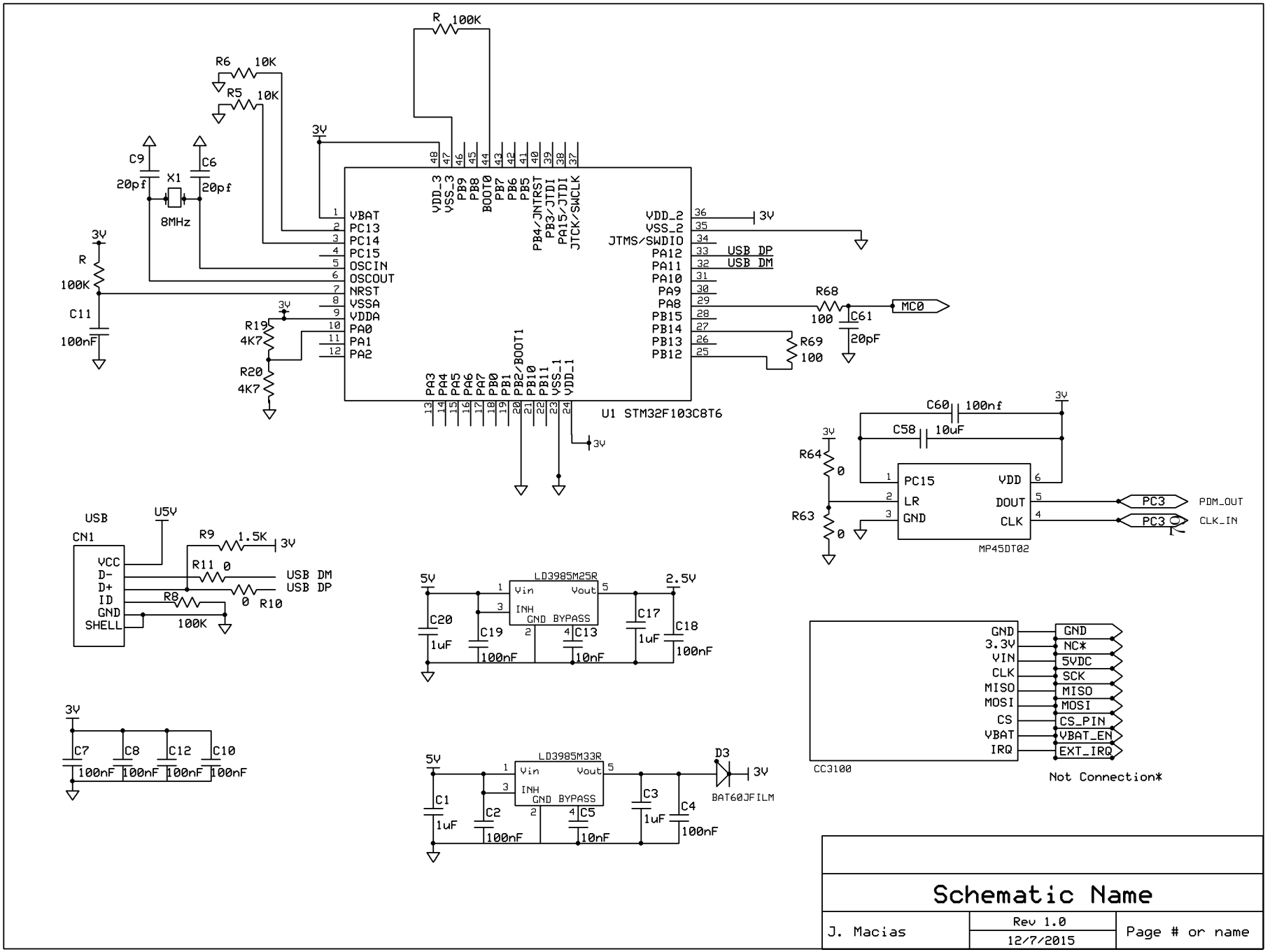
2.2.28 Ethernet MAC interface with dedicated DMA and IEEE 1588 support . 35

2.2.29 Controller area network (bxCAN) 36



Schematic Name

J. Macias	Rev 1.0 12/7/2015	Page # or name


R 100K
R6 10K
R5 10K
3V
C9 20pf
X1 8MHz
C6 20pf
R 100K
C11 100nF
R19 4K7
R20 4K7
U1 STM32F103C8T6
VBAT
PC13
PC14
PC15
OSCIN
OSCOUT
NRST
VSSA
VDDA
PA0
PA1
PA2
VDD_3
VSS_3
PB9
PB8
BOOT0
PB7
PB6
PB5
PB4/JNTRST
PB3/JTDI
PA15/JTDI
JTCK/SWCLK
VDD_2
VSS_2
JTMS/SWDIO
PA12
PA11
PA10
PA9
PA8
PB15
PB14
PB13
PB12
PA3
PA4
PA5
PA6
PA7
PB0
PB1
PB2/BOOT1
PB10
PB11
VSS_1
VDD_1
USB DP
USB DM
R68 100
C61 20pF
MC0
R69 100
C60 100nf
C58 10uF
R64 0
R63 0
PC15
LR
GND
VDD
DOUT
CLK
MP45DT02
PC3
PDM_OUT
CLK_IN
USB
CN1
VCC
D-
D+
ID
GND
SHELL
U5V
R9 1.5K
R11 0
R8
100K
0 R10
USB DM
USB DP
LD3985M25R
Vin
Vout
INH
GND BYPASS
5V
2.5V
C20 1uF
C19 100nF
C13 10nF
C17 1uF
C18 100nF
C7 100nF
C8 100nF
C12 100nF
C10 100nF
LD3985M33R
C1 1uF
C2 100nF
C5 10nF
C3 1uF
C4 100nF
D3
BAT60JFILM
GND
3.3V
VIN
CLK
MISO
MOSI
CS
VBAT
IRQ
CC3100
GND
NC*
5VDC
SCK
MISO
MOSI
CS_PIN
VBAT_EN
EXT_IRQ
Not Connection*
Schematic Name
J. Macias
Rev 1.0
12/7/2015
Page # or name