

CAPITULO III: DISEÑO DEL MICRÓFONO

3.1 CARACTERÍSTICAS BUSCADAS EN EL MICRÓFONO.

El problema radica en la necesidad de tener un micrófono que pueda enviar la información de manera digital por medio de una red de área local y la inexistencia en el mercado de un micrófono que comparta estas características.

La meta es diseñar un micrófono que pueda capturar los sonidos con calidad, que pueda muestrear y cuantizar el sonido, que guarde la información de 256 muestras para ser empaquetadas y finalmente que pueda enviar la información por medio de un cable de Ethernet a una dirección IP destino fija, la cual sería la dirección de la computadora donde se ubica el emulador de la consola de audio.

En el diseño del micrófono se va a dar solo un pequeño esbozo de los componentes que se piensan incluir y como estos trabajarán. Para poder hacer que el micrófono cumpla con su función se requieren solo cuatro cosas, un adaptador de red, un micro controlador el cual se va a encargar de las operaciones de memoria y hacer los paquetes para ser enviados a través de la red, un convertor analógico a digital y por último un micrófono de entrada.

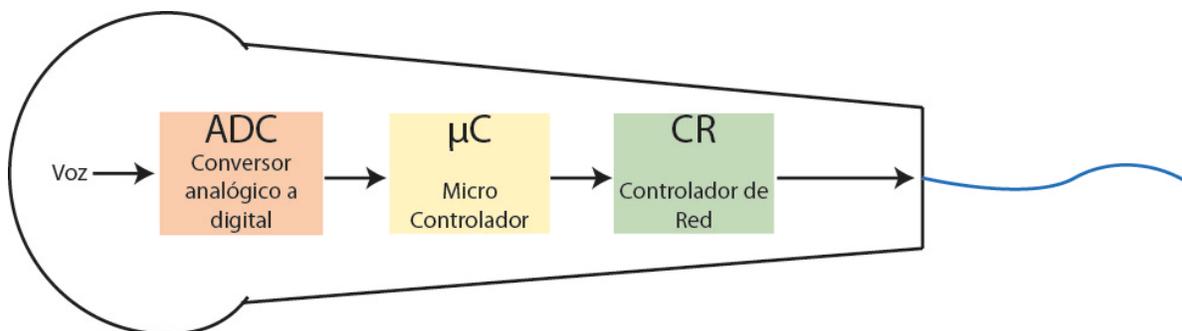


Figura 27. Componentes básicos buscados en el micrófono.

Dependiendo de la configuración final que tome el dispositivo, puede incluir dos o más partes en una sola, es decir, el micro controlador puede tener incluido el controlador de red o el convertor analógico a digital (o ambos). Para el diseño que vamos a hacer, acotaremos el tema a expresar los componentes básicos de forma independiente y explicaremos

por qué los utilizamos. No vamos a trabajar en la programación del micro controlador o de los componentes.

El micrófono se va a alimentar mediante el mismo cable de Ethernet, esto con el sistema llamado "Power over Ethernet" (PoE), para esto necesitamos un divisor de señal a la entrada del cable de red, de tal manera que los cables que transportan la corriente para el funcionamiento del micrófono sean desviados y sirvan como fuente de alimentación.

A la hora de diseñar un micrófono de capacidades inalámbricas, cambiaremos el controlador de Ethernet, por un adaptador de red inalámbrico y la alimentación se cambia a una pila incluida en algún lugar de la carcasa del micrófono.

3.2 COMPONENTES DEL MICRÓFONO

3.2.1 EL CONVERTIDOR ANALÓGICO A DIGITAL (ADC)

Este es un elemento clave en todos los sistemas digitales, ya que es el encargado de convertir la señal analógica a una señal digital, es decir, a una señal conformada por unos y ceros.

El proceso realizado por este convertidor es muy sencillo, lo único que hace es muestrear y cuantizar la muestras y tan solo tenemos que seguir unas reglas básicas para asegurar que la señal cuente con un buen nivel de calidad.

La información obtenida por este medio se envía sin codificar, ya que se envía la señal como PCM (pulse code modulation), tan solo es acomodada en paquetes para poder ser enviada por medio de una red que funciona en un estándar de paquetes IP.

Para captar la voz es necesario un micrófono, el cual va a ser el encargado de pasar las ondas mecánicas en el aire a impulsos eléctricos que serán manejados por el convertidor. Para esta etapa del proceso es necesario tener un micrófono confiable y buscar obtener de él una respuesta plana en frecuencia.

Por las características buscadas, la simplicidad y el precio se escogió un micrófono electret, ya que es fácil de encontrar, tiene un preamplificador incorporado, es sensible y el voltaje de alimentación puede oscilar entre los 2 y 20 volts, cosa que lo hace idóneo para nuestra aplicación.

Para obtener un nivel de la señal óptimo, es necesario introducir un etapa de amplificación antes de entrar al conversor analógico digital, esta amplificación se puede obtener con un amplificador operacional, ya que no necesitamos una gran corriente. El circuito de entrada ya se ha implementado para otras aplicaciones de voz por Texas Instruments¹⁵, de donde podemos obtener el diagrama de conexiones para el correcto funcionamiento de la captura de audio mostrado en la figura 28.

¹⁵ (Texas Instruments 2001)

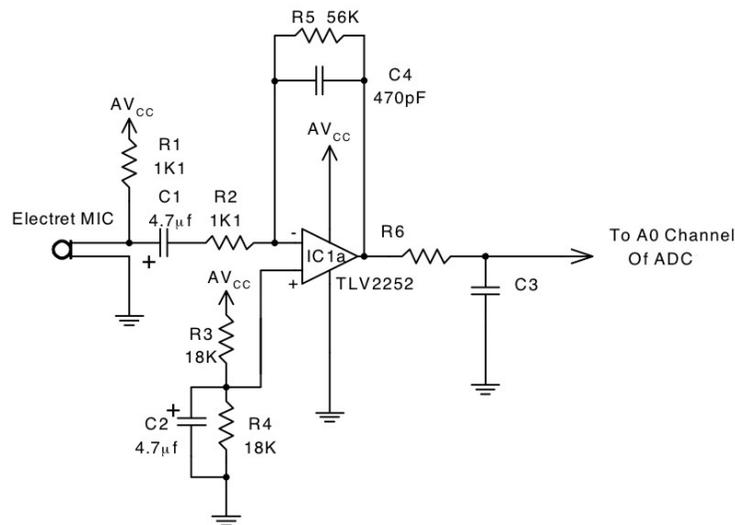


Figura 28. Circuito encargado de adecuar la voz para el ADC.

En el circuito viene integrado un pequeño filtro pasa bajas (conformado por R6 y C3), el cual lo utilizamos como filtro antialiasing antes de entrar al conversor analógico a digital y con esto evitar la pérdida de calidad de la señal. En nuestro caso el circuito estaría alimentado con los 5 volts que provee el sistema PoE.

Primer paso: El muestreo

Para este paso lo que nos importa es conocer el número de muestras que se tienen que tomar de la señal, este número se expresa en muestras por segundo. La importancia de tomar un número mayor o menor de muestras se va a la hora de recuperar la información, ya que si se escoge un número por debajo del número mínimo, se crea un efecto llamado "aliasing", el cual nos deja con información ambigua en la zona de traslape y hace imposible su utilización.

Lo primero a tomar en cuenta a la hora de escoger un convertidor para nuestro proyecto, es la velocidad de muestreo, ya que esto nos va a determinar las frecuencias a las cuales va a operar nuestro micrófono.

La decisión de escoger la frecuencia máxima a la cual queremos trabajar es importante en nuestro sistema, ya que esto determina la frecuencia a la que se pierde la calidad del audio; si transmitimos solo voz por el sistema, la frecuencia máxima puede ser tomada como 4 000 Hz, pero un gran

inconveniente es que solo la voz tendría una calidad aceptable y al transmitir sonidos mas agudos se daña el audio.

Los 4 000 Hz son un limite muy pequeño para nuestro sistema, ya que nos limita a usar el micrófono para la voz y lo imposibilita para captar sonidos mas agudos, como son los que producen los demás instrumentos, entonces al hacer un micrófono con esa limitante, solo daríamos servicio a una pequeña parte de todo el repertorio musical.

La solución entonces es elevar la frecuencia máxima, en este caso la frecuencia máxima la tomamos como 20 000 Hz, lo cual es el limite superior de audición humana.

Ahora comenzamos a trabajar para encontrar una frecuencia mínima de muestreo, así que vamos a utilizar el teorema de muestreo de la señal:

$$T_s \leq \frac{1}{2f_m} \quad (1)$$

Como sabemos que el inverso de la frecuencia es el periodo (T), podemos despejar a la ecuación (1) y obtenemos que:

$$f_s \leq 2f_m \quad (2)$$

Como podemos ver, la ecuación (2) nos indica que la frecuencia mínima de muestreo debe ser por lo menos del doble de la frecuencia máxima para evitar el aliasing, esto se conoce como el criterio de Nyquist¹⁶.

De esta manera sabemos que la frecuencia mínima para el sistema es de 40 000 Hz. El problema en este caso es que el criterio de Nyquist nos pone exactamente en la frontera, donde funciona bien o deja de hacerlo, lo cual le exige una precisión grande al conversor, aumentado el precio de este.

¹⁶ (Sklar 1988)

La solución a este problema es hacer un sobre muestreo (oversampling) de la señal, de esta manera trabajamos a una frecuencia de muestreo mayor a la requerida por el sistema de acuerdo a Nyquist, y tenemos mas muestras por segundo que las necesarias. La ventaja del sobre muestreo es que le permite a una señal alcanzar una calidad estipulada sin requerir mucha precisión de los componentes y de esta manera poder manejar componentes de menor precio¹⁷.

En la actualidad los sistemas de buena calidad (Blu-Ray, DVD, etc.) trabajan a frecuencias de muestreo de 96 000 Hz, de tal forma que se ha convertido en un estándar de la industria, por eso trabajaremos a esta frecuencia, además ciertas compañías que manufacturan ADC han demostrado una mayor calidad de sonido de conversores que corren a 96 Khz¹⁶.

Entonces debemos de buscar en el mercado un conversor que funcione para nuestro sistema que corra a esa velocidad.

Segundo paso: La cuantización

Una vez que tenemos las muestras, el siguiente paso es darle un valor numérico a la amplitud de las muestras y expresar ese valor de forma digital, es la acción de ponerle valores discretos a un atributo variable que puede tomar una infinidad de valores.

El trabajo del cuantizador es colocar las muestras obtenidas en un intervalo el cual tiene asignado un valor fijo, de esta manera se obtiene un número fijo de valores para la muestra. El error de cuantización se define como la diferencia entre el valor asignado y el valor real de la muestra, este error se da generalmente como una relación señal ruido de cuantización (SNR_Q). Como se puede intuir, se obtiene una mayor calidad de audio al poner mas niveles de cuantización en el sistema.

¹⁷ (Watkinson 2001)

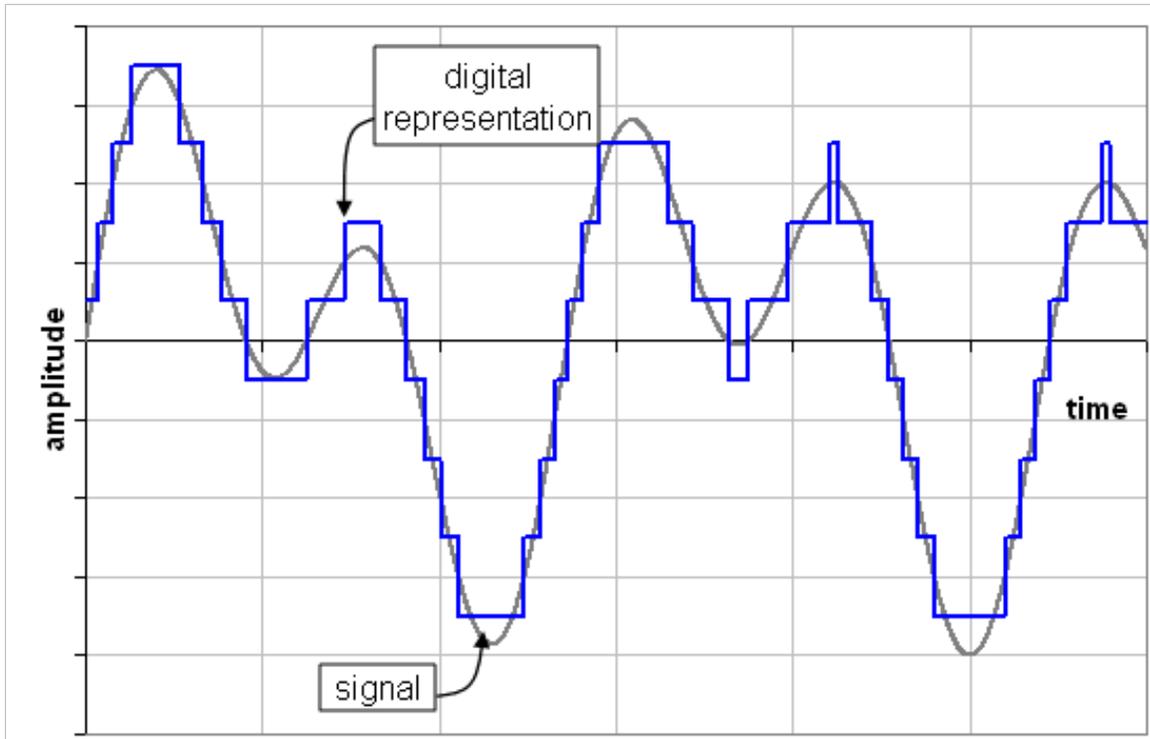


Figura 29. Realización y errores en el proceso de cuantización.

En la figura 28 se puede ver la suma del efecto de muestreo, que crea las mesetas de la representación digital de la señal, y los efectos de la cuantización de la señal, que es la parte encargada de darle ciertos valores discretos a la señal a lo largo de la amplitud.

El número de niveles de cuantización está dado por el número de bits que se utilizan para representarlos, de esta manera si tenemos 16 bits tendremos 2^{16} (65 536) niveles disponibles para nuestro sistema.

En la industria se manejaban estándares de 16 bits, como es el caso del Compact Disk, pero esta configuración presenta problemas cuando el sonido tiene niveles muy bajos y muy altos de sonido, al existir cambios tan drásticos de amplitud, se presentan problemas para ajustar el sonido a los niveles dados por los 16 bits sin introducir mucho ruido, lo que hizo necesario la implementación de sistemas de 24 bits, es por esta razón que nuestro sistema contará con un ADC de 24 bits.

Ahora que ya tenemos un perfil del producto que necesitamos, el cual trabaje con una frecuencia de muestreo de 96 KHz y cada muestra con un largo de 24 bits, podemos buscar un circuito integrado que se adapte a

nuestras necesidades, para esto buscamos e internet ADC que se ajusten a los requerimientos y encontramos uno con las siguientes características:

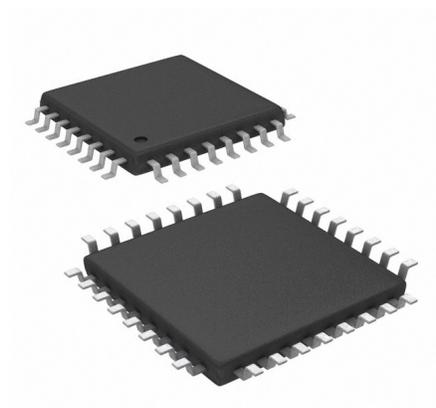


Figura 30. Chip ADC de Texas Instrument

Tabla 1. Características del chip a utilizar.

Referencia	PCM1851APJTR
Marca	Texas Instruments
Categoría	Integrated Circuits (ICs)
Número de bits	24
Empaque	32-TQFP, 32-VQFP
Interface de datos	I ² C, Serial
Almacenamiento	Tape & Reel (TR)
Tasa de muestreo (por segundo)	96k
Temperatura de operación	-40°C ~ 85°C
Fuente de voltaje	Analógica y digital
Cumplimiento de plomo	Libre de plomo
Cumplimiento RoHS	Cumple con RoHS

Tenemos la ventaja de que el chip maneja varias interfaces de datos, lo que nos da la posibilidad de conectarlo casi cualquier micro controlador, para nuestro micrófono obtendremos los datos de manera serial y lo conectaremos directamente al BIU (Bus Interface Unit) de nuestro microcontrolador, otra cosa agradable es el precio, ya que la unidad solo cuesta 3.43 dólares¹⁸.

3.2.2 EL MICRO CONTROLADOR

En la actualidad los micro controladores se utilizan como el cerebro de un sin fin de dispositivos, desde módems y teléfonos hasta relojes y osciloscopios. Esto demuestra su gran versatilidad y nos da una idea de la posibilidad de colocarlo en el micrófono.

La idea de tener un micro controlador surge para resolver el problema del manejo de los datos provenientes del ADC.

Para poder enviar los datos a través de la red, toca agregar encabezados y empaquetar la información para que pueda llegar con éxito a su destino, así que necesitamos un dispositivo capaz de guardar la información suficiente para crear un paquete y después empaquetarla (agregando encabezados) y enviarla por medio del adaptador de red.

¹⁸ (Digi-Key Corporation 2011)

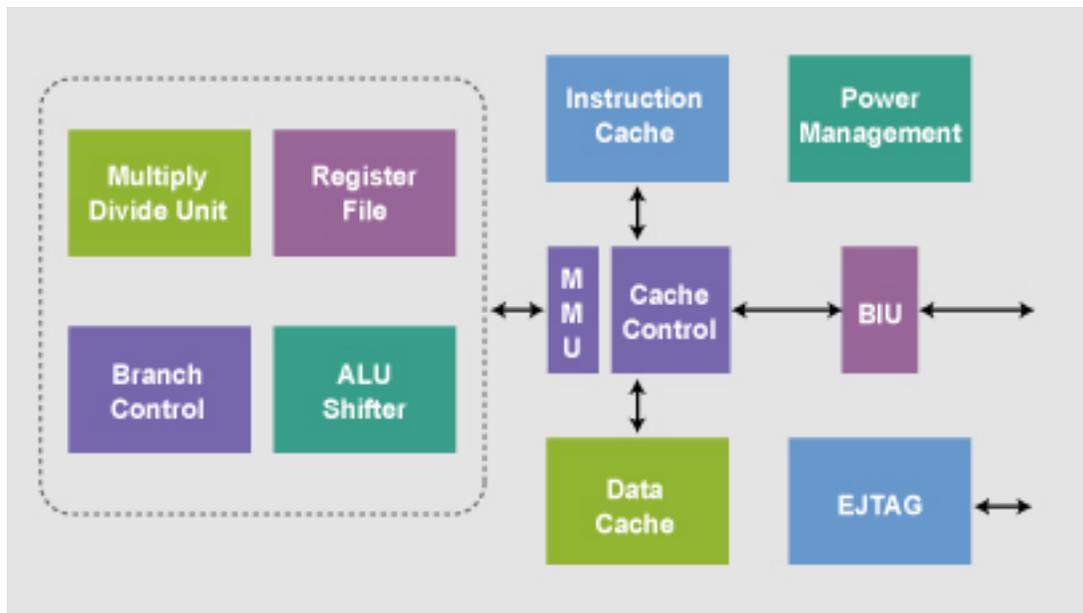


Figura 31. Componentes del micro controlador.

Los encabezados que vamos a trabajar pueden ser fijos o dinámicos, en el primer caso se llenan los espacios del encabezado con información predeterminada que permita su correcto envío. Para el segundo caso se puede agregar códigos de detección de errores, aunque es poco práctico, ya que al tratarse de la voz, no se pide una retransmisión del paquete.

Los datos se tienen que almacenar hasta llenar un paquete, en el capítulo 2 se propone que los paquetes estén conformados por 256 muestras, así que la memoria debe ser más grande que el tamaño de cada paquete, de esta manera necesitamos una memoria mayor a 6 144 bits (esta memoria solo es para los datos de audio, cualquier dato necesario para la ejecución del programa debe considerar un espacio de memoria extra).

La memoria con la que trabajan los microprocesadores pueden ser de tres tipos, una memoria ROM para el almacenamiento del programa (esta posiblemente sea una EEPROM por facilidades técnicas); una memoria RAM, donde se guardan las variables con la que trabajaremos, es en esta clase de memoria donde se guardaría la información de las muestras de audio para su empaquetamiento; y por último, una memoria EEPROM para el almacenamiento de las variables que no queremos que sean borradas ante la falta de suministro eléctrico (en este caso entraría la dirección IP del dispositivo y otras variables fijas). La mayoría de los micro procesadores ya tienen integrada alguna cantidad de memoria para el uso del programa, solo es necesario ver si el tamaño satisface nuestras

necesidades, en caso contrario se cuenta con puertos de expansión de memoria¹⁹.

Existen algunos micro controladores que ya incluyen un convertidor analógico digital, esto pueden ser bastante útiles, ya que nos ahorran un etapa del sistema descrito. El problema con estos convertidores incluidos es que no alcanzan la calidad que buscamos, como vimos queremos un sistema con una calidad espectacular y para ello necesitamos un ADC con una alta frecuencia de muestreo y una buena cantidad de bits por muestra, lamentablemente los sistemas actuales no nos brindan estas características dentro del micro controlador.

Ahora lo que buscamos es un procesador capaz de empaquetar los datos, para nuestro caso vamos a utilizar encabezados fijos, de esta manera la programación se simplifica y la carga de trabajo del microprocesador es mucho mas ligera, de esta manera podemos ajustarnos a un micro sencillo y bajar así los costos del sistema.

Dentro de esta mecánica del micro sencillo, vale la pena explicar por qué se escogió enviar la información sin codificar: Al codificar los datos se puede ahorrar ancho de banda, pero la operación de codificación esta hecha a base de operaciones matemáticas y transformadas, de esta manera se requiere cierto poder de procesamiento dentro del micrófono y de la computadora. La preocupación principal es dentro de la computadora, ya que ahí la información de los diversos canales se tiene que decodificar y presentar con calidad, cosa que se puede hacer bien para un número pequeño de canales, pero se complica demasiado para un numero alto de canales, pudiendo presentar retrasos grandes en el procesamiento y con originar un tiempo de retraso total del sistema mas alto de los valores aceptables.

Es por esto que se elige enviar la información sin codificar, ya que tenemos el ancho de banda suficiente y guardamos el poder de procesamiento de la consola para poder manejar mas canales de audio.

Sin embargo, esta puede ser una característica extra del sistema y se puede diseñar el micrófono con un micro capaz de hacer el procesamiento necesario para codificar la información y así poder trabajar con anchos de banda mas pequeños, útiles para redes inalámbricas donde el ancho de banda esta mas limitado.

¹⁹ (Peatman 1988)

A continuación elegimos salir a la búsqueda de un micro que nos convenga, ya que uno demasiado grande además de elevar el precio, saldría desperdiciado y uno muy pequeño, puede que no nos cubra la carga de trabajo que necesitamos imponerle. De esta manera buscamos en internet (ya que los libros siempre están un poco atrasados al respecto) y encontramos microprocesadores de toda clase, por lo que refinamos la búsqueda y elegimos usar micros pequeños y de 32 bits, ya que algunas operaciones se pueden simplificar.

El escogido es un micro procesador marca MIPS, de referencia 4Ktm core, el cual nos ofrece las siguientes ventajas:

- Hecho con tecnología de silicio, probada con cientos de implementaciones exitosas reduce el tiempo de lanzamiento al mercado y aumenta el éxito de primer paso.
- Basado en la arquitectura MIPS32 permite aplicaciones de vanguardia, con su alto rendimiento de 32 bits, de 5 etapas de pipelines.
- Sintetizable, configurable y portátil, incrementa la flexibilidad del sistema y la longevidad.
- Diseñada para facilitar la integración de SoC con un solo reloj, totalmente estático, diseño basado en flops.
- Rápido, un multiplicador de ciclo 32x16, soporta las funcionalidades básicas de DSP.
- EJTAG opcional, que simplifica y acorta el tiempo de depuración y de salida al mercado.
- Compatible con las tecnologías MIPS de 64 bits, lo que permite futuras actualizaciones del sistema.
- Con el apoyo de cientos de herramientas de desarrollo de terceras partes, software y aplicaciones para agilizar el tiempo de lanzamiento al mercado y un ciclo de diseño acertado.

El tamaño es importante al tratarse de un micrófono, ya que no podemos empaquetar un micro procesador que rebase las dimensiones del micrófono. De igual manera la potencia necesitada por el micro controlador es importante, ya que a mayor potencia, mayor calentamiento, cosa contraproducente en el micrófono, por esta razón nos fijamos en los datos suministrados por el fabricante para estos valores y podemos constatar que están en muy buenos rangos, ya que el consumo de potencia va de 1.3 a 2.2 mW/MHz (al requerir poco procesamiento, menor son los ciclos de reloj, necesitando menos potencia) y el área que

ocupa el chip es tan solo de 1,4 mm², constatando de esta manera que si es posible su uso para el lograr el fin buscado²⁰.

Dado que el uso que se le da al micro controlador es bastante pequeño (solo organiza memoria y empaqueta la información) se requiere de poca velocidad para lograr este fin, logrando un buen funcionamiento con un estimado de 1 MHz y lograr con esto un consumo energético bastante bajo.

Con este micro procesador queda totalmente cubierta nuestra necesidad del procesamiento de datos, ya sea solo empaquetando los datos sin comprimir o haciendo el procesamiento necesario para poder comprimir los datos y su respectivo empaquetamiento posterior.

3.2.3 EL CONTROLADOR DE RED

Tratándose de un sistema tan pequeño que tiene que caber en un micrófono, es importante poder tener componentes pequeños y de preferencia que puedan estar en la misma tarjeta que el resto de componentes, es por eso que descartamos el uso de una tarjeta de red del sistema y nos inclinamos a poner un controlador de red, el cual esta compuesto por un solo componente y puede ser conectado directamente a el puerto Ethernet del dispositivo.

Para poder lograr este objetivo tenemos el controlador RTL8019AS, de la marca Realtek, este dispositivo es un controlador de Ethernet Full-Duplex con la funcionalidad Plug and Play. Es un dispositivo que nos brinda muchas opciones de configuración y opciones de uso, cosa que vamos a desaprovechar un poco, ya que solo lo vamos a usar con un fin y su configuración será estática.

Este dispositivo nos ofrece un controlador integrado de Ethernet, el cual ofrece una solución simple a la implementación de un adaptador full dúplex y con funciones de energía, ya que tiene un control de tres niveles para el ahorro de energía y lo convierte en una buena elección para crear dispositivos energéticamente mas eficientes. Con la función de full dúplex se establece el envío y recepción de datos al mismo tiempo, cosa que vamos a desaprovechar, ya que la información que maneja el micrófono

²⁰ (MIPS Technologies 2011)

solo va hacia la consola y la información de regreso es nula. Aunque esta diseñado para su uso en computadores, también admite el uso de “jumpers” para su configuración, esta es la modalidad que vamos a utilizar nosotros, al no contar con un software en el micrófono encargado de la configuración del adaptador de red²¹.

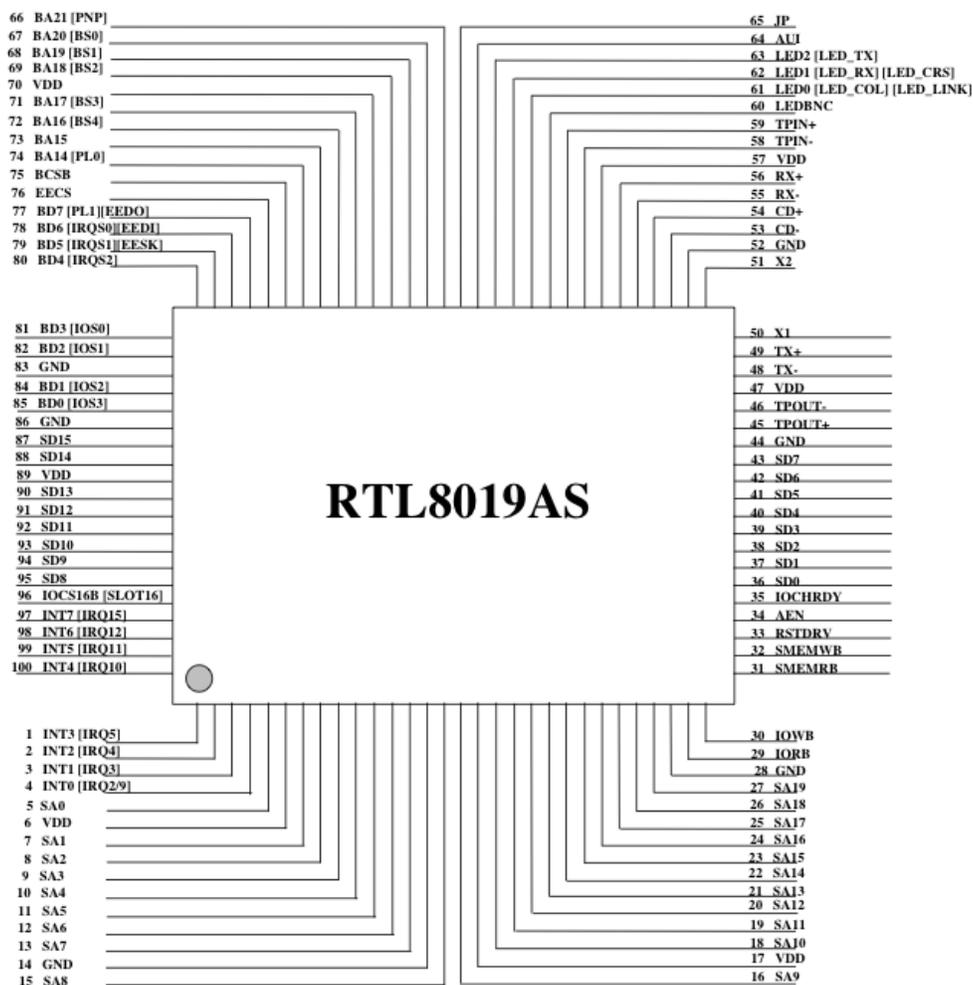


Figura 32. Patigrama del controlador de red Realtek RTL8019AS.

Aunque la configuración de este controlador puede verse un poco difícil, ya que tiene mas de 100 entradas y salidas, en realidad es sencilla, ya que varias de las entradas son conectadas a tierra o al voltaje de alimentación (12 en total) y muchas otras se van a dejar fijas, ya que las opciones que se

²¹ (Realtek 2001)

manejan con estas entradas, no nos resultan útiles para el fin buscado. También tiene la posibilidad de manejar leds que nos indican el correcto funcionamiento del sistema, estos leds pueden ser colocados en ventanas hacia el exterior del micrófono, esto con el fin de saber si el micrófono está bien conectado.

Los pines importantes para nosotros son donde salen y entran los datos, tanto hacia el micro procesador, como hacia la red de área local. Los pines (se envía uno con los datos y otro con tierra como referencia) donde la información se envía hacia la red son 45 y 46, mientras que se recibe la información por el 59 y 58. La información que es recibida se envía por los pines 56 y 55 y la información que se desea transmitir se envía al pin 49 y 48²⁰. La información viene directamente del micro controlador en forma serial, por medio de la unidad de interfaces de bus (BIU) de este último.

Las demás entradas y salidas se van a dejar en estado por default, algunas libres y otras conectadas a tierra o al voltaje positivo, de esta manera se dejan sin fijar ciertas características, como la dirección física, ciertas operaciones con la memoria y capacidades de reseteo del sistema.

Para el caso de una conexión inalámbrica del micrófono, es necesario sustituir toda esta parte por un adaptador de red inalámbrico, de esta manera la información que llega del micro procesador es enviada por un controlador de red inalámbrico hasta el punto de acceso, al llegar al punto de acceso la información es enviada por cable hasta la consola, como está planeado en el capítulo de la red de área local.

3.3 USO DE "POWER OVER ETHERNET" (POE) PARA LA ALIMENTACIÓN ELÉCTRICA A DISTANCIA DE LOS MICRÓFONOS.

Todos los procedimientos que hemos vistos que se realizan en el micrófono necesitan de una fuente de alimentación para funcionar, de esta manera es imperativo tener una fuente de alimentación, ya sea en cada micrófono o una general para todos los micrófonos conectados.

El problema radica a la hora de adjuntar la alimentación al sistema, ya que una batería dentro del sistema significaría tener un límite de tiempo para la utilización del micrófono y por otro lado el uso de un cargador. Además para que se quiere tener batería si de todas formas el micrófono ya tiene un cable.

Otra manera de conectarlo sería por medio de un eliminador de baterías, pero en este caso se requiere de otro cable para la alimentación eléctrica del micrófono.

La solución entonces es alimentar al dispositivo por medio del mismo cable por el cual se transportan los datos, solución que ya se utiliza para poder alimentar puntos de acceso inalámbrico sin necesidad de conectarla a una fuente eléctrica en el lugar de su instalación. Esta solución se llama "Power over Ethernet" o simplemente PoE.

Al utilizar esta tecnología la energía necesaria para el uso del micrófono se envía por un par sobrante del cable UTP (ya que solo se utilizan dos pares para el envío y recepción, dejando libre otros dos) y la energía llega al micrófono sin contratiempos.

La parte del diseño del micrófono para adecuarlo a este sistema es sencilla, ya que el par de cables por el cual viaja la corriente son separados al momento de llegar a la fuente y los cables de datos son conectados al controlador de red del micrófono.

Donde se introduce la corriente al cable es la parte interesante de este diseño, ya que venden dispositivos que sirven como fuentes de alimentación o switchs con la capacidad de enviar la corriente, lamentablemente estos switchs solo son de ocho puertos y de velocidades de 100 Mbps, de tal forma que son un poco pequeños para lo que estamos buscando en el diseño²².

²² (TP-LINK 2011)

La solución entonces es hacer nosotros mismos la fuente de voltaje, donde lo único que necesitamos es separar el par de cables por donde se va a enviar la corriente al dispositivo, antes de que llegue al switch y con una fuente de poder alimentar el dispositivo separador, para que de esa manera se alimenten los micrófonos necesarios sin necesidad de gastos extras.

La fuente de voltaje no necesita un amperaje grande, ya que la potencia eléctrica que necesitamos en el micrófono es bastante pequeña, dado que la corriente que circula en el dispositivo no llega a 100 mA

En este caso recordamos que la información de acuerdo con la norma TIA/EIA-568-B.1-2001, donde se establece el orden de los cables, se envía por los pines 1 y 2 y se recibe por los pines 3 y 6 (en el caso del switch se cambian estos valores para permitir el uso de un cable sencillo), por lo que solo se utilizan los cables verdes y los cables naranjas, donde cada par lleva la información en el sentido contrario. De esta manera podemos elegir los otros cables para el envío de corriente a través de ellos.

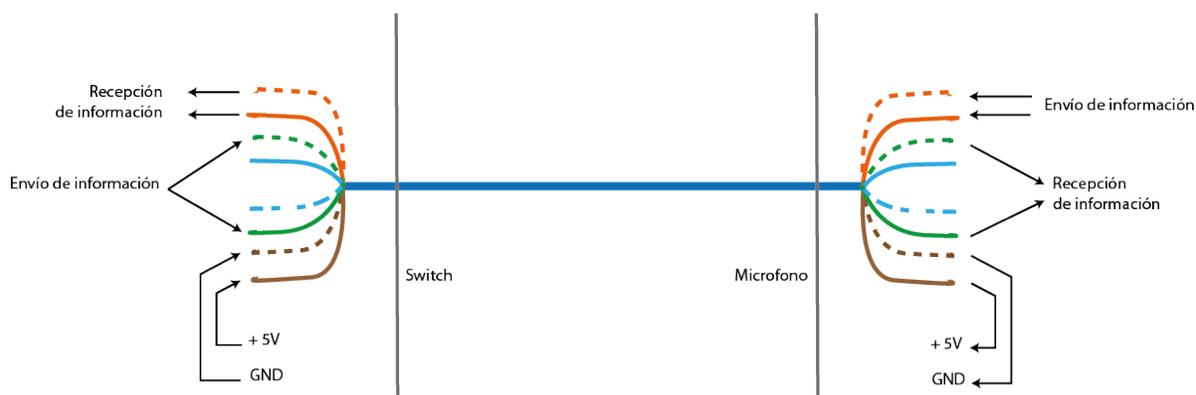


Figura 33. Envío y recepción de la información y la corriente por el cable UTP.

En este punto solo hace falta desviar los cables cafés del controlador de red del micrófono y del switch para evitar daños y conectarlo a la fuente de alimentación en el lado del switch y usar las terminales en el micrófono como tierra y el voltaje positivo.

Esta característica estaría deshabilitada para la conexión inalámbrica de los micrófonos, ya que al alimentar al micrófono por un cable dejaría de ser inalámbrico. Para solucionar este problema, se plantea el uso de pilas para alimentar eléctricamente al micrófono y tener la funcionalidad de poder

mover el micrófono por donde haya cobertura de la red inalámbrica, sin riesgo de perder la calidad del audio.