21
2ej

Universidad Nacional Autónoma de México

Facultad de Ingeniería

CRIPTOGRAFIA PARA TRANSMISION DE VOZ

T E S I S

para obtener el título de:

INGENIERO EN COMPUTACION

P r e s e n t a n :

Adrián Martínez Aguilar

y

Raúl Abraham Sánchez Sánchez

Director de Tesis: Dr. Federico Kuhlmann Rodríguez

México, D. F. 1988

UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

I N D I C E

PAGINA

introducción.

> ..... y es dudoso que un ingenio humano pueda llegar a construir un enigma de esta especie, que otro ingenio humano no consiga descifrar adecuadamente.....
>
> "EL ESCARABAJO DE ORO". EDGAR ALLAN POE.

El mundo está entrando a un nuevo período: la riqueza de las naciones, que en ésta su fase industrial, depende de los recursos naturales, de la acumulación de capital e incluso de las armas entre otras cosas, dependerá en el futuro de la información y de su uso adecuado. La información se está convirtiendo en un recurso tan estratégico como el petróleo y constituirá la nueva riqueza de las naciones, de tal forma que la seguridad sobre la información repercutirá sustancialmente en el desarrollo de un país.

Tal vez este pronóstico, (proveniente de los japoneses), sea un tanto engañoso, pero lo cierto es que día a día la información va adquiriendo mayor importancia, ya que ésta representa ventajas estratégicas; la persona que tiene información importante, de manera confiable y privada, tiene poder. Para darle privacía y confiabilidad a la información se utiliza la "Criptografía".

Día a día y gracias a las investigaciones sobre criptografía, en los paises industrializados, se cuenta con sistemas seguros y complejos de información, ya que existe una contienda entre los investigadores que pretenden asegurar la privacía de la información.

Para ilustrar esta aseveración, considérese el esquema propuesto por Martin Hellman, Merkle y Diffie de la universidad de Stanford: diseñaron un sistema de dos claves y anunciaron que su sistema era indescifrable ya que se requería de una clave para el cifrado y otra para el descifrado. Compararon la solución del problema con una mochila llena de cajas; aunque el descifrador conozca el volúmen total de la misma, no tiene ninguna posibilidad de saber cuantas cajas caben en su interior, aunque conozca el volúmen de una o de varias cajas. Pronosticaron millones de años para que se pudiese descifrar el contenido, pero en solo seis años el matemático Israelí Adi Shamir descubrió la clave.

Así pues, el desarrollo de la criptografía en los sistemas de comunicaciones, ha ido en aumento día a día en los países industrializados.

El objetivo de esta tesis es mostrar las simulaciones de varios sistemas criptográficos para la transmisión de voz analógica por vía telefónica. De los sistemas utilizados, algunos son sistemas ya existentes y otros son el resultado de combinaciones de aquéllos. Además, se presenta el software de estos métodos en lenguaje ensamblador del microprocesador TMS32010 y se sugiere la arquitectura requerida para la implementación de un sistema criptográfico, basado en el TMS32010, que hace uso del software mencionado.

La estructura de este trabajo, que refleja asimismo la metodología utilizada, se presenta a continuación.

En el capítulo uno se presentan los aspectos básicos sobre la generación de los distintos sonidos que conforman la voz en el ser humano. Es importante conocer estos aspectos ya que los mensajes a encifrar en el sistema criptográfico son de voz.

En el capítulo dos se presentan las generalidades sobre la digitalización de la voz. A lo largo de este trabajo podrán observarse las ventajas que tiene digitalizar las señales analógicas para su proceso.

En el capítulo tres mostramos la teoría general sobre la criptografía digital de datos. En este tipo de criptografía residen los principios de la criptografía digital de señales de voz analógicas.

En el capítulo cuatro presentamos las principales técnicas de criptografía digital de voz, tanto el dominio del tiempo como en el dominio de la frecuencia.

En el capítulo cinco mostramos las simulaciones realizadas, en un principio en una computadora PC-XT y posteriormente en el módulo de evaluación para el microprocesador TMS-32010, en esta parte mostramos las comparaciones entre los diversos métodos criptográficos y las ventajas que proporcionan cada uno.

Después de las simulaciones en PC-XT se presenta, a manera de introducción, las características del módulo de evaluación del microprocesador TMS32010 y de la tarjeta de interfase analógica, mediante estos recursos probamos los métodos criptográficos y presentamos la arquitectura sugerida para el sistema

criptográfico.

Por último, mostramos las conclusiones, las perspectivas y las consideraciones finales de la criptografía.

Es importante hacer notar que en la literatura sobre criptografía de señales analógicas se le hace referencia como scramblers a éste tipo de métodos criptográficos.

I.1.- Introducción.

El medio natural de comunicación del ser humano es la voz. El hombre, a través de su evolución, ha sido capaz de desarrollar métodos de codificación y decodificación para transmitir información (cualquier lenguaje oral, de hecho, es un proceso de codificación). Con el desarrollo de la electrónica y de las telecomunicaciones, se ha hecho de la transmisión a distancia de la voz un medio relativamente fácil de comunicación entre los hombres. Y con la aparición de la informática se han podido estudiar ciertos fenómenos que antes eran prácticamente imposibles a acceder.

El hombre trata de comprenderse y de adaptar las máquinas a sí mismo, para lograr de ésta forma una comunicación más natural y directa. Siendo pues la voz el medio natural de comunicación entre los hombres, parece simple el tratar de crear sistemas capaces de reproducirla.

La comunicación hombre-máquina con voz era hasta hace poco tiempo una utopía. Actualmente ya no es así, y el aprendizaje a este respecto aunque ha avanzado mucho, aún falta también mucho por conocer y desarrollar. El estudio de la voz incluye muchas disciplinas y presupone el análisis de funciones complejas tales como la percepción, el aprendizaje, la memoria y la inteligencia. Requiere del esfuerzo y la colaboración de lingüistas, médicos, ingenieros en comunicaciones, electrónica, computación,

psicólogos, profesores de lenguas vivas y muchos otros investigadores más.

Los conocimientos acumulados por cada disciplina son vastos y algunas veces contradictorios; el lenguaje difiere de una disciplina a otra. El intentar conocer todo respecto a la comunicación con voz sería en estos momentos imposible.

I.2.- Mecanismos de generación de voz desde el punto de vista físico.

El elemento básico del lenguaje hablado es el fonema, la fonación se refiere a la emisión de fonemas, es decir, a la generación de voz.

La fonación no es realizada por medio de un aparato u órgano específico; aparte de la laringe, los órganos que se utilizan son afectados por otras funciones vitales, como la respiración y la nutrición.

Las cuerdas vocales, fig.(1.1), entran en vibración cuando hay un excedente de presión por abajo de la glotis. Muchas teorías se han establecido desde el siglo XVIII, pero los medios de investigación modernos, y muy especialmente la cineradiografía, han permitido establecer que las cuerdas vocales vibran en toda su altura y no solamente en el plano horizontal: parecen ser el conjunto de las ondas de superficie que nacen dentro del cono elástico y se desplazan progresivamente hacia abajo. Las cuerdas

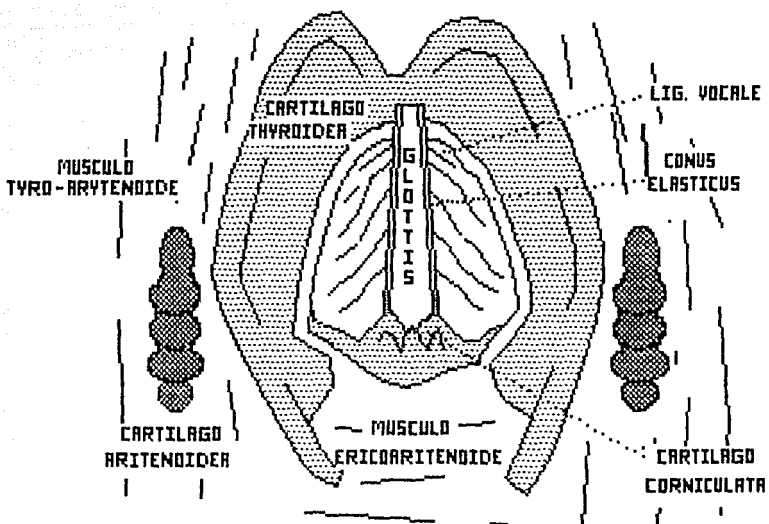


FIG. (1.1) CONO ELASTICO, LIGAMENTOS VOCALES, CARTILAGOS Y MUSCULOS DE LA LARINGE.

vocales son aplicadas una contra la otra sobre todo el espesor. La presión sub-glótica se aplica igualmente sobre todas las paredes del cono elástico, fig.(1.1), y tienden a separar las cuerdas vocales al mismo tiempo que las deforma hacia arriba. Las cuerdas vocales acaban por separarse bruscamente dejando pasar el aire; pero ese desplazamiento de aire produce una fuerza de

contracción sobre las paredes del cono elástico que se suma a la elasticidad de los músculos vocales.

Las cuerdas entran de nuevo en contacto, en su parte inferior, y el ciclo puede volver a empezar.

La presión acústica supraglótica sigue sencillamente la apertura de la glotis, que puede asociarse a una función triangular disimétrica.

Este modo de funcionamiento genera fenómenos observables, como el defasamiento existente dentro de la vibración de las partes superiores e inferiores de las cuerdas vocales. Esto no implica necesariamente un cierre absoluto de la glotis; en caso de una unión imperfecta de las cuerdas vocales, las fuerzas aerodinámicas y elásticas que inicialmente produjeron la vibración continúan existiendo. La onda acústica emitida tiende a una forma más simétrica, cercana a una senoide, al mismo tiempo que aparece una componente continua de la abertura. El resultado en el plan acústico es un empobrecimiento del timbre, debido a que el espectro contiene menos armónicas y un enriquecimiento en ruido debido a la fuga turbulenta de aire dentro de la laringe.

La disposición de los cartílagos y de los músculos de la laringe, fig.(1.1), permiten variar la tensión, alargamiento y separación de las cuerdas vocales, las cuales, en función de la presión subglótica, pueden adoptar una infinidad de puntos de funcionamiento.

Cuando se respira sin fonación, las cuerdas vocales son separadas al máximo.

La voz de pecho es obtenida al aproximar los arytenoides, por medio de los sistemas musculares disponibles, y contraer los músculos tyro-arytenoides que forman el cuerpo de las cuerdas vocales. Para pasar a un registro medio se disminuye la contracción de las cuerdas vocales y se aumenta la longitud bajando el cartílago tyroide.

En la voz de falsete, las cuerdas vocales son estiradas al máximo, pero vibran solamente en una parte de su longitud (parte ligamentosa o anterior).

La voz "grandote", es obtenida con ajustes como los anteriores, aparte de la relajación de los músculos ericoaritenoides, lo que permite a las cuerdas vocales, estando tensas, comprimirse no tan fuertemente y dejar pasar aire por una ventana delgada sin entrar en vibración.

I.3.- Tipos de sonidos que forman la voz.

El elemento básico del lenguaje hablado es el fonema. Las variaciones distinguibles se llaman alófonos.

Los fonemas pueden considerarse como un código relacionado únicamente a los gestos articulados de un lenguaje.

Los alófonos de un fonema pueden considerarse representativos de la libertad acústica permitida al especificar el símbolo

codificado. Esta libertad no depende únicamente del fonema sino también de la posición de una uterancia.

Se busca la clasificación de distintos sonidos y grupos acústicamente distingibles pertenecientes al mismo fonema, o que no son de distinto significado. Los alófonos difieren en pronunciación, pero esta diferencia no es importante dado el punto de vista semántico del lenguaje, (un gesto basta para distinguir un fonema).

El alfabeto de la asociación fonética internacional (IPA) proporciona símbolos para representar sonidos de voz de la mayoría de las lenguas del mundo. La clasificación de sonidos de voz se hace de acuerdo a la manera y lugar de la producción. Por ejemplo, la articulación de sonidos vocálicos se describe por la posición de la lengua a lo largo del conducto vocal, fig.(1.2), y el grado de constricción del mismo.

A).-Vocales.

Las vocales son producto exclusivamente de la excitación de las cuerdas vocales del conducto vocal.

En condiciones normales, el conducto se mantiene en una configuración relativamente estable durante la mayor parte de la duración del sonido. Las vocales se caracterizan por un acoplamiento nasal despreciable y por la radiación de la boca (excepto aquélla que pasa por las paredes de las cavidades).

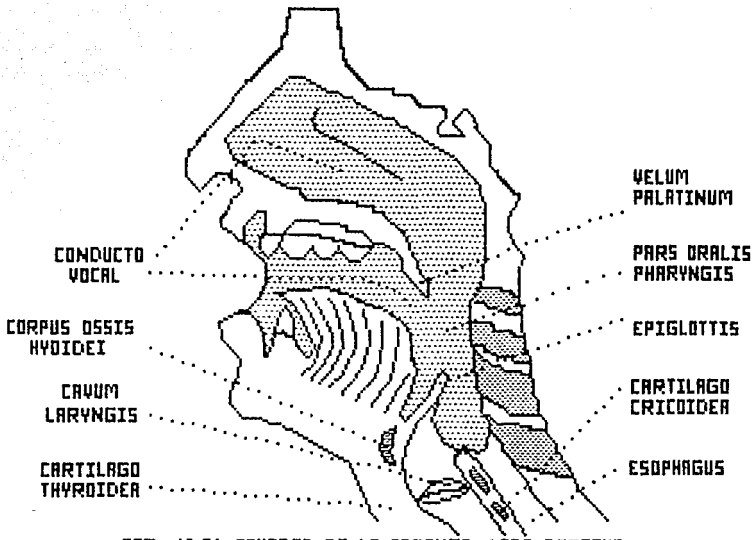


FIG. (1.2). CAVIDAD DE LA FARINGE, LADO DERECHO; ASPECTO INTERIOR.

La producción de sonidos vocalizados al mínimo nivel posible, requieren una presión pulmonar del orden de 4cm.de agua. Para sonidos fuertes, la presión es de aproximadamente 20cm. de agua. Los sonidos vocalizados son producidos por la acción vibratoria de las cuerdas vocales.

Imaginemos una masa relativa, las cuerdas vocales tensas inicialmente juntas, la presión subglótica se incrementa suficientemente para forzarlas con una aceleración lateral. El flujo de aire pasa a través del orificio, la presión local se reduce y una fuerza actúa para regresar las cuerdas a una posición aproximada. A medida que el flujo disminuye, las cuerdas se juntan y la presión local se aproxima al valor subglótico. El ciclo de relajación se repite. La masa y compilancia de las cuerdas asi como la presión subglótica, determinan el período de oscilación.

El orificio de área variable, producido por la vibración de dos cuerdas, permite pulsos quasi-periódicos de aire que excitan el sistema acústico de las cuerdas vocales, la frecuencia de vibración es de 125 hz. La forma de onda es aproximadamente triangular, los componentes de alta frecuencia disminuyen en amplitud a razón de 12 dB/octava y toma aproximadamente 100 mseg. en alcanzar su amplitud máxima.

Si el conducto nasal se acopla al conducto oral durante la producción de la vocal, ésta se vuelve nasalizada.

B).-Consonantes.

Las consonantes constituyen aquellos sonidos que no son exclusivamente vocalizados, y que son formados a partir de una configuración vocal relativamente estable. Ellas se caracterizan

por constricciones del conducto más grandes que las vocales. Ellas pueden ser excitadas y/o radiadas diferentemente. Los movimientos dinámicos a cierto tiempo del aparato vocal son fundamentales para la producción de una parte importante de las consonantes. Aquellas consonantes donde el movimiento vocal no interviene se llaman continuantes.

Consonantes Fricativas: Son producidas por una excitación de ruido incoherente del conducto vocal.
Este ruido es generado por un flujo de aire turbulento en algún punto de constricción. Algunas constricciones comunes son aquéllas formadas por la lengua abajo de los dientes (dental), los dientes superiores sobre el labio inferior (labio dental), etc.
La radiación de las fricativas ocurre generalmente por la boca. Si la fuente de las cuerdas vocales ocurre en conjunto con la fuente de ruido, la fricativa es vocalizada, Si ocurre solamente la fuente de ruido, la fricativa es no vocalizada o continuante. Una constante fricativa típicamente alcanza su amplitud máxima de 20 a 50 mseg. y la mayoría de su densidad de potencia está concentrada entre 1 y 3 Khz. Un ejemplo de una consonante fricativa es !sssss....!.

Consonantes de alto (Explosivas): Son aquéllas que dependen de la dinámica del conducto vocal. Para producir estos sonidos debe

haber un cierre total en algún lugar del conducto vocal.
La presión generada por los pulmones es soltada de repente con un movimiento abrupto de los articuladores. El cerrado puede ser labial, alveolar, palatar o velar, o no puede haber vocalización. Un ejemplo de una consonante explosiva de cerrado labial es una 'p'.
Este tipo de consonantes se caracterizan por el hecho de que las componentes de alta frecuencia alcanzan el 90% de la amplitud máxima en menos de 5 mseg.

Consonantes Nasales: Son generalmente excitadas por las cuerdas vocales. La mayor parte de la radiación se produce por los orificios nasales y la cavidad oral sirve únicamente como resonador que puede influenciar sustancialmente el sonido. Los sonidos nasales pueden considerarse como continuantes debido a que pueden mantenerse constantes. Un ejemplo de estas consonantes son 'm' o 'n'.

Glides y Semi-Vocales: Son sonidos consonánticos que se parecen a las vocales. Invariablemente parecen una vocal. En el caso de las glides el movimiento es dirigido hacia una vocal. En las semi-vocales, el canal oral está más cerrado que en el caso de las vocales y la punta de la lengua está arriba. Ejemplo de semivocales son : 'w', 'l', 'r', y 'y'.

Sonidos Combinados: Diptongos y Africativas. Un par de vocales que forman un diptongo, es como una vocal, pero se caracteriza por el cambio de una posición a otra. En el caso de las consonantes, es llamada africativa.

I.4.- Fisiología del aparato de generación de voz.

El conjunto del sistema vocal se compone de pulmones y del conducto traqueo-bronquial formado por la laringe, faringe y las cavidades urales y nasales.
Analizando, en función al papel que desempeñan cada una en la fonación:

A).- los pulmones y el conducto traqueo bronquial:

La tráquea, fig.(1.3), es un conducto quasi-cilíndrico de aproximadamente 12 cm. de longitud y de 1.5 cm a 2.5 cm. de diámetro. Su estructura le permite una gran elasticidad vertical (se puede alargar varios centímetros y desplazarse de atrás hacia adelante) con una necesaria indeformidad de acción.
La extremidad superior soporta la laringe, mientras que su extremidad inferior se divide en dos ramas de longitudes y secciones desiguales que alimentan los pulmones.

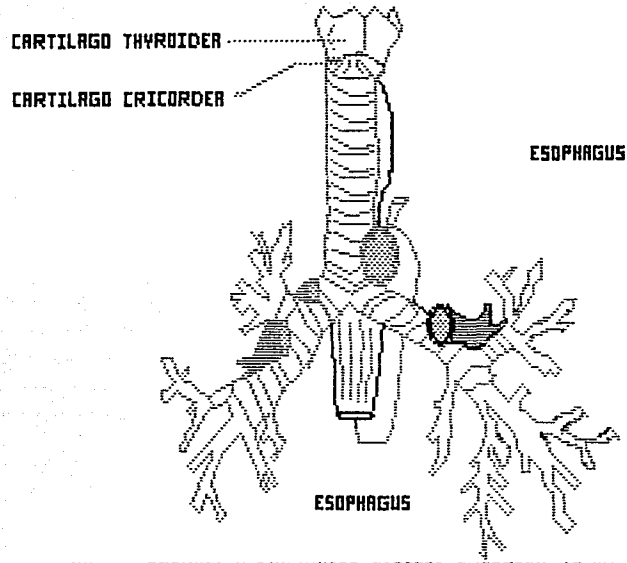


FIGURA 1.3. TRAQUEA Y BRONQUIOS ASPECTO ANTERIOR (3/4).

El tórax, fig.(1.4), con sus doce costillas se comporta como una bomba de aire: en el proceso de la inspiración un sistema de músculos eleva las costillas, mientras que el diafragma se baja contractándose.

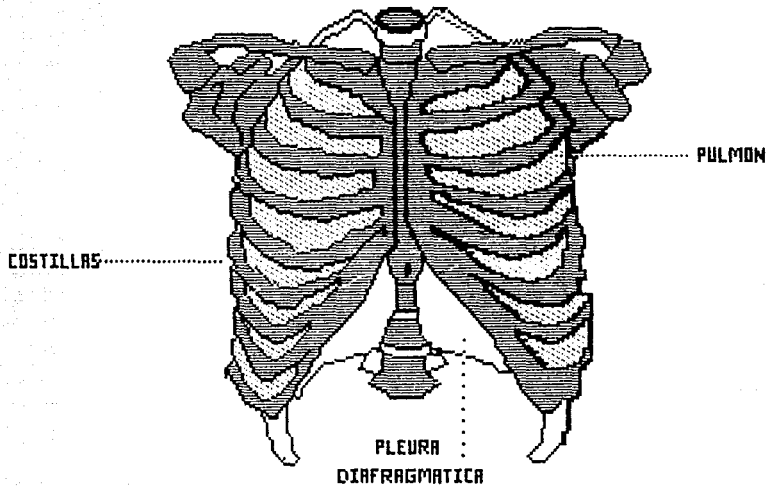


FIGURA 1.4 LIMITES DE LOS LOBULOS DE LOS PULMONES Y DE LA PLEURA.

La respiración no requiere ordinariamente de ninguna contracción muscular: es consecuencia de la elasticidad del tejido pulmonar, de la contracción de los músculos torácicos y del diafragma.

B).- La laringe:

La laringe, fig.(1.2), es un conjunto de cartílagos articulados, ligamentos, músculos y mucosas que sobrepasan la tráquea y penetran en la faringe. En conjunto con la epiglotis, impiden el paso de los alimentos a la tráquea. El esófago es un conducto situado atrás de la tráquea, fig.(1.2). Su sección es deformable, en el estado normal la parte superior está aplastada. Al momento de la deglución, la laringe se desplaza hacia adelante y hacia arriba gracias a un sistema de músculos dispuestos según una estrella de tres ramas.
En la laringe, fig.(1.1), se pueden distinguir:

El cartílago ericoide.
El cartílago arytenoide.
El cartílago curniculés.
El cartílago tiroide.
El hueso hyóide.
La epiglotis.

Los movimientos de todos esos cartílagos resultan de la acción conjunta de varios músculos.
A pesar de la complejidad oseosa y muscular, la forma interior de la laringe es relativamente simple y continua.

C).- El conducto vocal.

El conducto vocal, fig.(1.2), se compone de dos partes. La parte oral, cuya longitud es de 17 a 20 cm., comprende la faringe y las diversas cavidades vocales, de forma y volúmen extremadamente variables y la parte nasal, que es más simple, porque se compone de dos cavidades fijas, conectadas en paralelo sobre el conducto oral por intermediario del "velum palatinum". Este constituye la parte móvil del paladar, normalmente en posición baja para permitir la respiración, se puede alzar y pegar contra la pared posterior de la faringe en el momento de la deglución o en el transcurso de la fonación.

En la parte oral, generalmente el maxilar inferior se encuentra enfrente del maxilar superior, pero el maxilar inferior tiene una gran movilidad: no solamente se puede abrir varios centímetros, sino también se puede desplazar lateralmente hacia adelante aproximadamente un centímetro.

La parte más móvil del conjunto oral es la lengua, que tiene mas de 17 músculos diferentes. Esta puede modelar a voluntad la forma del conducto oral, cambiando rápidamente.

Finalmente los labios, que pueden deformarse de manera continua y en grandes proporciones debido a que se componen de varios músculos.

Haciendo un resumen de los músculos que intervienen en el aparato vocal, (laringe incluída), se constata que hay

aproximadamente 60 músculos que intervienen de manera coordinada. El área transversal en la parte anterior del conducto vocal puede variar de 0 a 20 cm. cuadrados.

Las cavidades nasales miden aproximadamente 12 cm. y contienen un volúmen de aproximadamente 60 cm. cúbicos.

II.1.- Introducción.

La forma general para representar digitalmente una señal de voz analógica se muestra en la fig.(2.1). Como ahí se observa, la

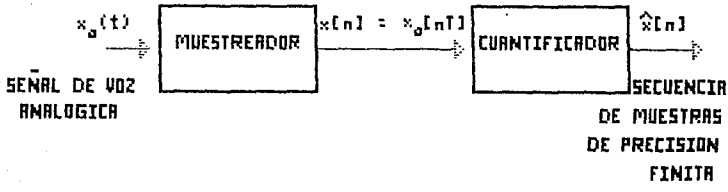


FIG. (2.1) DIAGRAMA GENERAL DE BLOQUES PARA LA REPRESENTACION DIGITAL DE SEÑALES ANALOGICAS

forma de onda de voz, puede ser concebida como una función continua de tiempo, es muestreada, generalmente en forma periódica, para producir una sucesión de muestras $x_a[nT]$. Estas muestras se cuantizan, dentro de un conjunto finito de valores, con el propósito de obtener una representación digital, (discreto

tanto en tiempo como en amplitud), de la señal de voz analógica. Estos dos conceptos básicos, muestreo y cuantización, están inherentes en todos los esquemas para la representación digital de la voz, (o de casi cualquier otra señal continua en tiempo y amplitud).

II.2.- Muestreo.

En el uso de métodos de procesamiento digital de señales analógicas como la voz, es necesario representar digitalmente la señal analógica como una sucesión de números. Esto se hace comúnmente al muestrear periódicamente la señal analógica, denotada como $x_a(t)$, para producir la sucesión:

$$x[n]=x_a[nT] \qquad -\infty < n < \infty$$

donde n, toma valores enteros.

II.2.1.- Teorema del muestreo.

Las condiciones para que la sucesión de muestras, en la igualdad anterior, sea una representación única de la señal analógica original, se resumen en el siguiente teorema de muestreo

Si una señal $x_a(t)$ tiene una transformada de Fourier de banda limitada $X_a(j\omega)$, tal que $X_a(j\omega)=0$, para $|\omega| \geq 2\pi Fn$, entonces $x_a(t)$ puede ser reconstruida unívocamente de un espaciamiento de muestras $x_a[nT]$, para $-\infty < n < \infty$, si $1/T \geq 2Fn$.

La transformada de Fourier de $x_a(t)$ está definida como:

$$X_a(j\omega)=\int_{-\infty}^{\infty} x_a(t)\, e^{-j\omega t}\, dt$$

y la transformada de Fourier de la sucesión x[n] está definida como:

$$X(e^{j\omega T})= \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a\left[j\omega + j\,\frac{2\pi}{T}\, k\right] \quad \ldots\ldots\ldots (A)$$

Supóngase que $X_a(j\omega)$ es como se muestra en la fig.(2.2.a) y $X_a(j\omega)=0$ para $|\omega|>\omega_n=2\pi F_n$. F_n es la frecuencia de Nyquist: de acuerdo a la ecuación (A), $X[e^{j\omega t}]$ es la suma de un número infinito de réplicas de $X_a(j\omega)$, centrada cada una en múltiplos enteros de $2\pi/T$. La fig.(2.2.b), muestra la situación cuando $1/T>2F_n$, en este caso las reproducciones de la transformada de Fourier no se traslapan en la banda base, esto es, $|\omega|<2\pi F_n$. La fig.(2.2.c), muestra la situación cuando $1/T<2F_n$; en este caso la imagen centrada en $2\pi/T$ se traslapa en parte con la banda base. Esta condición donde una alta frecuencia se toma sobre la identidad de una baja frecuencia, es llamado traslape

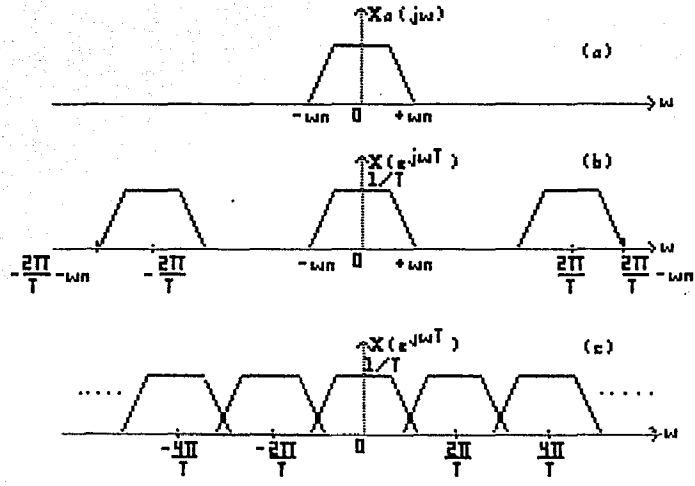


FIG. (2.2)

(aliasing). Evidentemente el traslape puede ser evitado solo si la transformada de Fourier es de banda limitada y si la frecuencia de muestreo (1/T) es al menos igual al doble de la frecuencia de Nyquist ($1/T>2F_n$) sobre la condición $1/T>2F_n$, bajo esta condición la transformada de Fourier de la sucesión de muestras es proporcional a la transformada de Fourier de la señal

analógica en banda base, esto es:

$$X(e^{j\omega T})=\frac{1}{T}X_a(j\omega) \quad |\omega|<\frac{\pi}{T}$$

Usando este resultado, se puede mostrar que la señal original puede ser reconstruída a partir de la sucesión de muestras utilizando la fórmula de interpolación:

$$x_a(t)=\sum x_a[nT]\left[\frac{\text{sen}\ [\pi(t-nT)/T]}{\pi(t-nT)/T}\right]\ldots\ldots\ldots(B)$$

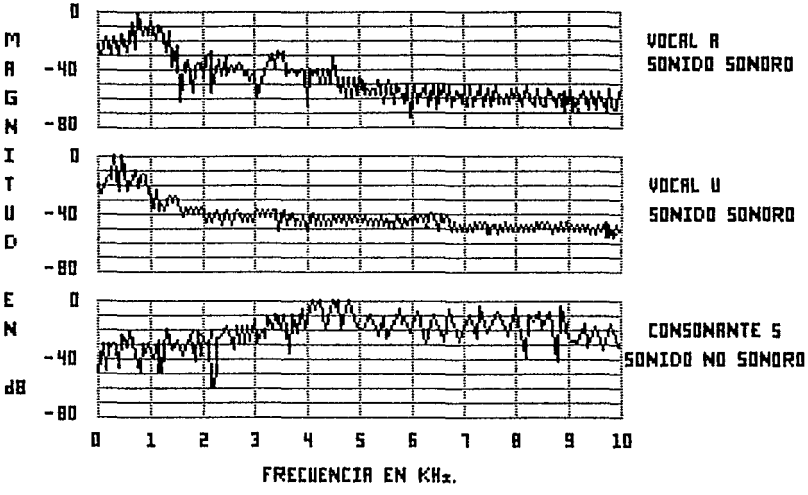
De ahí que en el diseño de convertidores Digital-Analógico se busque la aproximación a la ecuación (B). Es importante hacer notar que la señal de voz no puede ser muestreada directamente, ya que no es una señal de banda limitada, con lo cual no puede aplicarse el teorema de muestreo, por lo tanto, es necesario filtrar la señal antes del muestreo, y así limitar su ancho de banda.

II.2.2.- Muestreo de señales de voz.

Para analizar el muestreo de señales analógicas de voz, consideremos las propiedades espectrales de la voz. En los modelos para la producción de vocales y sonidos fricativos, las señales de voz no son inherentemente de banda limitada, aunque el

espectro declina rápidamente en muy altas frecuencias. Como se observa en la fig.(2.3), en los sonidos sonoros, las altas

FIG. (3.3). ESPECTROS DE SONIDOS, FRECUENCIA DE MUESTREO 20 KHz.



frecuencias, aproximadas a 4 khz, tienen magnitudes mayores de 40 dB, por otro lado, para sonidos no sonoros, el espectro no baja sensiblemente para frecuencias mayores a 8khz. Por lo tanto, para una representación exacta de todos los sonidos de la voz,

requeriremos un promedio de muestras mayor que 20 khz. En muchas aplicaciones, sin embargo, este promedio de muestreo no es requerido. Por ejemplo, si la operación de muestreo es un preludio para el proceso de estimar las primeras componentes de frecuencia de voz sonora, nosotros estaríamos interesados solamente en la porción del espectro hasta aproximadamente de 3.5 khz. Si la voz es filtrada con una frecuencia de corte de 4 khz. antes de muestrearse, la frecuencia de muestreo sería aproximadamente 8 khz., este es el caso cuando la voz se transmite sobre una linea telefónica, la fig.(2.4) muestra la curva de respuesta en frecuencia típica para transmisión por vía telefonica.

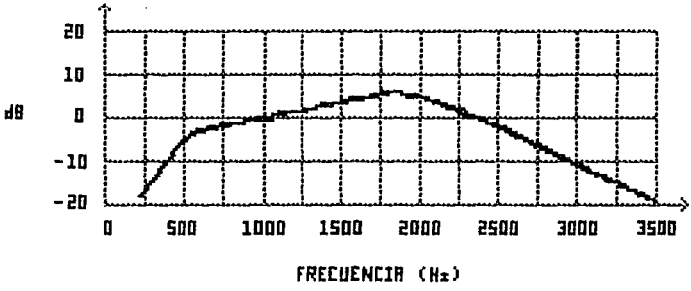


FIG. (2.4). CURVA TIPICA DE FRECUENCIAS PARA TRANSMISION DE VOZ POR VIA TELEFONICA.

Como se observa en esa figura, la transmisión por vía telefónica de señales de voz tiene el efecto de limitación de banda. Un punto importante del muestreo es que, aunque la forma de onda de la señal a procesar digitalmente pueda tener un espectro de banda-limitada, la señal puede ser degenerada por el ruido aleatorio en el ancho de banda anterior a la conversión analógica-digital. En este caso, la señal suma una combinación de ruido que será filtrado con un filtro analógico paso-bandas que cortará sobre la frecuencia de Nyquist, así que las imágenes del ruido de altas frecuencias no serán conocidas en la banda base. Por lo tanto, para cualquiera de los dos casos mencionados, es necesario filtrar la señal analógica antes que se procese digitalmente.

II.3.- Modelos Estadísticos para la Voz.

Para la representación digital de señales de voz analógica es conveniente asumir que las formas de onda de la voz pueden ser representadas por un proceso aleatorio ergódico. Si nosotros asumimos que la señal $x_a(t)$ es una función muestreada de un proceso aleatorio de tiempo continuo, entonces la secuencia de muestras derivadas de un muestreo periódico puede ser igualmente concebido como una sucesión de muestras de un proceso aleatorio de tiempos discretos. Para muchos propósitos en análisis de sistemas de comunicaciones, una adecuada caracterización de la

señal analógica consiste de una densidad de probabilidad de 1er. orden p(x), y la función de autocorrelación del proceso aleatorio, que es definido como:

$$\phi_a(\tau)=E\ [x_a(t)x_a(t+\tau)]$$

donde E [ ] denota la esperanza del soporte. El espectro de potencia analógica es la transformada de Fourier de $\phi_a(\tau)$, esto es:

$$\Phi_a(\omega)=\int\phi_a(\tau)e^{-j\omega\tau}\ d\tau$$

La señal obtenida en tiempo discreto de muestrear la señal aleatoria $x_a(t)$ tiene como función de autocorrelación:

$$\phi(m)=E\ [x[n]x[n+m]]$$

$$=E\ [x_a(nT)x_a(nT+mT)]=\phi_a(mT)$$

Así, después $\phi(m)$ es una versión actual de muestreo de $\phi_a(\tau)$, entonces el espectro de potencia está dado por:

$$\Phi(e^{j\omega T})=\sum_{-\infty}^{\infty}\phi(m)e^{-j\omega Tm}$$

$$=\frac{1}{T}\sum_{k=-\infty}^{\infty}\Phi_a(\omega+\frac{2\pi}{T}k)$$

La ecuación anterior muestra que para el proceso aleatorio del modelo de voz, el espectro de potencia de la señal muestreada es una versión conocida del espectro de potencia de la señal analógica original.

La función densidad de probabilidad para las amplitudes $x[n]$, es la misma para las amplitudes $x_a(t)$, después que $x[n]=x_a[nT]$. Así la media y la variancia, son las mismas, tanto para las muestras como para la señal analógica original.

Los procesos estadísticos a señales de voz se aplican para estimar la función densidad de probabilidad y la función de autocorrelación, (o espectro de potencia), de las formas de onda de voz.

La densidad de probabilidad se puede estimar, a partir de un histograma de amplitudes, para un número grande de muestras, sobre un tiempo grande.

Una buena aproximación para medir la densidad de amplitud de la voz es una distribución "Gamma" de la forma:

$$p(x)=\left[\frac{\sqrt{3}}{8\pi\sigma_x|x|}\right]^{1/2} e^{-\frac{\sqrt{3}\,|x|}{2\sigma_x}}$$

Una aproximación algo más simple es la densidad "Laplaciana":

$$p(x)=\frac{1}{\sqrt{2}\,\sigma_x}\,e^{-\frac{\sqrt{2}\,|x|}{\sigma_x}}$$

La fig. (2.5), muestra una medida de la densidad de amplitud para

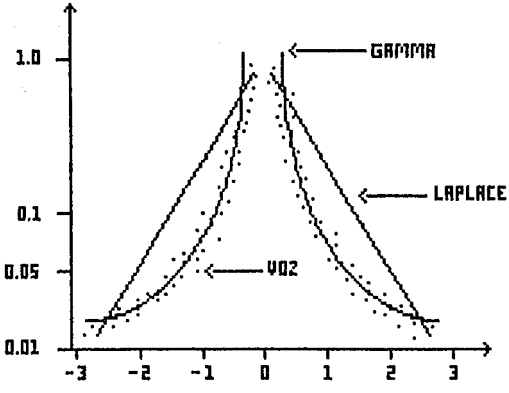


FIG. (2.5). VOZ REAL Y DENSIDADES DE PROBABILIDAD GAMMA Y LAPLACE

la voz, junto con las densidades "Gamma" y "Laplaciana", todas han sido normalizadas, así que la media es cero y la variancia es

unitaria (σ_x^2). La densidad "Gamma", como se muestra en la fig.(2.5), es evidentemente una mejor aproximación que la densidad "laplaciana", pero ambas son razonablemente cercanas. La función de autocorrelación y espectro de potencia de señales de voz, pueden ser estimadas con técnicas estándar de análisis de series de tiempo. Una estimación de la función de autocorrelación de un proceso aleatorio ergódico puede ser obtenida al estimar la función de autocorrelación en tiempos promedio de un segmento grande, (pero finito), de la señal. Por ejemplo, la definición de la función de autocorrelación de tiempos cortos, puede ser ligeramente modificada para darnos la estimación de la función promedio de autocorrelación de tiempos largos, esto es:

$$\hat{\phi}(m) = \frac{1}{L} \sum_{n=0}^{L-1-m} x(n)x(n+m) \quad ; \qquad 0 \leq |m| \leq L-1$$

donde L es un entero grande.

El espectro de potencia puede ser estimado en una variedad de formas para la voz, uno de los mejores resultados se obtiene de la medida del promedio de salida de un conjunto de filtros paso-banda. Una alternativa apropiada para la estimación del espectro de potencia promedio de términos grandes, es una primera estimación de $\hat{\phi}(m)$, como en la ecuación anterior, y entonces calculamos:

$$\Phi(e^{j\omega T}) = \sum_{m=-M}^{M} \omega(m)\hat{\phi}(m)e^{-j\omega mT}$$

Para un conjunto discreto de frecuencias. $\omega_k = 2\pi k/T$, k=0,1...,N-1, usando la transformada discreta de Fourier, donde $\omega(m)$ es una función de ventana en la función de autocorrelación.

II.4.- Cuantización.

II.4.1.- Cuantización Instantánea.

Después de obtener una sucesión de muestras a partir de una señal analógica, el valor de la amplitud de cada muestra debe ser extraída de un conjunto finito de amplitudes, de manera que puedan ser representadas por un conjunto finito de símbolos. Estos procesos llamados de cuantización y codificación, respectivamente se muestran en la fig.(2.6.a), donde el proceso de cuantización produce una secuencia de amplitudes cuantizadas $\{\hat{x}[n]\}=\{Q[x[n]]\}$ y el proceso de codificación representa cada muestra cuantizada a partir de un código de palabras c[n]. La cantidad Δ en esta figura representa el tamaño del paso de cuantización.

Para recuperar las muestras cuantizadas, un decodificador, fig.(2.6.b), toma una secuencia de muestras de un código de palabras, $\{c'[n]\}$, y transforma estas posteriormente en una secuencia de muestras cuantizadas, $\{\hat{x}'[n]\}$. Si el código de

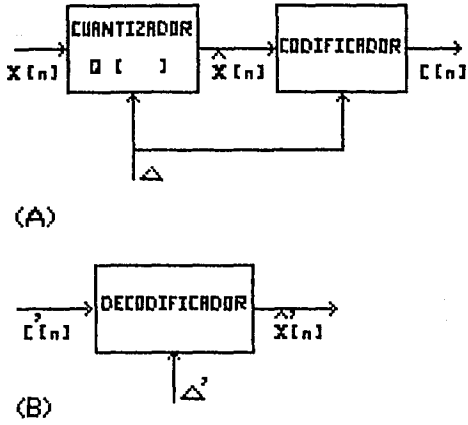


FIG. (2.5). PROCESO DE CUANTIZACION Y CODIFICACION
(A) CODIFICADOR ; (B) DECODIFICADOR

palabras c'[n] son de las mismas muestras que el código de palabras c[n], y considerando que no se han introducido errores, la salida del decodificador ideal es idéntica a las muestras cuantizadas. Con un código binario de longitud B es posible representar 2^B diferentes niveles de cuantización. La capacidad de información requerida para transmitir o almacenar la representación digital es por lo tanto:

$$L=B*F_s = \text{Promedio de bits/seq.}$$

Donde F_s es el promedio de muestreo, (muestras/seq.), y B es el número de bits/muestra. En general, es razonable asumir que las muestras {x[n]} caerán en un rango finito de amplitudes tales que:

$$|x[n]| \leq X_{max.}$$

Las amplitudes de las muestras serán cuantizadas al dividir el rango completo de amplitudes en un conjunto finito de rangos de amplitudes, y asignar el mismo valor de amplitud a todas las muestras que caen dentro del rango dado. Esto se observa en la fig.(2.7), para un cuantizador de 8 niveles. Por ejemplo, vemos que para todos los valores de x[n] entre x1 y x2, la salida del cuantizador es $\hat{x}(n)=Q[x(n)]=\hat{x}_2$, cada uno de los 8 niveles del cuantizador está referido a un código de palabra binario de 3 bits que sirve como una representación del nivel de amplitud. Por ejemplo, en la fig.(2.7), el código de salida para una muestra cuya amplitud esta entre x1 y x2 sería el número binario 101.

II.4.2.- Cuantización Uniforme.

Los rangos y niveles de cuantización pueden ser elegidos en una variedad de formas que dependen de la aplicación en la representación digital. Cuando la representación digital es para

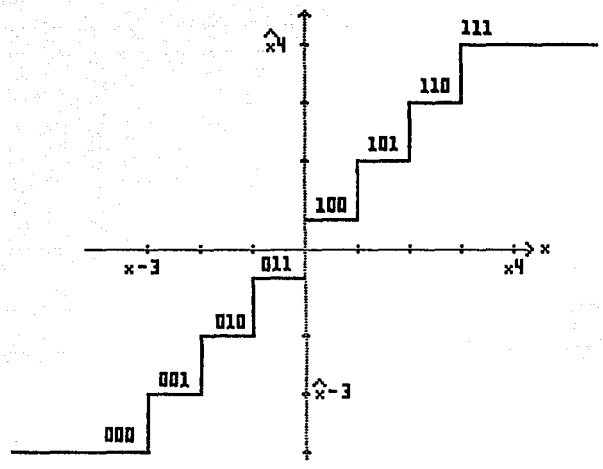


FIG. (2.7). CARACTERISTICAS DE ENTRADA Y SALIDA DE UN CUANTIZADOR DE TRES BITS

ser procesada por un sistema digital, los niveles y rangos de cuantización son generalmente distribuidos uniformemente.

Así, para definir un cuantizador uniforme, usando el ejemplo de la fig.(2.7), establecemos:

$$x_i - x_{i-1} = \Delta$$

y

$$X_i - x_{i-1} = \Delta$$

donde Δ es el tamaño del intervalo de cuantización. Dos características comunes de un cuantizador uniforme se muestran en la fig.(2.8), para el caso de 8 niveles de cuantización.

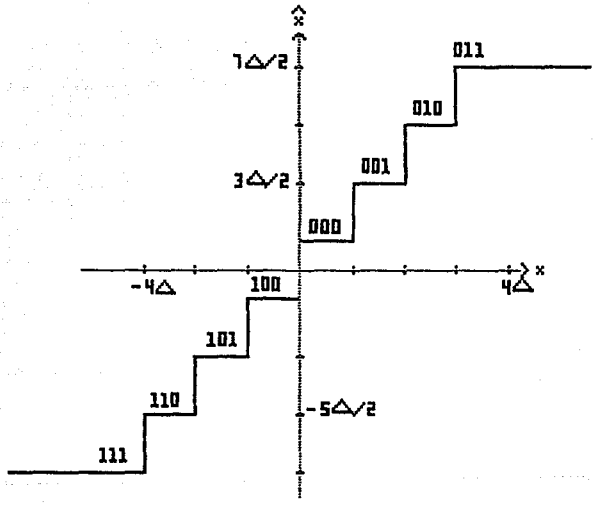


FIG. (2.8a). CUANTIZADOR MID-RISER

La fig.(2.8.a) muestra el caso donde el origen parece estar en la mitad de la altura de la escalera de la función, a esta clase de cuantizadores se le llama "mid-riser". De la misma forma la fig.(2.8.b) muestra un ejemplo del cuantizador "mid-tread".

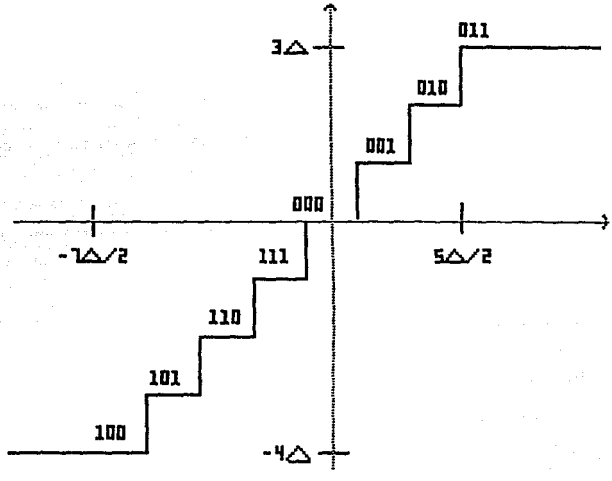


FIG. (2.8b). CUANTIZADOR MID-TREAD

Para el caso donde el número de niveles es una potencia de 2, como es conveniente para un esquema de codificación binario, puede verse que el cuantizador "mid-riser" tiene el mismo número de niveles positivos y negativos, y éste está posicionado simétricamente sobre el origen. En contraste, el cuantizador "mid-tread" tiene un nivel negativo mas que un positivo; sin embargo, en este caso uno de los niveles de cuantización es cero, mientras que este nivel no es cero en el caso "mid-riser".
Para los cuantizadores uniformes como los de la fig.(2.8), se tienen dos parámetros: el número de niveles y el tamaño del intervalo de cuantización (Δ). El número de niveles es generalmente elegido para ser de la forma 2^B de tal forma que es más eficiente el uso del código de palabras binarios de B-bits. Juntos Δ y B deben ser elegidos para cubrir el rango de entrada de las muestras. Si asumimos que $|x[n]| \leq X_{max.}$, entonces colocaríamos:

$$2X_{max.} = \Delta 2^B \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (C)$$

Al discutir los efectos de cuantización es útil representar las muestras cuantizadas $\hat{x}(n)$ como:

$$\hat{x}[n] = x[n] + e[n]$$

donde x[n] es la muestra no cuantizada y e[n] es el error de

cuantización o ruido, esto puede verse de las figs.(2.8.a) y (2.8.b) que si Δ y B son elegidos como en la ecuación (C), entonces:

$$-\frac{\Delta}{2} \leq e[n] \leq \frac{\Delta}{2a}$$

Para el estudio de los efectos de cuantización, es conveniente y usual asumir un simple modelo estadístico para la cuantización del ruido, este modelo está basado en las siguientes asunciones:

1.- La cuantización del ruido es un proceso de ruido blanco estacionario, esto es:

$$E\,[e[n]e[n+m]]=\sigma_e^2 \quad \text{si } m=0$$
$$=0 \text{ de otra forma.}$$

2.- La cuantización del ruido está incorrelacionada con la señal de entrada, esto es:

$$E\,[x[n]e[n+m]]=0 \quad \text{para toda m.}$$

3.- La distribución de los errores de cuantización es uniforme sobre cada intervalo de cuantización y después todos los intervalos son de la misma longitud:

$$p_e(e)=\frac{1}{\Delta}\ ; \qquad -\frac{\Delta}{2} \leq e \leq \frac{\Delta}{2}$$
$$=0\ ; \qquad \text{de otra forma.}$$

Estas asunciones son claramente irrealistas para algunos tipos de señales, por ejemplo si la entrada es una constante para toda n, las anteriores asumpciones no son apropiadas. La voz, sin embargo, es una señal complicada que fluctúa rápidamente entre todos los niveles de cuantización, y si Δ es suficientemente pequeña, la amplitud de la señal es apropiada para recorrer muchos pasos de cuantización yendo de muestra en muestra. Para este caso se han efectuado experimentos con estas asunciones obteniendose buenos resultados.

Con este modelo estadístico para la cuantización del ruido, es posible relacionar el tamaño del ruido con el tamaño de la señal y los parámetros del cuantizador. Para este propósito es conveniente calcular el promedio señal-ruido cuantizado que se define como:

$$SNR=\frac{\sigma_x^2}{\sigma_e^2}=\frac{E[x^2[n]]}{E[e^2[n]]}=\frac{\sum_n x^2[n]}{\sum_n e^2[n]} \quad .....(1)$$

Si el rango dinámico del cuantizador es asumido para ser $2X_{max.}$, entonces, para un cuantizador de B-bits, tenemos:

$$\Delta=\frac{2X_{max.}}{2^B} \quad .....(2)$$

Si asumimos una distribución de amplitudes uniforme para el ruido, obtenemos:

$$\sigma_e^2 = \frac{\Delta^2}{12} = \frac{X_{max.}^2}{(3)2^{2B}} \quad \text{.....(3)}$$

sustituyendo (3) en (1) obtenemos:

$$SNR = \frac{(3)2^{2B}}{\left[\frac{X_{max.}}{\sigma_x}\right]^2} \quad \text{.....(4)}$$

o expresando la señal-ruido cuantizado en unidades dB:

$$SNR = 10 \log_{10}\left[\frac{\sigma_x^2}{\sigma_e^2}\right] = 6B+4.77-20 \log_{10}\left[\frac{X_{max}}{\sigma_x}\right] \quad \text{.....(5)}$$

ahora si asumimos que el rango del cuantizador es tal que $X_{max.}=4\sigma_x$, entonces la ecuación (5) llega a ser:

$$SNR\ (dB) = 6B - 7.2$$

La ecuación anterior nos da a entender que cada bit en el código de palabra contribuye 6 dB a el promedio señal-ruido.

II.5.- Relaciones señal a ruido con compansión.

Para muchas clases de señales no existe un valor pico específico y el nivel de la señal puede, de hecho variar de una forma aleatoria. El ejemplo más común es el de la transmisión de voz con diferentes personas utilizando las mismas instalaciones de transmisión. El intervalo de transmisión de la voz puede ser hasta 40 dB, desde el murmullo de una persona que habla suavemente hasta los gritos que puede proferir otra. Es evidente que para cubrir este intervalo dinámico debe usarse efectivamente el espaciamiento no uniforme de niveles de cuantización, o, en forma equivalente, la compresión de la señal. Si esto no se pone en práctica y se emplean los niveles igualmente espaciados para cubrir el mayor espacio posible de variación de la señal, las voces suaves no se transmitirán. El mismo problema se presenta, obviamente, en la transmisión de PCM de cuaquier señal analógica que se espera que cubra un determinado intervalo dinámico.

Teóricamente una variable aleatoria gaussiana, en este caso, la amplitud de la señal, puede tomar cualquier valor posible, no habiendo máximo teórico. Sin embargo, hay un 99.99% de probabilidad de que la variable caiga en el intervalo comprendido por $\pm 4\sigma$. Por tanto, puede escogerse $V=X_{max.}=4\sigma$ para mayor seguridad.

Para el ruido de cuantización cuadrático promedio se tiene:

$$E\ (\varepsilon^2) = \frac{a^2}{12} = \frac{V^2}{3M^2}$$

donde M son los niveles de cuantización y utilizando la característica del cuantizador a=2V/M. Considerando una potencia de entrada de la señal σ^2, se tiene para la relación señal a ruido:

$$SNR = \frac{\sigma^2}{E(\varepsilon)^2} = 3M^2\left[\frac{\sigma^2}{V^2}\right]$$

Como el ruido de cuantización es fijo, independiente de σ^2, para el caso de los niveles uniformemente espaciados, la SNR es proporcional a σ^2. A medida que el locutor reduce la intensidad de la voz, la SNR se reduce en proporción. El ruido de cuantización se hace cada vez más notorio. Este es el problema al que se ha aludido. Para disminuirlo y obtener una SNR relativamente fija, en un intervalo dinámico de señales más amplio, es necesario introducir la desuniformización de los niveles de cuantización. Una opción consiste en que en la práctica es más simple comprimir la señal en forma no lineal y después aplicar el espaciamiento uniforme de los niveles de la señal ya comprimida.

En el receptor la señal es expandida entonces con una característica inversa no lineal. Es obvio que esto equivale al

espaciamiento no uniforme de los niveles. En la fig.(2.9) se muestra un ejemplo de esta clase, x' representa la señal de entrada y y' la salida. Según la característica elegida, los niveles equivalentes de entrada se desplazan cada vez más lejos a medida que la amplitud de entrada tiende a $\pm$ V , lo cual se debe a la compresión de los valores superiores de la entrada en un intervalo relativamente menor de valores de salida.

La característica de compresión típica tiene la forma logarítmica. Una forma especialmente común que se realiza en la práctica para la transmisión telefónica de voz es la compansión de la ley μ. Esta tiene la siguiente forma:

$$y(x') = \frac{\ln(1 + \mu x'/V)}{\ln(1 + \mu)} \qquad 0 \leq x' \leq V$$

la cual tiene simetría impar con respecto al punto x'=0. y(x') varía entonces en el intervalo $\pm$ 1 . El parámetro μ que aparece puede variarse con el objeto de obtener una familia de curvas características. Nótese que para $x' \ll V/\mu$ la característica es casi lineal:

$$y(x)' \doteq \frac{\mu x}{V \ln(1+\mu)} \qquad x' \ll \frac{V}{\mu}$$

A medida que x' aumenta hasta el punto V/μ, la característica del logaritmo se satura. Para $\mu \ll 1$, $y(x') \doteq x'/V$, la compresión lineal desaparece y el espaciamiento uniforme de la salida y corresponde

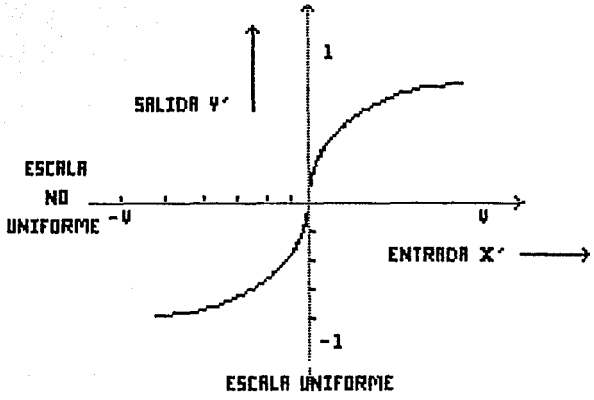


FIG. (2.9). CARACTERISTICA DEL COMPRESOR NO LINEAL

al espaciado uniforme de la entrada x'. En Estados Unidos, Bell System ha adoptado una ley de compansión μ=255 para sus sistemas de portadora digitales. Esta característica en particular se dibuja para x'>>0, en la fig.(2.10). Cuando se lleva a cabo un análisis de SNR y del ruido de cuantización para la característica de compansión de la ley μ, es de utilidad definir

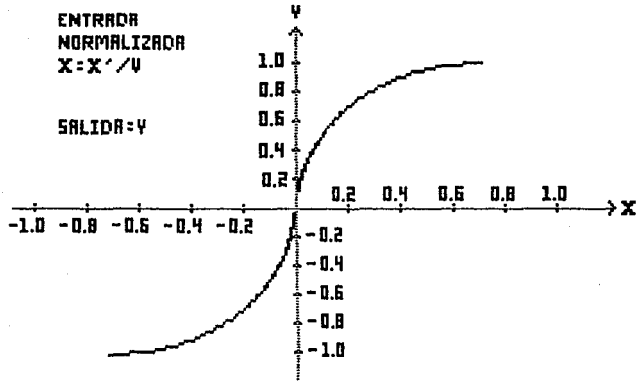


FIG. (2.10). CARACTERISTICA DEL COMPRESOR, COMPANSION DE LA LEY MU, CON MU=255.

una señal de entrada normalizada x=x'/V, en términos de la cual, la característica del compresor de la ley μ está dada por:

$$y(x)=\frac{\ln\ (1+\mu x)}{\ln\ (1+\mu)} \qquad 0 \leq x \leq 1$$

La forma de la entrada normalizada es la que se muestra en la fig.(2.10).

Se han propuesto otras leyes logaritmicas compansoras, además de la característica de la ley μ que ya mencionamos. Un ejemplo es la ley A, que tiene la siguiente característica (la señal de entrada x se normaliza de nuevo en relación con la máxima señal de entrada):

$$y = \frac{1 + \ln Ax}{1 + \ln A} \qquad \frac{1}{A} \leq x \leq 1$$

$$= \frac{Ax}{1 + \ln A} \qquad 0 \leq x \leq \frac{1}{A}$$

La característica tiene simetria impar respecto a x=0. El valor A=100 es típico de los compansores que emplean esta característica. La SNR que resulta para este valor de A tiene un intervalo dinámico más amplio que la del compansor de ley μ, con μ=255, pero la SNR de salida es algo menor.

Nótese que la característica de la ley A se define lineal para x pequeña, y logaritmica para x grande. Tanto la característica de la ley μ, como la de la ley A, se desarrollan en la práctica por medio de aproximaciones lineales por tramos.

III.1.- Introducción.

El arte de comunicarse de forma que sólo el destinatario auténtico del mensaje lo entienda, es casi tan antiguo como la historia del lenguaje escrito. Este tipo de comunicación fue utilizado por los chinos, persas, babilonios, asirios, griegos y romanos.

El primer encifrador conocido fue realizado por los lacedemonios, y consistía de un bastón, de diámetro específico, en el que se enrollaba una cita escrita en un orden tal que solo podía leerse correctamente una vez enrollada. Después se utilizaron los sistemas de transposición, es decir, el cambiar el orden lógico de las letras; le siguieron los sistemas de sustitución, en estos sistemas, se cambian los números o letras por otros inventados y se escribe el mensaje con los nuevos signos en lugar de las letras usuales. Ya para el renacimiento se inventó en Roma la primera máquina cifradora, creada por Alberti, la cual consistía de dos discos concéntricos en cuyos margenes se encuentran, por un lado, un alfabeto normal y por otro, uno con las letras en orden alterado. Basta hacer coincidir dos letras preacordadas para que el cifrador y receptor del mensaje puedan tener la comunicación deseada. Para que la relación mensaje-mensaje encriptado fuera diferente a la anterior basta recorrer una letra, así el mensaje sería escrito con la misma clave cada 28 mensajes.

La gran aportación a la criptografía la daría Vernam muchos años después, ya que utilizó la electricidad en su sistema cifrador

apoyándose en la telegrafía. En el código Morse se utilizan señales cortas y señales largas, a las señales cortas se les relacionó con un cero y a las señales largas con un uno, además introdujo el azar a los sistemas de cifrado. Utilizando una moneda, cara representaría el cero y cruz representaría el uno, así, los valores causales de la moneda se sumarían a los reales mediante la siguiente clave: 0+0=1, 1+0=1, 1+1=0. Se sumaban pues tantas tiradas de moneda como letras al mensaje, previamente en Morse y después en binario.

Durante la segunda guerra mundial se dió la importancia decisiva a la criptografía creándose las máquinas descifradoras y ganando la electrónica la batalla contra la mecánica.

El ataque a Pearl Hearbour fue, según muchos, conocido con anterioridad aunque no se hiciese nada por evitarlo. La máquina "magic" que permitió tales conocimientos a los estadounidenses, descifraba con toda facilidad el código "púrpura" japonés.

El almirante japonés Yamamoto sufrió la emboscada de unos cazas americanos que conocían con anterioridad su desplazamiento en un avión de bombardeo, su ruta y su horario. Su muerte en este hecho cambió el curso de la guerra en el pacífico. De qué manera los servicios norteamericanos obtuvieron el código púrpura nunca se ha aclarado.

La máquina "enigma" que los mandos militares alemanes consideraban como imposible de descifrar, se basaba en un perfeccionamiento de otro criptógrafo famoso, a su vez

perfeccionado del viejo sistema Alberti, que su inventor fue el presidente norteamericano Jefferson. En lugar de discos se utilizó en la máquina un cilindro formado por una serie de anillos. Sobre cada uno de éstos se graba un álfabeto código que puede, combinado con el siguiente, originar 26 códigos diferentes, a mayor número de anillos mayor número de claves. Con sólo tres se obtienen mas de 17,000 combinaciones.

En la máquina "enigma" el código era reversible: con el mismo sistema que se cifraba era posible el descifrado rápido. Sus creadores aseguraban que si se cambiaban los códigos cada 48 horas el enemigo necesitaría unos 42,000 años para encontrar la clave de un solo criptograma.

Sin embargo, con la primera computadora de descifrado "Bomb" creada por el británico Alan Turing, los aliados supieron siempre los secretos de sus enemigos, y de esta forma se ayudó a que los aliados finalmente ganaran la guerra.

Desde entonces la criptografía no sólo se aplica a fines bélicos, actualmente bancos, empresas, y otras instituciones se valen de redes de computadoras, télex, teléfono, etc. para sus operaciones y tienen que hacer uso de la criptografía para proteger información valiosa.

III.2.- Elementos de la Criptografía.

Un sistema de comunicación en el cual la información es confidencial y ésta no es entendida por personas no autorizadas se denomina sistema cifrador. En un sistema cifrador se pueden reconocer los siguientes elementos, observar la fig.(3.1) :

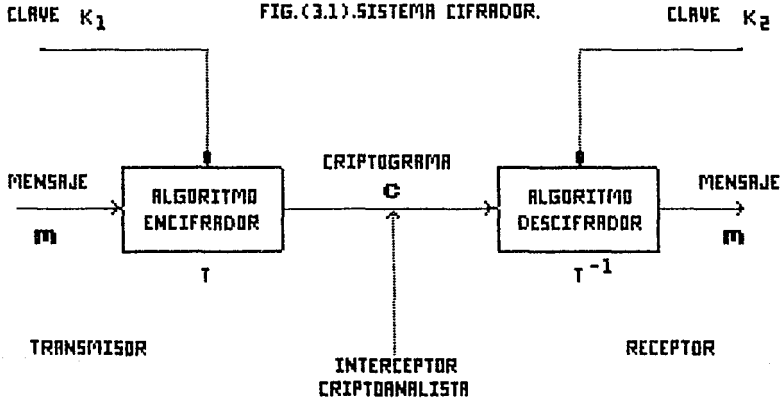


FIG.(3.1).SISTEMA CIFRADOR.

Mensaje (Plaintext): Es la información confidencial que se desea hacer llegar desde el transmisor hasta el receptor.

Algoritmo encifrador: Es el conjunto de reglas que con una cierta clave convierten un mensaje en un criptograma.

Criptograma: Es la información que se transmitirá desde el transmisor hasta el receptor y puede ser obtenida por el interceptor.

Interceptor o criptoanalista: Es la persona que desea conocer el mensaje y que no está autorizada para ello, no es un elemento del sistema cifrador pero lo incluímos aquí ya que éste da la razón de existencia a los sistemas cifradores.

Algoritmo descifrador: Es el conjunto de reglas que con una cierta clave convierten un criptograma en mensaje original.

En la práctica la seguridad de un sistema cifrador no depende, en general, del desconocimiento del algoritmo encifrador. Depende de la facilidad que se tenga para deducir la clave ki que fue usada en el algoritmo encifrador, ya que el algoritmo encifrador puede recibir como entrada una clave ki de n claves ki, donde n es un número muy grande.

En los sistemas cifradores en los cuales se tiene una clave única

para obtener un criptograma se hace referencia a un código y no a un algoritmo.

En este tipo de sistemas cifrados la seguridad depende del desconocimiento que se tenga del código. Un ejemplo de este tipo de sistemas es un convertidor A/D donde existe una relación unívoca entre una señal analógica y una señal digital.

Al arte o la ciencia de diseñar sistemas cifrados se le conoce como CRIPTOGRAFIA, esta palabra tiene raíces griegas: kruptos: oculto y graphein: escribir. Al proceso de deducir el mensaje a partir del criptograma, sin conocer la llave o código, según sea el caso, se le conoce como criptoanálisis.

III.3.- Principios de la Criptografía.

La mayor influencia que tiene la criptografía moderna se debe a los trabajos realizados por Shannon en los años cuarentas. Shannon observó que la clave ki, fig.(3.1), determina una transformación del conjunto de todos los posibles mensajes al conjunto de todos los posibles criptogramas. Estos dos conjuntos son denominados respectivamente espacio del mensaje y espacio del criptograma.

Otra definición para un sistema cifrador de acuerdo a Shannon es: Un sistema cifrador es un conjunto de transformaciones T sobre un conjunto finito del espacio del mensaje M que dan como resultado

un conjunto finito del espacio del criptograma C.

Un requerimiento fundamental en los sistemas cifrados es que conociendo el criptograma, el algoritmo encifrador y la clave empleada, se puede obtener el mensaje original que es único. Es decir :

Si C=t(m) [El mensaje m es transformado en el criptograma C por la transformación t].

Si $m = t^{-1}(C)$ [El mensaje m puede ser determinado por la transformación inversa de t sobre el criptograma C].

De lo anterior y observando la fig.(3.1), se determina que la transformación t depende del algoritmo encifrador y de la clave K1, mientras que la transformación inversa t^{-1} depende del algoritmo descifrador y de la clave K2.

El receptor conociendo C y t está habilitado para deducir m, mientras que el interceptor conoce C y probabilidades de varias ts, con dichos elementos se espera que el interceptor no pueda deducir m, teniéndose así seguridad perfecta en el sistema cifrador.

Siendo :

M = (M1,M2,....,Mn)	Espacio del mensaje.
C = (C1,C2,....,Cn)	Espacio del Criptograma.
P(Mi)	Probabilidad de que el mensaje Mi sea transmitido.
Pj(Mi)	Probabilidad de que Mi haya sido transmitido dado que Cj fue recibido por el criptoanalista.

MM subconjunto de M. $Mn \in MM$

Si $Pj(Mh) > Pj(Mi) \quad \forall \quad Mi, Mh \in MM.$

Entonces el criptoanalista deducirá que el mensaje transmitido fue Mh.

Para que un sistema cifrador tenga seguridad perfecta se debe cumplir que :

$Pj(Mi)=P(Mi) \quad \forall Mi \text{ y } \forall j$ [La probabilidad de que Mi fue transmitido, dado que Cj fue recibido por el criptoanalista, sea igual a la probabilidad de que el mensaje Mi sea transmitido].

Ejemplifiquemos la seguridad perfecta de un sistema cifrador, primero para uno que no la tiene y después para uno que sí la tiene.

A).- Si

M=(m1,m2)	Espacio del mensaje.
C=(c1,c2,c3)	Espacio del criptograma.
P(m1)=P(m2)=1/2	La probabilidad de que m1 sea transmitida es igual a la probabilidad de que m2 sea transmitida y es igual a 1/2.
t1(m1)=c1	Transformación 1 sobre el mensaje 1 da como resultado el criptograma 1.
t1(m2)=c2	Transformación 1 sobre el mensaje 2 da como resultado el criptograma 2.
t2(m1)=c1	Transformación 2 sobre el mensaje 1 da como resultado el criptograma 1.
t2(m2)=c3	Transformación 2 sobre el mensaje 2 da como resultado el criptograma 3.
P3(m1)=0	Probabilidad de que el mensaje 1 fue transmitido, dado que el criptograma 3 fue recibido por el criptoanalista, es igual a cero.

como $P3(m1) = 0$ y $P(m1) = 1/2 \Rightarrow P3(m1) \neq P(m1)$

El sistema no tiene una seguridad perfecta.

B).- Siendo :

m1 = "si";

m2 = "no";

P(m1)=p(m2)=1/2 La probabilidad de transmitir cualquiera de los dos mensajes "si" o "no" es igual a 1/2.

t1(m1) = c1;

t1(m2) = c2;

t2(m1) = c2;

t2(m2) = c1;

P1(m1) = P1(m2);

Al interceptar el criptoanalista el criptograma c1, sabe que este criptograma es el resultado de dos transformaciones t1(m1)=c1 y t2(m2)=c1; como ambas tienen la misma probabilidad, el interceptor no tiene base para determinar a partir de c1, cual de los dos mensajes "si" o "no" fue el transmitido, y por lo tanto, el sistema cifrador tiene una seguridad perfecta.

Las condiciones que se tienen que dar para que un sistema cifrador tenga seguridad perfecta los postuló Shannon y son los siguientes:

Si un sistema cifrador tiene el mismo número de mensajes, criptogramas y claves, el sistema tendrá una seguridad perfecta si y solo si :

a).- Para un mensaje m dado y un criptograma c dado hay una sola clave en la transformación t para transformar m en c, c=t(m).

b).- Todas las claves tienen la misma probabilidad de ser usadas.

III.4.- Sistema Cifrador de un solo paso.

Este es un sistema particular que tiene una gran influencia sobre sistemas cifrados modernos con seguridad perfecta; es conocido como sistema one-time pad (sistema de un solo paso), observar fig.(3.2).

Sea el mensaje M=m1,m2,m3,.....,mn el mensaje a ser encifrado. (En este mensaje cada mi es un carácter de un alfabeto y M es una palabra de dicho alfabeto).

En este sistema para cada pareja (ki,mi) el mezclador o algoritmo encifrador produce un carácter ci del criptograma c. Las funciones del mezclador son del tipo :

a).- ci = mi + ki(mod 26) Sumador módulo 26, para un alfabeto de 26 caracteres donde cada uno de estos caracteres es representado por un entero del 0 al 25,
ci $\in$ { 0....25 } y
ki $\in$ { 0....25 }

b).- $ci = mi + ki \pmod 2$ Sumador módulo 2 o compuerta OR exclusiva,para mensajes codificados en forma binaria.

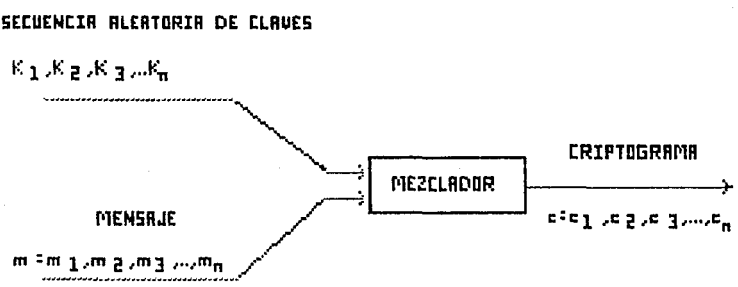


FIG.(3.2) SISTEMA CIFRADOR DE UN SOLO PASO

A partir del sistema cifrador de un solo paso, hay esencialmente tres tipos de sistemas cifradores modernos :

A).- Cifradores de bloques.

B).- Cifradores de cadenas.

C).- Cifradores de claves públicas.

III.5.-Cifradores de bloques.

En un cifrador de bloques, fig.(3.3), el mensaje m=m1m2m3...msms+1...m2sm2s+1... se particiona en bloques de tamaño S y para encifrar el bloque m1m2...ms se usa una llave K y S funciones f1,f2,...,fs, (normalmente diferentes), para obtener el criptograma c1c2...cs.
Usando la misma clave y las mismas funciones se encifra el siguiente bloque del mensaje ms+1ms+2 m2s.

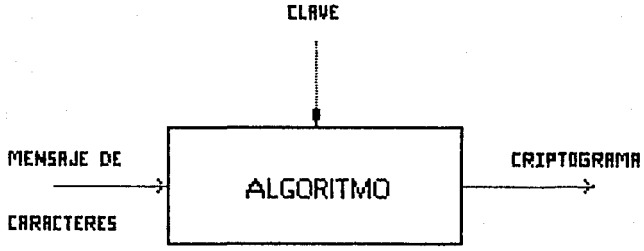


FIG.(3.3) CIFRADOR DE BLOQUES

Así el mensaje se encifra con S caracteres a la vez por lo que el criptograma es producido en bloques de S caracteres. En la práctica es deseable que las funciones fi sean complejas ya que le dan características de confusión y difusión al criptoanálisis. Confusión y difusión son conceptos introducidos por Shannon y se refieren a lo siguiente:

Confusión: Asegura que la relación entre un criptograma y su clave correspondiente es compleja, el objetivo es hacer más difícil el análisis estadístico que las características de la clave. Para asegurar esto, es deseable que el cifrado de todos los caracteres del mensaje dependan de la clave.

Difusión: Asegura una relación compleja entre el mensaje y el criptograma. Esto sirve para extender las estadísticas del mensaje sobre grandes porciones del criptograma. La idea es entonces asegurar que el criptoanalista necesita interceptar una gran parte del criptograma antes de que él pueda descifrarlo estadísticamente.

El sistema cifrador en el cual se basan actualmente los sistemas cifradores de bloques que hacen público el algoritmo encifrador, es el sistema conocido como sistema cifrador Feisel. En éste, el tamaño de cada bloque es 2n, cada bloque es dividido en dos partes iguales de tamaño n y escrito como m=(m0,m1). En este sistema cada clave define un conjunto de subclaves (k1,k2,...,kn) y cada subclave determina una transformación fki la cual mapea cada bloque de tamaño n (m0 o m1) en otro. El mensaje μ es entonces encifrado en h iteraciones usando las siguientes reglas, observar fig.(3.4):

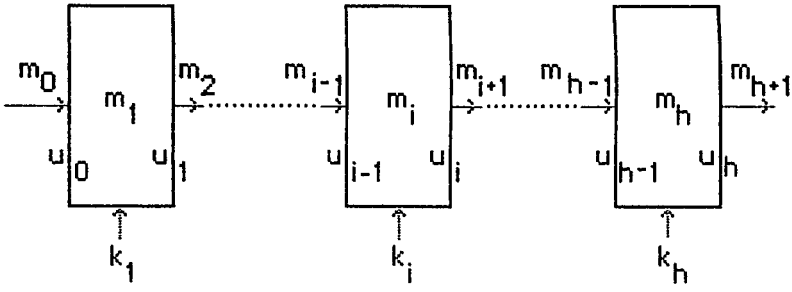


FIG.(3.4) SISTEMA CIFRADOR FEISEL

Iteración 1: $\mu 0 = (m0, m1) \longrightarrow \mu 1 = (m1, m2)$

Iteración i: $\mu i-1 = (mi-1, mi) \longrightarrow \mu i = (mi, mi+1)$

Iteración h: $\mu h-1 = (mn-1, mh) \longrightarrow \mu h = (mh, mh+1)$

Donde:

$mi+1 = mi-1 + fki(mi)$ para cada i>=1. El criptograma es entonces el block (mh,mh+1).

Para descifrar el criptograma nótese que la ecuación:

$$mi+1 = mi-1 + fki(mi) \quad \ldots\ldots (a)$$

también puede ser escrita como:

$$mi-1 = mi+1 + fki(mi) \quad \ldots\ldots (b)$$

Ya que todas las sumas son módulo 2.

Entonces, la ecuación (a) se usa en el proceso de encifrado y la ecuación (b) se usa en el proceso de descifrado que en nuestro ejemplo es:

Iteración 1: $\bar{\mu}h = (mh+1, mh) \longrightarrow \bar{\mu}h-1 = (mh, mh-1)$

Iteración h+1-i $\bar{\mu}i = (mi+1, mi) \longrightarrow \bar{\mu}i-1 = (mi, mi-1)$

Iteración h $\bar{\mu}1 = (m2, m1) \longrightarrow \bar{\mu}0 = (m1, m0)$

Una característica fundamental de un cifrado de bloques es que, desde que el mensaje es encifrado en bloques, un número, (igual a la longitud del bloque), de caracteres del mensaje son encifrados simultáneamente y dependientemente. Así cada carácter del criptograma depende de un número de caracteres del mensaje, y en el receptor, cada carácter del mensaje depende de varios caracteres del criptograma. Entonces, si hay un solo error en la transmisón del criptograma habrá varios errores en el mensaje transmitido. Este efecto de que un error cause varios errores es denominado error de propagación. La magnitud del problema que dan los errores de propagación dependen del tipo del mensaje y de las características físicas del sistema de comunicación. Por ejemplo, si el mensaje es un texto en inglés, la redundancia del lenguaje permite al receptor entender y corregir el mensaje, mientras que si el mensaje son transacciones monetarias de un banco, y a los errores de propagación le añadimos problemas en el canal, el mensaje recibido no podría ser usado.

III.6.- Cifradores de cadenas.

En este tipo de cifradores no se presentan los errores de propagación. La principal característica de este tipo de cifradores reside en que el encifrado de cada bit de datos es independiente del resto del mensaje. Aparte de los trabajos de Shannon, los factores importantes en el desarrollo de este tipo de sistemas cifradores fue el auge del uso de las computadoras y el auge del uso de la microelectrónica en los años 60s.
Un cifrador de cadena típico es el mostrado en la fig.(3.5).

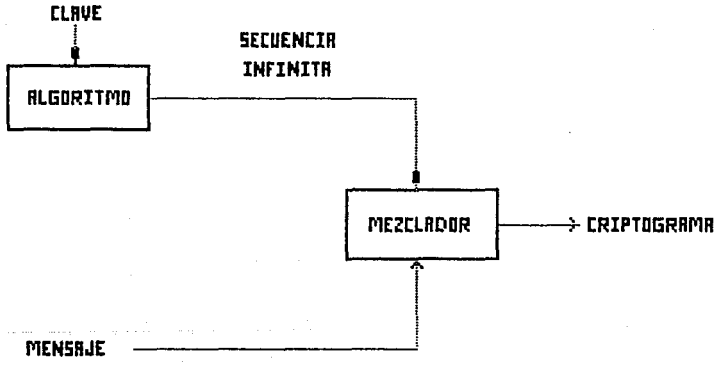


FIG.(3.5) CIFRADOR DE CADENAS

En este sistema la clave es alimentada en el algoritmo, usando éste la clave para generar una secuencia infinita (idealmente). Se hace referencia al algoritmo como el generador de cadena de llaves.

Los generadores de cadenas de llaves producen una sucesión de dígitos pseudoaleatorios. Una sucesión de dígitos pseudoaleatorios es una sucesión de dígitos en los que no hay una relación obvia entre ellos. Un generador de cadenas puede ser implementado a partir de un cifrador de bloques para que forme parte de un cifrador de cadenas, fig.(3.6).

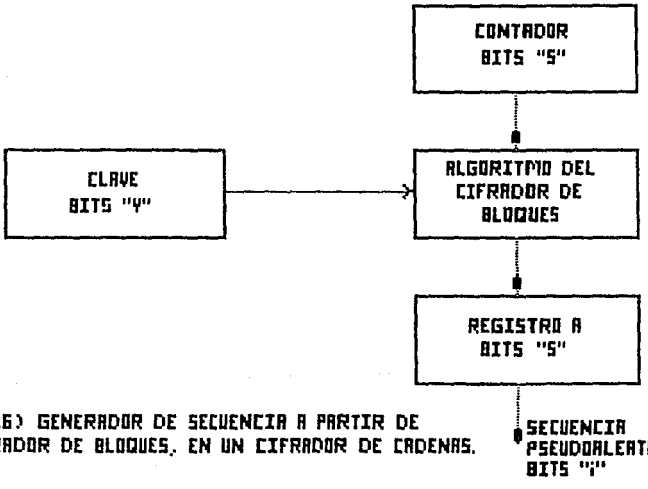


FIG. (3.6) GENERADOR DE SECUENCIA A PARTIR DE UN CIFRADOR DE BLOQUES, EN UN CIFRADOR DE CADENAS.

En esta realización, la entrada de datos al algoritmo es remplazada por un contador, el cual se incrementa cada vez que el cifrador de bloques y el registro cumplen con un ciclo de reloj. Después de que los bits "y" de la clave entran al algoritmo, éste último produce los bits "s" y los almacena en el registro A. Cada vez que el contador es incrementado se obtienen nuevos bits "s" en el registro A. Para cualquier i≤s, cada vez podemos sacar i de esos bits y de esta forma obtenemos nuestra sucesión binaria.

Es claro que las propiedades de la sucesión de salida dependen del algoritmo cifrador de bloques y de la i seleccionada, pero con un buen diseño del algoritmo, la sucesión podría parecer aleatoria. La selección de i repercute sobre los requerimientos de velocidad, ya que repercute sobre el tiempo necesario para que el algoritmo cifrador de bloques produzca una salida. Por ejemplo si S=100, requerimos de 10 Kbit/sec y el algoritmo tiene una máxima velocidad de 100 S's/sec, entonces tomaríamos el valor de i=100 y tomaríamos los 100 bits de S que produce el cifrador de bloques. De otra manera, si nosotros requerimos 2K bit/sec, tomaríamos el valor de i=20, es decir, tomaríamos 20 de cada 100 bits de S.

Las sucesiones generadas por los algoritmos de sucesiones realizables son periódicas, así, si la sucesión sp=s0s1s2...sp con período igual a p (T=p), nosotros sabemos que esta secuencia se repetirá después de p términos:

Sp = s0; Sp+1 =s1; y para cualquier m Sm+p = sm.

Entonces lo aconsejable es que el período P-->∞ para que la

sucesión tienda a un comportamiento aleatorio y de esta forma se destruya las propiedades estadísticas del mensaje transmitido, y por tanto el criptoanalista no pueda usar el análisis estadístico para descifrar el mensaje original.

Si st es una sucesión binaria, una corrida es un conjunto de elementos de una secuencia que son idénticos y son precedidos o sucedidos por el símbolo diferente a la corrida. Por ejemplo, la sucesión:

01111000110

inicia con una corrida de un cero y le sigue una corrida de 4 unos, una corrida de 3 ceros, y finaliza con una corrida de un cero.

A la corrida de ceros se le denomina hueco(gap) y a la corrida de unos se le denomina bloque.

III.6.1.- Función de Autocorrelación.

Sea st una sucesión binaria de periodo "p" y "a" un desplazamiento constante. Comparando los primeros p términos de st con los p términos de st+a, A sería el número de posiciones en las cuales estas dos sucesiones son iguales y D sería el número de posiciones en las cuales no hay coincidencia. Entonces, la función de autocorrelación C(a) está definida por:

$$C(a) = (A-D/p) \quad y \quad C(a+p) = C(a) \quad \text{para } 0 \leq a \leq p .$$

Si a=0:

$A=p$ y $D=0$ $\Rightarrow C(0)=1$ se tiene en fase la autocorrelación

III.6.2.- Registros de Corrimiento.

Los registros de corrimiento son comúnmente usados para la implementación de la sucesión de claves en los cifradores de cadenas, ya que estos son fáciles de obtener en el mercado (a muy bajo precio), además de que hay técnicas matemáticas y estadísticas para analizar las sucesiones que éstos generan y, en consecuencia, se puede evaluar el nivel de seguridad del sistema de que ellos forman parte.

Un registro de corrimiento de n localidades consiste en n elementos de almacenamiento binarios conectados en serie. El contenido de cada localidad del registro cambia cada ciclo de reloj de acuerdo a la siguiente regla:

Sean s0,s1,.....sn-1 las localidades binarias del registro de corrimiento.

Si(t) denota el contenido de si después del pulso t del reloj: entonces:

$$si(t+1)=si+1(t) \quad \text{para } i=0,...n-2 \text{ y}$$

$$sn-1(t+1)=f(s0(t),s1(t),...,sn-1(t)).$$

La función f es llamada la función de retroalimentación del registro.

Si

si=s0(i) para toda i donde 0≤i≤n-1 la sucesión si es completamente determinada por s0,s1,...,sn-1 y por la función de retroalimentación f.

Si

$$f(s0(t),s1(t),...,sn-1(t))=\sum_{i=0}^{n-1} CiSi(t) \mod 2$$

Donde Ci=0 o Ci=1, se hace referencia al registro de corrimiento con retroalimentación lineal, donde las constantes c0,c1,...cn-1 son denominadas coeficientes de retroalimentación. Los valores de los coeficientes serán iguales a 1 para una conexión cerrada y 0 para una conexión abierta (observar la fig.(3.7)).

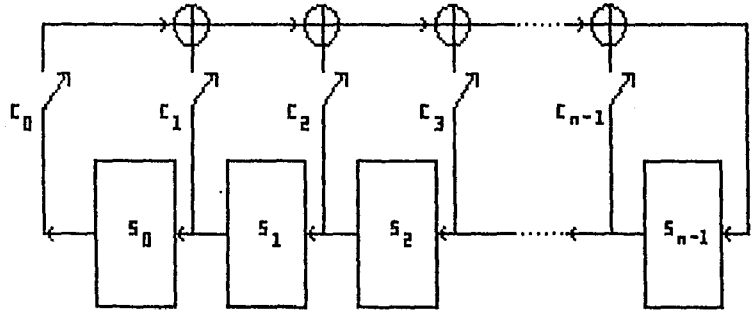


FIG.(3.7) REGISTRO DE CORRIMIENTO CON REALIMENTACION LINEAL

Al contenido de un registro de corrimiento en un tiempo dado to se le denomina estado, (esto es, a la sucesión binaria formada por las componentes), y puede ser representado por un vector binario de longitud n o como un número binario con rango de 0 a 2^{n-1}.

Consideremos un registro de corrimiento de cinco localidades con f=s0+s3, (ver la fig.(3.8)).

Si el estado inicial del registro, para t=0 es 01010, el estado para t=1 será 1010f(0,1,0,1,0) donde f(0,1,0,1,0)=0+1=1, por lo que el estado para t=1 es 10101.

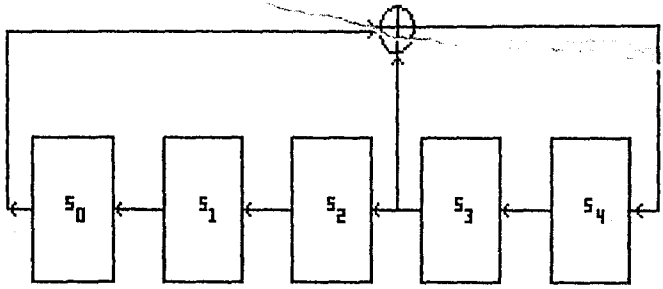


FIG. (3.8) REGISTRO DE CORRIMIENTO CON CINCO LOCALIDADES, CON FUNCION DE RETROALIMENTACION $f = S_0 + S_3$

La secuencia de estados en el registro de corrimiento se muestra en la tabla(3.6.2). Observando la tabla, la primera repetición del estado t=0 ocurre cuando t=31, entonces la sucesión de estados tiene un período 31 ($=2^5-1$).

t	estado	t	estado	t	estado	t	estado
0	01010	8	01100	16	11100	24	10010
1	10101	9	11000	17	11001	25	00100
2	01011	10	10001	18	10011	26	01000
3	10111	11	00011	19	00110	27	10000
4	01110	12	00111	20	01101	28	00001
5	11101	13	01111	21	11010	29	00010
6	11011	14	11111	22	10100	30	00101
7	10110	15	11110	23	01001	31	01010

Tabla (3.6.2). Secuencia de estados para el registro de corrimiento de la fig.(3.8).

Para un registro de corrimiento de n localidades, hay 2^n diferentes estados iniciales por lo que hay 2^n diferentes secuencias binarias periódicas.

III.6.3.- SISTEMA CIFRADOR DES (Data Encryption Standard).

Los sistemas DES son sistemas cifradores de bloques en los cuales se da a conocer el algoritmo cifrador como estandar. Al tener un algoritmo encifrador estandar se cuenta con tres ventajas:

a).- Si un chip o un conjunto de chips son diseñados para implementar el estandar, los costos se ven reducidos.

b).- La existencia de un estandar incrementa el número de usuarios del sistema cifrador ya que al convertirse un algoritmo encifrador en estandar, da confiabilidad a la seguridad del sistema.

c).- Existe una compatibilidad entre diversos sistemas de comunicación que usan el sistema cifrador DES.

En el diseño de sistemas DES se tienen que tomar en cuenta las siguientes observaciones:

a).- La seguridad del sistema dependerá de la problemática que tenga el criptoanalista para determinar la clave usada.

b).- Si un criptoanalista puede interceptar parte de una comunicación en un determinado tiempo, entonces el criptoanalista podrá interceptar otras comunicaciones.

c).- Si el criptoanalista conoce la representación en plaintext de una parte de un criptograma, por deducción podrá conocer el mensaje completo.

III.7.- Sistemas de clave publica.

La idea básica en los sistemas de clave pública se refiere a que todos los usuarios del sistema tengan un par de claves, una clave secreta (privada) y otra clave hecha pública. La clave pública se utiliza en el algoritmo encifrador para convertir el mensaje en criptograma, mientras que la clave privada se utiliza en el algoritmo descifrador para convertir el criptograma de nuevo al mensaje original. La idea de estos tipos de sistemas se basan en el sistema creado en 1978 por Rwest, Shamir y Ademan, que es conocido como sistema RSA.

En este sistema el mensaje se divide en bloques codificados entre el 0 y n-1. Donde n es un número entero que conforma la clave pública. Siendo mi el bloque i del mensaje, donde $0 \leq i \leq n-1$, el correspondiente bloque i del criptograma, denominado Ci, se obtiene mediante la fórmula:

$$Ci = m_i^h \pmod{n}$$

La fórmula inversa para obtener el bloque i del mensaje a partir del bloque i del criptograma es:

$$mi = C_i^d \pmod{n}$$

donde d es la clave privada.

Un requerimiento para complicar la labor del criptoanalista que desea encontrar d, es hacer que h, n y la propia d sean números enteros grandes. Si bien es cierto que n es del dominio público, pero n=p.q, donde p y q son números grandes enteros y primos, los cuales, son secretos y se presentan en las siguientes igualdades que sirven para elegir h y d:

$$n = p\ q$$

$$\psi = (p-1)(q-1)$$

$$hd \bmod \psi = 1 \qquad \text{(se elige h y se encuentra d).}$$

Como ejemplo:

sean

i=0; m=2; p=5; q=11; n=p.q=(5)(11)=55;

$\psi = (5-1)(11-1) = 40$;

si h=7; d=23 : (h)(d)=161

161 mod 40=1

para obtener el bloque del criptograma:

$$Ci = m_i^h \pmod n$$

$$C_o = m_o^7 \bmod 55$$

$$C_o = 2^7 \bmod 55 = 18$$

para recuperar el bloque del mensaje asociado:

$$mi=C_i^d \pmod n$$

$$m_o=18_o^{23} \bmod 55$$

$$m_o=18^1 18^2 18^4 18^{16}$$

$$m_o=18 \cdot 49 \cdot 36 \cdot 26 \bmod 55$$

$$m_o=2$$

IV.1.- Introducción.

Para cifrar las señales de voz se puede operar en el dominio del tiempo, en el dominio de la frecuencia o en las amplitudes que conforman la señal.

El procesamiento al que se somete una señal analógica de voz para cifrarse digitalmente se muestra en el sistema de la fig.(4.1).

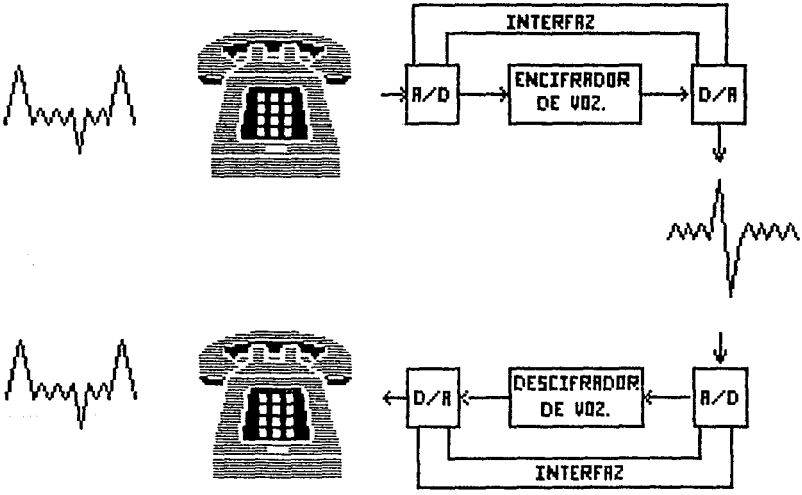


FIG. (4.1). SISTEMA CIFRADOR DE VOZ.

Como se observa, en la parte digital de este sistema se realiza el procesamiento de cifrado al que es sometida la señal de voz. El procesamiento digital de señales analógicas, ha tenido un gran auge debido al gran desarrollo tecnológico que ha tenido la computación y la electrónica digital, con el diseño de microprocesadores para el procesamiento digital de señales se puede: diseñar filtros digitales, aplicar algoritmos que requieren de intensivos cálculos matemáticos, aplicar transformadas rápidas de Fourier (FFT), etc., todo esto en tiempo real. De ahí la importancia que tiene el procesamiento digital de señales.

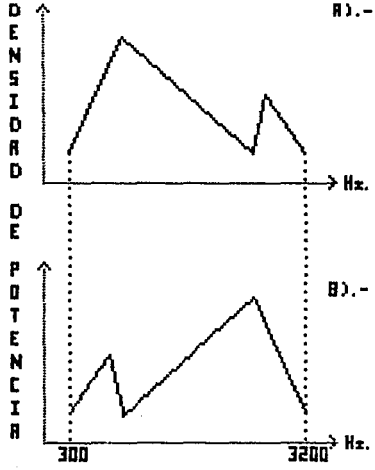
En este capítulo se presentará la teoría de los métodos más importantes de criptografía digital de voz, los principios de la criptografía digital de voz son los principios de la criptografía digital de datos mostrados en el capítulo anterior.

IV.2.- Criptografía en el dominio de la frecuencia.

IV.2.1.- Inversión en Frecuencia.

La inversión en frecuencia es un sistema clásico de cifrado que empezó a utilizarse en la primera mitad de este siglo, actualmente ya no se usa como un sistema cifrador, ya que presenta un nivel bajo de seguridad, pero se utiliza en adición a otras técnicas modernas de cifrado.

Su implementación resulta más sencilla de lo que pareciera, como se verá más adelante. La inversión en frecuencia, como literalmente se especifica, es mover las componentes de frecuencia altas de la señal a las correspondientes frecuencias bajas, y las componentes de frecuencias bajas a las correspondientes frecuencias altas, esta acción se presenta en la fig.(4.2).



A).- DENSIDAD ESPECTRAL DE POTENCIA DE LA SEÑAL DE VOZ ORIGINAL.

B).- DENSIDAD ESPECTRAL DE POTENCIA DEL CRIPTOGRAMA (SEÑAL CON FRECUENCIA INVERTIDA)

FIG. (4.2) INVERSION EN FRECUENCIA.

En un principio se pensaría que, para obtener la frecuencia invertida de una señal discreta de n muestras, primero se obtendría la transformada discreta de Fourier (DFT) de las n muestras de la señal, lo cual nos daría n componentes de frecuencia, y después se realizaría el proceso de inversión de la siguiente manera:

No. de componentes en frecuencia.	nueva posición de la inversión.
0	N-1
1	N-2
2	N-3
⋮	⋮
N-3	2
N-2	1
N-1	0

Pero la acción anterior no tiene el efecto esperado, esto se comprueba a partir de las dos siguientes aseveraciones:

a).- La inversión de la DFT de una señal, excluyendo la primera componente, causa una correspondiente inversión en el tiempo de la señal, excluyendo la primera muestra.

b).- La inversión de la frecuencia analógica se obtiene si las muestras impares de la señal son multiplicadas por -1.

Por lo tanto, la inversión de la DFT de una señal no tiene ningún uso para la encripción de voz. Ya que no representa la inversión de la frecuencia analógica.

Justifiquemos analíticamente las dos aseveraciones anteriores:

a).- Sea: $x(0),x(1),\ldots,x(N-1)$ las N muestras de una señal en el tiempo discreto

y

$X(0),X(1),\ldots,X(N-1)$ las N componentes de la DFT de la señal.

con N=4:

La transformación $x(i) \longrightarrow X(i)$ para $0 \leq i \leq 3$ está dada por:

$$F = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 \end{bmatrix}$$

donde:

$$W = e^{-j\frac{2\pi}{N}} \quad \text{con } N=4 \quad \rightarrow \quad W = e^{-j\frac{2\pi}{4}} = e^{-j\frac{\pi}{2}}$$

$$W = \cos\frac{\pi}{2} - j\,\text{sen}\frac{\pi}{2} = -j$$

y la transformación $X(i) \longrightarrow x(i)$ para $0 \leq i \leq 3$

esta dada por:

$$F^{-1} = \frac{1}{N}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^{-1} & W^{-2} & W^{-3} \\ 1 & W^{-2} & W^{-4} & W^{-6} \\ 1 & W^{-3} & W^{-6} & W^{-9} \end{bmatrix}$$

ya que:

$$X = Fx$$

$$x = F^{-1}Fx \qquad F^{-1}F = I$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W & W^{2} & W^{3} \\ 1 & W^{2} & W^{4} & W^{6} \\ 1 & W^{3} & W^{6} & W^{9} \end{bmatrix} \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^{-1} & W^{-2} & W^{-3} \\ 1 & W^{-2} & W^{-4} & W^{-6} \\ 1 & W^{-3} & W^{-6} & W^{-9} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = F^{-1}F$$

con:

$$x = \begin{bmatrix} 4 \\ 2 \\ 1 \\ 3 \end{bmatrix} \qquad P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

invirtiendo en el tiempo:

$$x_{it} = Px = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 4 \\ 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 1 \\ 2 \end{bmatrix}$$

invirtiendo en la frecuencia:

$$x_{if} = F^{-1}PFx$$

$$X = Fx = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W & W^2 & W^3 \\ 1 & W^2 & W^4 & W^6 \\ 1 & W^3 & W^6 & W^9 \end{bmatrix} \begin{bmatrix} 4 \\ 2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 + 2 + 1 + 3 \\ 4 + 2W + W^2 + 3W^3 \\ 4 + 2W^2 + W^4 + 3W^6 \\ 4 + 2W^3 + W^6 + 3W^9 \end{bmatrix}$$

tomando los siguientes valores para las W:

$$W = -J \qquad W^2 = -1 \qquad W^3 = J \qquad W^4 = 1 \qquad W^6 = -1 \qquad W^9 = -J$$

tendremos la siguiente matriz:

$$X = F_x = \begin{bmatrix} 10 \\ 4-2J-1+3J \\ 4-2+1-3 \\ 4+2J+1-3J \end{bmatrix} = \begin{bmatrix} 10 \\ 3+J \\ 0 \\ 3-J \end{bmatrix}$$

$$PF_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 10 \\ 3+J \\ 0 \\ 3-J \end{bmatrix} = \begin{bmatrix} 10 \\ 3-J \\ 0 \\ 3+J \end{bmatrix}$$

$$x_{if} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^{-1} & W^{-2} & W^{-3} \\ 1 & W^{-2} & W^{-4} & W^{-6} \\ 1 & W^{-3} & W^{-6} & W^{-9} \end{bmatrix} \begin{bmatrix} 10 \\ 3-J \\ 0 \\ 3+J \end{bmatrix} =$$

$$= \begin{bmatrix} 10 + 3-J + 0 + 3 + J \\ 10 + (3-J)(-1/J) + 0 + (3+J)(1/J) \\ 10 + (3-J)(-1) + 0 + (3+J)(-1) \\ 10 + (3-J)(1/J) + 0 + (3+J)(-1/J) \end{bmatrix}$$

$$x_{if} = \frac{1}{4}\begin{bmatrix} 16 \\ 10 - 3/J + 1 + 3/J + 1 \\ 10 - 3 + J - 3 - J \\ 10 + 3/J - 1 - 3/J - 1 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 16 \\ 12 \\ 4 \\ 8 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \\ 1 \\ 2 \end{bmatrix}$$

$$x_{if} = x_{it} = \begin{bmatrix} 4 \\ 3 \\ 1 \\ 2 \end{bmatrix}$$

La igualdad $x_{if} = x_{it}$, (inversión en el tiempo = inversión en frecuencia excluyendo la primera muestra en ambos dominios), se debe a la característica de simetría que tiene la DFT de una señal real, esto es: $|X(r)| = |X(N-r)|$ ya que:

$$^{*}X(r) = X(N-r) \qquad \text{para } r=0,1,2,\ldots,N-1$$

donde el pivote de simetría es la muestra N/2.

b).- Sí la inversión de la DFT no representa la inversión de la frecuencia analógica, veamos a través de la DFT como podemos representar dicha inversión de frecuencia analógica de una señal real.

sea:

Una señal periódica cada N muestras

N = número de muestras de la señal por periodo.

N_s = número de muestras por segundo.

$T=N/N_s$ Periodo de la señal.

$f \geq N_s/N$ Frecuencia fundamental de la señal.

entonces:

Del teorema del muestreo, nuestra señal de banda limitada tendrá un rango de 0 a $N_s/2$ Hz., y la componente de frecuencia más alta de la DFT corresponde a la armónica N_s/N, la cual ocurre a $1/2N_s$ Hz, dividiendo el ancho de banda entre la frecuencia fundamental, resulta:

$$(N_s/2)/(N_s/N) = N/2$$

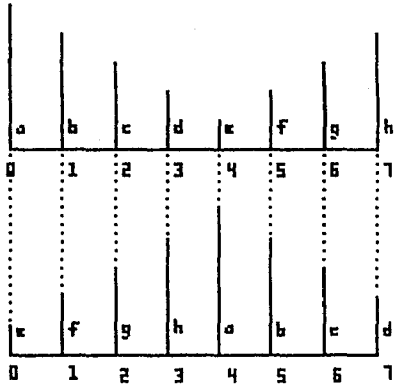
Donde N/2 es la armónica de mayor frecuencia y a la vez el pivote de simetría:

X(0) Corresponde a la componente espectral de 0 Hz.

X(1) Corresponde a la componente espectral de N_s/N Hz.

X(N/2) Corresponde a la componente espectral de $N_s/2$ Hz.

Por lo tanto, la inversión de frecuencia analógica de una señal de n muestras se obtiene al realizar una permutación cíclica a la derecha de N/2 muestras en la DFT, observar fig.(4.3).



A).- DFT CON N=8 PARA UNA SEÑAL REAL.

B).- INVERSION DE FRECUENCIA DE A).

FIG. (4.3) INVERSION DE FRECUENCIA ANALOGICA UTILIZANDO DFT.

Esta permutación cíclica se realiza al multiplicar la DFT por la matriz de permutación P dada por:

$$P = \left[\begin{array}{cccccccc} 0 & 0 & 0\ldots\ldots & 0 & 1 & 0 & 0\ldots\ldots & 0 \\ 0 & 0 & 0\ldots\ldots & 0 & 0 & 1 & 0\ldots\ldots & 0 \\ 0 & 0 & 0\ldots\ldots & 0 & 0 & 0 & 1\ldots\ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \; \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots \; \vdots & \vdots & \vdots & \vdots & \ddots \; \vdots \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \begin{array}{l} \left.\right\} N/2 \\ \\ \left.\right\} N/2 \end{array}$$

$$\underbrace{\qquad\qquad}_{N/2} \; \underbrace{\qquad\qquad}_{N/2}$$

siendo: N=4 y

$$x = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$$

$$x_{if} = F^{-1}PFx$$

$$F^{-1}P = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^{-1} & W^{-2} & W^{-3} \\ 1 & W^{-2} & W^{-4} & W^{-6} \\ 1 & W^{-3} & W^{-6} & W^{-9} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ W^{-2} & W^{-3} & 1 & W^{-1} \\ W^{-4} & W^{-6} & 1 & W^{-2} \\ W^{-6} & W^{-4} & 1 & W^{-3} \end{bmatrix}$$

$$F^{-1}PF = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ W^{-2} & W^{-3} & 1 & W^{-1} \\ W^{-4} & W^{-6} & 1 & W^{-2} \\ W^{-6} & W^{-4} & 1 & W^{-3} \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W^{1} & W^{2} & W^{3} \\ 1 & W^{2} & W^{4} & W^{6} \\ 1 & W^{3} & W^{6} & W^{9} \end{bmatrix} =$$

$$=\frac{1}{4}\begin{bmatrix} 4 & 1+W^{1}+W^{2}+W^{3} & 1+W^{2}+W^{4}+W^{6} & 1+W^{3}+W^{6}+W^{9} \\ W^{-2}+W^{-3}+1+W^{-1} & W^{-2}+W^{-2}+W^{2}+W^{2} & W^{-2}+W^{-1}+W^{4}+W^{5} & W^{-2}+1+W^{+6}+W^{+8} \\ W^{-4}+W^{-6}+1+W^{-2} & W^{-4}+W^{-5}+W^{2}+W & W^{-4}+W^{-4}+W^{4}+W^{4} & W^{-4}+W^{-3}+W^{6}+W^{7} \\ W^{-6}+W^{-9}+1+W^{-3} & W^{-6}+W^{-3}+W^{2}+1 & W^{-6}+W^{-7}+W^{4}+W^{3} & W^{-6}+W^{-6}+W^{6}+W^{6} \end{bmatrix}$$

$$F^{-1}PF=\frac{1}{4}\begin{bmatrix} 4 & 1-j-1+j & 1-1+1-1 & 1+j-1-j \\ -1+1/j+1-1/j & -1-1-1-1 & -1-1/j+1-j & -1+1-1+1 \\ 1-1+1-1 & 1-1/j-1-j & 1+1+1+1 & 1+1/j-1+j \\ -1-1/j+1+1/j & -1+1-1+1 & -1+1/j+1+j & -1-1-1-1 \end{bmatrix}$$

como: $-1/j - j = 0$ y $1/j + j = 0$

$$F^{-1}PF = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

$$x_{if} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} = \begin{bmatrix} 2 \\ -3 \\ 4 \\ -5 \end{bmatrix}$$

Por lo tanto comprobamos que para invertir la frecuencia de una señal de N muestras basta con multiplicar por -1 las muestras impares de la señal.

IV.2.2.- Reordenadores de Bandas.

En la técnica de este tipo de sistemas cifradores, al obtener el espectro de frecuencia del bloque del mensaje de voz a encifrar, se le divide en sub-bandas, las cuales, son reordenadas para obtener el espectro del correspondiente bloque del criptograma. En un caso más sofisticado algunas de las sub-bandas deben ser invertidas. La fig.(4.4) ilustra un ejemplo para un bloque de un mensaje de voz de 5 sub-bandas.

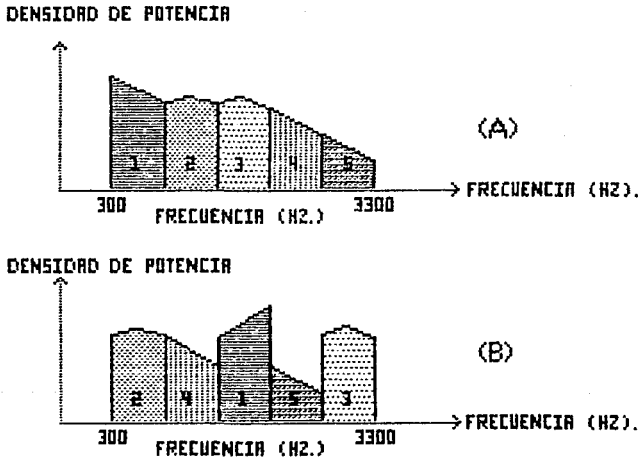


FIG. (4.4) CORTES DE BANDA. (A) ORDEN ORIGINAL DE LAS SUB-BANDAS (B) ESPECTRO REORDENADO.

IV.2.2.a).- Diseño de reordenadores de bandas a través de moduladores balanceados (BM).

Como ejemplo, sea el mensaje de voz una señal de banda limitada con ancho de banda de 3 KHz. en el rango de 300 a 3300 Hz., utilizando 5 sub-bandas hay 5! posibles reordenamientos y 2^5 decisiones para cuántas y cuáles sub-bandas serán invertidas. Por lo tanto, hay 5! x 2^5=3840 posibles reordenamientos de un bloque, siendo el ancho de banda de la señal 3000 Hz., el ancho de banda de las sub-bandas será de 3000/5=600 Hz., la implementación del reordenador de bandas es mostrado en la fig.(4.5).

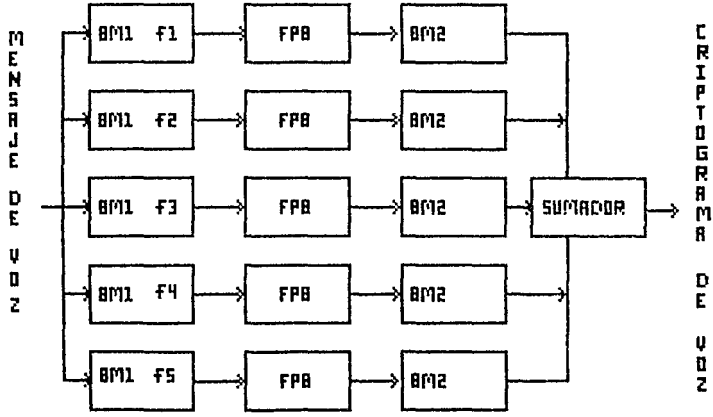


FIG. (4.5) REORDENADOR DE BANDAS EN FRECUENCIA A TRAVES DE MODULADORES BALANCEADOS

En este sistema se requerirá de excelentes filtros paso-bandas con un ancho de banda de 600 Hz. centrados en 10 KHz. El principio de funcionamiento es el siguiente:

1).- Modular el mensaje de voz en los moduladores BM1, y con los corrimientos de frecuencia f1,f2,f3,f4,f5, todas las sub-bandas quedarán centradas en 10 Khz., esto se muestra en la fig.(4.6).

2).- Filtrar la señal en filtro paso-bajas (FPB), y así, en la salida de cada uno de los filtros, habrá una sub-banda diferente.

3).- Modular cada una de las sub-bandas en BM2 con corrimientos de frecuencia f_{i1},f_{i2},f_{i3},f_{i4},f_{i5}, con lo cual se llevará a cada una de las sub-bandas a una banda diferente de la cual se encontraba.

4).- Sumar las sub-bandas.

La tabla (4.1) muestra los corrimientos de frecuencia requeridos para cada una de las sub-bandas.

	Corrimiento de frecuencia.	
No. Sub-banda.	Sin inversión (Khz.)	Con inversión (Khz.)
1	9.4 f_1, fi_1	10.6 f_1
2	8.8 f_2, fi_2	11.2 f_2
3	8.2 f_3, fi_3	11.8 f_3
4	7.6 f_4, fi_4	12.4 f_4
5	7.0 f_5, fi_5	13.0 f_5

Tabla (4.1). Corrimientos de frecuencias para un reoerdenador de 5 sub-bandas.

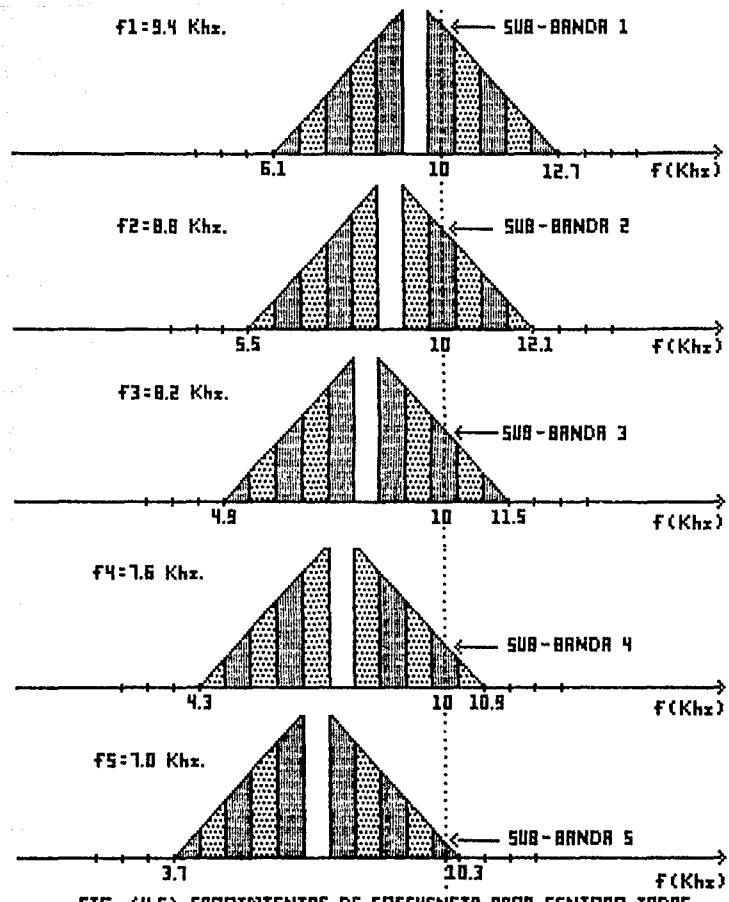


FIG. (4.6) CORRIMIENTOS DE FRECUENCIA PARA CENTRAR TODAS LAS SUB-BANDAS A 10 Khz.

El mayor corrimiento de frecuencia utilizado cuando hay inversión, se debe a que trabajamos con la parte baja de la banda entregada por el modulador, mientras que cuando no hay inversión, trabajamos en la parte alta de la banda. La fig.(4.7) muestra la densidad espectral de potencia de la salida de un modulador balanceado.

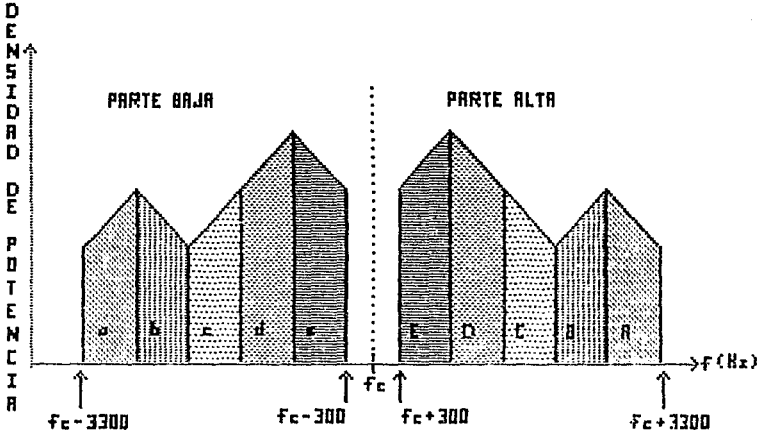


FIG. (4.7) DENSIDAD ESPECTRAL DE POTENCIA DE LA SALIDA DE UN MODULADOR BALANCEADO.

Como se observa en la figura, la parte alta y la parte baja son simétricas respecto a un espejo sobre f_c, (frecuencia de corrimiento). Si a cada parte la dividimos en sub-bandas de 600 Hz. obtendremos las sub-bandas **a**, b, c, d y **e** en la parte baja y las sub-bandas A, B, C, D y E en la parte alta. Como puede observarse, las sub-bandas de la parte baja son respectivamente iguales a las sub-bandas de la parte alta, pero invertidas, esto es para nuestro ejemplo, si al espectro entregado por el modulador además del corrimiento de 9.4 Khz., le agregamos un corrimiento de 1.2 Khz., en el rango de frecuencias de 9.7 - 10.3 Khz. tendremos la sub-banda 1, pero ésta invertida, observar la tabla (4.1). Para recorrer las sub-bandas a la frecuencia 9.7- 10.3Khz., se utilizan los corrimientos f_1, f_2, f_3, f_4, f_5, de inversión o de no inversión, según sea el caso, pero para asignarles su nueva posición a las sub-bandas en el rango 300 - 3300 Khz., se usarán los corrimientos f_{i1}, f_{i2}, f_{i3}, f_{i4}, f_{i5}, los cuales tendrán valores unívocos de los corrimientos sin inversión y negativos. Esto es, el ejemplo de la fig.(4.4) se genera a través del proceso señalado en la tabla (4.2):

No. Sub-banda.	Esta invertida Si/No.	Corrimiento en KHz. a la banda 9.7 - 10.3 KHz.	Reordenamiento corrimiento neg.en KHz.a la banda 0.3 - 3.3 KHz.
1	Si	f_1=10.6	f_{i1}=f_2=8.8
2	No	f_2=8.8	f_{i2}=f_4=7.6
3	Si	f_3=11.8	f_{i3}=f_1=9.4
4	No	f_4=7.6	f_{i4}=f_5=7.0
5	Si	f_5=13.0	f_{i5}=f_3=8.2

Tabla (4.2). Proceso de generación del ejemplo de la fig.(4.4).

La selección de los corrimientos f_{is} en el reordenamiento utilizan una clave, la cual puede ser utilizada de tres formas:

a).- Usar una clave para seleccionar un reordenamiento particular.

b).- Usar una clave para seleccionar un conjunto de reordenamientos y entonces utilizar este conjunto secuencialmente.

c).- Usar una clave que inicializará un generador de números pseudoaleatorios, el cual seleccionará un reordenamiento.

Para implementar la forma en que se utilizará la clave se recurre a las formas vistas en el capitulo III.

IV.2.3.- Reordenadores de espectro de frecuencia utilizando DFT.

Este tipo de sistemas realizan un procesamiento digital de señales muy rápido y son considerados como la generalización de los sistemas reordenadores de bandas. Estos sistemas, dada la velocidad que requieren para un procesamiento en tiempo real, necesitan de recursos de cómputo especializados para el procesamiento digital de señales.

El principio de funcionamiento de los sistemas reordenadores de espectros es el siguiente:

A).- Se tiene un bloque del mensaje de n muestras.

B).- Se le aplica DFT al mensaje.

C).- Se reordena la DFT.

D).- A la DFT reordenada se le aplica DFT^{-1} para obtener el criptograma a transmitir.

Como el mensaje es una señal real, se requiere que el criptograma también sea una señal real, lo cual nos da restricciones en el reordenamiento de la DFT.

Como se sabe, la DFT de una señal real cumple con las reglas de simetría:

Siendo

$x[r]$ el conjunto de muestras de la señal y

$X[r]$ el conjunto de muestras de la DFT de

$x[n]$ para $r = 0,1,...,N-1$.

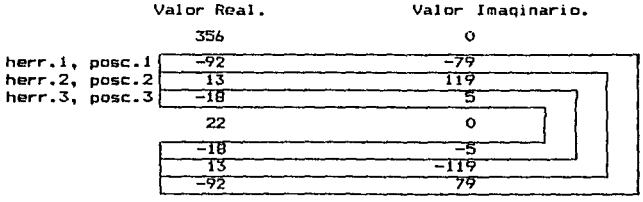
$|X(r)| = |X(N-r)|$ y

$X(N-r)^*$

Entonces, para la DFT reordenada se deben seguir cumpliendo estas reglas. Sea por ejemplo una señal de N=8:

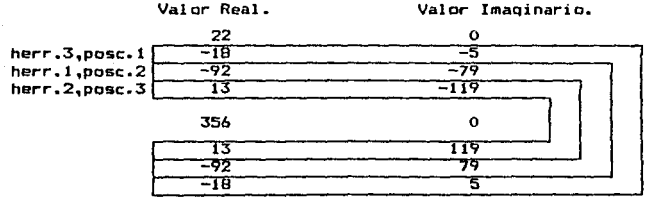
N	Valores Discretos de la señal.	DFT Asociada.		
		Amplitud	Valor Real	Valor Imag.
1	23	356.00	356	0 i
2	45	121.94	-92	-79 i
3	23	119.71	13	119 i
4	12	18.10	-18	5 i
5	78	22.00	22	0 i
6	98	18.10	-18	-5 i
7	65	119.71	13	-119 i
8	12	121.94	-92	79 i

Como podrá observarse, el par de valores de la DFT que sirven como eje de simetría son los correspondientes a N=5:



Las únicas muestras que no estan asociadas a una herradura son la 1 y la 5 que son respectivamente la 1a. y la del eje de simetría, y como se observa, ninguna de estas tiene componente imaginaria, y solo pueden intercambiarse entre sí. Mientras que el reordenamiento de las muestras restantes se hace al cambiar de posición las herraduras, por ejemplo:

DFT REORDENADA:



como se muestra:

— La herradura 1 fue reordenada a la posición 2 sin voltearla.

— La herradura 2 fue reordenada a la posición 3 volteándola.

— La herradura 3 fue reordenada a la posición 1 volteándola.

— La muestra 1 fue intercambiada con la muestra 5.

IV.3. Criptografía en el dominio del tiempo.

IV.3.1.- Inversión en el tiempo.

En los sistemas cifradores que utilizan la técnica de inversión segmentada en el tiempo, las muestras del mensaje de voz son agrupadas en segmentos de tiempo que son almacenados en memoria, cada vez que se completan las muestras de un segmento, éstas son entregadas al convertidor D/A en orden invertido, obteniéndose de esta forma las correspondientes muestras del criptograma. Esta técnica es ilustrada en la fig.(4.8). En estos sistemas cifradores el tamaño de los segmentos se adecúa de acuerdo a los requerimientos del sistema, evidentemente los segmentos grandes causan grandes retrasos de tiempo, pero en contraste proveen bajos residuos de inteligibilidad. En este tipo de sistemas el mecanismo de procesar la señal de voz en tiempo real, es sencillo de explicar:

mensaje original

..A11....A1N A21....A2N A31....A3N A41....A4N

retraso de encifrado

criptograma

ruido

- - - - - - - -..A1N....A11 A2N....A21 A3N....A31 A4N....A41

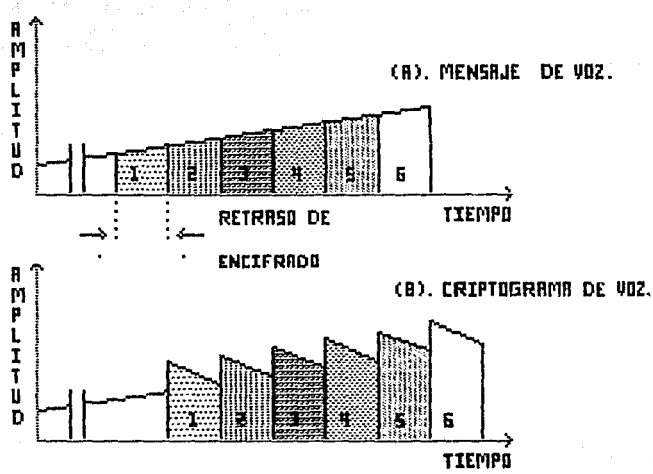


FIG. (4.8) INVERSION SEGMENTADA EN EL TIEMPO.

Como se muestra arriba, es suficiente tener almacenado dos segmentos de voz. Mientras en un segmento se van almacenando las muestras en el otro, se sacan en orden invertido.

Un sistema cifrador de este tipo puede ser implementado usando una memoria RAM como se muestra en la fig.(4.9), en este caso, el tamaño de la RAM restringe la longitud de tiempo para los segmentos. En este tipo de sistemas el método de encifrado y el de descifrado es idéntico. Ya que es claro que si nosotros ejecutamos inversión en el tiempo sobre una señal donde

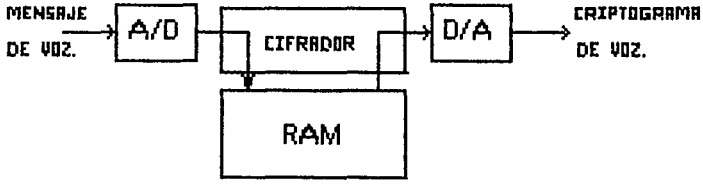


FIG. (4.9) CIFRADOR DE INVERSION DE SEGMENTOS EN EL TIEMPO

los segmentos ya están invertidos, (criptograma), entonces, dado que no cambiamos las longitudes de los segmentos que se usa el mismo periodo de tiempo, el transmisor y el receptor deben estar en completa sincronía.

La técnica de inversión en el tiempo ofrece una relativa seguridad, y como consecuencia, es concebida generalmente como una técnica de privacía de voz. Sin embargo, la simplicidad y bajo costo de su implementación, con un microprocesador y con un circuito sencillo, permite que ésta técnica se use en situaciones donde la seguridad no es un parámetro de estricta importancia.

Esta técnica no contiene ninguna llave y por lo tanto, se asemeja mas a un código antes que a un sistema cifrador.

Se puede introducir una llave para obtener una variación en el tamaño de los segmentos, pero, como ya vimos antes, el tamaño de la RAM restringe los posibles tamaños de los segmentos, de esta forma las longitudes de los segmentos están destinadas a ser limitadas a un rango pequeño. La seguridad extra obtenida al agregar una llave es limitada. Una razón para esto es que si un interceptor tiene un receptor ajustable a una longitud de segmento, que esté sobre el tamaño promedio de todas las posibilidades, entonces se puede obtener una señal que, aunque no se escuche perfectamente, exhibirá una muy alta inteligibilidad residual.

IV.3.2.- Reordenamiento de muestras en el tiempo.

Los sistemas cifradores que usan este método funcionan de la siguiente manera, observar la figura (4.10):

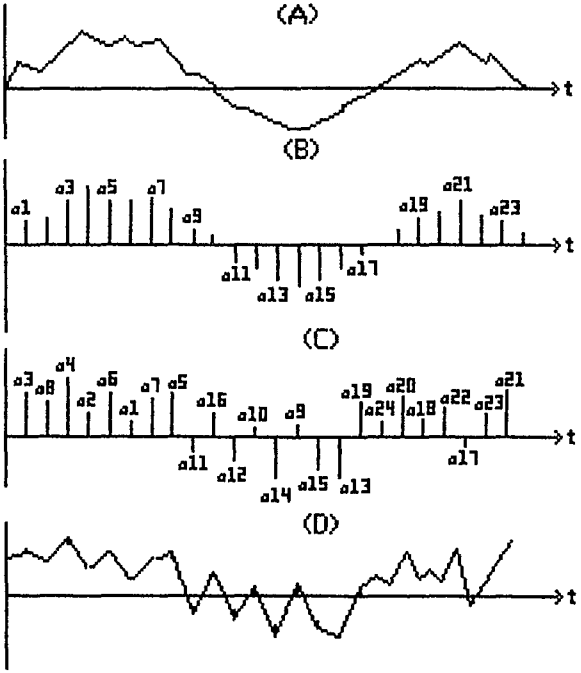


FIG. (4.10) REORDENAMIENTO DE MUESTRAS EN EL TIEMPO

a) Se recibe el mensaje análogico de voz.

b) A través de un convertidor A/D, se obtiene el mensaje digital de voz.

c) El mensaje digital de voz se transforma en el criptograma digital, esto se realiza reordenanado las muestras de cada bloque del mensaje.

d) A través de un convertidor D/A, se obtiene un criptograma analógico.

Para el ejemplo de la figura (4.10), los bloques fueron de 8 muestras y se usó repetitivamente la permutación:

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 6 & 4 & 1 & 3 & 8 & 5 & 7 & 2 \end{bmatrix}$$

Que puede implementarse con un registro de corrimiento.
En esta técnica, un simple par de permutaciones sistemáticas son suficientes para obtener bajos niveles de inteligibilidad.
Los dos principales inconvenientes para adoptar este metodo son: el acompañamiento de la extensión del ancho de banda y la integridad de las muestras individuales.

Como se observa en la figura (4.11), este tipo de sistemas cifradores tienen una etapa de prefiltrado para evitar que el ancho de banda de la señal procesada se expanda.

El problema de preservar la integridad de las muestras es considerablemente más difícil. Para un canal de comunicación real, es casi inevitable que la señal sea distorsionada.

Y estos niveles de distorsión llegan a ser significativamente más perceptibles cuando se presentan las discontinuidades en la operación inversa del reordenamiento.

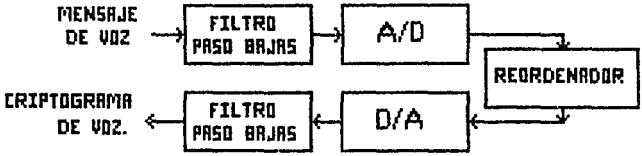


FIG. (4.11).REORDENAMIENTO DE MUESTRAS EN EL TIEMPO.

IV.3.3.- Permutación de bloques en el tiempo.

En esta técnica la señal analógica es dividida en períodos iguales de tiempo llamados marcos. Cada marco es entonces subdividido en pequeños períodos de tiempo llamados segmentos, obteniendose de esta forma el bloque de criptograma. Este proceso es ilustrado en la figura (4.12), en la figura (4.13) se muestran los marcos de

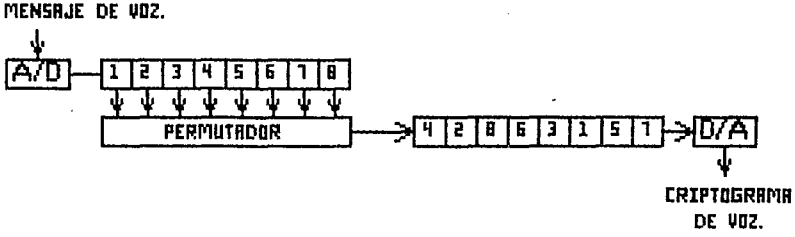


FIG.(4.12). PERMUTACION DE BLOQUES EN EL TIEMPO

mensaje de voz y su respectivo marco de criptograma de voz, para el proceso ilustrado en la figura (4.12).

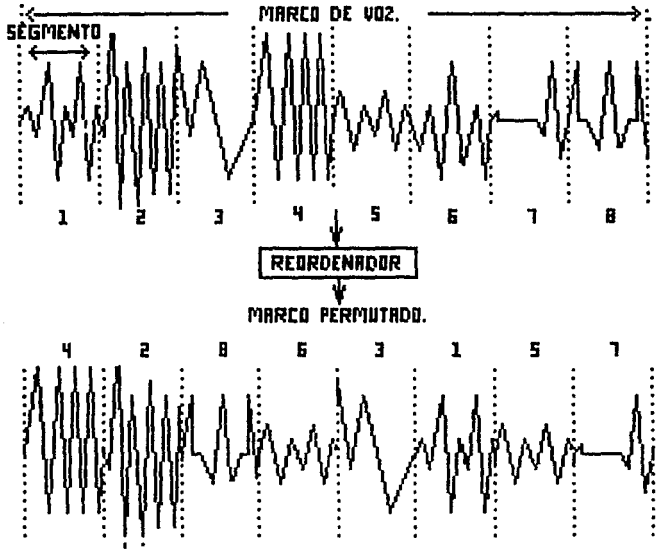


FIG. (4.13) PERMUTACION DE UN MARCO DE VOZ.

En el diseño de este tipo de sistemas cifradores es importante la elección de tres parámetros: longitud del segmento, longitud del marco y tipo de permutador. Veamos las consideraciones para estos parámetros.

Tamaño del segmento:

El tamaño del segmento debe ser lo suficientemente pequeño para que no contenga más de un fonema, pero, entre mas pequeño sea el segmento, habra mayores discontinuidades en la señal recuperada, lo cual provoca una expansión del ancho de banda, ya que estos súbitos cambios implican componentes de alta frecuencia.

Longitud de marco:

La longitud de marco afecta el retraso entre el mensaje de voz analógico transmitido y el mensaje de voz recibido. Tomando como ejemplo el proceso mostrado en la figura (4.12), la longitud del segmento es igual a T seg., entonces se requiere de 8T seg., para que los ocho segmentos que conforman un marco entren al parmutador, consecuentemente no se puede empezar a transmitir los segmentos hasta que no se encuentre el grupo de estos ocho segmentos en el permutador, para que de esta forma se transmita el marco de voz reordenado. Esto significa que el receptor no puede empezar a decifrar hasta que haya recibido los ocho segmentos, lo cual implica otro retraso de 8T seg. Por lo tanto, el retraso total para un segmento de voz desde la transmisión hasta la recepción es de 16T seg. La situación anterior es mostrada en la figura (4.14). En general, el tiempo de retraso total para un sistema con s segmentos por marco con una longitud por marco de T seg, es $2sT$ seg.

Desde el punto de vista de seguridad es aconsejable longitudes de marcos grandes, ya que si tenemos s segmentos por marco, entonces tenemos s! permutaciones posibles. Si s! permutaciones son pocas, evidentemente facilitamos la labor del criptoanalista. El número sugerido varía de 4 a 16 segmentos, que son los rangos usados en este tipo de sistemas.

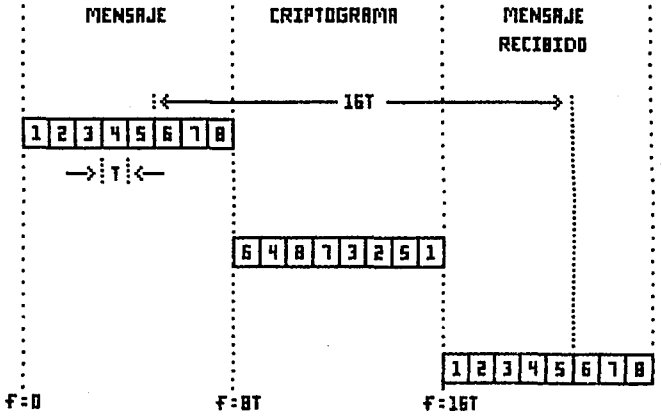


FIG.(4.14).DIAGRAMA DE TIEMPOS PARA UN MARCO DE VOZ.

Tipo de permutador:

Se puede tener una clave, la cual selecciona una permutación fija, esta permutación es usada para transmitir todos los marcos de voz del criptograma. Este tipo de selección es aconsejable para la transmisión de mensajes cortos.

De otra manera, para cada marco, se puede utilizar un generador de números pseudoaleatorios para seleccionar una permutación. Del ejemplo de la figura (4.12), si tenemos 8 segmentos por marca, el número total de permutaciones disponibles es de 8!=40,320, pero de este número de permutaciones no todas pueden ser utilizadas. Por ejemplo la permutación:

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 1 & 3 & 4 & 6 & 5 & 7 & 8 \end{bmatrix}$$

tiene un alto grado de inteligibilidad residual, por lo que después de algunas repeticiones del mensaje lo podríamos entender. Una forma para utilizar solo las "buenas" permutaciones es almacenando éstas en una memoria ROM. Posteriormente con un generador de secuencias se accesa la memoria ROM para obtener las permutaciones.

IV.4.- Criptografía en la amplitud.

El principio de funcionamiento de los cifradores de amplitud de señales de voz se basa en el enmascaramiento del mensaje, es decir, se refiere a la adición lineal de amplitudes pseudoaleatorias a las amplitudes de la señal de voz.
Un ejemplo de este tipo de sistemas cifradores usa la adición de amplitudes de ruido pseudoaleatorio utilizando la función módulo. En este método se asume que las amplitudes de las muestras del mensaje de voz caen dentro del intervalo (-A,A), asimismo, las muestras de amplitudes del ruido pseudoaleatorio están uniformemente distribuídas en el mismo intervalo.
A la muestra de amplitud del mensaje y a la muestra de amplitud del ruido pseudoaleatorio se le adiciona A:

$$M_A=M+A \qquad \text{(mensaje)}$$

$$R_A=R+A \qquad \text{(ruido)}$$

De tal forma, ambas muestras de amplitudes estarán en el intervalo (0,2A). Estas nuevas amplitudes son las que se usan para obtener el criptograma de voz; a la muestra de amplitud de ruido pseudoaleatorio R_A se le aplica la función módulo 2A, se le suma la muestra de amplitud del mensaje M_A y se le resta A para obtener la muestra del criptograma:

$$C_A = M_A + (R_A \bmod 2A)$$

$$C = C_A - A \qquad \text{(Criptograma a transmir)}$$

Ya en el receptor, para recuperar la muestra del mensaje de voz a partir de la muestra del criptograma, a la muestra de amplitud del ruido pseudoaleatorio R_A se le aplica la función módulo 2A, este resultado es restado de la muestra de amplitud del criptograma C_A, y finalmente a C_A se le resta A para recuperar el mensaje:

$$M_A = C_A - (R_A \bmod 2A)$$

$$C = M_A - A$$

Por ejemplo:

Sea $M=-3$, $R=4$ (amplitudes del mensaje y del ruido respectivamente)

$A=5$ (rango de amplitudes soportadas: $(-A,A)$)

entonces:

$$M_A = M + A \qquad M_A = -3+5 = 2$$

$$R_A = R + A \qquad R_A = 4+5 = 9$$

$$C_A = M_A + (R_A \bmod 2A)$$

$$C_A = 2+9 \bmod 10$$

$$C_A = 2+1 = 3$$

$$C = C_A - A$$

C=3-5=-2 (amplitud de la muestra del criptograma a transmitir)

$M_A=C_A-(R_A \bmod 2A)$

$M_A=3-9 \bmod 2A$

$M_A=3-1=2$

M=M_A-A=-5=-3 (amplitud de la muestra del mensaje recibido)

Una de las mayores ventajas que presenta este método es la posibilidad que la señal del criptograma se escuche como ruido blanco, además de la gran cantidad del número de amplitudes del ruido pseudoaleatorio que pueden generarse.
La principal desventaja de este método es la pérdida significativa en la relación señal-ruido en el receptor, debido a que parte de la energía se pierde en el receptor debido al ruido pseudoaleatorio que se transmitió.

IV.5. Criptografía bidimensional.

La criptografía bidimensional se refiere a los métodos que combinan dos técnicas de enciframiento, una en el dominio del tiempo y el otro en el dominio de la amplitud. Por ejemplo, combinar el método de inversión en frecuencia con el método de permutación de bloques en el tiempo, a combinar un reordenador de bandas de frecuencia con un permutador de bloques en el tiempo.
Estos sistemas cifradores presentan una mayor seguridad, pero a la vez su implementación tiene un mayor grado de dificultad.

V.1.- Introducción.

Los procesos de simulación fueron realizados en dos etapas, para que de esta forma se alcanzaran los objetivos que se pretendían. La primera etapa de las simulaciones fue desarrollada en una computadora PC-XT, junto con algunos otros recursos requeridos. Ya que la segunda etapa se trabajó con el conocimiento adquirido de la primera, obteniéndose como producto final los parámetros óptimos de los métodos cifradores que se pueden implementar en la arquitectura sugerida.

V.2. Simulaciones en PC.

Los objetivos que se cubrieron en esta fase fueron los siguientes:

a) Implementar mediante software los elementos de criptografía requeridos, ya vistos en el capitulo III.

b) Implementar mediante software los elementos de los algoritmos diseñados de los métodos cifradores.

c) Observar la inteligibilidad residual de los métodos cifradores, así como observar su comportamiento, tanto en el dominio del tiempo, como en el dominio de la amplitud.

V.2.1. Equipo para las simulaciones.

Para realizar las simulaciones se contó con el siguiente equipo, observar fig.(5.1):

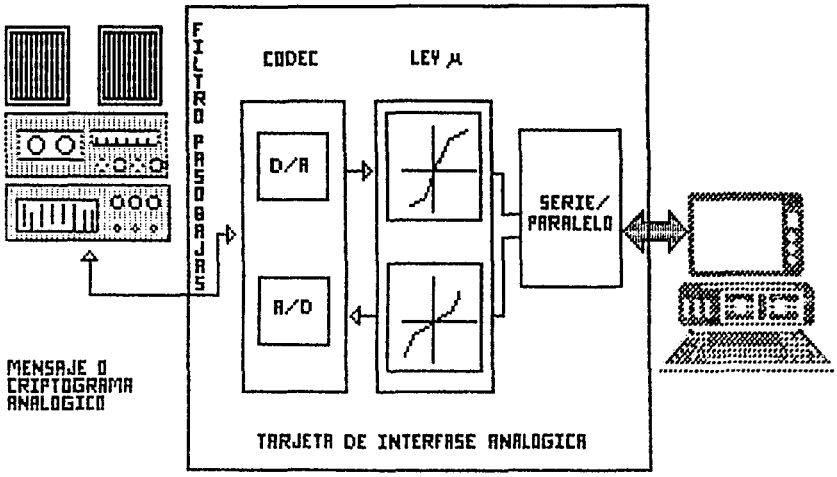


FIG. (5.1) EQUIPO UTILIZADO PARA LA SIMULACION DE METODOS CRIPTOGRAFICOS. (FASE I)

a.- Computadora personal: computadora PC-XT, con 640 kbytes de memoria RAM.

b.- Tarjeta de desarrollo para la interfase analógica de la computadora PC-XT. Tarjeta que cuenta con CODEC de 8 bits, filtro paso-bajas de 300 a 3300 Hz y compresión ley μ y manejo del controlador DMA 8237.

c.- Equipo modular estereofónico.

La frecuencia de muestreo fue de 8 KHz y la duración del mensaje de voz fue de 8seg. Los programas de las simulaciones fueron desarrollados en lenguaje pascal y el tamaño de los bloques fue de 128 muestras. Finalmente, para los métodos de permutación fueron utilizadas las funciones para permutaciones uniformes:

$$c=k_1 m \bmod N$$

$$m=k_2 c \bmod N$$

donde

$$k_1 k_2 \pmod N = 1 \qquad y$$

$$m= 1,2,3,\ldots N$$

y donde:

N: número total de bloques a encifrar (cada bloque tiene 128 muestras).
c: número de bloque asignado en el criptograma al bloque m del mensaje.
k_1: clave de encifrado.
k_2: clave de descifrado.

La clave k_1 se obtuvo mediante la función de turbo pascal que genera números pseudoaleatorios.

Por ejemplo:

Si $n=32$

con $k_1=7$ y $k_2=23$

entonces:

$$k_1k_2=161 \bmod 32=1$$

La fig.(5.2) muestra la matriz de permutaciones para este ejemplo.

DIRECCION DE BLOQUE

ENTRADA	SALIDA
1	7
2	14
3	21
4	28
5	3
6	10
7	17
--	--
M	C

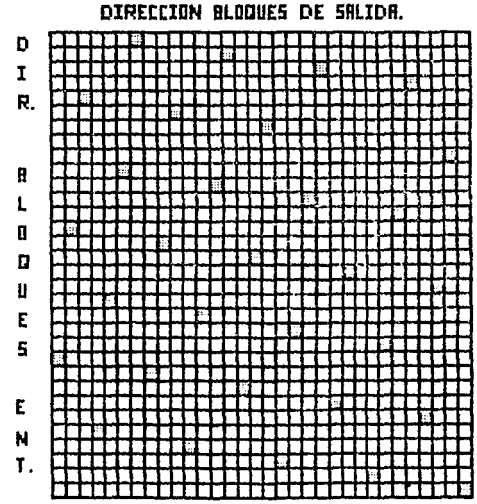


FIG. (5.2). PERMUTACIONES UNIFORMES

V.2.2.- Resultados de la primera etapa.

V.2.2.1.- Tiempo de retraso.

El tiempo de retraso es la adición del tiempo de encifrado mas el tiempo de descifrado. en la tabla 5.1 se muestran los tiempos de retraso para los cuatro métodos simulados, más importantes.

	M E T O D O	TIEMPO DE RETRASO (seg.)
a	Inversión en frecuencia.	84.73
b	Inversión en el tiempo.	70.36
c	Permutación de bloques en el tiempo.	115.40
d	Bidimensional c y a	129.55

TABLA 5.1. TIEMPOS DE RETRASO.

El tiempo de retraso es uno de los parámetros importantes en los sistemas cifradores, como se observa en la tabla 5.1, entre más complejo es un sistema cifrador, mayor es su tiempo de retraso. y generalmente, entre mayor complejidad tenga un sistema cifrador, mayor es su grado de seguridad. Los tiempos de retraso son aceptables hasta de 100 mseg. para este tipo de cifradores.

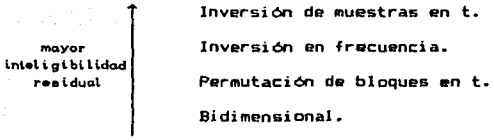
V.2.2.2.- Inteligibilidad Residual.

El término inteliqibilidad residual se refiere al qrado de reconicimiento de un mensaje de voz.

Para entender de una mejor manera lo que es intelegibilidad residual, suponqamos el siquiente ejemplo:

El único idioma que conocemos es el español, y queremos entender la plática entre un brasileño y un alemán, sabremos el tema de su plática si ésta es en portuqués. Ya que esta lengua tiene similitudes con el español, pero si la plática fuera en alemán, seguramente no entenderíamos nada de lo que están hablando. La similitud existente entre el portuqués y el español se debe a las raíces latinas de ambos, por lo que hay palabras iquales con el mismo siqnificado. Estas palabras, para nuestro ejemplo, son la intelegibilidad residual del portuqués respecto al español y viceversa.

La inteliqibilidad residual es una medida subjetiva, nosotros para evaluarla en nuestros métodos criptográficos. Solicitamos a distintas personas que nos hicieron favor de escuchar los distintos criptogramas, de lo que resultó:

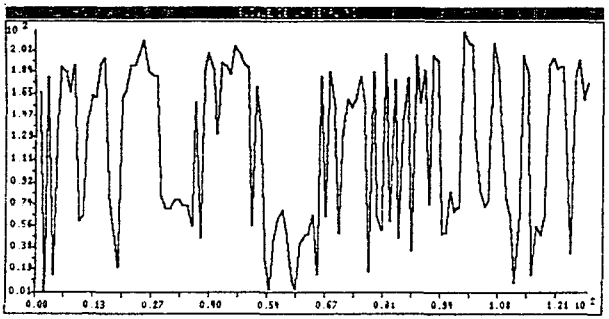


V.2.2.3.- Comprobaciones gráficas.

El aspecto gráfico fue muy importante, ya que nos permitió comprobar y observar algunas consideraciones ya mostradas en el capítulo anterior. En la fig. (5.3) se muestra un segmento del mensaje de voz y sus respectivos criptogramas de voz, tanto en el dominio del tiempo como en el dominio de la frecuencia.
En estas figuras se excluyen las primeras componentes espectrales. Si al espectro de la fig. (5.3b), lo dividiremos en dos partes con un corte en el eje de simetría, e intercambiamos ambas partes, el espectro resultante será el correspondiente al de la fig. (5.3a), que es el espectro del mensaje original. De esta forma comprobamos que para invertir la frecuencia, que es un recorrimiento del espectro, basta cambiar el signo de las amplitudes impares del mensaje.
El segmento en el tiempo de la fig (5.3c) es el segmento invertido del mensaje de voz, fig. (5.3a), como podrá observarse. Pero las respectivas densidades espectrales son iguales. Con lo cual comprobamos que la inversión de la FFT, excluyendo la primera

TIEMPO

FRECUENCIA



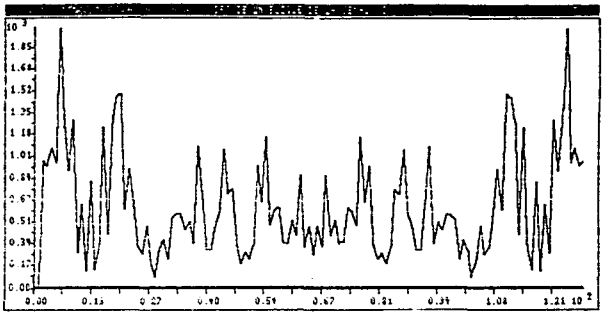
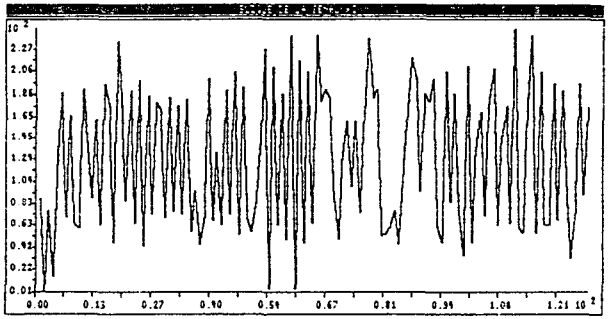


FIG.(5.3a). MENSAJE DE VOZ



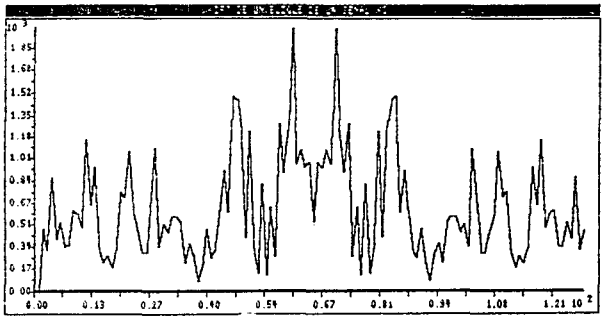
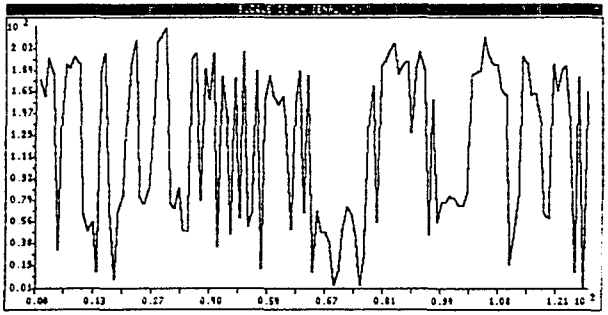


FIG.(5.3b). INVERSION EN FRECUENCIA



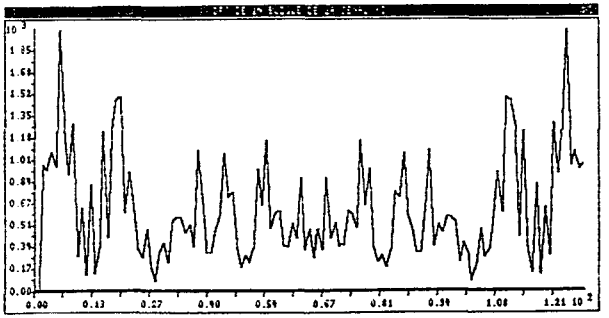


FIG.(5..3c). INVERSION EN EL TIEMPO

FIG. (5.3). SEGMENTO DE MENSAJE Y SEGMENTO DE CRIPTOGRAMA DE VOZ.

muestra causa una inversión en el tiempo. La inversión en el tiempo repercute en las componentes reales e imaginarias del espectro, como se observa en el siguiente ejemplo:

Los valores discretizados del mensaje son:

VALOR REAL	VALOR IMAGINARIO
1	0i
2	0i
3	0i
4	0i

La FFT del mensaje es:

AMPLITUD	VALOR REAL	VALOR IMAGINARIO
20000.00	20000	0i
5656.85	-4000	-4000i
4000.00	-4000	0i
5656.85	-4000	4000i

Los valores discretizados del criptograma son:

VALOR REAL	VALOR IMAGINARIO
4	0i
3	0i
2	0i
1	0i

La FFT del criptograma es:

AMPLITUD	VALOR REAL	VALOR IMAGINARIO
20000.00	20000	0i
5656.85	4000	4000i
4000.00	4000	0i
5656.85	4000	-4000i

Como se observa las magnitudes de la FFT no varían, pero si varían los signos de las componentes reales e imaginarias.

V.3.- Simulaciones en TMS32010.

El objetivo en esta segunda etapa de simulaciones era observar el comportamiento de los métodos cifradores, que en la etapa anterior estaban desarrollados en lenguaje de alto nivel, correspondía implementarlos en lenguaje ensamblador del TMS32010. En esta parte se compararon los tiempos de retraso y la inteligibilidad residual de los distintos métodos.

V.3.1- Equipo para las simulaciones.

El equipo utilizado para las simulaciones fue el siguiente, observar fig. (5.4):

a) Modulo de Evaluación del TMS32010.

b) Tarjeta de interfase analógica (AIB) del TMS32010.

c) Computadora PC-XT.

d) Grabadora.

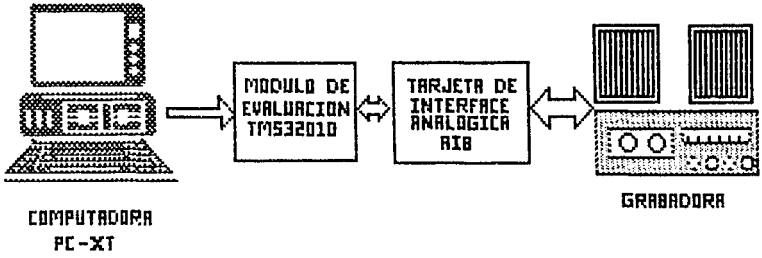


FIG. (5.4) EQUIPO PARA LA SIMULACION DE METODOS CIFRADORES (FASE II)

V.3.1.1.- Microprocesador TMS32010.

Los programas son desarrollados en la computadora PC-XT con cualquier editor. Mediante cierta comunicación se transfieren al módulo de evaluación, el cual, al recibirlos los va ensamblando. Desde el monitor conectado al módulo de evaluación se manipula éste y se ejecutan los programas. A continuación se describen las características del microprocesador TMS32010.

1. Arquitectura Harvard Modificada.
2. Pipeling.
3. Hardware dedicada a multiplicaciones.
4. Instrucciones especiales para el procesamiento digital de señales.

V.3.1.1.1.- Arquitectura Harvard.

El TMS32010 utiliza una arquitectura Harvard modificada para tener flexibilidad y comodidad. En una arquitectura Harvard el programa y los datos se almacenan en 2 memorias separadas, permitiendo una total coincidencia en parte de las instrucciones FETCH y EXECUTE. Las modificaciones en la familia TMS de la arquitectura Harvard permite mayor transferencia entre la memoria de datos y de programa, de este modo se incrementa la flexibilidad del dispositivo. Esta modificación de la arquitectura elimina la necesidad de dividir los coeficientes de la ROM y también maximiza el poder de procesamiento para mantener los dos bus de estructura separados, (programa y datos), para una ejecución totalmente rápida.

V.3.1.1.2.- Pipelining.

En conjunción con la arquitectura Harvard, el pipelining es usado extensivamente para reducir el tiempo de ciclo de una instrucción,

para que ésta sea absolutamente mínima. El pipelining puede ser en cualquiera de 2 de los 4 niveles, dependiendo de que procesador de la familia es usado. La arquitectura de la familia TMS320 usa un pipelining de 2 niveles para esta primera generación como el TMS32010, un pipelining de 3 niveles para la segunda generación y un pipelining de 4 niveles para los procesadores de la tercera generación. De forma tal que el dispositivo procesa de 2 a 4 instrucciones en paralelo y cada instrucción está en un diferente estado en la ejecución. La fig.(5.5), muestra un ejemplo de una operación pipeline de tres niveles.

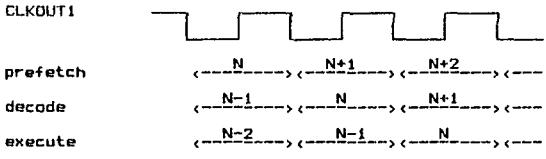


FIG. (5.5). OPERACION PIPELINE DE TRES NIVELES.

En la operación pipeline, las operaciones prefetch, decodificación y ejecución pueden ser manejadas independientemente, así permite la ejecución de operaciones en traslape. Durante un ciclo de instrucción, tres diferentes instrucciones son activadas, en cada uno de los diferentes estados de terminación.

Por ejemplo, como la N_{ava} instrucción está inicialmente en prefetch, la previa $(N-1)_{ava}$ instrucción está iniciando la

decodificación y la $(N-2)_{ava}$ está iniciando la ejecución. En general la pipeline es transparente al usuario.

V.3.1.1.3.- Hardware dedicado a multiplicaciones.

La multiplicación es una parte importante en el procesamiento digital de señales. La velocidad de la realización de las multiplicaciones es debida al alto desarrollo del procesador digital de señales. En los microprocesadores de propósito general, la instrucción de la multiplicación es construida por una serie de sumas, por lo tanto toman muchos ciclos de instrucción.
En la familia TMS320 la multiplicación está en una sola instrucción de un ciclo, como resultado de un Hardware dedicado a las multiplicaciones.

V.3.1.1.4.- Instrucciones especiales para el procesamiento digital de señales.

Otra característica del microprocesador TMS32010 es el uso de instrucciones especiales. Estas instrucciones están enfocadas para realizar las operaciones aritméticas requeridas en el procesamiento digital de señales, como las sumatorias de productos, de una manera más fácil. Existen además instrucciones que realizan un conjunto de instrucciones en un ciclo y de ésta forma se reducen los números de ciclos por instrucción.

V.3.1.1.5.- Ciclos rápidos de instrucción.

La capacidad de procesamiento en tiempo real es 4 veces mejorado por la velocidad del procesador en la ejecución de las instrucciones.

El tiempo del ciclo de instrucción para el microprocesador TMS32010 es de 160 a 200 ns.

La familia de procesadores TMS320 es altamente segura para muchas aplicaciones de procesamiento digital de señales en tiempo real.

Como podemos ver en la fig. (5.6), muchos ciclos de instrucción son adecuados para el proceso de la señal o para generar comandos para aplicaciones en tiempo real. Por lo tanto, para simples aplicaciones de control, los microprocesadores de propósito general o controladores pueden ser adecuados. Sin embargo para muchas aplicaciones de control con rigurosos procesos matemáticos, tales como robótica y control adaptivo, los microprocesadores digitales de señales tienen un mejor uso. El número de ciclos de instrucción adecuado es reducido a medida que incrementemos el período de muestreo, de 8 KHz. para las aplicaciones típicas de telecomunicaciones y de 40 a 48 KHz. para procesamiento de audio. Algunas de estas aplicaciones de tiempo real requieren solo de unos pocos cientos de instrucciones por muestra en el TMS32010.

Para aplicaciones con alto promedio de muestreo, como procesamiento de imágenes o video, el procesador digital de señales adecuado todavía no es capaz de manejar el procesamiento

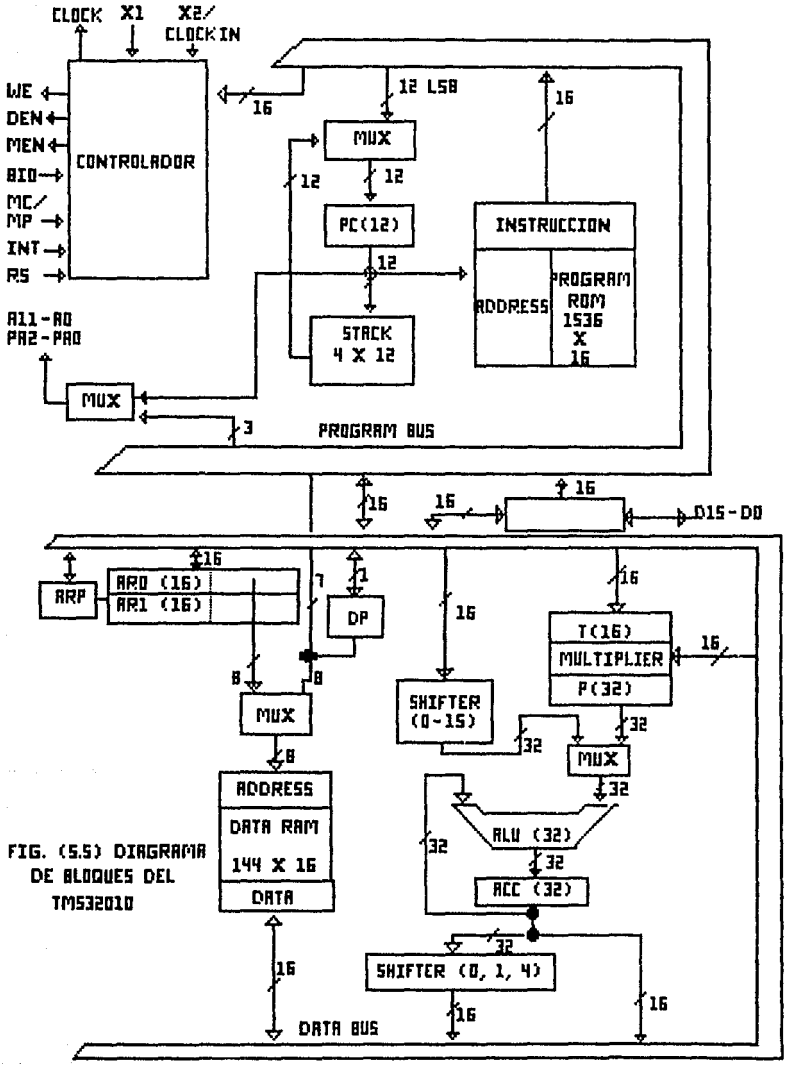


FIG. (5.5) DIAGRAMA DE BLOQUES DEL TMS32010

memoria de datos es de 144 x 16 bits y es de tipo RAM. Los 4 elementos aritméticos básicos del TMS32010 son: Alu, Acumulador, el Multiplicador y los Registros de Corrimiento. Todas las operaciones aritméticas son ejecutadas usando aritmética de complemento a 2.

ALU: Es una unidad aritmética lógica de propósito general que opera con una palabra de datos de 32 bits. La unidad puede sumar, restar y ejecutar operaciones logicas.

ACUMULADOR: El acumulador almacena la salida del ALU y también la entrada, éste opera con una palabra de longitud de 32 bits, el acumulador está dividido en partes: palabra alta (bits 31-16) y palabra baja (bits 15-0). Las instrucciones están provistas para almacenar estas dos partes del acumulador en la memoria de datos, (SACH para almacenar la parte alta en el acumulador y SACL para almacenar la parte baja del acumulador.

MULTIPLICADOR: El multiplicador paralelo de 16 x 16 bits consiste de 3 unidades: El registro T, el registro P y el arreglo de multiplicación. El registro T es un registro de 16 bits que almacena al multiplicando, mientras que el registro P es un registro de 32 bits que almacena el producto. Con el propósito para usar el multiplicador, el multiplicando puede

primero ser cargado de la RAM de datos al registro T, usando alguna de las siguientes instrucciones: LT, LTA o LTD. Entonces las instrucciones MPY (multiplicación) o MPYK (multiplicación inmediata) son ejecutadas. Estas operaciones de multiplicar y acumular, pueden ser acompañadas en 2 ciclos de instrucción con las instrucciones LTA/LTD .

REGS. DE CORRIMIENTO: Dos corrimientos son habilitados para la manipulación de datos: El primero ejecuta un corrimiento a la izquierda de 0 - 16 bits sobre todas las palabras de la memoria de datos que están por ser cargados, sustraídos o sumados al acumulador. El segundo, un corrimiento paralelo, activado por la instrucción SACH puede ejecutar un corrimiento de 0, 1 o 4 bits.

V.3.1.2.- Tarjeta de interface analógica del TMS320C10 (AIB).

El propósito de esta descripción, del uso de la tarjeta de interface analógica del TMS32010, es que sirva como una referencia que informe de las características de la misma.

V.3.1.2.1.- Descripción.

La tarjeta de interface analógica es una herramienta educacional, que provee interfases A/D y D/A para el módulo de evaluación del

TMS32010. La AIB provee convertidores A/D y D/A de 12 bits con puertos de expansión para otros convertidores adicionales. En la fig. (5.8), se muestra un diagrama a bloques de la tarjeta de interface analógica del TMS:

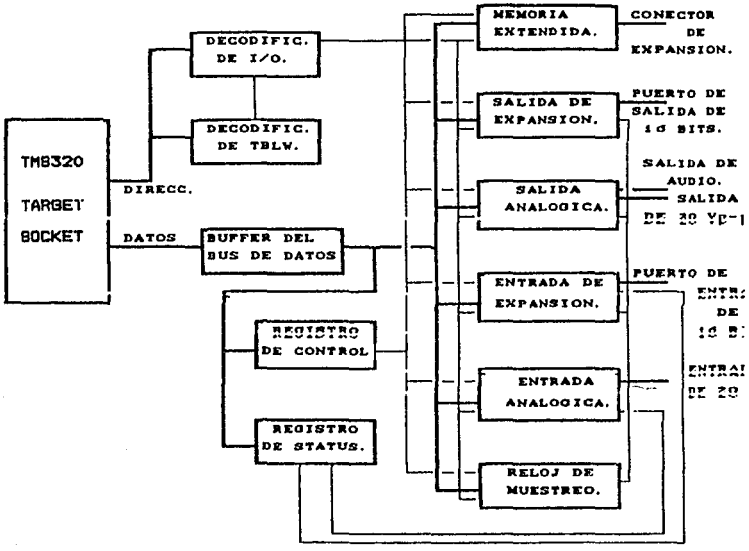


FIG. (5.8), DIAGRAMA A BLOQUES DEL AIB DEL TMS32010.

El reloj de la tasa de muestreo de la AIB es derivado del CLKOUT en el TMS32010, y puede ser programado para proveer entradas o

salidas analógicas o ambas. Esta tiene 2 filtros analógicos paso-bajas. Un filtro a la entrada del A/D limita la entrada a la banda para minimizar los efectos del ruido. El otro filtro está colocado a la salida del D/A. La respuesta de frecuencia de los filtros es controlada por la variación de componentes externos en los estados del filtro. La frecuencia de corte de los filtros es de 4.7 KHz., pero pueden ser programados para su uso.

V.3.1.2.2.- Características.

La AIB provee una combinación de características que cubren muchas aplicaciones, mientras mantienen flexibilidad para expansión:

- Convertidor A/D de 12 bits.
- Convertidor D/A de 12 bits.
- Un puerto de salida de 16 bits, para adicionales A/D.
- Un puerto de entrada de 16 bits, para adicionales D/A.
- Dos filtros paso bajas.
- Amplificador de audio.
- Tabla de escritura (TBLW decodificada).
- Memoria de datos de I/O extendida.
- Area para aplicaciones.

V.3.1.2.3.- Especificaciones Generales.

CONVERTIDOR ANALOGICO - DIGITAL.

- resolución : 12 bits.
- entrada analógica : -10 v a +10 v
- salida digital : 16 bits en complemento a 2.
- tiempo de conversión : 25 μs. (máximo).

MUESTREO Y RETENCION.

- tiempo de adquisición a 0.1% : 4 μs.
- promedio de salida : 0.3 V/seg.
- paso de retención : 16 bits en compl. a 2

CONVERTIDOR DIGITAL - ANALOGICO.

- resolución : 12 bits.
- salida analógica :-10 v a +10 v
- entrada digital : 16 bits en complemento a 2.
- tiempo de asentamiento : 25 μs.

RELOJ DE MUESTREO.

- rango : 76.29 Hz. a 5.00 MHz.

MEMORIA DE DATOS EXTENDIDA.

- capacidad de la tarjeta : 8192 X 16 bits.

V.3.2.- Resultados de la segunda etapa.

El listado de los programas de los principales métodos y de algunas de sus combinaciones se presentan en el apéndice. La forma en que están diseñados los programas de los métodos cifradores básicos permiten usar éstos como subrutinas, para desarrollar sistemas mas complejos y de esta forma reducir la inteligibilidad residual. Pero deben tenerse en cuenta los tiempos de retardo para no rebasar los cien milisegundos que es el tiempo razonble de retraso en este tipo de sistemas cifradores.

Nuestro almacenamiento está restringido para almacenar hasta 128 muestras aproximadamente, ya que la memoria de datos del TMS32010 es de 144 palabras de 16 bits. En todos los programas los parámetros importantes, como tamaño de marco, tamaño de segmento, etc. están en memoria de programa, para tener de esta forma una facilidad en la modificación de parámetros desde el monitor.

V.3.2.1.- Descripción de los programas de los principales métodos de cifrado.

En el método de inversión en frecuencia, la muestra A/D es alamacenada en una localidad de memoria de datos. Mediante una bandera se determina si es muestra impar, si esto ocurre, multiplicamos esta muestra por -1 y le restamos una unidad. Esta última resta se debe hacer ya que el TMS32010 trabaja con

aritmética de complemento a 2 y para cambiar la polaridad de un voltaje de salida de la AIB, basta con cambiar el valor de cada bit. Ya que por ejemplo, para sacar un voltaje de 10 volts, se debe mandar por el puerto dos un valor de 8000_H y para sacar un voltaje de -10 volts se debe mandar un valor de $7FFF_H$.

En el método de inversión de muestras en el tiempo, se utilizaron dos segmentos contiguos que conforman un bloque. En el primer segmento se van almacenando las muestras y en el segundo se van vaciando en orden inverso. De esta forma, mientras se va llenando el bloque por el inicio, se va vaciando por el final. Al terminar de vaciar el bloque, que ocurre al mismo tiempo de terminar de llenar el bloque, se intercambian los apuntadores de inicio de llenado e inicio de vaciado.

En el método de permutación de bloques en el tiempo, entra el aspecto de pseudoaleatoridad. Este aspecto se implementó a un nivel muy simple con un registro de corrimiento de siete estados. En el apéndice mostramos un generador de números pseudoaleatorios para que a modo de subrutina se implemente. La idea de implementar el registro de corrimiento tuvo la finalidad de comparar métodos de permutaciones en igualdad de circunstancias, ya que conocíamos el desempeño del registro de corrimiento. En este método utilizamos dos marcos y dos etapas de proceso:

Etapa 1. LLenado secuencial de marco 1 y vaciado permutando los segmentos del marco 2.

Etapa 2. LLenado secuencial de marco 2 y vaciado permutando los segmentos del marco 1.

Estas dos etapas se van alternando todo el tiempo, pero al ocurrir cambio de etapa se cambian dirección de inicio de llenado y magnitud de offset de vaciado.

El generador de números aleatorios propuesto en el apéndice se basa en el uso de la ecuación del método de congruencia lineal:

$$R_{n+1} = (a\ R_n + c) \bmod m$$

donde:

$$R >= 0$$

$$a >= 0$$

$$c >= 0$$

$$m > R,\ a,\ c$$

Estos parámetros se encuentran en memoria de programa para su fácil manipulación, de la elección de estos parámetros depende la periodicidad del generador de números pseudoaleatorios:

-El módulo m debe ser bastante grande ya que determina el rango de los números pseudoaleatorios.

-El multiplicador a debe ser bastante grande.

-El incremento c debe ser bastante pequeño.

-La semilla R debe ser un número pequeño.

De la elección de estos parámetros depende la periodicidad del generador de números pseudoaleatorios.

V.3.2.2.- Retraso de tiempo.

Los retrasos de tiempo observados en los métodos cifradores caen dentro de los límites aceptados, (aproximadamente cien milisegundos), esto lo podemos observar a continuación:

	ciclos por segundo	retraso de tiempo (seg)
a) Inversión en frecuencia	137,088	0.02741760

	ciclos por segundo	retraso de tiempo (seg)
b) Inversión de muestras en el tiempo.		
longitud del segmento		
32	81,648	0.016262440
48	81,312	0.01626240
c) Permutaciones de bloques en el tiempo.		
longitud del segmento	segmento por marco	
8	7	124,704	0.02494080
4	7	176,832	0.03536640
2	7	281,088	0.05621760
d) permutación de bloques en el tiempo, con inversión en tiempo en segmentos.		
longitud del segmento	segmento por marco	
4	7	168,768	0.03375360
2	7	264,960	0.05299200

		ciclos por segundo	retraso de tiempo (seg)
e) Permutación de bloques en el tiempo con inversión en frecuencia.			
longitud del segmento	segmento por marco		
4	7	297,792	0.05955840
2	7	402,048	0.08040960
f) Permutación de bloques en el tiempo, con nversión en tiempo en segmentos e inversión en frecuencia.			
longitud del segmento	segmento por marco		
4	7	289,728	0.05794560
2	7	385,920	0.07718400

De los métodos mostrados el que tuvo menor tiempo de retraso fue el de inversión de muestras en el tiempo (método b), con longitud de segmento igual a 48. El mayor, retraso de tiempo le correspondió al método bidimensional de permutación de bloues en

el tiempo con inversión en frecuencia (método c), para una longitud de segmento igual a 2 y el número de marcos igual a siete .

De los resultados obtenidos se puede observar que el método d tiene un menor retraso de tiempo que el método c, a pesar de que el método d contempla al método c. Esto debido a las características del algoritmo encifrador y a las características de programación del TMS.

Otra observación importante se refiere al añadir el método a al método c o d, se incrementa de forma constante el retraso total, el incremento es de 24 mseg. Por lo tanto, añadir el método a a un método combinado, es aumentar el retraso de tiempo propio del método a.

V.3.1.3.- Inteligibilidad Residual.

La inteligibilidad residual es una medida subjetiva que puede variar de persona a persona. La inteligibilidad residual aumenta, y es de poco valor su calificación, si el calificador de ésta conoce el mensaje original.

Para la inteligibilidad residual, al igual que en la fase anterior, diversas personas evaluaron el criptograma de voz, para esta evaluación hubo dos pruebas, la primera fue reconocer a la persona que hablaba y la segunda reconocer el contenido del mensaje.

En el reconocimiento del parlante, todos los métodos pasaron la prueba ya que nadie supo reconocer al parlante.
Sobre el contenido del mensaje, a excepción de un método, la inteligibilidad residual fue baja. Los métodos con menor inteligibilidad residual fueron los métodos combinados y el método con mayor inteligibilidad residual fue el de inversión en el tiempo.

METODO	COMPORTAMIENTO A LA INTELIGIBILIDAD
-PERMUTACION DE BLOQUES EN EL TIEMPO CON INVERSION DE TIEMPO EN SEGMENTOS E INVERSION EN FRECUENCIA.	MUY BUENO
-PERMUTACION DE BLOQUES EN EL TIEMPO CON INVERSION EN FRECUENCIA.	MUY BUENO
-INVERSION EN FRECUENCIA CON INVERSION EN EL TIEMPO.	MUY BUENO
-PERMUTACION DE BLOQUES EN EL TIEMPO CON INVERSION EN TIEMPO EN SEGMENTOS. LONGITUD DE SEGMENTO: 4	BUENO

METODO	COMPORTAMIENTO A LA INTELIGIBILIDAD
-INVERSION EN FRECUENCIA	BUENO
-PERMUTACION DE BLOQUES EN EL TIEMPO. LONGITUD DE SEGMENTO:4	BUENO
-PERMUTACION DE BLOQUES EN EL TIEMPO CON INVERSION EN TIEMPO EN SEGMENTOS. LONGITUD DEL SEGMENTO: 8.	REGULAR
-PERMUTACION DE BLOQUES EN EL TIEMPO. LONGITUD DEL SEGMENTO: 8	REGULAR
-PERMUTACION DE BLOQUES EN EL TIEMPO CON INVERSION EN TIEMPO. LONGITUD DEL SEGENTO 2.	ACEPTABLE
-PERMUTACION DE BLOQUES EN EL TIEMPO. LONGITUD DEL SEGMETNO: 2.	ACEPTABLE

METODO	COMPORTAMIENTO A LA INTELIGIBILIDAD
-INVERSION DE MUESTRAS EN EL TIEMPO.	MALO

El resultado observado en los métodos de inversión de muestras en el tiempo, es debido a la longitud del segmento, ya que este método requiere de mayores longitudes por segmento, las longitudes para este método son del orden de 800 muestras. Este método además debe considerarse como un código, al igual que el método de inversión en frecuencia. Aunque este último muestra buenos resultados en conjunción con algún método de permutaciones. De los resultados obtenidos pueden tenerse las siguientes conclusiones:

1. Pese a los buenos resultados mostrados por el método de inversión en frecuencia, no se sugiere su utilización como método cifrador.

2. Pese a los buenos resultados mostrados por el método de inversión en frecuencia con inversión segmentada en el tiempo, no se sugiere su utilización como método cifrador, ya que este método es de tipo código.

3. Es necesario hacer uso de la combinación de dos o más métodos.

4. Es importante hacer uso de algún método de permutaciones.

5. En el uso de permutaciones se sugiere el uso de una llave para modificar la secuencia del generador de números pseudoaleatorios. Esto puede realizarse modificando los parámetros del generador de números pseudoaleatorios presentado en el apéndice.

6. Todos los métodos presentados en este trabajo no presentan problemas con el tiempo de retraso, por lo que se sugiere aumentar a la mayor complejidad posible.

7. Se aconseja el método de permutación de bloques en el tiempo con inversión de tiempo en segmentos e inversión en frecuencia como un buen método de seguridad.

8. Se sugiere implementar diversos métodos a manera de subrutina e ir encifrando el mensaje con la elección de uno de estos métodos pseudoaleatoriamente.

9. En la elección del método encifrador deben tenerse en consideración los siguientes aspectos:

a) ¿De quién se debe proteger el sistema de comunicación?

b) ¿Qué capacidad tiene el criptoanalista de accesar el criptograma?

c) ¿Qué alcances tiene el criptoanalista?

d) ¿Qué consecuencias se tienen si el criptoanalista descifra el criptograma?

V.4.- Diseño del Sistema Cifrador.

El diagrama de bloques del sistema cifrador es mostrado en la fig.(5.9).

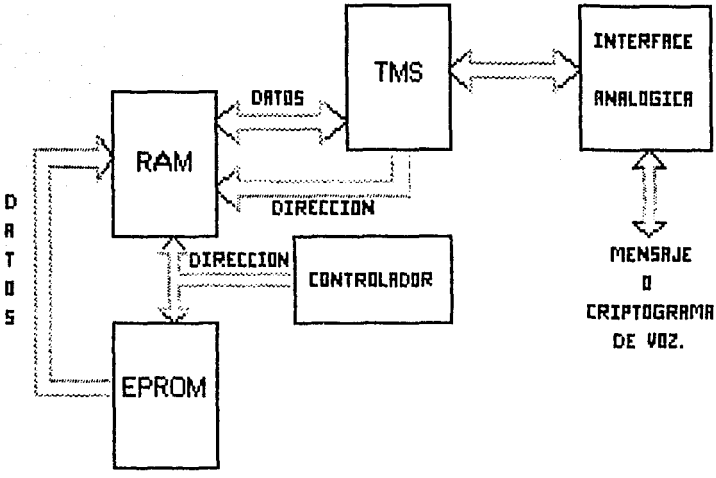


FIG.(5.9). DIAGRAMA DE BLOQUES DEL SISTEMA CIFRADOR.

V.4.1.-Descripción del Funcionamiento.

Al momento de encender el equipo se inicializa el sistema (reset), entonces el controlador empieza a funcionar. La finalidad del controlador, implementado con dos Flip-Flop's, es leer las instrucciones del algoritmo encifrador, almacenadas en la memoria EPROM 2732, y escribir éstas en las memorias RAM 2148 de rápido acceso. Para realizar esta operación el controlador se vale de un arreglo de contadores 74171, para obtener así ambas direcciones, de lectura en la EPROM 2732 y de escritura en la RAM 2148, es decir, se lee y se escribe haciendo uso de la misma dirección. Se cuenta con dos selectores de tres estados 74241 para seleccionar quien tendrá acceso a las direcciones de la memoria RAM 2148, el acceso lo puede tener el microprocesador TMS32010 o los contadores 74161. También se cuenta con un decodificador 74139 para seleccionar un renglón del arreglo de memorias RAM 2148. En la fig.(5.10) se muestra el diagrama lógico del sistema cifrador.

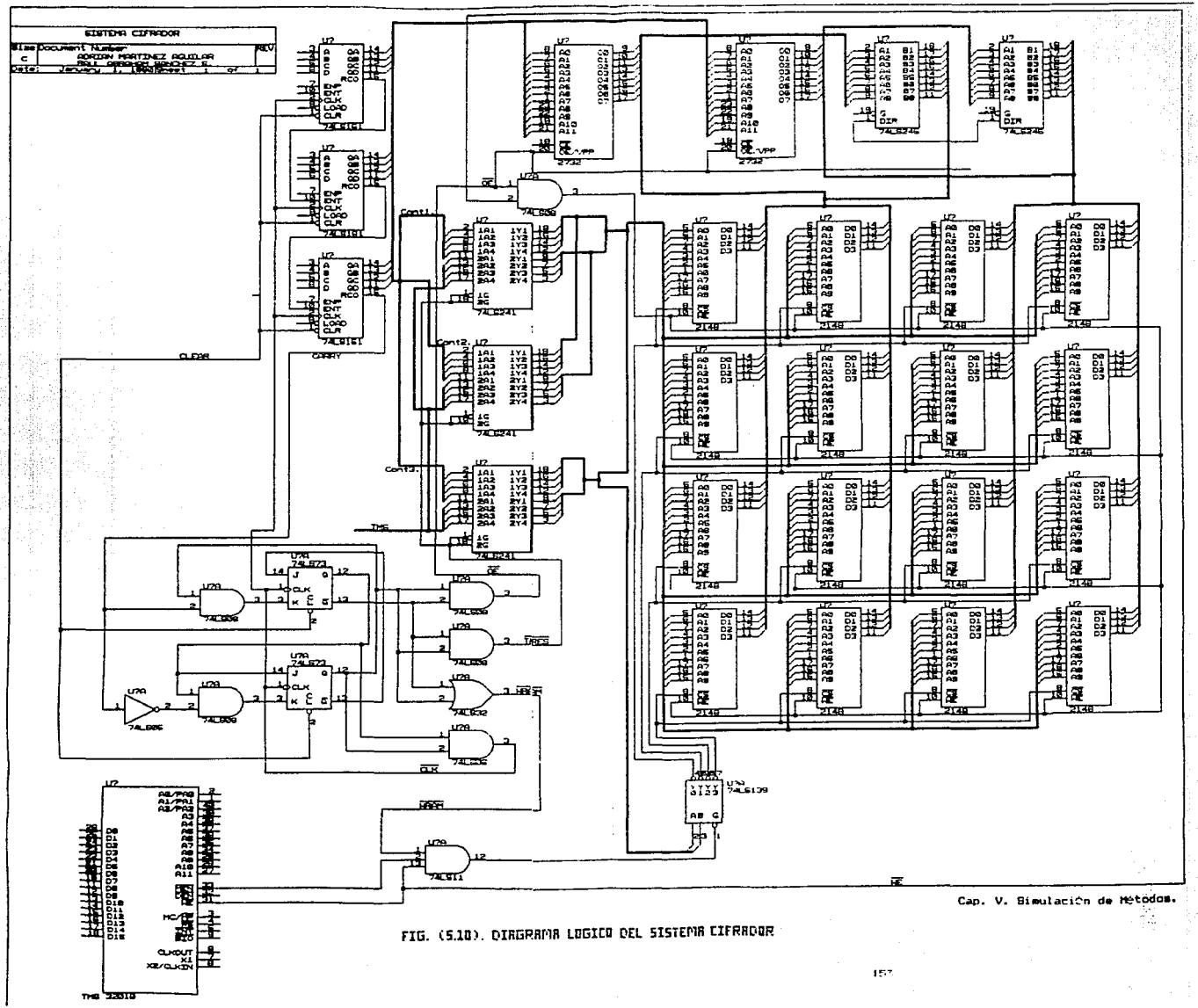


FIG. (5.10). DIAGRAMA LOGICO DEL SISTEMA CIFRADOR

Conclusiones.

> Shannon's ideas, ABOUT INFORMATION THEORY, were influenced by work he did on criptography during World Ward II.
>
> Richard E. Blahut.

La década de los 20's es una de las décadas más creativas de este siglo tanto en la ciencia y el arte, dentro de la criptografía es denominada la edad de oro de las máquinas cifradoras.

Podemos justificar este calificativo si observamos la fig. (6.1), en esta figura se muestran las patentes registradas relacionadas con la criptografía en los Estados Unidos. Como se observa el primer crecimiento acelerado sobre patentes criptográficos se muestra en la década de los 20's , esto como resultado de la experiencia adquirida sobre seguridad de la información en la primera guerra mundial. Después viene un declive en el registro de patentes, para que crezca éste en las postrimerías de la segunda guerra mundial. A partir de 1970 se inicia el gran crecimiento de patentes, donde la mayoría de ellas no son aplicadas para fines bélicos, ya que para la transmisión de información clasificada bélica no se dan a la luz pública los algoritmos cifradores que se utilizan. Cabe mencionar que desde la primera patente criptográfica en 1871 hasta 1980 se tienen en los E.U. 1769 patentes registradas relacionadas con la criptografía.

El desarrollo de la criptografía, que en un principio tenía unicamente fines bélicos, actualmente, como ya hemos mencionado,

es utilizada además para otras aplicaciones.

Al ir cobrando cada vez mayor importancia los sistemas de

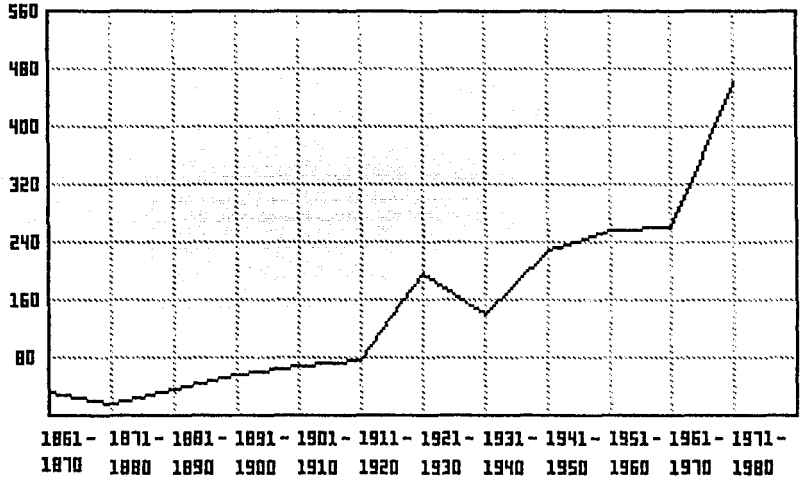


FIG. 6.1 NUMERO DE PATENTES CRIPTOGRAFICAS REGISTRADAS POR DECADA EN E.U.

comunicaciones, los sistemas cifradores también la cobran. Ya que, como se observa en la fig. (6.2), un sistema cifrador es un elemento de un sistema de comunicación. Actualmente existen asociaciones como la American Cryptogram Association, la New York

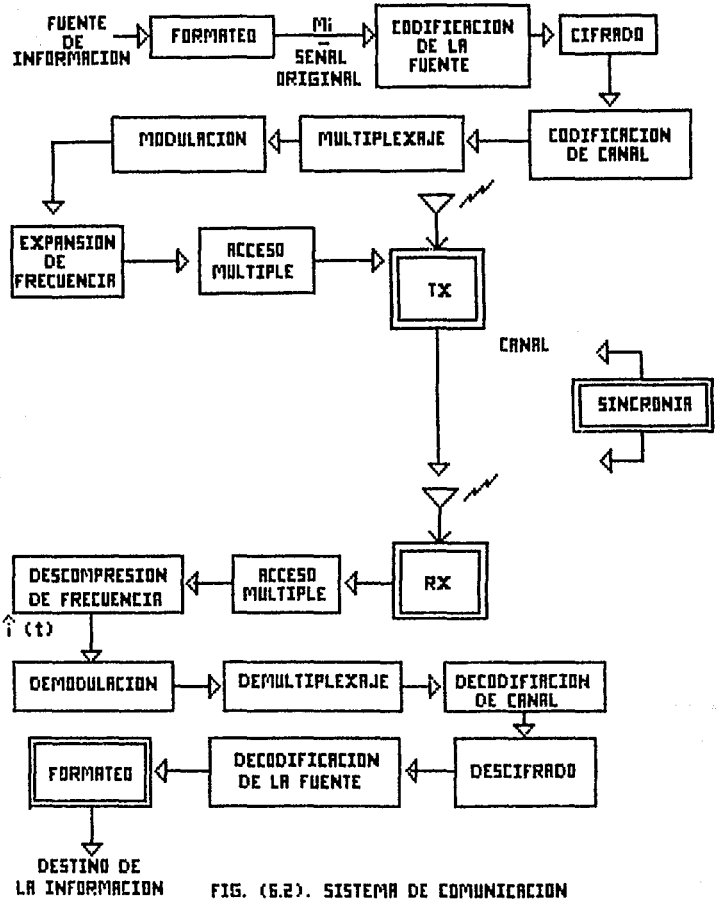


FIG. (6.2). SISTEMA DE COMUNICACION

Cipher Society o revistas como Cryptologia que denotan la importancia que va adquiriendo día a día la criptografía.

El sistema cifrador de privacidad que proponemos, es para voz analógica por vía telefónica, ya que el sistema telefónico es el mayor sistema de comunicación internacional que existe y dentro del cual está nuestro sistema cifrador. Este sistema cifrador es para mensajes analógicos, ya que no se vislumbra a corto plazo cambiar la red telefónica analógica por digital en el país. Actualmente, existen aproximadamente 60 proveedores en el mundo de equipo cifrador, con las condiciones existentes en nuestro país, consideramos que podemos crear algoritmos cifradores o mejorar los existentes, apoyándonos en los recursos computacionales existentes, para desarrollar sistemas cifradores aplicables a las necesidades que se requerirán en el país, obviamente sin pretender llegar al estado del arte en comunicaciones.

APENDICE

```
0001                ***************************************************
0002                *                                                 *
0003                *  PROGRAMA  PARA ENCIFRADO  DE UNA SE%AL ANALOGICA   *
0004                *  QUE UTILIZA EL METODO DE INVERSION EN FRECUENCIA.  *
0005                *                                                 *
0006                ***************************************************
0007                *
0008                *
0009                *VAR (LOCALIDADES DE LA MEMORIA DE DATOS)
0010                *
0011       0000     DATA0   EQU  0
0012       0001     DATA1   EQU  1
0013       0002     SIGNO   EQU  2
0014       0003     ONE     EQU  3
0015                *
0016 0000           ADRG 0
0017 0000 F900      B    BEGIN
     0001 0006
0018                *
0019                *CONST (LOCALIDADES DE LA MEMORIA DE PROGRAMA)
0020                *
0021 0002 0271      RATE    DATA 625                *TASA DE MUESTREO: 8 Khz.
0022 0003 00FA      MODE    DATA >FA                *MODO DE LA AID
0023 0004 0001      UNO     DATA 1
0024 0005 FFFF      SGN     DATA -1
0025                *
0026 0006 6E00      BEGIN   LDPK 0                  *SELECCIONA PAGINA CERO DE
0027 0007 7E02              LACK RATE               *
0028 0008 6700              TBLR DATA0               *PROGRAMA TASA DE NUESTRE
0029 0009 4900              OUT  DATA0,1            *
0030                *
0031 000A 7E03              LACK MODE               *
0032 000B 6700              TBLR DATA0               *PROGRAMA MODO DE OPERACI
0033 000C 4800              OUT  DATA0,0            *
0034                *
0035 000D 7E04              LACK UNO
0036 000E 6701              TBLR DATA1              *(DATA1) -- 1
0037 000F 6703              TBLR ONE                *(ONE)   -- 1
0038                *
0039 0010 7E05              LACK SGN
0040 0011 6702              TBLR SIGNO              *(SIGNO) -- -1
0041                *
0042 0012 F600      LOOP1   BIOZ LOOP2              *IF BIOZ = 0 THEN WAIT
     0013 0016
0043 0014 F900              B    LOOP1              *              ELSE OBTEN MU
     0015 0012
0044                *
0045 0016 4200      LOOP2   IN   DATA0,2            *(DATA0) -- MUESTRA
0046 0017 2001              LAC  DATA1              *IF NUMERO DE MUESTRA IMPA
0047 0018 FF00              BZ   CAMBIO
     0019 001E
0048                *
0049 001A 7F89              ZAC
0050 001B 5001              SACL DATA1              *APUNTADOR DE MUESTRA = IM
0051 001C F900              B    OUTPUT
     001D 0025
0052                *
```

```
0053 001E 6A02  CAMBIO  LT   SIGNO              *
0054 001F 6D00          MPY  DATA0               *
0055 0020 7F8E          PAC                        * (DATA0) -- -(DATA0) -
0056 0021 1003          SUB  ONE                   *  SE INVIERTE LA MUESTRA
0057 0022 5000          SACL DATA0               *
0058          *
0059 0023 2003          LAC  ONE
0060 0024 5001          SACL DATA1               *APUNTADOR DE MUESTRA = PA
0061          *
0062 0025 4A00  OUTPUT  OUT  DATA0,2             *SACA MUESTRA ENCIFRADA
0063 0026 F900          B    LOOP1               *REGRESA POR OTRA MUESTRA
     0027 0012
0064                    END
NO ERRORS, NO WARNINGS
```

```
ND*IDT      32010 FAMILY MACRO ASSEMBLER   PC2.1 84.107     02:04:55  01-01-80
                                                              PAGE 0001

0001            ***************************************************
0002            *                                                 *
0003            *   PROGRAMA DE ENCIFRADO DE SEÑALES ANALOGICAS   *
0004            *   QUE UTILIZA EL METODO DE INVERSION DE MUESTRAS *
0005            *   EN EL TIEMPO.                                 *
0006            *                                                 *
0007            ***************************************************
0008            *
0009            *VAR (LOCALIIDADES DE MEMORIA DE DATOS)
0010            *
0011      0064  INI1    EQU 100
0012      0065  INI2    EQU 101
0013      0066  DATA    EQU 102
0014      0067  UNO     EQU 103
0015            *
0016 0000               AORG 0
0017 0000 F900          B    BEGIN
     0001 0008
0018            *
0019            *CONST (LOCALIDADES DE MEMORIA DE PROGRAMA)
0020            *
0021 0002 0001  START1  DATA 1                      *DIRECCION DE INICIO DE B
0022 0003 0018  START2  DATA 24                     *DIRECCION DE INICIO DE B
0023 0004 00FA  MODO    DATA >FA                    *MODO DE OPERACION DE LA
0024 0005 0271  RATE    DATA 625                    *TASA DE MUESTREO 8 Hz.
0025 0006 0001  ONE     DATA 1
0026 0007 0000  FOUR    DATA 0
0027            *
0028 0008 6E00  BEGIN   LDPK 0                      *SELECCIONA PAGINA 0 DE M
0029 0009 7E05          LACK RATE                   *
0030 000A 6766          TBLR DATA                    *CARGA TASA DE MUESTREO
0031 000B 4966          OUT  DATA,1                 *
0032            *
0033 000C 7E04          LACK MODO                   *
0034 000D 6766          TBLR DATA                    *CARGA MODO DE OPERACION
0035 000E 4866          OUT  DATA,0                 *
0036            *
0037 000F 7E02          LACK START1
0038 0010 6764          TBLR INI1                   *(INI1) -- INICIO1
0039 0011 7E03          LACK START2
0040 0012 6765          TBLR INI2                   *(INI2) -- INICIO2
0041 0013 7E06          LACK ONE
0042 0014 6767          TBLR UNO                    *(UNO)  -- 1
0043            *
0044 0015 3864  LOOP00  LAR  0,INI1                 *AR0 -- INI1
0045 0016 3965          LAR  1,INI2                 *AR1 -- INI2
0046            *
0047 0017 F800  LOOP0   BIOZ LOOP1                  *IF BIOZ = 0 THEN WAIT
     0018 001B
0048 0019 F900          B    LOOP0                                ELSE OBTEN M
     001A 0017
0049            *
0050 001B 42A1  LOOP1   IN   *+,2,1                 *(AR0) -- MUESTRA, AR0 =
0051 001C 4A90          OUT  *-,2,0                 *SACA MUESTRA (AR1), AR1
0052 001D 3166          SAR  1,DATA
0053 001E 2066          LAC  DATA                   *
0054 001F FE00          BNZ  LOOP0                   *IF DOS BLOQUES VACIADOS
```

```
     0020 0017
0055 0021 F900          B     LOOP00              *     ELSE VACIEMOS BLOQUE
     0022 0015
0056                    END
NO ERRORS, NO WARNINGS
```

```
0001                ************************************************
0002                *                                              *
0003                *  PROGRAMA DE ENCIFRADO DE SE%ALES ANALOGICAS QUE  *
0004                *  UTILIZA EL METODO DE PERMUTACION DE BLOQUES  EN  *
0005                *  EL TIEMPO.                                  *
0006                *                                              *
0007                ************************************************
0008                *
0009                *VAR (LOCALIDAES DE MEMORIA DE DATOS)
0010                *
0011          0072  SHIFT     EQU  114
0012          0073  Y0        EQU  115
0013          0074  Y2        EQU  116
0014          0075  ADDER     EQU  117
0015          0076  UNO       EQU  118
0016          0077  OFFSET    EQU  119
0017          0078  TOFFSE    EQU  120
0018          0079  TEMP      EQU  121
0019          007A  INI0      EQU  122
0020          007B  INI1      EQU  123
0021          007C  LSEGM     EQU  124
0022          007D  NSEGM     EQU  125
0023          007E  SEGM      EQU  126
0024                *
0025 0000                     AORG 0
0026 0000 F900                B    BEGIN
     0001 000D
0027                *
0028                *CONST (LOCALIDAES DE MEMORIA DE PROGRAMA)
0029                *
0030 0002 0001  ONE       DATA 1
0031 0003 0007  NREGIS    DATA 7                    *NO. DE LOCALIDADES DEL R
0032 0004 0039  INIC0     DATA 57                   *INICIO DEL BLOQUE2
0033 0005 0001  INIC1     DATA 1                    *INICIO DEL BLOQUE1
0034 0006 0038  NOFFSE    DATA 56                   *OFFSET PARA EL BLOQUE2 E
0035 0007 0008  SEGML     DATA 8                    *LONGITUD DEL SEGMENTO 8
0036 0008 0007  SEGMN     DATA 7                    *NO. DE SEGMENTOS POR MAR
0037 0009 0038  TOFF      DATA 56
0038 000A 0000  OFFT      DATA 0
0039 000B 00FA  MODO      DATA >FA                  *MODO DE LA AIB
0040 000C 0271  RATE      DATA 625                  *TASA DE MUESTREO 8 Khz
0041                *
0042 000D 2E00  BEGIN     LDPK 0                    *SELEECIONA PAGINA CERO D
0043                *
0044 000E 7E0C            LACK RATE                 *
0045 000F 6779            TBLR TEMP                  *CARGA TASA DE MUESTREO
0046 0010 4479            OUT  TEMP,1               *
0047                *
0048 0011 7E0B            LACK MODO                 *
0049 0012 6779            TBLR TEMP                  *CARGA MODO DE OPERACION
0050 0013 4079            OUT  TEMP,0               *
0051                *
0052 0014 7E09            LACK TOFF
0053 0015 6778            TBLR TOFFSE               *(TOFFSE) -- 56
0054                *
0055 0016 7E0A            LACK OFFT
0056 0017 6777            TBLR OFFSET               *(OFFSET) -- 0
```

```
0057                    *
0058 0018 7E02          LACK ONE
0059 0019 6776          TBLR UNO                *(UNO)    -- 1
0060                    *
0061 001A 7E04          LACK INIC0
0062 001B 677A          TBLR INI0               *(INI0)   -- 57
0063                    *
0064 001C 7E05          LACK INIC1
0065 001D 677B          TBLR INI1               *(INI1)   -- 1
0066                    *
0067 001E 7E03          LACK NREGIS             *(SHIFT)  -- 7
0068 001F 6772          TBLR SHIFT              *ESTADO INICIAL DEL REGIS
0069                    *
0070 0020 7E07          LACK SEGML
0071 0021 677E          TBLR SEGM               *(SEGM)   -- 8
0072                    *
0073 0022 2078  CHANGE  LAC  TOFFSE             *
0074 0023 5079          SACL TEMP                *
0075 0024 2077          LAC  OFFSET               *SE INTERCAMBIAN OFFSET
0076 0025 5078          SACL TOFFSE               *TOFFSE ==== OFFSET
0077 0026 2079          LAC  TEMP                *
0078 0027 5077          SACL OFFSET             *
0079                    *
0080 0028 207A          LAC  INI0               *
0081 0029 5079          SACL TEMP                *
0082 002A 207B          LAC  INI1                 *SE INTERCAMBIAN INCIOS
0083 002B 507A          SACL INI0                 *INI0 ==== INI1
0084 002C 2079          LAC  TEMP                *
0085 002D 507B          SACL INI1               *
0086 002E 387A          LAR  0,INI0             *AR0 -- INI0
0087                    *
0088 002F 7E08          LACK SEGMN
0089 0030 677D          TBLR NSEGM              *(NSEGM0) -- SEGMN
0090                    *
0091 0031 7E07  LOOP00  LACK SEGML
0092 0032 677C          TBLR LSEGM              *(LSEGM)  -- SEGML
0093                    *
0094 0033 F800          CALL CORRIM             *RECORRE REGISTRO DE CORR
     0034 0050
0095                    *
0096 0035 2072          LAC  SHIFT              *
0097 0036 1076          SUB  UNO                 *
0098 0037 5079          SACL TEMP                 *
0099 0038 6A79          LT   TEMP                 *OBTIENE DIRECCION A/D
0100 0039 6D7E          MPY  SEGM                 *
0101 003A 7F8E          PAC                      *
0102 003B 0076          ADD  UNO                *
0103 003C 0077          ADD  OFFSET             *SUMA EL OFFSET A DIRECCI
0104 003D 5079          SACL TEMP
0105 003E 3979          LAR  1,TEMP             *AR1 -- DIRECCION A/D
0106 003F 207C          LAC  LSEGM              *AC  -- LSEGM
0107                    *
0108 0040 F600  LOOP0   BIOZ LOOP1              *IF BIOZ = 1 THEN WAIT
     0041 0044
0109 0042 F900          B    LOOP0                        ELSE OBTEN M
     0043 0040
0110                    *
```

```
NO&IDT      32010 FAMILY MACRO ASSEMBLER   PC2.1 84.107     08:22:50  01-01-80
                                                                 PAGE 0003

 0111 0042 1076            SUB  UNO       *IF NO TERMINA DE SACAR SEGMENTO SIG
 0112 0043 FC00            BGZ  LOOP0     *  ELSE
      0044 003C
 0113              *
 0114 0045 207D            LAC  NSEGM     *
 0115 0046 1076            SUB  UNO        * IF NO TERMINA CON MARCO THEN CONTI
 0116 0047 507D            SACL NSEGM      *   ELSE VA POR OTRO MARCO
 0117 0048 FC00            BGZ  LOOP00     *
      0049 0031
 0118              *                       *
 0119 004A F900            B    CHANGE    *
      004B 0022
 0120              *
 0121 004C 7F89  CORRIM    ZAC              *RUTINA DEL REGISTRO DE CORRIMIENTO
 0122 004D 2076            LAC  UNO         *CON FUNCION  f = S0 + S2
 0123 004E 7972            AND  SHIFT       *
 0124 004F FF00            BZ   CERO0       *
      0050 0055
 0125              *                         *    IF S2 = 0 THEN Y0 = 0
 0126 0051 2076            LAC  UNO          *              ELSE Y1 = 1
 0127 0052 5073            SACL Y0            *
 0128 0053 F900            B    CONTO         *
      0054 0057
 0129              *                         *
 0130 0055 7F89  CERO0     ZAC               *
 0131 0056 5073            SACL Y0          *
 0132              *
 0133 0057 2276  CONTO     LAC  UNO,2       *
 0134 0058 7972            AND  SHIFT        *
 0135 0059 FF00            BZ   CERO1         *
      005A 005F
 0136 005B 2076            LAC  UNO           *
 0137 005C 5074            SACL Y2             * IF S0 THEN Y2 = 0
 0138 005D F900            B    CONT1          *       ELSE Y1 = 1
      005E 0061
 0139              *                          *
 0140 005F 7F89  CERO1     ZAC               *
 0141 0060 5074            SACL Y2          *
 0142              *
 0143 0061 7F89  CONT1     ZAC              *
 0144 0062 2073            LAC  Y0           *
 0145 0063 7874            XOR  Y2           *ADDER = Y0 XOR Y1
 0146 0064 5075            SACL ADDER       *
 0147 0065 7F89            ZAC
 0148 0066 2F72            LAC  SHIFT,15    *UN CORRIMIENTO DE SHIFT
 0149 0067 5872            SACH SHIFT       *A LA IZQUIERDA
 0150 0068 2275            LAC  ADDER,2     *
 0151 0069 7A72            OR   SHIFT        *SHIFT = SHIFT OR ADDER
 0152 006A 5072            SACL SHIFT       *
 0153 006B 7F80            RET
 0154              *
 0155              END
NO ERRORS, NO WARNINGS
```

```
NO$IDT      32010 FAMILY MACRO ASSEMBLER   PC2.1 84.107     08:22:50  01-01-80
                                                              PAGE 0001

                        ***************************************************
                        *                                                 *
                        *   PROGRAMA QUE UTILIZA DOS METODOS DE ENCIFRADO  *
                        *   PARA SE%ALES ANALOGICAS:                      *
                        *   1. PERMUTACION DE BLOQUES EN EL TIEMPO        *
                        *   2. INVERSIONDE MUESTRAS EN EL TIEMPO          *
                        *                                                 *
                        ***************************************************
                        *
                        *VAR (LOCALIDADES DE MEMORIA DE DATOS)
                        *
      0072              SHIFT   EQU  114
      0073              Y0      EQU  115
      0074              Y2      EQU  116
      0075              ADDER   EQU  117
      0076              UNO     EQU  118
      0077              OFFSET  EQU  119
      0078              TOFFSE  EQU  120
      0079              TEMP    EQU  121
      007A              INI0    EQU  122
      007B              INI1    EQU  123
      007C              LSEGM   EQU  124
      007D              NSEGM   EQU  125
      007E              SEGM    EQU  126
                        *
0000                            AORG 0
0000 F900               B       BEGIN
0001 000D
                        *
                        *CONST (LOCALIDADES DE MEMORIA DE PROGRAMA)
                        *
0001 0002 0001          ONE     DATA 1
0002 0003 0007          NREGIS  DATA 7          *NO. DE LOCALIDADES DEL REGISTRO DE
0003 0004 0039          INIC0   DATA 57         *INICIO DEL BLOQUE0
0004 0005 0001          INIC1   DATA 1          *INICIO DEL BLOQUE1
0005 0006 0038          NOFFSE  DATA 56         *MAGNITUD DEL OFFSET PARA EL BLOQUE1
0006 0007 0008          SEGML   DATA 8          *LONGITUD DEL SEGMENTO = 8 MUESTRAS
0007 0008 0007          SEGMN   DATA 7          *TAMA%O DEL MARCO = 7 SEGMENTOS
0008 0009 0038          TOFF    DATA 56
0009 000A 0000          OFFT    DATA 0
000A 000B 00FA          MODO    DATA >FA        *MODO DE OPERACION DE LA AIB
000B 000C 0271          RATE    DATA 625        *TASA DE MUESTREO 8 KHz
                        *
000C 000D 6E00          BEGIN   LDPK 0          *SELECCIONA 0 DE MEMORIA DE DATOS
                        *
000D 000E 7E0C                  LACK RATE       *
000E 000F 6779                  TBLR TEMP        *PROGRAMA TASA DE MUESTREO
0010 4979                       OUT  TEMP,1     *
                        *
0011 7E08                       LACK MODO       *
0012 6779                       TBLR TEMP        *PROGRAMA MODO DE OPERACION DE LA A
0013 4879                       OUT  TEMP,0     *
                        *
0014 7E09                       LACK TOFF
0015 6778                       TBLR TOFFSE     *(TOFFSE) -- 56
                        *
0016 7E0A                       LACK OFFT
```

```
0057 0017 6777          TBLR OFFSET    *(OFFSET) -- 0
0058                    *
0059 0018 7E02          LACK ONE
0060 0019 6776          TBLR UNO       *(UNO)    -- 1
0061                    *
0062 001A 7E04          LACK INIC0
0063 001B 677A          TBLR INI0      *(INI0)   -- 57
0064                    *
0065 001C 7E05          LACK INIC1
0066 001D 677B          TBLR INI1      *(INI1)   -- 1
0067                    *
0068 001E 7E03          LACK NREGIS
0069 001F 6772          TBLR SHIFT     *(SHIFT)  -- 7
0070                    *
0071 0020 7E07          LACK SEGML
0072 0021 677E          TBLR SEGM      *(SEGM)   -- 8
0073                    *
0074 0022 2078  CHANGE  LAC  TOFFSE    *
0075 0023 5079          SACL TEMP       *
0076 0024 2077          LAC  OFFSET     *SE INTERCAMBIAN OFFSET'S
0077 0025 5078          SACL TOFFSE     *TOFFSE ==== OFFSET
0078 0026 2079          LAC  TEMP       *
0079 0027 5077          SACL OFFSET    *
0080                    *
0081 0028 207A          LAC  INI0      *
0082 0029 5079          SACL TEMP       *
0083 002A 207B          LAC  INI1       *SE INTERCAMBIAN INICIOS:
0084 002B 507A          SACL INI0       *INI0 ==== INI1
0085 002C 2079          LAC  TEMP       *
0086 002D 507B          SACL INI1      *
0087 002E 387A          LAR  0,INI0    *AR0 -- INI0
0088                    *
0089 002F 7E0B          LACK SEGMN
0090 0030 677D          TBLR NSEGM     *(NSEGM) -- SEGMN
0091                    *
0092 0031 7E07  LOOP00  LACK SEGML
0093 0032 677C          TBLR LSEGM     *(LSEGM) -- SEGML
0094                    *
0095 0033 F800          CALL CORRIM    *RECORRE REGISTRO DE CORRIMIENTO
     0034 004C
0096                    *
0097 0035 6A72          LT   SHIFT     *
0098 0036 6D7E          MPY  SEGM       *
0099 0037 7F8E          PAC             *OBTIENE DIR. DE MEMORIA DE DATOS
0100 0038 0077          ADD  OFFSET     *
0101 0039 5079          SACL TEMP      *
0102 003A 3979          LAR  1,TEMP    *AR1 -- DIRECCION D/A
0103 003B 207C          LAC  LSEGM     *AC  -- LSEGM
0104                    *
0105 003C F600  LOOP0   BIOZ LOOP1     *IF BIOZ = 1 THEN WAIT
     003D 0040
0106 003E F900          B    LOOP0                ELSE OBTEN MUESTRA A/D
     003F 003C
0107                    *
0108 0040 42A1  LOOP1   IN   *+,2,1    *(AR0) -- MUESTRA A/D, AR0 = AR0 + 1
0109 0041 4A90          OUT  *-,2,0    *SACA MUESTRA D/A DE (AR1), AR1 = AR
0110                    *
```

```
NO$IDT      32010 FAMILY MACRO ASSEMBLER   PC2.1 84.107    08:17:02  01-01-80
                                                            PAGE 0003

 0111 0044 42A1  LOOP1    IN    *+,2,1              *(AR0) -- MUESTRA A/D. AR
 0112 0045 4AA0           OUT   *+,2,0              *SACA MUESTRA D/A DE (AR1
 0113             *
 0114 0046 1076           SUB   UNO                 *IF NO TERMINA DE SACAR S
 0115 0047 FC00           BGZ   LOOP0               *   ELSE
      0048 0040
 0116             *
 0117 0049 207D           LAC   NSEGM               *
 0118 004A 1076           SUB   UNO                  *IF NO TERMINA DE SACAR
 0119 004B 507D           SACL  NSEGM                *   ELSE VE POR OTRO MAR
 0120 004C FC00           BGZ   LOOP00              *
      004D 0031
 0121             *                                 *
 0122 004E F900           B     CHANGE              *
      004F 0022
 0123             *
 0124 0050 7F89  CORR1H   ZAC                       *RUTINA DEL REGISTRO DE C
 0125 0051 2076           LAC   UNO                 *CON FUNCION f = S0 + S2
 0126 0052 7972           AND   SHIFT               *
 0127 0053 FF00           BZ    CERO0                *
      0054 0059
 0128             *                                   *
 0129 0055 2076           LAC   UNO                    *
 0130 0056 5073           SACL  Y0                      *IF S2 = 0 THEN Y0 =
 0131 0057 F900           B     CONT0                   *          ELSE Y1 =
      0058 0058
 0132             *                                     *
 0133 0059 7F89  CERO0    ZAC                          *
 0134 005A 5073           SACL  Y0                   *
 0135             *
 0136 005B 2276  CONT0    LAC   UNO,2               *
 0137 005C 7972           AND   SHIFT                *
 0138 005D FF00           BZ    CERO1                 *
      005E 0063
 0139             *                                     *
 0140 005F 2076           LAC   UNO                      *IF S0 = 0 THEN Y2 =
 0141 0060 5074           SACL  Y2                       *          ELSE Y1 =
 0142 0061 F900           B     CONT1                    *
      0062 0065
 0143             *                                     *
 0144 0063 7F89  CERO1    ZAC                          *
 0145 0064 5074           SACL  Y2                   *
 0146             *
 0147 0065 7F89  CONT1    ZAC                       *
 0148 0066 2073           LAC   Y0                   *
 0149 0067 7874           XOR   Y2                    *ADDER = Y0 XOR Y1
 0150 0068 5075           SACL  ADDER                *
 0151 0069 7F89           ZAC                       *
 0152 006A 2F72           LAC   SHIFT,15            *UN RECORRIMIENTO DE SHIF
 0153 006B 5872           SACH  SHIFT               *A LA IZQUIERDA
 0154 006C 2275           LAC   ADDER,2             *
 0155 006D 7A72           OR    SHIFT                *SHIFT = SHIFT OR ADDER
 0156 006E 5072           SACL  SHIFT               *
 0157 006F 7F8D           RET
 0158             *
 0159             END
NO ERRORS, NO WARNINGS
```

```
0001                ************************************************
0002                *                                              *
0003                * PROGRAMA GENERADOR DE NUMEROS PSEUDOALEATORIOS *
0004                * QUE UTILIZA EL METODO DE CONGRUENCIA LINEAL    *
0005                *                                              *
0006                *          Rn+1 = (K1 Rn + c) MOD M            *
0007                *                                              *
0008                ************************************************
0009                *
0010                *VAR (MEMORIA DE DATOS)
0011      0000      R       EQU  0
0012      0001      K1      EQU  1
0013      0002      K2      EQU  2
0014      0003      K3      EQU  3
0015      0004      SUM     EQU  4
0016      0005      SGN     EQU  5
0017      0006      K32     EQU  6
0018      0007      TEMP    EQU  7
0019                *
0020 0000                   AORG 0
0021 0000 F900              B    START
     0001 0008
0022                *CONST (MEMORIA DE PROGRAMA)
0023                *
0024 0002 0011      MR      DATA 17        *VALOR INICIAL DE LA SEMILLA R
0025 0003 0579      MK1     DATA >579       *
0026 0004 0013      MK2     DATA 19          *VALORES INCIALES DE LAS CONSTANTES
0027 0005 0800      MK3     DATA >800       * ESTOS VALORES SON LOS QUE SE MODIFIC
0028 0006 0400      MK32    DATA >400     *   PARA ADAPTAR LA SECUENCIA PSEUDOALEA
0029 0007 FFFF      MUNO    DATA -1
0030                *
0031 0008 6E00      START   LDPK 0       *
0032 0009 7E02              LACK MR       *
0033 000A 6700              TBLR R         *
0034                *                       *
0035 000B 7E03              LACK MK1         *
0036 000C 6701              TBLR K1           *
0037                *                           *
0038 000D 7E04              LACK MK2             *
0039 000E 6702              TBLR K2                *LOS PARAMETROS SE VACIAN A
0040                *                              *MEMORIA DE DATOS
0041 000F 7E05              LACK MK3              *
0042 0010 6703              TBLR K3              *
0043                *                          *
0044 0011 7E06              LACK MK32           *
0045 0012 6706              TBLR K32           *
0046                *                         *
0047 0013 7E07              LACK MUNO      *
0048 0014 6705              TBLR SGN     *
0049                *
0050 0015 6A00      LOOPO   LT   R       *INICIA PROCESO DE GENERACION
0051 0016 6D01              MPY  K1      *
0052 0017 7F8E              PAC           *
0053 0018 0002              ADD  K2        * K1 Rn + C
0054 0019 5004              SACL SUM     *
0055                *
0056 001A 2004              LAC  SUM     *SE INICIA FUNCION MODULO
```

```
0057 001B 1003          SUB  K3
0058 001C FC00          BGZ  LOOP
     001D 0022
0059 001E FF00          BZ   ZERO
     001F 002C
0060 0020 F900          B    NCERO
     0021 002E
0061                *
0062 0022 2004  LOOP    LAC  SUM
0063 0023 1003          SUB  K3
0064 0024 5004          SACL SUM
0065 0025 1003          SUB  K3
0066 0026 FC00          BGZ  LOOP
     0027 0022
0067                *
0068 0028 2004          LAC  SUM
0069 0029 1003          SUB  K3
0070 002A FE00          BNZ  NCERO
     002B 002E
0071                *
0072 002C 7F89  ZERO    ZAC
0073 002D 5004          SACL SUM
0074                *
0075 002E 2004  NCERO   LAC  SUM
0076 002F 5000          SACL R
0077 0030 1006          SUB  MK32
0078 0031 FC00          BGZ  NEG
     0032 0036
0079 0033 2004          LAC  SUM
0080 0034 F900          B    SACA
     0035 003A
0081 0036 5007  NEG     SACL TEMP
0082 0037 6A05          LT   SGN
0083 0038 6D07          MPY  TEMP
0084 0039 7F8E          PAC
0085 003A 5007  SACA    SACL TEMP
0086 003B 4A07          OUT  TEMP,2
0087 003C F900          B    LOOP0
     003D 0015
0088                *
0089                    END
NO ERRORS, NO WARNINGS
```

BIBLIOGRAFIA

-Beker, H.J, and Piper, F.C. cap. II-IV
"SECURE SPEECH COMUNICATIONS"
Academic Press, 1985, Orlando, Florida.

-Deavorus, C.A. and Kruh, L. cap. III
"MACHINE CRYPTOGRAPHY AND MODERN CRYPTANALYSIS"
Artech House, 1985, Norwood, Mass.

-Diffie, W. and Hellman, M.E. cap. III
"PRIVACY AND AUTHENTICATION: AN INTRODUCTION TO CRIPTOGRAPHY"
Proc IEEE, Mar 1979, vol, 67, pp. 397-427.

-Hellman, M.E. cap. III
"AN OVERVIEW OF PUBLIC KEY CRIPTOGRAPHY"
IEEE Commun. Soc, Magazine, Nov. 1978, vol.16 No.6, pp 24-32.

-Hellman, M.E. cap. III
"THE INFORMATION THEORETIC APPROACH TO CRYPTOGRAPHY"
U, Stanford Calif., 1974.

-Jayant, N.S., Cox, R.V., McDermontt, B.J. and
Quinn, A.M. cap. IV
"A COMPARISION OF FOUR METHODS FOR ANALOG SPEECH PRIVACY"
IEEE Trans. Commun. 1981, com-29(1), 18-23.

-Kahn, D. cap. III
"THE CODEBREAKERS"
Mac Millan, 1967, New York.

-Kak, S.C. and Jayant, N. S. cap. IV
"ON SPEECH ENCRYPTION USING WAVEFORM SCRAMBLING"
Bell Syst. Tech J. May-June 1977, vol.56, pp.781-808.

-Kuhlmann, F. y Buzo, A. cap. II,IV
"ANALISIS DE SEÑALES"
DEPFI-UNAM, Nov. 1987.

-Kun-Shan, L. ed. cap. V
"DIGITAL SIGNAL PROCESSING APPLICATIONS WHIT THE TMS320 FAMILY"
Prentice Hall, Englewood Cliffs, New Jersey, 1987.

-Kun-Shan, L., Frantz, G.E. and Simar, Ray jr. cap. V
"THEN TMS320 FAMILY OF DIGITAL SIGNAL PROCESSORS"
Proc. IEEE, Sep. 1987, vol. 75, pp. 1143-1159

-Orceyre, M.J. and Heller, R.M. cap. IV-V

"AN APPROACH TO SECURE VOICE COMMUNICATION BASED ON THE DATA ENCRYPTION STANDAR"

IEEE Commun. Soc. magazine, Nov 1978, 16(6), pp 41-50.

-Papoulis. cap. II,IV

"SIGNAL ANALYSIS"

Mc Graw Hill, 1977, New York.

-Rabiner L.R. and Shafer, R.W. cap. I-II

"DIGITAL PROCESSING OF SPEECH SIGNALS"

Prentice Hall, Englewood Chiffs, New Jersey, 1978.

-Rivest, R.L., Shamir, A. and Adleman, L. cap. III

"A METHOD FOR OBTAINING DIGITAL SIGNATURES AND PUBLIC KEY CRYPTOSYSTEMS"

Commun. ACM, 1978, 21(2), pp 120-126.

-Sambur, M.R. and Jayant, N.S. cap. IV

"SPEECH ENCRYPTION BY MANIPULATIONS OF LPC PARAMETERS"

Bell Syst. Tech J., Nov. 1976, vol.55, pp.1373-1388.

-"TMS32010 ANALOG INTERFACE BOARD USER'S GUIDE" cap. V
Texas Instruments, 1984, Houston, Tx.

-"TMS32010 EVALUATION MODULE USER'S GUIDE" cap. V
Texas Instruments, 1984, Houston, Tx.

-"TMS32010 USER'S GUIDE" cap. V
Texas Instruments, 1983, Houston, Tx.

-Torrieri, D.J. cap. II-IV
"PRINCIPLES OF SECURE COMMUNICATION SYSTEMS"
Artech House, 1975, Norwood, Mass.