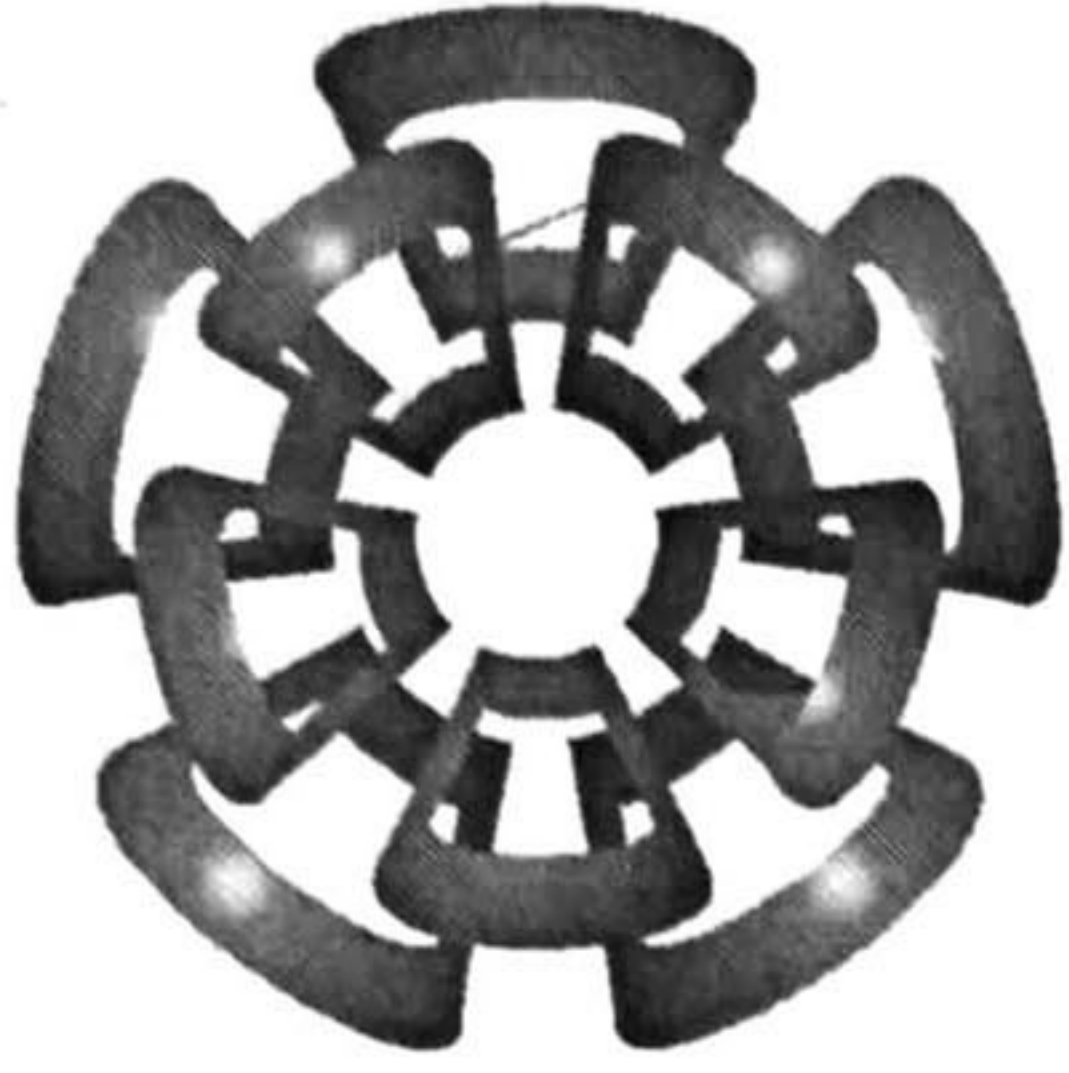


xx (29890.1)



CINVESTAV - IPN

Centro de Investigación y de Estudios Avanzados del IPN
Unidad Guadalajara

**LABORATORIO DE INGENIERÍA ELÉCTRICA Y CIENCIAS
DE LA COMPUTACIÓN - GUADALAJARA**

**ESTUDIO Y SIMULACIÓN DE ALGUNOS ESQUEMAS
DE MODULACIÓN DE FASE CONTINUA (CPM)**

**TESIS QUE PRESENTA
JORGE HERMOSILLO GUTIÉRREZ**



**PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS**

**EN LA ESPECIALIDAD DE
INGENIERÍA ELÉCTRICA**

**CINVESTAV I.P.N.
SECCION DE INFORMACION
Y DOCUMENTACION**

Guadalajara, Jal. Octubre de 1999.

CLASIF.	
ADQUIS.	Tesis-00
FECHA:	16-03-00
PROCED.	Depto. Servicios Bibliográficos

Depto. Servicios Bibliográficos

***ESTUDIO Y SIMULACIÓN DE ALGUNOS ESQUEMAS
DE MODULACIÓN DE FASE CONTINUA (CPM)***

**Tesis de Maestría en Ciencias
Ingeniería Eléctrica**

por:

Jorge Hermosillo Gutiérrez

Ingeniero en Sistemas Computacionales
Universidad Autónoma de Aguascalientes, 1992-1997

Becario del CONACYT, expediente no. 121135

Directores de Tesis:

**Dr. Arturo Veloz Guerrero
Dr. Fernando Ramírez Mireles**

CINVESTAV del IPN Unidad Guadalajara, Octubre de 1999

AGRADECIMIENTOS:

A Dios, por ser la mano que va guiando mi viaje en esta vida: Sé que mi jornada siempre ha sido acompañada fielmente por Él.

A mis padres, Jorge y Cristina, ya que de ellos he recibido la educación, los valores y principalmente el amor de que me precio.

A mis hermanos Silvia, Ale, Karla, Miguel Angel, Sergio y Juan Carlos, que muchas veces han sido mis motivaciones para seguir adelante y con quienes siempre he mantenido una relación muy agradable.

A mis abuelos, que siempre han estado al pendiente de ayudarme servicial e incondicionalmente, y de quienes también he recibido educación que valoro, Cirilo y Cristina aquí con nosotros, e Hilario y Silvina, que estoy seguro que desde allá arriba lo han estado haciendo.

A mis tíos y demás familiares que han ayudado de alguna u otra manera para formar la persona que soy hoy en día. Valoro realmente todo lo que han hecho por mí.

A mis profesores el Dr. Arturo Veloz, el Dr. Manuel Guzmán, y el Dr. Deni Torres, de quienes he recibido conocimientos, confianza y apoyo para realizar mis proyectos.

A mis asesores, el Dr. Veloz y el Dr. Fernando Ramírez Mireles, ya que en esta Tesis he encontrado un área de trabajo muy atractiva, y he aprendido mucho de sus formas de trabajar.

A mis compañeros de la Maestría en Telecomunicaciones, por haberme permitido llevar con cada uno de ellos una relación realmente de amistad y compañerismo. Fue muy agradable el llevar a cabo mis estudios en este ambiente.

A mis futuros compañeros de Doctorado, Ramón Parra y Ramón Rodríguez, de quienes he aprendido mucho y siempre he recibido ayuda incondicional.

A mis amigos de prepa, por haber recibido siempre de ellos la amistad que cualquier persona anhelaría encontrar. Yo no solo encontré un amigo, encontré un grupo de ellos: Jorge H. Ramírez, Jorge Franco, Fernando, Ismael, Hugo, Miguel Angel, José de Jesús, Joel, René y Juan Enrique.

Al CONACYT, por haber sido el medio a través del cual el pueblo de México me dio la maravillosa oportunidad de continuar mis estudios.

Jorge Hermosillo Gutiérrez.

Prefacio	1
Objetivos de la tesis.	1
Metas.	1
Motivación.	2
Organización.	2
1. Introducción	5
1.1. Razones para la comunicación digital.	5
1.2. Elementos de un sistema de comunicaciones digitales.	6
1.3. Razones para llevar a cabo modulación.	7
1.4. Amplitud y Fase instantáneas de las señales.	8
1.5. Técnicas de modulación digital.	10
1.5.1. ASK (<i>Conmutación Digital de Amplitud</i>).....	11
1.5.2. PSK (<i>Conmutación Digital de Fase</i>).....	11
1.5.3. FSK (<i>Conmutación Digital de Frecuencia</i>).....	12
1.5.4. APK o QASK (<i>Conmutación Digital de Amplitud y Fase</i>).....	13
1.6. Modelos matemáticos para canales de comunicación.	14
1.6.1 <i>El canal con ruido aditivo.</i>	14
1.6.2 <i>El canal de filtro lineal.</i>	15
1.6.3 <i>El canal de filtro lineal variante en tiempo.</i>	15
2. Conceptos Básicos de Modulación de Fase Continua (CPM)	17
2.1. Señales moduladas en fase.	17
2.1.1. PSK y FSK.	18
2.2. Modulación de Fase Continua o CPM.....	19

2.2.1. Notación para CPM.	20
2.2.2. Descripción por Trellis de CPM.	24
2.2.3. Características de las señales CPM.	25
2.2.3.1. Características espectrales.	25
2.2.3.2. Características en tiempo.	26
3. Transmisores para CPM	27
3.1. Transmisores CPM en general.	27
3.2. Transmisores con ROMs.	28
3.2.1. Implementación Directa.	29
3.2.2. Implementación con ROMs reducidas.	31
3.2.3. Implementación con almacenamiento de fases.	32
3.3. Implementaciones con PLLs (circuitos de amarre de fase).	32
3.4. Implementaciones con filtro pasabanda y limitador.	34
3.5. Transmisor simplificado de MSK.	35
3.6. Simulaciones realizadas.	36
3.6.1. BPSK y QPSK.	38
3.6.2. Pulso LREC.	39
3.6.2.1. Respuesta Total.	39
3.6.2.2. Respuesta Parcial.	41
3.6.3. Pulso LRC.	43
3.6.3.1. Respuesta Total.	43
3.6.3.2. Respuesta Parcial.	45
3.6.4. Pulso LSRC.	47
3.6.4.1. Respuesta Total.	47
3.6.4.2. Respuesta Parcial.	49
3.6.5. Pulso GMSK.	51
3.6.6. Pulso TFM.	53
3.7. Efectos del filtraje y no linealidades sobre la envolvente y el espectro de la señal CPM.	56
3.7.1. Efectos sobre la señal BPSK.	57
3.7.2. Efectos sobre la señal MSK (1REC).	58
3.7.3. Efectos sobre la señal 3RC.	60
3.8. Conclusiones de las simulaciones.	62
4. Receptores para CPM	63
4.1. Receptores para CPFSK	63
4.1.1. Receptor Coherente Óptimo para CPFSK.	64
4.1.2. Receptor No Coherente Óptimo para CPFSK.	69
4.2. Receptores Óptimos de Viterbi.	71
4.3. Receptores Simples No Coherentes	78
4.3.1. Detección Diferencial	78
4.3.2. Detección por Discriminador	81
4.4. Simulaciones de detectores por estimación de secuencias.	84
4.5. Conclusiones de las simulaciones.	88

5. Conclusiones 91

6. Perspectivas y Trabajo Futuro..... 93

Bibliografía 95

Apéndice 97

Objetivos de la Tesis.

El objetivo principal de esta tesis es el de realizar un estudio de los esquemas de modulación de fase continua. Para alcanzar este objetivo, se estudiará primero la teoría en la cual se basa este tipo de modulación y luego se llevarán a cabo algunas implementaciones para su simulación.

El trabajo de esta tesis resulta como compilación de los estudios que sobre estos esquemas de modulación se encuentran en la bibliografía existente; sin embargo se añaden resultados obtenidos para enriquecerlos. Entre éstos se puede mencionar las formas de las densidades espectrales de potencia para una gran variedad de estos sistemas, así como algunas curvas de probabilidad de error para el demodulador que utiliza el algoritmo de Viterbi. En el presente documento se hace énfasis en las características de las señales CPM, estableciendo una comparativa con otros esquemas de modulación. De esta forma, esta tesis resulta en un compendio de la teoría en que se basan los esquemas de modulación de fase continua.

Metas.

Para lograr el objetivo principal antes mencionado, se plantean las siguientes actividades:

- Estudio general de los esquemas de Modulación de Fase Continua, con el fin de tener la base teórica para alcanzar las siguientes metas.

- Implementación en MatLab de algunos transmisores o moduladores para estos esquemas, con el fin de conocer sus formas de onda, espectros y anchos de banda.
- Implementación en MatLab del receptor o demodulador que aplica el algoritmo de Viterbi, con el fin de conocer el desempeño de estos sistemas en un medio de transmisión con ruido blanco gaussiano aditivo.

Motivación.

El interés por el estudio de los esquemas de Modulación de Fase Continua nace de conocer algunas de las ventajas que presentan ante los esquemas utilizados tradicionalmente, en aspectos como anchos de banda, respuesta a no linealidades, y capacidad de demodulación de secuencias en lugar de símbolo a símbolo. Sin embargo, como cualquier otro tema en las comunicaciones, el obtener ventajas en algún aspecto significa tener sacrificios en otro. Éstos son algunos de los temas que se estudiarán en esta Tesis.

Por otro lado, hay una motivación extra para el estudio de estos esquemas: existe el interés de en un futuro cercano y como base de mi tesis doctoral, llevar a cabo investigación sobre su aplicación conjuntamente con la Modulación de Espectro Disperso, para tratar de obtener sistemas de comunicación que combinen las ventajas de ambos esquemas.

Organización.

El trabajo de esta Tesis se organiza como se indica a continuación:

En el primer capítulo se hace un repaso a manera de introducción de las bases de la modulación digital en general, considerando los sistemas tradicionales para que se pueda llevara cabo una comparativa luego con los sistemas de Fase Continua.

En el segundo capítulo se establece la definición y los conceptos básicos de los esquemas de Modulación de Fase Continua.

En el tercer capítulo se establecen las bases de diferentes esquemas para moduladores de Fase Continua. Sobre la base de uno de estos moduladores se llevaron a cabo simulaciones en MatLab y se presentan los resultados en cuanto a formas de onda, espectros y anchos de banda de los sistemas de Modulación de Fase Continua.

El cuarto capítulo se dedica a los demoduladores para estos esquemas de modulación. Al igual que para transmisores, se eligió uno de los demoduladores para su implementación y simulación en MatLab. Se presentan los resultados obtenidos en la demodulación de estas señales en presencia de ruido blanco gaussiano aditivo.

Y por último, el quinto capítulo presenta las conclusiones obtenidas en esta Tesis y en el sexto capítulo se establecen perspectivas para trabajo que pueda continuar los resultados obtenidos en esta Tesis.

Telecomunicación es la transmisión de información de una fuente a un lugar remoto o a un tiempo remoto [1]. En las comunicaciones digitales la transmisión de la información se realiza en forma digital. En este primer capítulo se establecerán brevemente las bases de las comunicaciones digitales, haciendo énfasis en la parte de modulación y demodulación, que es la que tiene relación directa con el tema de la tesis, y en los modelos matemáticos para los canales de comunicación.

1.1. Razones para la comunicación digital.

Estamos viviendo en tiempos en que los antiguos sistemas de comunicaciones analógicas están siendo reemplazados por sistemas de comunicaciones digitales. Existen muchas razones para que ésto se lleve a cabo, y a continuación se mencionarán algunas de ellas [1].

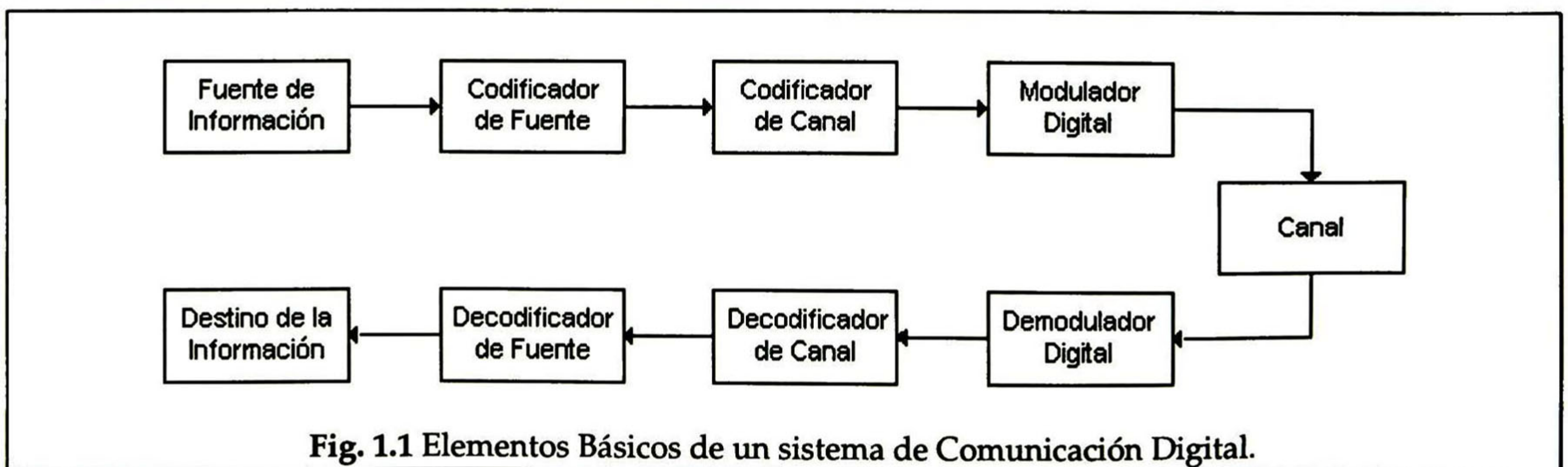
- **Avances en hardware.** El avance tecnológico podría establecerse como uno de los factores importantes en el desarrollo de las comunicaciones. En estos tiempos la tecnología en electrónica ha avanzado de manera sorprendente. Los circuitos en que se basan las comunicaciones digitales son cada vez más pequeños, más rápidos, más baratos y realizan mas funciones.
- **Demanda de nuevos servicios.** Una fuente de información que por naturaleza es digital obviamente requiere transmisión digital. Existen en nuestro medio muchas fuentes de información digitales, como el fax o los mismos datos almacenados en una computadora.
- **Compatibilidad y Flexibilidad.** Un sistema de transmisión complejo y costoso es mejor utilizado si puede ser base para la comunicación de una gran cantidad de tipos

de información. La conversión de todas las fuentes a un formato común, los bits, representa que todas pueden ser soportadas por el mismo equipo. Los bits se pueden agrupar para lograr eficiencia. El multiplexado y el conmutado se pueden hacer más fácilmente. Los métodos de acceso múltiple se abaratan. Se puede lograr la privacidad de la transmisión.

- Fidelidad de reproducción y control de errores. La transmisión digital en este aspecto tiene ventajas significativas sobre la analógica. En las transmisiones con distancias largas, es necesario tener etapas intermedias que aseguren que la información llegue hasta el destino. En estos casos, la transmisión digital puede presentar mejores características de robustez frente a la analógica, ya que en el primer caso una de estas etapas puede detectar cada símbolo recibido y volver a generarlo eliminando la mayor parte del ruido y la interferencia que pudiera traer, mientras que en el segundo caso esto no es posible debido a que las etapas intermedias están formadas por amplificadores que hacen poca distinción entre la señal de información y las interferencias que ésta contenga. Además, en las comunicaciones digitales se pueden aplicar métodos de detección y corrección de errores que mejoren las condiciones del canal.
- Costos. Uno de los factores más importantes en el desarrollo de las comunicaciones (y de casi cualquier área en que se desenvuelva el ser humano) es el económico. En el caso en que se tenga que elegir entre transmisión analógica o digital, en estos momentos la digital puede ser menos costosa, ya sea en ancho de banda, costo del equipo, control de errores, compatibilidad, flexibilidad o necesidad de seguridad.

1.2. Elementos de un sistema de comunicaciones digitales.

En la Fig. 1.1 se muestra un diagrama a bloques de los elementos que conforman un sistema de comunicaciones digitales [2].



La salida de la fuente de información puede ser de tipo analógica o digital. En un sistema de comunicaciones digitales, la información a transmitir debe ser digital, por lo que, si la fuente es analógica, la información debe ser transformada a una consecución

de dígitos binarios. Idealmente, sería deseable representar la salida de la fuente de información con el menor número de bits posible, es decir, se busca la eficiente representación de la información que resulte en poca o ninguna redundancia. Este proceso de convertir eficientemente la salida de la fuente de la información en una secuencia de dígitos binarios se conoce como codificación de fuente.

El propósito del siguiente bloque, denominado codificación de canal, es el introducir, de manera controlada, cierta redundancia en la información, que sea utilizada por el receptor para detectar y corregir los errores que se encuentran durante la transmisión. Así, esta redundancia añadida sirve para incrementar la confiabilidad de los datos recibidos y mejorar la fidelidad de la señal recibida.

El modulador digital es el bloque que recibe la salida del codificador de canal y la adecua a las características del canal para su transmisión. Sirve como interfaz hacia el canal, realizando un mapeo de la información a señales que puedan ser transmitidas a través del canal.

El medio físico por el cual viaja la información del transmisor hacia el receptor es denominado canal.

En el receptor existen los correspondientes bloques que realizan las tareas inversas a los bloques de que se compone el transmisor: El demodulador digital realiza el mapeo de la señal recibida (compuesta de la señal transmitida con ruido e interferencias encontradas en el canal) hacia dígitos binarios. Esta secuencia de dígitos binarios es pasada hacia el decodificador de canal que, al tener conocimiento del código usado por su contraparte el codificador de canal, hace uso de la redundancia añadida para tratar de reconstruir la información original. La salida de este bloque se alimenta al decodificador de fuente, que le dará el formato adecuado para el destinatario de la información.

Como se mencionó anteriormente, el objetivo de esta tesis está centrado en algunos esquemas especiales de modulación, denominados de fase continua. Entonces, de los elementos de un sistema de comunicaciones digitales antes descritos, los de interés para este trabajo son el modulador y el demodulador digitales, por lo que se ahondará un poco mas en este tema de modulación.

1.3. Razones para llevar a cabo modulación.

La modulación digital es el proceso mediante el cual los símbolos digitales son transformados en formas de onda o señales que tienen características compatibles con las características del canal [3].

En el caso de la modulación en banda base (denominada así porque la transmisión se realiza en un ancho de banda que se extiende desde el cero hasta un valor finito de frecuencia), estas formas de onda son pulsos. Pero en el caso de modulación pasabanda (llamada así porque la transmisión se realiza recorriendo el espectro de la señal en

banda base hacia una banda dada de frecuencias), la señal de la información modula a una sinusoidal llamada portadora.

Uno podría preguntarse el por qué de la necesidad de modular una portadora para realizar la transmisión, y existen varias respuestas a esta cuestión:

Primeramente, si la transmisión se realiza por medio de ondas electromagnéticas, se deben utilizar antenas tanto para la transmisión como para la recepción. Y para acoplar eficientemente la energía electromagnética al espacio, las dimensiones de la abertura de la antena deberían ser mayores a $1/10$ de la longitud de onda de la señal a transmitir [2]. La longitud de onda λ está dada por c/f , donde c es la velocidad de la luz (3×10^8 m/s) y f es la frecuencia de la señal. Así, si quisiéramos transmitir una señal con frecuencia hasta de 3KHz (como usualmente se considera la señal de voz) sin usar modulación de portadora, requeriríamos una antena con dimensiones de al menos 10Km, lo que constituye un tamaño de antena imposible de realizar. Pero si utilizáramos modulación de una portadora de 300MHz, ahora estaríamos hablando de antenas con dimensiones de 10cm, que sí son realizables en la práctica.

Además, [3] la modulación pasabanda puede proveer otros beneficios que con la modulación en banda base son imposibles de lograr: Si se requiere que más de una señal utilice el medio de transmisión, la modulación puede ser utilizada para separar las distintas señales asignándoles distintas bandas de frecuencias.

Y por otro lado, la modulación pasabanda puede ser utilizada también para colocar una señal en una banda de frecuencias donde requerimientos de diseño, como filtrado y amplificación, puedan ser alcanzados fácilmente.

1.4. Amplitud y Fase instantáneas de las señales.

La amplitud y la fase instantáneas son conceptos básicos en todas las cuestiones donde se trate con la modulación de las señales, especialmente en comunicaciones y procesamiento de la información [4]. Se debe recordar que una señal como

$$s(t) = a \cos(\omega t + \phi) \quad (1.1)$$

no puede transmitir información; para lograr este objetivo de la transmisión de la información se requiere modulación, ya sea sobre su amplitud, o su ángulo. Por ésto, es necesario tener una definición de estos conceptos que sea clara y no se preste a ambigüedades. Estas definiciones muchas veces se dan empíricamente. Pero es bien sabido que para definir correctamente estos conceptos es necesario utilizar la señal analítica. A continuación se presentan las bases para estas definiciones.

Se desea definir sin ambigüedad la amplitud y la fase instantáneas de una señal $x(t)$. El problema es escribir a esta señal $x(t)$ como la expresión siguiente

$$x(t) = a(t) \cos(\phi(t)) \quad (1.2)$$

usando un par de funciones

$$[a(t), \phi(t)] \quad (1.3)$$

que tenga una correspondencia única con $x(t)$. Para este propósito, asociamos a $x(t)$ su señal analítica $z(t)$. Esta señal analítica se obtiene haciendo pasar la señal $x(t)$ por un filtro con función de transferencia $H(f)$ definida por:

$$H(f) = \begin{cases} 2 & f > 0 \\ 0 & f < 0 \end{cases} \quad (1.4)$$

con lo que se obtiene que, en el dominio de la frecuencia:

$$Z(f) = X(f)H(f)$$

Es obvio que:

$$x(t) = \text{Re}[z(t)] \quad (1.5)$$

Así, si $x(t)$ es real, existe una correspondencia de uno a uno entre ella y su señal analítica $z(t)$. Además, como la transformada de Fourier de $z(t)$ es 0 para frecuencias negativas, esta señal debe ser compleja. Así, puede ser escrita como:

$$z(t) = a(t) \exp[j\phi(t)] \quad (1.6)$$

Como conclusión, utilizando la señal analítica es posible asociar con cualquier señal real un único par como el de (1.3), denominado *par canónico* asociado con $x(t)$.

Definición: Sea $x(t)$ una señal real y $[a(t), \phi(t)]$ el par canónico asociado a ella. La función $a(t)$ que aparece en este par se denomina la *amplitud instantánea* de $x(t)$, y $\phi(t)$ su *fase instantánea*. La *frecuencia instantánea* de la señal $x(t)$ se define como la derivada con respecto al tiempo de $\phi(t)$.

Cuando la fase instantánea es lineal con respecto al tiempo, es posible caracterizar un par canónico a partir de las propiedades espectrales de la señal. Pero esto se vuelve mucho más difícil cuando la fase instantánea de las señales ya no es lineal (y esto ocurre especialmente con señales de amplitud constante, como las presentadas en esta tesis). A estas señales se les denomina *señales de fase* y son la base para los sistemas con modulaciones de fase o de frecuencia.

1.5. Técnicas de modulación digital.

La modulación pasabanda (ya sea analógica o digital) es el proceso mediante el cual una señal de información modifica a una señal sinusoidal [3]. Para la modulación digital, esta sinusoidal de duración T se refiere como un símbolo digital.

Los esquemas de modulación básicos se pueden definir en función del parámetro que modifican en la portadora para transmitir la información: la amplitud instantánea o la fase instantánea. Debe recordarse que una modificación de la frecuencia instantánea es equivalente a una modificación en la fase instantánea. Así, se tienen los siguientes esquemas básicos: ASK (*Amplitude Shift Keying*), si el parámetro modulado es la amplitud; PSK (*Phase Shift Keying*), si es la fase; FSK (*Frequency Shift Keying*) si es la frecuencia; y APK (*Amplitud Phase Keying*), que es una combinación de la modulación de la amplitud y de la fase (QAM o *Quadrature Amplitude Modulation* es un caso de este esquema).

La modulación de fase continua es un esquema en que el parámetro a modular, como su nombre lo indica, es la fase instantánea de la portadora. Pero estos esquemas imponen una restricción sobre esta manera de variar la fase, y es que sea descrita por una función continua; es decir, que no haya cambios bruscos o repentinos en la fase instantánea de la portadora.

En todos los tipos de modulación, si el receptor explota el conocimiento de la fase de la portadora para la detección de las señales, ésta última se denomina detección coherente. Y cuando el receptor no utiliza esta información de la fase, se denomina detección no coherente. En comunicaciones digitales se utilizan los términos de detección y demodulación en forma indiferente, aunque el de demodulación hace énfasis en la remoción de la portadora y el de detección incluye el proceso de la decisión del símbolo.

Como se ha establecido, existen dos parámetros en una señal en los que podemos introducir la información a transmitir: la amplitud y la fase instantáneas (se debe recordar que la frecuencia instantánea es la derivada de la fase instantánea, por lo que los esquemas de modulación de frecuencia también pueden ser vistos como de modulación de fase). En las siguientes secciones se presenta una breve descripción de algunos de los esquemas básicos y tradicionales de modulación. Se presentan para que el lector tenga una cierta base de cómo es que estos sistemas funcionan. Son establecidas las bases de esquemas que realizan modulación en alguno de los dos parámetros, o en ambos. En los siguientes capítulos se irán estableciendo comparativas entre varios de los esquemas que realizan modulación sobre la fase instantánea, que son los de interés para esta trabajo.

Para la descripción de estos sistemas, se utilizan las siguientes convenciones: 1) el subíndice i indica el número de intervalo de señalización actual, 2) T es la duración de este intervalo, 3) todos los esquemas de modulación digital utilizan un tiempo de duración de símbolo T fijo, 4) la información a transmitir tiene un alfabeto o conjunto de

valores posibles con M elementos, es decir, es M -ario, y 5) el subíndice k indica cual de los elementos de este alfabeto es el símbolo a transmitir.

1.5.1. ASK (Conmutación Digital de Amplitud).

La expresión general de este esquema de modulación está dada por [2]:

$$\begin{aligned} s(t) &= \text{Re}\{A_i g(t) e^{j\omega_0 t}\} \\ s(t) &= A_i g(t) \cos(\omega_0 t) \end{aligned} \quad iT \leq t < (i+1)T \quad (1.7)$$

Donde el término de la amplitud puede tener M distintos valores, y $g(t)$ es un pulso real cuya forma influye en las características espectrales de la señal. Y como se puede observar la fase es lineal (y por lo tanto la frecuencia es constante), mientras que se varía la amplitud en cada intervalo de señalización, lo que equivale a modificar la energía de la señal transmitida. En la Fig. 1.2 se muestra un ejemplo de una señal binaria modulada con este esquema.

Las señales ASK pueden ser demoduladas tanto con métodos coherentes como con métodos no coherentes.

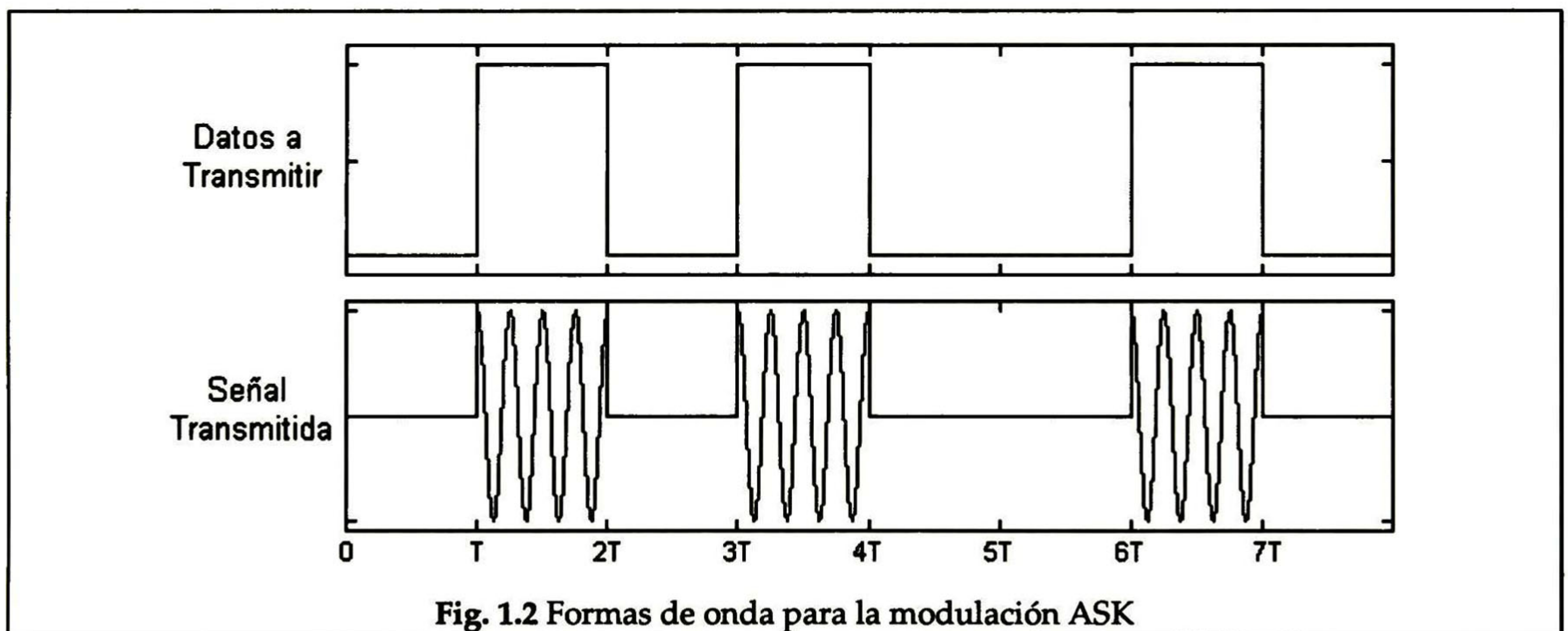


Fig. 1.2 Formas de onda para la modulación ASK

1.5.2. PSK (Conmutación Digital de Fase).

La modulación de fase denominada PSK fue desarrollada durante los primeros días de las comunicaciones espaciales [2]; pero hoy en día es ampliamente usada tanto en aplicaciones militares como en aplicaciones comerciales. La expresión analítica general que caracteriza las señales PSK es la siguiente:

$$\begin{aligned} s(t) &= \text{Re}\{g(t) e^{j2\pi(k-1)/M} e^{j\omega_0 t}\} \\ s(t) &= g(t) \cos(\omega_0 t + \phi_k) \end{aligned} \quad iT \leq t < (i+1)T \quad (1.8)$$

Ahora, en cada intervalo de señalización se mantiene constante la energía de la señal y dentro de él la fase varía linealmente (la frecuencia permanece constante), pero existe un cambio abrupto en la variación de esta fase en los instantes de cambio de símbolo. Igualmente, estos cambios de fase pueden tomar uno de M valores distintos, dados por:

$$\phi_k = \frac{2\pi(k-1)}{M} \quad k = 1, 2, \dots, M \quad (1.9)$$

En la Fig.1.3 se muestra un ejemplo de las formas de onda obtenidas al modular con este esquema utilizando un alfabeto binario. Como se puede observar, uno de los valores del dato es representado por una fase inicial de la señal (en este caso 0) y el otro es representado con una fase que tiene una diferencia de 180° respecto de la primera (en este caso π), con cambios lineales durante el intervalo de señalización. Como se ve, la frecuencia de la señal permanece constante en ese intervalo.

Este tipo de modulación solamente puede ser demodulado usando técnicas coherentes, ya que se requiere una referencia precisa sobre la fase de la portadora; pero se pueden usar técnicas no coherentes si primero se aplica una codificación diferencial a los datos a transmitir.

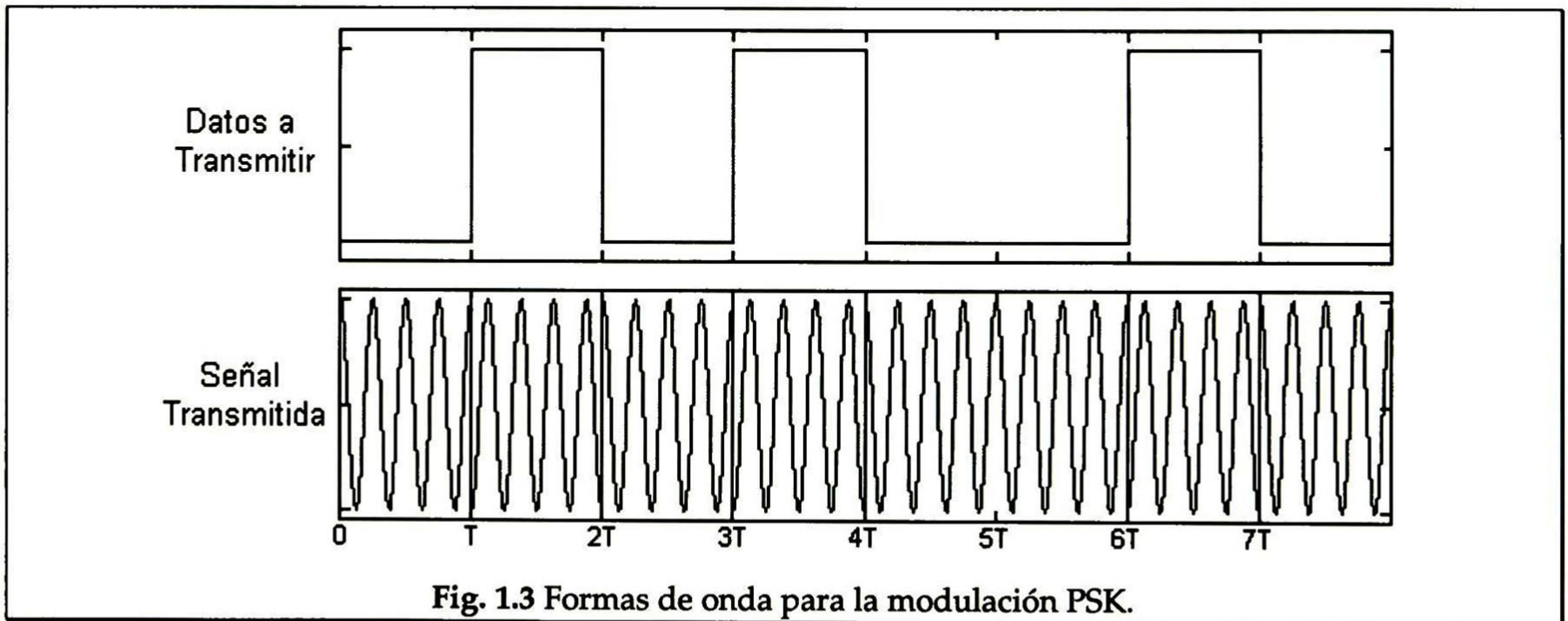


Fig. 1.3 Formas de onda para la modulación PSK.

1.5.3. FSK (Conmutación Digital de Frecuencia).

La expresión general [2] para este tipo de modulación está dada por:

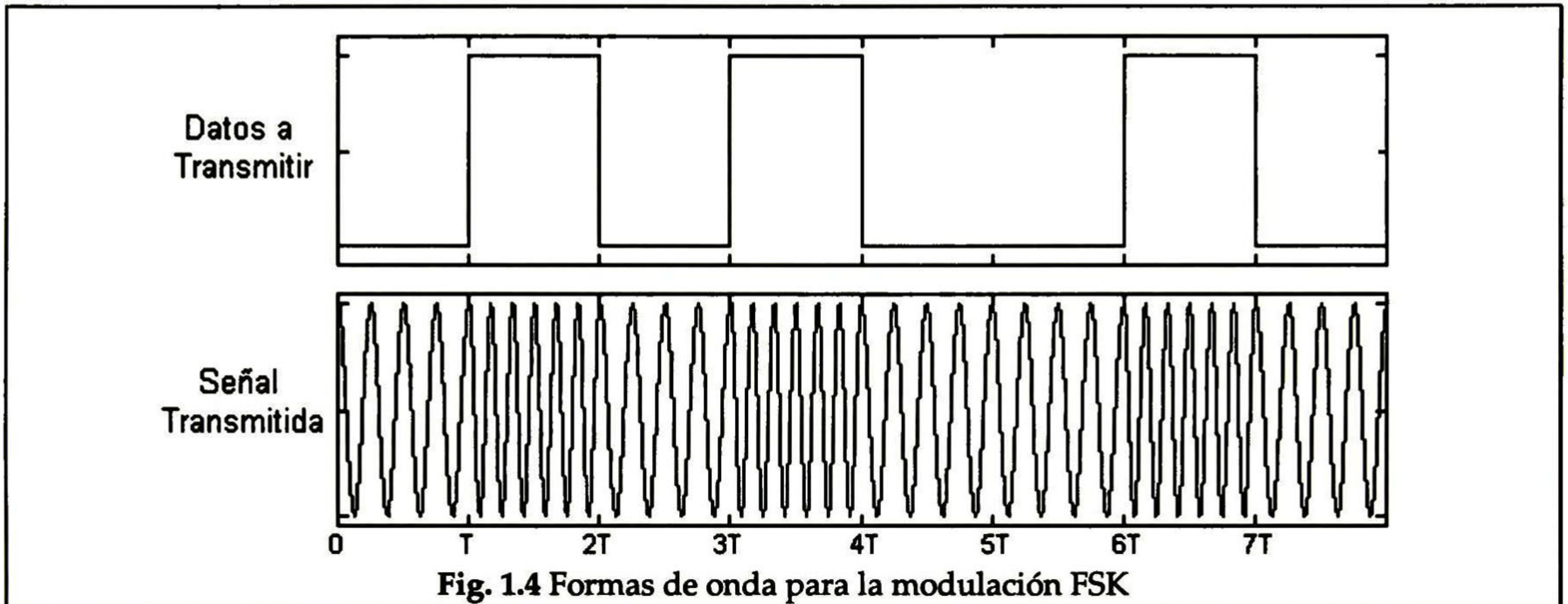
$$s(t) = \text{Re} \left\{ \sqrt{\frac{2E}{T}} e^{jk\Delta\alpha} e^{j\omega_0 t} \right\} \quad iT \leq t < (i+1)T \quad (1.10)$$

$$s(t) = \sqrt{\frac{2E}{T}} \cos(\omega_k t)$$

Ahora, se mantiene fija la amplitud y la fase se hace variar linealmente con distinta pendiente en cada intervalo de señalización, lo que da distintas frecuencias, aunque constantes durante ese intervalo; análogamente a los casos anteriores, la frecuencia puede tomar uno de M valores distintos.

En la Fig. 1.4 se muestra un ejemplo de FSK binario.

Para este esquema de modulación se pueden utilizar tanto detección coherente como no coherente.



1.5.4. APK o QASK (Conmutación Digital de Amplitud y Fase).

El esquema de modulación digital por conmutación de amplitud y fase resulta de la combinación de ASK y PSK, y su expresión analítica general es la dada por la siguiente expresión[2]:

$$s(t) = \text{Re}\{(A_{ic} + jA_{is})g(t)e^{j\omega_0 t}\} \quad iT \leq t < (i+1)T \quad (1.11)$$

$$s(t) = A_{ic}g(t)\cos(\omega_0 t) - A_{is}g(t)\text{sen}(\omega_0 t)$$

Donde A_{ic} y A_{is} representan las amplitudes del pulso $g(t)$ en cada una de las componentes en cuadratura. Ahora la frecuencia instantánea se mantiene constante y se tienen M combinaciones de amplitud y de fase instantáneas en el inicio del símbolo. Durante este intervalo, la fase tiene una variación lineal.

La Fig. 1.5 muestra un ejemplo de este tipo de modulación para un alfabeto con 8 elementos u octal. En esta figura se puede observar como para cada valor de datos existe una combinación de amplitud y fase inicial. Para este caso en particular se escogieron dos amplitudes distintas y cuatro fases distintas.

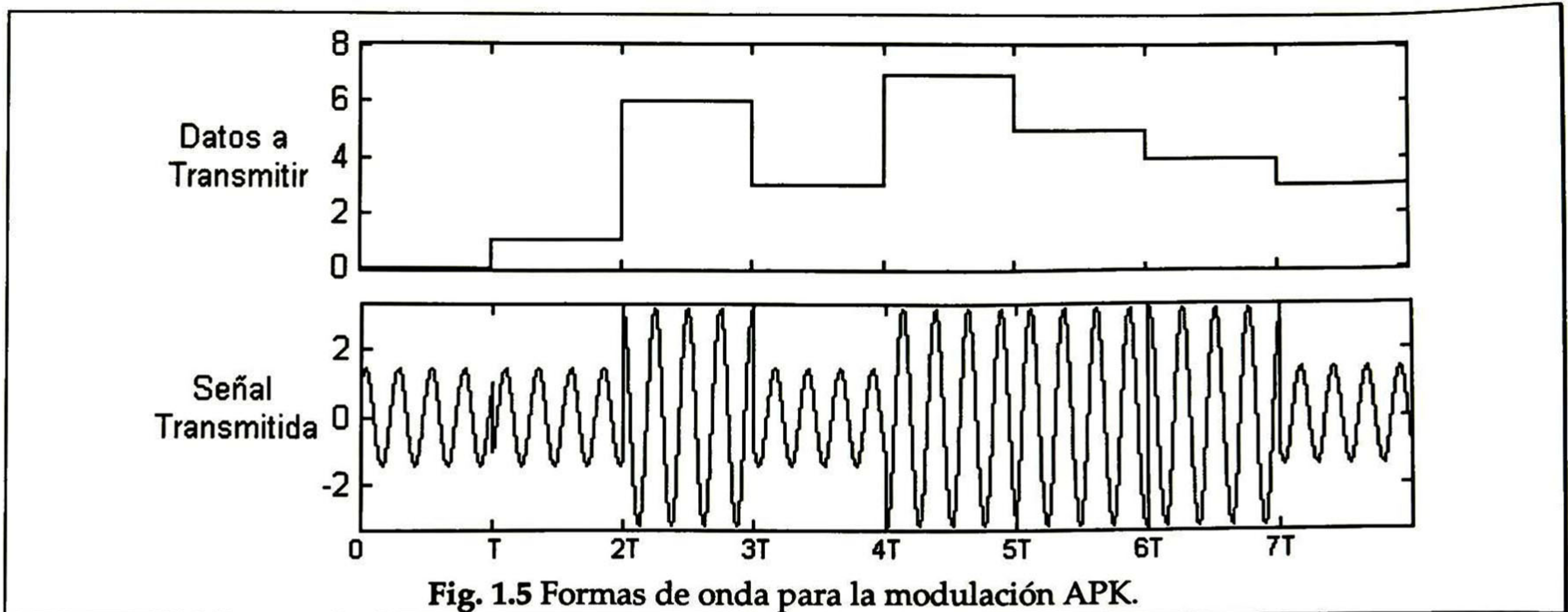


Fig. 1.5 Formas de onda para la modulación APK.

1.6. Modelos matemáticos para canales de comunicación.

Como ya se explicó antes, el canal es el medio físico que utilizamos para la transmisión de la información. En el diseño de sistemas de comunicación para transmitir información a través de medios físicos, es conveniente construir modelos matemáticos que reflejen las características más importantes de estos medios [2]. Y así, el modelo matemático del canal se usa en el diseño del codificador de canal y del modulador, en el lado del transmisor, y de sus contrapartes, el decodificador de canal y del demodulador en el lado del receptor. A continuación se hará mención brevemente de algunos de los modelos matemáticos de canal más utilizados.

1.6.1 El canal con ruido aditivo.

El modelo más simple de canal es el del ruido aditivo. Así la señal transmitida $s(t)$ es corrompida al sumársele un proceso aleatorio $n(t)$, como se muestra en la Fig. 1.6.

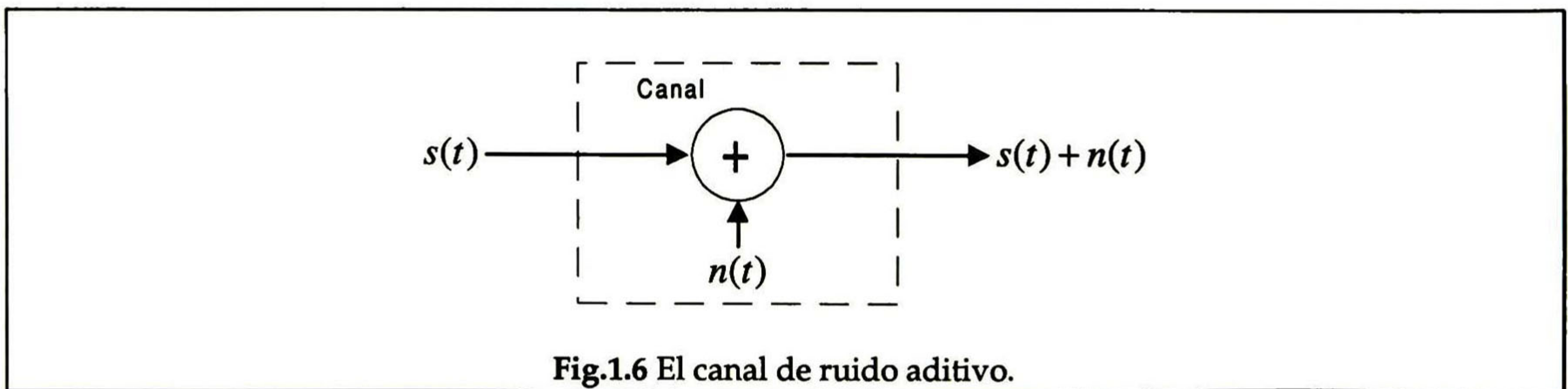


Fig.1.6 El canal de ruido aditivo.

El proceso de ruido puede deberse a fenómenos como el ruido propio de los componentes electrónicos utilizados y los amplificadores, o de interferencias durante la transmisión. Si se debe al ruido propio de los componentes electrónicos, se denomina ruido térmico y se caracteriza estadísticamente como un proceso gaussiano. Dado que

estas características están presentes en la mayoría de los sistemas de comunicaciones y a que es relativamente fácil su tratamiento matemático, el canal de ruido blanco gaussiano aditivo es uno de los modelos de canal mas utilizado. En lo sucesivo en esta tesis se asume este tipo de canal, excepto en los casos en que se indique explícitamente lo contrario.

1.6.2. El canal de filtro lineal.

En algunos canales se utilizan filtros para asegurar que la señal transmitida no exceda las limitaciones de ancho de banda y de esta forma no interfiera con otra señal que use una banda cercana. Estos canales se caracterizan generalmente como filtros lineales con ruido aditivo, como se muestra en la Fig. 1.7.

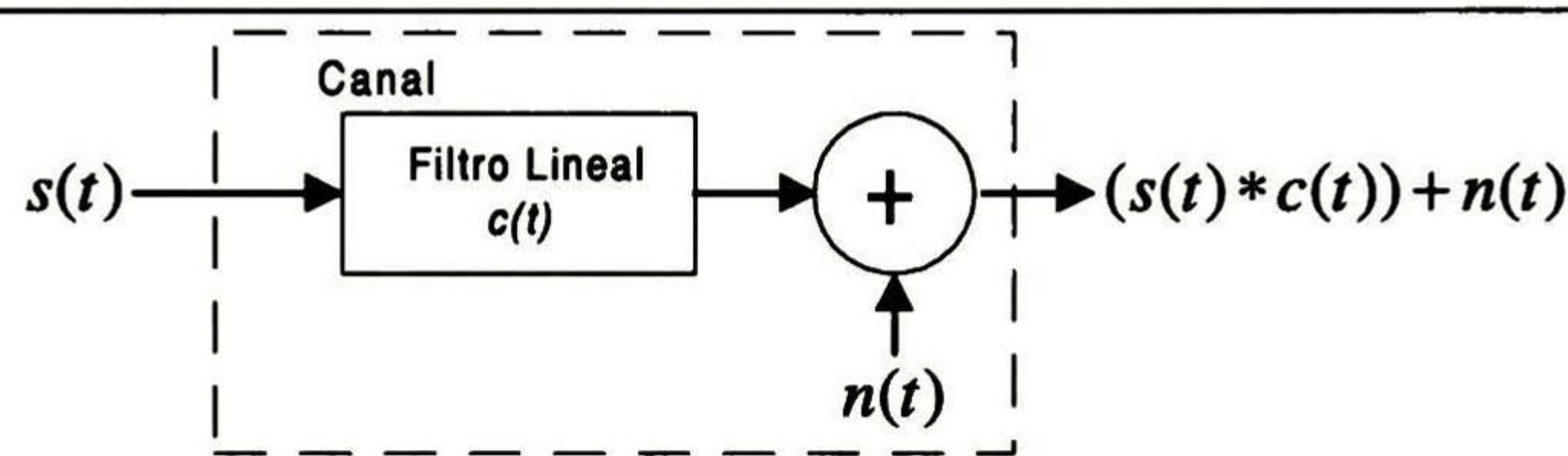


Fig. 1.7 El canal de filtro lineal con ruido aditivo

El filtro lineal que se encuentra en el modelo del canal tiene una respuesta a impulso que lo caracteriza. Entonces, la salida de este canal será la señal $s(t)$ convolucionada con la respuesta al impulso del filtro lineal, y además se le añadirá un proceso aleatorio como ruido.

1.6.3. El canal de filtro lineal variante en tiempo.

Algunos canales que resultan en muchas trayectorias de propagación que están variando en el tiempo, se modelan como filtros lineales variantes en tiempo, como se muestra en la Fig. 1.8.

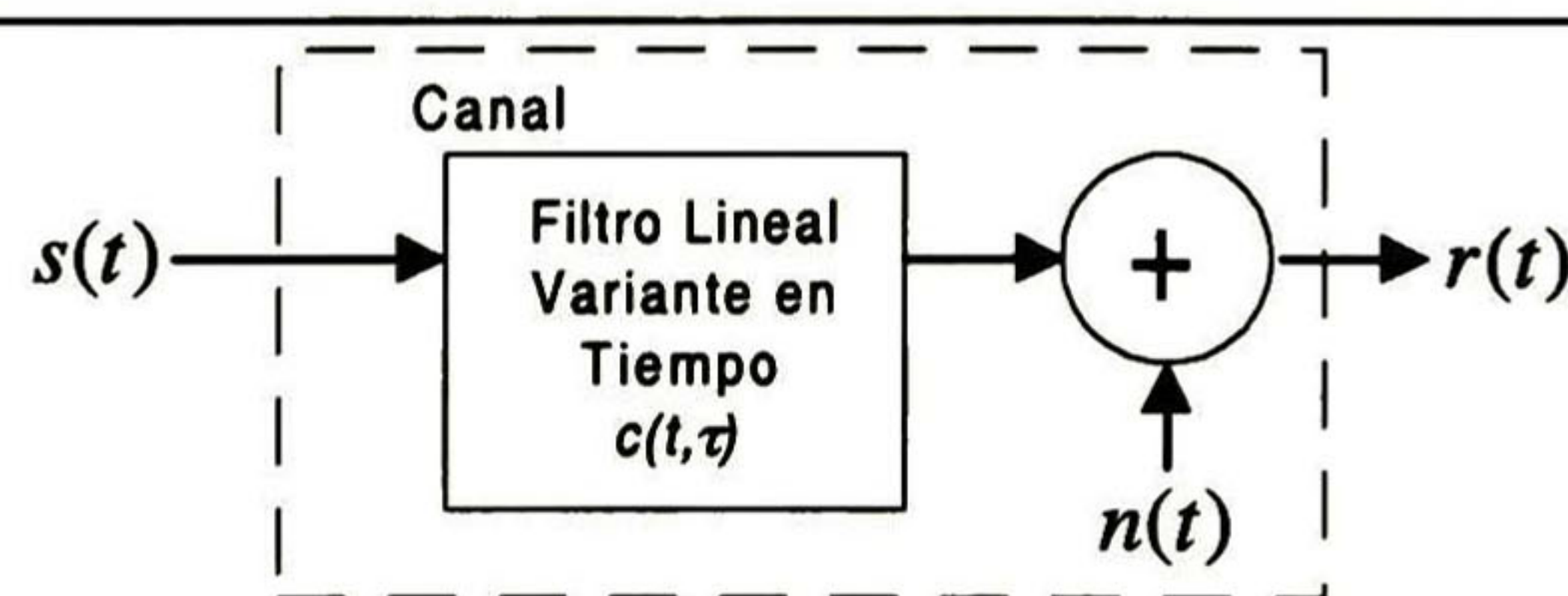


Fig. 1.8 El canal de filtro lineal variante en tiempo.

Estos filtros se caracterizan por una respuesta a impulso variante en tiempo, $c(t, \tau)$ que es la respuesta del canal en el tiempo t debida a un impulso aplicado en el tiempo $t - \tau$.

CONCEPTOS BÁSICOS DE MODULACIÓN DE FASE CONTINUA (CPM)

Como se estableció en el capítulo anterior, existen dos parámetros de una portadora que podemos modificar para transmitir información: la amplitud y la fase instantáneas. Esta Tesis se centra en esquemas de modulación que utilizan cambios en la fase de la portadora; específicamente, se estudian esquemas cuya función que determina los cambios de la fase en el tiempo es una función continua. Estos esquemas se denominan modulación de fase continua, o CPM por sus siglas en inglés, *continuous phase modulation*.

En este capítulo se presentarán los conceptos básicos e introductorios de los esquemas CPM. Asimismo, se establecerá la notación y las convenciones que se utilizarán a lo largo del desarrollo de esta tesis.

2.1. Señales moduladas en fase.

La expresión general para una señal que hace uso de la modulación de la fase para la transmisión de la información está dada por la siguiente ecuación [1]:

$$s(t) = \sqrt{\frac{2E}{T}} \cos(\omega_0 t + \Phi(t)) \quad (2.1)$$

El equivalente pasabajos de esta señal es [2]:

$$s_l(t) = \sqrt{\frac{2E}{T}} \exp\{j\Phi(t)\} \quad (2.2)$$

En estas expresiones T es la duración del símbolo a transmitir, o intervalo de señalización, y E es la energía de la señal durante este intervalo. En cada intervalo un nuevo símbolo perteneciente a un alfabeto M -ario aparece y la fase $\Phi(t)$ sigue algún patrón en respuesta a este símbolo. Además, ω_0 es la frecuencia de la portadora, y muchas veces se logran simplificaciones matemáticas asumiendo que es mucho mayor que $2\pi/T$ (como en el caso de comprobar que E es la energía de la señal en un tiempo T).

Una herramienta que ayuda muchas veces a entender el comportamiento de los esquemas de modulación de fase es el graficar la fase contra el tiempo. Estas gráficas se usarán muchas veces durante el desarrollo de esta tesis.

2.1.1. PSK y FSK.

El primer esquema de modulación de fase utilizado fue el de PSK, descrito ya brevemente en el Cap. 1. En la Fig. 2.1 se muestra un diagrama de fase contra tiempo para este esquema de modulación cuando los datos a transmitir pertenecen a un alfabeto binario.

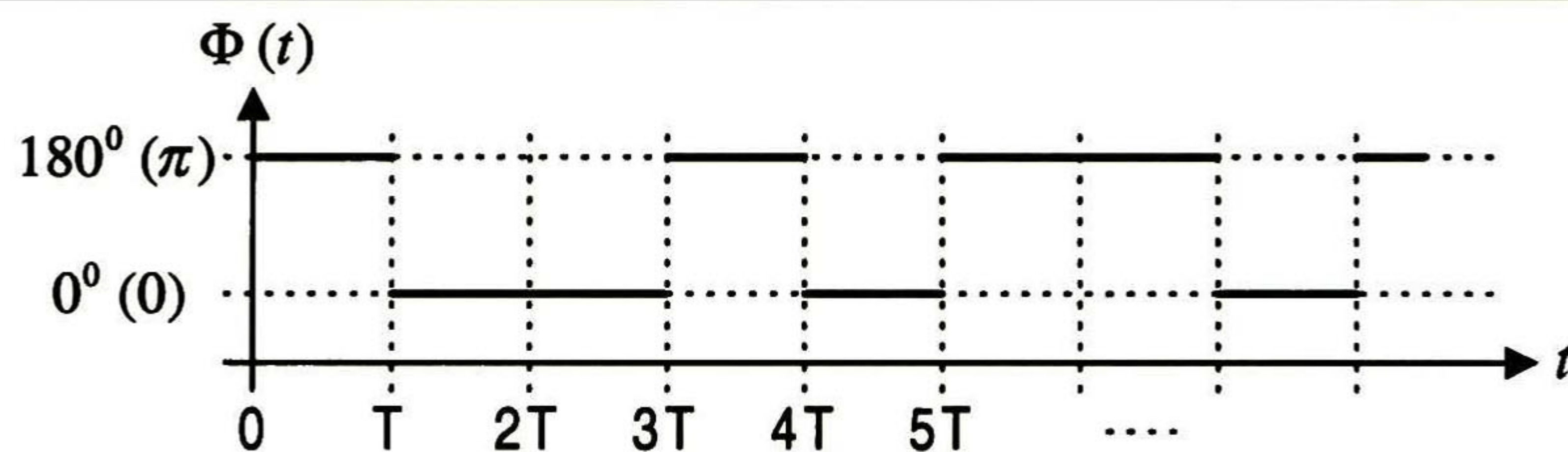


Fig. 2.1 Ejemplo de los cambios de fase en PSK

Como se puede ver, la modulación PSK binaria (o BPSK) utiliza dos fases distintas de la portadora para representar cada uno de los símbolos transmitidos. En cada instante en que llega un nuevo símbolo se efectúa un cambio de fase; por esto se ven las discontinuidades en la fase en cada intervalo de señalización.

Cuando el alfabeto de los símbolos no es binario sino M -ario, con $M > 2$, se tienen M fases distintas de la portadora, manteniéndose los cambios abruptos de fase en los instantes de cambio de símbolo. Así por ejemplo, para un PSK cuaternario las fases posibles pueden ser $0, \pi/2, \pi$ y $3\pi/2$. Para cualquier alfabeto, los niveles de las fases pueden tener un corrimiento, pero siempre debe mantenerse la distancia de $2\pi/M$ entre los valores usados.

En el caso de un alfabeto binario, el sistema se denomina BPSK (*Binary PSK*). Cuando el alfabeto es cuaternario se le denomina QPSK (*Quaternary PSK*). Existe además otro esquema denominado OQPSK (*Offset QPSK*), muy parecido a QPSK, con la única diferencia de que los flujos de datos en fase u cuadratura están defasados en medio

intervalo de señalización cuaternario, mientras que en QPSK están en fase. Ésto para evitar que la señal tenga cambios de 180° en su fase y su envolvente sufra menos distorsión al ser filtrada.

La modulación FSK puede ser vista como una modulación de fase en la que $\Phi(t)$ se representa mediante líneas rectas en una gráfica de cambio de fase. La pendiente de cada línea es la frecuencia instantánea; es decir, la fase en un intervalo de señalización va cambiando linealmente con el tiempo, ya sea con pendientes negativas o positivas.

En la Fig. 2.2 se muestra un ejemplo de la gráfica de cambio de fase de una señal FSK con un alfabeto cuaternario. Como puede verse, la fase en cada intervalo de señalización sufre un cambio lineal con una de cuatro pendientes (frecuencias) distintas.

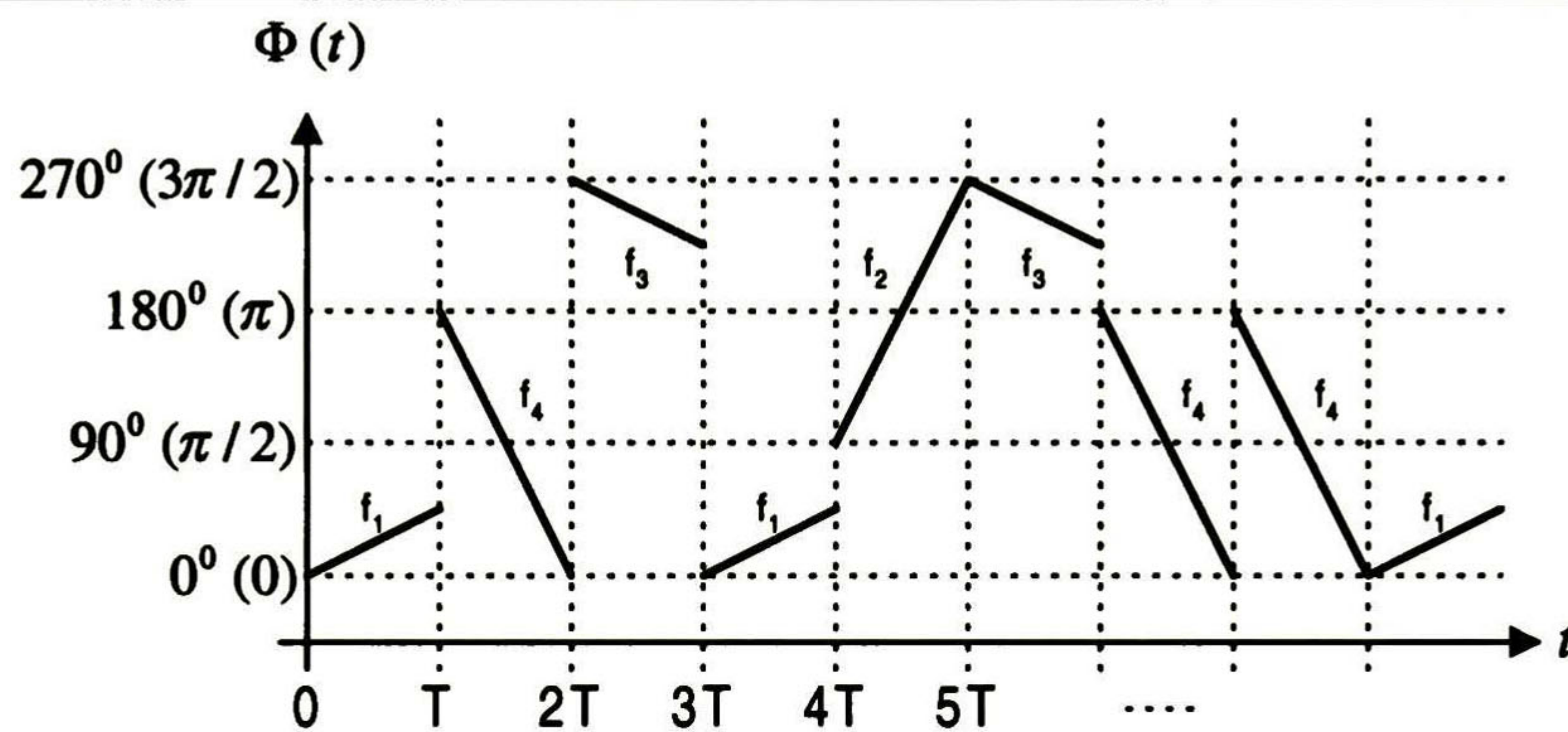


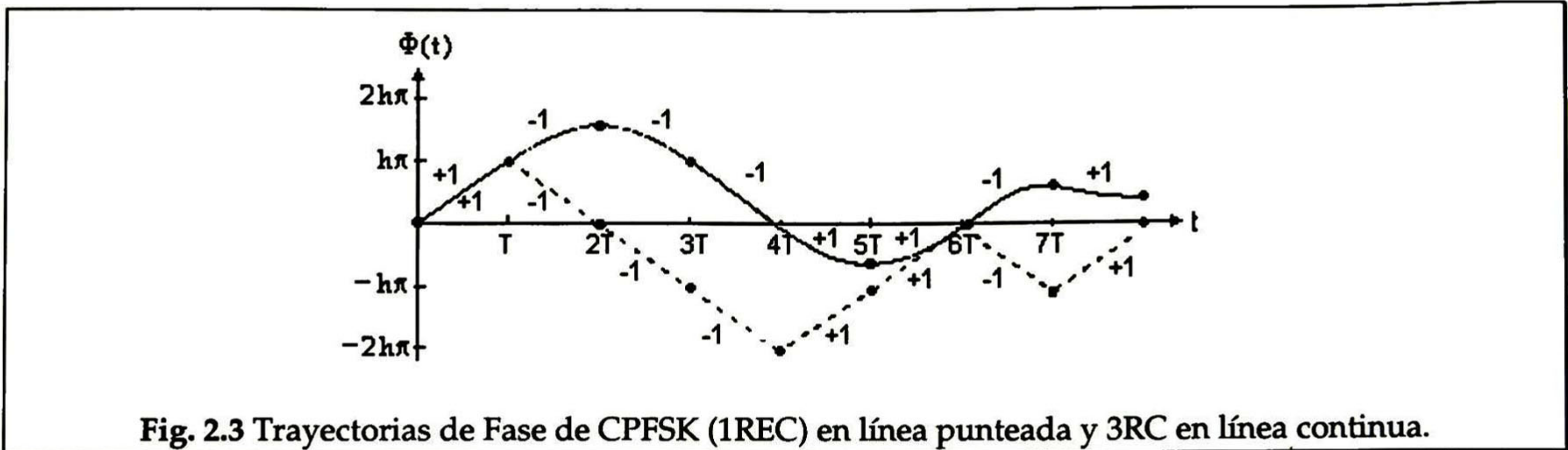
Fig. 2.2 Ejemplo de los cambios de fase en FSK

Como se puede observar en la Fig. 2.2, para FSK sólo importa el mantener una frecuencia en un instante dado. No interesan los cambios abruptos de fase que ocurren en los instantes de cambio de símbolo. Si se lograra mantener un control sobre la fase en cada instante de cambio de símbolo y se restringiera a que en ese punto comenzara la fase del siguiente intervalo, entonces la figura anterior se convertiría en una función continua. Ésto es lo que se pretende con los esquemas CPM. Cuando se logra obtener un FSK en el que la función de cambio de fase es continua, éste esquema se denomina CPFSK (*Continuous Phase Frequency Shift Keying*) o FSK de Fase Continua.

2.2. Modulación de Fase Continua o CPM.

Para la transmisión digital sobre canales limitados en banda, la demanda de esquemas de señalización de envolvente constante y de eficiencia en ancho de banda ha crecido [10]. Un sistema muy utilizado es el PSK; pero éste tiene la desventaja del ancho de banda grande debido a las discontinuidades permitidas en su fase. Por esta razón, los esquemas de CPM representan ventajas significativas. En la Fig. 2.3 se muestran algunos ejemplos de como cambia la fase en dichos esquemas.

Los esquemas de modulación de fase continua o CPM son aquellos en que la envolvente de la señal es constante y la función de cambio de fase $\Phi(t)$ es continua en todo el tiempo [1]; es decir, la amplitud siempre es la misma y no se tienen cambios abruptos en la fase de la portadora como los mostrados en PSK o FSK.



Entonces, de lo anteriormente expuesto, se puede observar que PSK es un sistema de modulación en que la fase instantánea de la señal tiene una transición abrupta en los instantes de cambio de símbolo, y el valor de esta fase permanece constante durante todo el intervalo de señalización, y por esto, la frecuencia instantánea es cero. FSK es un sistema en que, al igual que en PSK, la fase instantánea de la señal sufre un cambio brusco en los instantes de cambio de símbolo, pero la diferencia es que durante el intervalo de señalización no permanece constante, sino que sufre un cambio lineal con respecto al tiempo; así, la frecuencia instantánea no es cero, sino una constante en este intervalo. En las señales CPM se tiene un cambio en la fase instantánea de la señal, pero a diferencia de las anteriores, los cambios abruptos de la fase en los instantes de cambio de símbolo se eliminan, y además se tiene que en general el cambio que sufre esta fase durante la duración del símbolo, se da en forma suave, así, generalmente para las señales CPM se tiene que tampoco la frecuencia instantánea es cero durante el intervalo de señalización.

2.2.1. Notación para CPM.

Para representar las señales de CPM, utilizaremos la notación y convenciones dadas a continuación [1].

La señal a transmitir estará determinada por la secuencia de símbolos de datos a transmitir, denotados por $\alpha_1, \alpha_2, \alpha_3, \dots$, donde cada uno de estos α_i es escogido de un alfabeto *M*-ario con valores posibles $\pm 1, \pm 3, \pm 5, \dots, \pm(M-1)$. Siempre es conveniente restringir a *M* para que sea un entero par. Entonces, la fase en el $(n+1)$ -ésimo intervalo de señalización se representa por:

$$\Phi(t) = 2\pi \sum_{i \leq n} \alpha_i h_i q(t - iT) \quad nT \leq t \leq (n+1)T \quad (2.3)$$

donde h_i es el índice de modulación y $q(t)$ es una función de suavizamiento de fase. El conjunto de los h_i , denominado H , es el conjunto de índices de modulación. Cuando $h_i=h$ para toda i , se dice entonces que el índice de modulación es fijo y es el mismo para todos los símbolos. Cuando éste varía de símbolo a símbolo, la señal CPM se denomina multi- h . En este caso los índices de modulación se hacen variar de una manera cíclica. La función de suavizamiento de fase $q(t)$ que describe la manera en que el cambio de fase se va dando en el tiempo. Esta función puede ser especificada en general como la integral de algún pulso $g(t)$ que representa el cambio en la frecuencia instantánea [2]:

$$q(t) = \int_{-\infty}^t g(\tau) d\tau \quad (2.4)$$

Esta función $q(t)$ puede ser cualquier función que antes de un tiempo t_1 es cero y después de otro tiempo $t_2 > t_1$ su valor sea o tienda a $\frac{1}{2}$. Si se cumple que $g(t)=0$ para $t > T$, la señal CPM se denomina de respuesta total. Si $g(t) \neq 0$ para $t > T$, la señal se llama de respuesta parcial.

De todo lo anterior se puede ver que existe un número infinito de esquemas CPM con todas las combinaciones que se pueden dar entre los pulsos $g(t)$, el conjunto de índices de modulación h_i , y el tamaño del alfabeto. En la Tabla 2.1. se muestran las expresiones analíticas de algunos de los pulsos $g(t)$ más utilizados en los esquemas CPM [8], y en la Fig. 2.4 se muestran gráficamente algunos de ellos. En esta tabla, los nombres de los pulsos son:

LRC: *Raised Cosine* o Coseno Alzado de longitud L .

TFM: *Tamed Frequency Modulation* o Modulación de Frecuencia Suavizada.

LSRC: *Spectral Raised Cosine* o Coseno Alzado Espectral de longitud L .

GMSK: *Gaussian Minimum Shift Keying* o MSK Gaussiano.

LREC: *Rectangular Frequency Pulse* o Pulso Rectangular de longitud L .

En estas expresiones generalmente se tiene que cuando $L=1$ los esquemas son de respuesta total y cuando $L > 1$ son de respuesta parcial. Sin embargo, algunos pulsos, como GMSK, TFM y SRC son de duración infinita en tiempo, y para utilizarlos deben ser truncados y corridos en tiempo. Entonces, cuando se utilizan se especifica la cantidad de intervalos de señalización a que fueron truncados. En la Fig. 2.4 se muestran ejemplos de estos pulsos $g(t)$ con su respectiva función de suavizamiento $q(t)$. En estos ejemplos se ha escogido $L=6$ para LREC y LRC, y $L=2$ para LSRC y los pulsos de duración infinita fueron truncados a 6 intervalos de señalización.

Los sistemas de respuesta parcial, al ser comparados con los de respuesta total, ofrecen mejores características en el momento de negociar entre la probabilidad de error y el ancho de banda [9]. Las características espectrales son notablemente mejores en los sistemas de respuesta parcial, y ésto se logra prácticamente sin incrementar la

probabilidad de error. Sin embargo, se tiene el costo del incremento en la complejidad del sistema; especialmente en el receptor.

Pulso	Ecuación
LRC	$g(t) = \begin{cases} \frac{1}{2LT} \left[1 - \cos\left(\frac{2\pi t}{LT}\right) \right] & 0 \leq t \leq LT \\ 0 & \text{otro caso} \end{cases}$
TFM	$g(t) = \frac{1}{8} [g_0(t-T) + 2g_0(t) + g_0(t+T)]$ $g_0(t) \approx \text{sen}\left(\frac{\pi t}{T}\right) \left[\frac{1}{\pi} - \frac{2 - \frac{2\pi t}{T} \cot\left(\frac{\pi t}{T}\right) - \left(\frac{\pi t}{T}\right)^2}{\frac{24\pi t^3}{T^2}} \right]$
LSRC	$g(t) = \frac{1}{LT} \cdot \frac{\text{sen}\left(\frac{2\pi t}{LT}\right)}{\frac{2\pi t}{LT}} \cdot \frac{\cos\left(\beta \frac{2\pi t}{LT}\right)}{1 - \left(\frac{4\beta}{LT}t\right)^2} \quad 0 \leq \beta \leq 1$
GMSK	$g(t) = \frac{1}{2T} \left[Q\left(2\pi B_b \frac{t - \frac{T}{2}}{\sqrt{\ln(2)}}\right) - Q\left(2\pi B_b \frac{t + \frac{T}{2}}{\sqrt{\ln(2)}}\right) \right] \quad 0 \leq B_b T \leq \infty$ $Q(t) = \int_t^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{\tau^2}{2}} d\tau$
LREC	$g(t) = \begin{cases} \frac{1}{2LT} & 0 \leq t \leq LT \\ 0 & \text{otro caso} \end{cases}$

Tabla 2.1. Definición de pulsos de frecuencia $g(t)$

Sobre la base de todo lo anteriormente expuesto, podemos establecer que, dado que el pulso de suavización del cambio de fase permanece constante después de un tiempo LT , la expresión general de un sistema CPM se puede escribir como:

$$s(t) = \sqrt{\frac{2E}{T}} \cos(2\pi f_0 t + \Phi(t, \alpha_n) + \Phi_0) \quad nT \leq t \leq (n+1)T \quad (2.5)$$

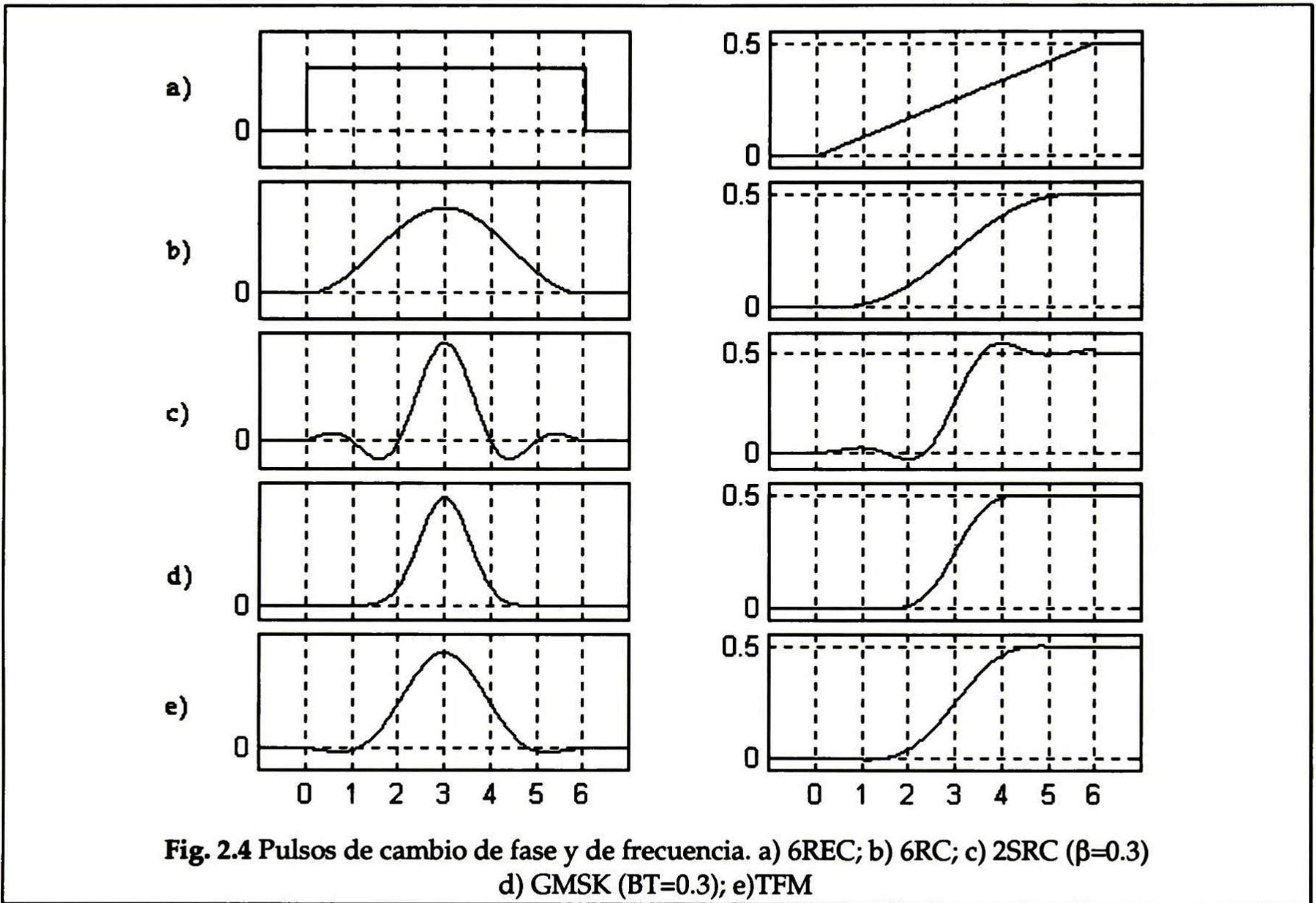
En donde α_n es el vector de símbolos transmitidos hasta el n -ésimo intervalo, Φ_0 es la fase inicial arbitraria, y:

$$\Phi(t, \alpha_n) = \theta(t, \alpha_n) + \theta_n \quad (2.6)$$

$$\theta(t, \alpha_n) = 2\pi \sum_{i=n-L+1}^n h_i \alpha_i q(t - iT) \tag{2.7}$$

$$\theta_n = \pi \sum_{i=-\infty}^{n-L} h_i \alpha_i \tag{2.8}$$

Al término (2.8) se le denomina estado de fase. El concepto de estado de fase es muy importante, ya que utilizándolo podemos establecer una representación de las señales CPM por medio de una máquina de estados finitos. Generalmente para representar esta máquina de estados se utilizará el trellis o entramado.



Existe un esquema de modulación muy popular que cae en la categoría de CPM, particularmente en la de CPFSK o FSK de Fase Continua. Éste es el denominado MSK o *Minimum Shift Keying* por su nombre en inglés. Recibe este nombre ya que para transmitir la información se hace uso de dos frecuencias cuya diferencia es la mínima que asegura ortogonalidad entre ambas señales; ésta es $1/2T$ [3]. Para este esquema se tiene que: $L=1, h=1/2, M=2$, con $\alpha_i = \pm 1$ y $g(t)$ es un pulso REC. Con todo esto se tiene que una señal MSK esta descrita por:

En el siguiente capítulo se presentará la arquitectura de algunos transmisores para CPM, y además se presentarán los anchos de banda obtenidos para sus señales. Ahí se podrá ver el cambio en ancho de banda ocupado para distintos pulsos $g(t)$ e índices de modulación.

2.2.3.2. Características en tiempo.

De la expresión general para las señales CPM (2.5), se establece como primera característica en tiempo de esta señal que su amplitud (y por tanto su envolvente) es constante. Ésta es una característica importante para los métodos de modulación a usar en aplicaciones portátiles o móviles, que emplean generalmente amplificadores no lineales (clase C) para minimizar el consumo de energía y con ésto alargar la vida de la batería [6]. La amplificación no lineal puede degradar el desempeño del sistema si los esquemas de modulación transmiten la información en la amplitud de la portadora. Por ésto, para los sistemas que utilizan la amplificación no lineal es preferible utilizar esquemas de modulación que mantengan la amplitud (envolvente) de la portadora constante; como es el caso de los esquemas CPM y otros como BPSK, QPSK y OQPSK.

Una señal de envolvente constante al ser filtrada para reducir la potencia fuera de la banda de interés, su envolvente deja de permanecer constante. Cuando la fase de la señal cambia en forma continua la distorsión de la envolvente debida al filtraje es menor que cuando la fase varía en forma discontinua. Por ésto, en aplicaciones que utilizan amplificadores no lineales es preferible utilizar los esquemas CPM sobre otros esquemas que aunque también usan envolvente constante su fase no varía de forma continua; como ejemplos de éstos últimos se tiene BPSK, QPSK, OQPSK.

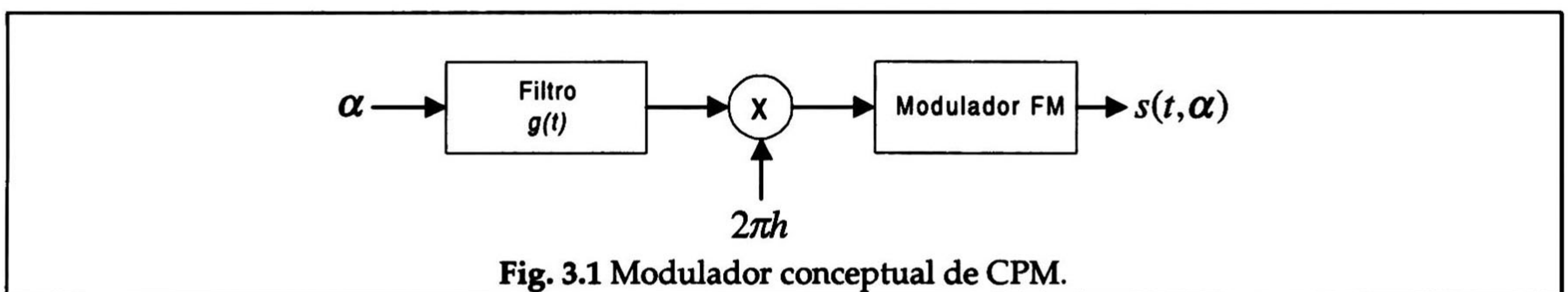
En el siguiente capítulo, además de los espectros de las señales CPM, se presentarán ejemplos de los resultados obtenidos al pasar señales moduladas (tanto con esquemas tradicionales como con los de fase continua) por filtros y por amplificadores no lineales. Ésto con el fin de observar en cuales se presenta una mayor distorsión en la envolvente y en cuales la amplificación no lineal resulta en mayor regeneración de lóbulos no deseados.

TRANSMISORES PARA CPM

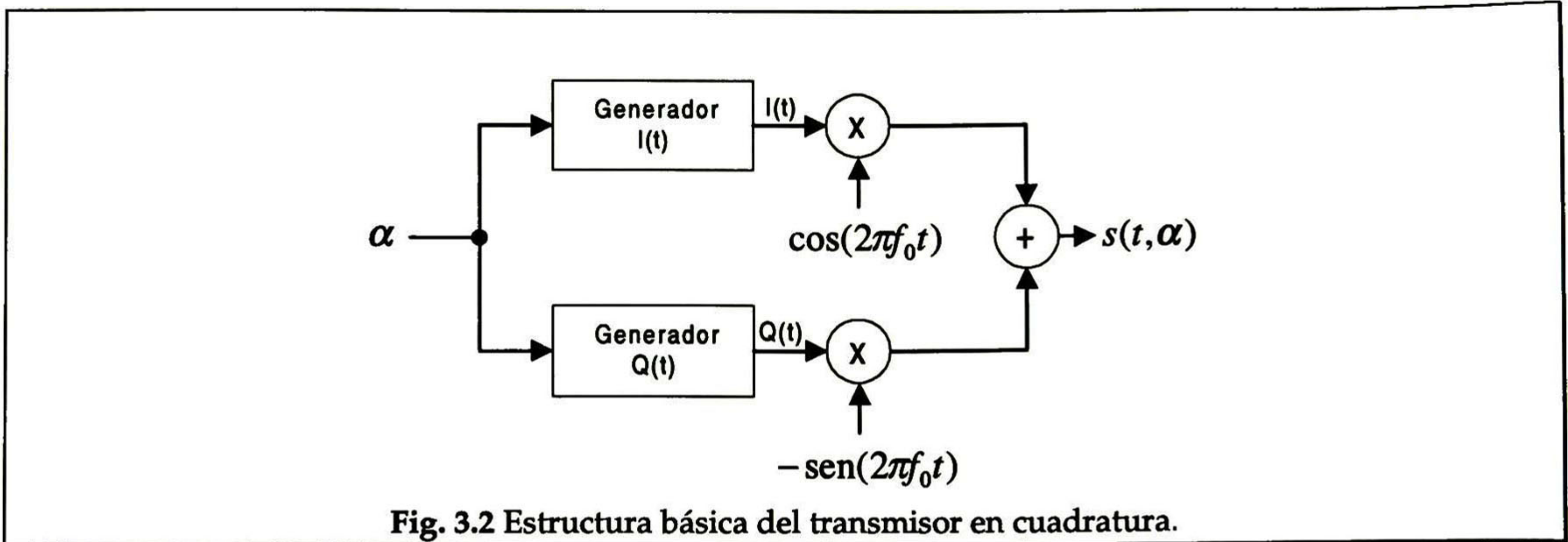
En este capítulo se presentan varias alternativas de arquitectura para un transmisor de CPM [1]. Se considerarán solamente esquemas en que el índice de modulación es constante; sin embargo, los transmisores analizados se pueden generalizar para transmisiones multi-h.

3.1. Transmisores CPM en general.

El modelo general de un transmisor CPM se muestra en la Fig. 3.1, y se basa en el hecho de que el pulso $g(t)$ elegido representa el cambio en frecuencia instantánea de la señal a transmitir [1].



Existe otra alternativa para las modulaciones directas sobre la portadora y es la de generar primero la señal modulada en banda base y después hacer el corrimiento en frecuencia utilizando la portadora. Todas las simulaciones que se presentarán en este capítulo se hacen con base en un transmisor de este tipo y se llevan a cabo en banda base tomando en consideración que el uso de una portadora implicaría utilizar un número mucho mayor de muestras y con esto el tiempo de simulación se incrementaría. El concepto general de este transmisor en cuadratura se muestra en la Fig. 3.2. Las variantes de este transmisor se obtienen al cambiar la forma en que se generan las señales $I(t)$ y $Q(t)$.



Se presentan a continuación entre las arquitecturas de los transmisores CPM, las siguientes: 1) implementaciones con ROMs, 2) implementaciones con PLLs, 3) implementaciones con filtros pasabanda y limitadores y 4) el caso especial del transmisor simple para MSK.

3.2. Transmisores con ROMs.

La forma más fácil y general de implementar un transmisor CPM está basada en el modulador básico en cuadratura, mostrado en la Fig. 3.2 [1]. Éste consiste en utilizar como generadores de las señales en cuadratura $I(t)$ y $Q(t)$ memorias de sólo lectura (ROMs) que son accedidas por una máquina de estados. Para encontrar el contenido de las ROMs hay que comenzar con la siguiente expresión general para las señales CPM en la que se ha considerado, sin perder generalidad, que se ha normalizado la señal y que se ha tomado una fase inicial de 0:

$$s(t, \alpha_n) = \cos(2\pi f_0 t + \Phi(t, \alpha_n)) \tag{3.1}$$

En esta expresión el subíndice n indica que se considera la señal transmitida en el intervalo de señalización correspondiente al símbolo α_n . Aplicando una transformación trigonométrica elemental a (3.1) se obtiene:

$$s(t, \alpha_n) = \cos(\Phi(t, \alpha_n))\cos(2\pi f_0 t) - \text{sen}(\Phi(t, \alpha_n))\text{sen}(2\pi f_0 t); \tag{3.2}$$

o bien:

$$s(t, \alpha_n) = I(t)\cos(2\pi f_0 t) - Q(t)\text{sen}(2\pi f_0 t), \tag{3.3}$$

En donde:

$$I(t) = \cos(\Phi(t, \alpha_n)) \tag{3.4}$$

$$Q(t) = \text{sen}(\Phi(t, \alpha_n)) \tag{3.5}$$

De las expresiones (2.6), (2.7) y (2.8):

$$\Phi(t, \alpha_n) = \theta(t, \alpha_n) + \theta_n \quad (3.6)$$

Sustituyendo (3.6) en (3.4) y (3.5) se obtienen las expresiones para las señales en banda base $I(t)$ y $Q(t)$.

En las implementaciones mediante ROMs se tiene un número determinado de muestras por símbolo m . Estas muestras son almacenadas con una precisión finita; es decir, una cantidad determinada de bits por muestra m_q . El número de muestras debe ser elegido de manera que los espectros al repetirse no causen problemas de "aliasing". Debe mencionarse además, que el filtro analógico pasabajas a la salida del transmisor CPM debe ser el adecuado para eliminar este problema.

El ruido de cuantificación creado por el número limitado de bits por muestra aparece como un ruido blanco de fondo. La magnitud de este ruido puede ser estimada como de orden de la mitad de un paso de cuantificación. Así, con m_q bits por muestra, el nivel del ruido estará en el orden de

$$-20 \log_{10} \left(2^{-(m_q+1)} \right) = -6(m_q + 1) \text{ dB}$$

por debajo del nivel de la portadora.

Existen tres maneras de implementar el transmisor basado en ROMs, y cada una de ellas se obtiene de hacer tratamientos a las expresiones a(3.2), (3.3) y (3.6). Cada una tiene distintos requerimientos en cuanto a tamaño de memorias y otros bloques como se verá a continuación.

3.2.1. Implementación Directa.

La primera forma de implementar los generadores de las señales en cuadratura es utilizando directamente las expresiones (3.4) y (3.5) [1]; es decir, guardando un conjunto de muestras para cada una de las distintas ondas que se obtengan a partir de estas expresiones en un intervalo de tiempo igual al intervalo de señalización T . Todas las muestras se almacenan en una ROM y son accedidas por el símbolo actual, α_n y por el estado del sistema σ_n dado por (2.11), que se recuerda a continuación:

$$\sigma_n = (\theta_n, \alpha_{n-1}, \alpha_{n-2}, \dots, \alpha_{n-L+1})$$

Existen pues pM^{L-1} estados del sistema, con M posibles valores para el símbolo actual. Considerando además que para cada forma de onda se almacenan m muestras, y que cada muestra consta de m_q bits, se tiene que el tamaño total de la ROM es:

$$pM^L m m_q \text{ bits} \quad (3.7)$$

Ahora bien, hay que considerar que se necesitan dos ROMs del mismo tamaño, una para cada señal en cuadratura. Se requiere además un registro de corrimiento que almacene los últimos $L-1$ símbolos, así como una máquina de estados que vaya generando los estados de fase. Esta máquina se puede simplificar si en lugar de generar los estados de fase se generan números secuenciales v_n que se relacionen con el estado de fase por medio de la siguiente expresión y se utiliza entonces simplemente un contador ascendente y descendente o bien una ROM.

$$\theta_n = (h\pi v_n) \bmod 2\pi \tag{3.8}$$

Se requiere adicionalmente alimentar el sistema con un reloj de frecuencia $f=m(1/T)$ para que se obtengan m muestras por cada símbolo. Todas estas muestras se alimentan a un convertidor Digital a Analógico para generar las formas de onda $I(t)$ y $Q(t)$. En la Fig. 3.3 se muestra el diagrama a bloques de este transmisor. Esta arquitectura fue la elegida para implementarse en MatLab y para realizar las simulaciones. Se eligió por ser la más simple en cuanto a operaciones matemáticas y porque el tamaño de las memorias no representa un costo muy grande para las simulaciones.

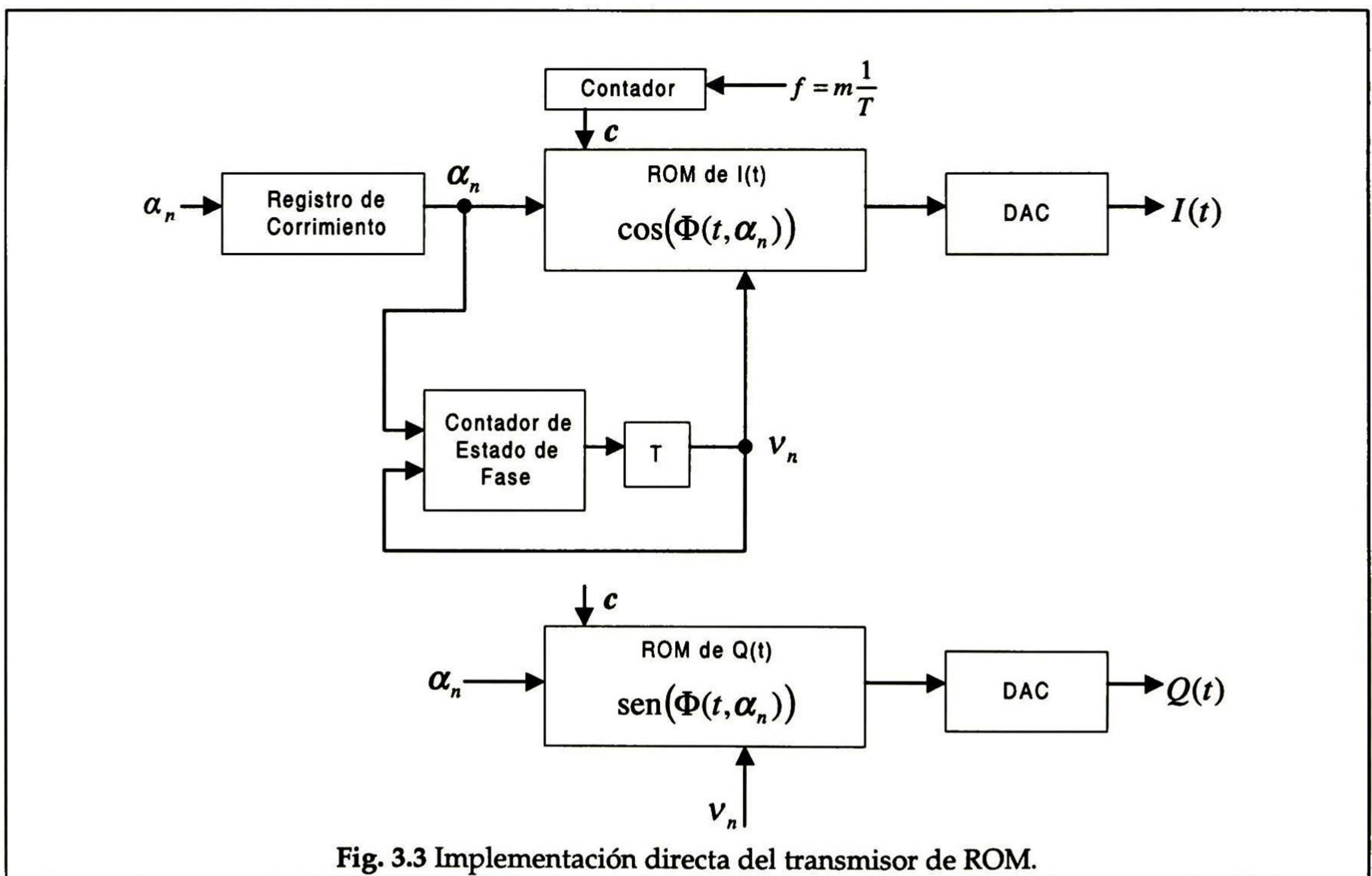


Fig. 3.3 Implementación directa del transmisor de ROM.

3.2.2. Implementación con ROMs reducidas.

El tamaño de las ROMs generadoras de las ondas $I(t)$ y $Q(t)$ puede ser reducido si se toma en cuenta que [1]:

$$\begin{aligned}
 I(t) &= \cos(\Phi(t, \alpha_n)) \\
 &= \cos(\theta(t, \alpha_n) + \theta_n) \\
 &= \cos(\theta(t, \alpha_n))\cos(\theta_n) - \text{sen}(\theta(t, \alpha_n))\text{sen}(\theta_n)
 \end{aligned}
 \tag{3.9}$$

$$\begin{aligned}
 Q(t) &= \text{sen}(\Phi(t, \alpha_n)) \\
 &= \text{sen}(\theta(t, \alpha_n) + \theta_n) \\
 &= \cos(\theta(t, \alpha_n))\text{sen}(\theta_n) + \text{sen}(\theta(t, \alpha_n))\cos(\theta_n)
 \end{aligned}
 \tag{3.10}$$

Entonces, ahora se necesitan cuatro ROMs, dos para el coseno y el seno del estado de fase θ_n , y dos para el coseno y el seno de $\theta(t, \alpha_n)$. Dado que existen p estados de fase distintos y, durante el intervalo de señalización el estado de fase es el mismo, las dos primeras ROMs tendrían un tamaño de pm_q bits. Como existen M^L combinaciones de valores para L símbolos de entrada, las otras dos memorias tendrían un tamaño de $M^L m m_q$ bits, tomando en cuenta la cantidad de muestras deseadas y la cantidad de bits por cada muestra. Se puede notar que hay una reducción en un factor de p en el tamaño de estas memorias con respecto a las memorias utilizadas en la implementación directa; sin embargo, se requiere utilizar otras dos memorias más pequeñas para los estados de fase, además de cuatro multiplicadores y dos sumadores. En la Fig. 3.4 se muestra el diagrama a bloques de este tipo de implementación.

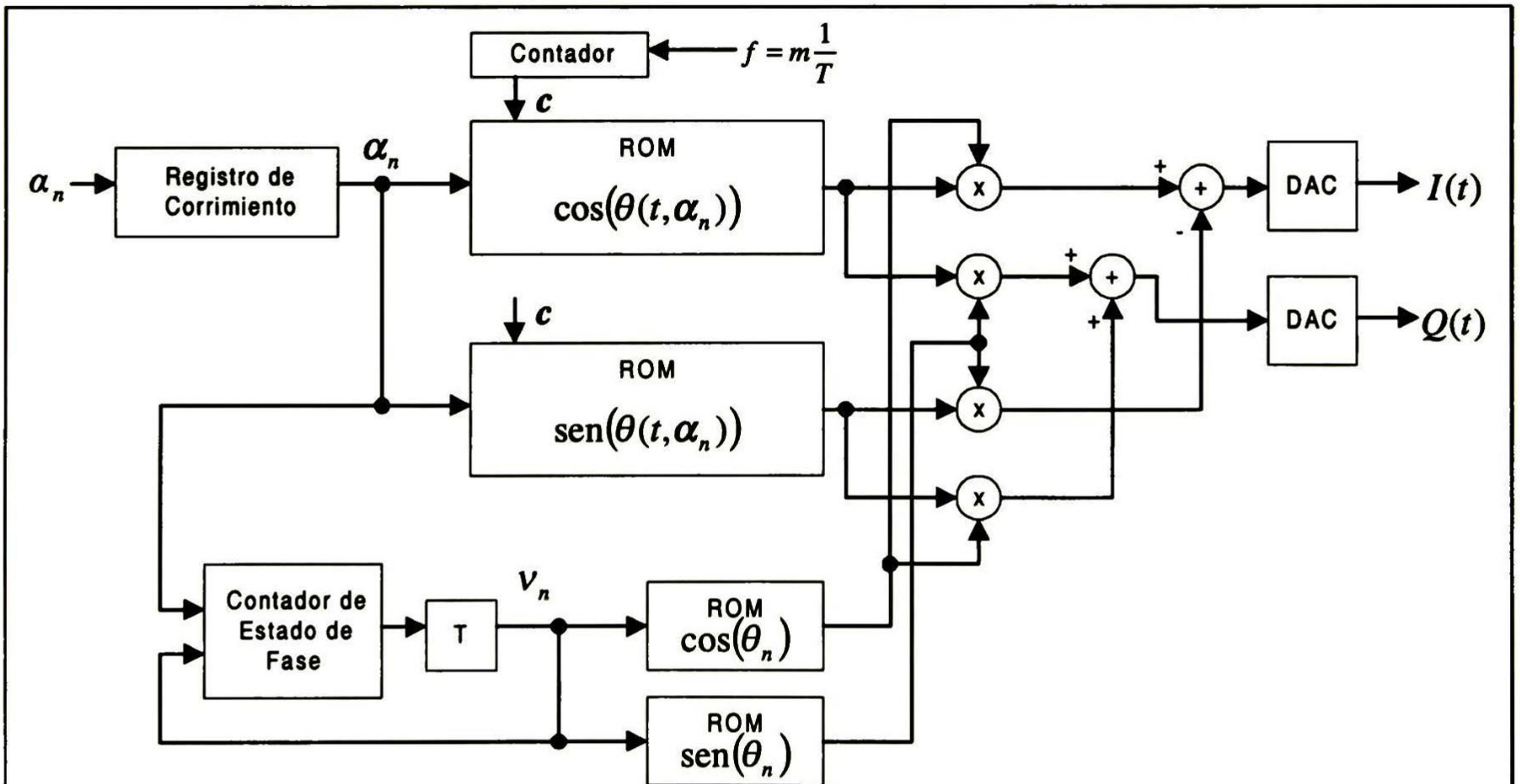


Fig. 3.4 Implementación con ROMs reducidas.

3.2.3. Implementación con almacenamiento de fases.

Otra posibilidad de implementación del transmisor con ROMs está basada en almacenar en estas memorias sólo los valores de la fase, en lugar de sus senos y cosenos [1].

En este caso se necesita una sola memoria que almacene las fases $\theta(t, \alpha_n)$, así como una memoria que calcule los valores de seno y coseno de éstas. La primera memoria tiene un tamaño de $M^L m m_q$ bits. En general, el tamaño total de las ROMs para esta implementación del transmisor es menor que el de los dos casos anteriores. Esta reducción se logra principalmente debido a que se almacena las fases y no los valores de las funciones trigonométricas aplicadas.

El circuito que calcula las funciones trigonométricas de los ángulos debe ser capaz de aceptar valores de fase fuera del rango normal de $\pm\pi$, ya que los cálculos de los ángulos pueden dar valores fuera del rango. El diagrama a bloques de este tipo de implementación se muestra en la Fig. 3.5.

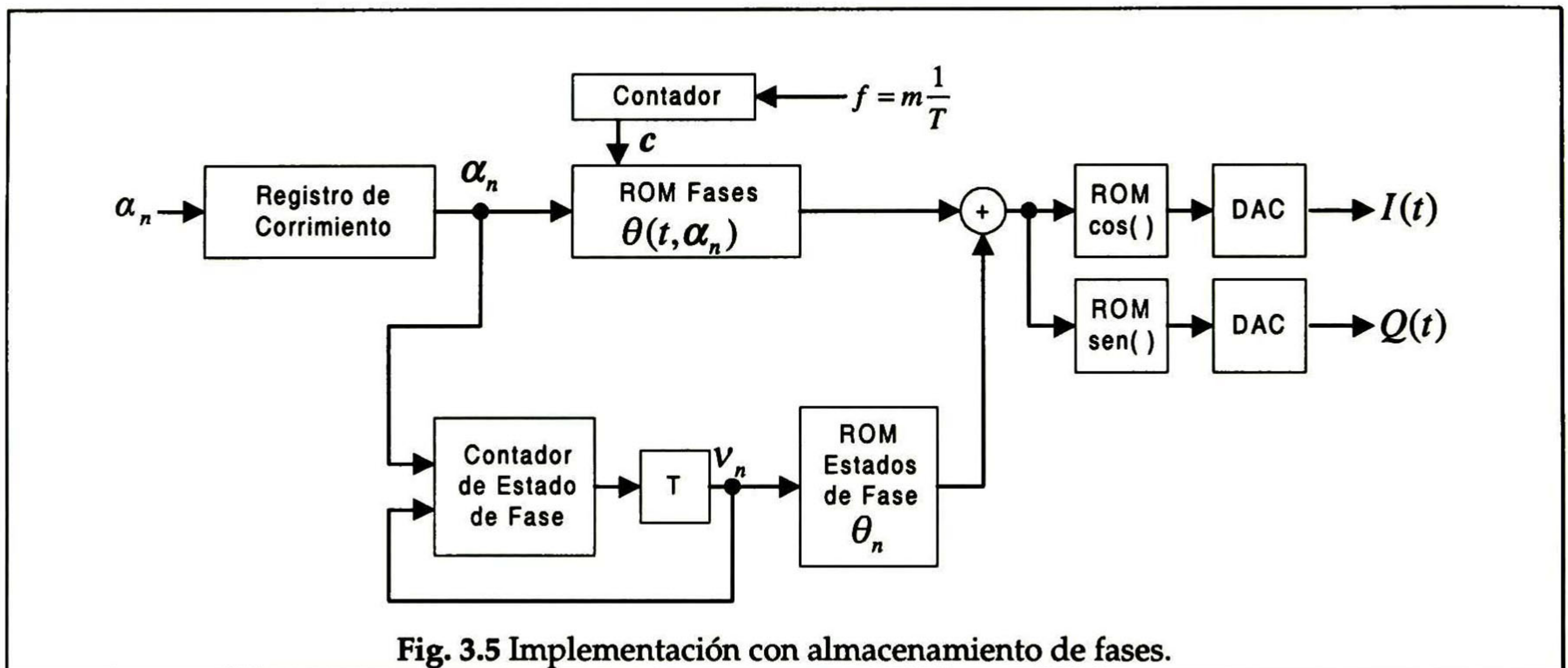


Fig. 3.5 Implementación con almacenamiento de fases.

3.3. Implementaciones con PLLs (circuitos de amarre de fase).

Otra forma de construir un transmisor para CPM es utilizando un modulador sencillo de fase, como es el PSK, para alimentar la entrada de un circuito de amarre de fase (PLL) con cierta función de transferencia [1]. La función del PLL es agregar un suavizamiento a la señal de entrada, creando una señal con correlación en varios intervalos de señalización y con un espectro más angosto.

Este tipo de implementaciones del transmisor de CPM es mucho más rápido que las basadas en ROMs anteriormente mencionadas. Sin embargo, existe un problema en los PLLs debido al rango de frecuencias en que puede operar. Ésto impone una severa

limitante en cuanto al índice de modulación que pueda alcanzarse. La Fig. 3.6 muestra el diagrama a bloques de este tipo de transmisor.

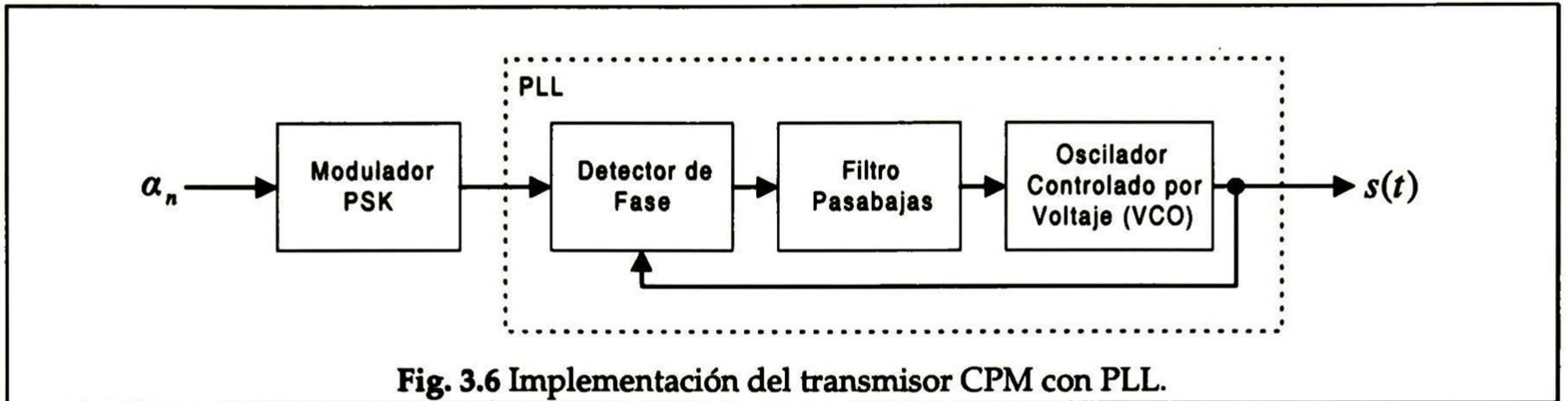


Fig. 3.6 Implementación del transmisor CPM con PLL.

Para hacer el análisis del modulador de la Fig. 3.6, se asumirá que el PLL es un dispositivo lineal que actúa sobre la fase de la señal, con una función de transferencia dada por:

$$H(s) = \frac{\Phi_o(s)}{\Phi_i(s)} \quad (3.11)$$

en donde $\Phi_o(s)$ es la transformada de Laplace de la fase en la salida y $\Phi_i(s)$ es la correspondiente a la señal de entrada.

Si se considera solamente el primer símbolo a transmitir, en el instante $t=0$, entonces la fase de la señal de entrada es

$$\varphi_i(t) = \alpha h \pi u(t), \quad (3.12)$$

en donde $u(t)$ es la función escalón unitario. Esta función se obtiene del modulador PSK. La transformada de esta señal es

$$\Phi_i(s) = \frac{\alpha h \pi}{s} \quad (3.13)$$

Entonces, con base en (3.11) se puede escribir:

$$\Phi_o(s) = \alpha h \pi \frac{H(s)}{s}; \quad (3.14)$$

pasando al dominio del tiempo se obtiene:

$$\varphi_o(t) = \alpha h \pi h_1(t) \quad (3.15)$$

donde:

$$H_1(s) = \frac{H(s)}{s} \quad (3.16)$$

Para las señales CPM, se tiene una función de cambio de fase establecida por la siguiente ecuación:

$$\varphi(t) = 2\pi h \alpha q(t); \tag{3.17}$$

ésta es la misma de la de salida del PLL dada por (3.15) si se cumple que:

$$h_1(t) = 2q(t). \tag{3.18}$$

Entonces, la respuesta a impulso del PLL debe estar en función del tipo de pulso deseado para la modulación CPM.

Se sabe que la función de transferencia $H(s)$ de un PLL está dada por la expresión

$$H(s) = \frac{F(s)}{s + F(s)} \tag{3.19}$$

En esta expresión, $F(s)$ es la función de transferencia del filtro que incluye constantes para el detector de fase y para el VCO. Entonces la función de transferencia $H_1(s)$ está dada por:

$$H_1(s) = \frac{F(s)}{s(s + F(s))} \tag{3.20}$$

Con base en las expresiones anteriores se debe diseñar el PLL de manera que se obtenga el esquema CPM deseado. Cabe hacer notar que no siempre es posible obtener circuitos con estas funciones de transferencia, aunque sí se pueden lograr buenas aproximaciones.

3.4. Implementaciones con filtro pasabanda y limitador.

Otra forma de implementar un transmisor para CPM, cuando el esquema de modulación es muy complejo (L grande, multi-h, pulso $g(t)$ difícil de generar), se basa en otro modulador CPM más sencillo seguido de un filtro pasabanda con una cierta respuesta al impulso y de un limitador de amplitud [1]. En la Fig.3.7 se muestra el diagrama a bloques de este transmisor.

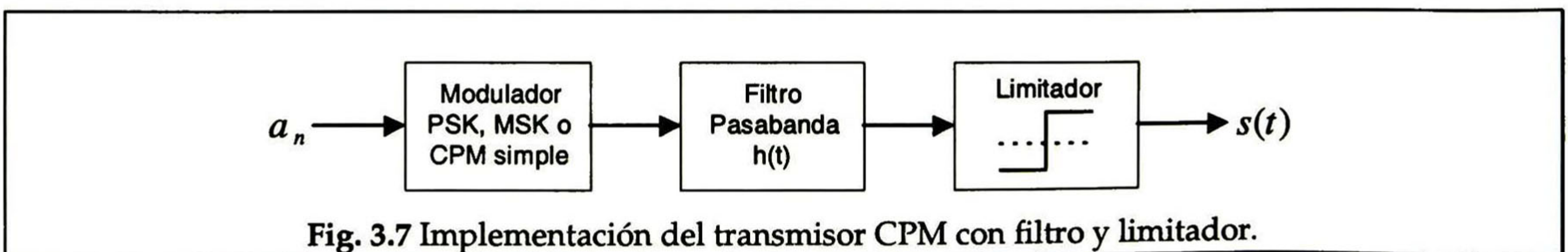


Fig. 3.7 Implementación del transmisor CPM con filtro y limitador.

El espectro y el árbol de fase obtenidos con este tipo de transmisores son muy parecidos a los propios de los esquemas de CPM. Por ésto, se puede usar para simplificar sistemas

CPM más complejos. El filtro pasabanda y el limitador son dispositivos rápidos por lo que al igual que la implementación con PLLs, ésta es muy rápida. Como desventaja de ésta, sin embargo, se tiene que diseñar el filtro con una respuesta al impulso determinada en función del esquema CPM deseado.

El filtro pasabanda para índices de modulación bajos actúa como un filtro sobre la fase en banda base; es decir, la suaviza. Por ésto, se debe calcular la respuesta a impulso del filtro con base en el pulso $g(t)$ deseado.

3.5. Transmisor simplificado de MSK.

El esquema MSK, que es un caso particular de CPM, también puede ser visto como una señal similar a la producida por OQPSK, con la diferencia que utiliza pulsos cosenoidales en lugar de cuadrados. Estos pulsos cosenoidales se escogen de tal forma que se asegura la continuidad en la fase de la portadora. La expresión que representa a MSK como una señal OQPSK con pulsos cosenoidales es [5]:

$$s(t) = \sqrt{\frac{2E}{T}} \left(\alpha_i(t) \cos\left(\frac{\pi t}{2T}\right) \cos(2\pi f_c t) + \alpha_q(t) \sin\left(\frac{\pi t}{2T}\right) \sin(2\pi f_c t) \right) \quad (3.21)$$

donde f_c es la frecuencia de la portadora y los flujos $a_i(t)$ y $a_q(t)$ se obtienen de separar la señal binaria de datos en dos flujos de bits; uno con los bits pares y otro con los impares. Cada uno de estos flujos tiene una velocidad de la mitad del flujo original, pero están defasados en un período de señalización binario uno con respecto al otro.

Con base en la expresión anterior se puede establecer el transmisor mostrado en la Fig. 3.8. Realizando las operaciones marcadas en esta figura y aplicando algunas propiedades trigonométricas elementales, se puede derivar la expresión (3.21).

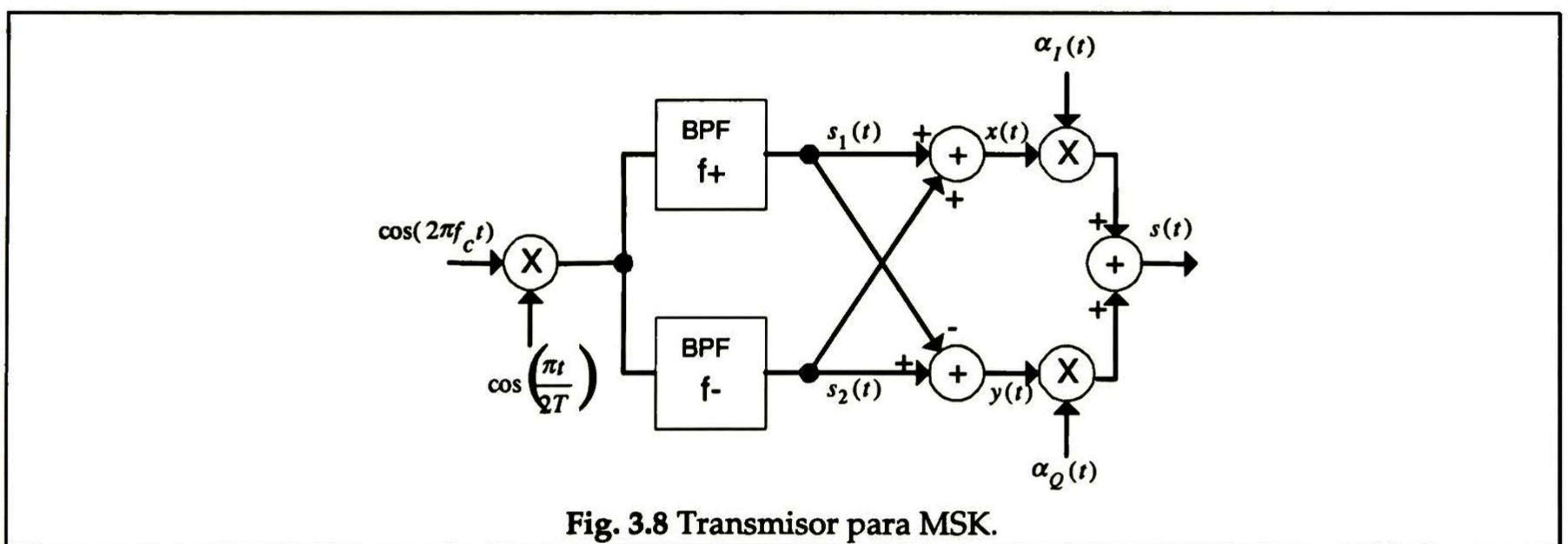


Fig. 3.8 Transmisor para MSK.

Existe una relación entre la representación de MSK dada por (2.9) y la dada por (3.22): Si se utiliza la primera representación, es decir la de CPM, la frecuencia instantánea en el intervalo de señalización está dada por el dato a transmitir; mientras que los flujos en

fase y cuadratura llevan la información codificada diferencialmente. Si por el contrario, se utiliza la representación en cuadratura de MSK, la información viaja en los flujos en cuadratura; mientras que la frecuencia en un intervalo de señalización queda codificada diferencialmente con los dos bits que intervienen en ese momento en los dos flujos.

3.6. Simulaciones realizadas.

A continuación se presentan los resultados obtenidos al realizar simulaciones para obtener la forma de onda de la señal de fase y los espectros de las señales CPM. Además, se presentan las mismas simulaciones aplicadas a los esquemas BPSK y QPSK, con el fin de utilizarlas como marcos de comparación. Estas simulaciones se realizaron para conocer primeramente las formas de onda de la señal de fase y de los espectros de los esquemas CPM, y además para hacer el cálculo de la potencia que concentran cada uno de estos esquemas en un ancho de banda determinado, para así establecer cuales son los más eficientes espectralmente hablando. Para generar las señales CPM, se implementó en MatLab un transmisor con la arquitectura de ROM con implementación directa. Para interpretar adecuadamente las gráficas se deben tomar en cuenta los siguientes puntos:

- Para las gráficas de los cambios de la fase en el tiempo, se presenta la escala vertical en unidades de π y la escala horizontal en unidades del período de señalización. Se presenta sólo el detalle del cambio de fase de los primeros 20 símbolos transmitidos.
- Para las gráficas de las densidades espectrales, se hizo una normalización en la frecuencia con respecto al período de bit.
- Todas las simulaciones se realizaron con 1000 símbolos transmitidos, los cuales se generaron aleatoriamente.
- Se realizaron simulaciones para todos los tipos de pulsos, para respuesta total y parcial con $L=1$ y con $L=3$ respectivamente, para tamaños de alfabeto $M=2$ y $M=4$, para los índices de modulación $h=0.5$ y $h=0.75$, constantes y no multi- h .
- En el caso de LSRC, se utilizó siempre $\beta=0.75$, y cuando $L=1$ el pulso se truncó a 2 intervalos de señalización; mientras que para $L=3$ se truncó a 6 intervalos.
- Para GMSK, se utilizó $BT=0.3$ y se truncó a 4 intervalos de señalización.
- Para TFM, el pulso $g(t)$ se truncó a 6 intervalos. En los tres casos anteriores (LSRC, GMSK y TFM) el pulso truncado $g(t)$ fue recorrido en tiempo para que se cumplieran las condiciones establecidas para este tipo de pulsos.
- Los espectros de las señales se obtuvieron a partir de la función *psd* de MatLab, que utiliza el método de Welch para hacer la estimación de espectros. Éste se basa en 1) aplicar ventanas a secciones de la señal, 2) efectuar la transformada rápida de Fourier

sobre cada ventana, 3) efectuar el periodograma de cada sección escalando la magnitud cuadrada de cada transformada y 4) promediando los periodogramas de las secciones que se traslapan.

Para establecer una comparación más clara y poder apreciar todo el conjunto de gráficas mostradas en las Figs. 3.9 a 3.42 de una forma cuantitativa se calculó, para cada uno de los esquemas, el porcentaje de la potencia total que se encuentra concentrado desde el 0 hasta $fT_b=1$. Ésto se hizo considerando como potencia total a la integral de la densidad espectral de potencia desde 0 hasta *infinito*. En las Tablas 3.1. y 3.2. se muestran los resultados obtenidos para alfabetos binarios y cuaternarios respectivamente.

3.6.1. BPSK y QPSK.

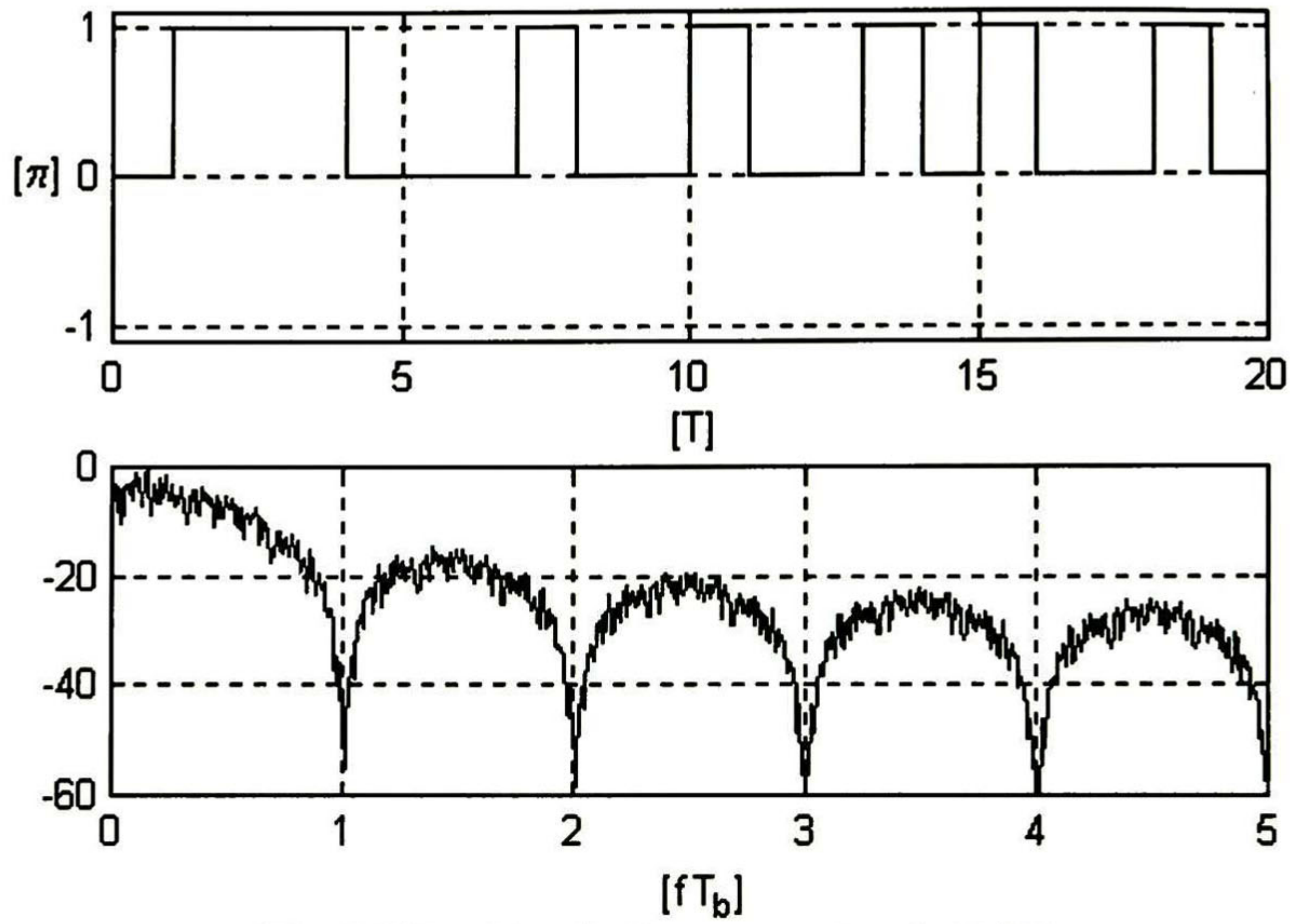


Fig. 3.9 Cambios de Fase y espectro de BPSK.

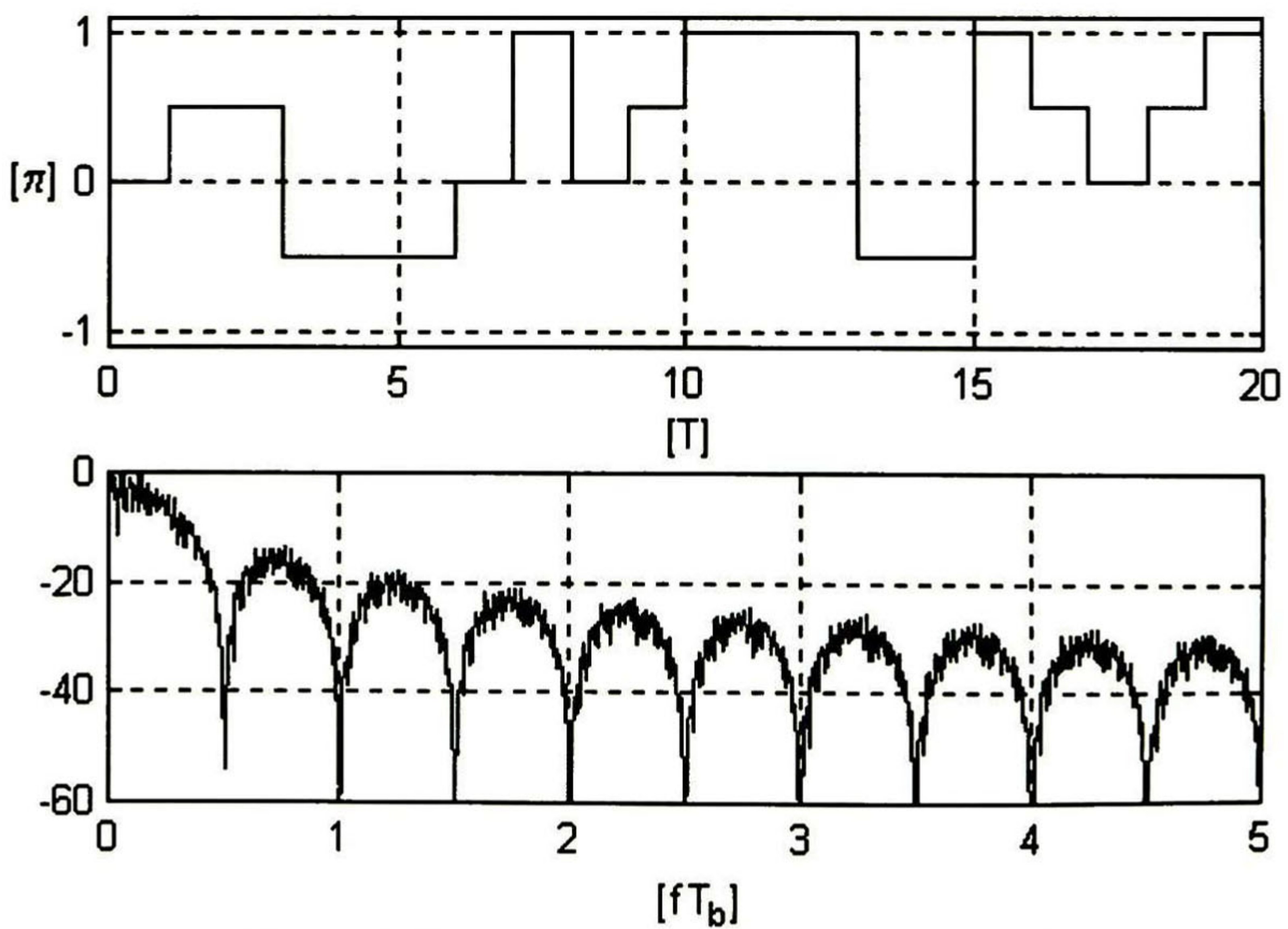


Fig. 3.10 Cambios de Fase y espectro de QPSK.

3.6.2. Pulso LREC.

3.6.2.1. Respuesta Total.

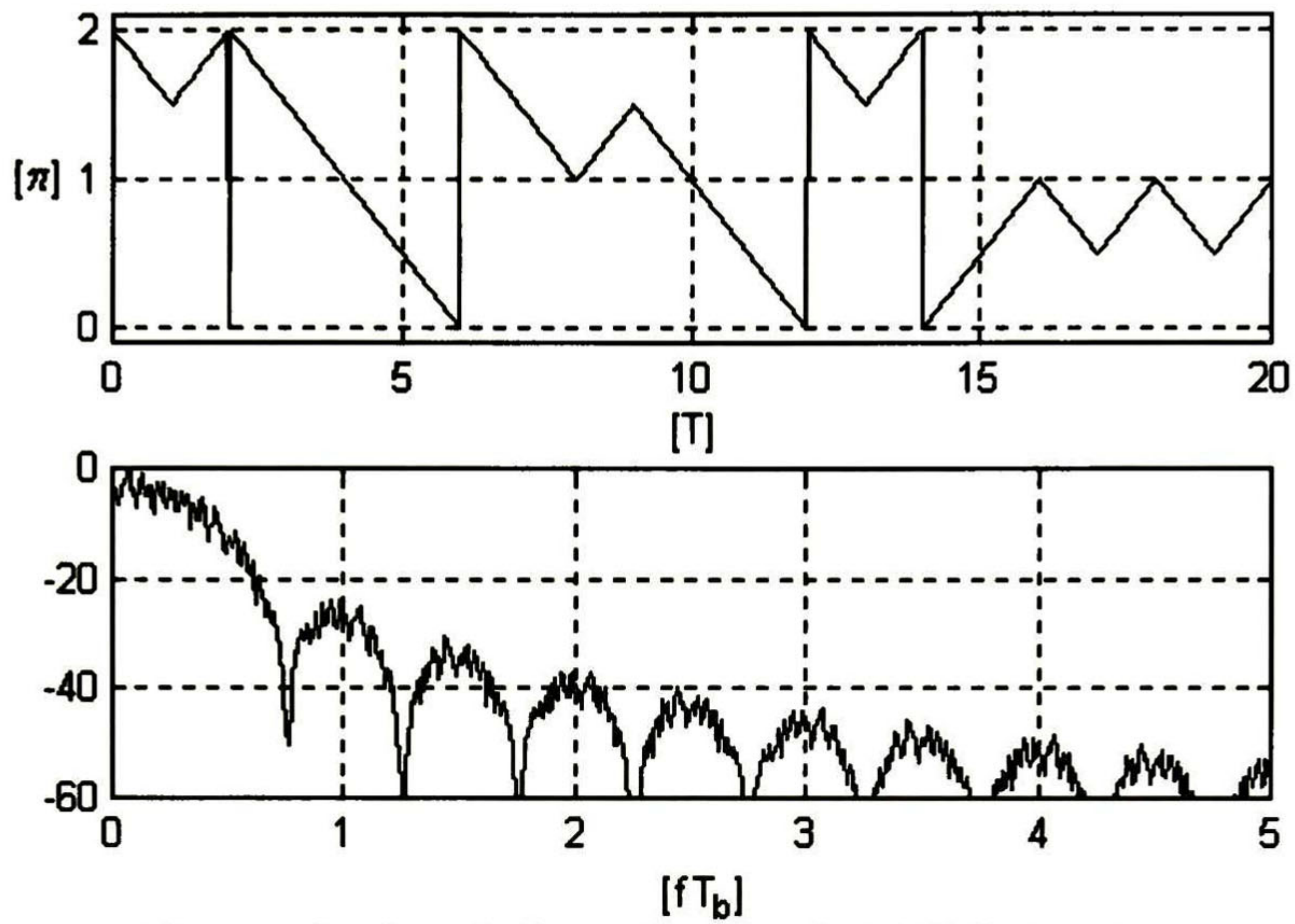


Fig. 3.11 Cambios de Fase y Espectro de 1REC, $h=0.5$, $M=2$.

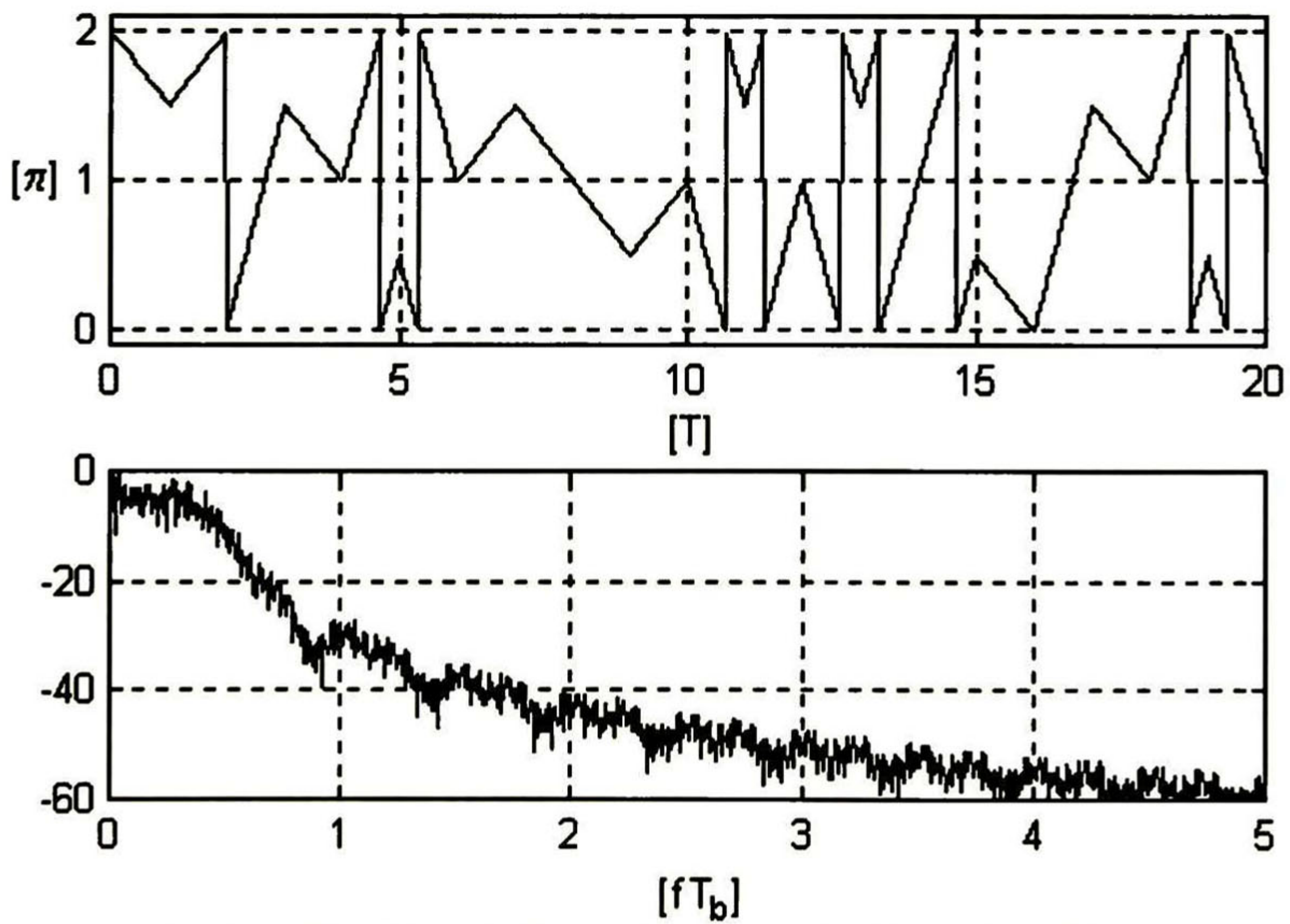


Fig. 3.12 Cambios de Fase y Espectro de 1REC, $h=0.5$, $M=4$.

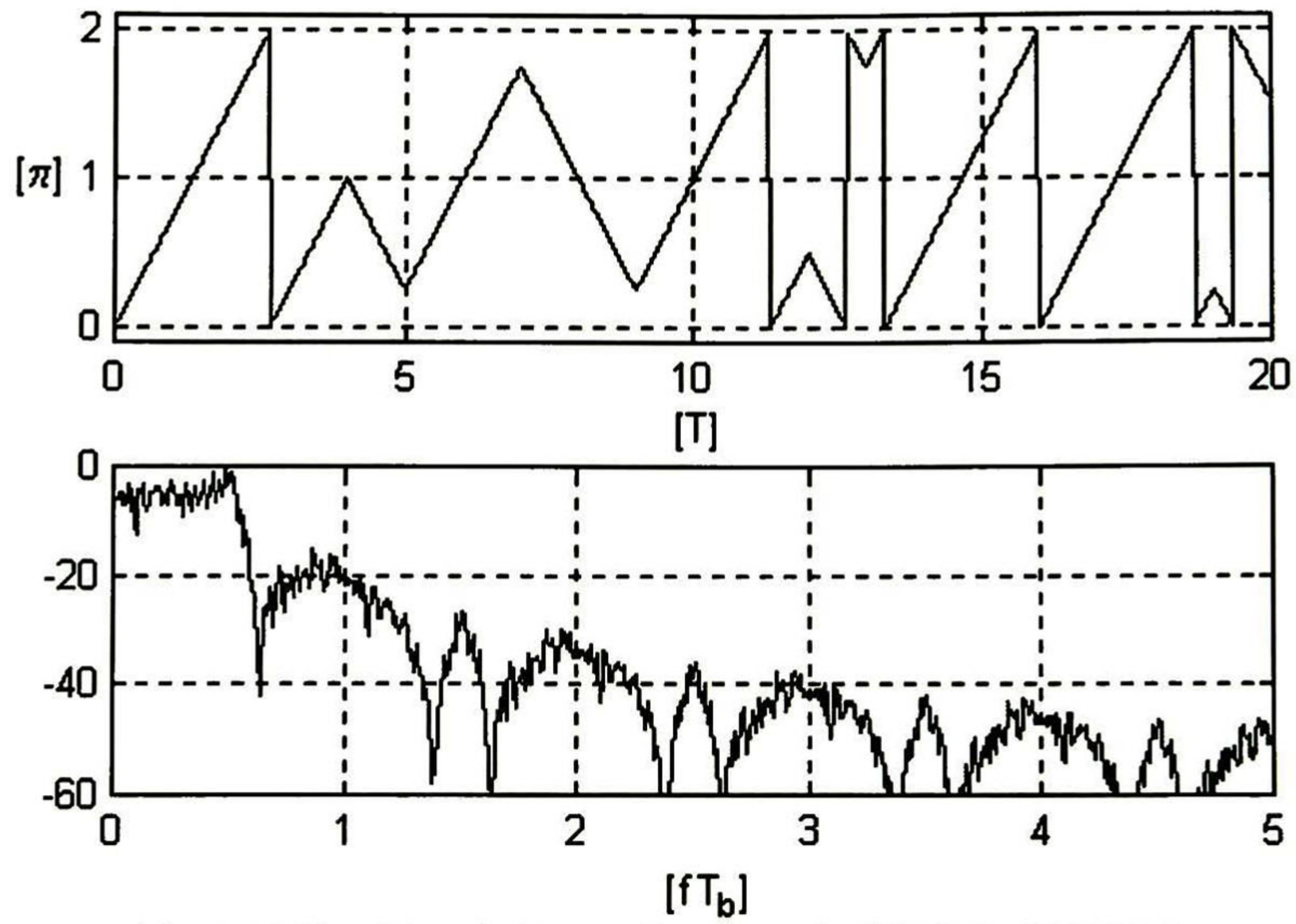


Fig. 3.13 Cambios de Fase y Espectro de 1REC, $h=0.75, M=2$.

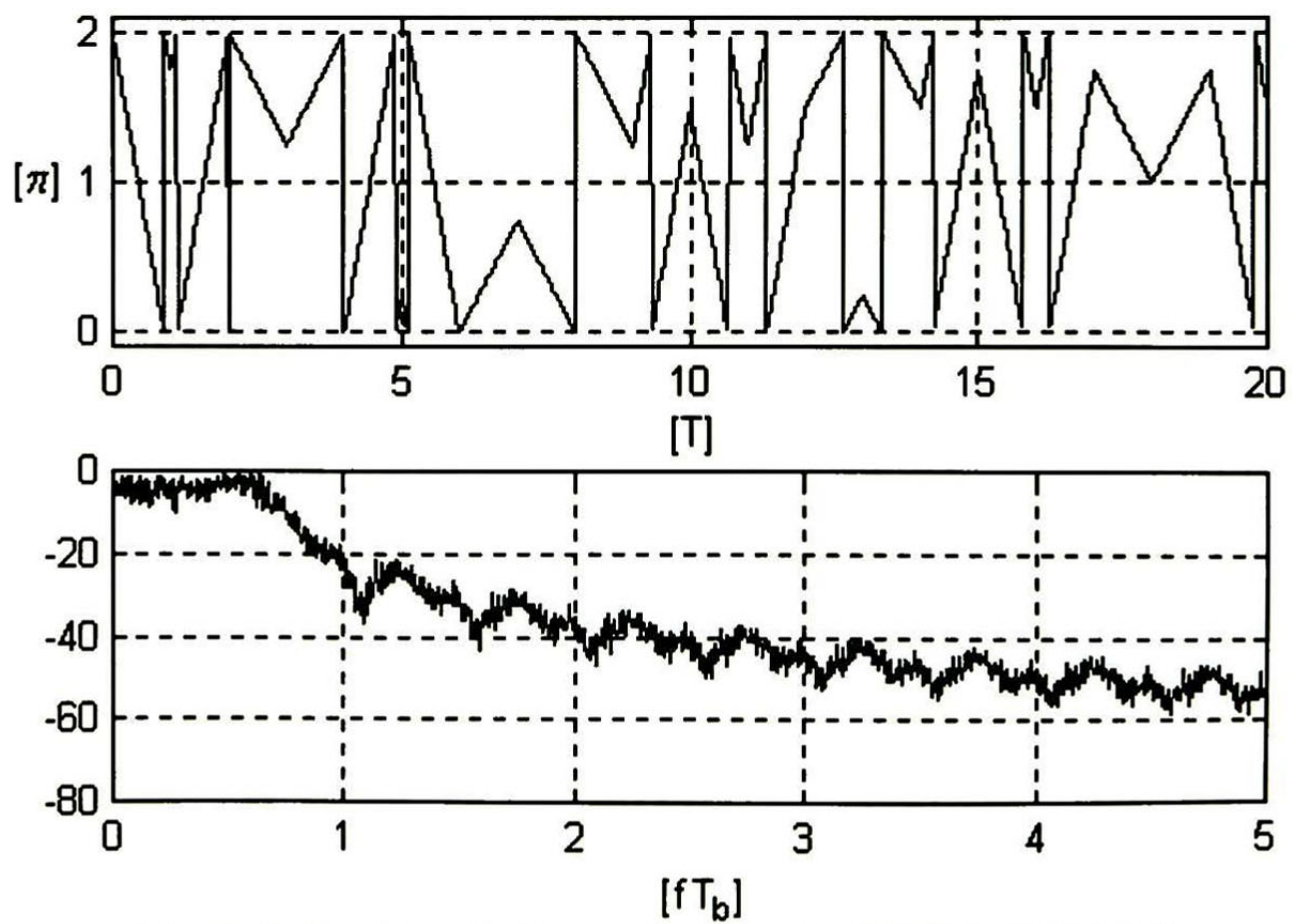


Fig. 3.14 Cambios de Fase y Espectro de 1REC, $h=0.75, M=4$.

3.6.2.2. Respuesta Parcial.

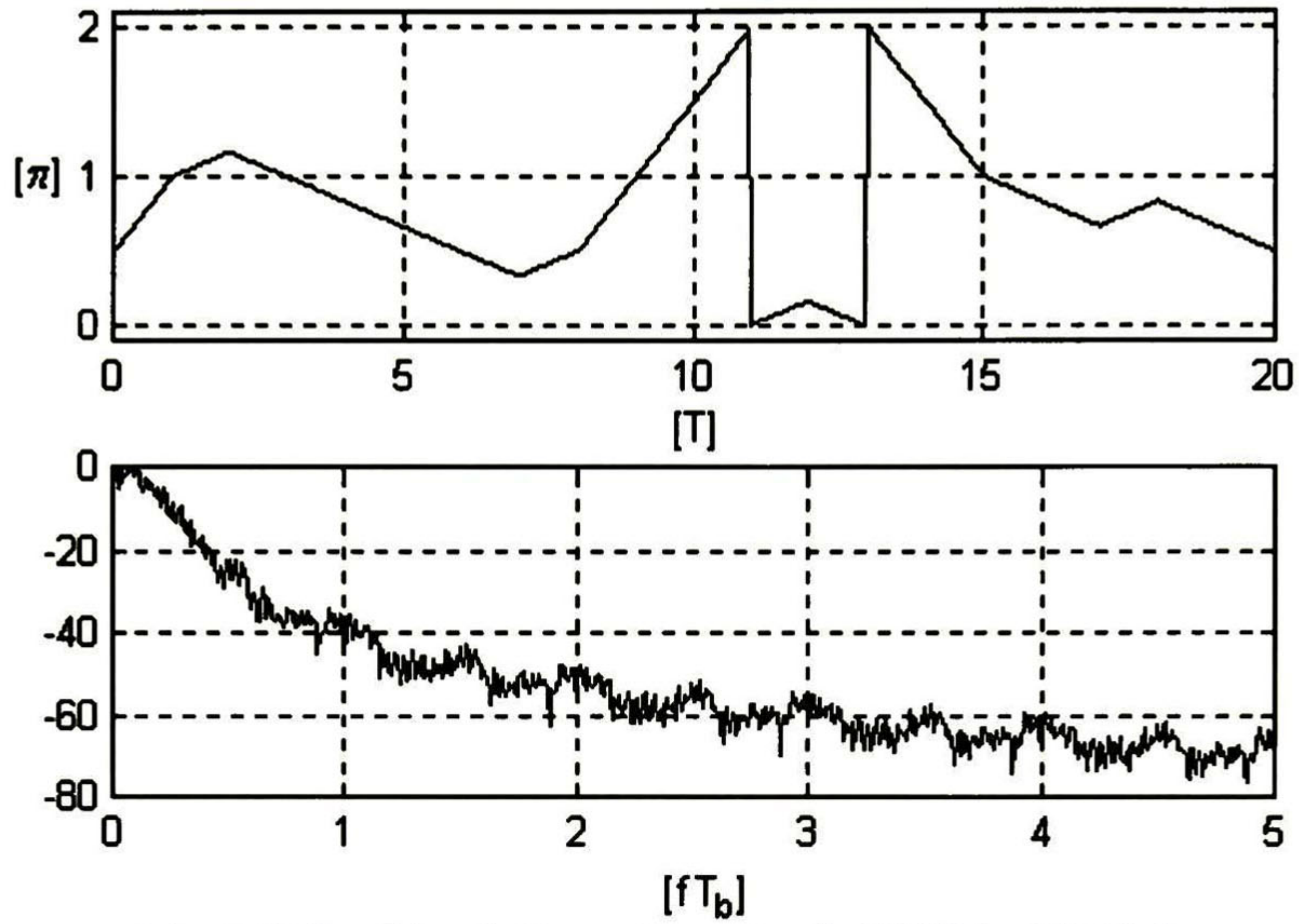


Fig. 3.15 Cambios de Fase y Espectro de 3REC, $h=0.5$, $M=2$.

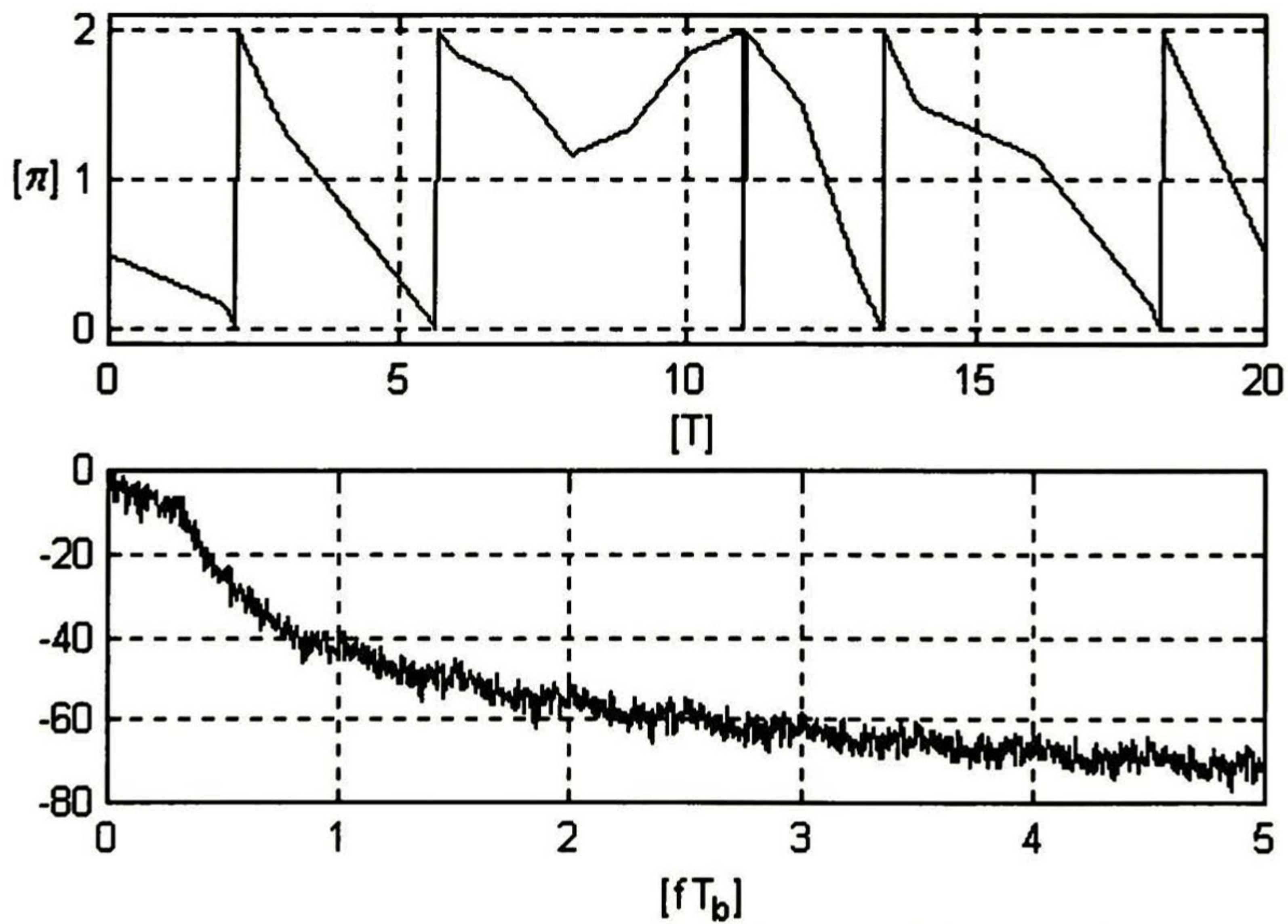


Fig. 3.16 Cambios de Fase y Espectro de 3REC, $h=0.5$, $M=4$.

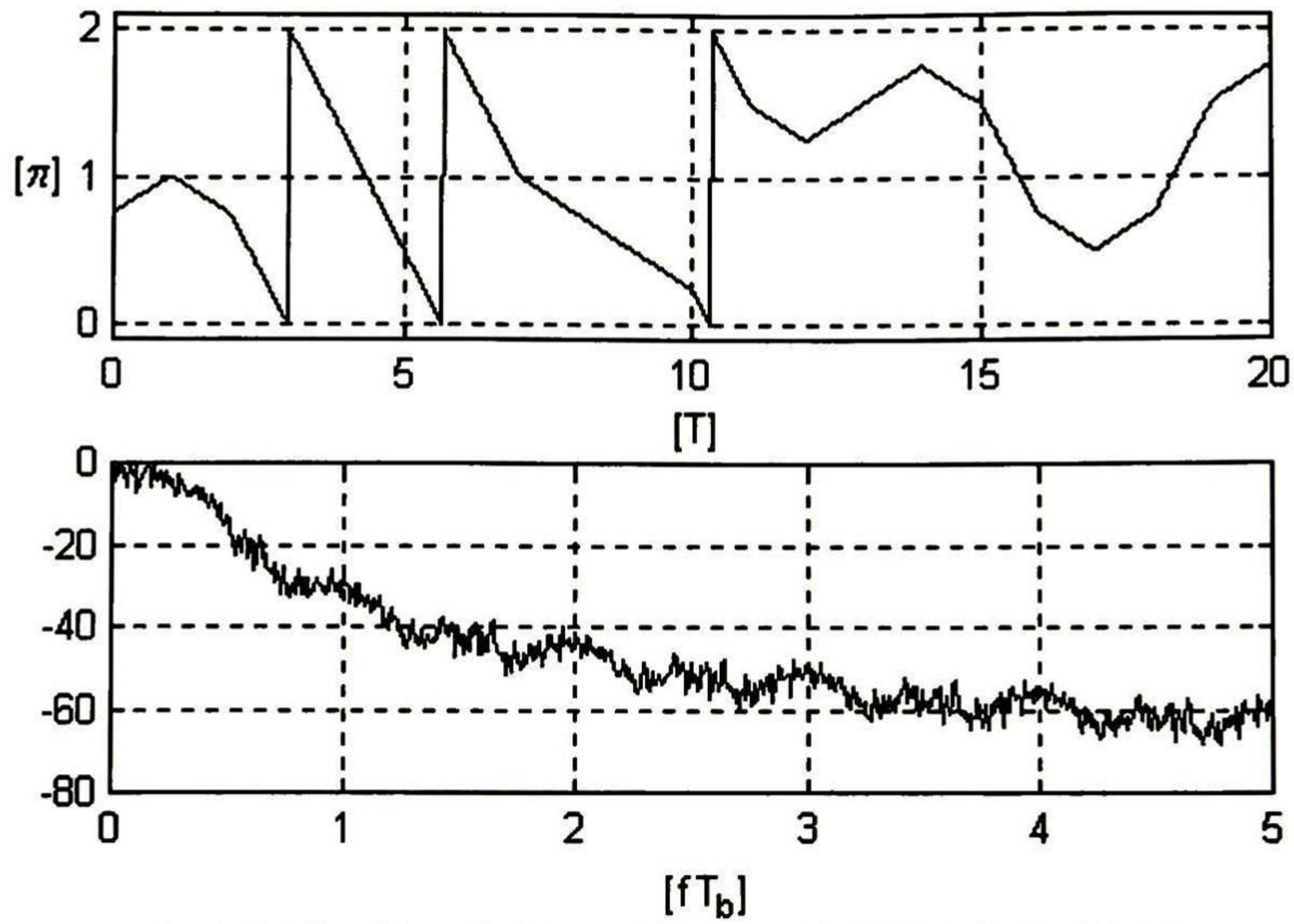


Fig. 3.17 Cambios de Fase y Espectro de 3REC, $h=0.75, M=2$.

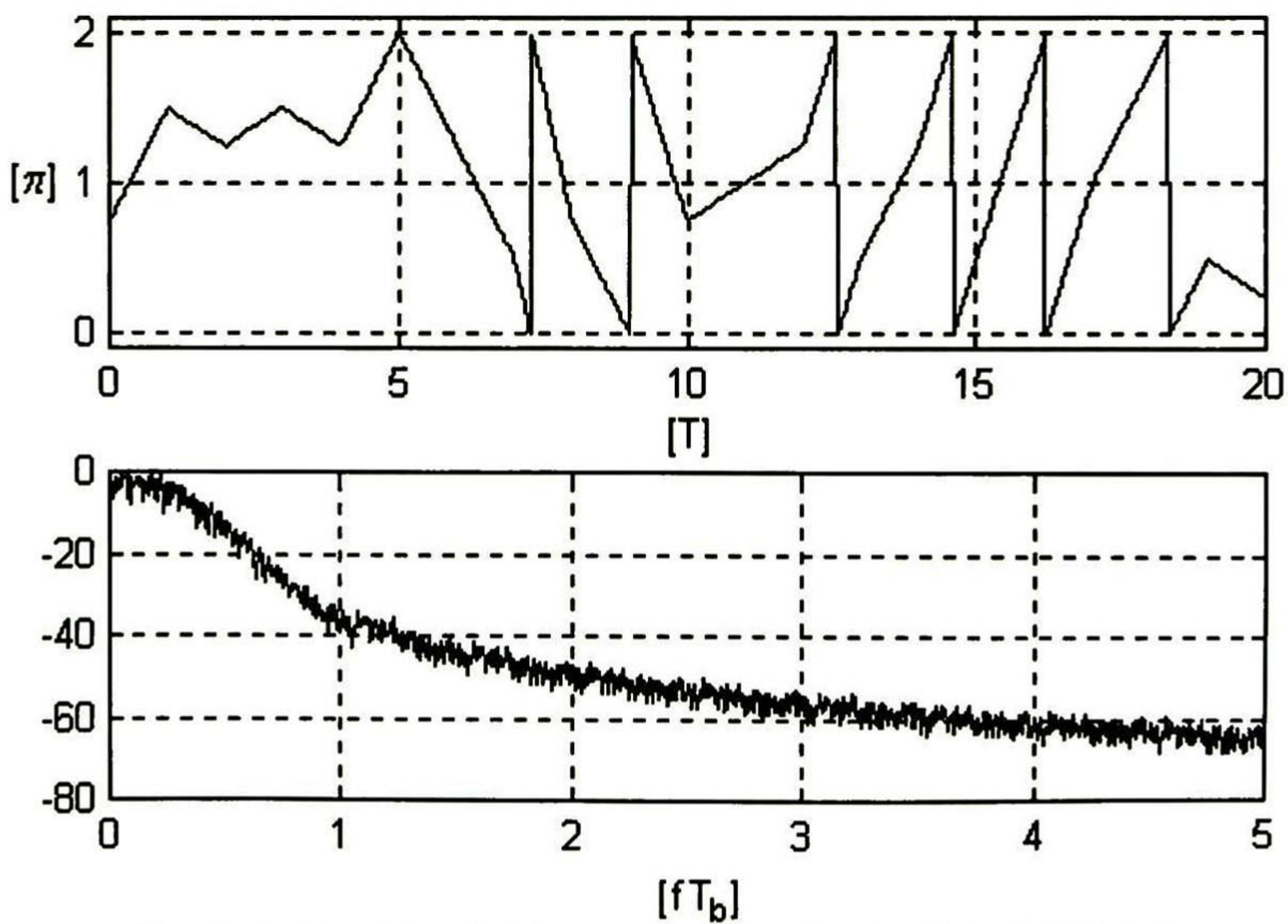


Fig. 3.18 Cambios de Fase y Espectro de 3REC, $h=0.75, M=4$.

3.6.3. Pulso LRC.

3.6.3.1. Respuesta Total.

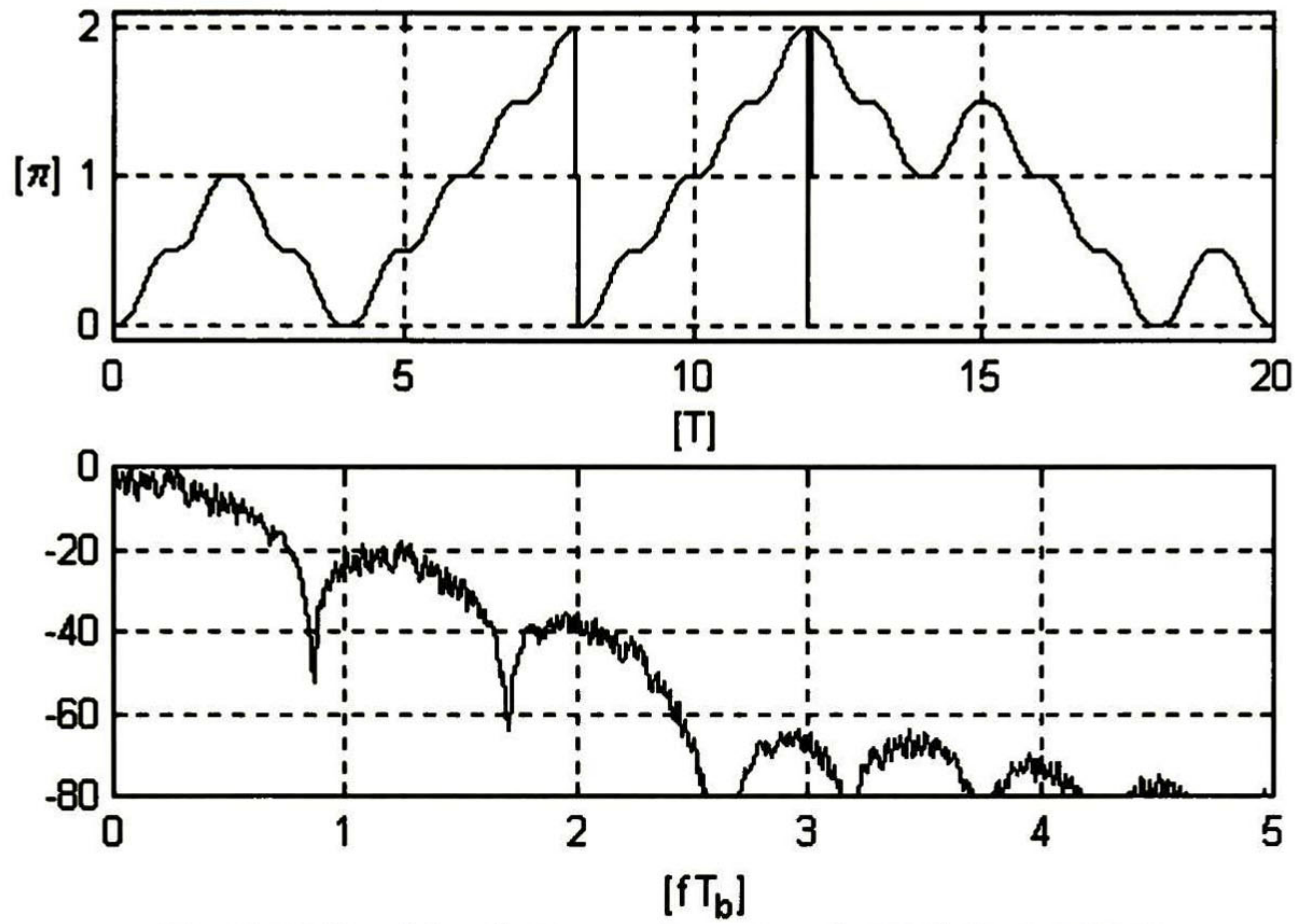


Fig. 3.19 Cambios de Fase y Espectro de 1RC, $h=0.5, M=2$.

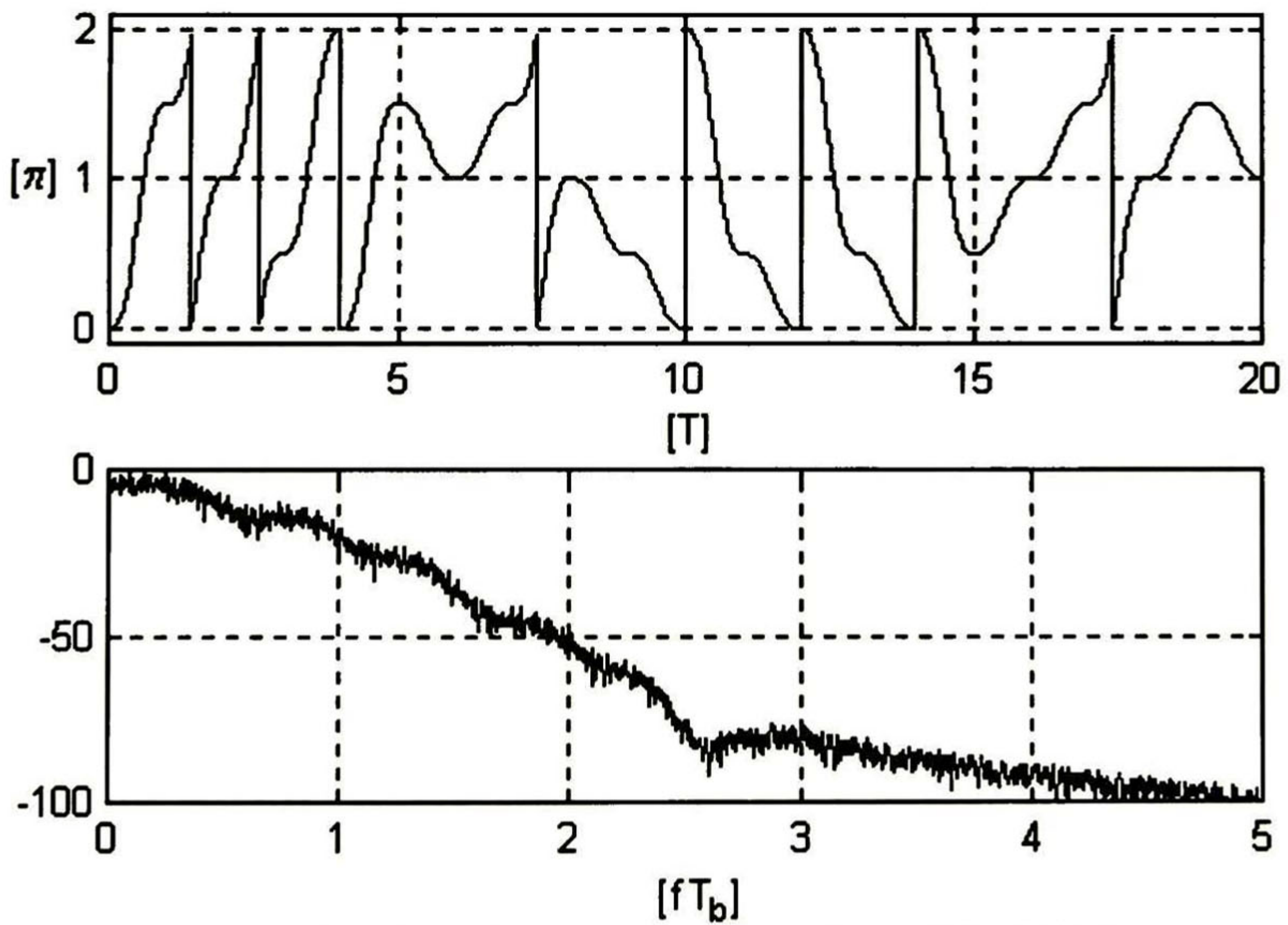


Fig. 3.20 Cambios de Fase y Espectro de 1RC, $h=0.5, M=4$.

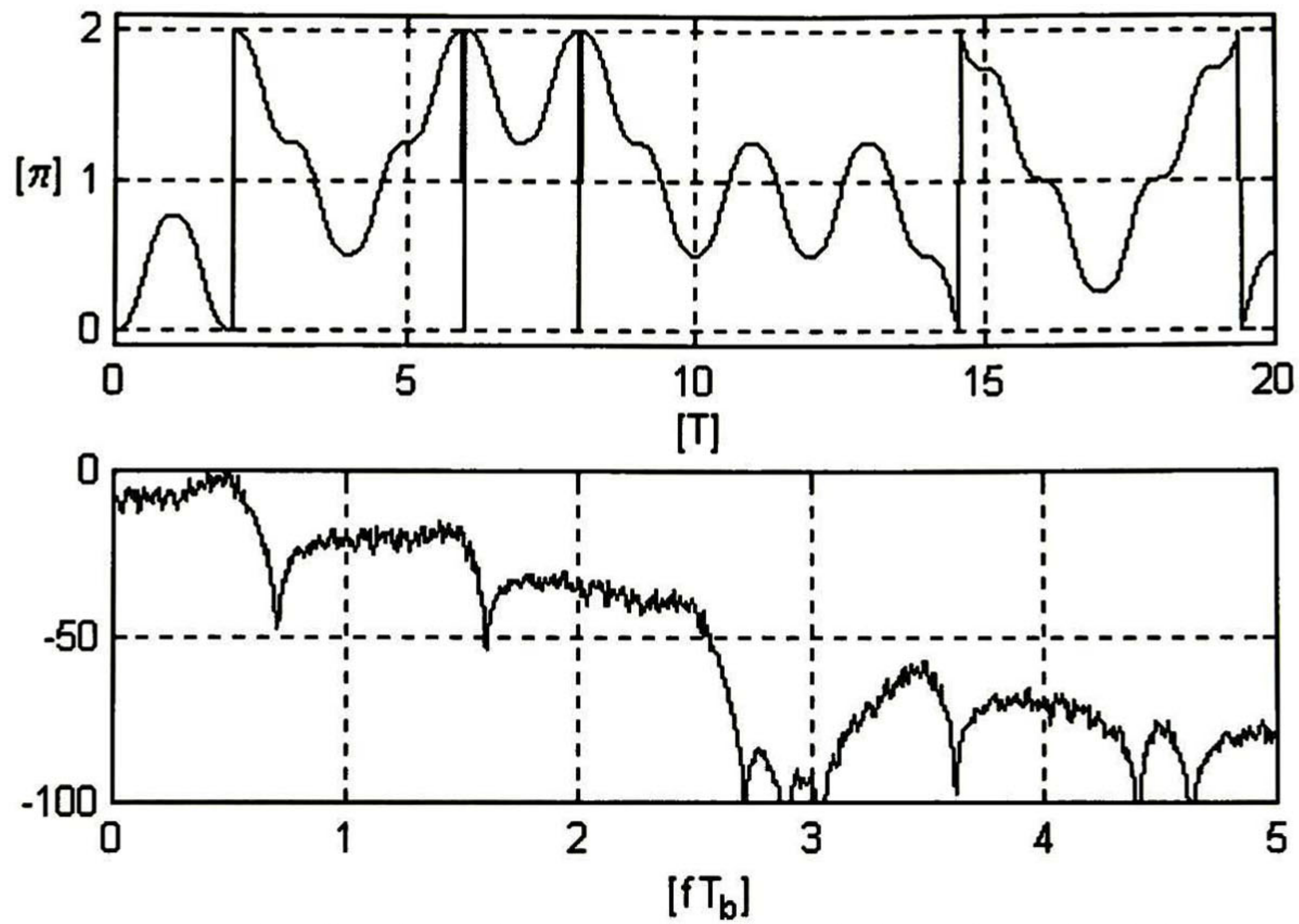


Fig. 3.21 Cambios de Fase y Espectro de 1RC, $h=0.75, M=2$.

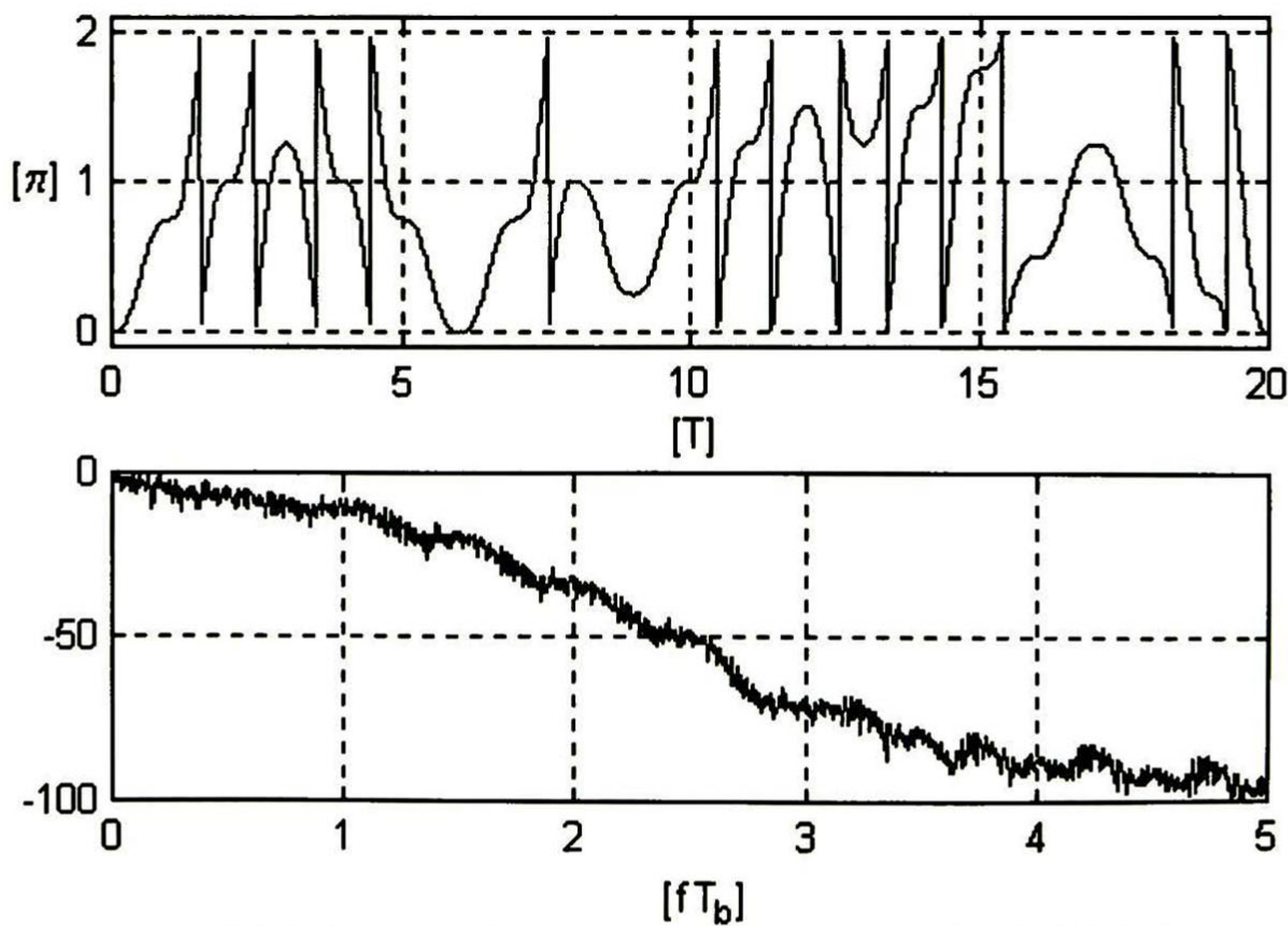


Fig. 3.22 Cambios de Fase y Espectro de 1RC, $h=0.75, M=4$.

3.6.3.2. Respuesta Parcial.

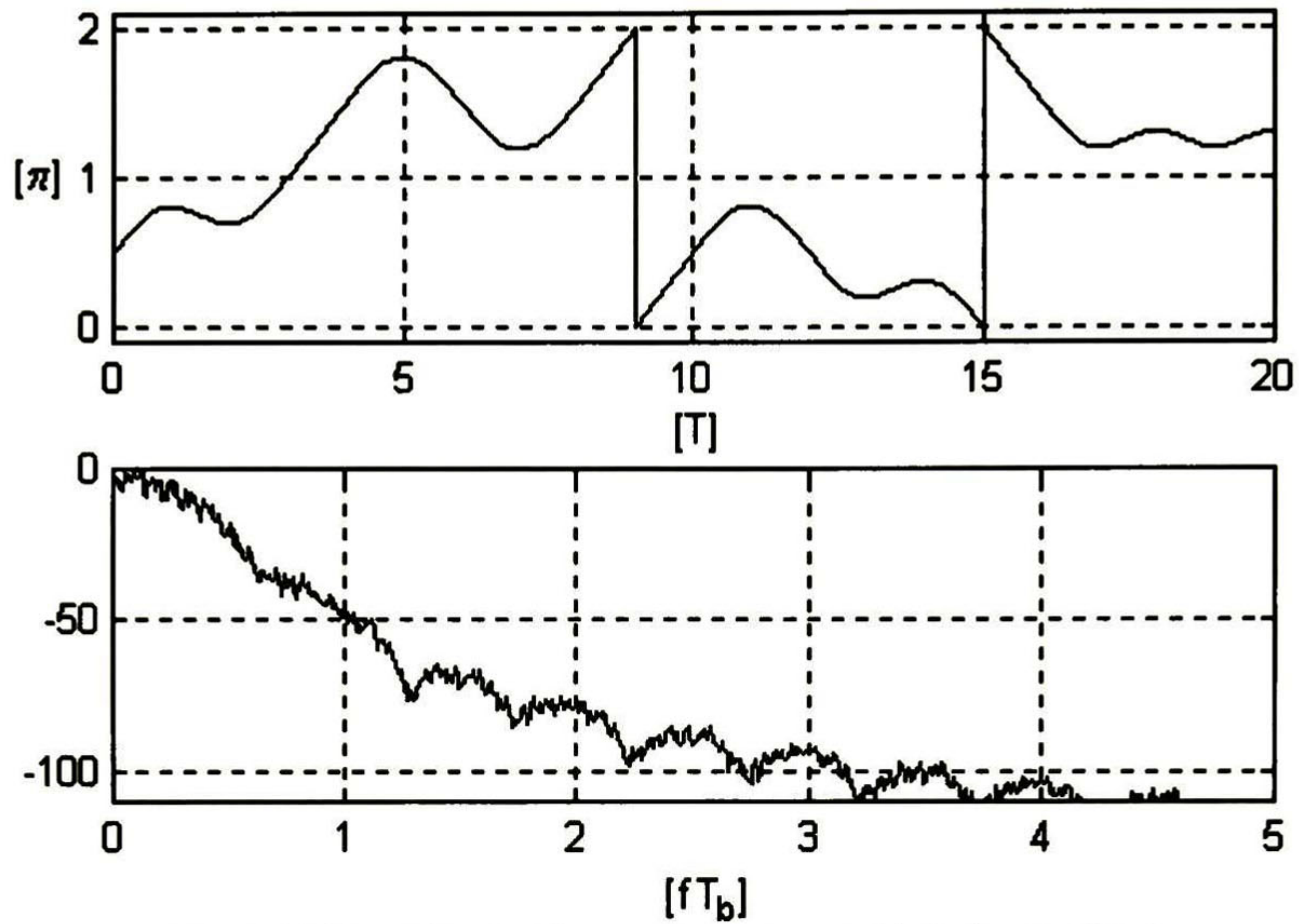


Fig. 3.23 Cambios de Fase y Espectro de 3RC, $h=0.5, M=2$.

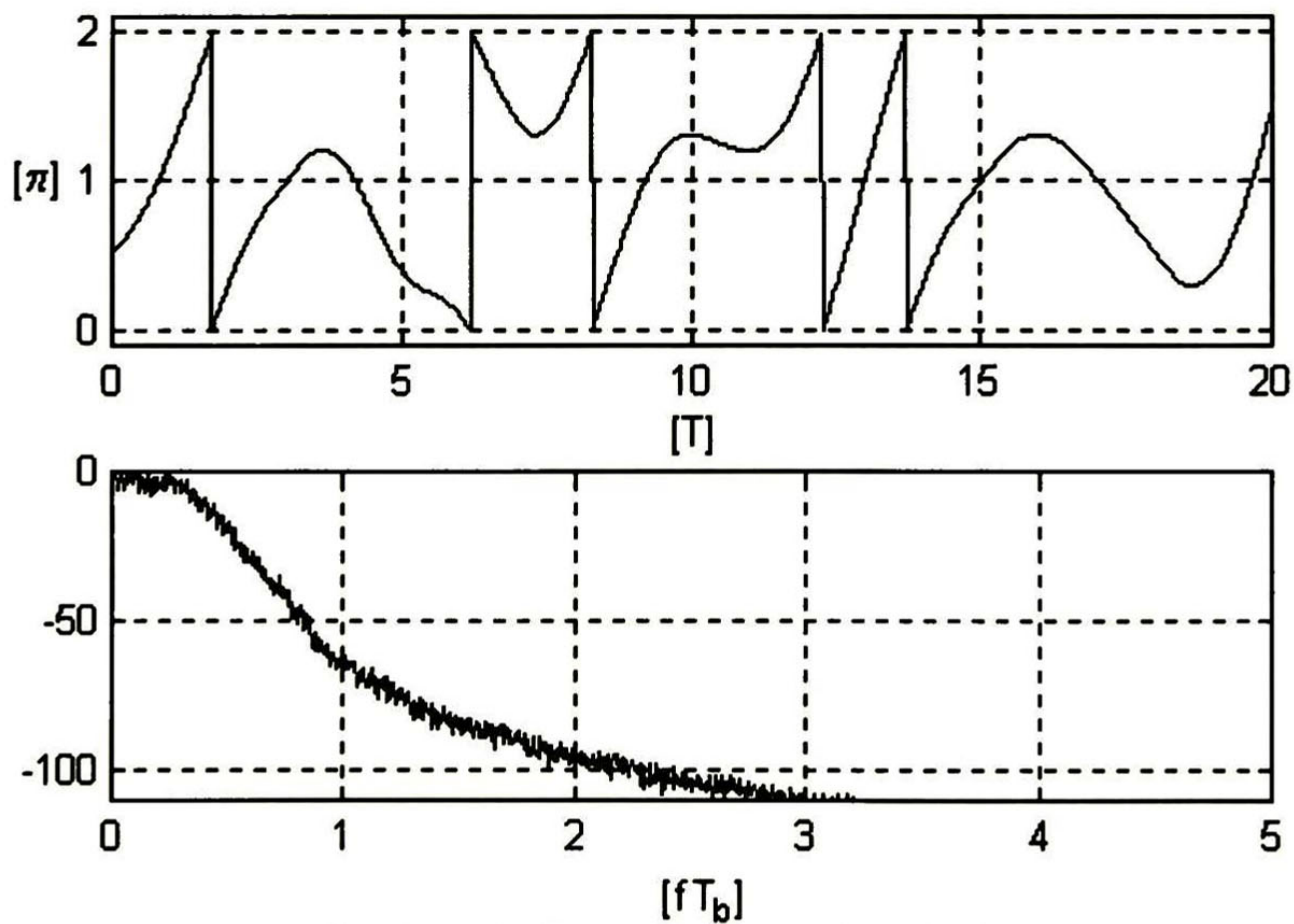


Fig. 3.24 Cambios de Fase y Espectro de 3RC, $h=0.5, M=4$.

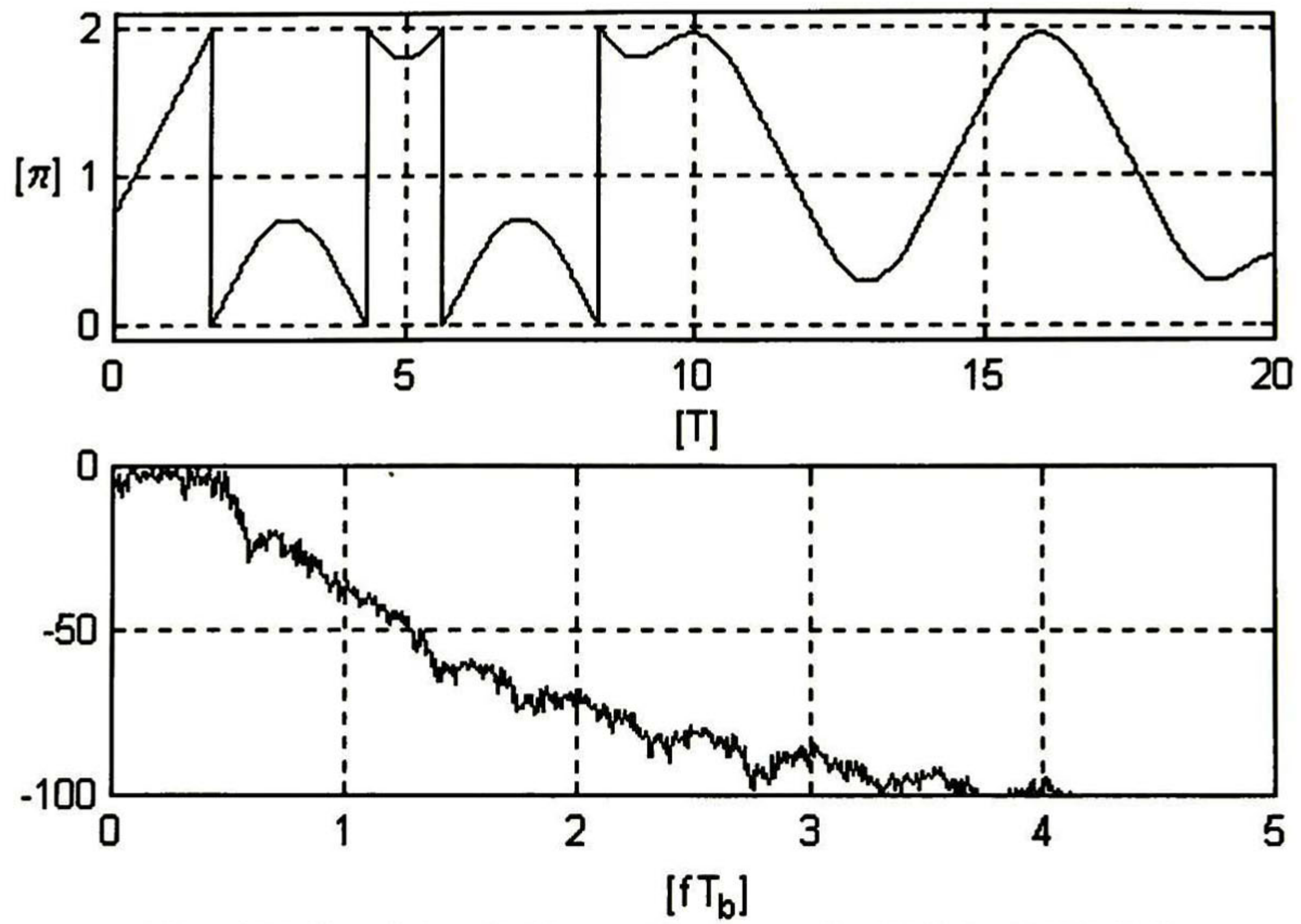


Fig. 3.25 Cambios de Fase y Espectro de 3RC, $h=0.75, M=2$.

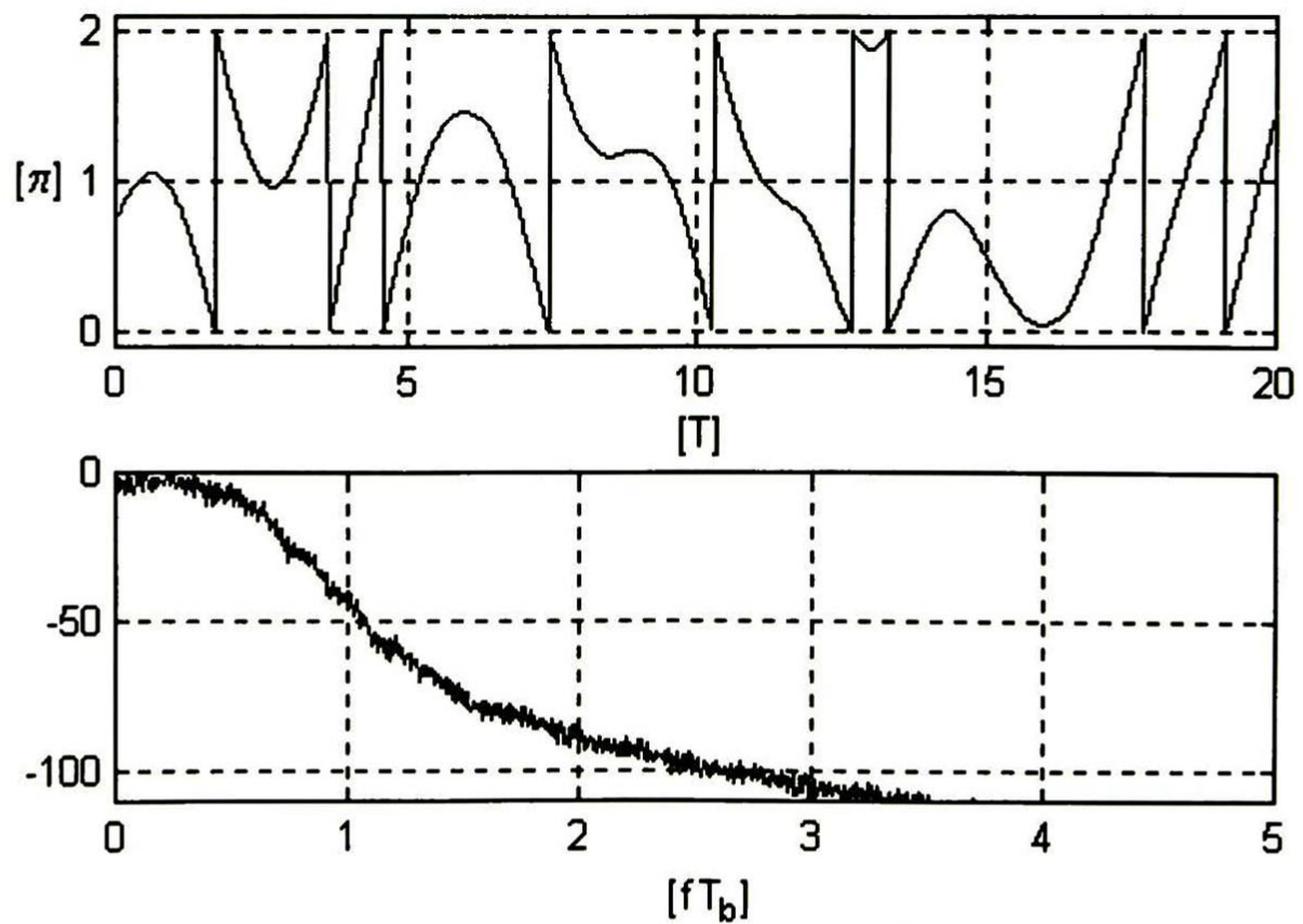


Fig. 3.26 Cambios de Fase y Espectro de 3RC, $h=0.75, M=4$.

3.6.4. Pulso LSRC.

3.6.4.1. Respuesta Total.

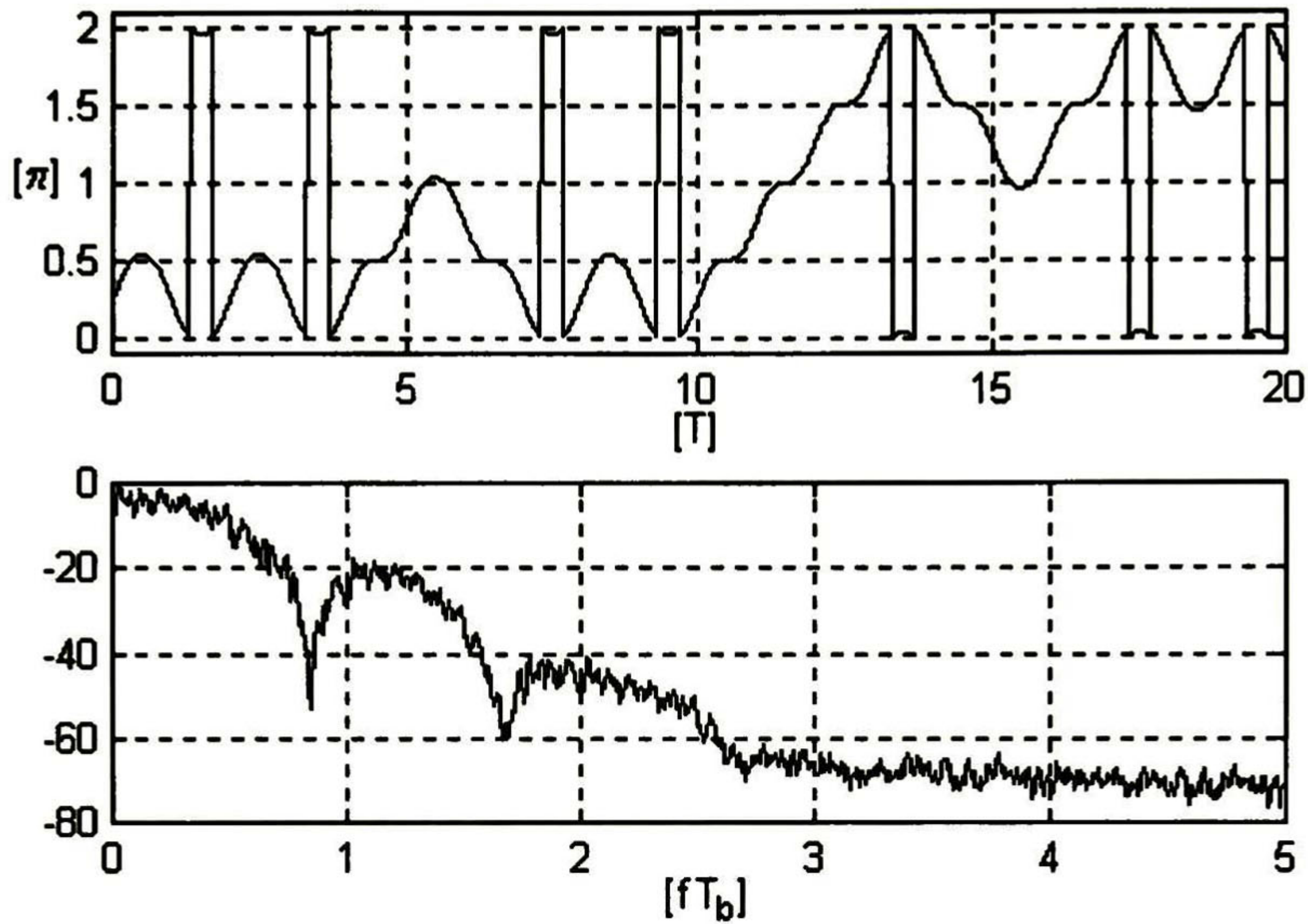


Fig. 3.27 Cambios de Fase y Espectro de 1SRC, $h=0.5$, $M=2$, $\beta=0.75$.

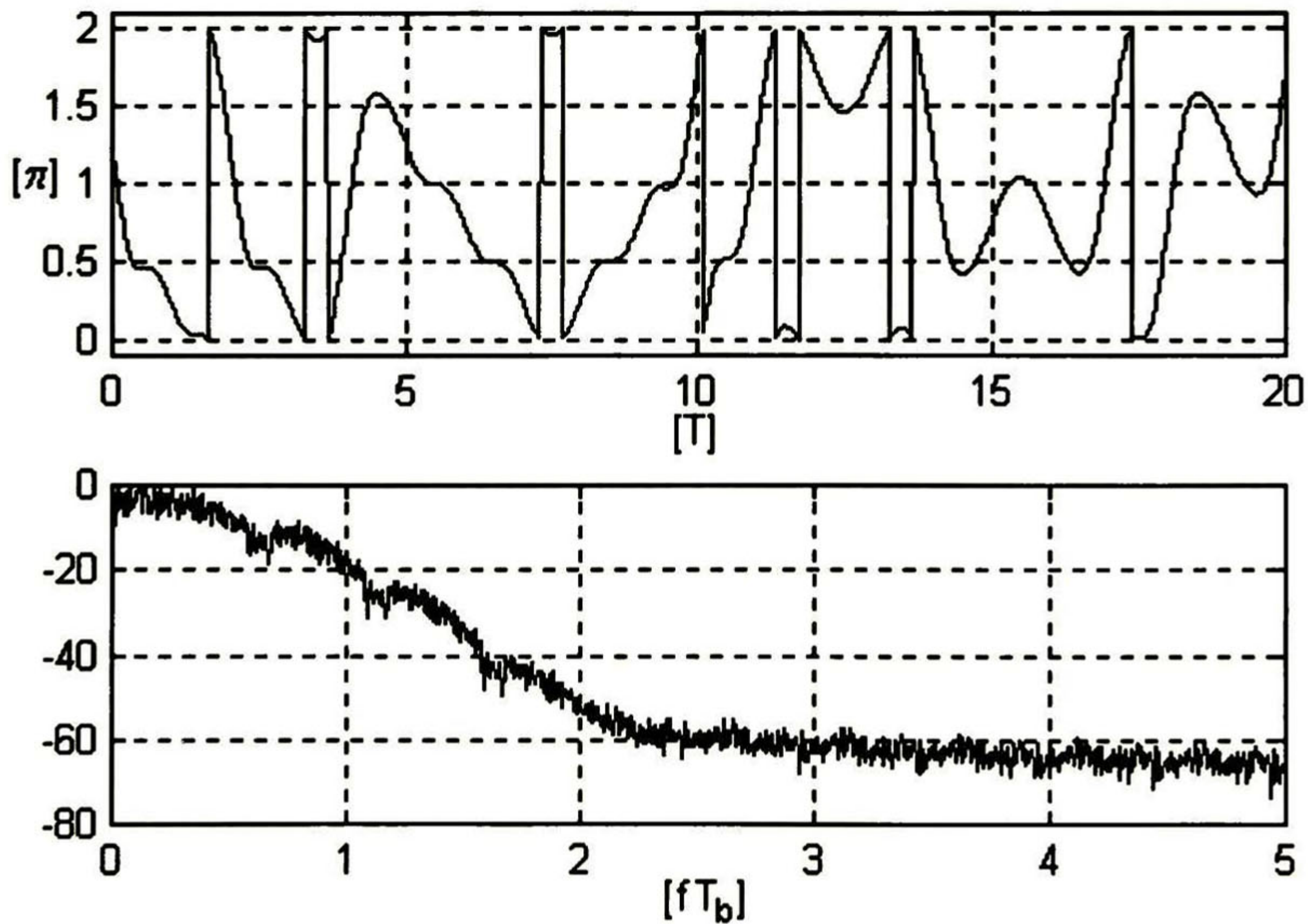


Fig. 3.28 Cambios de Fase y Espectro de 1SRC, $h=0.5$, $M=4$, $\beta=0.75$.

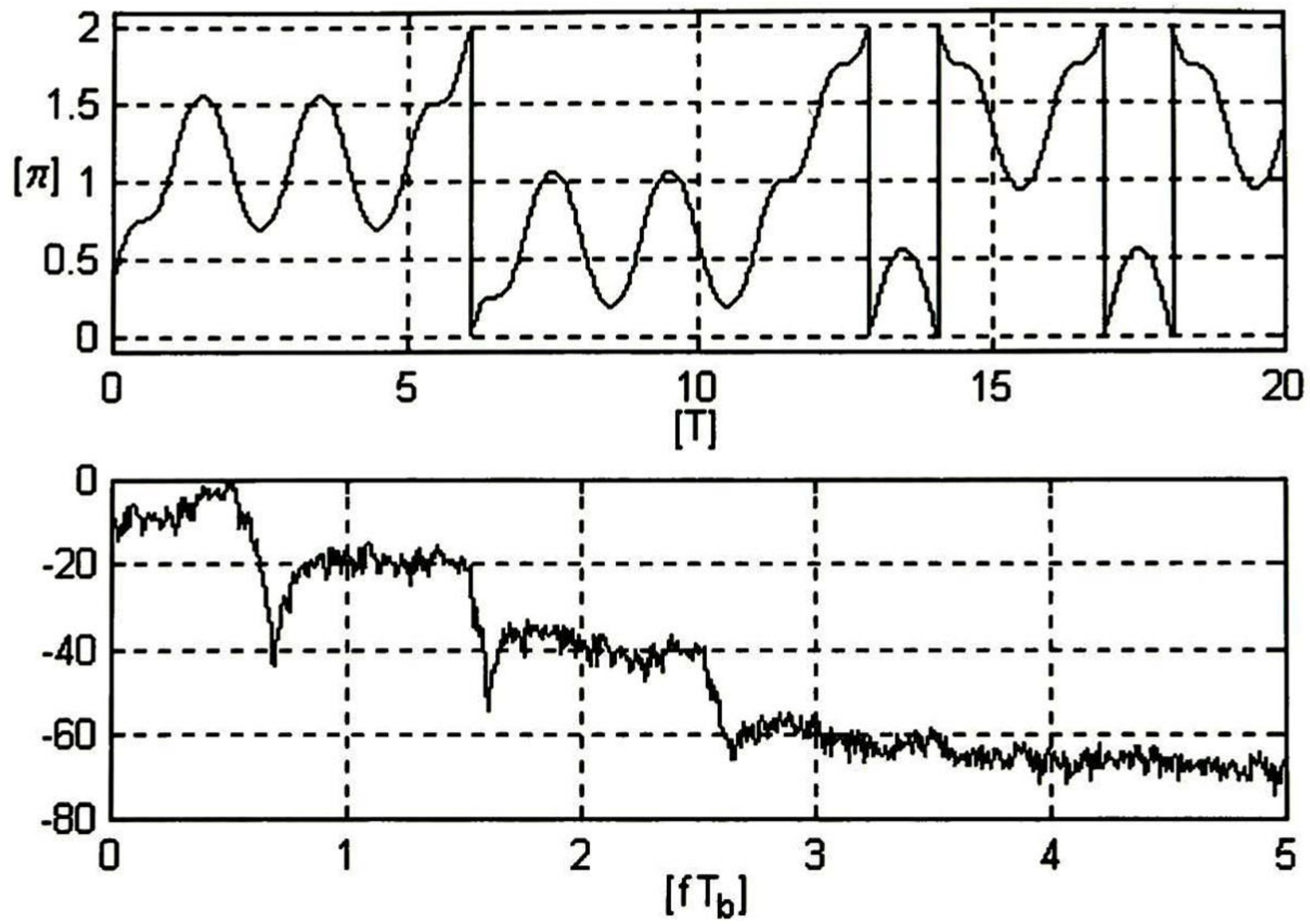


Fig. 3.29 Cambios de Fase y Espectro de 1SRC, $h=0.75$, $M=2$, $\beta=0.75$.

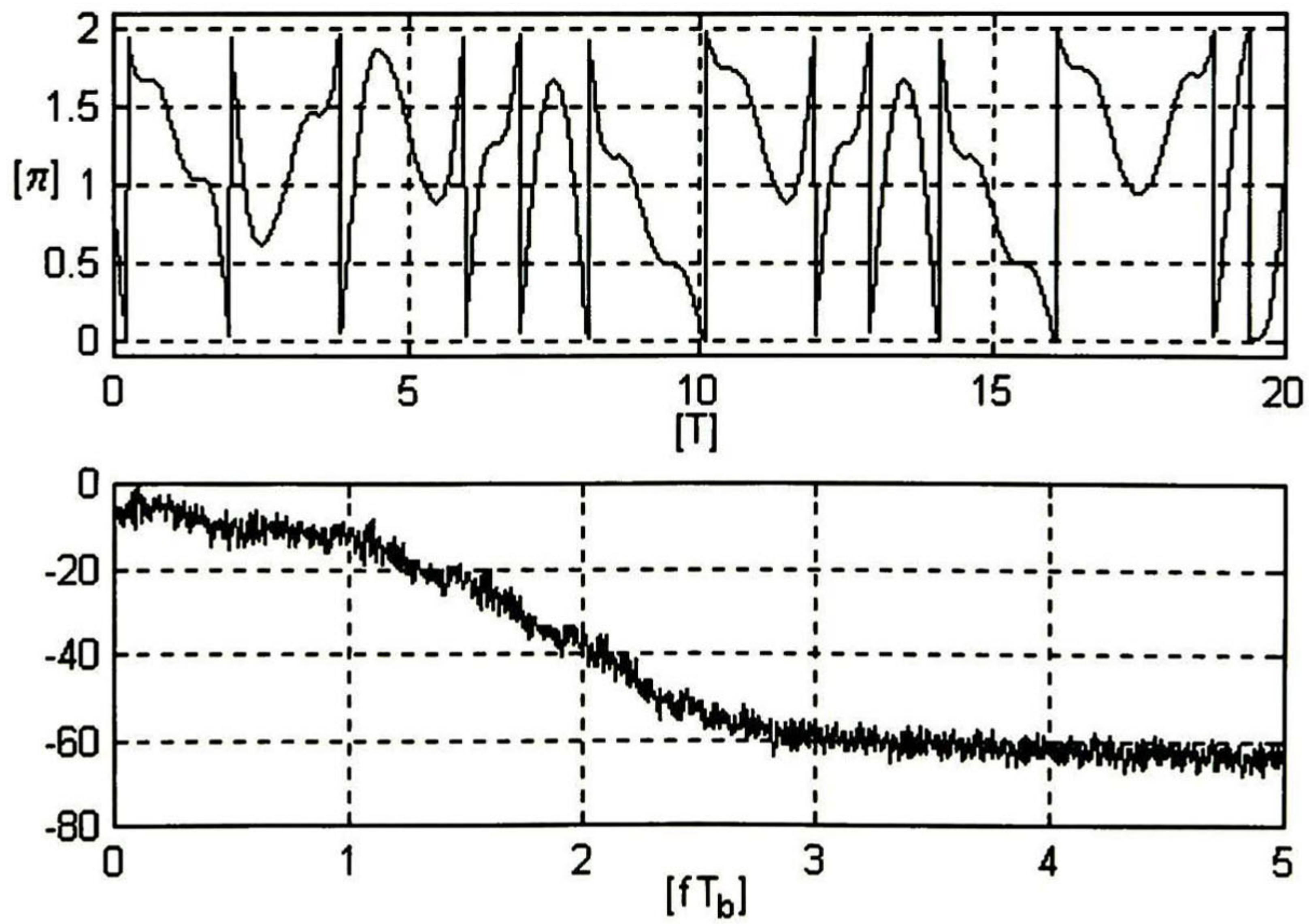


Fig. 3.30 Cambios de Fase y Espectro de 1SRC, $h=0.75$, $M=4$, $\beta=0.75$.

3.6.4.2. Respuesta Parcial.

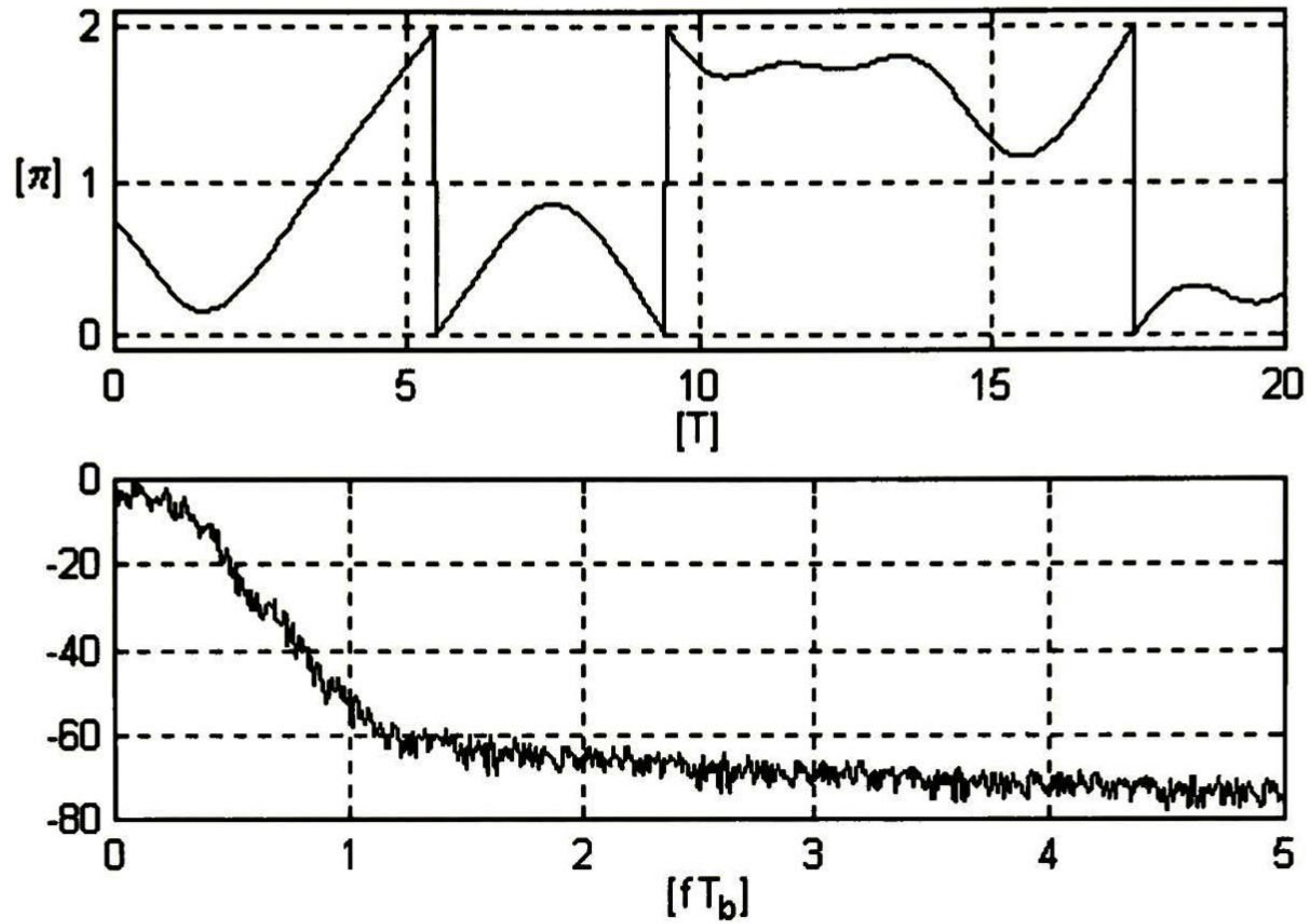


Fig. 3.31 Cambios de Fase y Espectro de 3SRC, $h=0.5$, $M=2$, $\beta=0.75$.

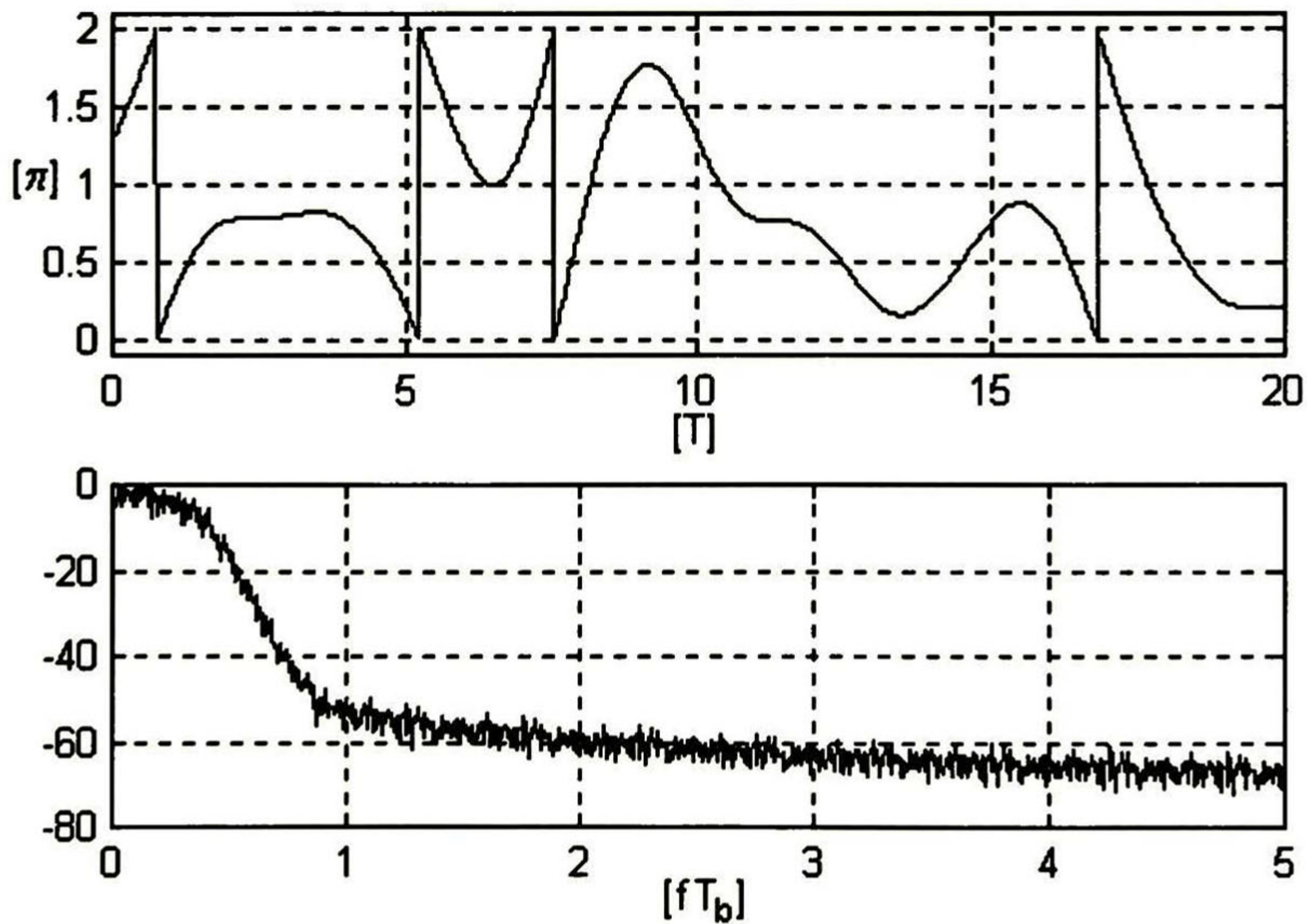


Fig. 3.32 Cambios de Fase y Espectro de 3SRC, $h=0.5$, $M=4$, $\beta=0.75$.

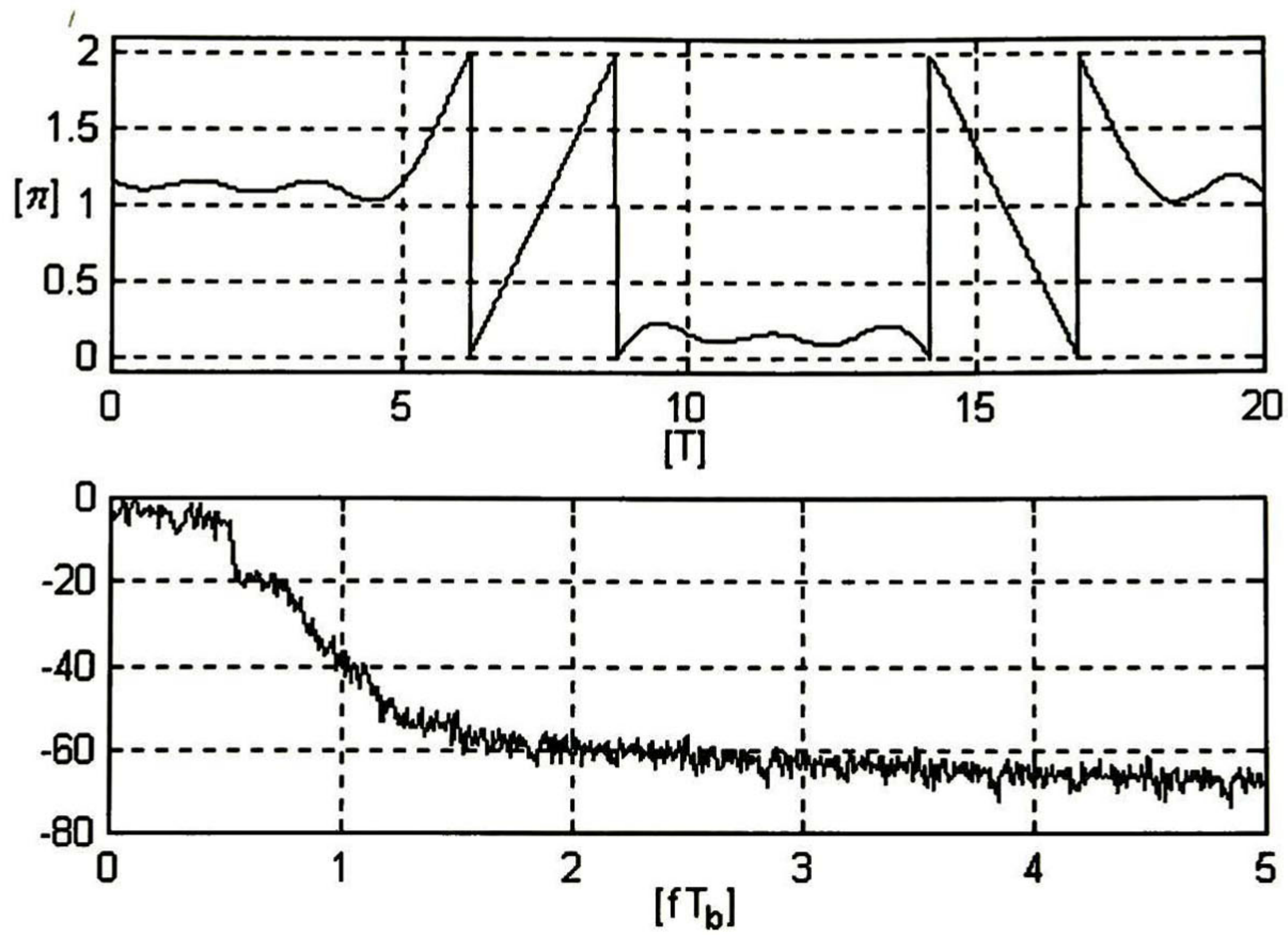


Fig. 3.33 Cambios de Fase y Espectro de 3SRC, $h=0.75$, $M=2$, $\beta=0.75$.

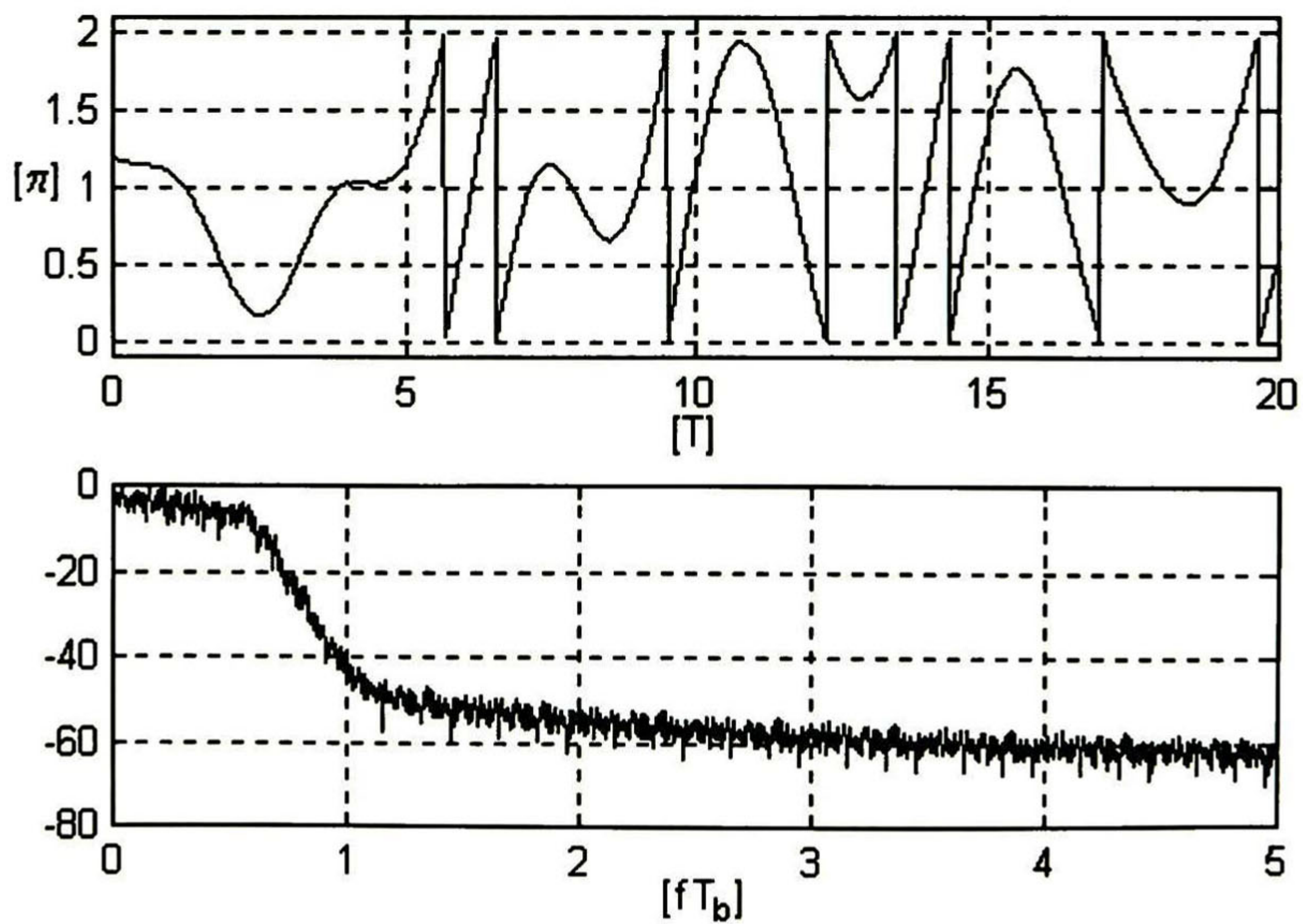


Fig. 3.34 Cambios de Fase y Espectro de 3SRC, $h=0.75$, $M=4$, $\beta=0.75$.

3.6.5. Pulso GMSK.

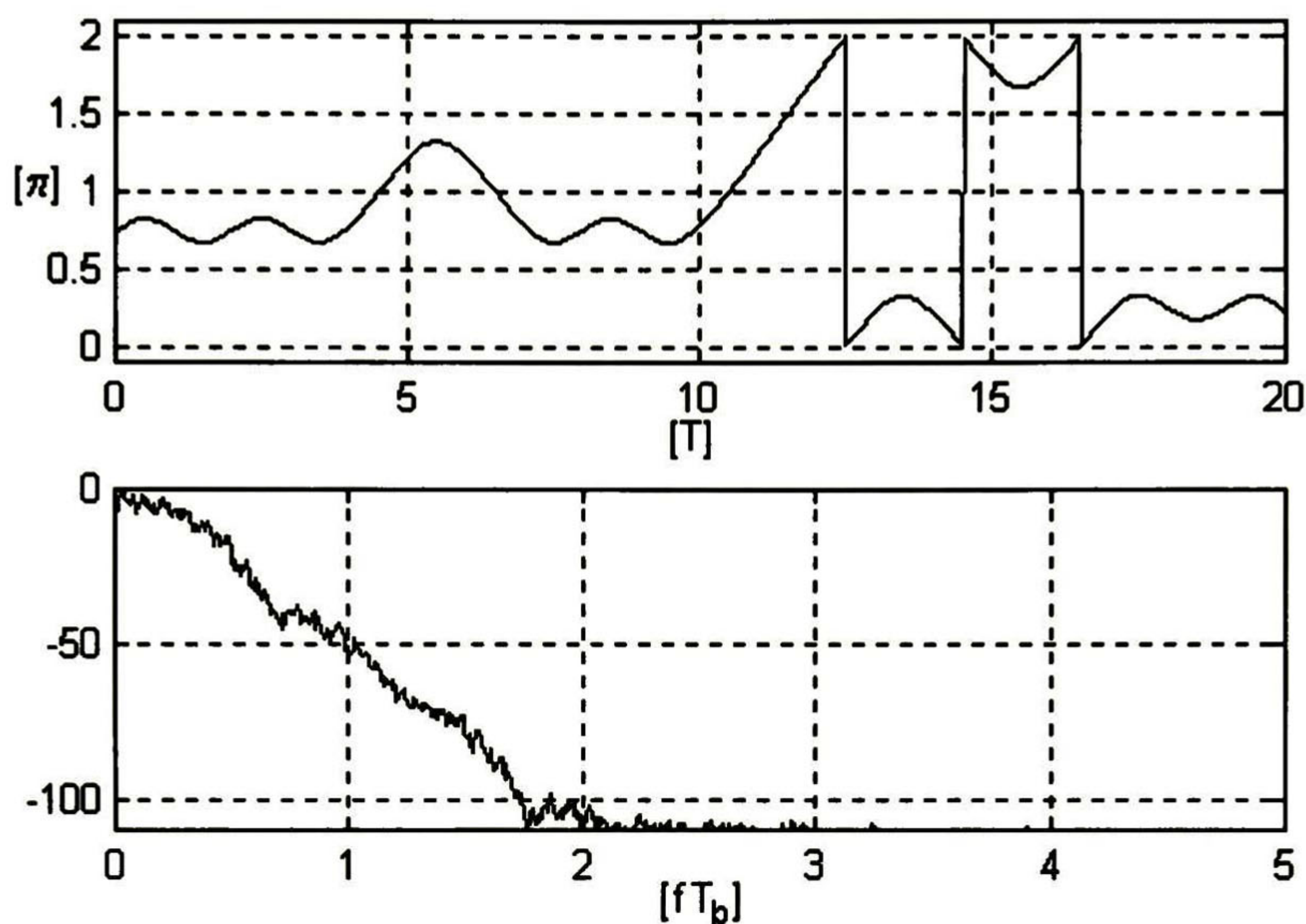


Fig. 3.35 Cambios de Fase y Espectro de GMSK, $h=0.5$, $M=2$, $BT=0.3$.

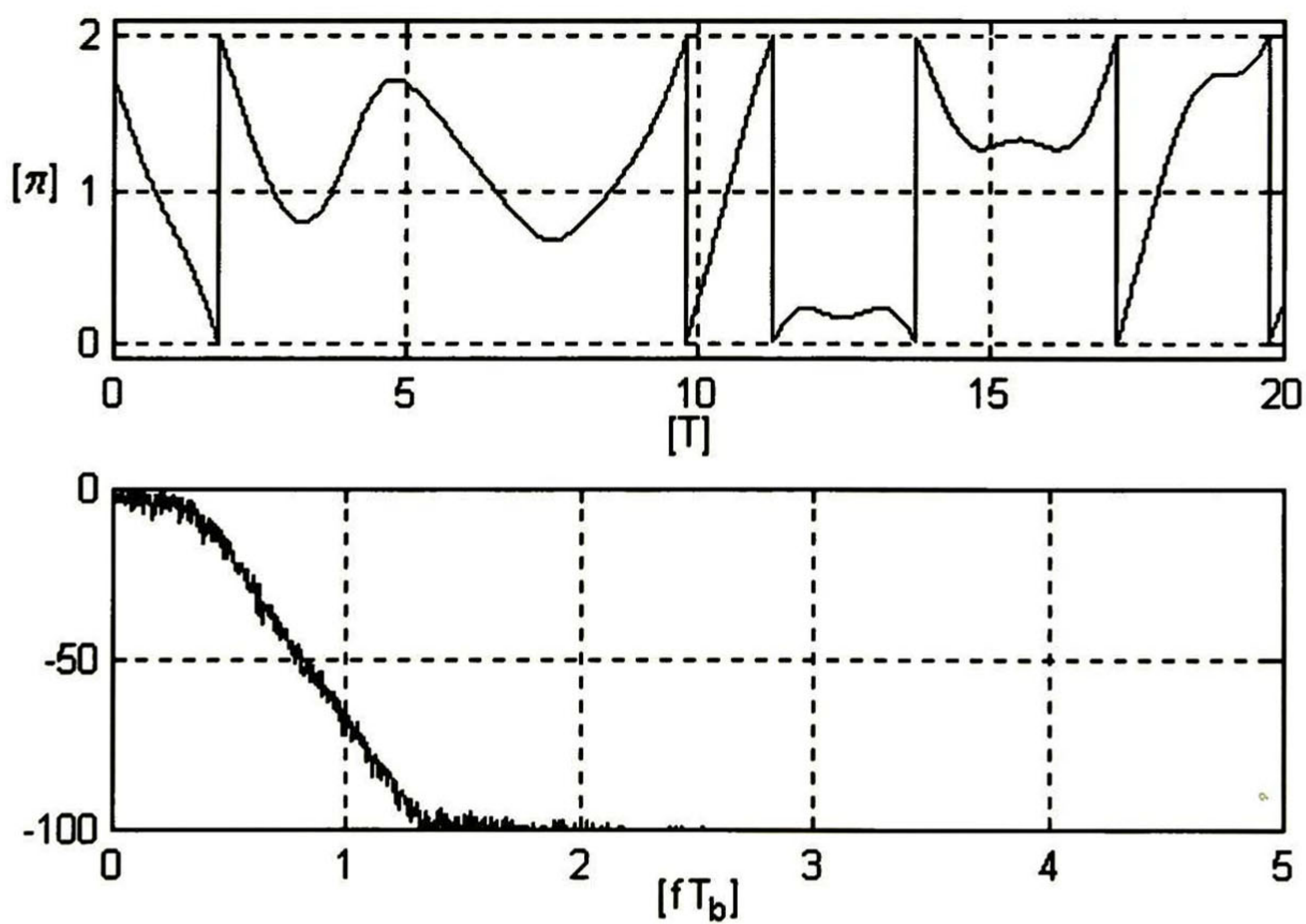


Fig. 3.36 Cambios de Fase y Espectro de GMSK, $h=0.5$, $M=4$, $BT=0.3$.

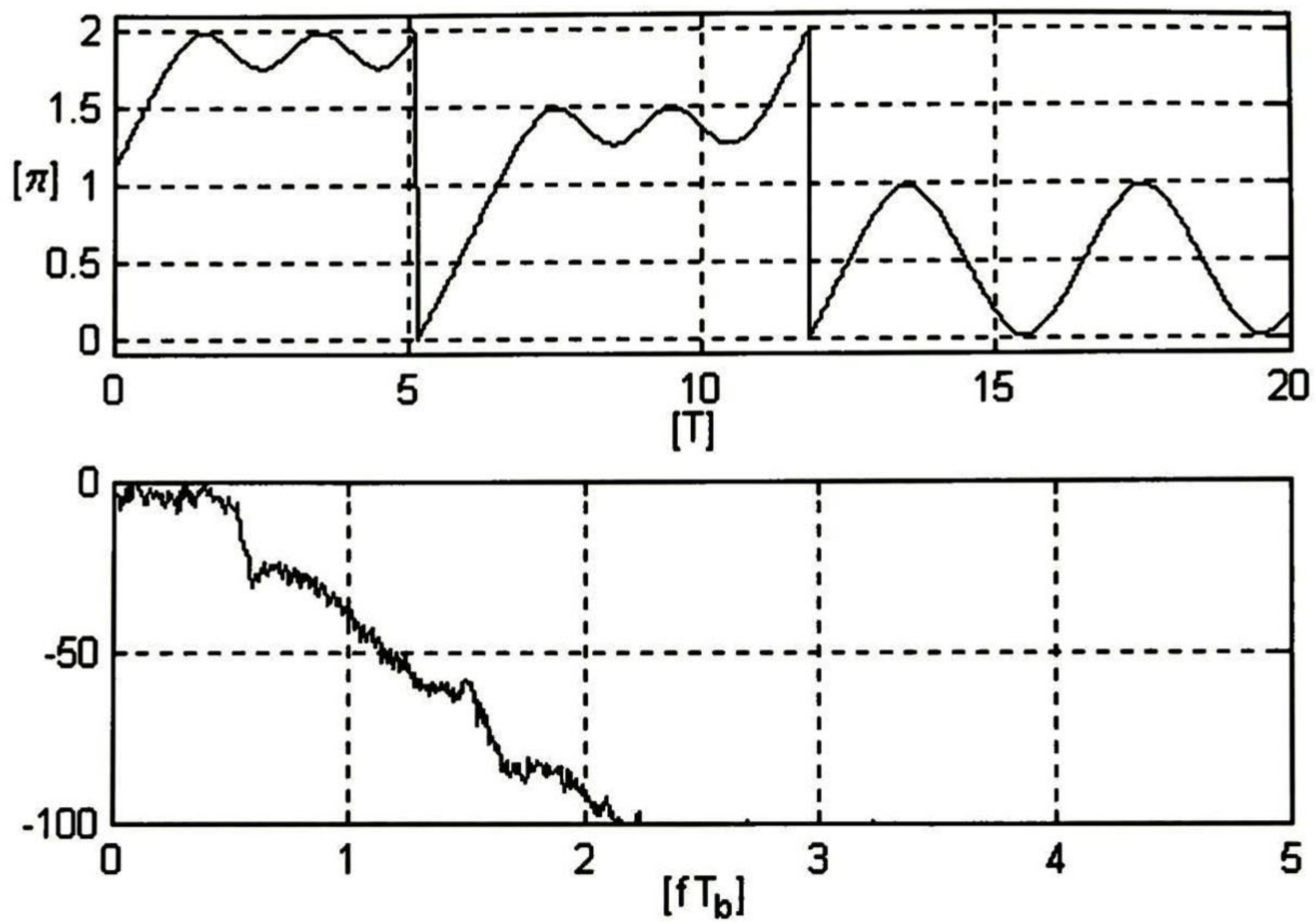


Fig. 3.37 Cambios de Fase y Espectro de GMSK, $h=0.75$, $M=2$, $BT=0.3$.

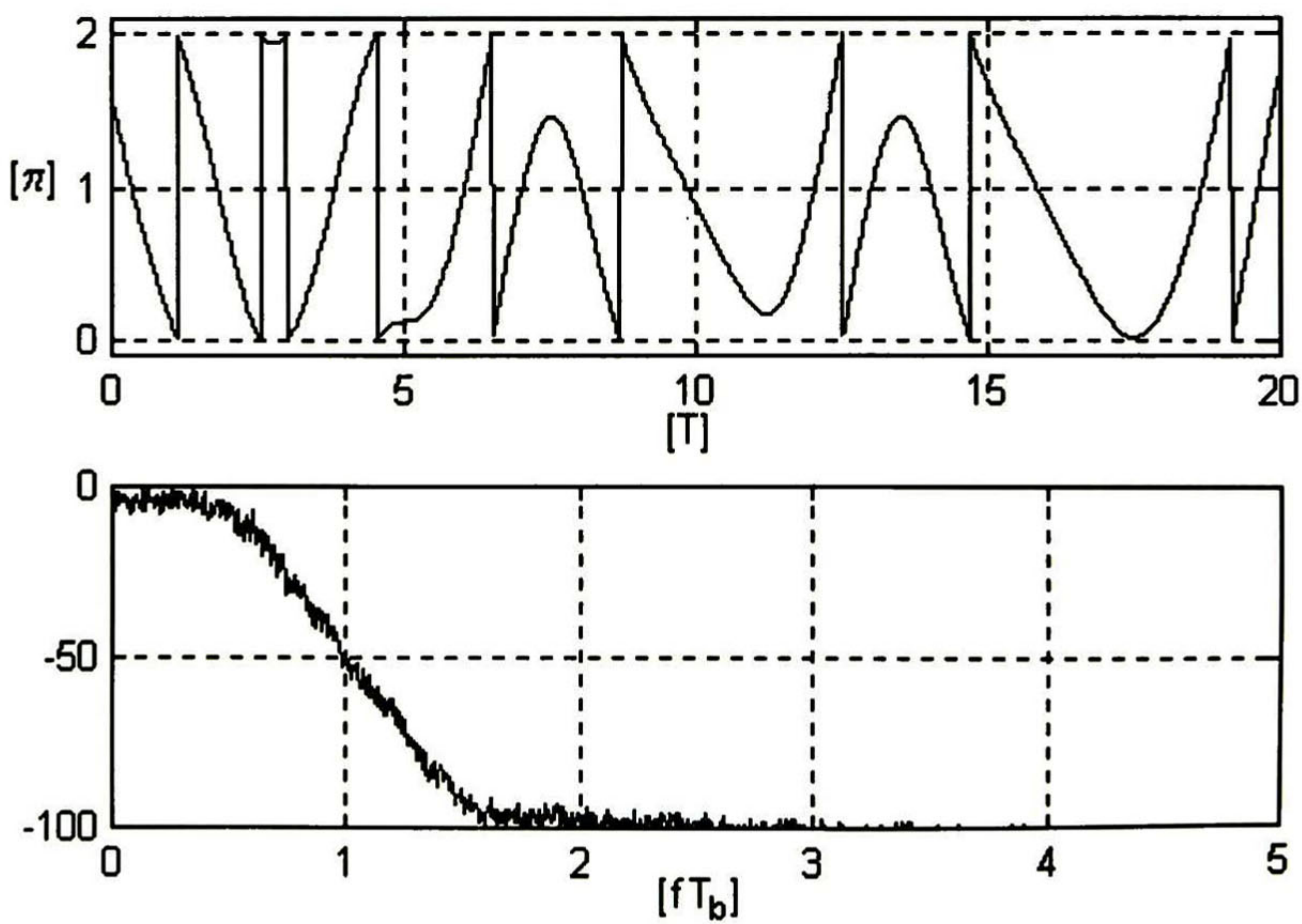


Fig. 3.38 Cambios de Fase y Espectro de GMSK, $h=0.75$, $M=4$, $BT=0.3$.

3.6.6. Pulso TFM.

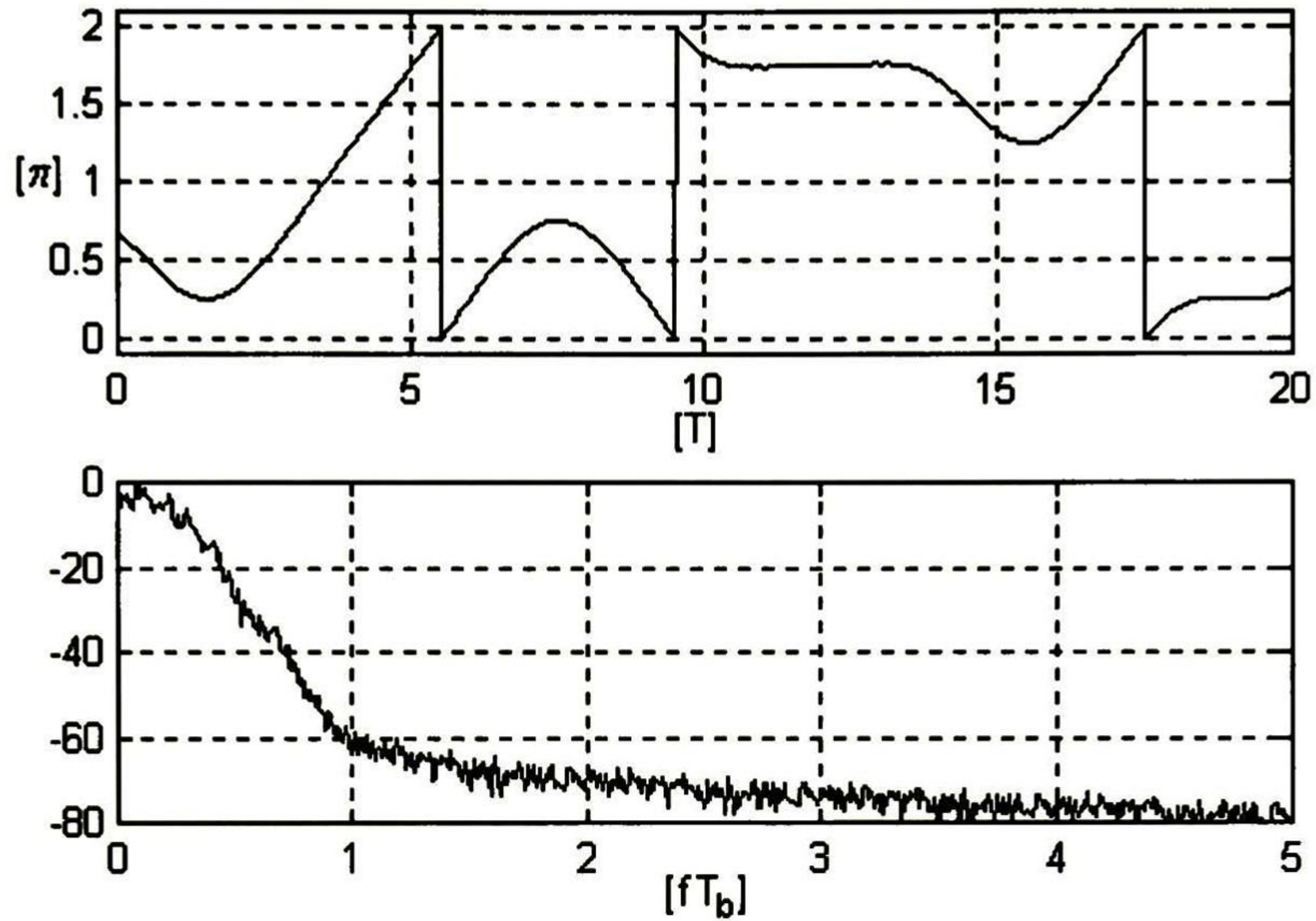


Fig. 3.39 Cambios de Fase y Espectro de TFM, $h=0.5$, $M=2$.

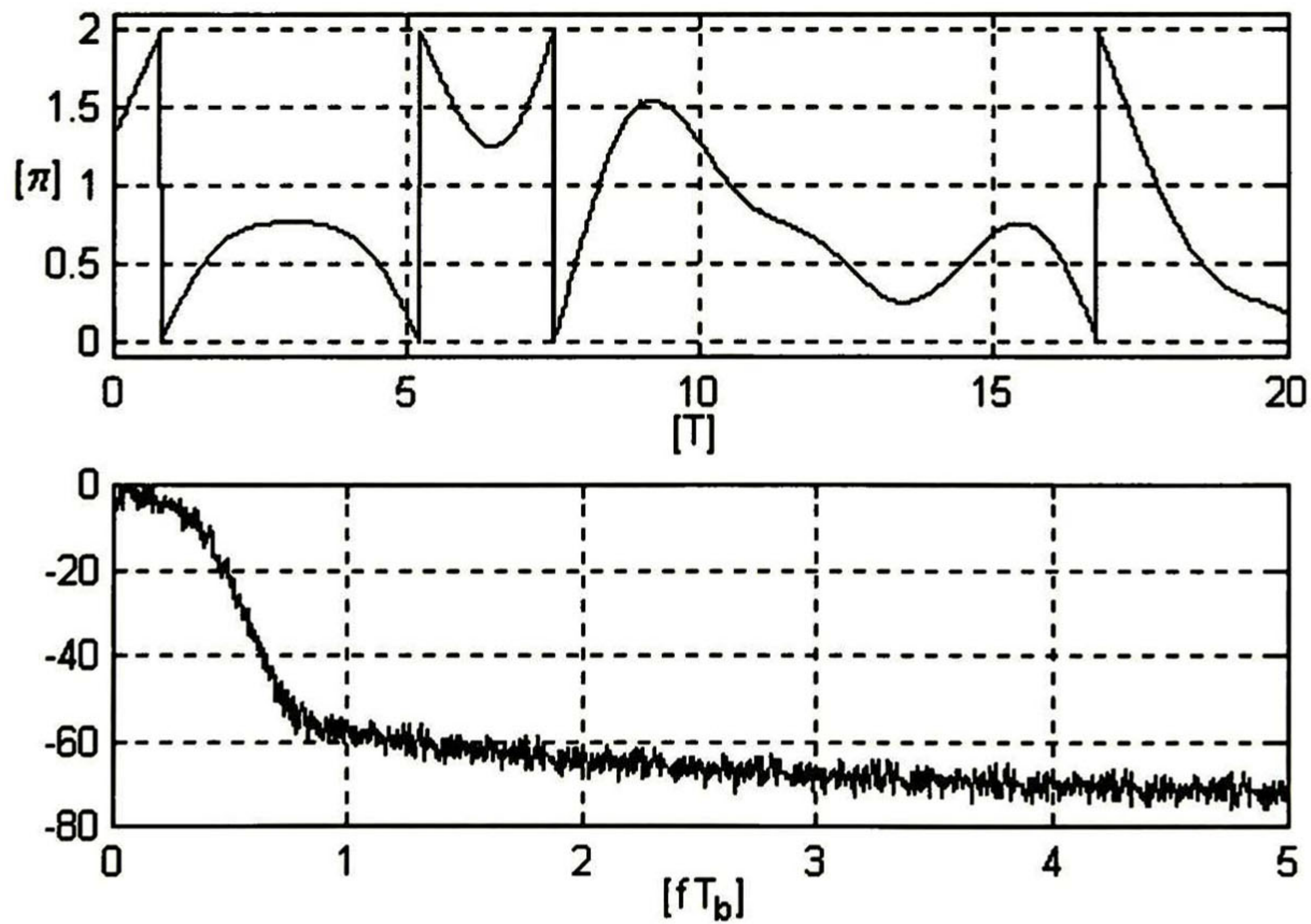


Fig. 3.40 Cambios de Fase y Espectro de TFM, $h=0.5$, $M=4$.

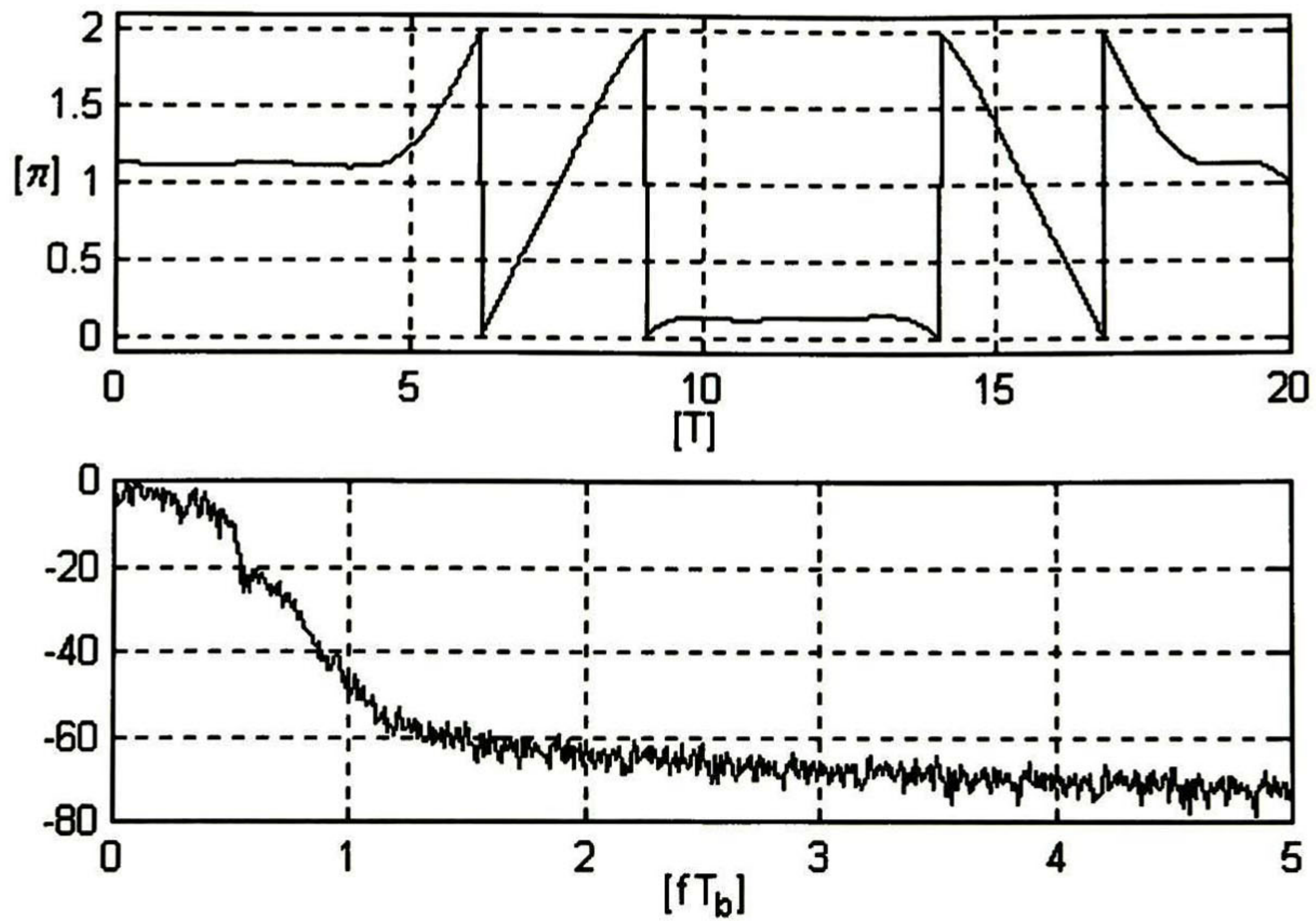


Fig. 3.41 Cambios de Fase y Espectro de TFM, $h=0.75$, $M=2$.

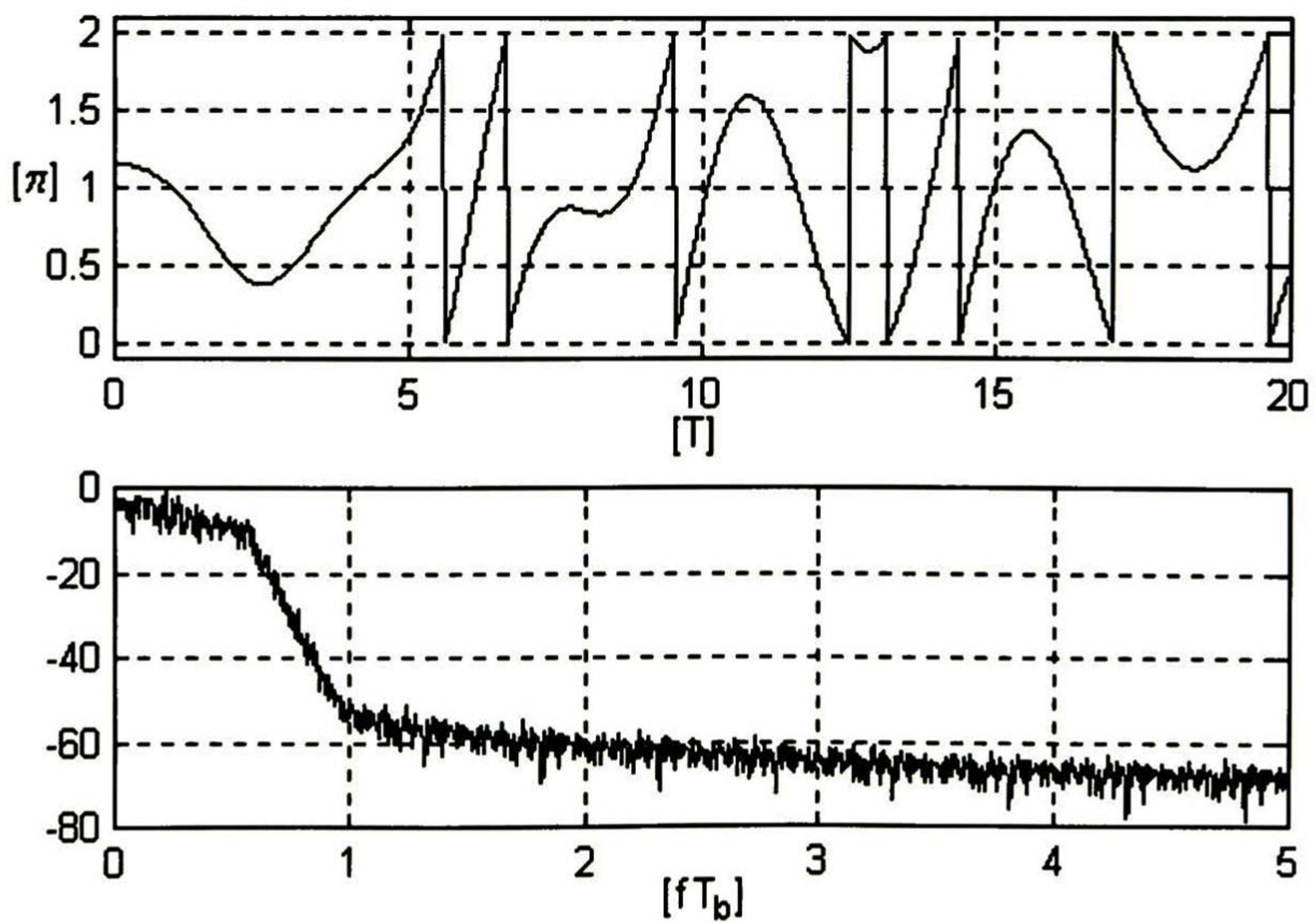


Fig. 3.42 Cambios de Fase y Espectro de TFM, $h=0.75$, $M=4$.

Esquemas Binarios			Porcentaje de potencia concentrado de $fT_b=0$ a 1.
BPSK			90.040 %
1REC	h=0.50		98.925 %
1REC	h=0.75		99.999 %
3REC	h=0.50		99.999 %
3REC	h=0.75		99.999 %
1RC	h=0.50		98.847 %
1RC	h=0.75		96.122 %
3RC	h=0.50		99.999 %
3RC	h=0.75		99.999 %
1SRC	h=0.50	$\beta=0.75$	98.706 %
1SRC	h=0.75	$\beta=0.75$	96.175 %
3SRC	h=0.50	$\beta=0.75$	99.933 %
3SRC	h=0.75	$\beta=0.75$	99.999 %
GMSK	h=0.50	BT=0.30	99.999 %
GMSK	h=0.75	BT=0.30	99.957 %
TFM	h=0.50		99.999 %
TFM	h=0.75		99.999 %

Tabla 3.1. Porcentajes de potencia concentrado en $fT_b=[0..1]$ para esquemas binarios.

Esquemas Cuaternarios			Porcentaje de potencia concentrado de $fT_b=0$ a 1.
QPSK			95.122 %
1REC	h=0.50		99.703 %
1REC	h=0.75		99.629 %
3REC	h=0.50		99.999 %
3REC	h=0.75		99.999 %
1RC	h=0.50		99.075 %
1RC	h=0.75		65.313 %
3RC	h=0.50		99.775 %
3RC	h=0.75		99.999 %
1SRC	h=0.50	$\beta=0.75$	99.155 %
1SRC	h=0.75	$\beta=0.75$	92.502 %
3SRC	h=0.50	$\beta=0.75$	99.702 %
3SRC	h=0.75	$\beta=0.75$	99.999 %
GMSK	h=0.50	BT=0.30	99.999 %
GMSK	h=0.75	BT=0.30	99.821 %
TFM	h=0.50		99.736 %
TFM	h=0.75		99.899 %

Tabla 3.2. Porcentajes de potencia concentrado en $fT_b=[0..1]$ para esquemas cuaternarios.

3.7. Efectos del filtraje y no linealidades sobre la envolvente y el espectro de la señal CPM.

En este apartado se presentan los resultados, en las simulaciones, del efecto que produce el filtrado sobre la envolvente de una señal CPM y su posterior aplicación a un sistema no lineal. Ésto, para comprobar que las señales CPM sufren una menor distorsión en su envolvente. Debido a que se eliminan los cruces por cero al pasar la señal por alguna no linealidad no se regeneran los lóbulos laterales de su espectro.

Se aplicó la misma simulación con BPSK, MSK, y 3RC, consistente en la generación de la señal con una portadora de 50Hz transportando datos a una velocidad de 1 bit por segundo, pasándola luego a través de un filtro Butterworth de 3^{er} orden con ancho de 2Hz para BPSK y de 2 y 1Hz para los casos de las señales CPM, y finalmente, pasándola a través de un amplificador limitador. Para cada una de la simulaciones se proporcionan la señal original, la señal filtrada que muestra la distorsión de la envolvente, y la señal después de la no linealidad. Las gráficas presentadas para estas formas de onda presentan la portadora con los cambios que sufre debido a la modulación; sólo que como esta portadora es más rápida que los cambios que se pretende mostrar, se observa completamente oscuro el interior de las señales. Por otro lado, se presentan también figuras que muestran los espectros de estas señales. En éstos las escalas horizontales están dadas en Hertz y las verticales en dBs, excepto en las gráficas de la respuesta del filtro.

3.7.1. Efectos sobre la señal BPSK.

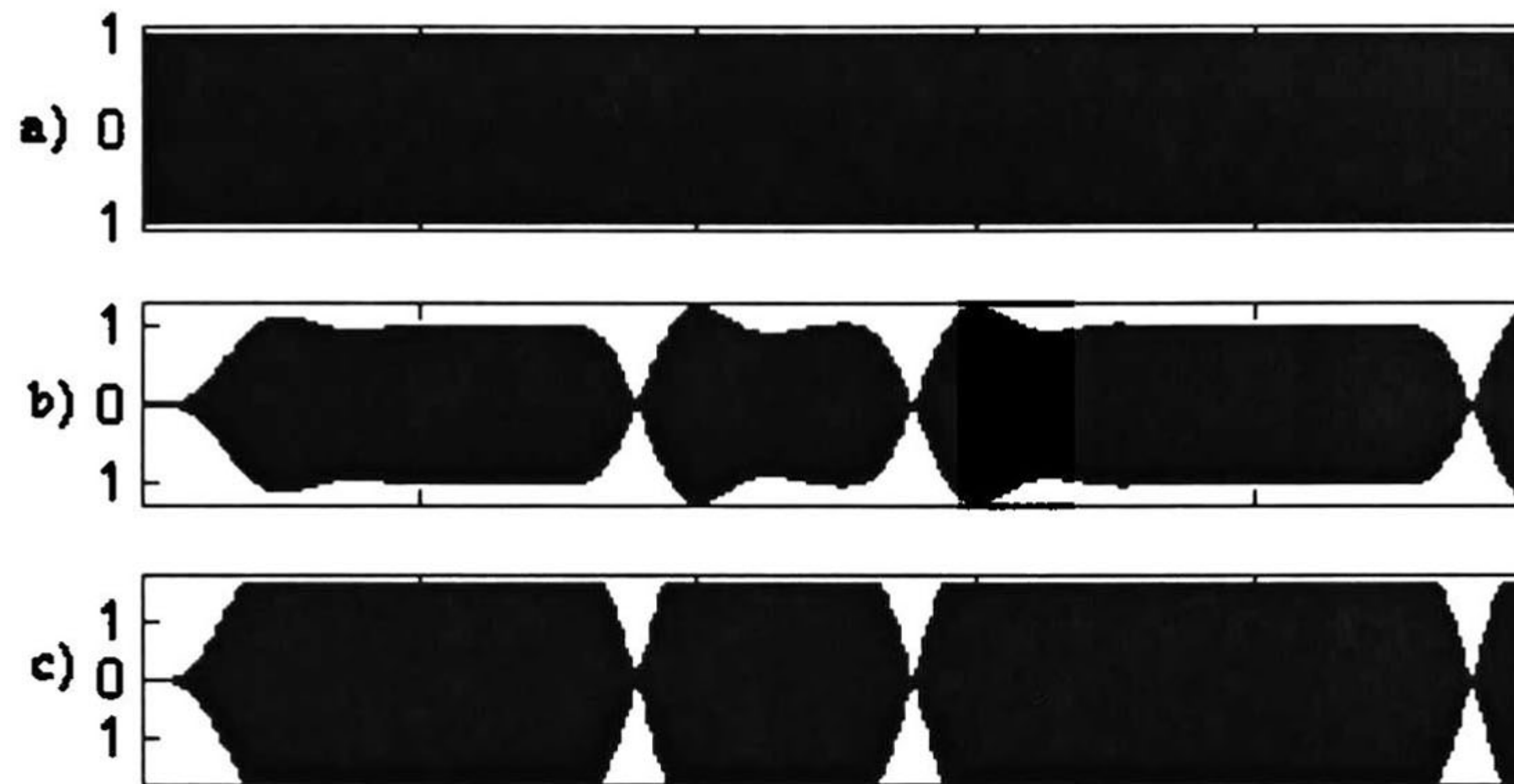


Fig. 3.43 Formas de onda de la señal BPSK: a)señal original; b)señal después de ser aplicada a un filtro pasabanda; c) señal después de ser aplicada a un amplificador no lineal.

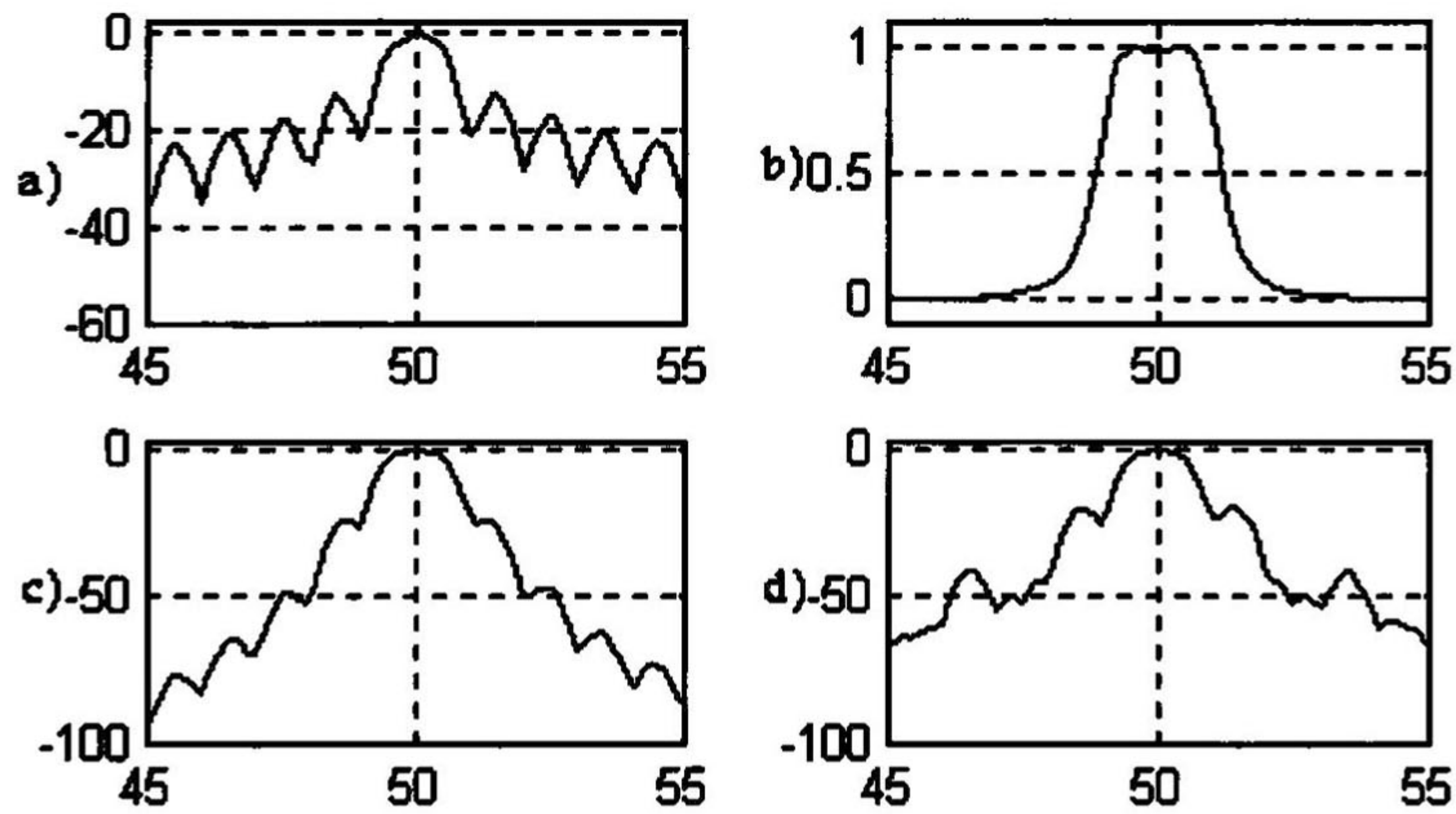


Fig. 3.44 Espectros de la señal BPSK: a)señal original; b)respuesta del filtro; c)señal después de ser aplicada al filtro pasabanda; d) señal después de ser aplicada a un amplificador no lineal.

3.7.2. Efectos sobre la señal MSK (1REC).

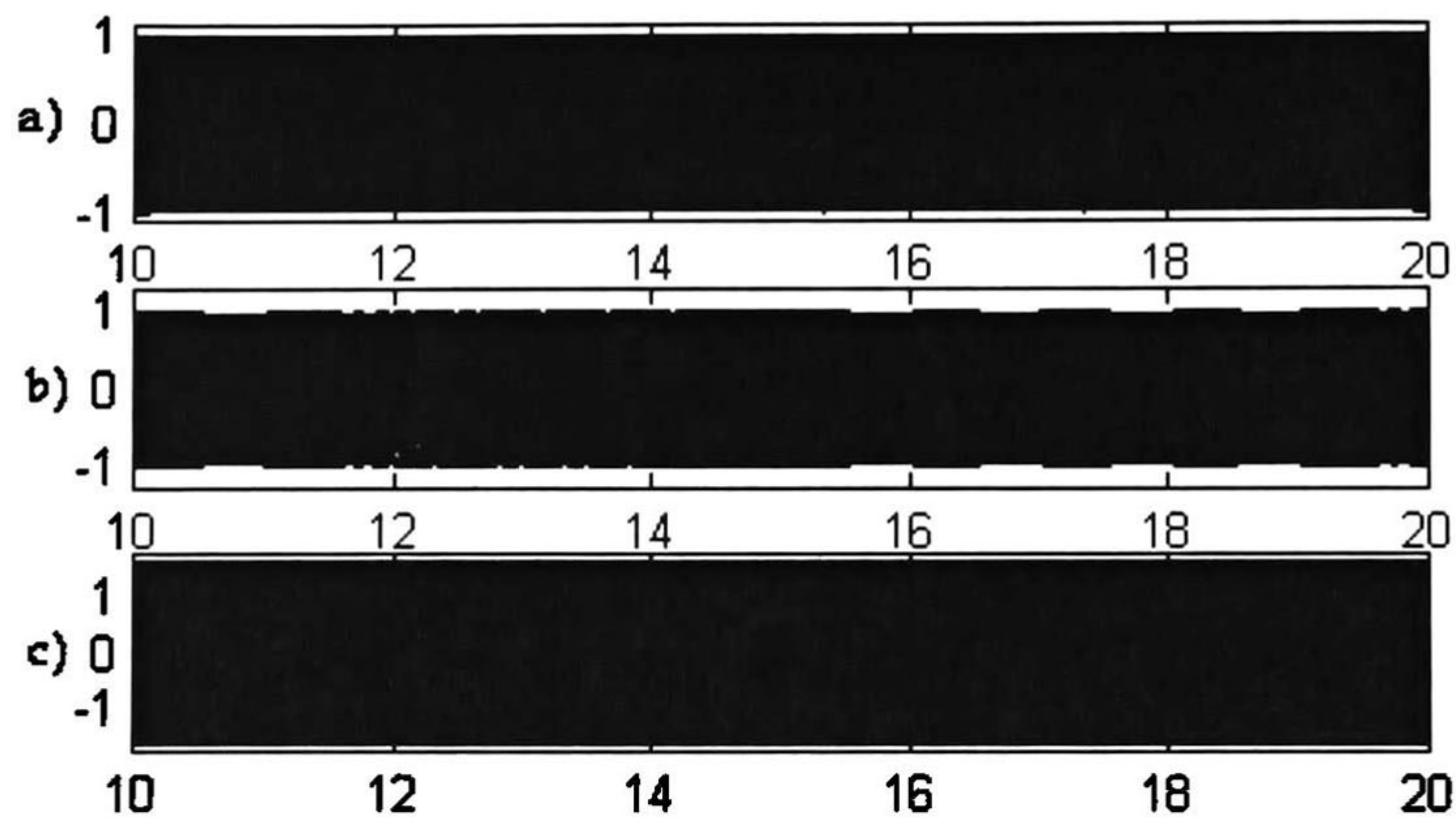


Fig. 3.45 Formas de onda de la señal MSK: a)señal original; b)señal después de ser aplicada a un filtro pasabanda de 2Hz de ancho de banda; c) señal después de ser aplicada a un amplificador no lineal.

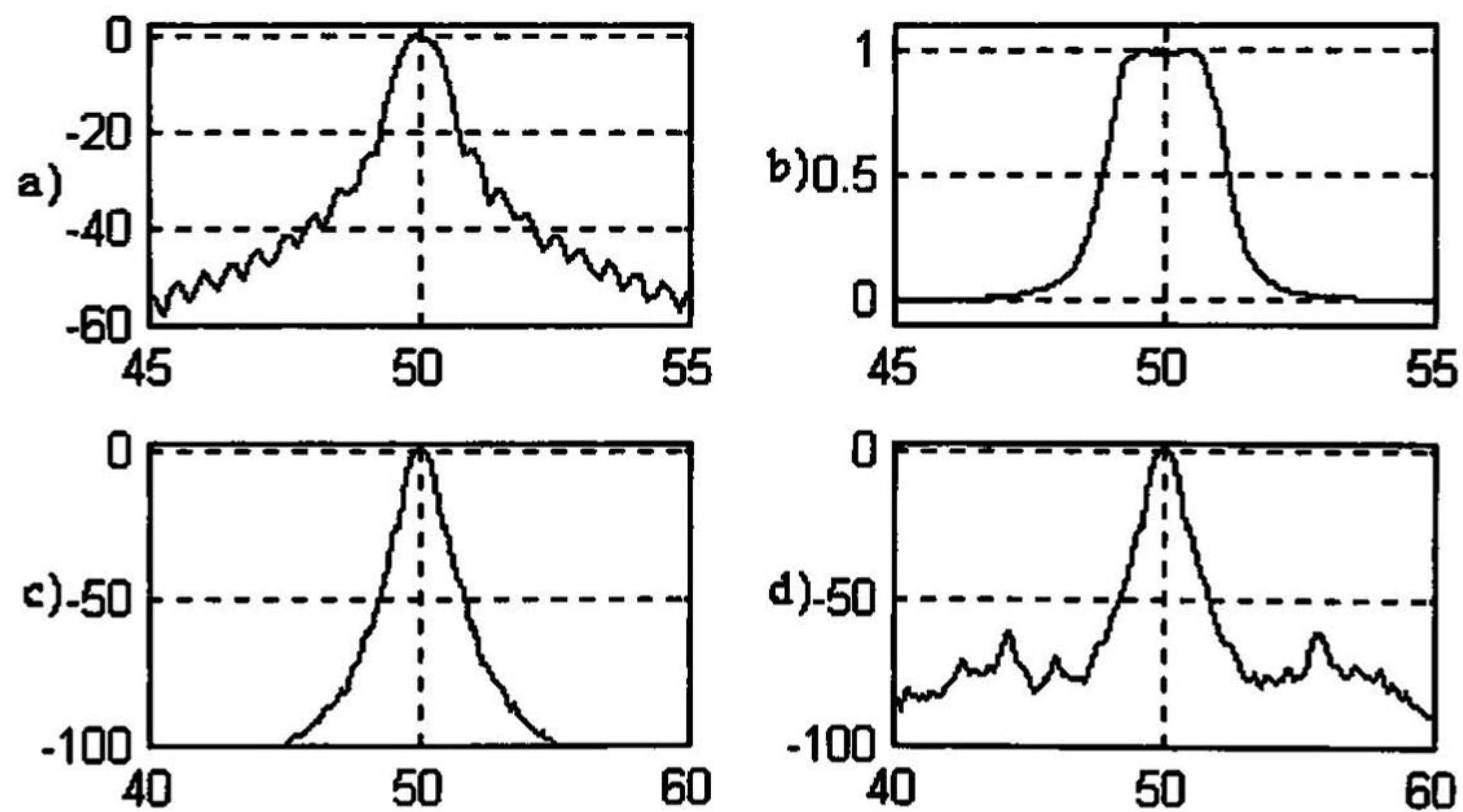


Fig. 3.46 Espectros de la señal MSK: a)señal original; b)respuesta del filtro; c)señal después de ser aplicada al filtro pasabanda de 2Hz de ancho de banda; d) señal después de ser aplicada a un amplificador no lineal.

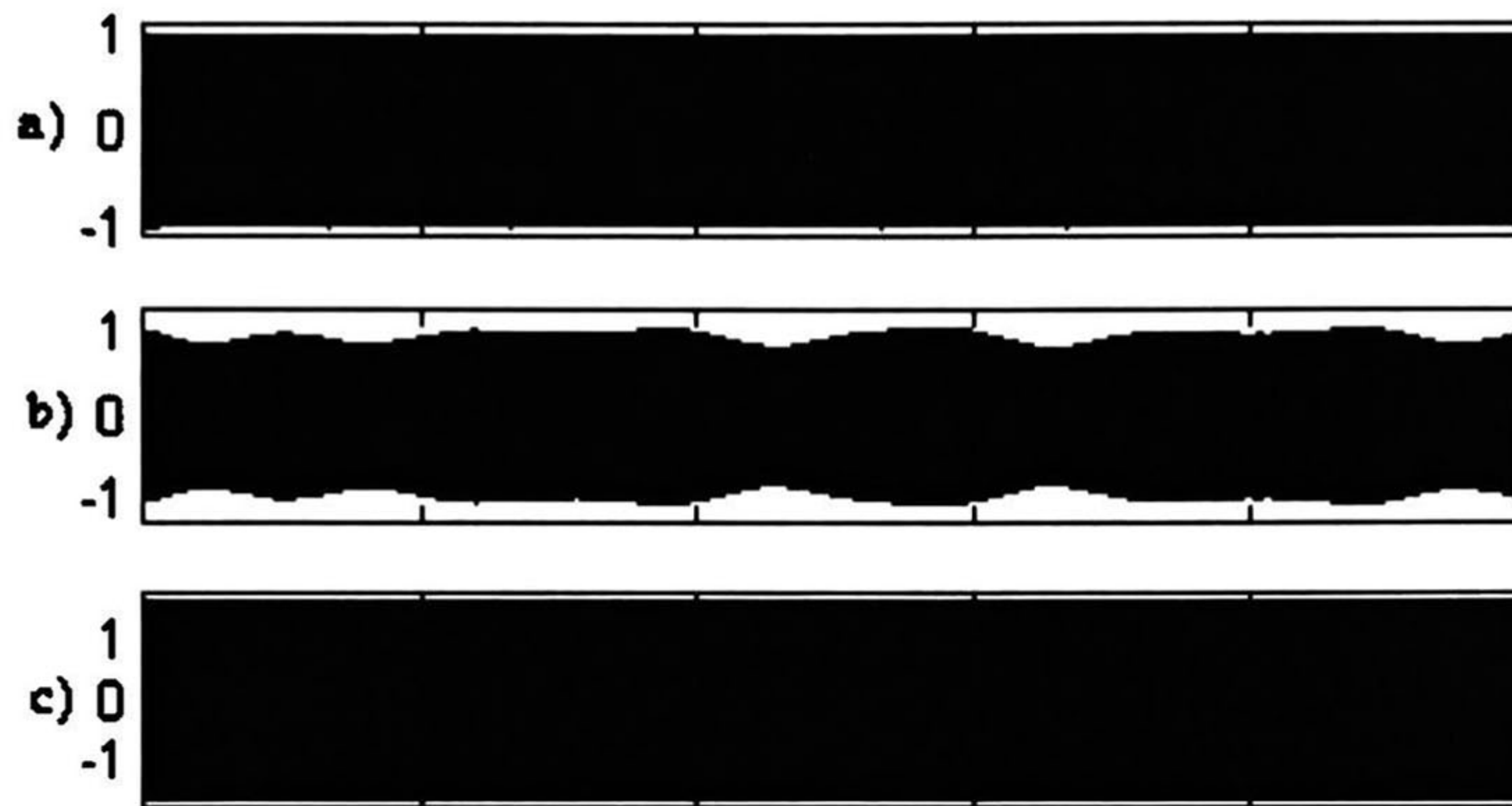


Fig. 3.47 Formas de onda de la señal MSK: a)señal original; b)señal después de ser aplicada a un filtro pasabanda de 1Hz de ancho de banda; c) señal después de ser aplicada a un amplificador no lineal.

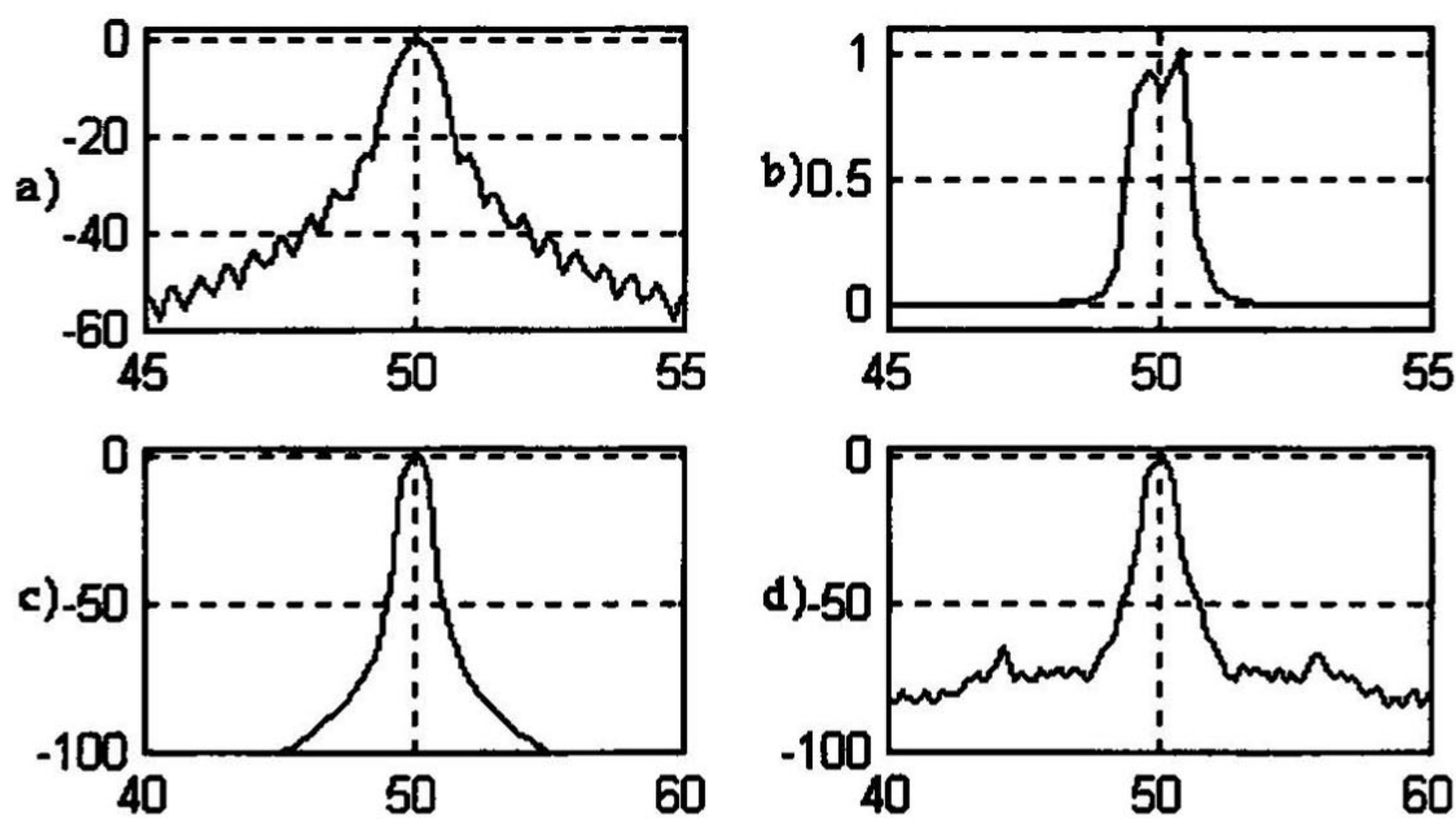


Fig. 3.48 Espectros de la señal MSK: a)señal original; b)respuesta del filtro; c)señal después de ser aplicada al filtro pasabanda de 1Hz de ancho de banda; d) señal después de ser aplicada a un amplificador no lineal.

3.7.3. Efectos sobre la señal 3RC.

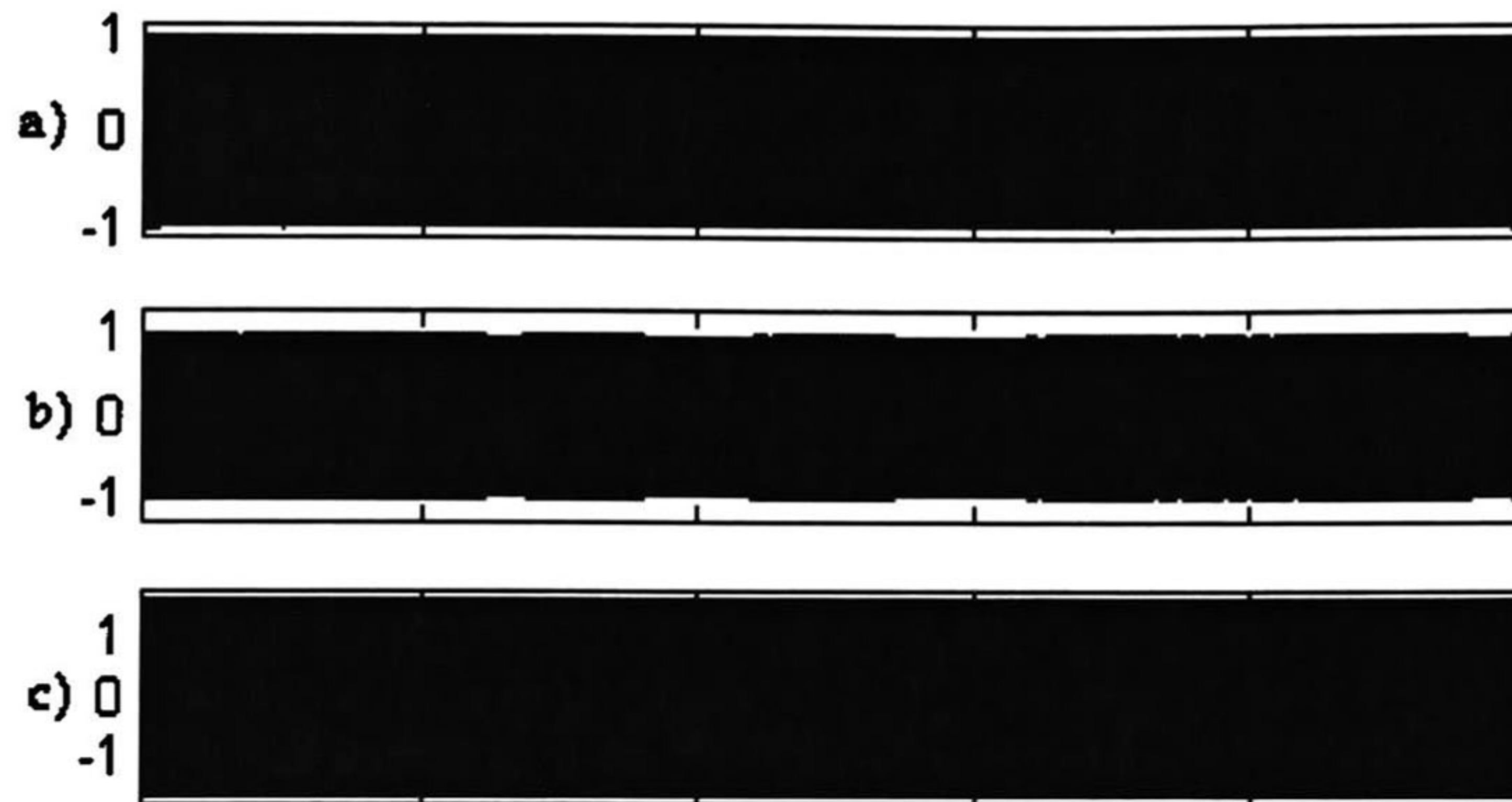


Fig. 3.49 Formas de onda de la señal 3RC: a)señal original; b)señal después de ser aplicada a un filtro pasabanda de 2Hz de ancho de banda; c) señal después de ser aplicada a un amplificador no lineal.

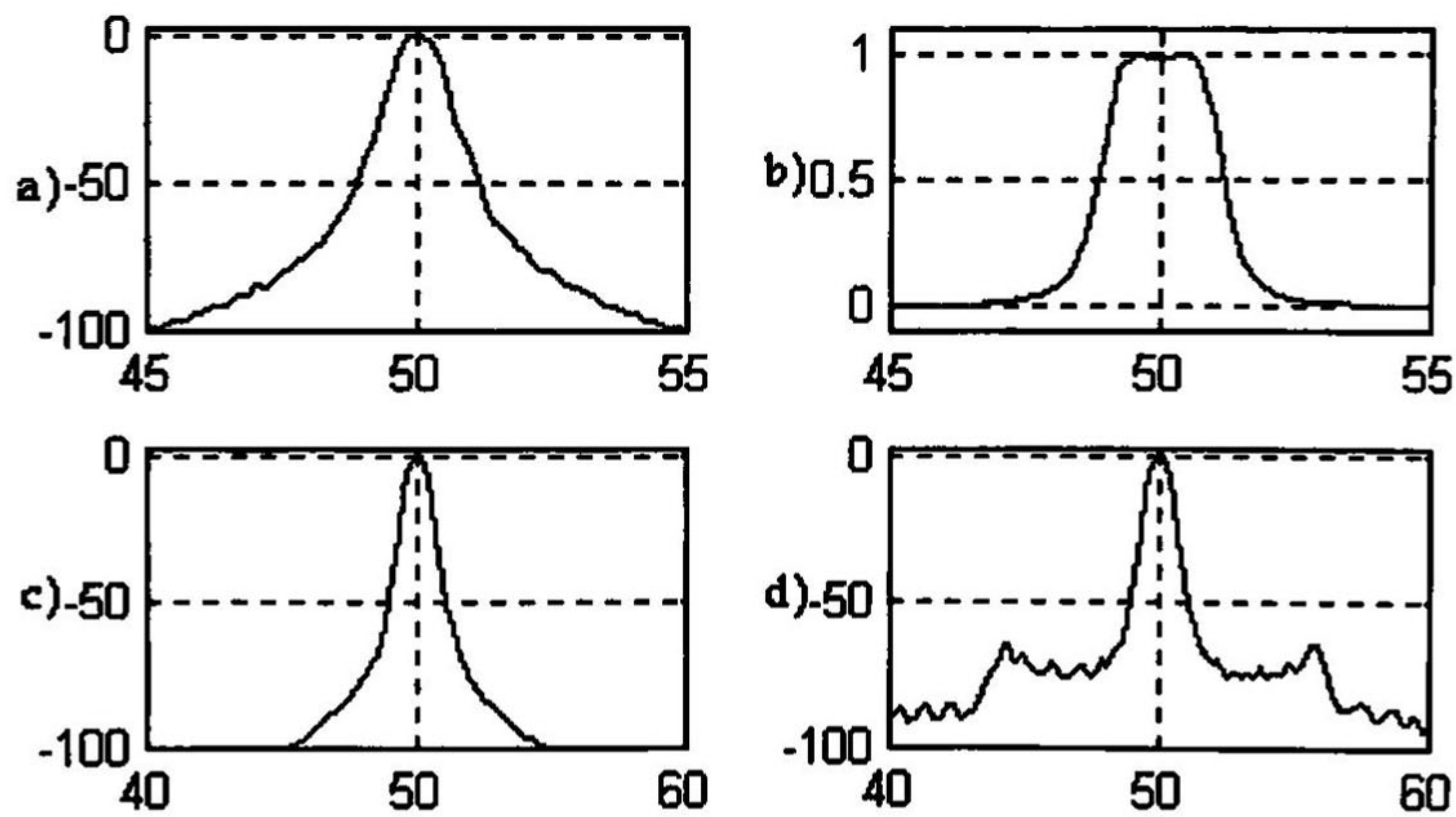


Fig. 3.50 Espectros de la señal 3RC: a)señal original; b)respuesta del filtro; c)señal después de ser aplicada al filtro pasabanda de 2Hz de ancho de banda; d) señal después de ser aplicada a un amplificador no lineal.

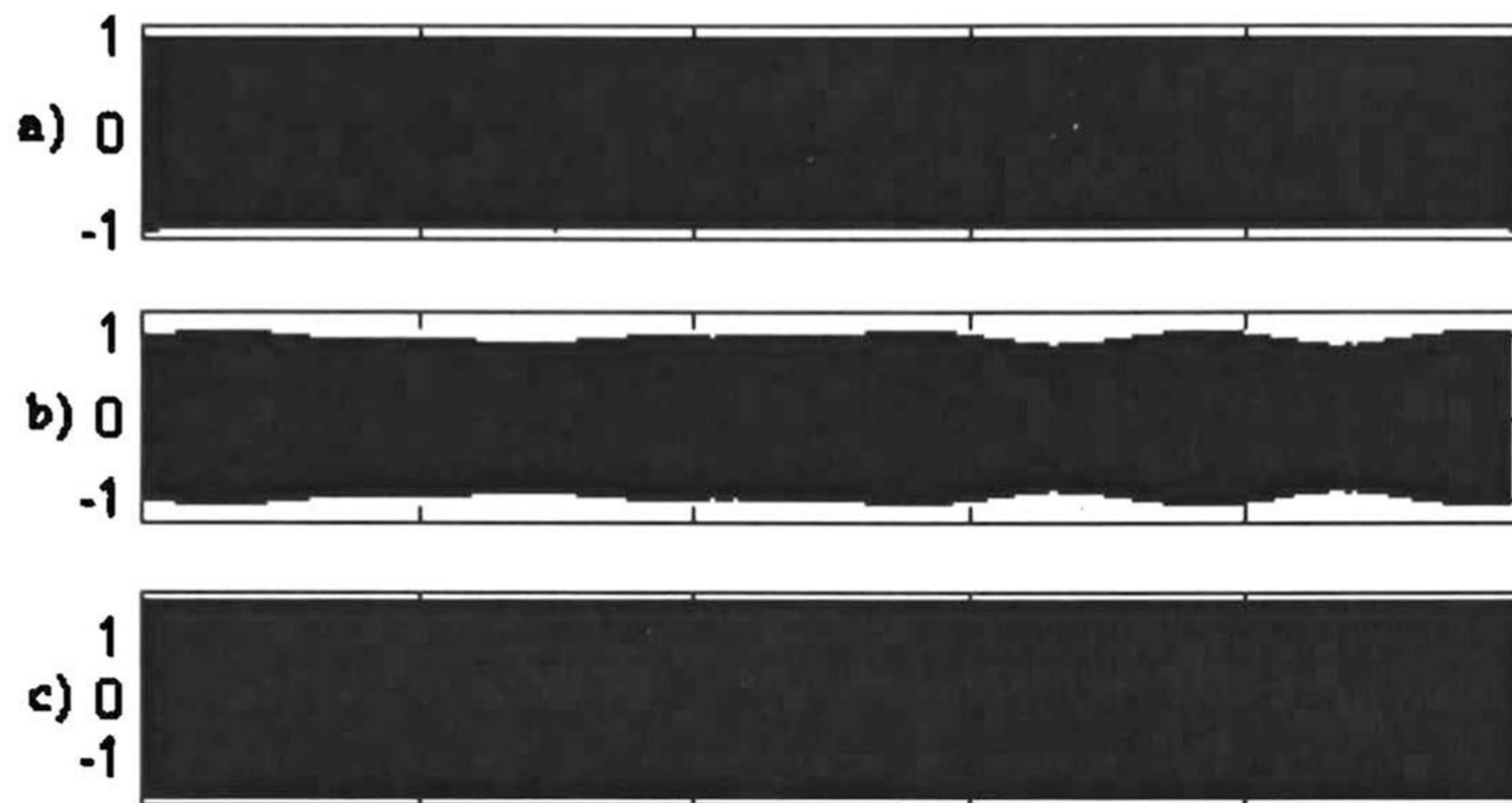


Fig. 3.51 Formas de onda de la señal 3RC: a)señal original; b)señal después de ser aplicada a un filtro pasabanda de 1Hz de ancho de banda; c) señal después de ser aplicada a un amplificador no lineal.

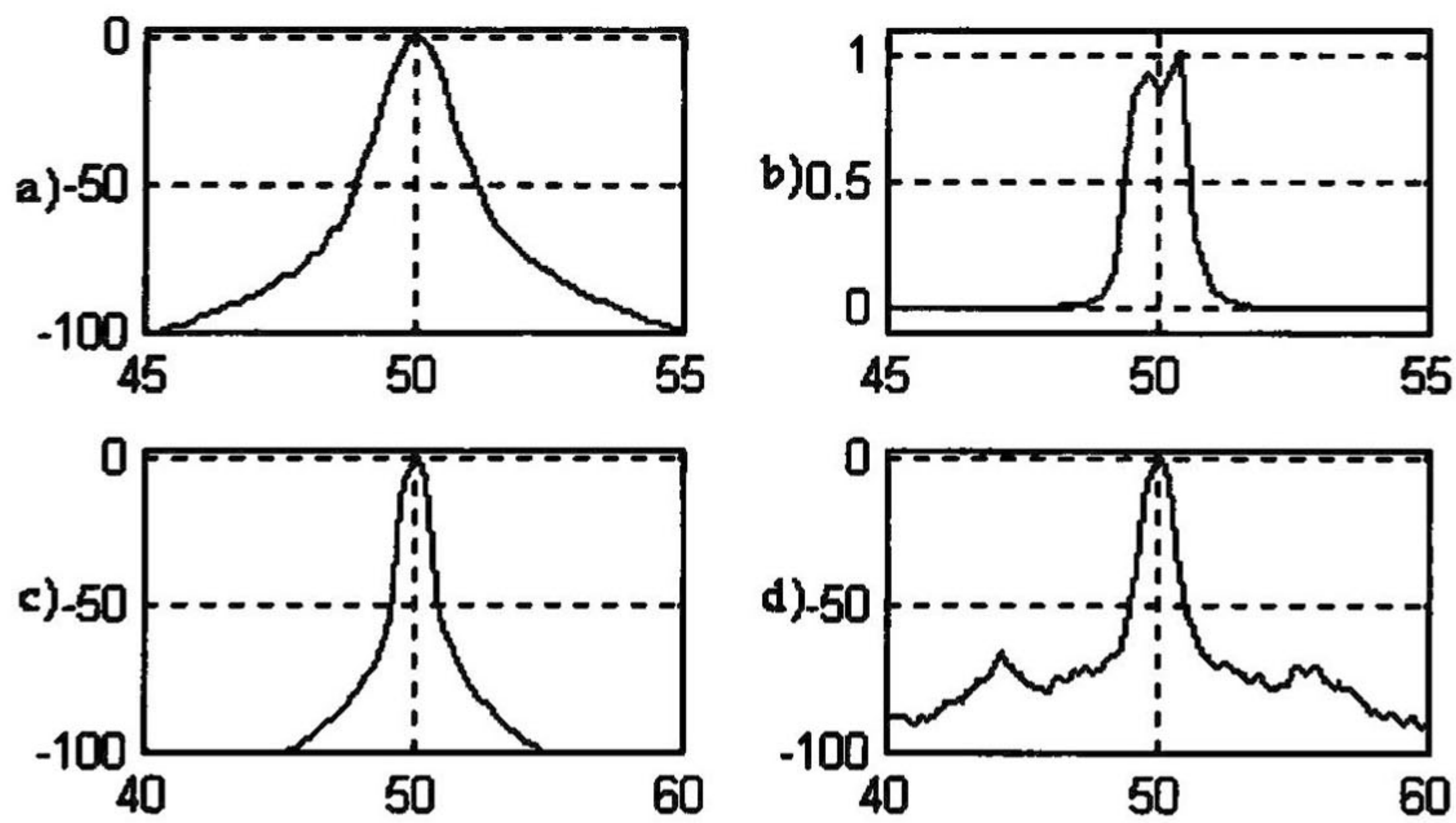


Fig. 3.52 Espectros de la señal 3RC: a)señal original; b)respuesta del filtro; c)señal después de ser aplicada al filtro pasabanda de 1Hz de ancho de banda; c) señal después de ser aplicada a un amplificador no lineal.

3.8. Conclusiones de las simulaciones.

De todo el conjunto de resultados de las simulaciones presentado anteriormente, se sigue que en general los esquemas CPM ocupan un ancho de banda menor que los esquemas tradicionales. Se puede observar además en las gráficas de cambio de fase como dicho cambio se produce en una forma continua. Dado que la gráfica se presenta sólo en el rango de 0 a 2π , se observan algunos cambios repentinos de la fase, pero éstos no significan pérdida de continuidad.

Algunos de los esquemas CPM tienen un lóbulo principal más ancho que los esquemas tradicionales, como se observa al comparar MSK con QPSK [2]; pero si se compara el ancho de banda que contiene el 99% de la potencia de la señal, para QPSK es de $8/T_b$ mientras que para MSK es de $1.2/T_b$. En consecuencia, MSK ocupa un espectro más angosto cuando es visto en términos de la potencia fraccional fuera de banda; o sea fuera de $fT_b=1$. Por ésto, MSK es más eficiente espectralmente hablando que QPSK. Se podría lograr una mayor eficiencia si se redujera el índice de modulación, pero es sabido que $h=0.5$ es el mínimo que permite ortogonalidad entre las señales por lo que la reducción resulta en incrementos en la probabilidad de error [3].

En general el ancho de banda ocupado por una señal CPM depende de la elección hecha en el índice de modulación, la forma del pulso $g(t)$ y su duración, y el número de señales M [2]. Como se ve de las figuras presentadas, valores pequeños de h resulta en señales con pequeño ancho de banda ocupado, mientras que valores mayores del índice de modulación resultan en mayores anchos de banda. Además, entre más suave sea el pulso de cambio en frecuencia $g(t)$ menor es el ancho de banda ocupado. Ésto se puede observar al comparar los anchos de banda para LREC y LRC bajo las mismas condiciones. Además conforme L se incrementa, el pulso $g(t)$ se hace más suave y por lo tanto el ancho de banda ocupado se reduce.

Se puede observar en las últimas comparaciones, por otro lado, que los esquemas con cambios de fase abruptos de 180° , como BPSK y QPSK, sufren gran distorsión en su envolvente al ser filtradas y, en ocasiones, presenta algunos cruces por cero. Cuando una de estas señales con envolvente distorsionada es aplicada a un elemento no lineal, se produce otra señal con un espectro en el que aparecen de nuevo algunos de los lóbulos laterales que antes habían sido eliminados con el filtro. En las señales CPM, en cambio, la envolvente sufre una menor distorsión evitando por completo los cruces por cero, incluso para filtros de ancho de banda menor; por tanto, el espectro de estas últimas señales luego de pasarlas por la no linealidad presentan una menor regeneración de los lóbulos laterales.

RECEPTORES PARA CPM

En el presente capítulo se hace una descripción de los principios de operación de varios receptores para Modulación de Fase Continua. Se estudian receptores tanto coherentes como no coherentes. En el caso de los receptores coherentes se asumirá que la sincronización está establecida, es decir, que la fase de la portadora y la temporización de símbolo se conocen con exactitud en el receptor. En el caso de recepción no coherente, se asumirá que el receptor conoce únicamente la temporización de símbolo.

A lo largo del reporte se asume un canal con ruido Gaussiano blanco aditivo. Se discuten los siguientes tipos de detectores. 1) los receptores óptimos coherente y no coherente para CPFSK, 2) el receptor óptimo por estimación de secuencias de máxima verosimilitud que utiliza el algoritmo de Viterbi y 3) detectores simples no coherentes, como lo son el detector diferencial y el detector por discriminador.

Para realizar las simulaciones se implementó el receptor óptimo de estimación de secuencias de máxima verosimilitud que aplica el algoritmo de Viterbi.

4.1. Receptores para CPFSK

En el caso de la modulación BPSK, el receptor óptimo de máxima verosimilitud coherente es simplemente un receptor lineal que observa un único intervalo, porque el comportamiento de la señal es independiente de los intervalos previos cuando los datos son independientes entre sí. Pero el esquema CPFSK, que es un caso particular de CPM, utiliza señales que tienen memoria entre intervalos. Ésta se debe a la restricción de continuidad en la fase, la cual otorga un mejor desempeño en ancho de banda y en energía. Por ésto es necesario que el receptor tenga una observación de múltiples intervalos en lugar de tomar las decisiones símbolo por símbolo. Aunque en la presente sección se hace hincapié en CPFSK, los receptores mostrados se pueden generalizar y

aplicar a esquemas CPM con cualquier forma de pulso y cualquier índice de modulación.

4.1.1. Receptor Coherente Óptimo para CPFSK

El receptor coherente óptimo para CPFSK toma una decisión sobre sólo un símbolo, basándose en la observación de una secuencia de símbolos consecutivos [11]. La señal CPFSK recibida durante el i -ésimo intervalo de señalización está dada por la siguiente ecuación que se obtiene haciendo las sustituciones correspondientes al esquema 1REC o CPFSK en la expresión (2.5) que es la expresión general de las señales de CPM:

$$r(t) = \sqrt{\frac{2E}{T}} \cos \left(2\pi f_0 t + \frac{\alpha_i \pi h [t - (i-1)T]}{T} + \pi h \sum_{j=1}^{i-1} \alpha_j + \varphi_0 \right) + n(t) \quad (i-1)T \leq t \leq iT \quad (4.1)$$

En esta ecuación, α_i es el dato correspondiente al i -ésimo intervalo, que pertenece a un alfabeto M -ario ($\pm 1, \pm 3, \dots, \pm M-1$), f_0 es la frecuencia de la portadora, φ_0 es una fase inicial arbitraria, el término de la sumatoria representa a la memoria del sistema y $n(t)$ es el ruido añadido a la señal, que se asume blanco y con distribución de probabilidad gaussiana. En lo sucesivo, se utilizará la notación siguiente:

$$r(t) = s(t, \alpha_1, D_k) + n(t) \quad (4.2)$$

En esta expresión, α_1 es el primer símbolo recibido, y D_k es la $(N-1)$ tupla dada por $D_k = \{\alpha_2, \dots, \alpha_N\}$. Esta tupla son los símbolos recibidos después de α_1 . Como se indicó anteriormente, para los sistemas coherentes se asumirá que la fase inicial es conocida y sin perder generalidad se tomará igual a cero.

Entonces, el receptor determinará los M parámetros para una detección óptima dados por:

$$\begin{aligned} l_1 &= \int \exp \left\{ \frac{2}{N_0} \int_0^{NT} r(t) s(t, 1, D_k) dt \right\} f(D_k) dD_k \\ l_2 &= \int \exp \left\{ \frac{2}{N_0} \int_0^{NT} r(t) s(t, -1, D_k) dt \right\} f(D_k) dD_k \\ &\quad \vdots \\ l_{M-1} &= \int \exp \left\{ \frac{2}{N_0} \int_0^{NT} r(t) s(t, M-1, D_k) dt \right\} f(D_k) dD_k \\ l_M &= \int \exp \left\{ \frac{2}{N_0} \int_0^{NT} r(t) s(t, -(M-1), D_k) dt \right\} f(D_k) dD_k \end{aligned} \quad (4.3)$$

donde $f(D_k)$ es la función de densidad de probabilidad *a priori* de la secuencia de datos D_k . La expresión (4.3) sugiere que el receptor deba tener réplicas de la señal transmitida con N símbolos correspondientes a N intervalos y se busca encontrar cual de estas

señales tiene la mínima distancia con la señal recibida (éste es el criterio óptimo para la detección de señales en presencia de ruido blanco Gaussiano). Ésto equivale a buscar la señal que logra una máxima correlación con la señal recibida.

Como se supone que el símbolo en un intervalo es independiente de todos los demás y que puede tomar cualquiera de los M valores del alfabeto con igual probabilidad, se tiene que:

$$\int f(D_k) dD_k = \int \int \cdots \int f(\alpha_2) f(\alpha_3) \cdots f(\alpha_N) d\alpha_2 d\alpha_3 \cdots d\alpha_N \quad (4.4)$$

$$f(\alpha_i) = \frac{1}{M} \sum_{j=0}^{M-1} \delta(\alpha_i + 2j - M + 1) \quad (4.5)$$

donde $\delta(t)$ es la función delta de Dirac. Asumiendo que todos los datos tienen la misma probabilidad de ocurrencia, los parámetros del detector óptimo se vuelven parámetros de máxima verosimilitud. Como D_k puede tener $m=M^{N-1}$ posibilidades diferentes y todas son igualmente probables, las integrales sobre D_k pueden ser evaluadas para obtener:

$$\begin{aligned} l_1 &= \sum_{j=1}^m \exp \left[\frac{2}{N_0} \int_0^{NT} r(t) s(t, 1, D_j) dt \right] \\ l_2 &= \sum_{j=1}^m \exp \left[\frac{2}{N_0} \int_0^{NT} r(t) s(t, -1, D_j) dt \right] \\ &\quad \vdots \\ l_{M-1} &= \sum_{j=1}^m \exp \left[\frac{2}{N_0} \int_0^{NT} r(t) s(t, M-1, D_j) dt \right] \\ l_M &= \sum_{j=1}^m \exp \left[\frac{2}{N_0} \int_0^{NT} r(t) s(t, -(M-1), D_j) dt \right] \end{aligned} \quad (4.6)$$

El receptor decide por el símbolo α_1 que corresponda al mayor parámetro entre l_1 y l_M . El diagrama a bloques de este receptor se muestra en la Fig. 4.1. Como se puede ver en esta figura, el receptor consta de M bloques cada uno con m correlacionadores y un sumador.

Ahora bien, si definimos:

$$x_{\lambda j} = \begin{cases} \int_0^{NT} r(t) s(t, \lambda, D_j) dt, & \lambda \text{ impar} \\ \int_0^{NT} r(t) s(t, -(\lambda-1), D_j) dt, & \lambda \text{ par} \end{cases} \quad \lambda = 1, 2, \dots, M, \quad (4.7)$$

entonces los parámetros de máxima verosimilitud se pueden reescribir como:

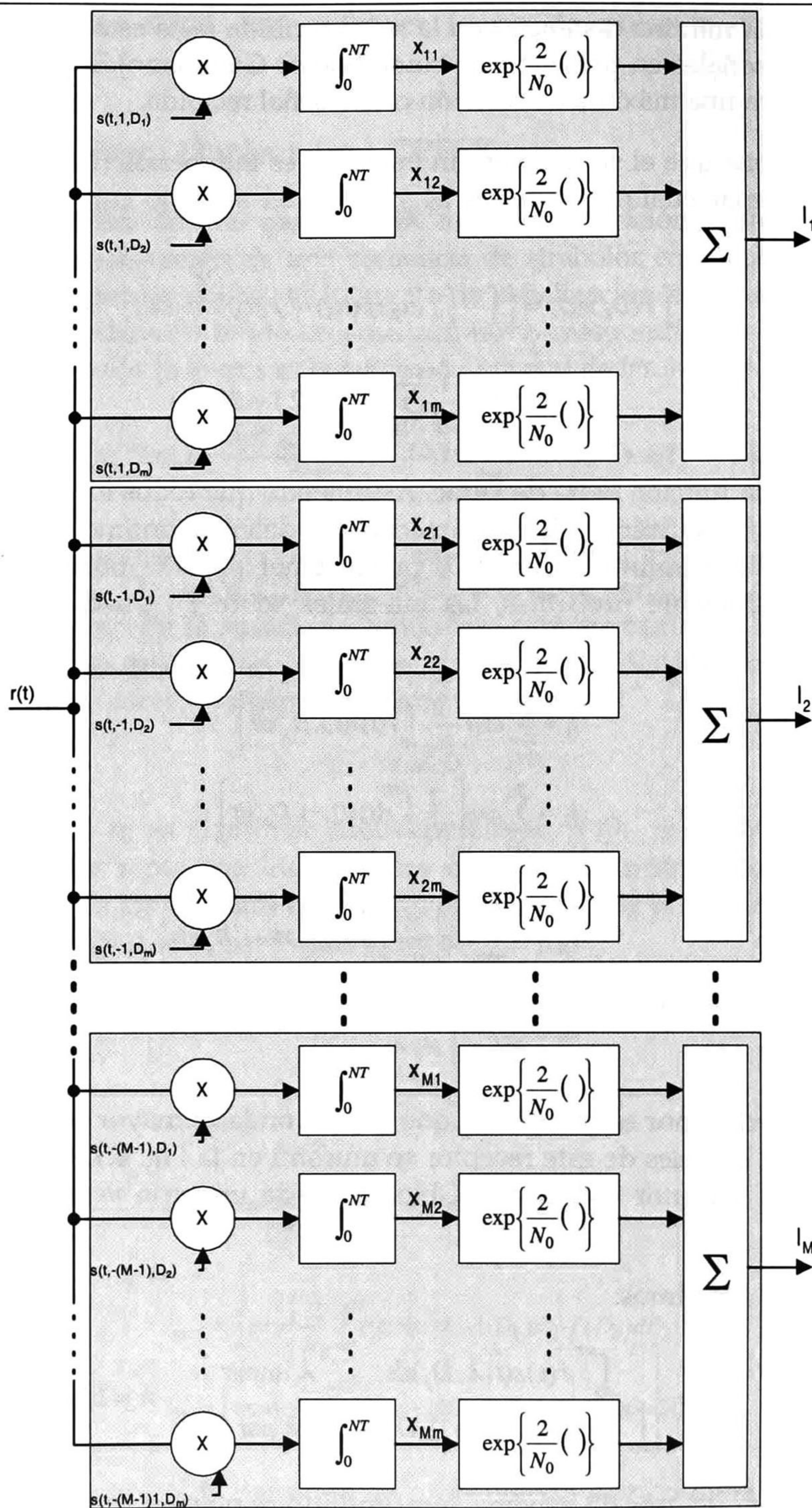


Fig. 4.1 Receptor coherente óptimo para CPFSK.

$$l_{\lambda} = \sum_{j=1}^m \exp\left(\frac{2}{N_0} x_{\lambda j}\right) \quad \lambda = 1, 2, \dots, M \quad (4.8)$$

y para relaciones señal a ruido (SNR) grandes se puede que:

$$\sum_{j=1}^m \exp\left(\frac{2}{N_0} x_{\lambda j}\right) \approx \exp\left(\frac{2}{N_0} x_{\lambda}\right) \quad (4.9)$$

donde x_{λ} es el mayor de los $x_{\lambda j}$. Ésto lleva al receptor subóptimo que, para relaciones señal a ruido grandes, utiliza esta aproximación en lugar de llevar a cabo la sumatoria.

El desempeño en cuanto a probabilidad de error del detector óptimo coherente es difícil de calcular pero ha sido acotado y evaluado numéricamente. Los resultados de las figuras que se presentan a continuación provienen de la literatura especializada [2]. En las Figs. 4.2, 4.3 y 4.4 se muestran comparaciones de las probabilidades de error obtenidas para alfabetos binario, cuaternario y octal [2]. El índice de modulación mostrado es el que resultó en la mínima probabilidad de error y el número de intervalos de señalización durante los cuales se observa la señal y se lleva a cabo la correlación es n . Como se puede ver de la Fig. 4.2, se obtiene una mejoría de 2.5 dB con respecto a FSK ortogonal (en el cual $n=1$) para un receptor que lleva a cabo correlación sobre dos intervalos de señalización. Hay además, una ganancia adicional de 1.5 dB extendiendo la correlación a tres símbolos. Pero el seguir aumentando el tiempo de correlación resulta, a partir de este punto, en ganancias poco significativas.

Resultados similares a los anteriores se logran con alfabetos mayores, como se puede ver en las Fig. 4.3 y 4.4 [2]. Igualmente, en estas figuras los índices de modulación mostrados son los que resultaron en la menor probabilidad de error de símbolo según la literatura especializada consultada.

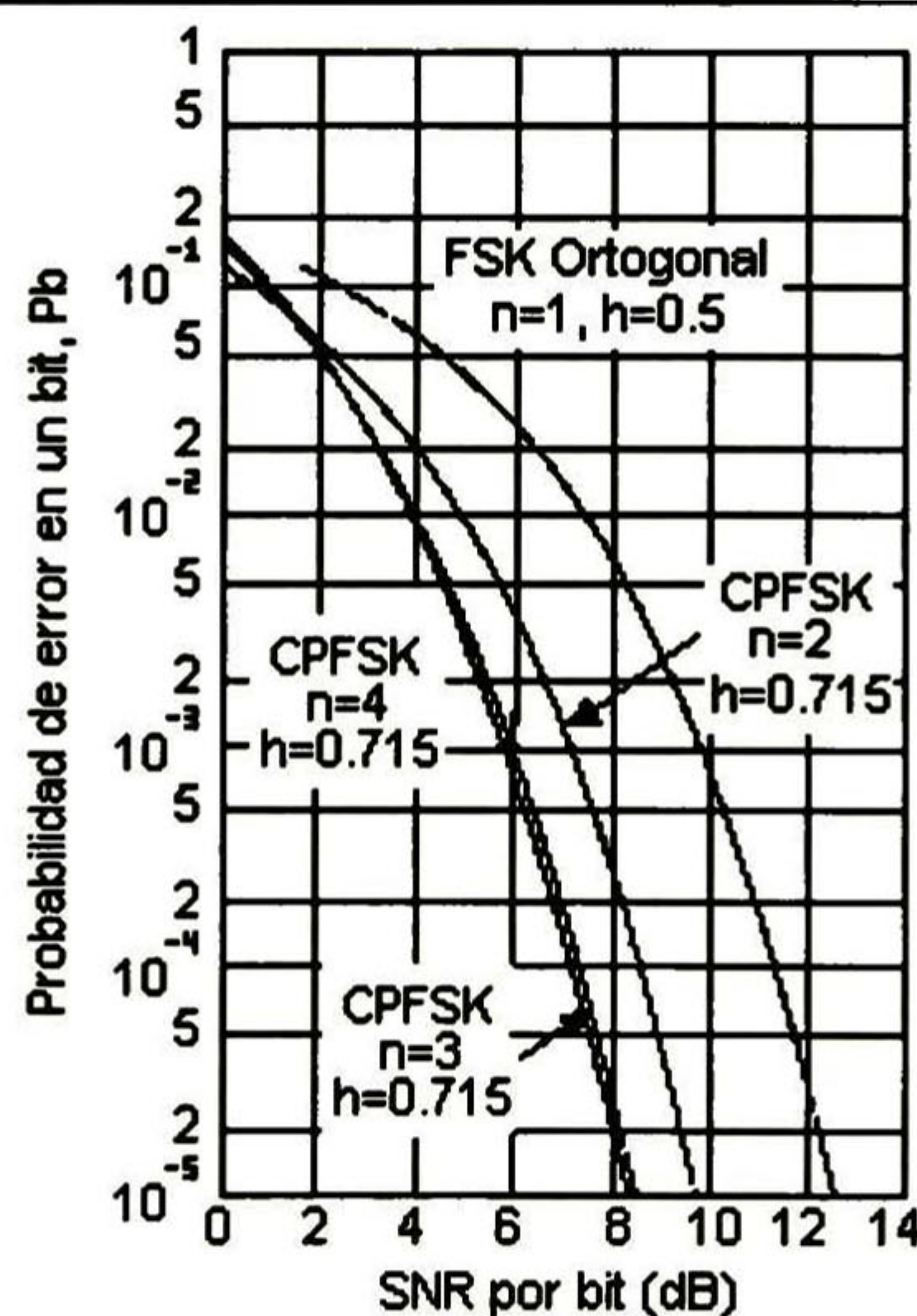


Fig. 4.2 Desempeño de CPFSK Binario con detección coherente

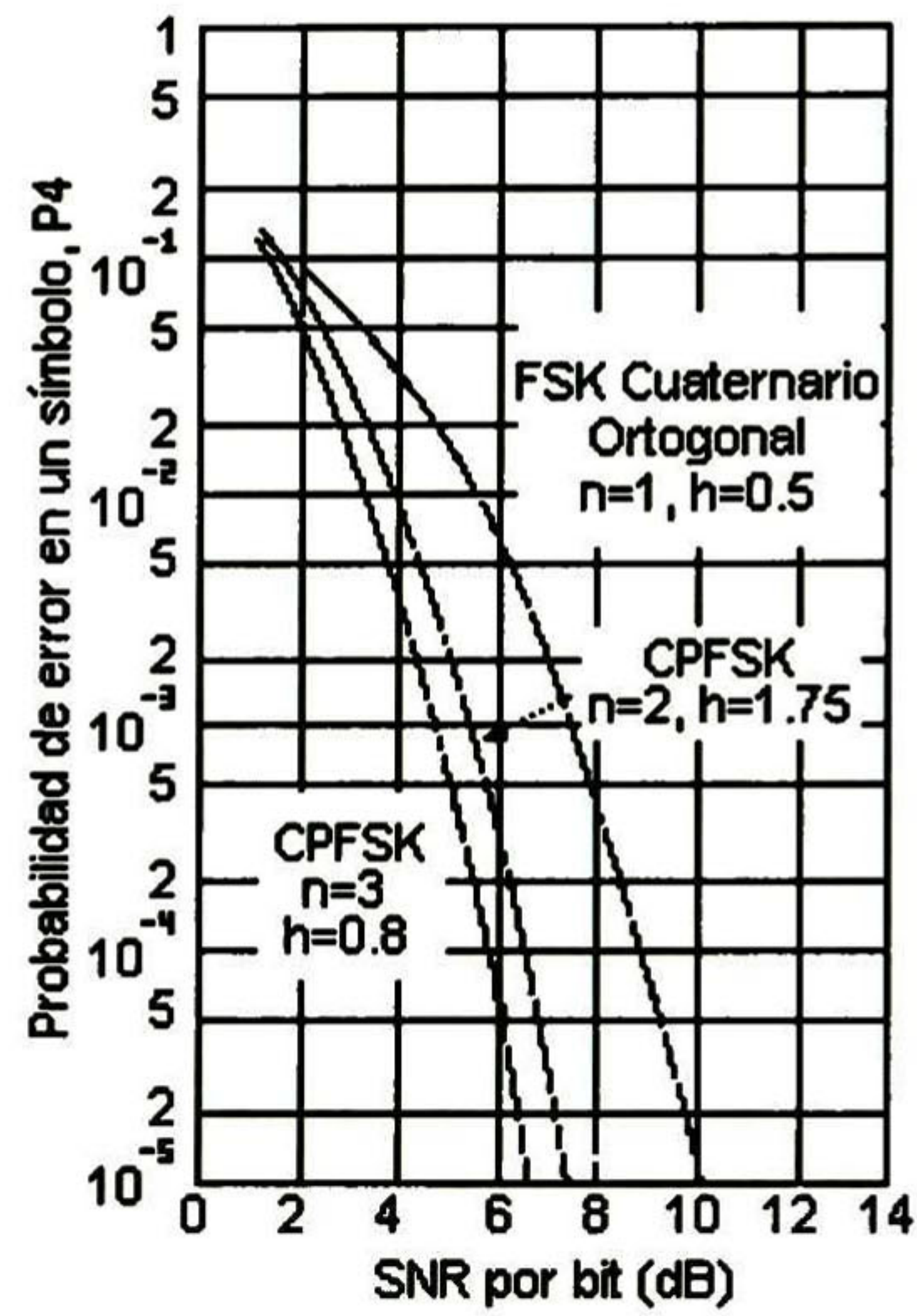


Fig. 4.3 Desempeño de CPFSK Cuaternario con detección coherente

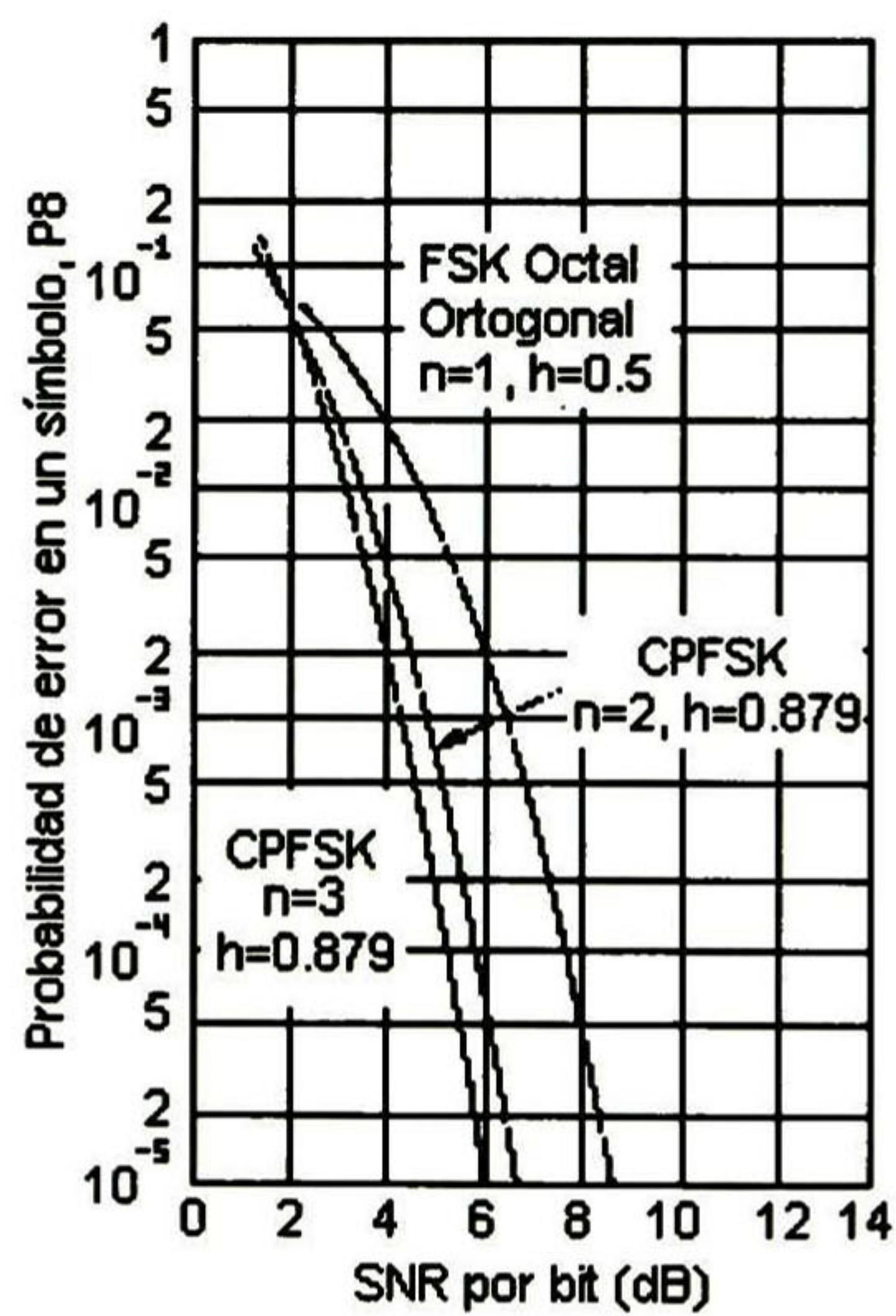


Fig. 4.4 Desempeño de CPFSK Octal con detección coherente

4.1.2. Receptor No Coherente Óptimo para CPFSK

En la detección no coherente, la fase inicial se asume no conocida y con una función de densidad de probabilidad uniforme en el intervalo de 0 a 2π . La señal recibida es entonces [11]:

$$r(t) = s(t, \alpha_{n+1}, D_k, \Phi) + n(t) \quad (4.10)$$

En este caso, la señal recibida es observada durante $N=2n+1$ símbolos y se hace una decisión sobre el símbolo del medio; por ésto se tiene que:

$$D_k = \{\alpha_1, \alpha_2, \dots, \alpha_n, \alpha_{n+2}, \dots, \alpha_{2n+1}\} \quad (4.11)$$

Se puede ver que D_k puede tomar $\mu=M^{2n}$ valores. Para este caso los parámetros del detector óptimo tendrán una forma parecida a los del caso coherente, con la diferencia de que ahora como la fase inicial es desconocida se debe incluir el promedio sobre esta fase:

$$\begin{aligned} l_1 &= \int_{\Phi} \int_D \exp\left\{\frac{2}{N_0} \int_0^{NT} r(t)s(t, 1, D_k, \Phi) dt\right\} f_D(D) f_{\Phi}(\Phi) dD d\Phi \\ l_2 &= \int_{\Phi} \int_D \exp\left\{\frac{2}{N_0} \int_0^{NT} r(t)s(t, -1, D_k, \Phi) dt\right\} f_D(D) f_{\Phi}(\Phi) dD d\Phi \\ &\quad \vdots \\ l_{M-1} &= \int_{\Phi} \int_D \exp\left\{\frac{2}{N_0} \int_0^{NT} r(t)s(t, M-1, D_k, \Phi) dt\right\} f_D(D) f_{\Phi}(\Phi) dD d\Phi \\ l_M &= \int_{\Phi} \int_D \exp\left\{\frac{2}{N_0} \int_0^{NT} r(t)s(t, -(M-1), D_k, \Phi) dt\right\} f_D(D) f_{\Phi}(\Phi) dD d\Phi \end{aligned} \quad (4.12)$$

Aplicando el mismo procedimiento que para el caso coherente, para llevar a cabo la integral sobre D , se tiene que:

$$\begin{aligned} l_1 &= \int_{\Phi} \frac{1}{\mu} \sum_{k=1}^{\mu} \exp\left\{\frac{2}{N_0} \int r(t)s(t, 1, D_k, \Phi) dt\right\} f(\Phi) d\Phi \\ l_2 &= \int_{\Phi} \frac{1}{\mu} \sum_{k=1}^{\mu} \exp\left\{\frac{2}{N_0} \int r(t)s(t, -1, D_k, \Phi) dt\right\} f(\Phi) d\Phi \\ &\quad \vdots \\ l_M &= \int_{\Phi} \frac{1}{\mu} \sum_{k=1}^{\mu} \exp\left\{\frac{2}{N_0} \int r(t)s(t, -(M-1), D_k, \Phi) dt\right\} f(\Phi) d\Phi \end{aligned} \quad (4.13)$$

Después de un desarrollo matemático sobre las expresiones anteriores, asumiendo la distribución uniforme de la fase, cambiando a $r(t)$ por su expresión completa y con varios cambios de variable se tiene que:

$$\begin{aligned} l_1 &= \frac{1}{\mu} \sum_{k=1}^{\mu} I_0 \left(\frac{2}{N_0} z_{1k} \right) \\ l_2 &= \frac{1}{\mu} \sum_{k=1}^{\mu} I_0 \left(\frac{2}{N_0} z_{2k} \right) \\ &\vdots \\ l_M &= \frac{1}{\mu} \sum_{k=1}^{\mu} I_0 \left(\frac{2}{N_0} z_{Mk} \right) \end{aligned} \quad (4.14)$$

donde

$$\begin{aligned} z_{ak}^2 &= x_{ak}^2 + y_{ak}^2, \\ I_0(z) &= \frac{1}{2\pi} \int_0^{2\pi} \exp(z \cos \theta) d\theta, \end{aligned} \quad (4.15)$$

$$x_{ak} = \begin{cases} \int r(t) s(t, a, D_k, 0) & a \text{ impar} \\ \int r(t) s(t, -(a-1), D_k, 0) & a \text{ par} \end{cases}$$

y

$$y_{ak} = \begin{cases} \int r(t) s(t, a, D_k, \pi/2) & a \text{ impar} \\ \int r(t) s(t, -(a-1), D_k, \pi/2) & a \text{ par} \end{cases}$$

Nótese que $I_0(\)$ es la función de Bessel modificada de orden cero.

En la Fig. 4.5 se muestra el diagrama a bloques de este receptor óptimo. Éste consta pues de M bloques, cada uno con μ sumandos. Cada sumando usa dos correlacionadores, un detector de envolvente y un dispositivo que lleve a cabo la función de Bessel.

Al igual que para el receptor coherente para el caso en que la relación señal a ruido sea grande, se puede establecer un receptor subóptimo no coherente que basa su decisión en la mayor de las z_{ak} . Ésto se obtiene dado que la función modificada de Bessel de orden cero es monótona y, por tanto, se puede hacer la siguiente aproximación:

$$\sum_{k=1}^{\mu} I_0(z_{ak}) \approx I_0(z_{a\Delta}) \quad (4.16)$$

siendo $z_{a\Delta}$ es el mayor de los z_{ak} .

En el caso del detector óptimo no coherente se ha observado que su desempeño es optimizado seleccionando el número de intervalos de observación impar y haciendo la decisión sobre el símbolo a la mitad [2]. Los resultados numéricos en la probabilidad de error son similares a los mostrados en la detección coherente; ésto es, se obtiene una ganancia de 2 dB incrementando el tiempo de correlación de 1 a 3 intervalos de símbolo y de 3 dB incrementándolo de 1 a 5 intervalos.

4.2. Receptores Óptimos de Viterbi

En esta sección se especifican los receptores óptimos del tipo de estimación de secuencias por máxima verosimilitud [1]. Para esta sección se asumirá que el pulso $g(t)$ tiene longitud finita LT ; es decir, $g(t)=0$ para $t<0$ y $t>LT$. Entonces, $q(t)=0$ para $t<0$ y constante para $t>LT$. Para pulsos positivos $g(t)$, $q(LT)=1/2$. La fase con la información se puede escribir como:

$$\begin{aligned}\varphi(t, \alpha) &= 2\pi h \sum_{i=-\infty}^n \alpha_i q(t - iT) \\ &= 2\pi h \sum_{i=n-L+1}^n \alpha_i q(t - iT) + h\pi \sum_{i=-\infty}^{n-L} \alpha_i \quad nT \leq t \leq (n+1)T \\ &= \theta(t, \alpha) + \theta_n\end{aligned}\quad (4.17)$$

La fase $\varphi(t, \alpha)$ está definida entonces, además de por símbolo actual, por el vector de estado correlativo $(\alpha_{n-1}, \alpha_{n-2}, \dots, \alpha_{n-L+1})$ y por el estado de fase θ_n .

Como se explicó anteriormente, el número de estados correlativos es finito e igual a M^{L-1} . Para índices de modulación racionales, de la forma $h=2k/p$, se tiene también un número finito p de estados de fase:

$$\Theta_s = \left\{ 0, \frac{2k}{p}\pi, \frac{4k}{p}\pi, \dots, \frac{2k(p-1)}{p}\pi \right\} \quad (4.18)$$

El estado total se define por el vector:

$$\sigma_n = (\theta_n, \alpha_{n-1}, \alpha_{n-2}, \dots, \alpha_{n-L+1}) \quad (4.19)$$

y existen pM^{L-1} estados totales distintos. Así, suponiendo el estado del modulador en $t=nT$ como σ_n , el efecto del símbolo nuevo en el intervalo de tiempo $nT \leq t \leq (n+1)T$ es cambiar el estado de σ_n por σ_{n+1} . Éste último está dado por:

$$\sigma_{n+1} = (\theta_{n+1}, \alpha_n, \alpha_{n-1}, \dots, \alpha_{n-L+2}), \quad (4.20)$$

siendo

$$\theta_{n+1} = \theta_n + \pi h \alpha_{n-L+1} \quad (4.21)$$

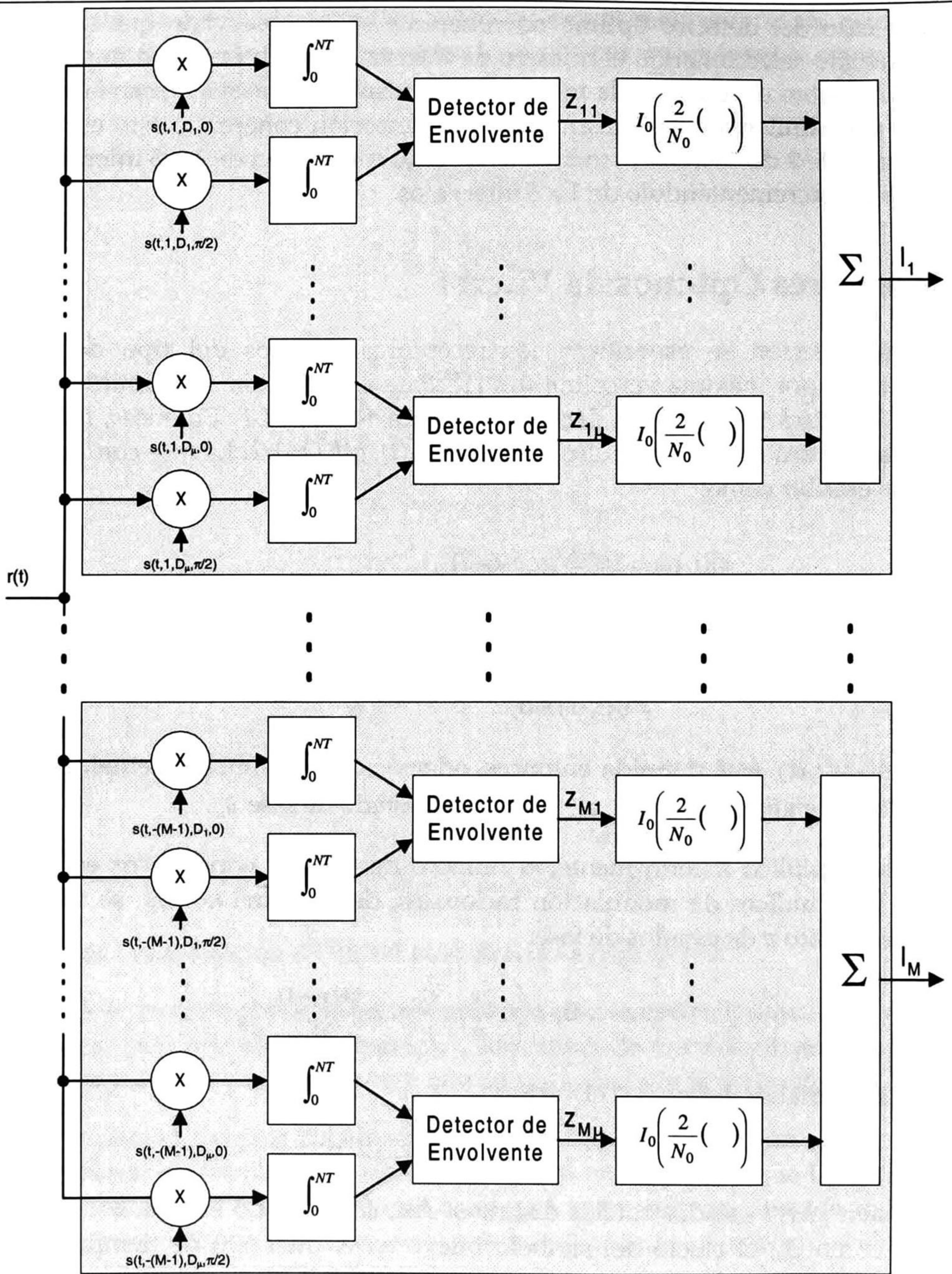


Fig. 4.5 Receptor no coherente óptimo para CPM.

De las expresiones (4.20) y (4.21) se puede establecer un diagrama de trellis para caracterizar un sistema de CPM con índice de modulación racional. El receptor óptimo para las señales de un sistema tal es un correlacionador seguido de un detector de secuencias de máxima verosimilitud que haga una búsqueda a través de todas las rutas del trellis, buscando aquella ruta que resulte en la mínima distancia euclidiana. El algoritmo de Viterbi es un método eficiente para realizar esta búsqueda.

El receptor observa la señal

$$r(t) = s(t, \alpha) + n(t) \quad (4.22)$$

donde el ruido $n(t)$ es blanco y Gaussiano. Entonces, el receptor por estimación de secuencias de máxima verosimilitud trata de maximizar la función de verosimilitud logarítmica dada por:

$$\log_e [p(r(t)|\alpha)] \approx - \int_{-\infty}^{\infty} [r(t) - s(t, \alpha)]^2 dt \quad (4.23)$$

Esta expresión se obtiene del desarrollo para obtener la probabilidad a posteriori de la señal recibida con ruido blanco Gaussiano aditivo, lo que da la expresión de la distancia entre las señales. Ésto es equivalente a maximizar la correlación dada por:

$$J(\alpha) = \int_{-\infty}^{\infty} r(t)s(t, \alpha) dt \quad (4.24)$$

En principio, la operación definida en (4.24) es la base para un receptor por correlación. En éste, todas las posibles señales transmitidas $s(t, \alpha)$ son correlacionadas con la señal recibida y la secuencia de datos α que maximiza la correlación es elegida como los datos recibidos. Cabe decir que esta estructura no es prácticamente realizable, ni siquiera para ráfagas de datos razonablemente pequeñas. Se puede sin embargo definir:

$$J_n(\alpha) = \int_{-\infty}^{(n+1)T} r(t)s(t, \alpha) dt \quad (4.25)$$

Esta expresión se puede escribir de la siguiente forma:

$$J_n(\alpha) = J_{n-1}(\alpha) + Z_n(\alpha) \quad (4.26)$$

siendo

$$\begin{aligned} Z_n(\alpha) &= \int_{nT}^{(n+1)T} r(t)s(t, \alpha) dt \\ &= \int_{nT}^{(n+1)T} r(t) \cos[\omega_0 t + \varphi(t, \alpha)] dt \end{aligned} \quad (4.27)$$

Es posible entonces calcular la función $J_n(\alpha)$ recursivamente usando (4.26) y la métrica $Z_n(\alpha)$ dada por (4.27). Esta métrica es la correlación entre la señal recibida y un estimado de dicha señal en el n-ésimo intervalo.

El Algoritmo de Viterbi es un procedimiento recursivo para elegir aquella secuencia que maximiza la función de verosimilitud logarítmica hasta el n-ésimo intervalo. Así, el receptor debe calcular las métricas $Z_n(\alpha_n, \theta_n)$ para todas las M^L secuencias posibles $\alpha_n = \{\alpha_n, \alpha_{n-1}, \dots, \alpha_{n-L+1}\}$ y para todos los p posibles estados de fase θ_n . Con ésto, se tendrán pM^L diferentes valores de Z_n . De esta forma, se puede implementar el detector con un banco de pM^L bloques como el mostrado en la Fig. 4.6 [2].

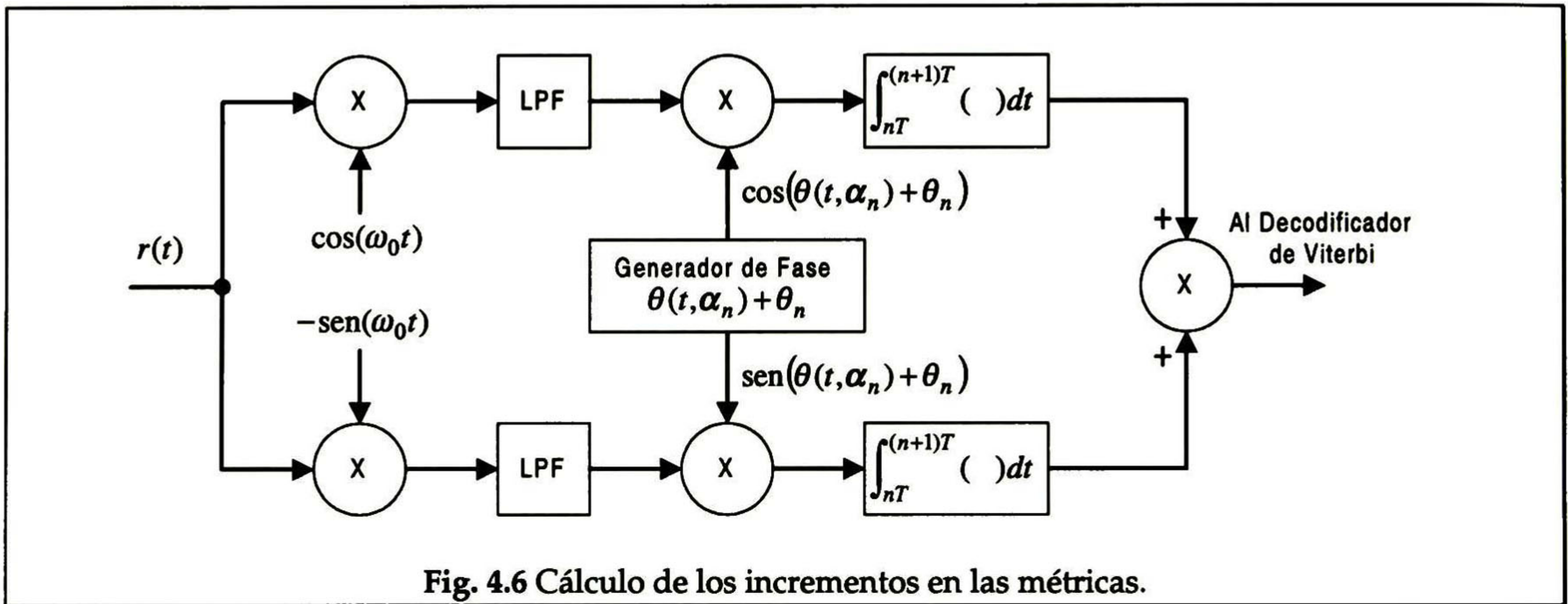


Fig. 4.6 Cálculo de los incrementos en las métricas.

Una forma alterna de implementación que puede reducir la cantidad de bloques se logra con el procedimiento descrito a continuación [1]. Definiendo la señal transmitida y el ruido que se le añade como:

$$s(t) = \sqrt{\frac{2E}{T}} [I(t) \cos \omega_0 t - Q(t) \text{sen} \omega_0 t] \tag{4.28}$$

$$n(t) = x(t) \cos \omega_0 t - y(t) \text{sen} \omega_0 t ,$$

Los componentes en cuadratura recibidos son entonces

$$\hat{I}(t) = \left[\sqrt{\frac{2E}{T}} I(t) + x(t) \right] \tag{4.29}$$

y

$$\hat{Q}(t) = \left[\sqrt{\frac{2E}{T}} Q(t) + y(t) \right]; \tag{4.30}$$

por tanto:

$$r(t) = \hat{I}(t) \cos \omega_0 t - \hat{Q}(t) \text{sen} \omega_0 t \quad (4.31)$$

Sustituyendo esta última expresión en (4.27) y, después de aplicar identidades trigonométricas elementales y eliminar los términos de doble frecuencia, se tendrá que:

$$\begin{aligned} Z_n(\alpha_n, \theta_n) = & \cos(\theta_n) \int_{nT}^{(n+1)T} \hat{I}(t) \cos[\theta(t, \alpha_n)] dt \\ & + \cos(\theta_n) \int_{nT}^{(n+1)T} \hat{Q}(t) \text{sen}[\theta(t, \alpha_n)] dt \\ & + \text{sen}(\theta_n) \int_{nT}^{(n+1)T} \hat{Q}(t) \cos[\theta(t, \alpha_n)] dt \\ & - \text{sen}(\theta_n) \int_{nT}^{(n+1)T} \hat{I}(t) \text{sen}[\theta(t, \alpha_n)] dt \end{aligned} \quad (4.32)$$

Entonces, para calcular cada métrica se requerirían 4 correlacionadores o filtros acoplados, cada uno con un multiplicador aparte. Pero si se tiene un procesador que esté almacenando los datos, que esté generando secuencialmente los valores de $\cos(\theta_n)$ y $\text{sen}(\theta_n)$, y que esté efectuando las multiplicaciones, entonces solamente se requerirían en total $4M^L$ correlacionadores o filtros acoplados en lugar de los $4pM^L$ que se requerían originalmente. Obviamente, si se usan filtros acoplados en lugar de correlacionadores sus respuestas al impulso estarán dadas por:

$$\begin{aligned} h_c(t, \alpha_n) = & \begin{cases} \cos \left[2\pi h \sum_{j=-L+1}^0 \alpha_j q((1-j)T - t) \right] \\ 0 \end{cases} \\ & \text{para } t \text{ fuera de } [0, T] \\ h_s(t, \alpha_n) = & \begin{cases} \text{sen} \left[2\pi h \sum_{j=-L+1}^0 \alpha_j q((1-j)T - t) \right] \\ 0 \end{cases} \\ & \text{para } t \text{ fuera de } [0, T] \end{aligned} \quad (4.33)$$

El diagrama a bloques del receptor que utiliza estos filtros acoplados se muestra en la Fig. 4.7. El número de filtros puede reducirse a la mitad observando que para cada secuencia de símbolos existe una secuencia correspondiente con signo contrario; entonces, para cada métrica existe también una métrica correspondiente de signo contrario.

El algoritmo de Viterbi es óptimo en el sentido de que obtiene el estimado de máxima verosimilitud de la secuencia de los estados. Dado que una secuencia específica de transiciones entre estados da una única secuencia de símbolos, entonces el algoritmo de Viterbi obtiene también un estimado de máxima verosimilitud de la secuencia. Aquí cabe hacer notar que los detectores antes mencionados para CPFSK son también óptimos, pero con otro criterio. Éstos encuentran, utilizando información de varios símbolos recibidos, el estimado óptimo de uno de ellos, más no el de la secuencia.

En su forma básica, el algoritmo de Viterbi asume que todas las transiciones entre los estados tienen un primer estado común, y un estado final también común. La decisión

de la secuencia transmitida es retrasada hasta que se recibe toda la secuencia. Las métricas se van actualizando sucesivamente. Cuando la secuencia completa es recibida, todas las trayectorias candidatas terminan en el mismo estado, y la que da la métrica óptima es la salida del receptor.

Como lo anterior no es prácticamente realizable lo que se hace en cada transición es guardar las trayectorias solamente hasta cierto punto o longitud denominada memoria de trayectoria N_T . Dado este retraso, se obliga una decisión sobre el símbolo más antiguo, ya sea porque se tiene sólo una trayectoria sobreviviente, por ser la que tiene la mejor métrica acumulada o por algún otro criterio. La elección de esta longitud de trayectoria debe hacerse tomando en cuenta la distancia mínima entre rutas del trellis del sistema. Esta distancia mínima la determinan parámetros como el índice de modulación y la longitud en intervalos de señalización del pulso $g(t)$.

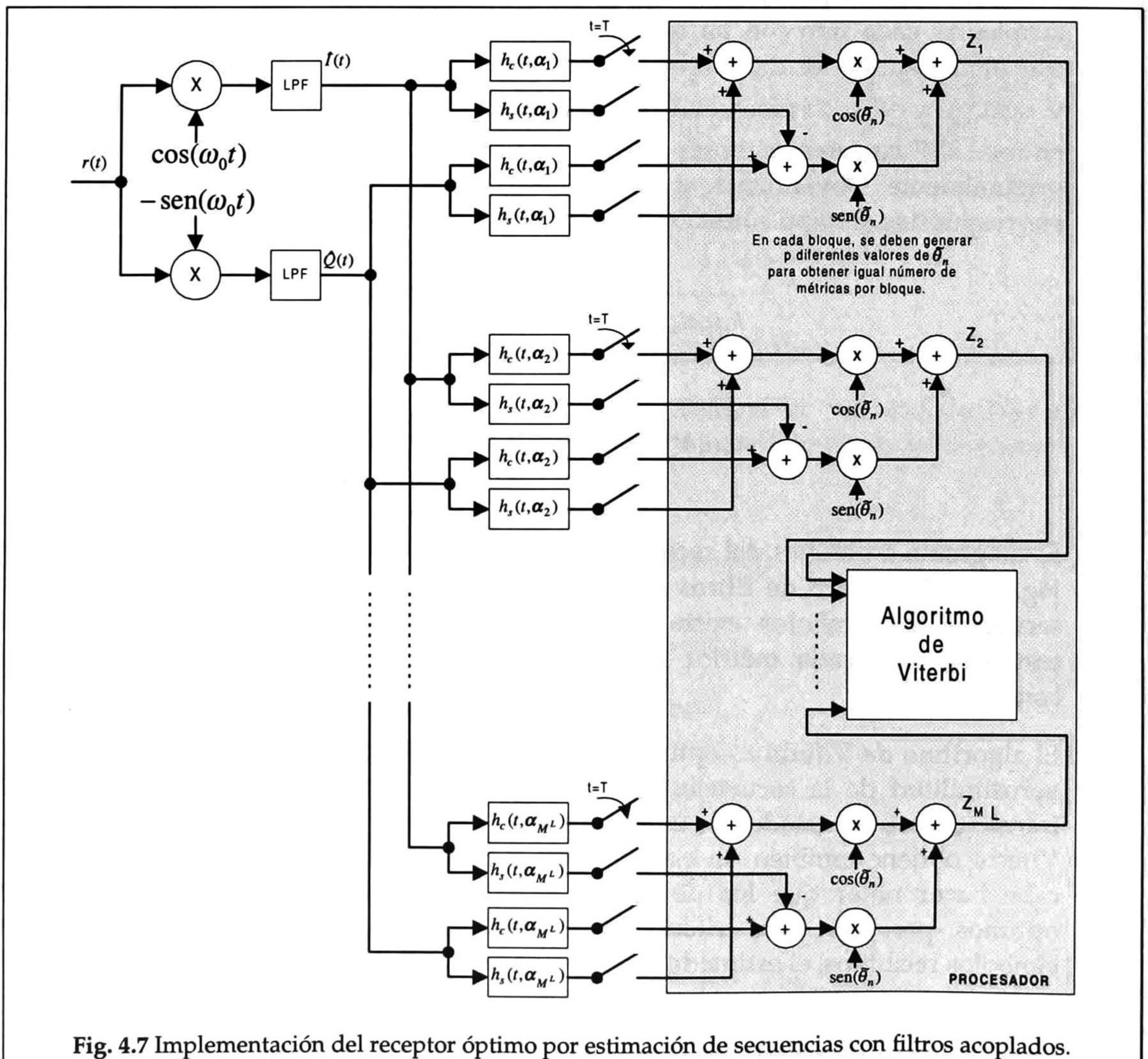


Fig. 4.7 Implementación del receptor óptimo por estimación de secuencias con filtros acoplados.

Para evaluar el desempeño de las señales CPM logrado con estimación de secuencias de máxima verosimilitud se debe determinar la distancia euclidiana mínima entre trayectorias del trellis que se separan de un nodo en $t=0$ y se vuelven a juntar después de un tiempo en el mismo nodo. La distancia entre estas dos trayectorias en el trellis está relacionada con las señales correspondientes como se demostrará a continuación.

Suponiendo dos señales, $s_i(t)$ y $s_j(t)$, correspondientes a dos trayectorias de fase $\Phi(t, \alpha_i)$ y $\Phi(t, \alpha_j)$. Las secuencias α_i y α_j deben ser diferentes en su primer símbolo. Entonces, la distancia euclidiana entre estas dos señales sobre un intervalo de longitud NT está dada por:

$$\begin{aligned}
 d_{ij}^2 &= \int_0^{NT} (s_i(t) - s_j(t))^2 dt \\
 &= 2NE - 2 \int_0^{NT} s_i(t) s_j(t) dt \\
 &= 2NE - 2 \frac{2E}{T} \int_0^{NT} \cos(\omega_0 t + \Phi(t, \alpha_i)) \cos(\omega_0 t + \Phi(t, \alpha_j)) dt \\
 &= 2NE - \frac{2E}{T} \int_0^{NT} \cos(\Phi(t, \alpha_i) - \Phi(t, \alpha_j)) dt \\
 &= \frac{2E}{T} \int_0^{NT} \{1 - \cos(\Phi(t, \alpha_i) - \Phi(t, \alpha_j))\} dt
 \end{aligned} \tag{4.34}$$

Esta ultima expresión demuestra que la distancia euclidiana está relacionada con la diferencia de fases entre trayectorias en el trellis.

La distancia d_{ij}^2 se puede expresar en términos de la energía de bit, con:

$$\begin{aligned}
 d_{ij}^2 &= 2E_b \delta_{ij}^2 \\
 \delta_{ij}^2 &= \frac{\log_2 M}{T} \int_0^{NT} \{1 - \cos(\Phi(t, \alpha_i) - \Phi(t, \alpha_j))\} dt
 \end{aligned} \tag{4.35}$$

La probabilidad de error en un símbolo para CPM está dominado por el término correspondiente a la distancia euclidiana mínima, y se puede expresar como:

$$P_M = K_{\delta_{\min}} Q \left(\sqrt{\frac{E_b}{N_0} \delta_{\min}^2} \right) \tag{4.36}$$

donde

$$\delta_{\min}^2 = \lim_{N \rightarrow \infty} \left[\min_{i, j} (\delta_{ij}^2) \right]$$

en donde $K_{\delta_{\min}}$ es el número de rutas que tiene esta distancia mínima.

4.3. Receptores Simples No Coherentes

La mayor parte del tratamiento analítico de CPM se refiere a receptores coherentes; pero, en las aplicaciones prácticas se prefieren los receptores no coherentes por ser éstos más simples y baratos. Cabe decir, sin embargo, que son más difíciles de analizar por su naturaleza no lineal.

4.3.1. Detección Diferencial

Para el análisis del detector diferencial se considera un sistema que está usando transmisión binaria con respuesta parcial [1]. En la Fig. 4.8 se muestra el diagrama a bloques de éste. La señal recibida se filtra para limitarla en banda. Como el filtro usado es lineal e invariante en el tiempo, a su salida se tiene una señal con envolvente cambiante en tiempo y fase distorsionada, la cual está dada por:

$$r_{IF}(t) = \sqrt{\frac{2E}{T}} A(t, \alpha) \cos[\omega_0 t + \Psi(t, \alpha)] + n'(t), \tag{4.37}$$

donde $n'(t)$ es el proceso resultante del filtrado del ruido. Este ruido filtrado se puede representar con sus componentes en cuadratura:

$$n'(t) = \sqrt{2}n'_c(t) \cos[\omega_0 t + \Psi(t, \alpha)] - \sqrt{2}n'_s(t) \text{sen}[\omega_0 t + \Psi(t, \alpha)] \tag{4.38}$$

donde $n'_c(t)$ y $n'_s(t)$ son procesos aleatorios Gaussianos pasabajas.

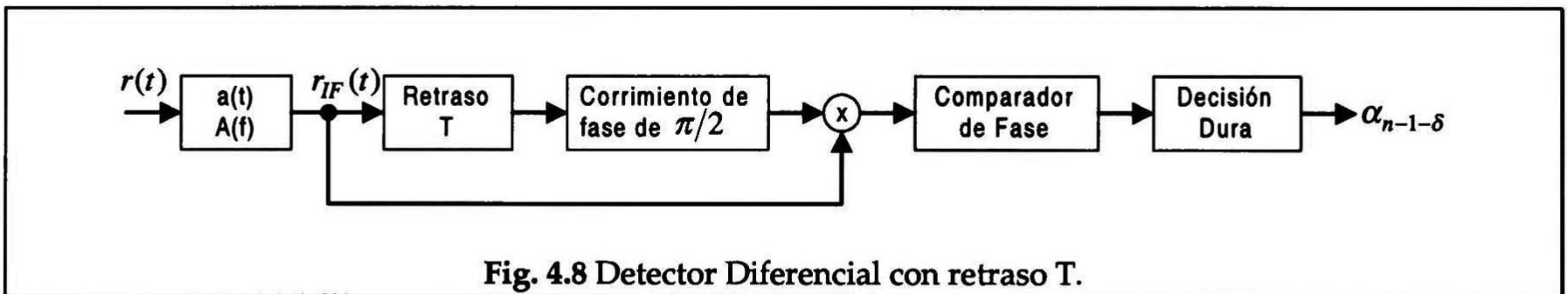


Fig. 4.8 Detector Diferencial con retraso T.

La señal $r_{IF}(t)$ se puede representar polarmente por:

$$r_{IF}(t) = R(t, \alpha) \cos[\omega_0 t + \Psi(t, \alpha) + \eta(t)], \tag{4.39}$$

donde:

$$R(t, \alpha) = \sqrt{2 \left\{ \left[\sqrt{\frac{E}{T}} A(t, \alpha) + n'_c(t) \right]^2 + n'^2_s(t) \right\}} \tag{4.40}$$

$$\eta(t) = -\arctan \left\{ \frac{n'_s(t)}{\sqrt{\frac{E}{T}} A(t, \alpha) + n'_c(t)} \right\} \quad (4.41)$$

El detector diferencial multiplica a $r_{IF}(t)$ por una versión de sí misma retrasada en T segundos y corrida en fase 90° ; por lo que da como resultado, ignorando los términos con el doble de la frecuencia de la portadora, la siguiente expresión.

$$y(t) = \frac{R(t, \alpha)R(t-T, \alpha)}{2} \text{sen}[\omega_0 T + \Delta\Psi(t, T, \alpha)], \quad (4.42)$$

donde:

$$\Delta\Psi(t, T, \alpha) = [\Psi(t, \alpha) - \Psi(t-T, \alpha) + \eta(t) - \eta(t-T)] \text{mod } 2\pi \quad (4.43)$$

Si se asume que $f_0 = k/T$, es decir, que $\omega_0 T = 2k\pi$, entonces:

$$y(t) = \frac{R(t, \alpha)R(t-T, \alpha)}{2} \text{sen}[\Delta\Psi(t, T, \alpha)] \quad (4.44)$$

Esta señal es muestreada en el instante en que se observa un ojo abierto para el conjunto de diferencias de fase dadas por:

$$\Psi(t, \alpha) - \Psi(t-T, \alpha)$$

Este instante es:

$$t_0 = \begin{cases} nT & \text{si } L + L_A \text{ es impar} \\ (n + 1/2)T & \text{si } L + L_A \text{ es par} \end{cases} \quad (4.45)$$

donde L_A es la longitud de la respuesta al impulso del filtro $a(t)$, la cual se mide en intervalos de señalización. Entonces, con la muestra tomada del punto de mayor abertura del ojo se toma la decisión, con un cierto retraso, sobre el símbolo recibido.

En las Figs. 4.9 y 4.10 se muestran las probabilidades de error obtenidas para este tipo de receptor con distintos tipos de filtros $a(t)$. Estas curvas se obtuvieron de la literatura consultada [1]. En estas figuras, cada curva se denota por el tipo de pulso usado en la demodulación y por la característica del filtro de entrada (*esquema transmitido/tipo de filtro*). Esta detección diferencial se acopla de mejor manera a esquemas con un índice de modulación alrededor de $h=1/2$. Con índices de modulación mayores, la diferencia de fases sobre un intervalo se acerca más a π , y, con esto, los errores se incrementan.

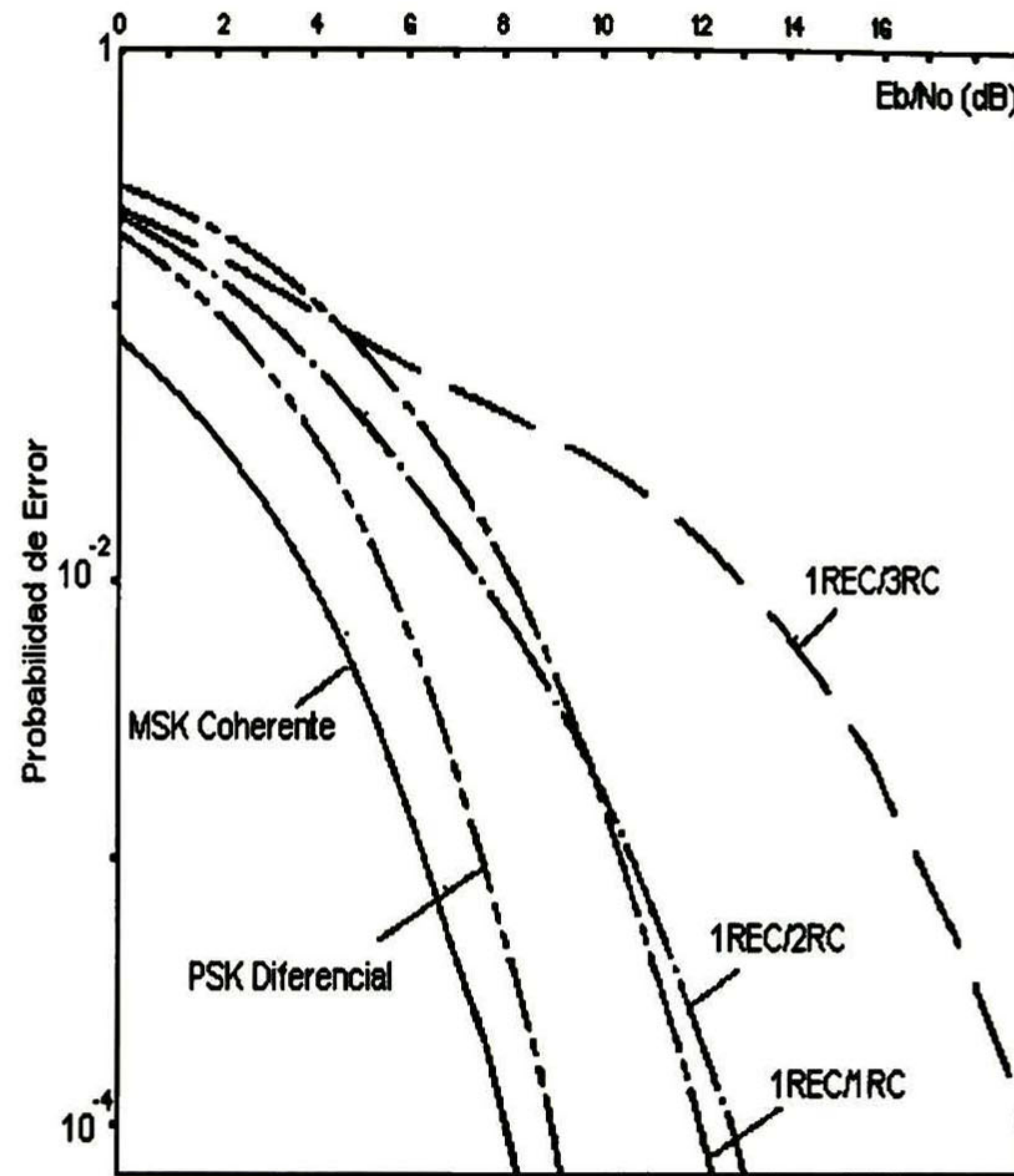


Fig. 4.9 Desempeño de 1REC filtrado con detección diferencial con $h=0.5$

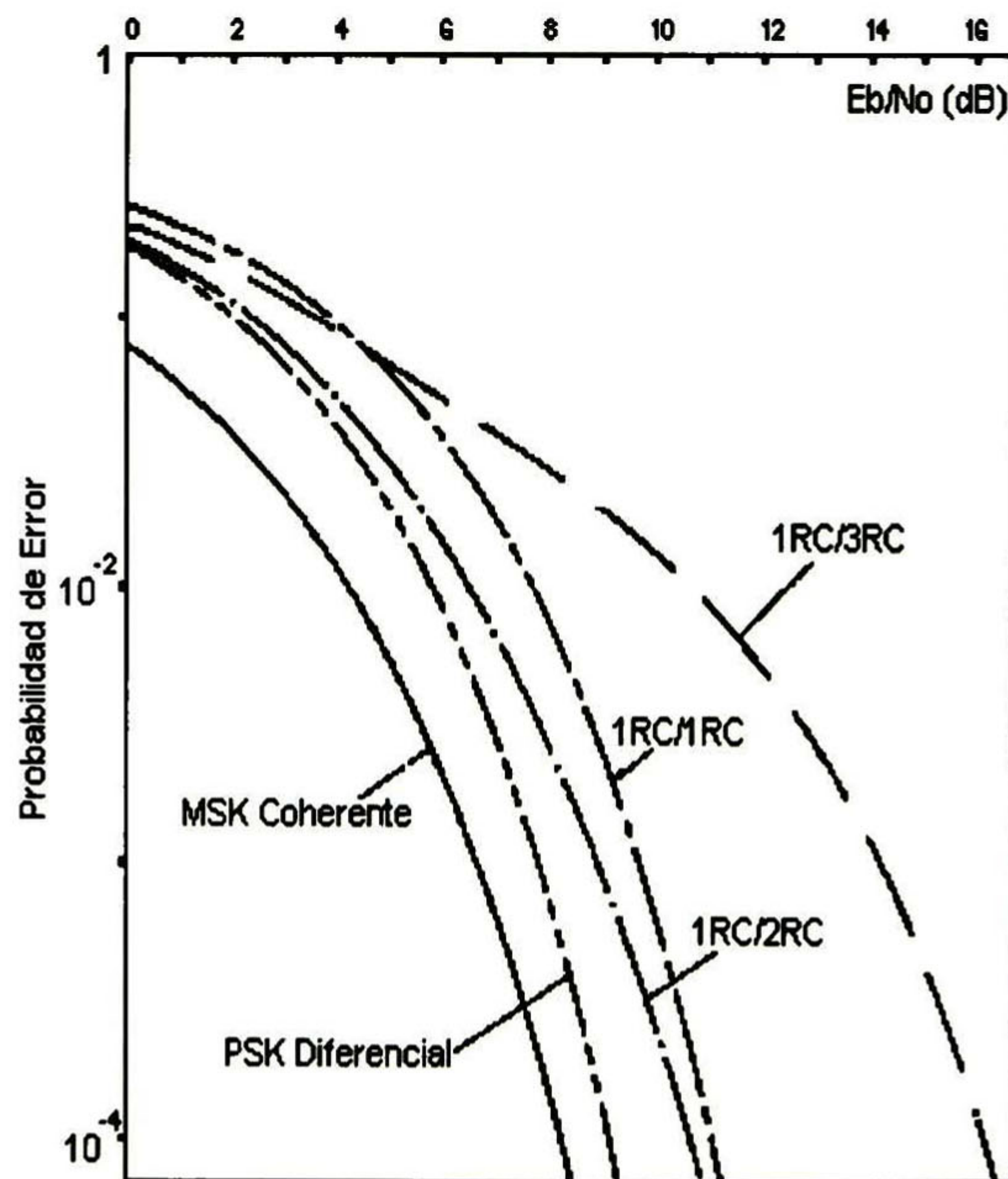


Fig. 4.10 Desempeño de 1RC filtrado con detección diferencial con $h=0.5$

4.3.2. Detección por Discriminador

Al igual que en la sección anterior, para el análisis del detector por discriminador se considera un sistema que está usando transmisión binaria con respuesta parcial [1]. En la Fig. 4.11 se muestra el diagrama a bloques de éste. El filtro $A(f)$ sirve para limitar en banda la señal recibida. Igualmente que en el caso del detector diferencial, se tiene que la señal después de este filtro está dada por:

$$\begin{aligned} r_{IF}(t) &= \sqrt{\frac{2E}{T}} A(t, \alpha) \cos[\omega_0 t + \Psi(t, \alpha)] + n'(t) \\ &= R(t, \alpha) \cos[\omega_0 t + \Psi(t, \alpha) + \eta(t)] \end{aligned} \quad (4.46)$$



Fig. 4.11 Detector por Discriminador.

El par de limitador-discriminador es un dispositivo no lineal que responde solamente a variaciones en la frecuencia instantánea [12]. Este par de dispositivos se modela matemáticamente suponiendo que cuando la entrada es:

$$r(t) = a(t) \cos[\omega_0 t + \varphi(t)], \quad (4.47)$$

la salida estará dada por:

$$s(t) = \frac{d\varphi(t)}{dt} \quad (4.48)$$

La operación de tal tipo de dispositivos se puede representar como otro dispositivo que extraiga la fase $\varphi(t)$ y después la deriva; es decir, que sea seguido por un dispositivo con función de transferencia $H(f) = j2\pi f$. Por lo tanto, la salida del discriminador en la Fig. 4.11 estará dada por la siguiente ecuación:

$$r_d(t) = \frac{d}{dt} \Psi(t, \alpha) + \frac{d}{dt} \eta(t) \quad (4.49)$$

Entonces aplicando esta señal $r_d(t)$ como entrada al integrador:

$$\begin{aligned} r_i(t) &= \int_{t-T}^t \left[\frac{d}{dt} \Psi(\tau, \alpha) + \frac{d}{dt} \eta(\tau) \right] d\tau \\ &= \Psi(t, \alpha) - \Psi(t-T, \alpha) + \eta(t) - \eta(t-T) \end{aligned} \quad (4.50)$$

La expresión (4.48) es básicamente el mismo resultado obtenido para el receptor diferencial. La decisión del símbolo recibido se lleva a cabo entonces en la misma forma que para este último.

En las Figs. 4.12, 4.13 y 4.14 se muestran curvas de probabilidad de error para la detección por discriminador. Estas gráficas fueron obtenidas de la literatura consultada [1]. La notación usada en las gráficas es la misma que la usada en las curvas de la sección anterior. Como se puede ver, la probabilidad de error con esquemas de $h=1/2$ es muy cercana a los resultados obtenidos con el detector diferencial; además, la probabilidad de error puede ser disminuida si se incrementa el índice de modulación. Se han encontrado buenas combinaciones de esquemas de modulación y de detectores (tanto diferenciales como por discriminador) en las cuales la pérdida en la relación señal a ruido es sólo del orden de 2dB, para una probabilidad de error de 10^{-6} , comparada con la detección coherente de MSK.

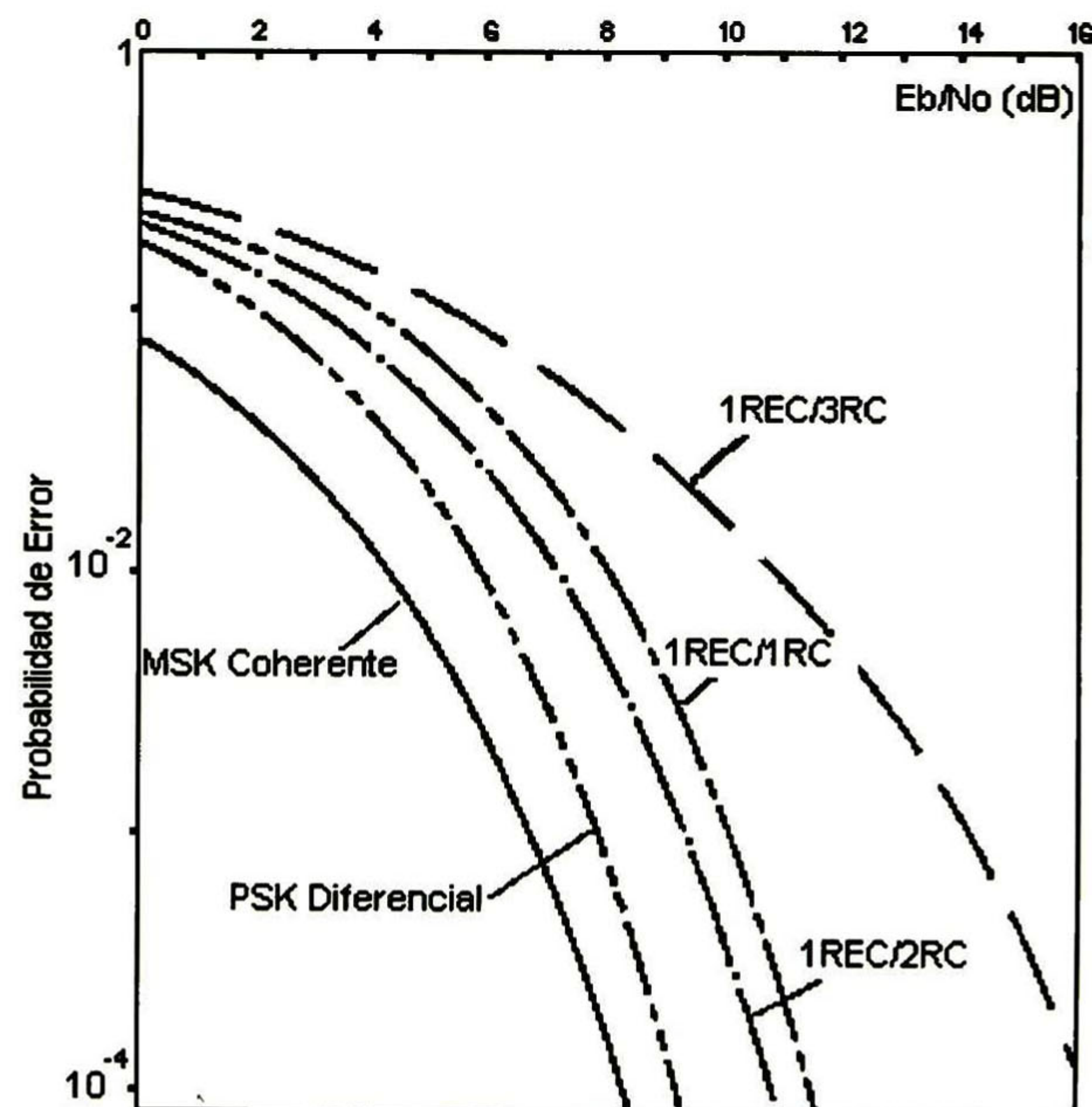


Fig. 4.12 Desempeño de 1REC filtrado con detección por discriminador con $h=0.718$

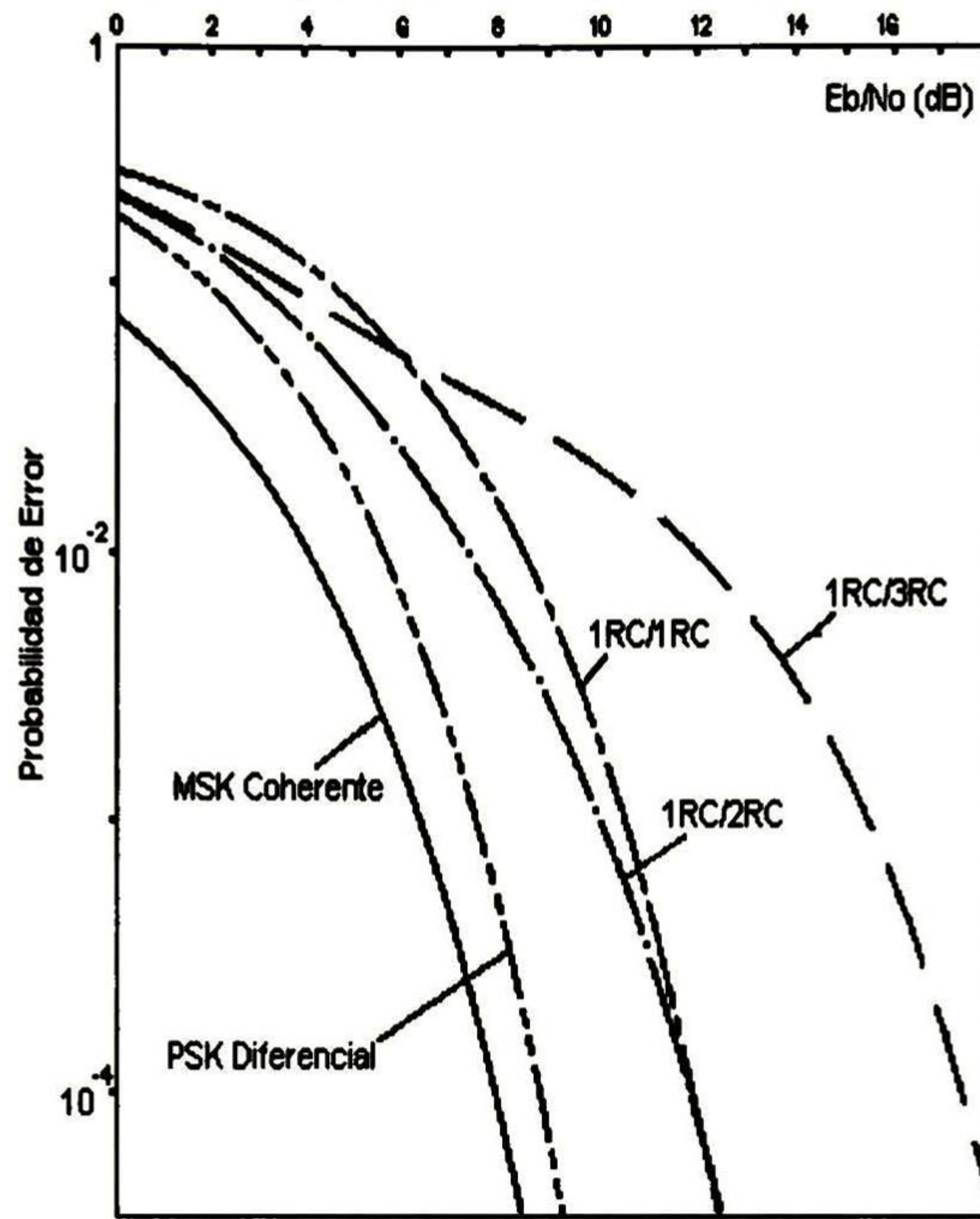


Fig. 4.13 Desempeño de 1RC filtrado con detección por discriminador con $h=0.4$

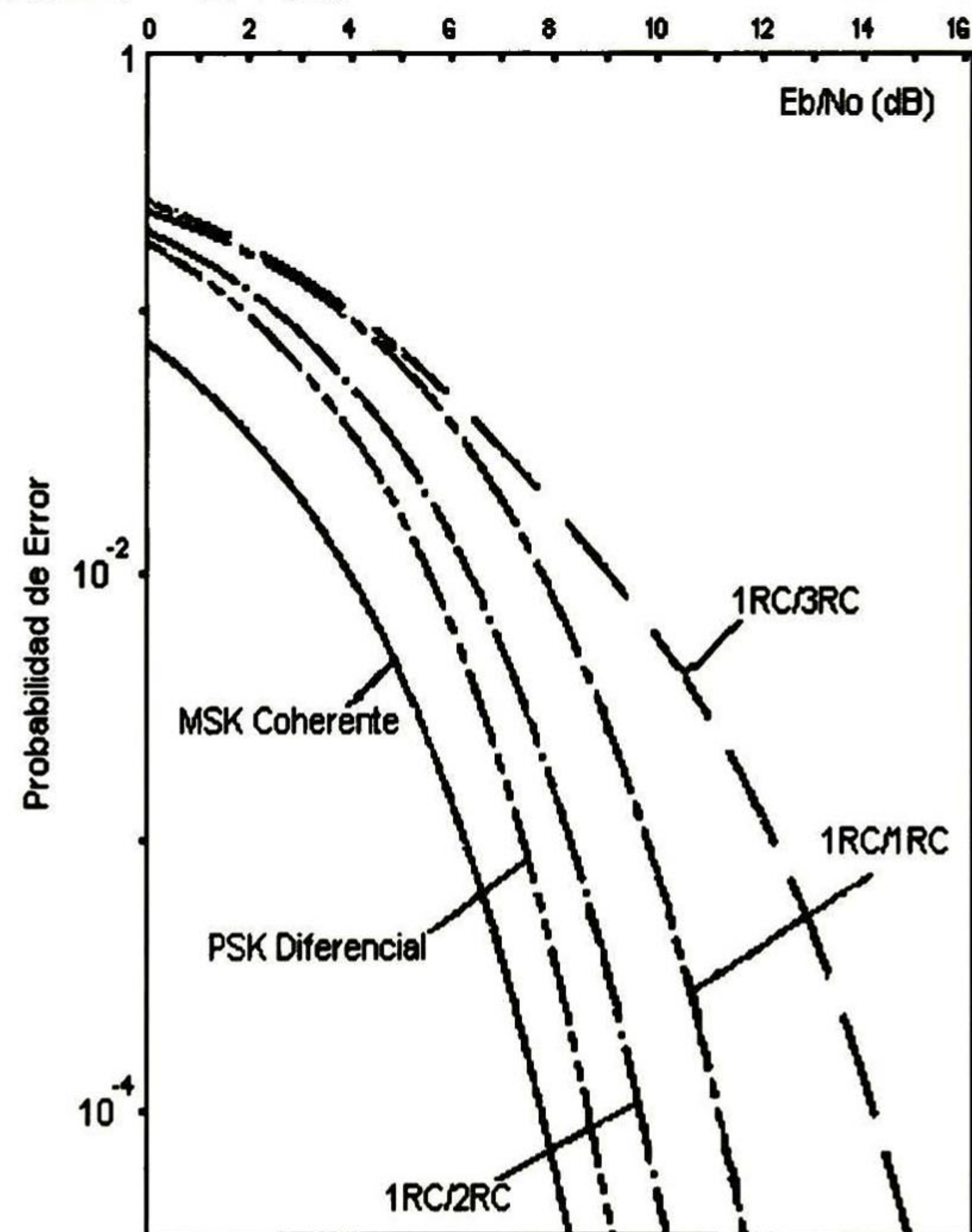
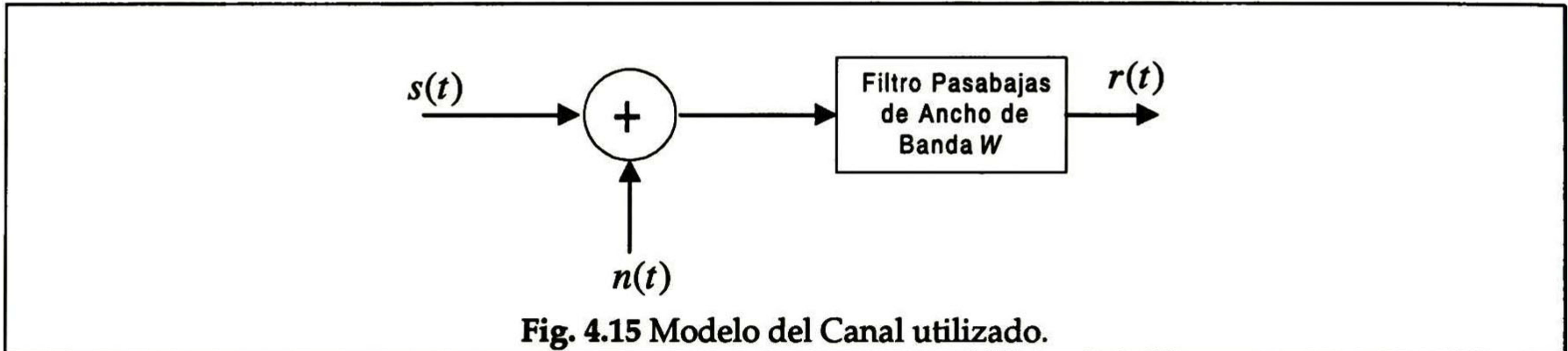


Fig. 4.14 Desempeño de 1RC filtrado con detección por discriminador con $h=0.62$

4.4. Simulaciones de detectores por estimación de secuencias.

El receptor por estimación de secuencias que utiliza el Algoritmo de Viterbi, es del que menos resultados se presentan en la literatura especializada. Por este motivo, se implementaron modelos de éste y para probar como se desempeña y compararlo con los otros esquemas de detectores. Las simulaciones se llevaron a cabo tomando el modelo del canal de ruido blanco gaussiano aditivo ya explicado en el Capítulo 1. Todas las simulaciones se llevaron a cabo en banda base considerado el esquema mostrado en la Fig. 4.15.



Para tener una base de comparación del desempeño de los sistemas, se determinará la razón señal a ruido a la salida de este canal que es la de entrada al demodulador utilizado.

De la expresión (2.1), se sigue que la potencia de la señal recibida durante un intervalo de señalización es:

$$P_s = \frac{E}{T} \quad (4.51)$$

La señal CPM en banda base tiene dos componentes en cuadratura. El ruido blanco tiene componentes en un número infinito de dimensiones, incluidas las dos en que se descompone la señal de interés. Pero sabemos también por el teorema de la irrelevancia [12], que el ruido en las dimensiones en que la señal no tiene componentes puede ser ignorado. Por ésto, en banda base se debe generar un ruido complejo que se suma a esta señal. El ruido adicionado tiene una densidad espectral de potencia bilateral de $\frac{1}{2}N_0$ sobre un ancho de banda B , que es mucho mayor que W , el ancho de banda del canal. Se sabe que esta densidad espectral de potencia está relacionada con la varianza del ruido por medio de:

$$\sigma_n^2 = \frac{1}{2} N_0 \quad (4.52)$$

Las componentes en cuadratura de este ruido tienen esta misma varianza. Sin embargo, al pasar por el filtro, dicha varianza de ruido cambia. Entonces, para las simulaciones se obtendrá la probabilidad de error en función de la relación señal a ruido a la salida del canal, la cual está definida por [13, 2]:

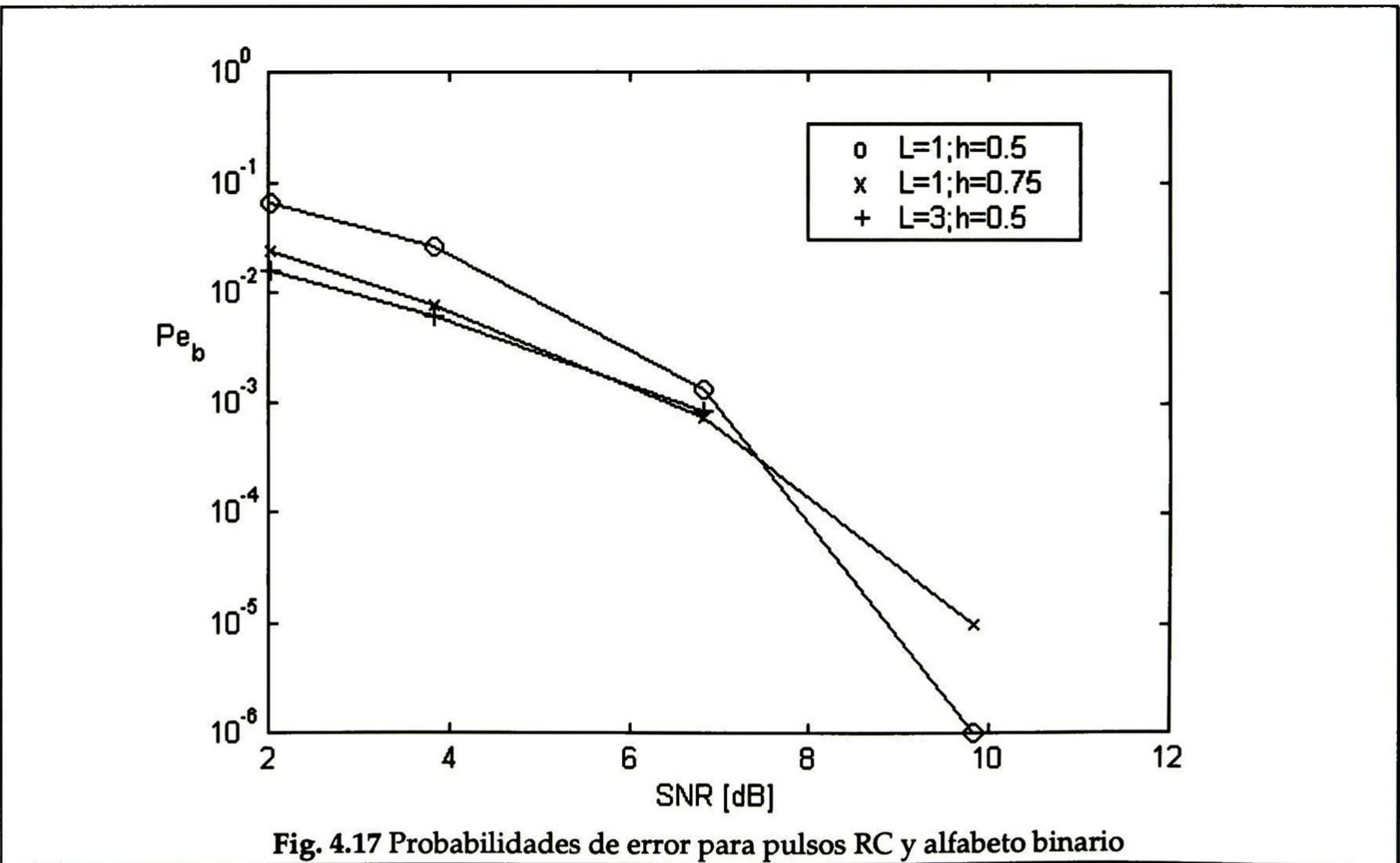
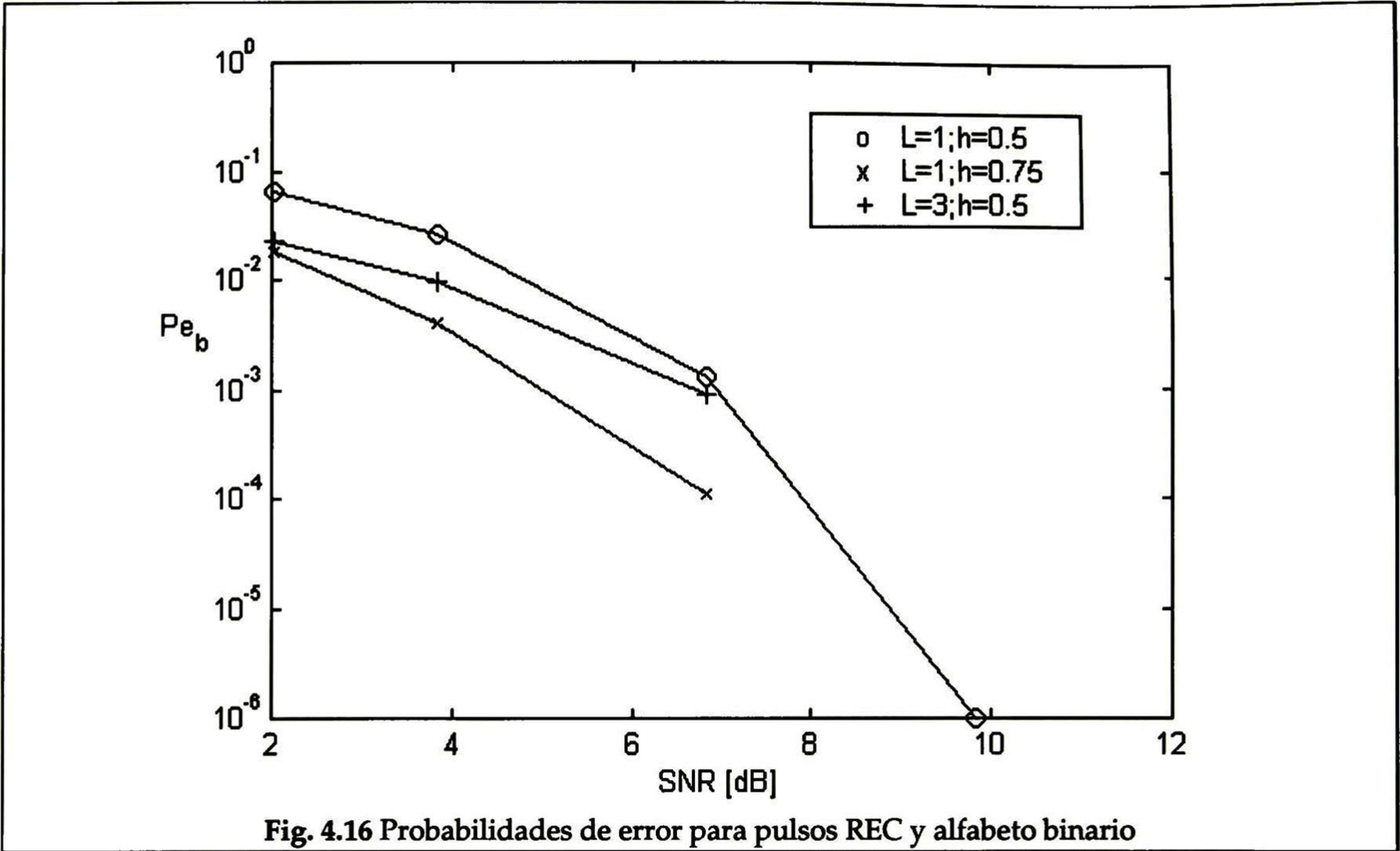
$$SNR_c = \frac{\text{Potencia Promedio de la señal de Información}}{\text{Potencia Promedio del Ruido a la salida del Canal}} \quad (4.53)$$

$$SNR_c = \frac{s^2(t)}{E\{n^2(t)\}}$$

Para las simulaciones de esta tesis se tomaron en cuenta las siguientes consideraciones:

- Se utilizaron 10,000 datos para generar la señal CPM en banda base.
- Como se utilizó $T=1$ y como en los programas en MatLab se generaron las formas de onda considerando la magnitud de la señal compleja igual a 1, se sigue que la energía de la señal en banda base es 1.
- La frecuencia de muestreo fue de $F_s=20$ Hz.
- Se generó un ruido gaussiano complejo con cierta potencia, donde sus componentes real e imaginaria tienen la varianza dada por (4.52) y son estadísticamente independientes.
- Se pasó el ruido por un filtro de ancho de banda semejante al ancho de banda de la señal (para los anchos de banda de la señal se utilizaron los resultados del capítulo anterior).
- Se calculó la varianza del nuevo ruido filtrado.
- Se adicionó el ruido filtrado ruido a la señal CPM.
- Se demoduló y se llevó a cabo un conteo de los errores para estimar la probabilidad de error.
- En todas las simulaciones, se utilizó una memoria de ruta para el algoritmo de Viterbi de 50 símbolos.
- Para calcular la SNR se utilizó (4.53).

Las figuras de la 4.16 a la 4.20 muestran los resultados obtenidos en las simulaciones. En los casos de alfabetos binarios se obtiene la probabilidad de error de bit; en cambio, en los casos de alfabetos cuaternarios se da la probabilidad de error de símbolo.



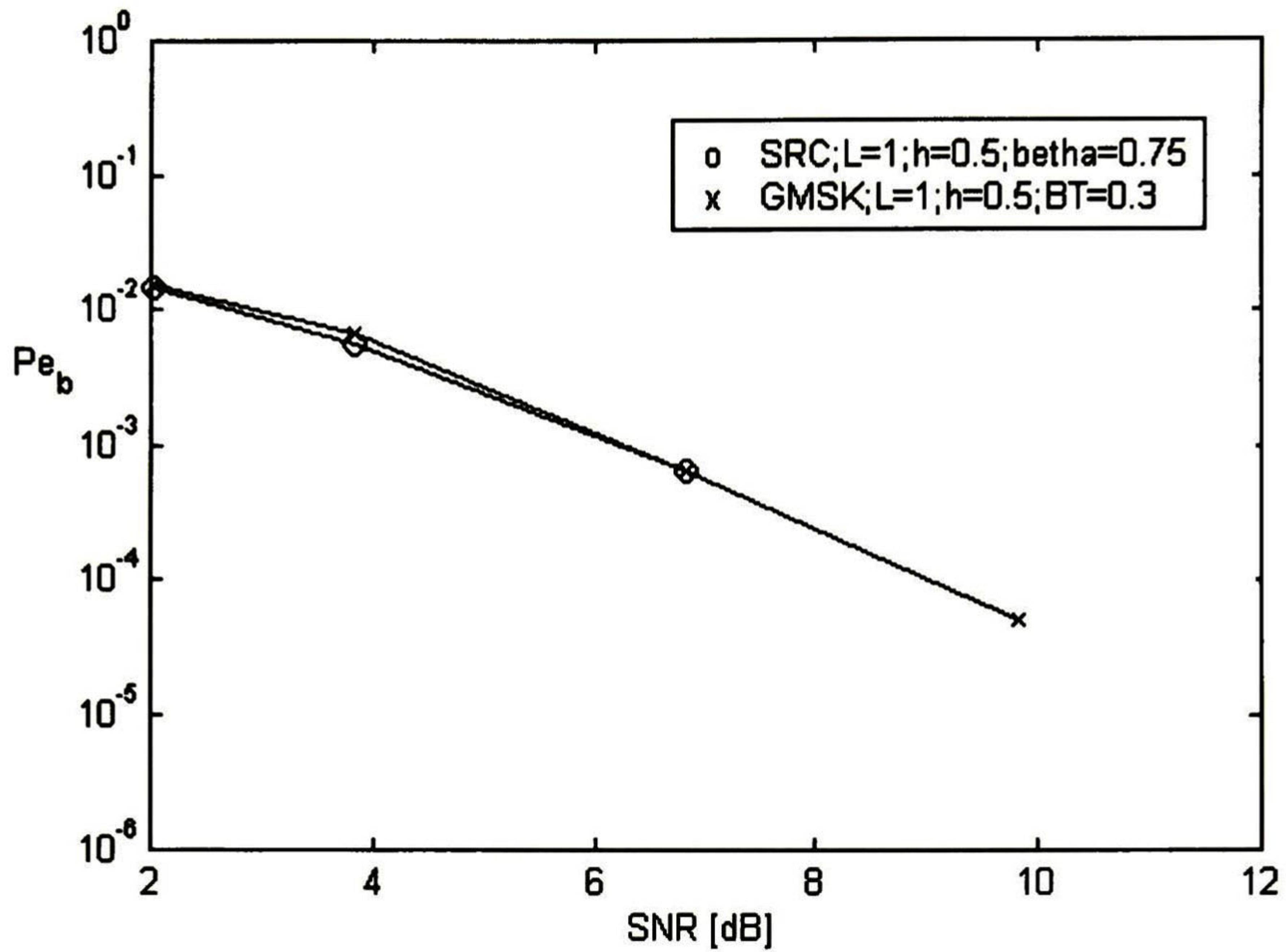


Fig. 4.18 Probabilidades de error para pulsos SRC y GMSK, con alfabeto binario

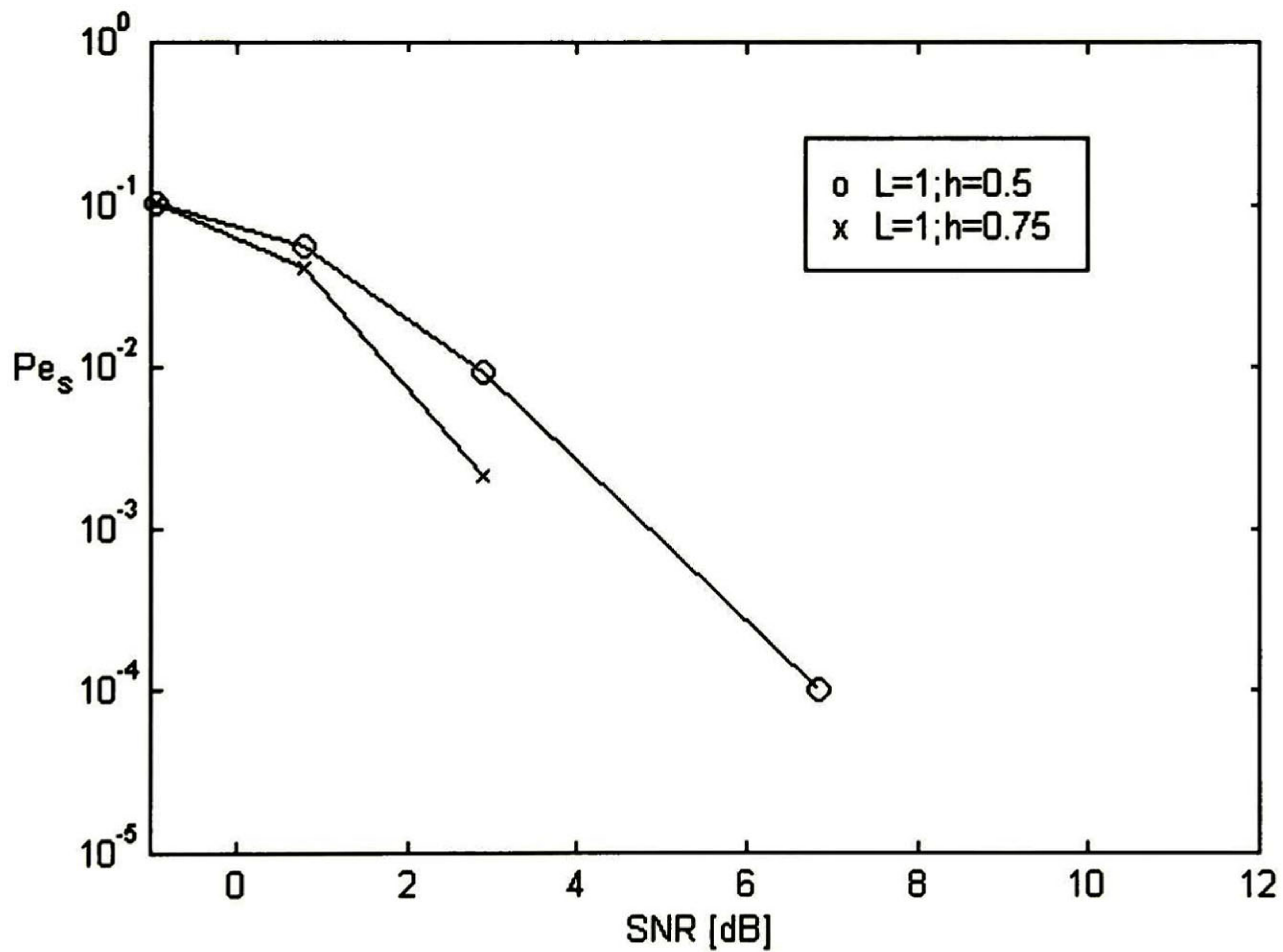
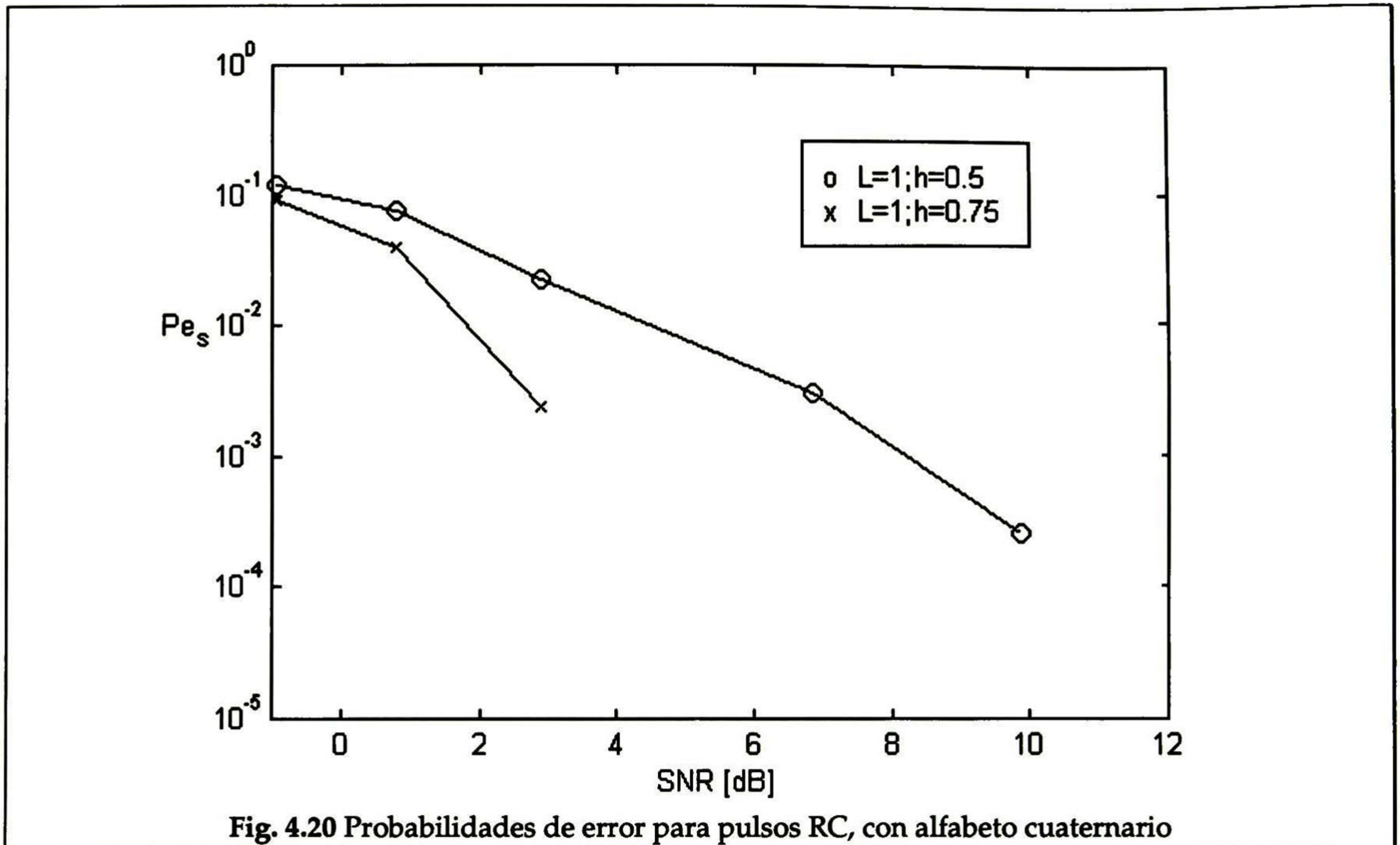


Fig. 4.19 Probabilidades de error para pulsos REC, con alfabeto cuaternario



4.5. Conclusiones de las simulaciones.

Se ha hecho el análisis de la operación de receptores de CPM, tanto coherentes como no coherentes. Se han proporcionado gráficas de desempeño para receptores coherentes de CPFSK binario, cuaternario y octal; para receptores no coherentes con los esquemas 1REC y 1RC con $h=0.5$.

Se ha implementado un modelo del receptor por estimación de secuencias basado en el algoritmo de Viterbi; se ha estimado su desempeño para distintos tipos de pulso, duración de este pulso e índices de modulación. Sobre el desempeño de estos receptores hay poca información en la literatura especializada. Resalta el hecho de que la probabilidad de error es independiente de la forma del pulso utilizado.

La complejidad de los receptores basados en el algoritmo de Viterbi se ha hecho patente en el tiempo de duración de las simulaciones. Aunque cabe decir que en la práctica se puede implementar en arquitecturas en paralelo para hacerlo más eficiente.

Las gráficas obtenidas aquí y en la literatura especializada han mostrado que los mejores resultados se dan incrementando las distancias mínimas en los trellis que describen al esquema CPM. Y esto se logra estableciendo pulsos de mayor duración (lo que representa introducir en el sistema más memoria que será aprovechada por el algoritmo de Viterbi) e incrementando p , el denominador cuando se expresa el índice de modulación h como un número racional de acuerdo a $h=2k/p$. Por ejemplo, en igualdad

de condiciones el esquema 3REC presenta mejor comportamiento contra el 1REC. Otro ejemplo se puede observar en el esquema RC cuaternario, en que se obtiene un mejor desempeño para un índice de modulación $h=3/4$ ($p=8$), que con uno de $h=1/2$ ($h=4$).

Podría establecerse también como conclusión que en caso de implementar físicamente uno de estos receptores, se deben buscar estructuras de datos que sean eficientes en el sentido de acortar el número de operaciones que lleva a cabo el algoritmo de Viterbi.

Y por último, cabe hacer resaltar aquí la versatilidad del algoritmo de Viterbi. Durante el desarrollo de esta tesis se encontraron varias aplicaciones para éste en campos muy diversos. Como ejemplos de éstas se tienen: 1) en la decodificación de datos en los que se utilizaron códigos convolucionales para codificación de canal 2) en la codificación conjunta de fuente y de canal, como son los algoritmos de cuantificación vectorial con codificación trellis y 3) como en esta tesis, en demodulación de señales con esquemas de modulación con memoria, como CPM.

CONCLUSIONES

Se han estudiado las características de las señales obtenidas en los esquemas CPM. Entre las principales están las siguientes:

- Tienen una envolvente constante, lo que las hace atractivas para sistemas que utilicen amplificadores no lineales. Éste es el caso en que el consumo de energía es una restricción importante.
- Su envolvente sufre poca distorsión cuando es filtrada, por lo que los amplificadores no lineales generarán señales con poca potencia fuera de la banda de interés.
- Son muy eficientes espectralmente hablando, ya que concentran la potencia de la señal en un ancho de banda menor que los sistemas tradicionales de modulación.
- Dado que son esquemas de modulación que inherentemente tienen memoria, se puede utilizar la demodulación de secuencias además de la detección símbolo a símbolo. Para lograr esto se aplica el algoritmo de Viterbi.
- Son versátiles en el sentido de que proporcionan alternativas para compensar las restricciones del sistema en que se apliquen, gracias a los parámetros libres que tienen.

Se han presentado varias arquitecturas para la construcción de moduladores de fase continua, como lo son las basadas en memorias de sólo lectura (ROMs), la de circuito de amarre de fase (PLL) y la de filtro y limitador; además, se ha presentado una arquitectura simple para el sistema MSK.

Considerando las simulaciones realizadas para encontrar los espectros de las señales CPM, se observa que el ancho de banda ocupado por éstas depende de la elección hecha en el índice de modulación, la forma del pulso $g(t)$ y su duración, y el tamaño del

alfabeto utilizado. Como se ve de las figuras presentadas, valores pequeños del índice de modulación h resultan en señales con pequeño ancho de banda ocupado, mientras que incrementar este índice resulta en anchos de banda mayores. Además, entre más suave sea el pulso de cambio en frecuencia $g(t)$, también es menor el ancho de banda ocupado.

Se ha obtenido una comprobación experimental de la distorsión que sufren las señales moduladas en fase al ser filtradas. Se ha comprobado que las señales CPM sufren una distorsión mucho menor en su envolvente que las señales de los esquemas de modulación tradicionales. Además, se ha comprobado experimentalmente el incremento en la potencia fuera de banda en las señales al ser pasadas por un amplificador no lineal. Se ha observado que las señales con poca distorsión de envolvente sufren una menor regeneración de lóbulos indeseados que las señales con gran distorsión de envolvente, como en el caso de las señales en que la envolvente tiene cruces por cero.

Se ha presentado el análisis de distintos tipos de receptores para CPM, como lo son los detectores coherente y no coherente óptimos para CPFSK, detectores no coherentes simples como el detector diferencial simple y el detector por limitador y discriminador, además de que se ha presentado el modelo para un receptor basado en la estimación de secuencias de máxima verosimilitud que utiliza el algoritmo de Viterbi. Se han presentado resultados de la estimación de la probabilidad de error para distintos esquemas CPM utilizando un receptor de este último tipo.

Se ha verificado que los mejores resultados en cuanto a las probabilidades de error con el receptor basado en el algoritmo de Viterbi se obtienen creando trellis que tengan la distancia mínima entre rutas mayor. La distancia entre las rutas de estos trellis está controlada por la longitud L de la respuesta del pulso de cambio en frecuencia y por el índice de modulación h . De aquí la importancia de estos parámetros para obtener características de desempeño deseadas.

Solamente resta hacer notar que, si bien los sistemas CPM son sistemas con varias características muy atractivas, también son sistemas con altos requerimientos en cuanto a la complejidad tanto de transmisores como de receptores. Afortunadamente los actuales avances tecnológicos permiten alcanzar metas cada vez más ambiciosas, así que se puede comenzar a pensar en estos sistemas como alternativas de modulación para los esquemas actualmente utilizados. Ésto se comienza a ver en sistemas comerciales actuales para transmisiones inalámbricas que utilizan algunos esquemas simples CPM, como son el MSK o el GMSK; además, cabe resaltar que este último es el aplicado en el estándar europeo GSM.

Como conclusión final podría establecerse que fue muy gratificante el llevar a cabo el estudio de estos sistemas de modulación, así como es interesante el reto de llevar a cabo investigación acerca de la forma de mezclar estos sistemas con los de espectro disperso para tratar de obtener las ventajas de cada uno de ellos.

PERSPECTIVAS Y TRABAJO FUTURO

Como ya se mencionó antes, los esquemas de CPM tienen características muy atractivas, pero también requieren de sistemas con una alta complejidad. Tal vez esa sea la principal razón por la que no se les ha difundido tanto. Sin embargo, con los avances tecnológicos que se han estado viviendo últimamente, es tiempo de que se comience a realizar implementaciones de estos esquemas. Ésta sería la primera perspectiva de trabajo futuro: implementaciones físicas de estos esquemas. Los transmisores, como ya se estudió, no establecen requerimientos muy altos, y por esto podrían ser implementados en circuitos integrados de aplicación específica (ASIC), o manejados por un microcontrolador. El caso de los receptores es distinto: para éstos es necesario la utilización de procesadores digitales de señales para poder implementar eficientemente el algoritmo de Viterbi y los filtros acoplados si se desean realizar digitalmente.

Como otra línea para trabajo futuro podría establecer la exploración de sistemas CPM que no se abordaron en esta tesis, como son los que utilizan más de un índice de modulación, denominados *multi-h*, o algunos que utilicen más de un nivel de amplitud, denominados *multi-amplitud*. Se debería entender qué ventajas representa el realizar estas modificaciones a los esquemas aquí estudiados, aunque es sabido que esto incrementará la complejidad de los sistemas de comunicación.

Se puede también realizar un análisis del comportamiento de estos sistemas de modulación en modelos de canal distinto al de ruido blanco gaussiano aditivo; esto para establecer su desempeño con modelos más reales de medios de transmisión.

BIBLIOGRAFÍA

- [1] Anderson, J.B., Aulin, T., y Sundberg C-E. *Digital Phase Modulation*. Plenum Press, 1ª Edición. Estados Unidos de Norteamérica, 1986. 504 pags.
- [2] Proakis, J.G. *Digital Communications*. McGraw-Hill, 3ª Edición. Estados Unidos de Norteamérica, 1995. 928 pags.
- [3] Sklar, B. *Digital Communications, Fundamentals and Applications*. Prentice Hall, 1ª Edición. Estados Unidos de Norteamérica, 1988. 776 pags.
- [4] Picinbono, B. *On Instantaneous Amplitude and Phase of Signals*. IEEE Transactions on Signal Processing, Vol. 45, No. 3, Marzo de 1997. Pags 552-560.
- [5] Pasupathy, S. *Minimum Shift Keying: A Spectrally Efficient Modulation*. IEEE Communications Magazine. Vol. 17, No. 4, Julio de 1979. Pags 14-22.
- [6] Gibson, J. *The Communications Handbook*. IEEE Press y CRC Press, 1ª Edición. Estados Unidos de Norteamérica, 1997. 1598 pags.
- [7] Ziemer, R.E., y Tranter, W.H., *Principles of Communications. Systems, Modulation and Noise*. Houghton Mifflin Company, 3ª Edición. Estados Unidos de Norteamérica, 1990. 812 pags.
- [8] Sundberg, C-E. *Continuous Phase Modulation*. IEEE Communications Magazine, Vol. 24, No. 4, Abril de 1986. Pags 25-38.
- [9] Aulin, T., Rydbeck, N., Sundberg, C-E. *Continuous Phase Modulation-Part II: Partial Response Signaling*. IEEE Transactions on Communications, Vol. COM-29, No. 3, Marzo de 1981. Pags 210-225.

- [10] Aulin, T., Sundberg, C-E. *Continuous Phase Modulation-Part I: Full Response Signaling*. IEEE Transactions on Communications, Vol. COM-29, No. 3, Marzo de 1981. Pags 196-210.
- [11] Osborne, W., Luntz, M. *Coherent and Noncoherent detection of CPFSK*. IEEE Transactions on Communications, Vol. COM-22, No. 8, Agosto de 1974. Pags. 1023-1036.
- [12] Wozencraft, J., Jacobs, I. *Principles of Communications Engineering*. Waveland Press, 1ª Edición. Estados Unidos de Norteamérica, 1965. 720 pags.
- [13] Haykin, S. *An Introduction to Analog and Digital Communications*. John Wiley and Sons, 1ª Edición. Estados Unidos de Norteamérica. 1989. 652 pags.
- [14] Forney, G. D. *The Viterbi Algorithm*. Proceedings of the IEEE, Vol. 61, No. 3, Marzo de 1973. Pags. 268-277.

En esta sección simplemente se mostrarán las funciones principales implementadas en MatLab para llevar a cabo las simulaciones presentadas en esta Tesis.

Existen dos funciones principales en MatLab:

- *cpmmodce*: Para realizar la modulación en banda base dado un vector de datos a transmitir y todos los parámetros del sistema CPM a usar.
- *cpmdemodce*: Para llevar a cabo la demodulación.

Cada una de ellas a su vez hace uso de otras funciones para realizar su función:

- *cpmrom*: Es una función que calcula las formas de onda posibles para un esquema dado. La de modulación hace uso de ella para obtener el contenido de las ROM de transmisor, y la de demodulación la utiliza para calcular la respuesta a impulso de los filtros acoplados.
- *buscamat*: Función implementada para realizar búsquedas de vectores entre las filas de una matriz.
- *cpmtrellis*: Usada por *cpmdemodce* para obtener el diagrama de trellis del sistemas CPM utilizado y aplicarlo en la demodulación de secuencias.

A su vez, la función *cpmrom* hace uso de otras funciones para calcular las formas de onda posibles para el sistema CPM en cuestión:

- *phase_pulse*: Que calcula el pulso $g(t)$ y en base a éste el pulso $q(t)$ del sistema CPM en base a las siguientes funciones:
- *lrec, lrc, lsrc, gmsk, tfm*: Funciones que calculan cada uno de los distintos pulsos $g(t)$.

A continuación se presenta el código en MatLab de cada una de las funciones de que se hace mención en lo anteriormente expuesto:

- CPMMODCE:

```
function [modwave,phase]=cpmmodce(method,cpm_pars,Fd,Fs,data);

% [modwave]=cpmmodce(method,cpm_pars,T,data);
% Funcion de modulacion de fase continua en banda base.
% Devuelve:
%   modwave: Vector complejo con la señal modulada en fase continua por el esquema
%             indicado. Su estructura es:
%             modwave= I(t) + i Q(t).
% Recibe:
%   method: Metodo utilizado para determinar el pulso de frecuencia. Valores posibles:
%           'LRC': Coseno alzado de longitud L intervalos de señalizacion.
%           'LREC': Pulso rectangular en frecuencia de longitud L intervalos de señalizacion
%           'GMSK': MSK con forma gaussiana.
%           'LSRC': Coseno alzado espectral le longitud L intervalos de señalizacion.
%           'TFM': Tamed FM.
%   cpm_pars: Vector con los parametros del esquema de modulacion de fase continua con
%             la estructura siguiente:
%             cpm_pars(1) = L
%                 Longitud de la respuesta parcial.
%             cpm_pars(2) = h
%                 Indice de modulacion.
%             cpm_pars(3) = extra_par (Vease la funcion cpmrom)
%                 Parametro extra para algunos esquemas.
%             cpm_pars(4) = M
%                 Cantidad de elementos en el alfabeto.
%             cpm_pars(5) = p
%                 Cantidad de estados de fase. (h=2k/p).
%             cpm_pars(6) = trunc
%                 Truncamiento de los pulsos infinitos.
%   Fd: Frecuencia de señalizacion.
%   Fs: Frecuencia de muestreo de la señal modulada.
%   IMPORTANTE: Fs/Fd debe ser un numero entero, y es la cantidad de veces que el
%             vector de salida es mayor que el de entrada.
%   data: Vector con los datos a modular.

% Declaraciones.
L=cpm_pars(1);           % Descomponer los parametros del esquema cpm de entrada.
h=cpm_pars(2);
extra_par=cpm_pars(3);
M=cpm_pars(4);
p=cpm_pars(5);
trunc=cpm_pars(6);
T=1/Fd;                 % Periodo de señalizacion.
m=Fs/Fd;                % Cantidad de muestras por periodo de señalizacion.
i_wave=zeros(1,m*length(data)); % Inicializa la onda en fase.
q_wave=zeros(1,m*length(data)); % Inicializa la onda en cuadratura.
phase=zeros(1,m*length(data)); % Inicializa el vector que muestra el cambio de fase en
tiempo.
counter_state=0;        % Inicializa contador de estado.
phase_state=0;          % Inicializa estado de fase.
shift_reg=[];           % Registro de corrimiento de los simbolos.
temp=[];                % Vector temporal.
j=0;                    % Contadores para FORs.
g=0;
y=0;
x=0;
ind_wave=0;             % Indice para la busqueda.

% Fin de declaraciones.

method=upper(method);
[index,crom,srom,tree]=cpmrom(method,L,h,T,m,extra_par,M,p,trunc);
% Obten el contenido de la rom para el modulador.
```

```

if strcmp(method, 'GMSK') | strcmp(method, 'LSRC') | strcmp(method, 'TFM')
    L=2*trunc; % Si el metodo es alguno de estos, entonces el tamaño de las
end % formas de onda cambia a 2*trunc en lugar de L
data=[ones(1,L-1),data]; % Agrega al vector de datos los del estado inicial.

for j=1:length(data)-L+1 % Para todos los datos de entrada...
    shift_reg=fliplr(data(j:j+L-1)); % Obten el contenido del registro de corrimiento, que
    % son todos los datos que afectaran la fase en el
    % estado actual.
    temp=[phase_state,shift_reg]; % Vector que servira como indice para buscar la forma
    % de onda requerida. Se compone del estado de fase y
    % de los datos que estan en la respuesta parcial.
    [y,x]=size(index); % Obten datos de la matriz index para hacer busqueda
    ind_wave=0;
    ind_wave=buscamat(index,temp); % Busca el vector en la matriz de indices.
    i_wave(((j-1)*m)+1:j*m)=crom(ind_wave,:); % Guarda las formas de onda correspondientes
    q_wave(((j-1)*m)+1:j*m)=srom(ind_wave,:); % al la parte real e imaginaria y al cambio
    phase(((j-1)*m)+1:j*m)=mod(tree(ind_wave,:),2*pi); % de fase
    counter_state=counter_state+shift_reg(L); % Actualiza el contador de estado.
    phase_state=mod(h*counter_state,2); % Con el contador de estado, actualiza el estado de
    phase_state=phase_state*pi; % fase.
end

%figure(10)
%subplot(2,1,1);plot(i_wave, '.');title('I(t)');axis([1,length(i_wave),-1,1])
%subplot(2,1,2);plot(q_wave, '.');title('Q(t)');axis([1,length(q_wave),-1,1])

%figure(11)
%subplot(2,1,1);plot(phase, '.');
%temp=axis;axis([temp(1) temp(2) -pi/2 2*pi]);grid on;
%title('Cambio de la Fase en el tiempo');

modwave=i_wave+(i.*q_wave); % Devuelve la señal modulada como numeros complejos.

```

• CPMDEMODCE:

```

function [data,metric]=cpmdemodce(method,cpm_pars,Fd,Fs,modwave,pathlength);

% [data]=cpmdemodce(method,cpm_pars,T,modwave);
% Funcion de demodulacion de fase continua en banda base. Es coherente, ya que asume que
% se tiene conocimiento de la fase de la portadora y de los intervalos de señalizacion.
% Devuelve:
% data: Vector con los datos recuperados de la señal modulada
% Recibe:
% method: Metodo utilizado para determinar el pulso de frecuencia. Valores posibles:
% 'LRC': Coseno alzado de longitud L intervalos de señalizacion.
% 'LREC': Pulso rectangular en frecuencia de longitud L intervalos de señalizacion.
% 'GMSK': MSK con forma gaussiana.
% 'LSRC': Coseno alzado espectral le longitud L intervalos de señalizacion.
% 'TFM': Tamed FM.
% cpm_pars: Vector con los parametros del esquema de modulacion de fase continua con
% la estructura siguiente:
% cpm_pars(1) = L
% Longitud de la respuesta parcial.
% cpm_pars(2) = h
% Indice de modulacion.
% cpm_pars(3) = extra_par (Vease la funcion cpmrom)
% Parametro extra para algunos esquemas.
% cpm_pars(4) = M
% Cantidad de elementos en el alfabeto.
% cpm_pars(5) = p
% Cantidad de estados de fase. (h=2k/p).
% cpm_pars(6) = trunc
% Truncamiento de los pulsos infinitos.
% Fd: Frecuencia de señalizacion.

```

```

% Fs: Frecuencia de muestreo de la señal modulada.
% IMPORTANTE: Fs/Fd debe ser un numero entero, y es la cantidad de veces que el
% vector de salida es mayor que el de entrada.
% modwave: Señal modulada en banda base con el formato:
% modwave= I(t) + i Q(t).
% pathlength: Longitud que se desea utilizar para almacenar las rutas en el algoritmo
% de Viterbi.

% Declaraciones
L=cpm_pars(1); % Descomponer los parametros del esquema cpm de entrada.
h=cpm_pars(2);
extra_par=cpm_pars(3);
M=cpm_pars(4);
p=cpm_pars(5);
trunc=cpm_pars(6);
T=1/Fd; % Periodo de señalizacion.
m=Fs/Fd; % Cantidad de muestras por periodo de señalizacion.
num_filt=0; % Cantidad de filtros
sampler=[m:m:length(modwave)]; % Vector que indica los instantes de muestreo a la salida
% de los filtros acoplados.

% Fin de declaraciones

[index_ROM,crom,srom,tree]=cpmrom(method,L,h,T,m,extra_par,M,p,trunc);
% Obtener las formas de onda posibles.

method=upper(method);
if strcmp(method,'GMSK') | strcmp(method,'LSRC') | strcmp(method,'TFM')
    L=2*trunc; % Si el metodo es alguno de estos, entonces el tamaño de las
end % formas de onda cambia a 2*trunc en lugar de L

% Declaraciones dependientes de los parametros
n_states=p*(M^(L-1)); % Cantidad de estados.
paths=zeros(n_states,pathlength); % Matriz donde se guardaran las rutas.
path_datos=zeros(n_states,pathlength); % Matriz donde se guardaran los datos de las rutas.
metric=ones(p*(M^L),length(sampler))*inf; % Matriz para las metricas.
data=zeros(1,length(sampler)); % Vector en que se devolveran los datos.
metric_ac=zeros(n_states,1); % Vector donde se guardara el acumulado de las metricas.
metric_sig=ones(n_states,1)*(-inf); % Vector de metricas acumuladas siguientes.
alphabet=[M-1:-2:-(M-1)]; % Alfabeto a trabajar.
% Fin de declaraciones dependientes de parametros.

num_filt=M^L; % Cantidad de bancos de filtros requeridos.
% La cantidad de filtros requeridos será igual a
% cuatro veces esa cantidad, ya que se requiere filtrar cada
% una de las componentes de cuadratura con cada uno de ellos.

cfilters=fliplr(crom(1:num_filt,:)); % Obtencion de los filtros, correspondientes al
sfilters=fliplr(srom(1:num_filt,:)); % acoplado de las primeras formas de onda obtenidas
% que tienen un estado de fase de cero.

cfiltered=zeros(num_filt,length(modwave)); % Inicializacion de las matrices en que se
sfiltered=zeros(num_filt,length(modwave)); % guardaran las señales filtradas.
cqfiltered=zeros(num_filt,length(modwave));
sqfiltered=zeros(num_filt,length(modwave));

I=real(modwave); % Separacion de la señal recibida en sus componentes en Fase
Q=imag(modwave); % y cuadratura.

for k=1:num_filt
    cfiltered(k,:)=filter(cfilters(k,:),1,I); % Filtraje del par de señales con el par de
    sfiltered(k,:)=filter(sfilters(k,:),1,I); % filtros obtenidos. Esto produce cuatro
    cqfiltered(k,:)=filter(cfilters(k,:),1,Q); % cuatro señales a la salida de los filtros
    sqfiltered(k,:)=filter(sfilters(k,:),1,Q); % para cada bloque
end

cisampled=cfiltered(:,sampler); % Muestreo de las señales de salida de los filtros
sisampled=sfiltered(:,sampler); % acoplados en los intervalos de señalización.
cqsampled=cqfiltered(:,sampler);
sqsampled=sqfiltered(:,sampler);

sum1=cisampled+sqsampled; % Aplicacion de los dos sumadores de las muestras

```

```

sum2=cqsampled-sisampled;           % obtenidas en el paso anterior.
theta_n=((2*pi)/p).*[0:1:p-1];      % Vector de todos los estados de fase posible.
index=1;                             % Generacion de los posibles sen(theta_n) y cos(theta_n)
                                     % para calcular p metricas por cada bloque de filtros.
for l=1:p
    for k=1:num_filt
        metric(index,:)=(sum1(k,:).*cos(theta_n(l)))+(sum2(k,:).*sin(theta_n(l)));
        index=index+1;
    end
end

[a,b]=size(index_ROM);
t=[index_ROM,(1:a)'];                % La matriz de metricas no tiene el orden adecuado
x=sortrows(-t,2);                    % para la aplicacion del algoritmo de Viterbi
x=-x;                                 % en la forma como sigue. Por esto, hay que darle
metric_temporal=metric;              % otro acomodo en base a que queden primero las
for j=1:a                             % metricas correspondientes a mandar M-1, luego
    metric(j,:)=metric_temporal(x(j,L+2),:); % M-3, etc., hasta -(M-1). Esto se encuentra
end                                     % en la segunda columna de la matriz de indices
                                     % de formas de onda.

[trellis,estados]=cpmtrellis(L,h,M,p,0,index_ROM);
                                     % Obtencion del Trellis correspondiente al
                                     % esquema utilizado

contador_datos=1;
for j=1:1:length(sampler)            % Para todas las muestras de entrada...
    contador_metrica=0;              % Contador para moverse entre las metricas del
                                     % instante actual.
    for l=1:1:M                       % Para cada uno de los estados...
        for k=1:1:n_states            % Para cada posible transicion de ese estado a
                                     % otro...
            if trellis(k,l)~=0        % Si la transicion es valida, hay que hacer ac-
                                     % tualizaciones de acuerdo al algoritmo de Viterbi
                contador_metrica=contador_metrica+1; % Para movernos entre las metricas...
                met=metric(contador_metrica,j); % Obtencion de la metrica para ese instante,
                                     % ese estado y con esa transicion.
                s_act=k;                % Estado actual
                s_sig=trellis(k,l);     % Estado siguiente
                if j<pathlength        % Si no se ha llenado la matriz de las rutas, hay
                                     % que ir llenando
                    if metric_ac(s_act)+met>metric_sig(s_sig) % Si la metrica es mayor, hay
                        metric_sig(s_sig)=metric_ac(s_act)+met; % que hacer las actualizaciones
                        paths(s_sig,j+1)=s_act;                % necesarias en las rutas, los
                                                                % datos y las metricas.
                    path_datos(s_sig,j+1)=alphabet(l);
                end
            else
                if metric_ac(s_act)+met>metric_sig(s_sig) % Si la matriz de rutas ya estaba
                    metric_sig(s_sig)=metric_ac(s_act)+met; % llena, hay que hacer las mismas
                    paths(s_sig,pathlength)=s_act;          % actualizaciones pero con dife-
                                                                % rentes indices.
                path_datos(s_sig,pathlength)=alphabet(l);
            end
        end
    end
end
end
metric_ac=metric_sig;                % Ahora la metrica acumulada se puede actualizar
metric_sig=ones(n_states,1)*(-inf); % Y la metrica siguiente se inicializa de nuevo

if j>=pathlength-1                  % Si la matriz de rutas ya se lleno, se puede realizar
                                     % el traceback para ir decodificando los datos.
                                     % Como hay mas de una ruta sobreviviente, se
tb=find(metric_ac-max(metric_ac)==0); % debe buscar la ruta con la metrica
if length(tb)>1                       % acumulada mayor. Si hay mas de una ruta
    tb=tb(1);                          % con la misma metrica acumulada, se eli-
end                                     % ge la primera.
estado_ant=tb;
for n=pathlength:-1:3

```



```

        estado_ant=paths(estado_ant,n);
    end
    data(contador_datos)=path_datos(estado_ant,2);
    contador_datos=contador_datos+1;
    paths(:,1:pathlength-1)=paths(:,2:pathlength); % Actualiza la matriz de rutas y la
    paths(:,pathlength)=zeros(n_states,1); % de datos
    path_datos(:,1:pathlength-1)=path_datos(:,2:pathlength);
    path_datos(:,pathlength)=zeros(n_states,1);
end
end

tb=find(metric_ac-max(metric_ac)==0); % Al finalizar, se decodifican los datos
if length(tb)>1 % restantes en base a la mayor metrica
    tb=tb(1); % acumulada. Como el recorrido a traves de la
end % ruta restante es hacia atras, tambien se
contador_datos=length(data); % decodifican los datos en forma inversa. Si
estado_ant=tb; % existe mas de una ruta con la misma metrica
for n=pathlength-1:-1:1 % acumulada, se escoge la primera de ellas.
    data(contador_datos)=path_datos(estado_ant,n);
    estado_ant=paths(estado_ant,n);
    contador_datos=contador_datos-1;
end
end

```

• CPMROM:

```

function [index,crom,srom,tree]=cpmrom(method,L,h,T,m,extra_par,M,p,trunc)

% [index,crom,srom,tree]=cpmrom(method,L,h,T,m,extra_par,M,p)
% Funcion que calcula el contenido de la ROM para un transmisor CPM, y devuelve esta ROM,
% ademas de los indices para identificar las formas de onda, y una parte del arbol de fase
% correspondiente al esquema de modulacion tratado.
% Devuelve:
%   index: Matriz indice de las formas de onda, con el orden:
%           [ theta_n  alfa_n  alfa_n-1  alfa_n-2  .... ]
%   crom: Matriz con las formas de onda que se deben guardar en la ROM del transmisor de
%         la componente en fase I(t).
%   srom: Matriz con las formas de onda que se deben guardar en la ROM del transmisor de
%         la componente en cuadratura Q(t).
%   tree: Parte del arbol de fase correspondiente al esquema de modulacion.
% Recibe:
%   method: Metodo utilizado para determinar el pulso de frecuencia. Valores posibles:
%           'LRC': Coseno alzado de longitud L intervalos de señalizacion.
%           'LREC': Pulso rectangular en frecuencia de longitud L intervalos de señalizacion.
%           'GMSK': MSK con forma gaussiana.
%           'LSRC': Coseno alzado espectral le longitud L intervalos de señalizacion.
%           'TFM': Tamed FM.
%   L: Cantidad de intervalos de señalizacion que afecta el pulso en frecuencia.
%   h: Indice de modulacion.
%   T: Intervalo de señalizacion.
%   m: Cantidad de muestras deseadas por intervalo de señalizacion.
%   extra_par: Parametro extra para algunos esquemas de modulacion.
%             Cuando la modulacion es GMSK, es el parametro B.
%             Cuando la modulacion es LSRC, es el parametro betta.
%   M: Cantidad de elementos en el alfabeto (binario, cuaternario, ...)
%   p: Cantidad de estados de fase (h=2k/p)
%   trunc: Cantidad de intervalos de señalizacion a que se truncaran los pulsos de
%          duracion infinita. Se truncará hacia la derecha y hacia la izquierda con el
%          mismo numero.

% Declaraciones
alfas=[]; % Combinaciones posibles de los simbolos.
alfa=[]; % Una de las combinaciones de simbolos.
alfav=[]; % Una de las combinaciones con los simbolos repetidos m veces.
q=[]; % Pulso de fase.
g=[]; % Pulso de frecuencia.
theta_n=[]; % Estados de fase.
theta_alfa=[]; % Vector para ir almacenando el resultado de la fase.
temp=[]; % Vector temporal.

```

```

samples=[]; % Formas de onda para cada combinacion de simbolos.
index=[]; % Referencia para búsquedas de las formas de onda.
phase=[]; % Vector con las formas de onda calculadas en cada iteracion.
contador=0; % Contador.
j=0; % Contadores para los FORs.
k=0;
l=0;
n=0;
% Fin de declaraciones

method=upper(method);
[q,g] = phase_pulse(method,L,T,m,extra_par,trunc);
% Calcula la forma del pulso modulador de
%_ de la fase.

theta_n=((2*pi)/p).*[0:1:p-1]; % Vector de todos los estados de fase posible.

if strcmp(method,'GMSK') | strcmp(method,'LSRC') | strcmp(method,'TFM')
    L=2*trunc; % Si el metodo es alguno de estos, entonces el tamaño de las
end % formas de onda cambia a 2*trunc en lugar de L

samples=zeros(p*(M^L),m); % Formas de onda para cada combinacion de simbolos.
index=zeros(p*(M^L),L+1); % Referencia para búsquedas de las formas de onda.

alfas=matcomb(M,L); % Calcula las posibles combinaciones de entradas restringidas a
% una longitud L de simbolos.

for j=1:length(theta_n) % Para cada estado de fase calcula las posibles formas de onda...
    for k=1:M^L % Para cada vector distinto de entrada de longitud L.
        alfa=alfas(k,:); % Vector de entrada.
        contador=contador+1;
        alfav=ones(1,L*m); % Vector con los datos repetidos
        for l=0:L-1
            alfav((l*m)+1:(l+1)*m)=repmat(alfa(l+1),1,m);
        end
        temp=alfav.*q; % Multiplica los datos por el pulso de modulacion de fase.
        theta_alfa=zeros(1,m); % Inicializa el vector donde se almacene el resultado.
        for n=1:L % Parte el vector temporal en L cachos de longitud T y
            % sumalos para obtener la respuesta en fase en ese
            % intervalo por el esquema de respuesta parcial.
            theta_alfa=theta_alfa+temp(m*(n-1)+1:m*n);
        end
        phase=theta_n(j)+(2*pi*h*theta_alfa); % Para completar la fase, ademas de considerar
            % el esquema de respuesta parcial, se debe sumar el
            % estado de fase.
        samples(contador,:)=phase; % Guarda la forma de onda.
        index(contador,:)=[theta_n(j) alfa]; % Guarda la referencia de la forma de onda.
    end
end
tree=samples;
crom=cos(samples); % A partir de los datos guardados, el contenido de la rom de la
srom=sin(samples); % componenten en fase es el coseno y el de la componente en
% cuadratura es el seno.

%figure(12);
%subplot(1,2,1); plot(tree','b');title('Arbol de fase del modulador');grid on
%subplot(2,2,2); plot(crom','b');title('Formas de onda para I(t)');grid on
%subplot(2,2,4); plot(srom','b');title('Formas de onda para Q(t)');grid on

function [alfas]=matcomb(M,L)
% Funcion que calcula las posibles combinaciones de un conjunto dado o alfabeto finito con
% un numero de elementos M, tomados de L en L.
% Devuelve:
% alfas: Matriz con las posibles combinaciones de elementos.
% Recibe:
% M: Cantidad de elementos en el alfabeto. Debe ser par y los posibles elementos se
% calculan a partir de:
% -(M-1), -(M-3), ..., -3, -1, 1, 3, ..., M-3, M-1

```

```

%      L: De cuantos en cuantos se van a tomar los elementos del alfabeto, o longitud de
%      cada una de las combinaciones.

% Declaraciones
alphabet=[M-1:-2:-(M-1)]';           % Alfabeto a trabajar.
alfas=ones(M^L,L);                   % Matriz de combinaciones.
j=0;                                  % Contadores para FORs.
% Fin de declaraciones

cont=L-1:-1:0;
for j=1:L
    alfas(:,j)=repmat(dmodce(alphabet,1,M^cont(j),'sample'),M^(j-1),1);
end

```

• CPMTRELLIS:

```

function [trellis,estados]=cpmtrellis(L,h,M,p,plot_flag,index);

% [data]=cpmdemodce(method,cpm_pars,T,modwave);
% Funcion que calcula el trellis del esquema CPM que es utilizado por el algoritmo de
% Viterbi en la demodulacion
% Devuelve:
%   trellis: Matriz con el trellis del esquema CPM.
%   estados: Nombres de los estados del trellis.
% Recibe:
%   L: Longitud de la respuesta.
%   h: Indice de modulacion.
%   M: Cantidad de elementos en el alfabeto.
%   p: Cantidad de estados de fase. (h=2k/p).
%   plot_flag: Bandera para graficar el trellis. Si es cero no lo grafica.

% Declaraciones
n_states=p*(M^(L-1));                 % Cantidad de estados que tendra el trellis
trellis=zeros(n_states,M);            % Matriz en que se devolvera el trellis
alfas=mstates(M,L);                  % Calculando las combinaciones de los datos
theta_n=((2*pi)/p).*[0:1:p-1];       % Vector de todos los estados de fase posible
estados=zeros(n_states,L);           % Matriz donde se guardaran los nombres de los estados
alphabet=[M-1:-2:-(M-1)]';          % Alfabeto a trabajar.
contador=0;
% Fin de declaraciones

if L==1                               % Obtencion de los nombres de todos los estados del
    estados=theta_n';                 % trellis. Si es L=1, los nombres de los estados son
else                                  % simplemente los estados de fase. Si no, para cada
    for j=1:p                          % estado de fase se repiten todas las combinaciones
        for k=1:M^(L-1)                 % de L-1 simbolos
            contador=contador+1;
            estados(contador,:)=[theta_n(j),alfas(k,:)];
        end
    end
end

for j=1:n_states                       % Para cada estado, calcula su siguiente estado
    for k=1:M
        vector_act=estados(j,:);
        theta_n_act=vector_act(1);

        dato_sig=alphabet(k);

        if L==1
            theta_n_sig=mod(theta_n_act+(pi*h*dato_sig),2*pi);
            vector_sig=[theta_n_sig];
        else
            theta_n_sig=mod(theta_n_act+(pi*h*vector_act(length(vector_act))),2*pi);
            vector_sig=[theta_n_sig,dato_sig,vector_act(2:length(vector_act)-1)];
        end
    end
end

```

```

        t2=buscamat(estados,vector_sig);
        trellis(j,k)=t2;
    end
end

if plot_flag
    if plot_flag<0
        plot_falg=-plot(flag);
    end
    t1=ones(1,n_states);
    t2=ones(1,n_states)*2;
    figure(plot_flag);clf;hold on
    plot(t1,[1:1:n_states],'.r');
    plot(t2,[1:1:n_states],'.r');
    set(gca,'YTick',[1:1:n_states])
    set(gca,'XTick',[0]);
    set(gca,'box','on');
    axis([0.5,2.5,0.5,n_states+0.5])

    estados

    for j=1:1:n_states
        for k=1:1:M
            if trellis(j,k)~=0
                line([1,2],[j,trellis(j,k)]);
            end
        end
    end
end

figure(plot_flag);hold off
end

function [alfas]=mstates(M,L)
% Funcion que calcula las posibles combinaciones de un conjunto dado o alfabeto finito con
% un numero de elementos M, tomados de L-1 en L-1.
% Devuelve:
%   alfas: Matriz con las posibles combinaciones de elementos.
% Recibe:
%   M: Cantidad de elementos en el alfabeto. Debe ser par y los posibles elementos se
%   calculan a partir de:
%       -(M-1), -(M-3), ... , -3, -1, 1, 3, ... , M-3, M-1
%   L: De cuantos en cuantos se van a tomar los elementos del alfabeto, o longitud de
%   cada una de las combinaciones.

% Declaraciones
alphabet=[M-1:-2:-(M-1)]';           % Alfabeto a trabajar.
alfas=ones(M^(L-1),L-1);           % Matriz de combinaciones.
j=0;                                % Contadores para FORs.
% Fin de declaraciones

cont=L-2:-1:0;
for j=1:1:L-1
    alfas(:,j)=repmat(dmodce(alphabet,1,M^cont(j),'sample'),M^(j-1),1);
end

```

● BUSCAMAT:

```

function [i]=buscamat(matriz,vector)

% [i]=buscamat(matriz,vector);
% Funcion que busca un vector fila entre los renglones de una matriz, y devuelve el indice
% del renglon donde se encontro.
% Devuelve:
%   i: Renglon donde se encontro el vector. Si no se encontro, devuelve 0.

```

```

% Recibe:
%   matriz: Matriz en la que se hara la busqueda (debe ser de dos dimensiones).
%   vector: Vector fila a buscar (debe ser de una dimension).

[mf,mc]=size(matriz);
[vf,vc]=size(vector);
if mc~=vc
    disp 'Error en las dimensiones de la matriz y el vector';
else
    t=matriz-repmat(vector,mf,1);
    if mc~=1
        s=sum(abs(t'));
    else
        s=abs(t);
    end
    i=find(s<1e-10);
    if isempty(i)
        i=0;
    end
end
end

```

● PHASE_PULSE:

```

function [q,g] = phase_pulse(method,L,T,samples,extra_par,trunc)

% [q,g] = phase_pulse(method,L,h,T,samples,extra_par)
% Funcion que devuelve los pulsos de modulacion tanto de fase como de frecuencia
% para diversos esquemas de modulacion de fase continua (CPM).
% Devuelve:
%   t: vector con las muestras del tiempo.
%   q: vector con el pulso de modulacion de fase.
%   g: vector con el pulso de modulacion en frecuencia.
% Recibe:
%   method: Metodo utilizado para determinar el pulso de frecuencia. Valores posibles:
%   'LRC': Coseno alzado de longitud L intervalos de señalizacion.
%   'LREC': Pulso rectangular en frecuencia de longitud L intervalos de señalizacion.
%   'GMSK': MSK con forma gaussiana.
%   'LSRC': Coseno alzado espectral le longitud L intervalos de señalizacion.
%   'TFM': Tamed FM.
%   L: Cantidad de intervalos de señalizacion que afecta el pulso en frecuencia.
%   T: Intervalo de señalizacion.
%   samples: Cantidad de muestras deseadas por intervalo de señalizacion.
%   extra_par: Parametro extra para algunos esquemas de modulacion.
%   trunc: Cantidad de intervalos de señalizacion a que se truncaran los pulsos de
%   duracion infinita. Se truncará hacia la derecha y hacia la izquierda con el
%   mismo numero.

% Declaraciones
warning off;
t=[]; % Vector del tiempo.
B=extra_par;
betha=extra_par;
q=zeros(1,L*samples); % Inicializacion de los vectores con los pulsos
g=zeros(1,L*samples); % de fase y de frecuencia.
j=0; % Contador para FORs.
% Fin de declaraciones

% Selecciona el pulso g(t) dependiendo del metodo de modulacion de fase continua que
% se esta utilizando.

switch upper(method)
case 'LREC'.
    t=[1:samples.*L].*(T/samples); % Vector del tiempo.
    g=lrec(t,L,T);
    for j=1:length(t)
        q(j)=quad8('lrec',0,t(j),[],[],L,T);
    end

```

```

case 'LRC',
    t=[1:samples.*L].*(T/samples);      % Vector del tiempo.
    g=lrc(t,L,T);
    for j=1:length(t)
        q(j)=quad8('lrc',0,t(j),[],[],L,T);
    end

case 'GMSK',
    t=[(-(trunc*T))+T/samples:T/samples:trunc*T];
    g=gmsk(t,T,B);
    for j=1:length(t)
        q(j)=quad8('gmsk',-trunc*T,t(j),[],[],T,B);
    end

case 'LSRC',
    t=[(-(trunc*T))+T/samples:T/samples:trunc*T];
    g=lsrc(t,L,T,betha);
    for j=1:length(t)
        q(j)=quad8('lsrc',-trunc*T,t(j),[],[],L,T,betha);
    end

case 'TFM',
    t=[(-(trunc*T))+T/samples:T/samples:trunc*T];
    g=tfm(t,T);
    for j=1:length(t)
        q(j)=quad8('tfm',-trunc*T,t(j),[],[],T);
    end

otherwise,
    disp('Unknown method');
end

warning on

```

- LREC:

```

function [g]=lrec(t,L,T)

for j=1:length(t)
    if t(j)<0 | t(j)>(L*T)
        g(j)=0;
    else
        g(j)=1/(2*L*T);
    end
end

```

- LRC:

```

function [g]=lrc(t,L,T)

for j=1:length(t)
    if t(j)<0 | t(j)>(L*T)
        g(j)=0;
    else
        temp=1-(cos((2*pi*t(j))/(L*T)));
        g(j)=(1/(2*L*T))*temp;
    end
end

```

- LSRC:

```

function [g]=lsrc(t,L,T,betha)

```

```

s=sinc((2.*t)/(L*T));
temp=((1)/(L*T)).*s;
temp2=cos(betha*((2*pi.*t)/(L*T)));
temp3=1-(((4*betha.*t)/(L*T)).^2);
temp4=(1./temp3).*temp2;
g=temp.*temp4;

if any(isinf(g)) % Elimina valores de infinito, interpolando entre el valor
temp=find(isinf(g)); % anterior y el siguiente.
for h=1:length(temp)
if temp(h)==1
g(1)=g(2);
elseif temp(h)==length(g)
g(temp(h))=g(temp(h)-1);
else
g(temp(h))=(g(temp(h)-1)+g(temp(h)+1))/2;
end
end
end

if any(isnan(g)) % Elimina valores de 'Not a Number'. interpolando entre el
temp=find(isnan(g)); % valor anterior y el siguiente.
for h=1:length(temp)
if temp(h)==1
g(1)=g(2);
elseif temp(h)==length(g)
g(temp(h))=g(temp(h)-1);
else
g(temp(h))=(g(temp(h)-1)+g(temp(h)+1))/2;
end
end
end
end

```

- **GMSK:**

```

function [g]=gmsk(t,T,B)

temp1=(2*pi*B)/sqrt(log(2));
temp2=temp1.*(t-(T/2));
temp3=temp1.*(t+(T/2));
temp4=(1/2).*erfc(temp2./sqrt(2));
temp5=(1/2).*erfc(temp3./sqrt(2));
g=(1/(2*T)).*(temp4-temp5);

```

- **TFM:**

```

function [g]=gmsk(t,T)

temp1=(pi/T).*t;
temp2=temp1-pi;
temp3=temp1+pi;

%calculo de g_o(t)
temp_a=sin(temp1)./temp1;
temp_b=2.*sin(temp1);
temp_c=2.*temp1.*cos(temp1);
temp_d=(temp1.^2).*sin(temp1);
temp_e=temp1.^3;
g2=(1/T).*( temp_a - (((pi*pi)/24)./temp_e).*(temp_b-temp_c-temp_d) );

%calculo de g_o(t-T)
temp_a=sin(temp2)./temp2;
temp_b=2.*sin(temp2);
temp_c=2.*temp2.*cos(temp2);

```

```

temp_d=(temp2.^2).*sin(temp2);
temp_e=temp2.^3;
g1=(1/T).*( temp_a - ((pi*pi)/24)./temp_e).*(temp_b-temp_c-temp_d) );

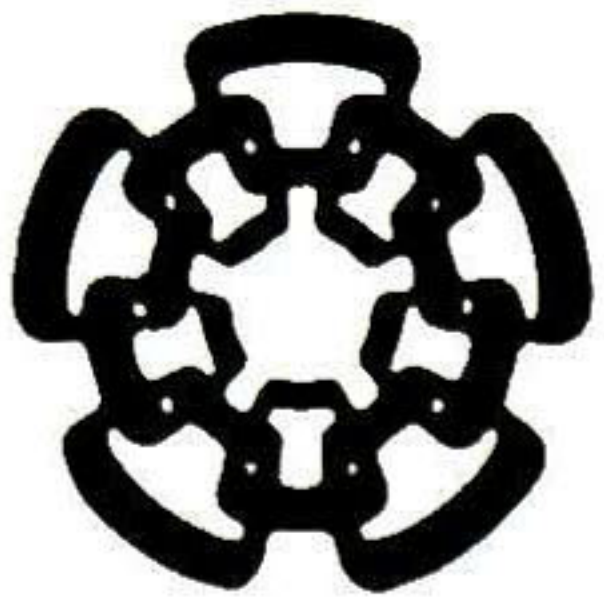
%calculo de g_o(t+T)
temp_a=sin(temp3)./temp3;
temp_b=2.*sin(temp3);
temp_c=2.*temp3.*cos(temp3);
temp_d=(temp3.^2).*sin(temp3);
temp_e=temp3.^3;
g3=(1/T).*( temp_a - ((pi*pi)/24)./temp_e).*(temp_b-temp_c-temp_d) );

g=(1/8).*(g1+(2.*g2)+g3);

if any(isinf(g))          % Elimina valores de infinito, interpolando entre el valor
    temp=find(isinf(g)); % anterior y el siguiente.
    for h=1:length(temp)
        if temp(h)==1
            g(1)=g(2);
        elseif temp(h)==length(g)
            g(temp(h))=g(temp(h)-1);
        else
            g(temp(h))=(g(temp(h)-1)+g(temp(h)+1))/2;
        end
    end
end

if any(isnan(g))          % Elimina valores de 'Not a Number'. interpolando entre el
    temp=find(isnan(g)); % valor anterior y el siguiente.
    for h=1:length(temp)
        if temp(h)==1
            g(1)=g(2);
        elseif temp(h)==length(g)
            g(temp(h))=g(temp(h)-1);
        else
            g(temp(h))=(g(temp(h)-1)+g(temp(h)+1))/2;
        end
    end
end
end

```

**CENTRO DE INVESTIGACION Y DE ESTUDIOS AVANZADOS DEL IPN
UNIDAD GUADALAJARA**

El Jurado designado por la Unidad Guadalajara del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, aprobó la tesis: **“Estudio y simulación de algunos esquemas de modulación de fase continua (CPM)”** del Sr. **Jorge Hermosillo Gutiérrez** el día 25 de Octubre de 1999.

Dr. José Luis Naredo Villagrán
Investigador Cinvestav 3A
CINVESTAV DEL IPN
Guadalajara

Dr. Antonio Ramírez Treviño
Investigador Cinvestav 2A
CINVESTAV DEL IPN
Guadalajara

Dr. Fernando Ramírez Mireles
Profesor Auxiliar
Sección de Comunicaciones
CINVESTAV DEL IPN
México
(Actualmente con licencia)



CINVESTAV
BIBLIOTECA CENTRAL



SSIT000003838