Proyecto de Equipos de Radio Definidos por Software

Sistema de comunicación en de transmisiones BFSK

18 de enero de 2018

Ander Doncel Llamas

Daniel Montesano

Tabla de contenido

[1. Especificaciones de proyecto 3](#_Toc502830000)

[2. Comunicación Monoportadora 4](#_Toc502830001)

[2.1. Transmisor 4](#_Toc502830002)

[2.2. Canal 4](#_Toc502830003)

[2.3. Receptor 4](#_Toc502830004)

[2.3.1. Filtrado Paso Banda & Detección de Envolvente 4](#_Toc502830005)

[2.3.2. Demodulación de Portadora & Filtrado Paso Banda & Detección de Envolvente 4](#_Toc502830006)

[2.3.3. Demodulación Coherente Utilizando PLLs sintonizados. 4](#_Toc502830007)

[2.4. Comunicación de espectro ensanchado 4](#_Toc502830008)

# Especificaciones de proyecto

Este documento describe el proceso de diseño e implementación de un sistema de comunicación basado en transmisiones BFSK desarrollado en MATLAB. Este esquema de modulación implementa una variación en frecuencia sobre la de la señal portadora para transmitir los símbolos, de manera que se utilicen diferentes frecuencias para transmitir cada uno de los mismos. Dado que este caso se corresponde a la implementación binaria de una modulación en frecuencia FSK, los símbolos a transmitir se corresponderán únicamente a ‘1’s y ‘0’s. La siguiente figura ilustra el concepto de modulación BFSK.

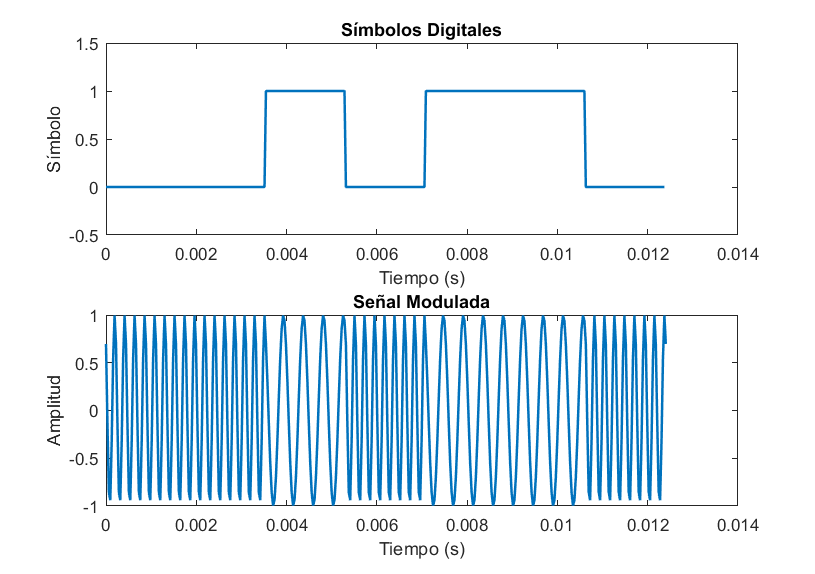


Figura 1. Modulación BFSK.

Los parámetros que describen el sistema de comunicaciones a implementar se describen a partir de los siguientes valores:

* Velocidad binaria (Rb) = 564,48 bps.
* Frecuencia de la portadora (fc) =
* Los símbolos ‘1’ estarán compuestos por 7 ciclos a frecuencia f1 utilizada.
* Los símbolos ‘0’ estarán compuestos por 8 ciclos a frecuencia f0 utilizada.
* Separación de frecuencias (Δf) = f1 - f0 = 564,48 Hz.
* Duración de símbolo (Tb) = = 1,77ms.

Los valores de f1 y f0 utilizados se detallarán en las posteriores secciones para cada uno de los escenarios de diseño implementados, ya que vendrán definidos de la frecuencia de la señal portadora escogida.

En primer lugar, el sistema deberá mostrar un comportamiento tolerante ante los siguientes posibles efectos producidos por el canal de transmisión:

* Retardo de propagación arbitrario.
* Ruido blanco y gaussiano.
* Dispersión de hasta 180 μs de exceso de retardo.
* Presencia de perturbación de banda estrecha intencionada.

Por otro lado, el sistema deberá tolerar posible diferencias en las frecuencias de portadora (fc) o de símbolo (f1 o f0) en el transmisor y en el receptor de acuerdo a unas tolerancias dadas. Estas posibles variaciones vienen dadas a partir de las siguientes indicaciones de tolerancia:

* Velocidad binaria (Rb) = 564,48 bps ± 2,5%.
* Separación de frecuencias (Δf) = 564,48 Hz ± 7%.

# Comunicación Monoportadora

En primer lugar se llevará a cabo el diseño de un transmisor nominal y un receptor para portadora fija. Para ello, los valores utilizados desde el punto de vista de las prestaciones de transmisión vienen definidos a partir de los siguientes requisitos:

* La frecuencia portadora (fc) utilizada deberá ser igual a 7.5·Rb = 4.233 kHz.
* La frecuencia de los símbolos ‘1’ deberá ser igual a 7·Rb = 3.951 kHz.
* La frecuencia de los símbolos ‘1’ deberá ser igual a 8·Rb = 4.515 kHz.
* La separación de frecuencias (Δf) deberá ser igual a la velocidad binaria (Rb) del sistema.
* La frecuencia de muestreo (fs) utilizada será de 56·Rb = 31.611 kHz.

A lo largo de las siguientes secciones se describirá la arquitectura y el funcionamiento del sistema implementado.

## Transmisor

Una vez definidos tanto el mensaje de ‘0’s y ‘1’s a transmitir como las frecuencias de símbolo, el transmisor transformará la información binaria en una señal modulada en frecuencia. Además, se impondrá la condición de que en las transiciones entre pulsos la fase deberá ser continua, por lo que será necesario llevar el seguimiento de la misma.

Para llevar a cabo este procedimiento, se ha optado por llevar a cabo el seguimiento de la fase de la señal para cada uno de los símbolos, de manera que la fase inicial del siguiente símbolo sea la final del anterior símbolo. Para ello, se computará la variación de fase entre intervalos de muestreo de la siguiente manera:

= ω1

= ω0

Una vez calculada esta variación de fase, se establecerá de manera arbitraria la fase inicial de la señal a transmitir y el transmisor procederá a la generación un vector que contenga la fase de la señal para cada instante temporal de muestreo. Conocida la duración de cada símbolo (Tb), el parámetro de variación de fase correspondiente al símbolo a transmitir se irá sumando al estado de fase anterior un determinado número de veces en base a la cantidad de instantes de muestreo contenidos en este intervalo de duración del símbolo Tb. La última actualización de fase del símbolo anterior se corresponderá a la fase inicial del siguiente símbolo, independiente de si es un ‘1’ o un ‘0’. Este proceso se realizará iterativamente hasta finalizar la generación del vector de fase de la señal. Por último, la señal modulada se podrá obtener simplemente a partir del coseno del vector obtenido.

## Canal

Como se ha visto previamente, el canal genera cuatro tipos de distorsiones que el sistema ha de ser capaz de soportar.

### Retardo

Como ocurre en cualquier canal de transmisión, la señal se retarda de manera arbitraria. Sin embargo, en este sistema de comunicación, el retardo debería ser transparente, ya que no hay comunicación en doble sentido ni la comunicación es a tiempo real. La única complicación que causa es detectar el origen de tiempos. Para mitigarlo, habrá que incorporar tramas para detectar el inicio de la trasmisión.

En el código el retardo del canal se calcula a partir del retardo de trasmisión de la señal para una distancia dada, si bien se podría indicar directamente el retardo.

### Ruido

El canal añadirá señales no deseadas a la transmisión. Como mínimo, se tratará de ruido blanco gaussiano. Se genera ruido blanco gaussiano de media cero, con una potencia arbitraria, y se suma a la señal. Cuanta más potencia de ruido sea capaz de soportar el sistema, mejor será el sistema. Es por esto que se estudian diferentes técnicas en la recepción, como se verá más adelante.

### Dispersión

El canal introducirá una dispersión de hasta 180µs de retardo. Esto se modelará como dos deltas: la primera es la señal directa y la segunda tendrá la misma amplitud y estará retardada hasta 180µs. Sin embargo, el programa se ha planteado para un caso más amplio: arbitrarias señales de multitrayecto con amplitudes arbitrarias. Se especifica el retardo máximo que pueden tener las réplicas de las señales, el número de reflexiones y la amplitud de estas. Las reflexiones se distribuyen aleatoriamente en el rango temporal que se ha especificado, y luego se ajustan las amplitudes (un caso general puede ser amplitud aleatoria pero siempre menor que la señal directa).

### Interferencias intencionadas

La técnica más común y sencilla de interferir en un sistema es el jamming. Esto se consigue introduciendo señales indeseadas de gran potencia en toda la banda usada por el sistema de comunicaciones. Las especificaciones indican que habrá interferencias que contengan como mucho la potencia original de la señal. Esto se entiende como que la potencia de transmisión será la misma, pero en todo el ancho de banda usado en el sistema, por lo que la densidad espectral de potencia de las interferencias será menor.

La potencia de una sinusoide, como es el caso de la señal deseada, es su voltaje RMS al cuadrado dividido entre la resistencia en la que se disipa la potencia. La resistencia se asumirá 1 para simplificar los cálculos, ya que no es relevante al estar comparándose dos potencias medidas en la misma resistencia. El voltaje RMS de un seno es su voltaje de pico entre √2. El voltaje de pico en el sistema es 1 V, por lo que la potencia serán 0.5 W.

El sistema usa un ancho de banda (BW) de Fc±Rb, es decir, 2\*Rb. El sistema interferente tendrá que usar la misma potencia en ese ancho de banda. Esto nos deja una densidad espectral de potencia de interferencia:

Para generar esta interferencia se generará ruido blanco gaussiano de media cero con varianza:

Este ruido será filtrado para que solo haya interferencia en la banda de trabajo con un filtro de un orden alto para que tenga cortes pronunciados.

Sin embargo, se puede apreciar que la interferencia tendrá muy poco efecto, estando la mayoría de las veces por debajo del ruido blanco. Esto se debe a que la señal concentra toda la potencia en una única sinusoide, mientras que la interferencia concentra la misma potencia en una banda bastante amplia.

## Receptor

Tras pasar por el canal de transmisión, la señal se recibirá en el receptor, que tratará de recuperar los símbolos enviados por el transmisor. Para ello, se han estudiado las 3 diferentes alternativas de recepción que se describen en los siguientes apartados.

### Filtrado Paso Banda & Detección de Envolvente

Este método de recepción realizará una detección homodina no coherente de la señal. La siguiente figura esquematiza la implementación de este modo de recepción.

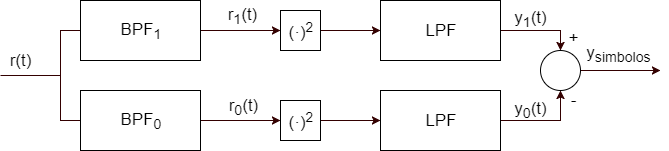


Figura 2. Esquema de Filtrado Paso Banda & Detección de Envolvente.

En primer lugar, se implementarán filtros paso banda para cada una de las frecuencias de símbolo. La señal recibida se filtrará de manera independiente con ambos filtros, de manera que se obtengan dos señales diferentes en las cuales estará contenida únicamente la señal correspondiente a uno de los dos símbolos. Finalmente, se buscará la envolvente de los símbolos utilizando un detector de ley cuadrática que combina una etapa de no-linealidad y un filtro paso bajo.

Las características de los 3 filtros implementados (el filtro paso bajo será el mismo para las dos señales) se resumen en la siguiente tabla:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Filtro Pasobanda ‘1’ | Filtro Pasobanda ‘0’ | Filtro Pasobajo |
| Orden | 2 | 2 | 2 |
| fcorte,1 (Hz) | fc-Rb | fc | Rb |
| fcorte,2 (Hz) | fc | fc+Rb | - |

Tabla 1. Descripción de los parámetros de los filtros.

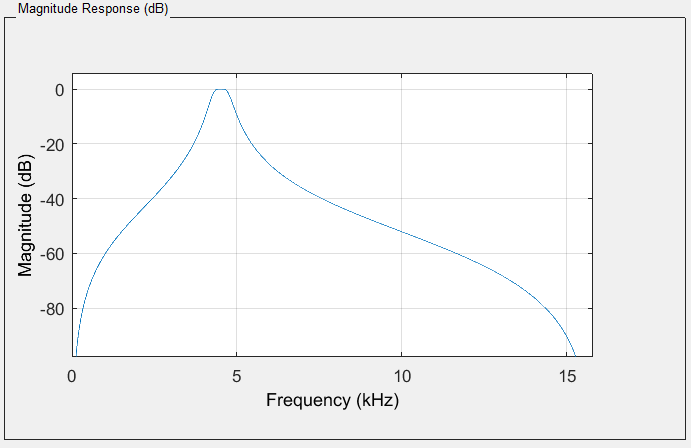
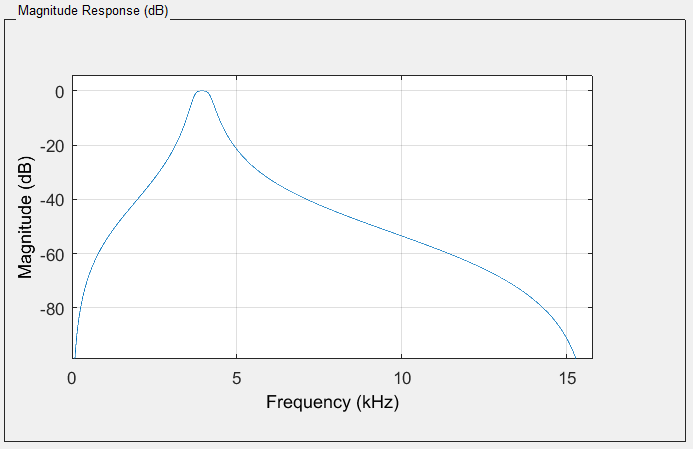


Figura 3. Respuesta en magnitud del filtro paso banda del símbolo '1' (izquierda) y símbolo '0' (derecha).

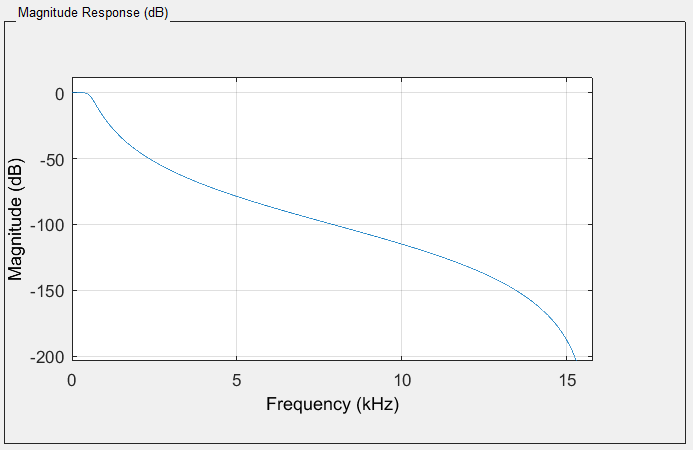
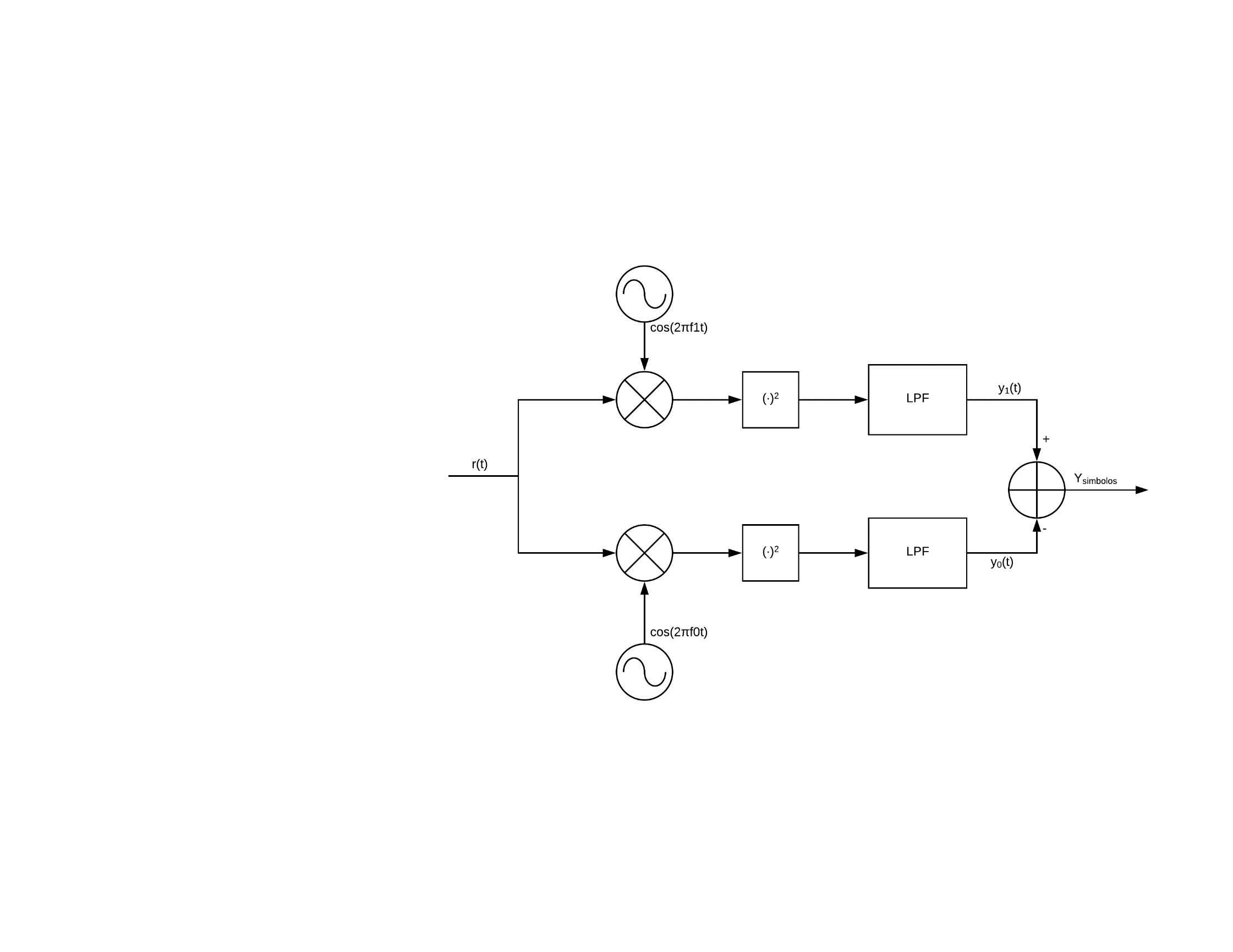


Figura 4. Respuesta en magnitud del filtro paso bajo.

### Demodulación (

Este método sigue una arquitectura de receptor heterodino, usando osciladores locales para la recepción. Sin embargo, es un sistema no coherente, es decir, estos osciladores no están enganchados en fase con el receptor. Esto puede causar graves consecuencias si los osciladores están desfasados. Este método queda obsoleto con el método 2.3.3, que usa PLL. Se hizo como una primera versión, más rápida y sencilla que los PLL, para poder probar el sistema completo.

En el siguiente diagrama se puede ver la arquitectura del receptor.



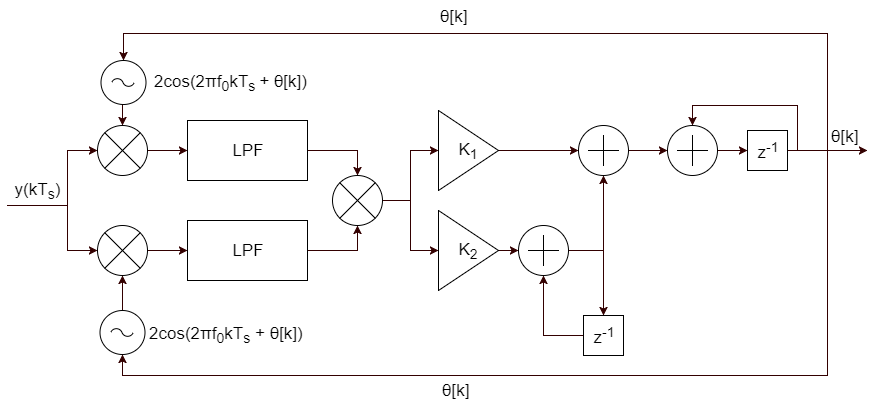
Se generan tonos a la frecuencia de cada símbolo, que al ser mezclados con la señal la bajan a banda base. Sin embargo, también mezclaran componentes no deseadas, que hay que filtrar con un filtro paso bajo. Este filtro ha de ser lo suficiente rápido como para dejar pasar los símbolos, pero nada más. Es por esto, que la frecuencia de corte del LPF será Rb.

### Demodulación Coherente utilizando PLLs sintonizados.

Al igual que en método descrito en la sección anterior, esta modalidad implementará una arquitectura de recepción heterodino, usando osciladores locales para la recepción. Sin embargo, este modo implementará un método de recepción coherente en el que se tendrá en cuenta la fase con la que se recibe la señal.

Para ello, se ha implementado un bucle de Costas … (LO DEJO PARA MAS ADELANTE)

Dado que los ‘1’s y los ‘0’s enviados corresponden a diferentes frecuencias, se implementarán dos PLLs sintonizados a la frecuencias de cada uno de los símbolos, de manera que cada uno de ellos pueda sintonizar la



Utilizando el vector de fases estimado por el bucle, se generará la señal coseno que se utilizará como oscilador local para demodular la señal a banda base al igual que en la sección anterior, donde el coseno utilizado estará sincronizado en fase con la de la señal.

## Comunicación de espectro ensanchado