

ANEXO B. IMPLEMENTACION DE LA SIMULACIÓN

INTRODUCCIÓN

El presente anexo muestra los aspectos más relevantes de la implementación en el entorno Simulink de Matlab® del modelo de simulación del nivel físico de un enlace unidireccional de un sistema basado en el estándar IEEE 802.11b. Esta implementación se basa en la metodología desarrollada en el Capítulo 2 de este documento y en el estándar IEEE 802.11b. La descripción de los bloques y parámetros principales de la simulación se divide principalmente en tres partes: Transmisor, Canal y Receptor

1. TRANSMISOR

El Transmisor comprende tanto la fuente de datos a transmitir como los procesos de formación del PDU y el modulador-ensanchador (Ver figura 1).

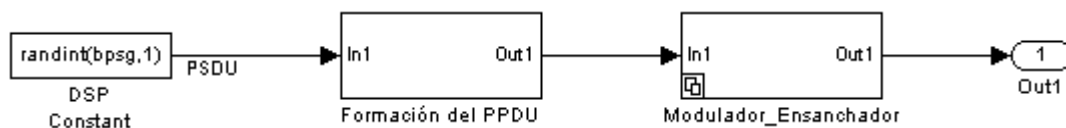


Figura 1. Transmisor

1.1 FUENTE DE DATOS

Para simular la señal de información de usuario que será enviada a través del sistema se utiliza el bloque “*DSP Constant*” del *blockset*¹ procesamiento de señales (Ver figura 2).

¹ *Blockset* es un conjunto de bloques para aplicaciones específicas del entorno Simulink de Matlab®.



Figura 2. Generador de datos binarios aleatorios

Descripción: Este bloque genera números binarios aleatorios utilizando la función *randint()*, esta función produce unos y ceros con probabilidad de $\frac{1}{2}$ para cada uno.

Parámetros:

- Valor constante: Especifica el valor a generar. Este parámetro puede ser una expresión de MATLAB que represente un escalar, un vector o una matriz.
- Modo de muestreo: Especifica el modo de muestreo de la salida. Puede ser continuo o discreto.
- Salida: Especifica cuando la salida es basada en muestras o basada en tramas.
- Tiempo de muestreo: Especifica el periodo de muestreo para la salida basada en muestras. Cuando se selecciona que el parámetro salida esta basado en tramas, este parámetro es llamado Periodo de Trama, y especifica la duración de cada trama.

1.2 FORMACIÓN DEL PPDU

La figura 3 muestra la implementación del bloque formación del PPDU definido en el modelo de simulación del Capítulo 2. Se implementa a través de dos procesos o bloques internos, los cuales se describen a continuación.

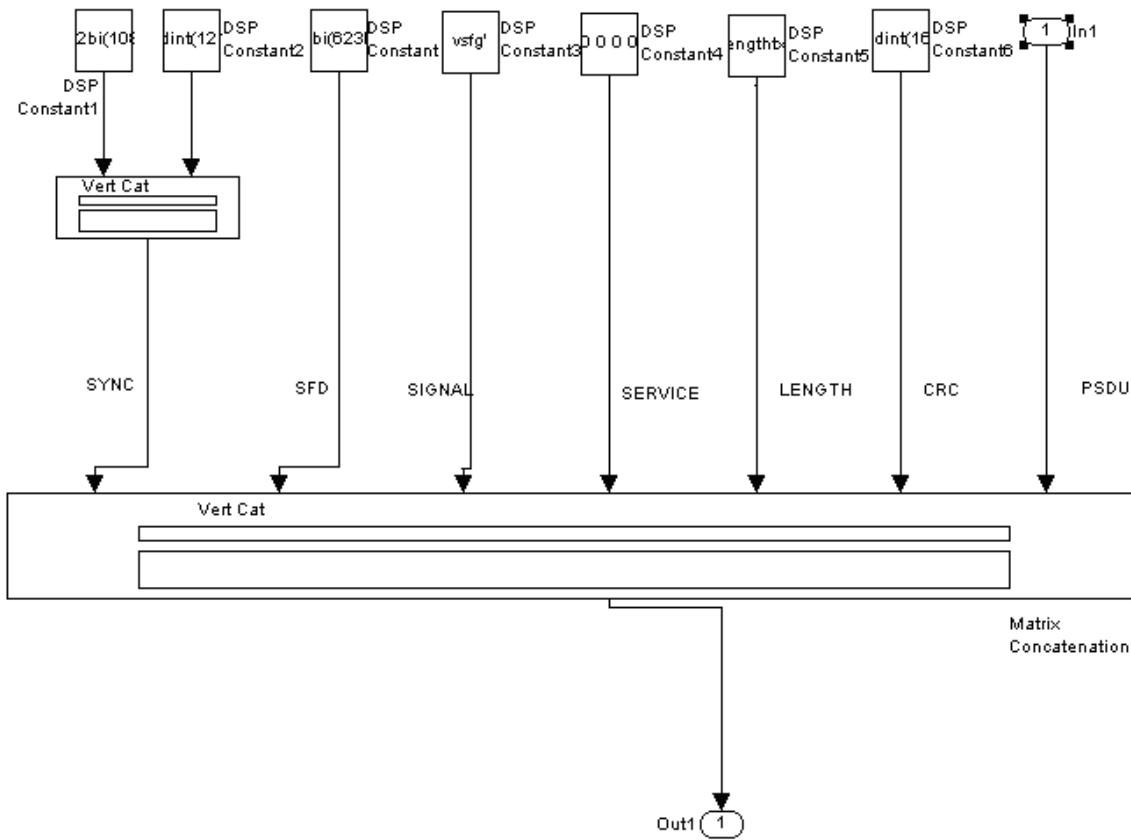


Figura 3. Implementación de la formación del PDU en Simulink

1.2.1 Agregador de bits según el formato de cada campo de la trama PDU

Para implementar este proceso se utilizan el bloque “*DSP Constant*” (Ver figura 4).

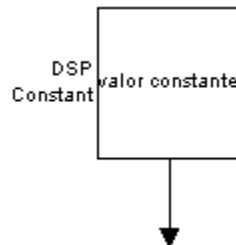


Figura 4. Agregador de bits según formato PDU

- Descripción: Este bloque genera una señal de valor constante durante la simulación.

Sus parámetros son los mismos de la sección 1.1.

1.2.2 Concatenar

Para terminar de implementar el proceso de formación del PPDU se utilizan el bloque “*Matriz Concatenation*” (Ver figura 5).

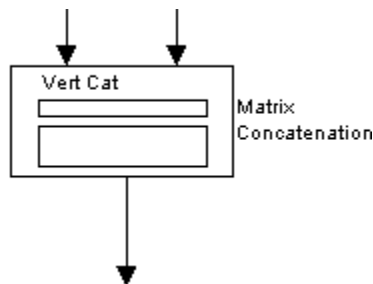


Figura 5. Concatenador vertical

Descripción: Este bloque concatena entradas de forma vertical u horizontal.

Parámetros:

- Número de entradas: Número de matrices a concatenar.
- Método de Concatenación: Puede ser horizontal o vertical

1.3 MODULADOR-ENSANCHADOR

Este bloque se simula de forma diferente dependiendo de las 4 velocidades de transmisión de datos que maneja el estándar.

1.3.1 Modulador-Ensanchador para 1 Mbps

La figura 6 muestra la implementación del bloque Modulador-Ensanchador para 1 Mbps definido en el modelo de simulación del Capítulo 2. Se implementa a través de tres procesos o bloques internos, los cuales se describen a continuación.

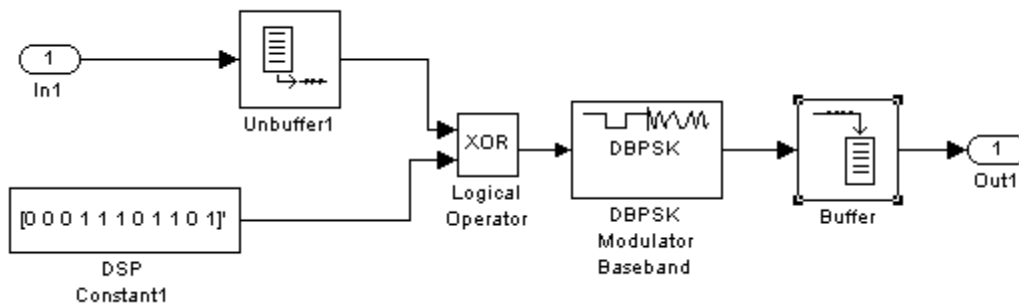


Figura 6. Bloque Modulador-Ensanchador para 1 Mbps

1.3.1.1 Unbuffer

Para tomar cada bit que viene desde de la trama PDU y realizar la operación lógica con el código barker, se utiliza el bloque “Unbuffer”.

Descripción: Este bloque toma una entrada basada en tramas y obtiene en la salida un escalar.

1.3.1.2 Código Barker

Para implementar este proceso se utilizan el bloque “DSP Constant” del Blockset de procesamiento de señales (Ver figura 7).

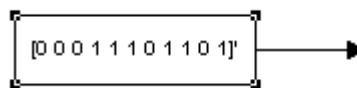


Figura 7. Constante Barker

Descripción: Este bloque genera una señal de valor constante que simula la secuencia o código Barker durante la simulación.

Parámetros:

- Valor constante: Especifica la constante a generar. En este caso este parámetro es un vector que representa el código Barker.

1.3.1.3 Operador Lógico

Para realizar el proceso de ensanchamiento se realiza la operación lógica XOR entre la secuencia Barker y la trama PPDU. El bloque que se utiliza para implementar este proceso es el “*Logical Operator*”.

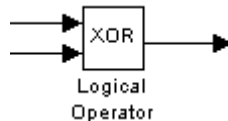


Figura 8. Operación lógica XOR

Descripción: Este bloque realiza la operación lógica especificada en sus señales de entrada.

Parámetros:

- Operador: Determina la operación lógica que se aplica a las entradas del bloque.
- Número de entradas: Especifica el número de entradas del bloque.

1.3.1.4 Modulador DBPSK

Para implementar este proceso se utiliza el bloque “*DBPSK Modulator BaseBand*” del *blockset* de Comunicaciones (*Communications Blockset*), este aparece en la figura 9.



Figura 9. Modulador DBPSK

Descripción: El bloque modulador DBPSK de banda base modula utilizando el método de desplazamiento de fase binaria diferencial. La salida es una representación de la banda base modulada.

Parámetros:

- Rotación de fase (*Phase rotation*): Es la diferencia de fase entre el símbolo anterior y actual cuando la entrada es cero.

1.3.2 Modulador-Ensanchador para 2 Mbps

Como se vio en el capítulo 1 el encabezado y el preámbulo para el modo “*long PLCP*” siempre se envía a una velocidad de 1 Mbps, únicamente los datos de usuario (los datos provenientes del “*Bernoulli Binary Generator*”) se envían a diferentes velocidades. Este bloque se divide en dos partes, como se muestra en la figura 10.

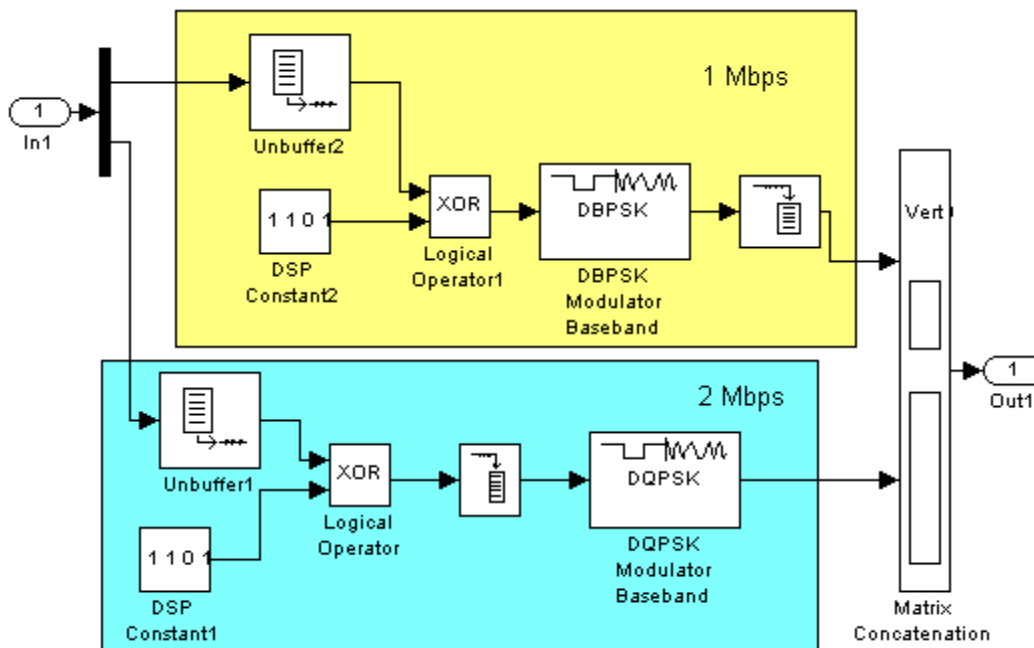


Figura 10. Bloque Modulador-Ensanchador para 2 Mbps

La parte amarilla es la misma que la explicada en la sección 1.3.1. La parte azul de la figura 10 se implementa a través de los procesos que se describen a continuación.

1.3.2.1 Código Barker

Su descripción y parámetros son los mismos de la sección 1.3.1.1.

1.3.2.2 Operador Lógico

Su descripción y parámetros son los mismos de la sección 1.3.1.2.

1.3.2.3 Modulador DQPSK

Para implementar este proceso se utiliza el bloque “*DQPSK Modulator BaseBand*”, este aparece en la figura 11.



Figura 11. Modulador DQPSK

Descripción: El bloque modulador de banda base DQPSK modula utilizando el método de desplazamiento de fase cuaternaria diferencial. La salida es una representación banda base de la señal modulada.

Parámetros:

- Tipo de entrada (*Input type*): Indica si la entrada consiste de enteros o de pares de bits
- Ordenamiento de la constelación (*Constellation ordering*): Determina como el bloque mapea cada par de bits de entrada a su entero correspondiente,

que posteriormente convertirá en números complejos para representar los puntos de la constelación DQPSK. En la simulación, el ordenamiento de la constelación deberá ser presentada de acuerdo al Código Gray, como se definió en el anexo A.

- Rotación de fase (*Phase rotation*): Es la diferencia de fase entre el previo y actual símbolo modulado cuando la entrada es cero.

1.3.2.4 Concatenación

Para unir la trama PLCP que se envía a 1 Mbps y la información del usuario que se envía a 2 Mbps se utiliza el bloque “*Matrix Concatenation*”.

Descripción: Este bloque concatena entradas de forma vertical u horizontal.

Parámetros:

- Número de entradas: Número de matrices a concatenar.
- Método de Concatenación: Puede ser horizontal o vertical

1.3.3 Modulador-Ensanchador para 5.5 Mbps

Este bloque se divide en dos partes, como se muestra en la figura 12. La parte amarilla es la misma que la explicada en la sección 1.3.1. La parte azul de la figura 12 se implementa a través de los procesos que se describen a continuación.

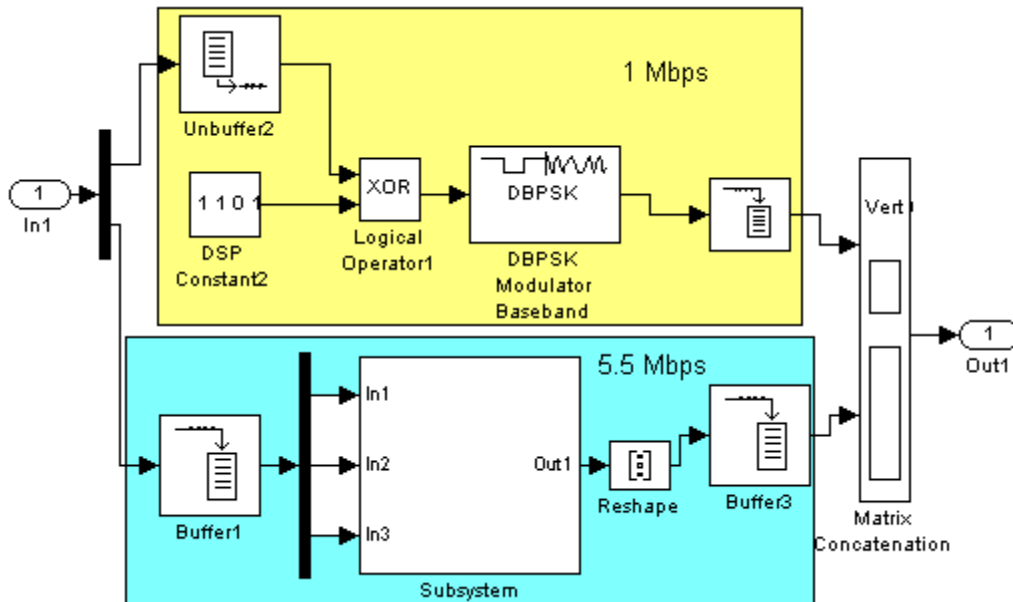


Figura 12. Bloque Modulador-Ensanchador para 5.5 Mbps

1.3.3.1 Buffer

Para que los datos de usuario sean enviados a una velocidad de 5.5 Mbps, los datos de usuarios son codificados en grupos de 4 bits, como se explica en el anexo A. Para formar las tramas de 4 bits se utiliza el bloque “*Buffer*” que se muestra en la figura 13.

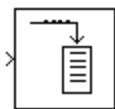


Figura 13. Bloque Buffer

Descripción: Este bloque recibe un flujo de bits y lo adapta en tramas. Se redistribuye las muestras de entrada en tramas de diferente tamaño, que pueden ser más largas o cortas que la trama original.

Parámetros:

- Tamaño del buffer de salida (*Output buffer size*): Determina el número de muestras consecutiva que formaran la trama de salida. En la simulación este valor corresponde a 4.

1.3.3.2 Demux

El estándar IEEE 802.11b define que para la velocidad de 5.5 Mbps los 2 primeros bits de los 4 que son codificados sean modulados en banda base utilizando DQPSK, y que a los otros dos bits se les aplique un procedimiento diferente para encontrar ϕ_1 , ϕ_2 , ϕ_3 y ϕ_4 . Para realizar la separación de los dos primeros bits el bloque utiliza el bloque “Demux” que se muestra en la figura 14.



Figura 14. Bloque Demux para 5.5 Mbps

Descripción: El bloque Demux extrae los componentes de una señal de entrada y los separa en diferentes señales de salida.

Parámetros:

- Número de salidas (*Number of outputs*): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a [2 1 1].

1.3.3.3 Subsystem (codificador CCK para 5.5 Mbps)

Para simular la codificación CCK para 5.5 Mbps se utilizan los bloques “Modulador DQPSK”, “Ganancia”, “Constante DSP”, “Sumador”, “Complejo a Magnitud-Angulo”, “Exponencial Complejo”, “Concatenación Horizontal” como se muestra en la figura 15

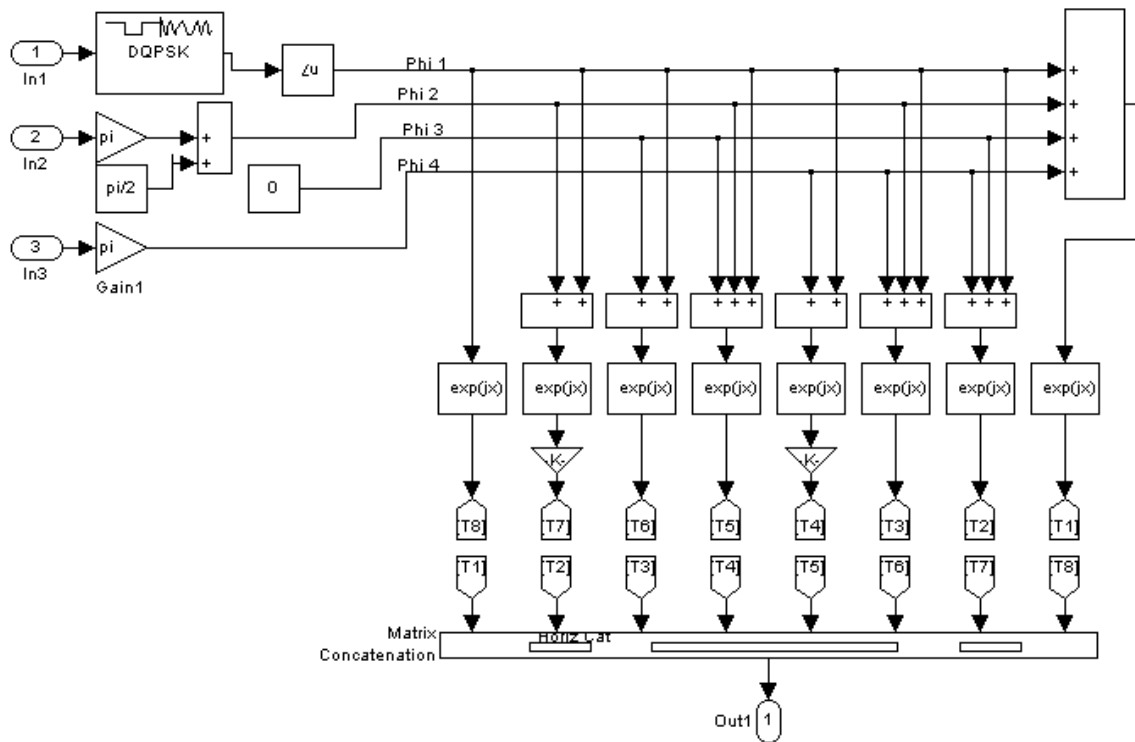


Figura 15.Codificador CCK

1.3.3.3.1 Modulador DQPSK

Su descripción y parámetros son los mismos de la sección 1.3.2.3.

1.3.3.3.2 Complejo a Magnitud-Angulo

A la salida del modulador DQPSK se tiene una señal compleja que representa una forma de onda, únicamente se toma el ángulo que representa el cambio de fase de la forma de onda y se le asigna a φ_1 , para implementar este proceso se utiliza el bloque “Complex to Magnitude-Angle”, el cual aparece en la figura 16.



Figura 16. Complejo a Magnitud-Angulo

Descripción: Este bloque acepta señales compuestas de valores complejos. El parámetro “*Output*” se configura de tal forma que en la salida del bloque se obtenga el ángulo que representa el cambio de fase, como se observa en la figura 17.



Figura 17. Complejo a Angulo

Parámetros:

- Salida (*Output*): Este parámetro determina la salida del bloque, se puede variar para que en la salida se obtenga la magnitud y el ángulo, únicamente la magnitud o únicamente el ángulo.

1.3.3.3 Ganancia

Para encontrar φ_2 el tercer bit de la trama de 4 bits se multiplica por π y luego se le suma $\pi/2$. Para multiplicar por π se utiliza el bloque “*Gain*”, este aparece en la figura 18.



Figura 18. Complejo a Magnitud-Angulo

Descripción: Este bloque multiplica la entrada por un valor constante. La entrada y la ganancia puede ser de tipo escalar, vectorial o matricial.

Parámetros:

- Ganancia (*Gain*): Especifica el valor por el cual debe ser multiplicada la entrada. En la simulación este valor corresponde a π (π).

- Multiplicación (*Multiplication*): Especifica el modo de la multiplicación que en el caso de la simulación corresponde a *Element-wise* ($K*u$), en donde cada elemento de la entrada es multiplicado por cada elemento de la ganancia.

1.3.3.3.4 Constante DSP

Como se explico en la sección 1.3.3.3.4 para encontrar el valor de φ_2 además de realizar la multiplicación por π se le suma $\pi/2$. En la simulación, para obtener el valor de $\pi/2$ se utiliza el bloque “*DSP Constant*”. El bloque aparece en la figura 19.



Figura 19. Constante $\pi/2$.

La descripción de este bloque se realizó en la sección 1.2.1.

1.3.3.3.5 Sumador

Para realizar la suma de $\pi/2$ y el bit 3 multiplicado por π y la suma de los ángulos φ se utiliza el bloque “*Sum*” que aparece en la figura 20.



Figura 20. Sumador.

Descripción: Este bloque realiza la suma o la resta de las señales de entrada; puede sumar o restar entradas escalares, vectores o matrices.

Parámetros:

- Lista de signos (*List of signs*): En este parámetros se introducen los caracteres “+” o “-” según el número y la operación que se desee realizar

con las entradas del bloque. La operación por defecto es la adición, si se quiere realizar esta operación con las señales de entrada del bloque únicamente se escribe el número de las señales de entrada.

1.3.3.3.6 Exponencial Complejo

Calcula la función exponencial compleja según la formula (1) del capítulo 2 para obtener cada termino de la palabra código. El bloque “*Complex Exponential*” aparece en la figura 21.



Figura 21. Exponencial Complejo.

Descripción: Este bloque calcula la función exponencial compleja (y) para cada elemento de una entrada real u .

$$y = e^{j\omega t} \quad (1)$$

La salida es compleja. Con el mismo tamaño de la entrada

1.3.3.3.7 Concatenación Horizontal

Para la formación del símbolo se utiliza el bloque “*Matrix Concatenation*”. Este bloque aparece en la figura 22.

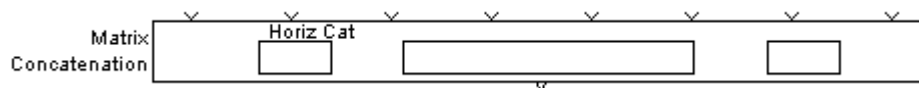


Figura 22. Concatenación Horizontal.

1.3.4 Modulador-Ensanchador para 11 Mbps

Este bloque se divide en dos partes, como se muestra en la figura 23. La parte amarilla es la misma que la explicada en la sección 1.3.1. La parte azul de la figura 22 se implementa a través de los procesos que se describen a continuación.

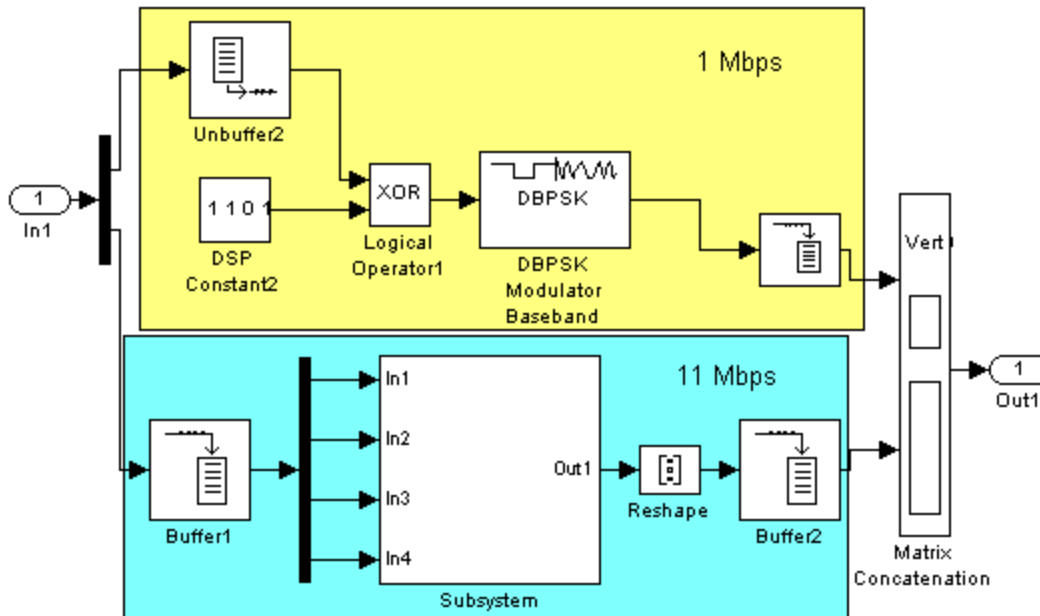


Figura 22. Bloque Modulador-Ensanchador para 11 Mbps

1.3.4.1 Buffer

Para que los datos de usuario sean enviados a una velocidad de 11 Mbps, los datos de usuarios son codificados en grupos de 8 bits, como se explico en el anexo A. Para formar las tramas de 8 bits se utiliza el bloque “*Buffer*” que se muestra en la figura 23.

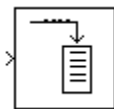


Figura 23. Bloque *Buffer*

Descripción: Este bloque recibe un flujo de bits y lo adapta en tramas. Se redistribuye las muestras de entrada en tramas de diferente tamaño, que pueden ser más largas o cortas que la trama original.

Parámetros:

- Tamaño del *buffer* de salida (*Output buffer size*): Determina el número de muestras consecutiva que formaran la trama de salida. En la simulación este valor corresponde a 8.

1.3.4.2 Demux

El estándar IEEE 802.11b define que para la velocidad de 11 Mbps los 2 primeros bits de los 8 que son codificados sean modulados en banda base utilizando DQPSK, y que a los otros 3 grupos de 2 bits modulen en banda base utilizando QPSK para encontrar ϕ_1 , ϕ_2 , ϕ_3 y ϕ_4 . Para formar los 4 grupos de 2 bits se emplea el bloque “Demux” que se muestra en la figura 24.



Figura 24. Bloque Demux para 11 Mbps

Descripción: El bloque Demux extrae los componentes de una señal de entrada y los separa en diferentes señales de salida.

Parámetros:

- Número de salidas (*Number of outputs*): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a [2 2 2 2].

1.3.4.3 Subsystem (codificador CCK para 11 Mbps)

Para simular la codificación CCK para 11 Mbps se utilizan los bloques “Modulador DQPSK”, “Modulador QPSK”, “Sumador”, “Complejo a Magnitud-Angulo”,

“Exponencial Complejo”, “Concatenación Horizontal” como se muestra en la figura 25

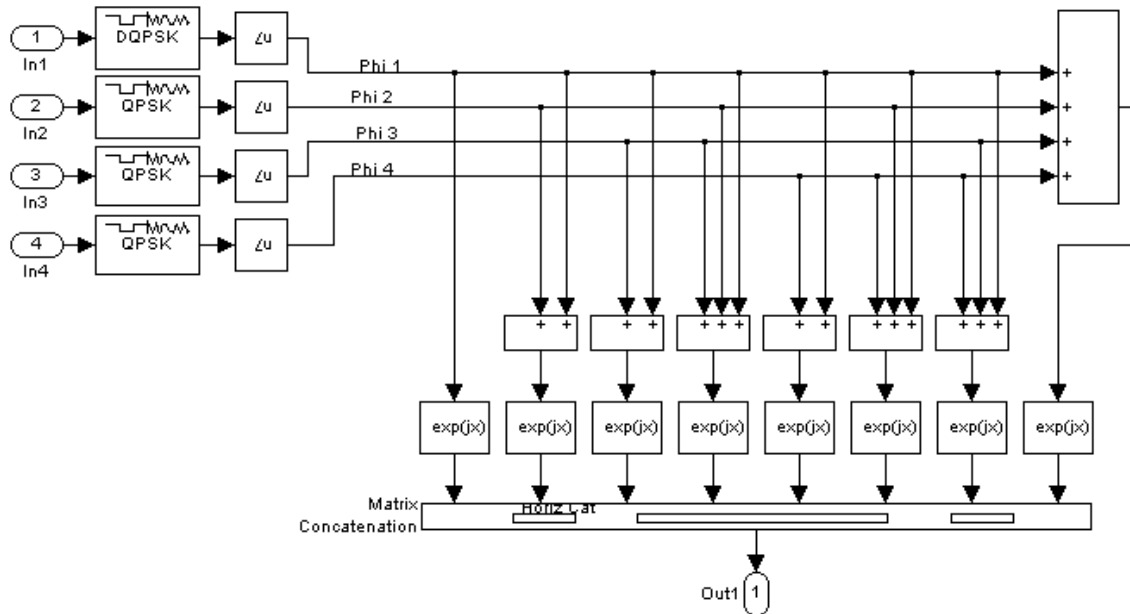


Figura 25 Codificador CCK

1.3.4.3.1 Modulador DQPSK

Su descripción y parámetros son los mismos de la sección 1.3.2.3.

1.3.4.3.2 Complejo a Magnitud-Angulo

A la salida del modulador DQPSK y de los moduladores QPSK se tiene una señal compleja que representa una forma de onda, únicamente se toma el ángulo que representa el cambio de fase de la forma de onda y se le asigna a φ_1 , φ_2 , φ_3 y φ_4 respectivamente, para implementar este proceso se utiliza el bloque “Complex to Magnitude-Angle”. La descripción de este bloque aparece en la sección 1.3.3.3.2.

1.3.4.3.3 Modulador QPSK

Para implementar este proceso se utiliza el bloque “QPSK Modulator BaseBand” del Blockset de Comunicaciones, este aparece en la figura 26.



Figura 26. Modulador QPSK

Descripción: El bloque modulador de banda base QPSK modula utilizando el método de desplazamiento de fase cuaternaria. La salida es una representación de la banda base modulada.

Parámetros:

- Tipo de salida (Output type): Determina si la salida es del tipo entero o bits. En la simulación este parámetro está establecido como bit.
- Ordenamiento de la constelación (Constellation ordering): Determina como el bloque mapea cada entero en un par de bits de salida y los convierte en números complejos para representar los puntos de la constelación QPSK. En la simulación, la conversión deberá ser presentada de acuerdo al mapeo de la constelación del código binario, teniendo en cuenta el anexo A.
- Fase de desplazamiento (Phase offset): En este campo se debe indicar el valor inicial de la fase en radianes del punto cero de la constelación de la señal. En la simulación este valor corresponde a $\pi/4$.

1.3.4.3.4 Sumador

Para realizar la suma de los ángulos φ se utiliza el bloque “Sum”, su descripción se detalla en la sección 1.3.3.3.5

1.3.4.3.5 Exponencial Complejo

La descripción de este bloque aparece en la sección 1.3.3.3.6

1.3.4.3.6 Concatenación Horizontal

La descripción de este bloque aparece en la sección 1.3.3.3.7

2. CANAL DE TRANSMISION

En el capítulo 2 se definió que en el canal de transmisión la señal estaría afectada por la multitrayectoria (4 trayectos) y por Ruido Blanco Gaussiano Aditivo (AWGN). La implementación interna del canal de transmisión contiene los siguientes módulos: “Camino 1”, “Camino 2”, “Camino 3”, “Camino 4” y el bloque “Matrix Concatenation”

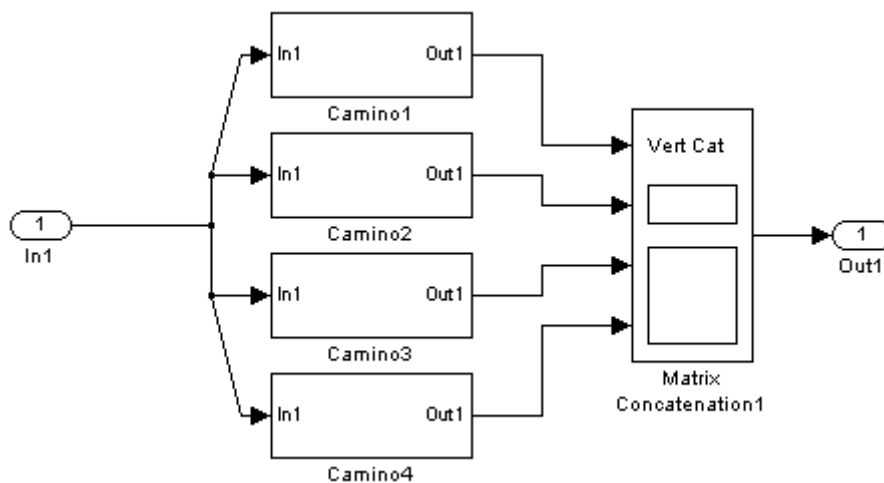


Figura 27. Canal de transmisión

2.1 CAMINO 1

Para realizar la simulación del primer camino se utilizan el bloque “AWGN Channel” del Blockset de comunicaciones, el cual aparece en la figura 28.

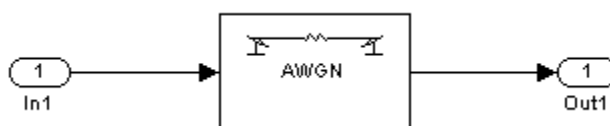


Figura 28. Camino 1

Descripción: El bloque "AWGN Channel" agrega ruido blanco Gaussiano (AWGN) a una señal de entrada real o compleja. Cuando la señal de entrada es real, este bloque agrega ruido real Gaussiano y produce una señal real de salida. Si la señal de entrada es una señal compleja, este bloque agrega ruido complejo Gaussiano y produce una señal compleja de salida.

Parámetros:

- Modo (Mode): Hace referencia a la manera en que se especifica el nivel de ruido que puede ser: relación energía de bit a densidad de potencia de ruido (E_b/N_0), relación energía de símbolo a densidad de potencia de ruido (E_s/N_0), relación señal a ruido (SNR) y variación de la máscara o variación del puerto.
- Semilla inicial (Inicial seed): La semilla para el generador de ruido Gaussiano.

2.2 CAMINOS 2, 3 Y 4

Para realizar la simulación de los caminos 2, 3 y 4 se utilizan el bloque "AWGN Channel" del Blockset de comunicaciones y el bloque "Integer Delay", el cual aparece en la figura 29.

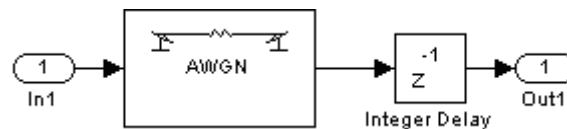


Figura 29. Caminos 2, 3 y 4.

El bloque "AWGN Channel" se describió en la sección 2.1.

2.2.1 Retardo

Para simular los ecos de la señal producidos por la multitrayectoria, se utiliza el bloque "Integer Delay".

Descripción: Este bloque retarda la señal de entrada en N periodos de tiempo. La entrada puede ser un escalar o un vector. Si la entrada es un vector todos los elementos del vector son retrasados por el mismo periodo de tiempo.

Parámetros:

- Número de retardos (Number of delays): Número de periodos para retardar una señal de entrada.

3 RECEPTOR

El Receptor comprende el receptor Rake, el demodulador y el conversor de tipo de dato (Ver figura 30).

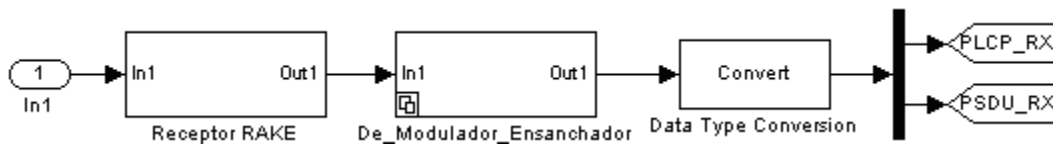


Figura 30. Receptor

3.1 RECEPTOR RAKE

Para la recepción de la señal generada en el transmisor y pasada a través del canal existen diferentes tipos de receptores, para la simulación se selecciono el receptor tipo RAKE. Este bloque comprende un Demux que maneja las señales que llegan al receptor por los diferentes caminos (4 en la simulación) y se los entrega al bloque que las correlaciona (Ver figura 31).

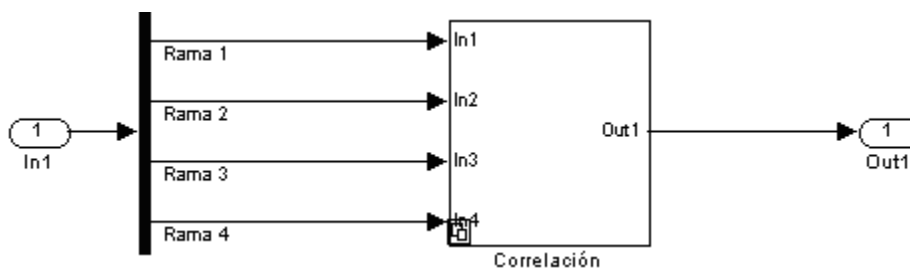


Figura 31 Receptor RAKE

3.1.1 Demux

Su descripción es la misma de la sección 1.3.3.2.

Parámetros:

- Número de salidas (Number of outputs): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a 4.

3.1.2 Correlación

Este bloque se simula de forma diferente dependiendo de las 4 velocidades que maneja el estándar.

3.1.2.1 Correlación a 1Mbps

El subsistema de Correlación a 1Mbps comprende un demux por cada entrada, sumadores, matriz de concatenación y bloques de ganancia. Luego de sumar las cuatro trayectorias recibidas de manera constructiva, se divide por cuatro para tener un promedio de la señal (Ver figura 32).

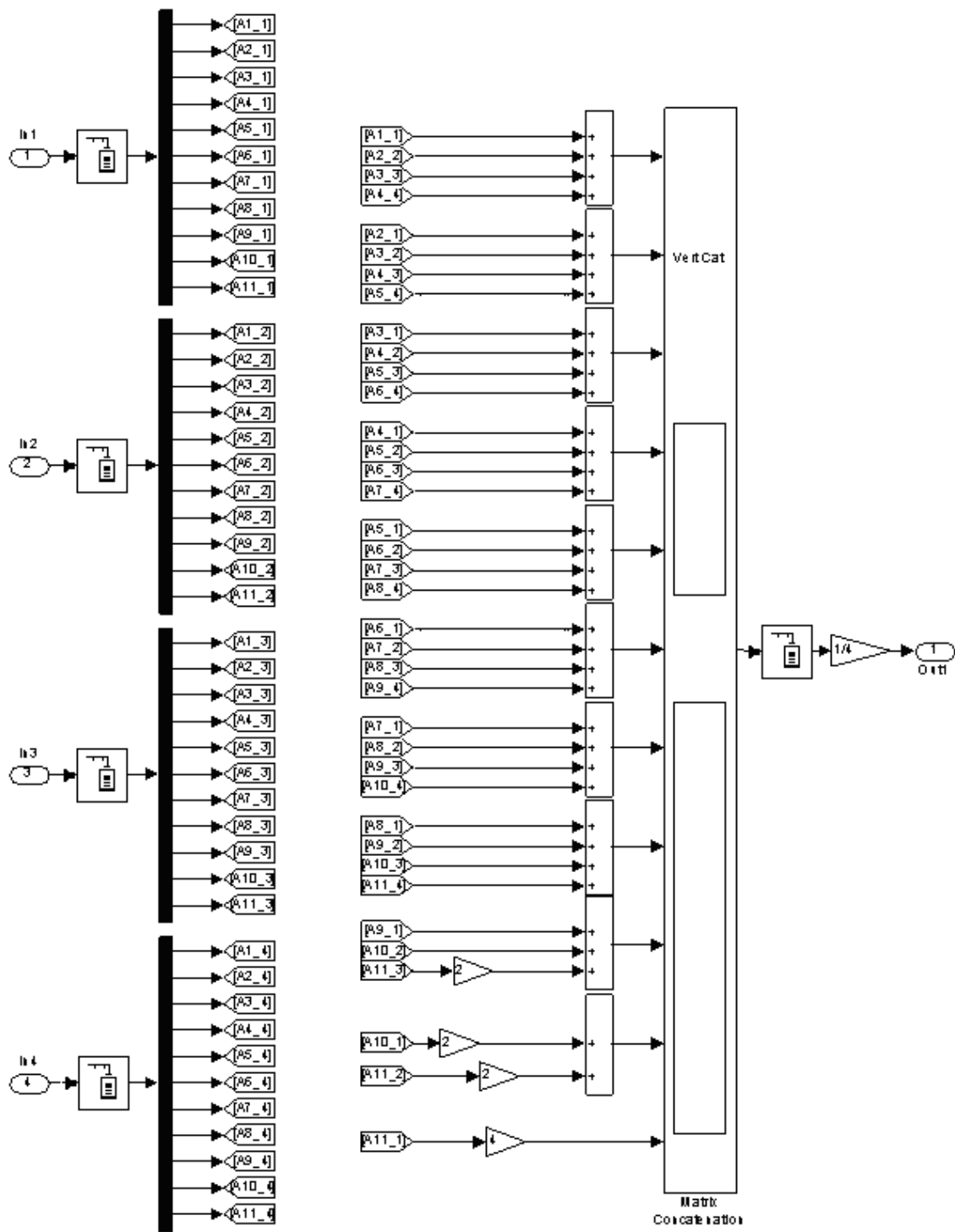


Figura 32 Correlación 1 Mbps

3.1.2.1.1 Demux

Cada Demux recibe una señal que llega al receptor por un determinado camino y la divide para el proceso de correlación. Su descripción es la misma de la sección 1.3.3.2.

Parámetros:

- Número de salidas (Number of outputs): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a 11.

3.1.2.1.2 Sumador

Cada sumador recibe las señales en fase de cada una de las trayectorias (4 en la simulación) y las suma. Su descripción y parámetros son los mismos de la sección 1.3.3.3.5.

3.1.2.1.3 Matriz de Concatenación

Con la concatenación se ordena nuevamente la señal transmitida. Su descripción es la misma de la sección 1.2.2.

3.1.2.1.4 Ganancia

A la señal que entrega la matriz de concatenación se le aplica una ganancia de $\frac{1}{4}$ para obtener una señal promedio. Es de $\frac{1}{4}$ por ser cuatro las señales que se suman en fase. Su descripción es la misma de la sección 1.3.3.3.3.

3.1.2.2 Correlación 2Mbps

El subsistema de Correlación a 2Mbps opera de manera similar al de 1Mbps especificado en 3.1.2.1, con diferencias en las dimensiones de la señal de entrada, que a su vez obliga la adición de algunos bloques sumadores y a cambiar algunos parámetros de los bloques de demultiplexación y concatenación, como se muestra en las figuras 33(a) y 33(b) (debido al tamaño la figura esta dividida en dos partes).

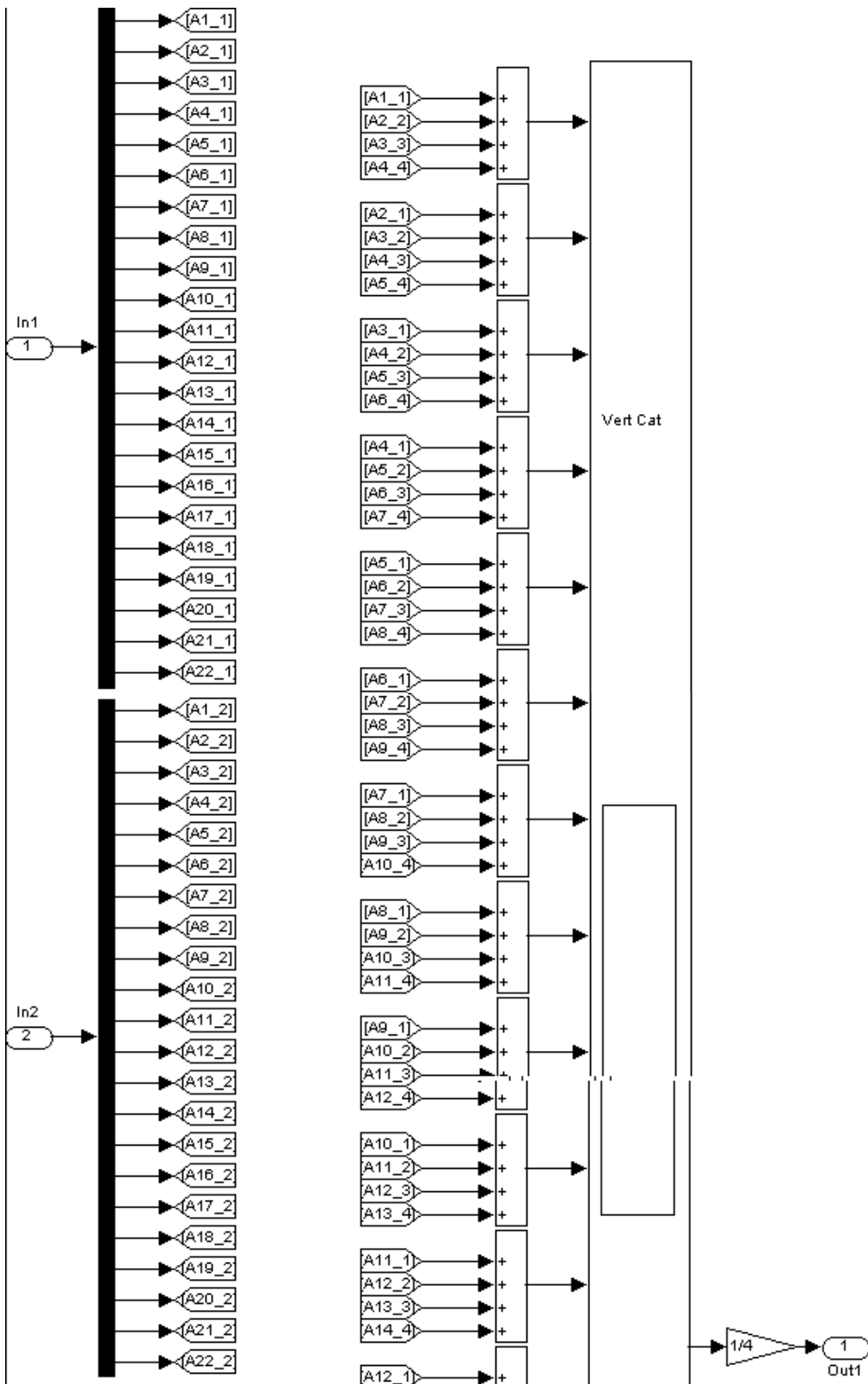


Figura 33(a) Correlación 2 Mbps

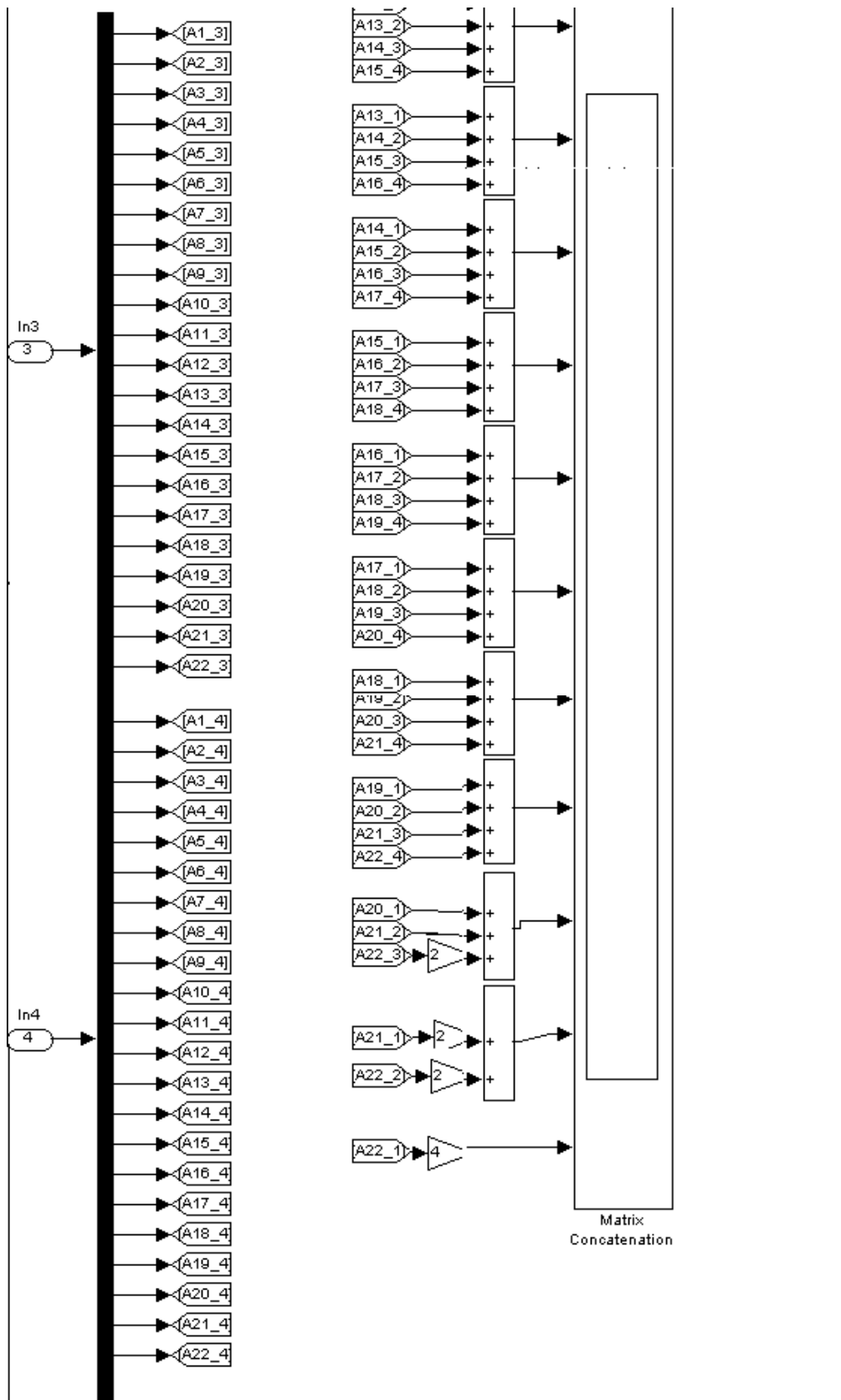


Figura 33(b) Correlación 2 Mbps

3.1.2.3 Correlación 5.5 y 11Mbps

El subsistema Correlación a 5.5 y 11Mbps opera de manera similar al de 1 Mbps especificado en 3.1.2.1, con diferencia en las dimensiones de la señal de entrada, que a su vez obliga la adición de algunos bloques sumadores y a cambiar algunos parámetros de los bloques de demultiplexación y concatenación, como se muestra en las figuras 34(a) y 34(b) (debido al tamaño la figura esta dividida en dos partes).

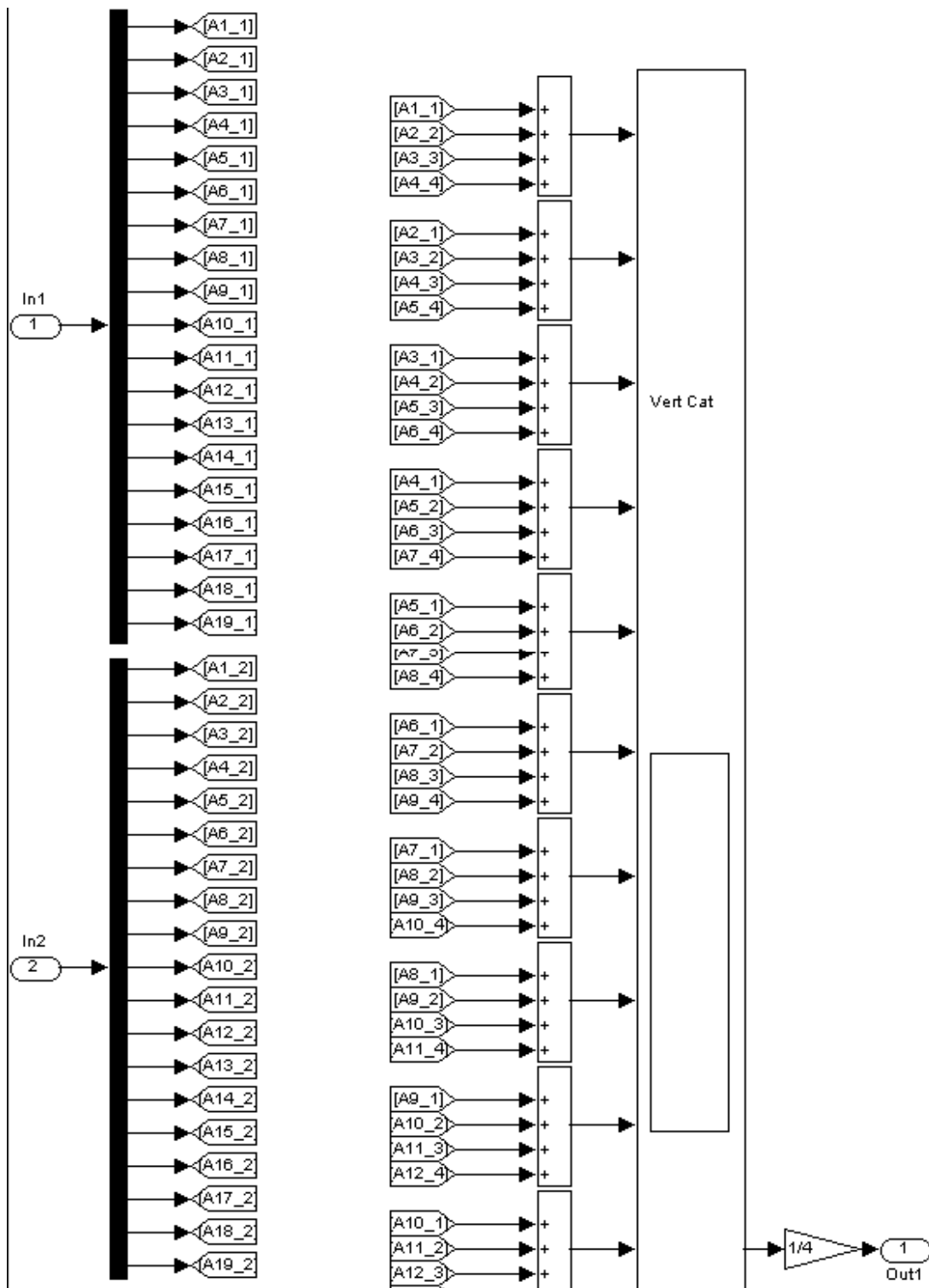


Figura 34(a) Correlación 5.5 y 11 Mbps

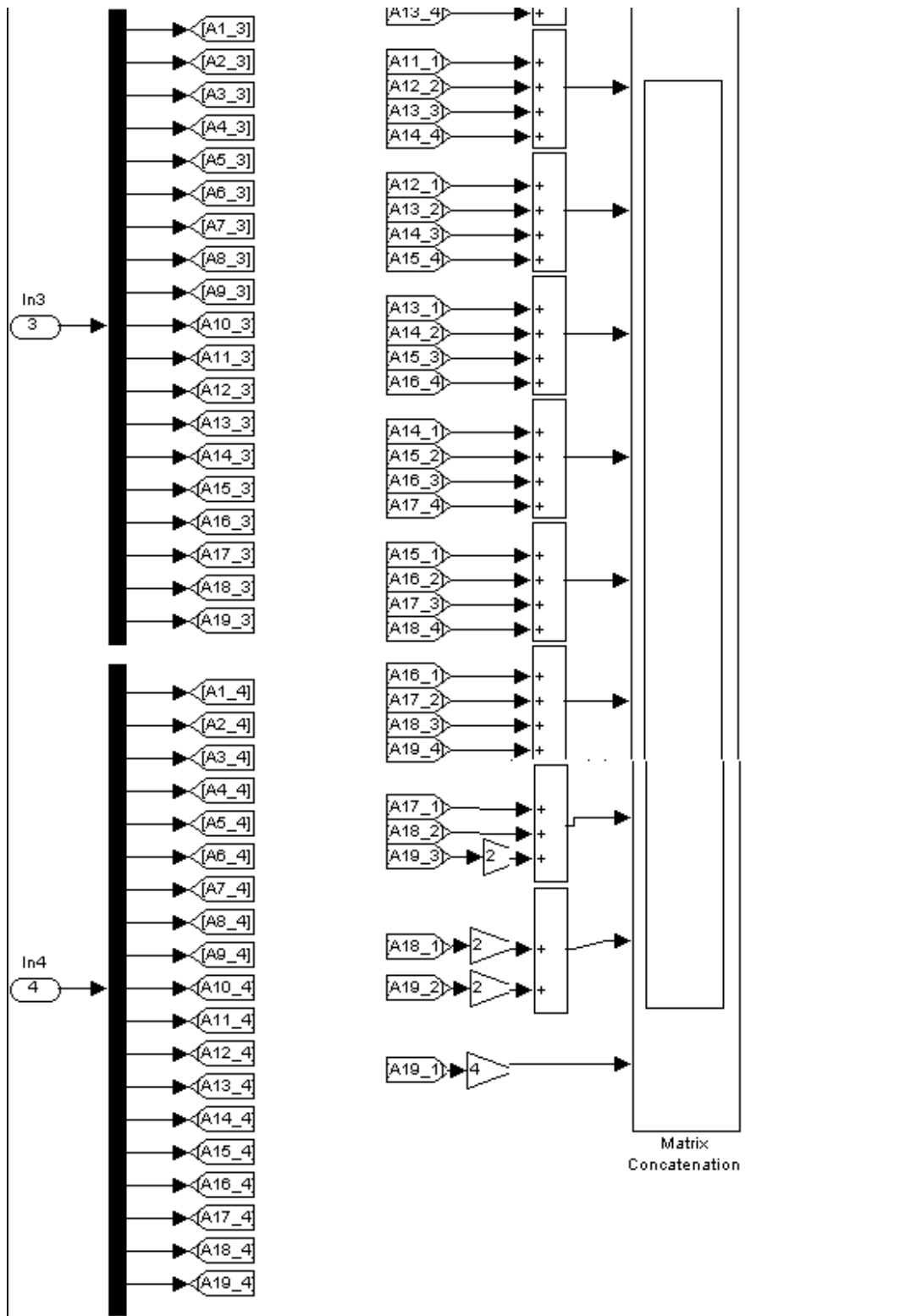


Figura 34(b) Correlación 5.5 y 11 Mbps

3.2 DE_MODULADOR_ENSANCHADOR

Este bloque se simula de forma diferente dependiendo de las 4 velocidades que maneja el estándar.

3.2.1 De_Modulador_Ensanchador 1Mbps

Este Subsistema se compone de cinco bloques como se muestra en la figura 35, y realiza el proceso inverso descrito en el numeral 1.3.1. A continuación se detalla la explicación de cada bloque.

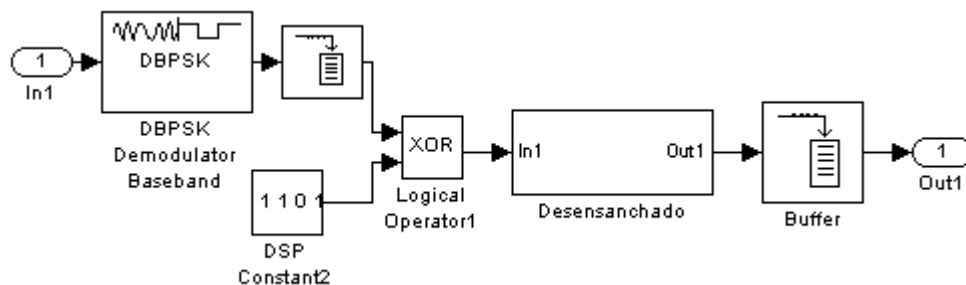


Figura 35 De_Modulador_Ensanchador 1 Mbps

3.2.1.1 Código Barker

Su descripción y parámetros son los mismos de la sección 1.3.1.2.

3.2.1.2 Operador Lógico

Su descripción y parámetros son los mismos de la sección 1.3.1.3.

3.2.1.3 Demodulador DBPSK

Para implementar este proceso se utiliza el bloque “*DBPSK Demodulator BaseBand*”, este aparece en la figura 36.



Figura 36 Demodulador DBPSK

Descripción: El bloque demodulador DBPSK de banda base demodula una señal que fue modulada utilizando el método de desplazamiento de fase binaria diferencial. La salida es una representación de la banda base demodulada.

Parámetros:

- Rotación de fase (Phase rotation): Es la diferencia de fase entre el símbolo anterior y actual cuando la entrada es cero.

3.2.1.4 Desensanchado

Este subsistema permite invertir el proceso de ensanchamiento llevado a cabo en el transmisor. Con el Operador Lógico descrito en 1.3.1.3 se consigue obtener una serie de 11 unos o 11 ceros consecutivos, pero con este bloque se convierten en un solo uno o cero respectivamente. Sus detalles se observan en la figura 37.

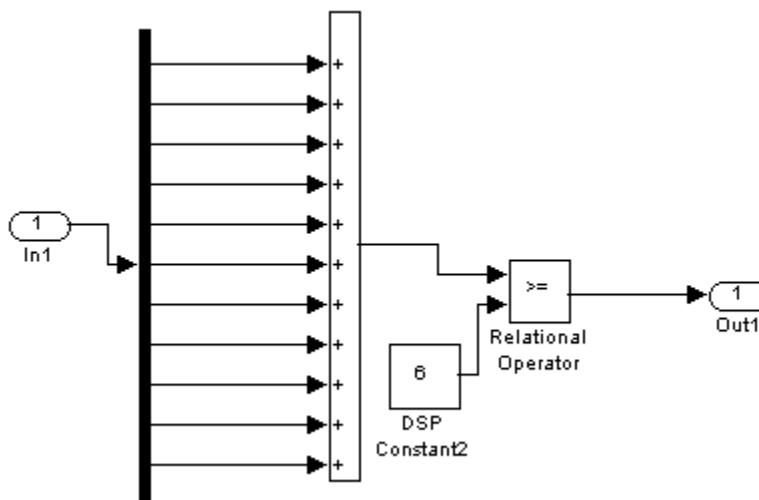


Figura 37 Desensanchado

3.2.1.4.1 Demux

Este bloque entrega cada uno de los 11 chips obtenidos con el desensanchamiento con el código Barker. Su descripción es la misma de la sección 1.3.3.2.

Parámetros:

- Número de salidas (Number of outputs): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a 11.

3.2.1.4.2 Sumador

Este bloque se encarga de contar los bits que hay en uno. Su descripción y parámetros son los mismos de la sección 1.3.3.3.5.

3.2.1.4.3 Constante

Este bloque otorga un punto de comparación para saber si el ensanchamiento realizado en la etapa de transmisión fue de un uno ó de un cero. Su descripción es la misma de la sección 1.2.1.

3.2.1.4.4 Operador Lógico

Este bloque compara el número de unos contados (entregados por el sumador) con el punto de decisión establecido y dependiendo de esto entrega un solo uno o cero a la salida. Su descripción y parámetros son los mismos de la sección 1.3.1.3.

3.2.1.5 Buffer

Su descripción y parámetros son los mismos de la sección 1.3.3.1.

3.2.2 De_Modulador_Ensanchador 2Mbps

Este subsistema realiza el proceso inverso descrito en el numeral 1.3.2. Este bloque se divide en dos partes, como se muestra en la figura 3.38. La parte amarilla es la misma que la explicada en la sección 3.2.1. La parte azul de la figura 38 se implementa a través de los procesos que se describen a continuación.

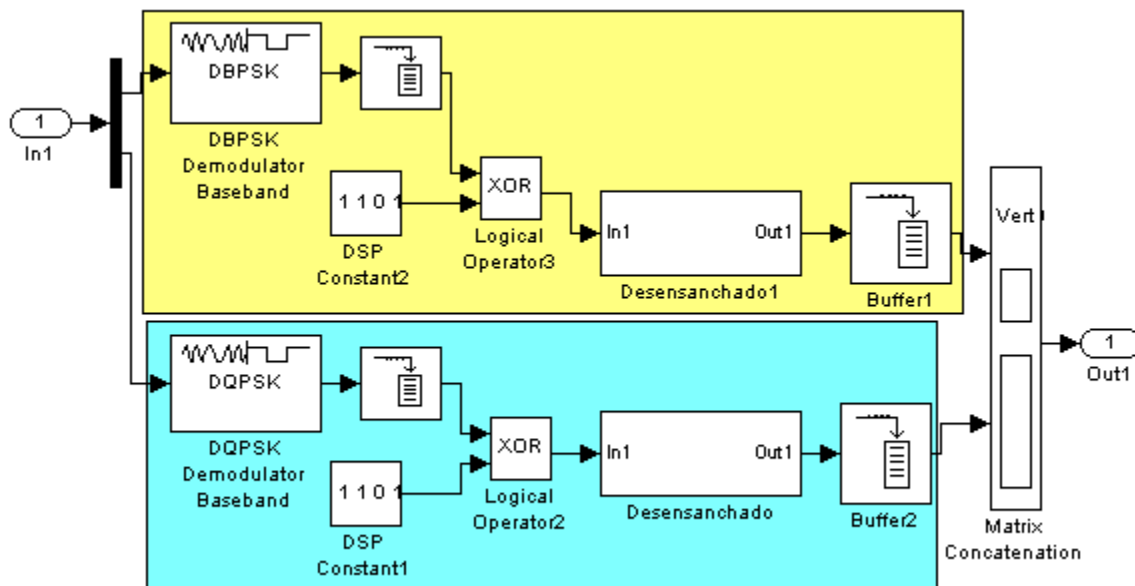


Figura 38 De_Modulador_Ensanchador 2 Mbps

3.2.2.1 Código Barker

Su descripción y parámetros son los mismos de la sección 1.3.1.2.

3.2.2.2 Operador Lógico

Su descripción y parámetros son los mismos de la sección 1.3.1.3.

3.2.2.3 Demodulador DQPSK

Para implementar este proceso se utiliza el bloque “DQPSK Demodulator BaseBand”, este aparece en la figura 39.



Figura 39 Demodulador DQPSK

Descripción: El bloque demodulador DQPSK de banda base demodula una señal que fue modulada utilizando el método de desplazamiento de fase cuaternaria diferencial. La salida es una representación de la banda base demodulada.

Parámetros:

- Tipo de salida (Output type): Determina si la salida consiste de enteros o de pares de bits
- Ordenamiento de la constelación (Constellation ordering): Determina como el bloque mapea cada entero a un par de bits de salida. En la simulación, el ordenamiento de la constelación deberá ser presentada de acuerdo al Código Gray, como se definió en el Anexo A.
- Rotación de fase (Phase rotation): Esta diferencia de fase entre el actual y previo símbolo modulado resulta en una salida de cero.

3.2.2.4 Des-ensanchado

Su descripción y parámetros son los mismos de la sección 3.2.1.4.

3.2.2.5 Buffer

Su descripción y parámetros son los mismos de la sección 1.3.3.1.

3.2.3 De_Modulador_Ensanchador 5.5Mbps

Este subsistema realiza el proceso inverso descrito en el numeral 1.3.3. Este bloque se divide en dos partes, como se muestra en la figura 39. La parte amarilla es la misma que la explicada en la sección 3.2.1. La parte azul de la figura 39 se implementa a través de los procesos que se describen a continuación.

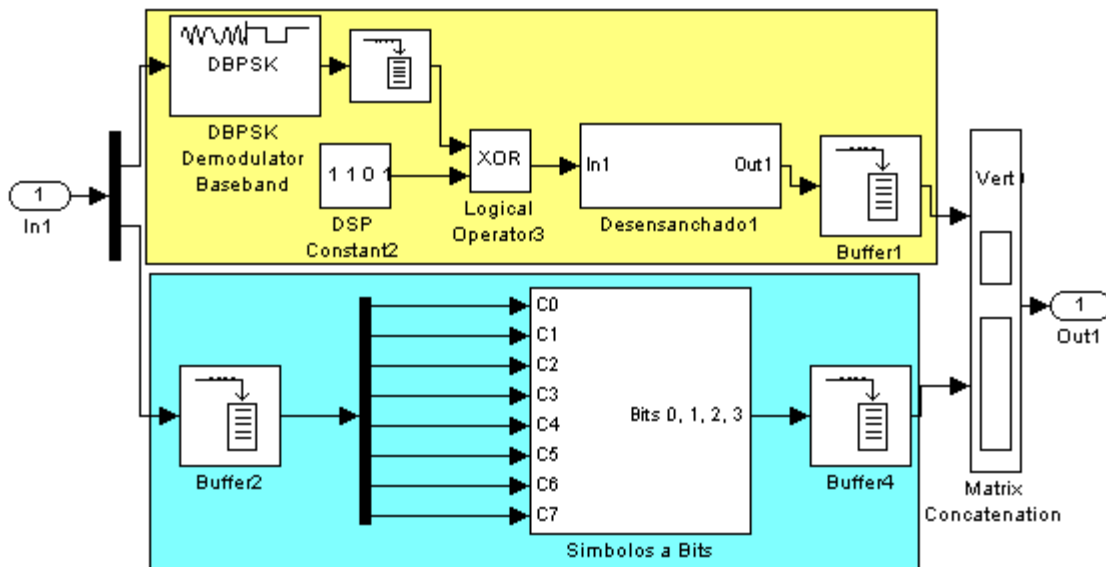


Figura 39. De_Modulador_Ensanchador 5.5 Mbps

3.2.3.1 Demux

Descripción: Este bloque separa cada uno de los símbolos transmitidos por el canal. Su descripción es la misma de la sección 1.3.3.2.

Parámetros:

- Número de salidas (Number of outputs): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a 8.

3.2.3.2 Conversión de Símbolos a Bits

Este bloque se encarga de recibir los 8 símbolos generados en el transmisor y entrega los 4 bits generados de los mismos. La descripción de las partes que componen este subsistema se detalla a continuación y su estructura se puede observar en la figura 40.

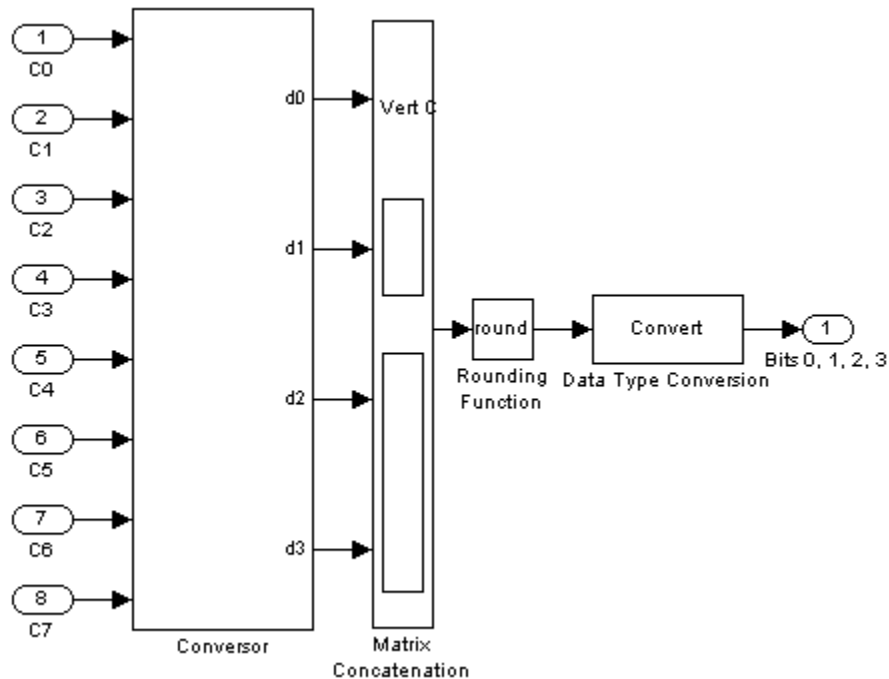


Figura 40. Símbolos a Bits

3.2.3.2.1 Conversor

Este bloque entrega cada uno de los cuatro bits utilizados en 5.5 Mbps a una matriz de concatenación para que vuelvan a ser puestos en una misma trama. Realiza el proceso inverso al descrito en el numeral 1.3.3.3. La descripción de sus partes se detalla a continuación (Ver figura 41).

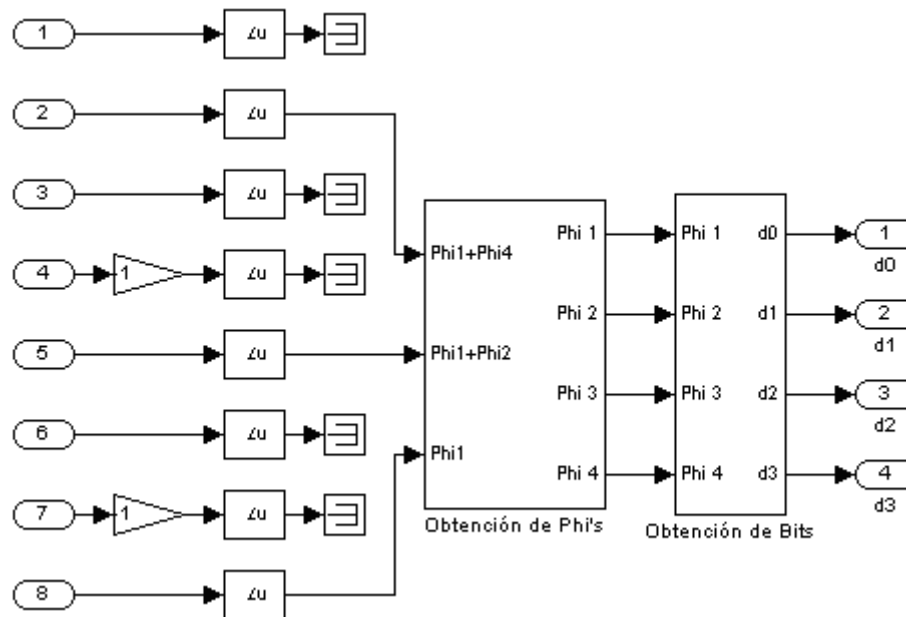


Figura 41. Conversor

3.2.3.2.1.1 Complejo a Magnitud-Angulo

Su descripción y parámetros son los mismos de la sección 1.3.3.3.2.

3.2.3.2.1.2 Obtención de Phi's

Este bloque obtiene los 4 valores de los Phi's ($\varphi_1, \varphi_2, \varphi_3$ y φ_4). Toma sólo 3 entradas, pues no se necesitan ocho ecuaciones para resolver un sistema de 3 incógnitas (recordando que Phi3 siempre es cero) (Ver figura 42).

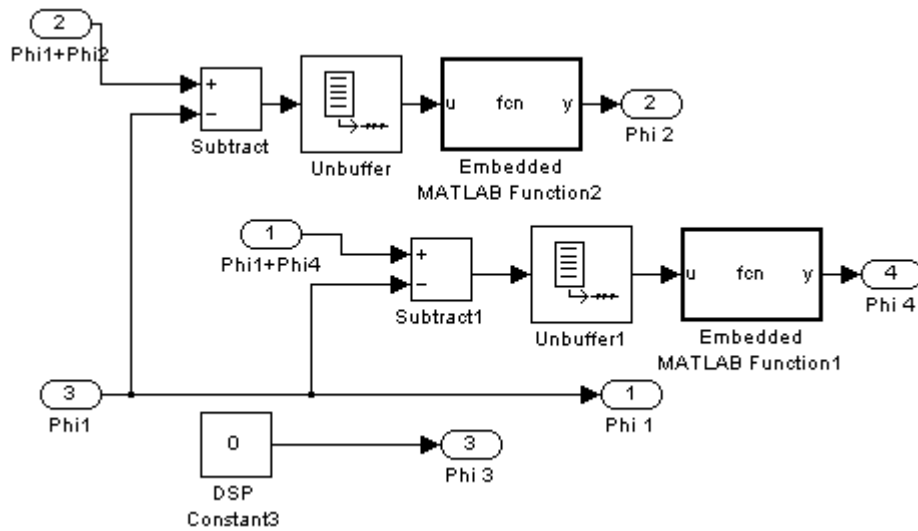


Figura 42. Obtención de Phi's

3.2.3.2.1.2.1 Sustracción

Su descripción es la misma de la sección 1.3.3.3.5.

3.2.3.2.1.2.2 Constante

Su descripción es la misma de la sección 1.2.1.

3.2.3.2.1.2.3 Unbuffer

Descripción: Este bloque convierte tramas a escalares con una tasa de muestreo mayor.

Parámetros:

- Condición inicial (Initial conditions): El valor de salida inicial del bloque para casos de latencia diferente de cero; un escalar, vector o matriz.

3.2.3.2.1.2.4 Función embebida de Matlab

Este bloque permite componer una función en lenguaje Matlab dentro de un modelo de Simulink que genera código embebido. Se utiliza para impedir que a los demoduladores QPSK lleguen valores que estos no puedan convertir o procesen mal.



Figura 43 Función de Matlab Embebida

El código es simple y se muestra a continuación:

```
function y = fcn(u)

aux=u;

if (u<(-3.9) && u>(-5.2))
    aux=pi/2;
end

if (u<(-5.5) && u>(-7))
    aux=0;
end

if (u<(7) && u>(5.5))
    aux=0;
end

y = aux;
```

3.2.3.2.1.3 Obtención de Bits

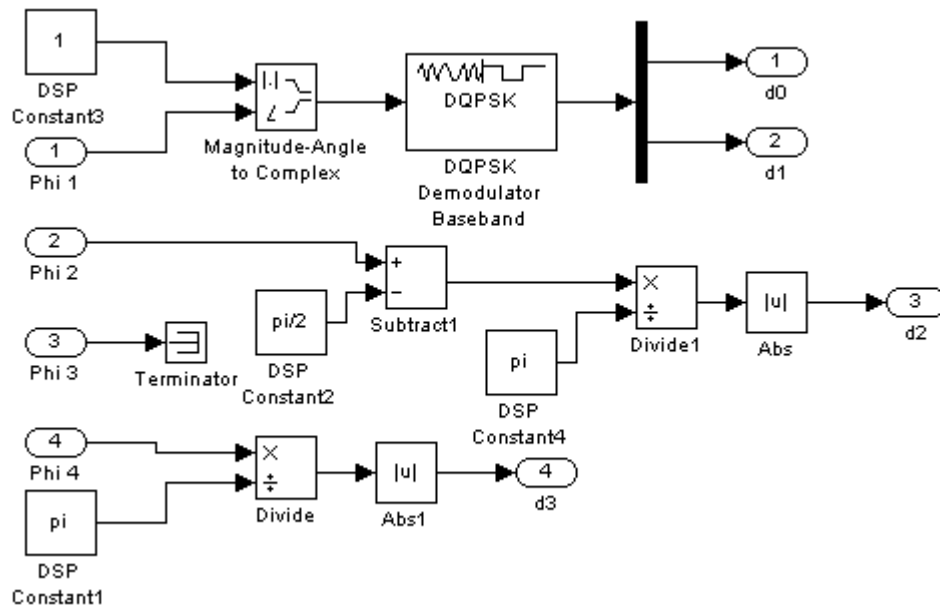


Figura 44 Obtención de Bits

3.2.3.2.1.3.1 Constante

Su descripción es la misma de la sección 1.2.1.

3.2.3.2.1.3.2 Divisor

Descripción: Este bloque permite la multiplicación o división de sus entradas; la multiplicación puede ser de matrices o de elementos.

Parámetros:

- Condición inicial (Initial conditions): El valor de salida inicial del bloque para casos de latencia diferente de cero; un escalar, vector o matriz.
- Condición inicial (Initial conditions): El valor de salida inicial del bloque para casos de latencia diferente de cero; un escalar, vector o matriz.

3.2.3.2.1.3.3 Magnitud-Angulo a Complejo

Descripción: Este bloque convierte una señal en magnitud y/o ángulo a una señal compleja.



Figura 45. Magnitud-Angulo a Complejo

Parámetros:

- Entrada (Input): Este parámetro determina el tipo de entrada del bloque, puede ser una magnitud, un ángulo, o ambos.
- Ángulo (Magnitud) (Angle (Magnitude)): Si la entrada es un ángulo, se especifica la magnitud constante de la señal de salida. Si la entrada es una magnitud, se especifica el ángulo constante en radianes de la señal de salida.

3.2.3.2.1.3.4 Demodulador DQPSK

Su descripción y parámetros son los mismos de la sección 3.2.2.3.

3.2.3.2.1.3.5 Demux

Su descripción es la misma de la sección 1.3.3.2.

3.2.3.2.2 Matriz de Concatenación

Su descripción y parámetros son los mismos de la sección 1.2.2.

3.2.4 De_Modulador_Ensanchador 11Mbps

Este subsistema realiza el proceso inverso descrito en el numeral 1.3.4. Este bloque se divide en dos partes, como se muestra en la figura 46. La parte amarilla

es la misma que la explicada en la sección 3.2.1. La parte azul de la figura 46 se implementa a través de los procesos que se describen a continuación.

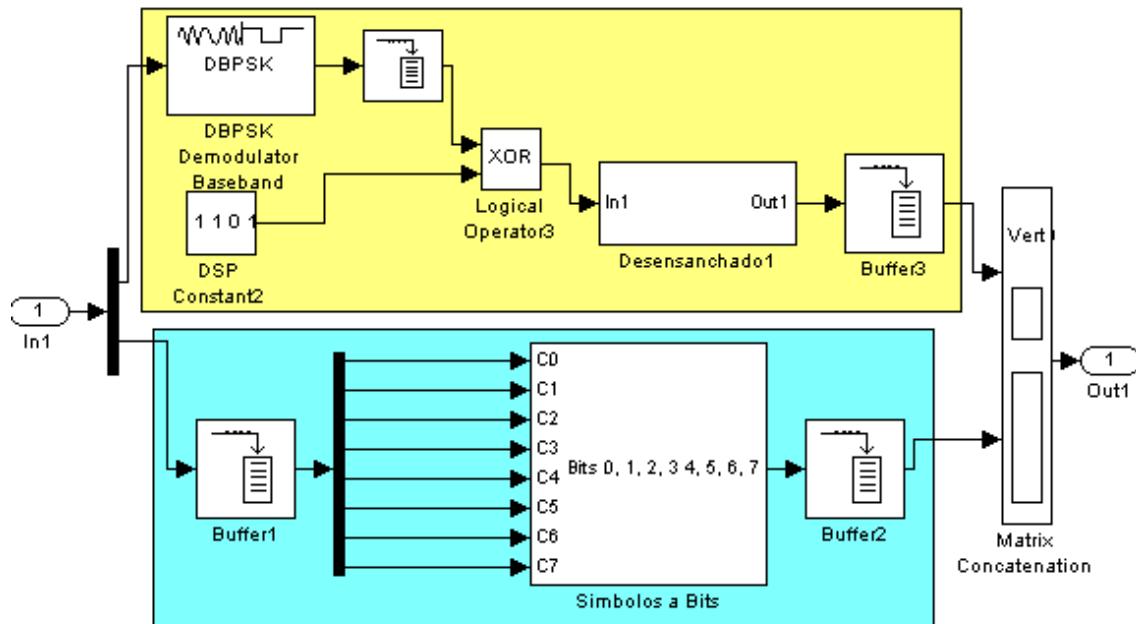


Figura 46 De_Modulador_Ensanchador 11 Mbps

3.2.4.1 Demux

Descripción: Este bloque separa cada uno de los símbolos transmitidos por el canal. Su descripción es la misma de la sección 1.3.3.2.

Parámetros:

- Número de salidas (Number of outputs): Este parámetro permite que se especifique el número, y opcionalmente la dimensión de cada puerto de salida. En la simulación este valor corresponde a 8.

3.2.4.2 Conversión de Símbolos a Bits

Este bloque se encarga de recibir los 8 símbolos generados en el transmisor y entrega los 8 bits generadores de los mismos. La descripción de las partes que

componen este subsistema se detalla a continuación y su estructura se puede observar en la figura 47.

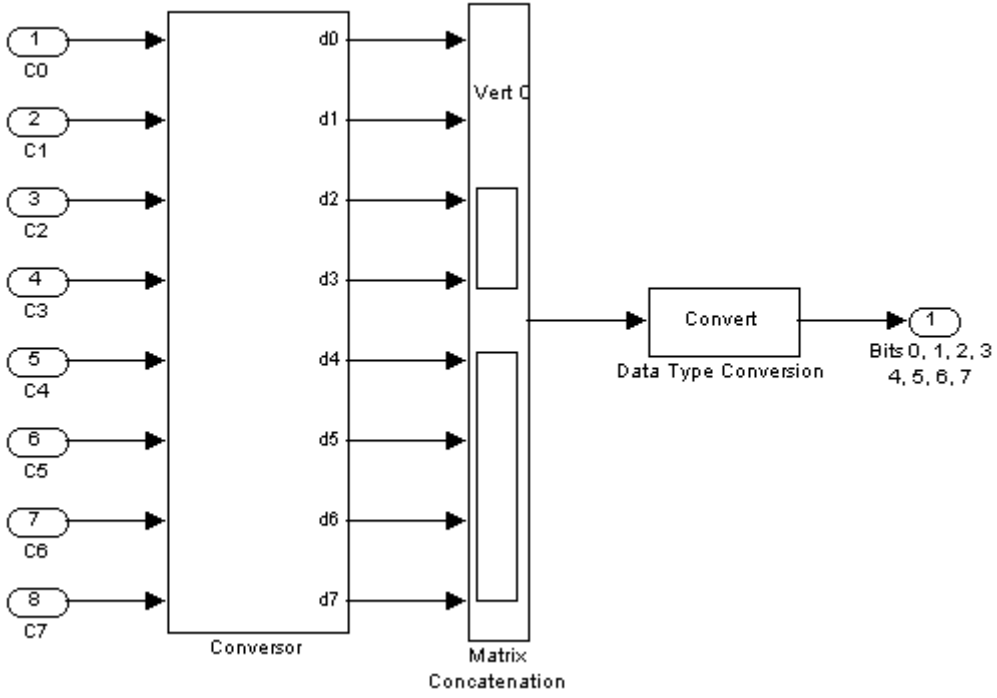


Figura 47 Símbolos a Bits

3.2.4.2.1 Conversor

Este bloque entrega cada uno de los ocho bits utilizados en 11 Mbps a una matriz de concatenación para que vuelvan a ser puestos en una misma trama. Realiza el proceso inverso al descrito en el numeral 1.3.4.3. La descripción de sus partes se detalla a continuación (Ver figura 48).

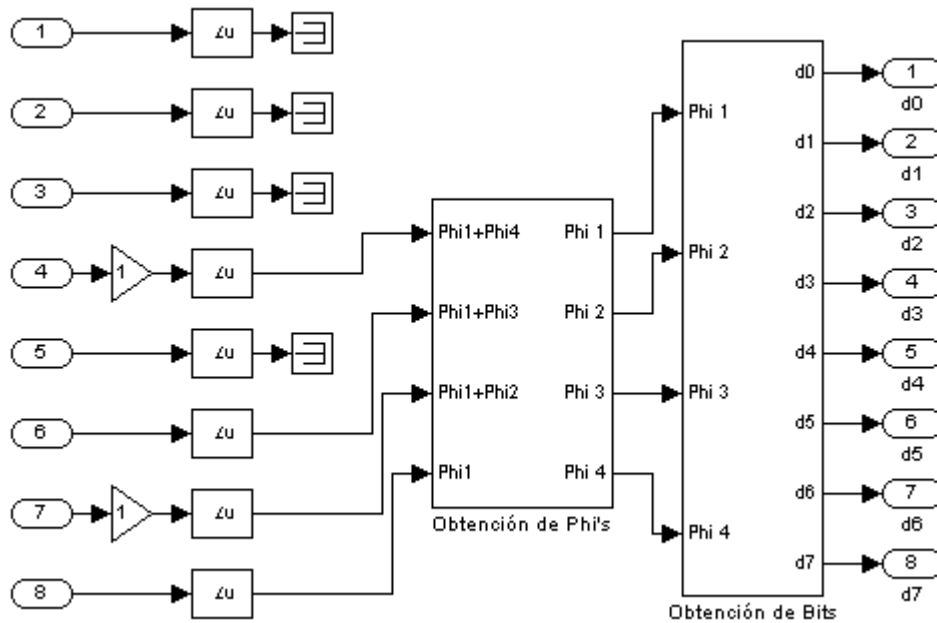


Figura 48 Conversor

3.2.4.2.1.1 Complejo a Magnitud-Angulo

Su descripción y parámetros son los mismos de la sección 1.3.3.3.2.

3.2.4.2.1.2 Obtención de Phi's

Su descripción y parámetros son los mismos de la sección 3.2.3.2.1.2.

3.2.4.2.1.3 Obtención de Bits

Este bloque recibe los 4 ángulos (Phi's), y entrega los 8 bits en que fue agrupada la información del usuario (Ver figura 49).

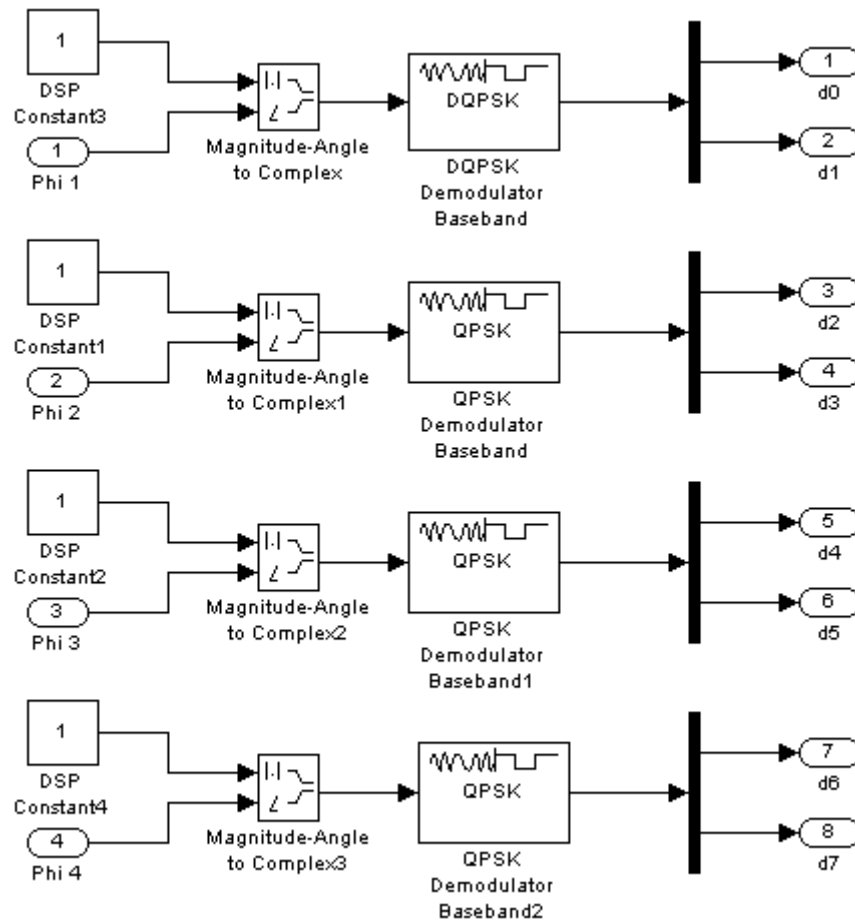


Figura 49 Obtención de Bits

3.2.4.2.1.3.1 Constante

Su descripción es la misma de la sección 1.2.1.

3.2.4.2.1.3.2 Magnitud-Angulo a Complejo

Su descripción y parámetros son los mismos de la sección 3.2.3.2.1.3.3.

3.2.4.2.1.3.3 Demodulador DQPSK

Su descripción y parámetros son los mismos de la sección 3.2.2.3.

3.2.4.2.1.3.4 Demux

Su descripción es la misma de la sección 1.3.3.2.

3.2.4.2.2 Matriz de Concatenación

Su descripción y parámetros son los mismos de la sección 1.2.2.