



« PROYECTO FINAL DE CARRERA »

presentado por

" Carlos Calero Moreno "

Universidad Telecom SudParis (Francia)

Departamento EPH

Ingeniería de Telecomunicación

"APLICACIÓN DE UN SISTEMA DE COMUNICACIONES UWB EN TIEMPO REAL"

Director : MULLER, Muriel - Télécom SudParis

Ponente : HERNÁNDEZ SOLANA, Ángela - CPS Zaragoza

Fecha : Marzo de 2011

RESUMEN

El objetivo de este trabajo es desarrollar una plataforma para transmitir datos en tiempo real en sistemas basados en redes de sensores.

Para validar el nuevo concepto de detección de pulsos propuesto por el Laboratorio de Electrónica y Física (EPH) de la universidad Télécom SudParis (Evry, Francia), se diseñó y se realizó un circuito receptor VLSI (Very Large Scale Integration) en el laboratorio empleando la tecnología CMOS 0.35 μm . Este nuevo método de detección es llamado TDSC (Time Delayed Sampling & Correlation). Las prestaciones del chip TDSC empleado ya fueron comprobadas en proyectos anteriores mediante un sencillo emisor y un receptor realizados en tarjetas impresas de test.

La finalidad de este proyecto final de carrera es continuar con la validación de este concepto mediante la transmisión de datos en un formato predefinido y, sobre todo, incorporar las operaciones de correlación y sincronización de datos entre el emisor y el receptor. Este es un punto importante para todo sistema de comunicaciones, pero especialmente en este caso porque tenemos que sincronizar la señal en una anchura de unos pocos nanosegundos.

El trabajo realizado ha sido: proposición y diseño de nuevas tarjetas de test emisoras y receptoras para hacer funcionar el sistema compuesto por un emisor y el receptor TDSC, implementación en un microcontrolador del proceso de transmisión de pulsos UWB (Ultra Wide Band), estudio del estado del arte de las comunicaciones LDR-UWB (Low Data Rate-Ultra Wide Band), especialmente de los diferentes tipos de receptores y las ventajas de cada uno y, finalmente, un estudio de la compatibilidad con la norma IEEE 802.15.4a de los protocolos utilizados en el emisor y en el receptor, sobre todo en las capas física (PHY) y de control de acceso al medio (MAC).

Índice

Introducción.....	5
--------------------------	----------

Capítulo 1

1. Tecnología UWB.....	8
1.1. Diferentes técnicas de recepción UWB-LDR.....	8
1.2. El método TDSC.....	12

Capítulo 2

2. Concepción y pruebas de los circuitos receptor y emisor	14
2.1. Circuito receptor.....	15
2.2. Test y diseños sobre el circuito receptor.....	16
2.3. Circuito emisor.....	25
2.4. Resultados obtenidos para validar el funcionamiento de las dos tarjetas elaboradas	26

Capítulo 3

3. Estudio de la compatibilidad del estándar IEEE 802.15.4a con la técnica TDSC	30
3.1. Descripción del estándar IEEE 802.15.4a	30
3.2. Compatibilidad con el sistema TDSC	31
3.3. Simulación en Matlab	34

Conclusión general.....	42
--------------------------------	-----------

Referencias bibliográficas.....	44
--	-----------

Anexos

A.1. Trama en la capa MAC.....	45
A.1.1. Esquema general de la trama	45
A.1.2. Número de octetos de una trama “beacon” en la capa MAC.....	47
A.2. Trama de la capa PHY IEEE 802.15.4a.....	48
A.2.1. Especificación PHY UWB.....	48
A.2.2. Formato de la trama PHY UWB.....	48
A.2.3. Estructura del símbolo PHY UWB.....	50
A.2.4. Parámetros del PSDU.....	51
A.2.5. Parámetros del preámbulo.....	52
A.2.6. Preámbulo SHR.....	53
A.2.6.1. Campo SYNC del SHR.....	54
A.2.6.2. Campo SFD del SHR.....	55
A.2.7. Cabecera PHY (PHR).....	55
A.3. Trama de la capa PHY IEEE 802.15.4 (ZigBee).....	57
A.3.1. Especificaciones 868/915 MHz BPSK PHY.....	57
A.3.1.1. Tasa de datos	57
A.3.1.2. La modulación y el ensanchamiento.....	57
A.3.2. Formato de trama.....	58
A.3.2.1. Campo Preámbulo del SHR.....	59
A.3.2.2. Campo SFD del SHR.....	59
A.3.2.3. Campo de longitud de trama.....	59
A.3.2.4. Campo PSDU.....	60
A.4. Gama de frecuencias.....	60
A.5. Código de simulación Matlab	61

Introducción

El acceso permanente a los servicios de información es una necesidad en la sociedad moderna. Las infraestructuras para las comunicaciones exteriores están bien desarrolladas. La necesidad de desarrollar las comunicaciones en el interior se hace necesaria en la actualidad para diferentes tipos de aplicaciones.

Estas aplicaciones no necesitan generalmente un gran volumen de datos a transmitir (algunos bits por segundo), y no es necesario transmitir estos datos de manera continua.

Esto demuestra una constante necesidad de sistemas inalámbricos para las comunicaciones de proximidad con el objetivo de transmitir informaciones a una tasa de datos baja o media, y que presenten una autonomía bastante extendida (bajo consumo). Estos sistemas deben ser de bajo coste y presentar una baja complejidad de concepción, por lo que hay que centrarse en realizaciones sobre silicio, junto a la tecnología CMOS, que en la actualidad está bien desarrollada y es poco costosa ya que está envuelta en un proceso de fabricación en masa.

Desde hace algunos años, la tecnología de banda ultra ancha (Ultra-Wide Band, UWB en inglés) ha puesto en vilo a todos los grupos de investigación, tanto industrial como académica. Ésta se basa en conceptos de transmisión de impulsos de muy baja duración que se asemejan al principio del radar.

Por definición, presenta propiedades que parecen responder a las necesidades citadas anteriormente. El consumo del sistema puede ser fijado por la cantidad de datos a enviar y, por lo tanto, será bajo para este tipo de aplicaciones.

Existen dos métodos principales para la tecnología UWB, llamados UWB con múltiples portadoras (Multi-Carrier UWB, MC-UWB) y radio impulsional (Impulse Radio UWB, IR-UWB) [1]. Nosotros nos interesamos solo en la forma impulsional, adaptada a las transmisiones de con una tasa de datos baja o media.

Existen varias arquitecturas de recepción de señales UWB: tipo “Rake”, detección coherente con estimación de canal, detección de energía y referencia transmitida (Transmitted Reference, TR).

Esta última, conocida bajo el nombre de Referencia Transmitida en Banda Ultra Ancha (TR-UWB), ha ganado en popularidad para las aplicaciones con una tasa de datos baja o media [2], y nosotros la hemos retenido para nuestro proyecto. Esta técnica utiliza dos impulsos fuertemente correlados y separados por una duración temporal T_D , muy precisa y bien conocida, para codificar un bit de datos. El sistema de recepción está compuesto por una línea de retardo analógica de banda ancha y por un correlador. La línea de retardo permite hacer una copia de la señal recibida con un retardo que corresponde exactamente a T_D . La correlación [3] es efectuada a continuación sobre estas dos señales. A pesar de la aparente

facilidad de este tipo de receptor, su implementación en una tecnología integrada permanece como una tarea difícil de realizar, en particular aquella de la línea de retardo, que debe tener propiedades de banda ancha para las aplicaciones UWB.

Para paliar esta dificultad, se ha propuesto [4] un nuevo método de detección bautizado como TDSC (Time Delayed Sampling & Correlation). Éste utiliza dos muestreadores analógicos temporalmente retardados, donde el retardo T_D es aplicado sobre la señal de control de los dos muestreadores. El retardo es generado de manera digital, por lo que se hace programable y puede ser utilizado en un contexto multiusuario [5].

Este documento describe la concepción y la realización de un prototipo de tarjeta receptora basada sobre el principio TDSC [6]. Entre las posibles aplicaciones de este sistema, para las comunicaciones con una tasa de datos media o baja en banda ultra ancha (Ultra-Wide Band Low Data Rate, UWB-LDR), se encuentran la televigilancia, la telemedicina, la localización en interiores, las redes de sensores, etc... Partiendo de los resultados obtenidos en los anteriores trabajos desarrollados en Télécom SudParis sobre el sistema TDSC, en los cuales se ha realizado y probado el chip TDSC, hemos concebido una tarjeta receptora que contiene toda la electrónica necesaria para el buen funcionamiento del receptor: microcontrolador, conversor analógico-digital, etc...

Nosotros hemos concebido igualmente un emisor basado sobre un generador de impulsos SRD con el objetivo de realizar pruebas con el sistema TDSC completo. El método TDSC simplifica la operación de sincronización, pero ésta queda sin embargo como un punto crítico a realizar. La idea de este trabajo es efectuar la comunicación entre el emisor y el receptor, realizando las operaciones de correlación y de sincronización del sistema, que no están aún hoy integradas en el chip CMOS diseñado y fabricado. Estas operaciones se harán en un primer momento en el ordenador, más tarde un algoritmo deberá permitir realizarlas sobre la tarjeta. Un punto crucial es la realización del retardo T_D , que debe ser exactamente el mismo a nivel del emisor y del receptor, sin el cual no se producirá la recepción. El primer paso a seguir para la continuación de este proyecto es la realización de este retardo de manera independiente del ordenador o de una acción humana.

Las comunicaciones UWB a velocidad baja o media estudiadas aquí se pueden inscribir en el marco de definición del estándar IEEE 802.15.4, relativo a las comunicaciones a baja velocidad y con bajo consumo, tal como la tecnología ZigBee. La capa física (PHY) de este estándar ha sido modificada para integrar las técnicas de banda ancha impulsionales, y se encuentra descrita en el estándar 802.15.4a [8].

Una parte del trabajo de este proyecto concierne también el estudio de la factibilidad del sistema TDSC para emitir y recibir datos siguiendo las recomendaciones de esta norma (a nivel PHY y MAC).

La finalidad era estudiar bajo qué condiciones podemos esperar beneficiarnos de las recomendaciones de la norma y cuáles eran las modificaciones a aportar a nivel del emisor, incluso en el método de adquisición y sincronización, para realizar una transmisión de datos.

Veremos en el primer capítulo que la trama preconizada por esta norma nos obliga a modificar el emisor actual para adaptarlo a la modulación TR (envío de un « doblete » o pareja de impulsos) que utilizamos. Presentaremos diferentes soluciones estudiadas en simulación. Finalmente daremos una conclusión final a este trabajo.

Capítulo 1

1. Tecnología UWB

La tecnología UWB [9] es una tecnología utilizada para transmitir información en banda ancha (al menos 500 MHz). Está basada en la propagación de una forma de onda compuesta de una secuencia de muy pequeños impulsos con una duración entorno al nanosegundo. Podemos observar en la figura 1 que para impulsos con una anchura α de 2.1 ns, 1.05 ns y 0.525 ns hay una señal transmitida con una ancho de banda (medido a -10 dB) de 562 MHz, 1.1 GHz y 2.2 GHz, respectivamente.

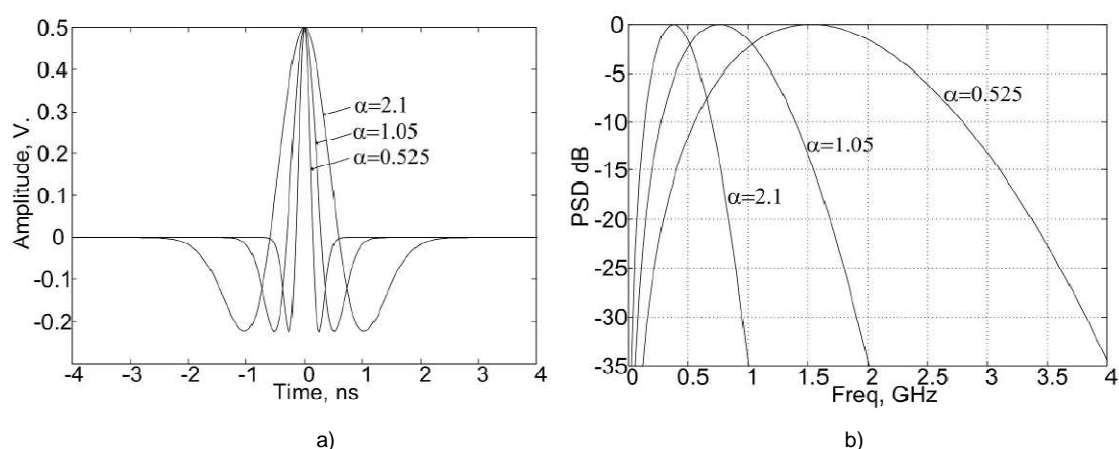


Figura 1. a) Impulsos con diferentes anchuras y b) su espectro

1.1. Diferentes técnicas de recepción UWB-LDR

Existe una gran variedad de arquitecturas de recepción indicadas para trabajar con la tecnología UWB, pero se puede hacer una clasificación en tres grandes familias que suele ser la más utilizada. Por un lado los receptores coherentes y, por otro, los receptores no coherentes y los receptores pseudo coherentes.

De manera general, la recepción y la detección de la señal UWB impulsional son delicadas por dos razones:

- La propagación multirrayecto deforma la forma de onda de la señal emitida inicialmente.
- La detección necesita una sincronización muy precisa entre la señal UWB recibida y la señal de referencia temporal local en la recepción.

Receptor Coherente

Las soluciones coherentes han sido desarrolladas extensamente, pero sobre todo para el UWB impulsional a alta velocidad. Para simplificar, diremos que los receptores coherentes necesitan que el impulso de referencia sea generado en el receptor para realizar la correlación

con la señal recibida. Esto plantea un problema, pues la señal recibida a menudo está muy deformada al atravesar el medio de transmisión, lo que complica las fases de sincronización y de demodulación. Será necesario entonces desarrollar un receptor de tipo Rake (basado en la estimación de canal) del cual presentamos un esquema en la figura 2, que en efecto permitirá explotar de mejor manera la diversidad del canal, pero que va a aumentar la complejidad del receptor y el consumo.

Este tipo de receptor estará más adaptado a la tecnología UWB-HDR (UWB-High Data Rate > 110Mbps), que posee, ya de por sí, la dificultad de conservar una sincronización precisa entre el emisor y el receptor, lo que aumenta la complejidad de los relojes y la necesidad de algoritmos complejos en la detección.

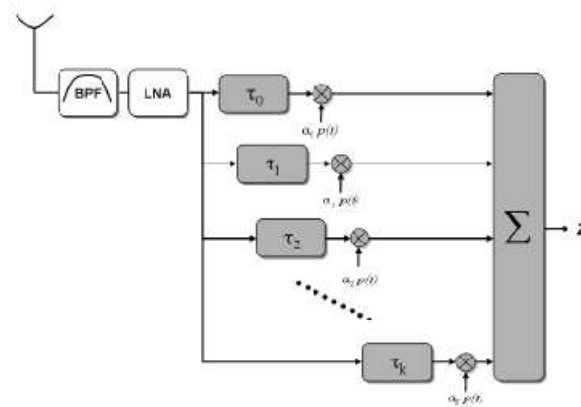


Figura 2. Esquema del principio de un receptor Rake

Receptores no coherente y pseudo coherente

Para las comunicaciones con una tasa de datos baja-media (<2 Mbps), hay una gran necesidad de receptores con una estructura sencilla, capaces de explotar la rica diversidad de canales UWB multirayecto con un coste abordable, un consumo razonable y una baja complejidad. Las estructuras de receptores óptimos, derivadas de las comunicaciones clásicas de banda estrecha, no son realizables para las comunicaciones UWB de baja potencia y, por esto, una parte considerable de la investigación ha sido orientada hacia la concepción de esquemas no coherentes o pseudo coherentes subóptimos, con una estructura basada en los sistemas de detección de energía (Energy Detectors (ED)) y en los sistemas de autocorrelación (Autocorrelation Receivers (AcR)) (figura 3).

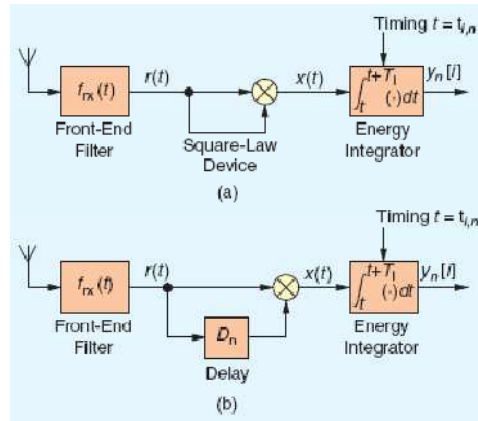


Figura 3. Arquitectura general de los receptores: (a) ED y (b) AcR.

Receptor no coherente

Para liberarse de estas dificultades, los sistemas más simples utilizan métodos de detección basados en la medida de la energía RF recibida (figura 3a). Esta detección de energía puede ser realizada, ya sea con un diodo rápido de tipo Schottky, lo cual es muy frecuente, o bien con un diodo de efecto túnel.

A pesar de la simplicidad, estos métodos no son muy eficaces por las siguientes razones:

- Un detector de energía no puede distinguir la señal útil del ruido, por lo que no podemos suavizarlo cuando hay varios impulsos.
- Un detector de energía tiene tendencia a diluir la energía de la señal en una duración mucho mayor que aquella del impulso inicial. La amplitud máxima de la señal de salida del detector se encuentra atenuada y el resultado es, a menudo, una muy baja sensibilidad.

Hay que indicar que, en el caso de los receptores ED [10], la información sobre la fase de los símbolos se pierde.

Sin embargo para un receptor no coherente óptimo genérico, es necesario un filtro adaptado a la respuesta del canal UWB antes del ED, lo que elimina la ventaja de la simplicidad de la estructura, por lo que se debe encontrar un compromiso entre la complejidad y las prestaciones.

Receptor pseudo coherente

En el caso UWB, se obtiene una arquitectura de recepción simple si se efectúa la comparación de fase antes de la integración de la energía. Típicamente, estos receptores integran una línea de retardo analógica y un mezclador para comparar la fase de la señal en dos instantes de tiempo correspondientes al desfase del retardo (figura 3b). Por lo tanto, la señal de salida es acumulada, recolectando la energía contenida en todas las componentes

multitrayecto. Este tipo de receptor es llamado generalmente AcR [11], porque su tratamiento de la señal es equivalente a la aplicación analógica de un dispositivo de autocorrelación con un retardo preestablecido.

Hay numerosas modulaciones posibles con este tipo de receptor, pero nosotros citaremos principalmente la modulación por Referencia Transmitida (TR) [12].

En la señalización TR (figura 4) cada impulso UWB es reemplazado por dos impulsos fuertemente correlados entre ellos y separados temporalmente por una duración muy precisa T_D . Cuando este doblete de impulsos llega en recepción, podemos suponer razonablemente que cada uno de estos impulsos ha sufrido las mismas deformaciones en su forma de onda y que continúan siempre fuertemente correlados.

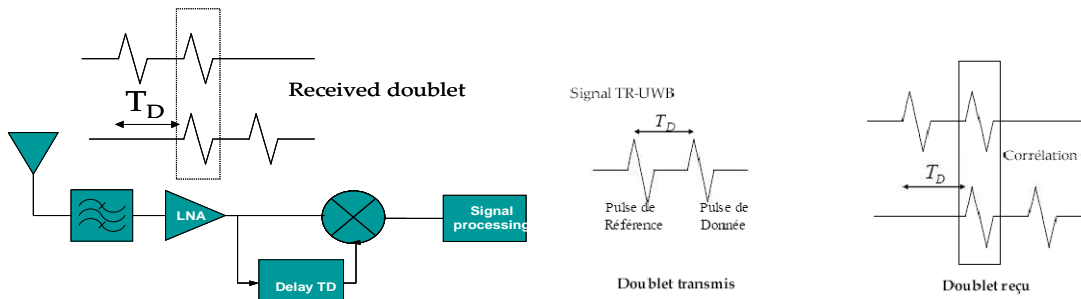


Figura 4. Esquema de Referencia Transmitida (TR)

Una variante muy interesante del sistema de detección TR es el esquema de Referencia Transmitida Diferencial (Differential Transmitted Reference (DTR)). La idea principal del DTR (figura 5) es que utiliza el impulso recibido como modelo para detectar la señal que llega posteriormente. Una ventaja del DTR es que mejora la tasa de error de bit (Bit Error Rate (BER)) en 3 dB, todo esto conservando una estructura de receptor tan simple como en el esquema TR. Esta modulación podrá ser utilizada con nuestro receptor por medio de una adaptación de éste.

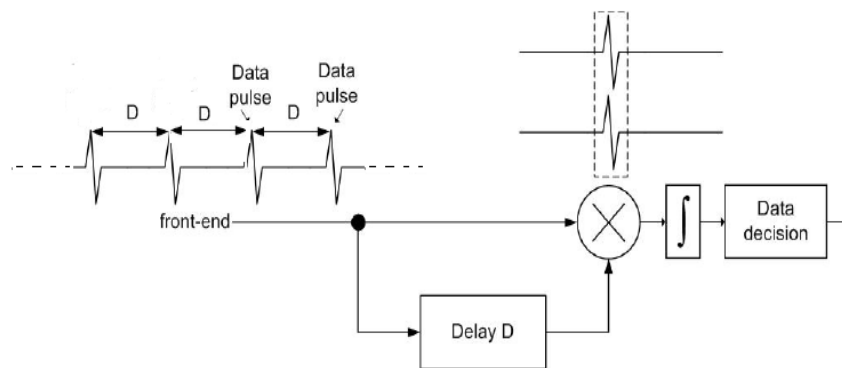


Figura 5. Esquema de Referencia Transmitida Diferencial (DTR)

Una alternativa al esquema TR-UWB es la estructura FSR (Frequency-Shifted Reference) [13], como aparece esquematizado en la figura 6. La idea principal del esquema FSR-UWB es que la referencia es retrasada en frecuencia y no en tiempo como en el esquema de detección TR. El reto ya no está en la implementación del retardo, sino en la implementación del divisor de potencia y de los mezcladores que deberán ser de banda ancha.

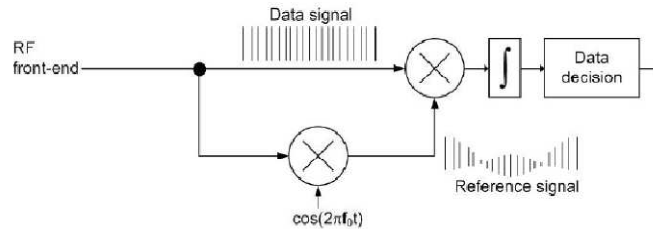


Figura 6. Esquema Frequency-Shifted Reference (FSR)

Actualmente, a nuestro conocimiento, no existe ninguna implementación física de la modulación TR-UWB. Nosotros nos hemos interesado en la implementación de este tipo de receptor buscando cómo realizar una línea de retardo de banda ancha analógica sobre un circuito integrado y con una gran precisión. El método TDSC ha sido propuesto para resolver este problema, y se ha concebido un circuito CMOS para validar este principio de detección. Vamos a describirlo a continuación.

1.2. El método TDSC

Existe un nuevo método de recepción UWB desarrollado por Télécom SudParis y llamado TDSC (Time Delayed Sampling & Correlation). El TDSC es una solución analógica y asíncrona basada en el principio TR-UWB. En la emisión, cada símbolo es codificado por una pareja de impulsos en la cual el retardo T_D entre los dos impulsos es perfectamente conocido. La detección se hace por correlación entre la señal original y su copia retardada.

La originalidad del concepto TDSC reside en la realización de la función de retardo. La línea de retardo es reemplazada por un retardo digital. El circuito contiene dos muestreadores analógicos A y B que memorizan una parte de la señal, y en los cuales los instantes de muestreo, de uno con respecto al otro, están temporalmente retardados del valor T_D . El retardo T_D es aplicado sobre la señal de control de los dos muestreadores (figura 7). El retardo es generado de manera digital, se vuelve programable y puede ser utilizado en un contexto multiusuario. Cuando los muestreadores han capturado las dos formas de onda, retardadas T_D , éstas son volcadas en un correlador que da una muestra de correlación al sistema de detección. El retardo temporal entre los dos muestreadores es perfectamente programable y, con alta precisión, reemplaza ventajosamente a la línea de retardo fijo necesaria en los métodos clásicos.

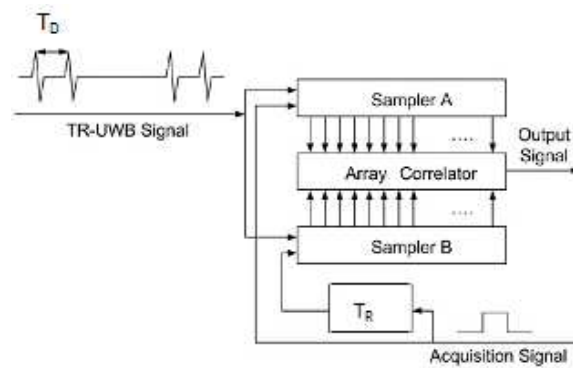


Figura 7. Receptor TDSC

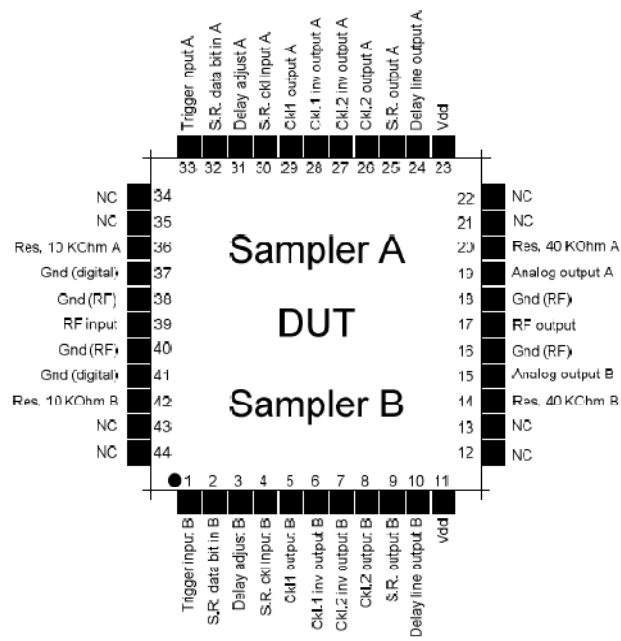
Como podemos observar en la figura 7, el concepto de doblete es empleado en el esquema de transmisión basado en la técnica TDSC, como en los otros esquemas TR-UWB. Los dos impulsos de cada doblete son idénticos para tener una correlación lo más próxima posible a la unidad.

La codificación de un bit '0' o '1' podrá hacerse de diferentes maneras, ya sea utilizando modulaciones de tipo OOK o BPSK por ejemplo, o bien codificando mediante dos valores distintos de retardo T_D . En el caso de una modulación BPSK, por ejemplo, bastará con correlar los dos impulsos recibidos y el resultado será positivo o negativo según si recibimos un '1' o un '0'.

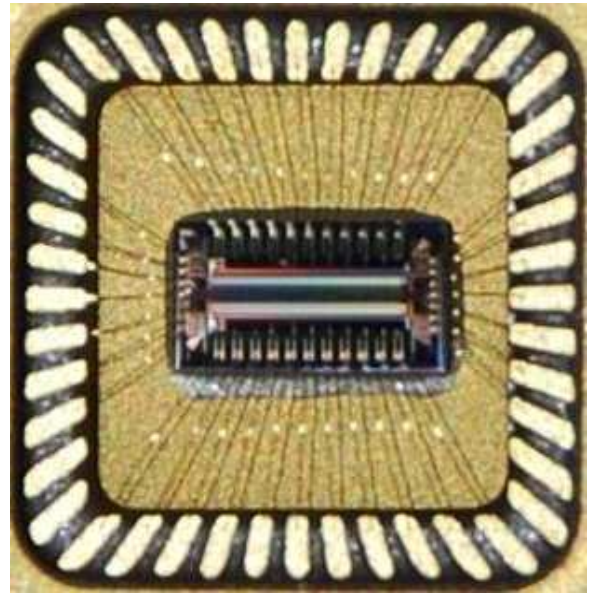
El chip TDSC ha sido concebido en el departamento EPH, realizado con la tecnología CMOS 0.35 μm (en CMP) e integrado en una cápsula CQFP (Ceramic Quad Flat Pack) de 44 pines, como podemos ver en la figura 8. La señal RF (TR-UWB) transmitida es recibida por el pin 39 (RF input), en la línea coplanaria del circuito, que permite transmitir los dos impulsos retardados T_D , en las 128×2 células de muestreo. Estas dos formas de onda analógicas salen del chip TDSC por los pines 19 (Analog output A) y 15 (Analog output B). Únicamente estas operaciones de programación del tiempo de retardo, de escritura de la señal RF (muestreo) y de lectura (salida) de los datos son controladas por un microcontrolador.

Las operaciones de correlación son efectuadas en el ordenador, pero la sincronización no está implementada. Esta última consiste en asegurarse de que los dos impulsos se encuentran presentes en las dos ventanas de detección, con el fin de permitir la detección de un bit de datos. El emisor, basado en un diodo SRD asociado a un generador, permite regular el tiempo T_D « manualmente », reproduciendo el valor autorizado por el microcontrolador utilizado en la recepción.

Para continuar la validación del método de detección debemos desarrollar un sistema de emisión-recepción relativamente autónomo que nos permita al menos grabar un gran número de impulsos reales para validar el método de sincronización propuesto por el laboratorio [3].



a)



b)

Figura 8. Chip TDSC en su cápsula (b) y la disposición de sus pines correspondiente (a)

Capítulo 2

2. Concepción y pruebas de los circuitos receptor y emisor

El trabajo de tesis del Sr. Hirata [7] consistía en mostrar la fiabilidad del método de detección UWB-TDSC, validando las funciones principales como el muestreo rápido y el ancho de banda en recepción y la posibilidad de retardar la activación de los registros de escritura de una manera lo bastante precisa como para permitir la detección del bit, mediante la concepción y la realización de un chip CMOS. Las pruebas realizadas en este contexto tuvieron como finalidad comprobar la funcionalidad de este chip, si bien ninguna función de conversión analógica/digital, correlación o sincronización habían sido implementadas.

El aspecto crucial al cual debemos responder hoy es la implementación de la sincronización. Para ello, se ha concebido un sistema de emisión/recepción, que permite grabar un tren de ondas y realizar la correlación y simular la sincronización en un ordenador con la ayuda del programa Matlab, con el objetivo de validar el método a partir de señales reales. Todo esto solo puede ser realizado si llegamos a implementar el retardo T_D de manera idéntica a nivel del emisor y del receptor.

La señal que debemos generar está representada en la figura 9, en la que se presenta un doblete de impulsos separados T_D y de período T_S . La duración T_D depende de varios parámetros, como el reloj principal del microcontrolador (cuarzo) que se ha escogido para el receptor.

La última etapa sería la elaboración del algoritmo de sincronización y su implementación en la tarjeta de recepción.

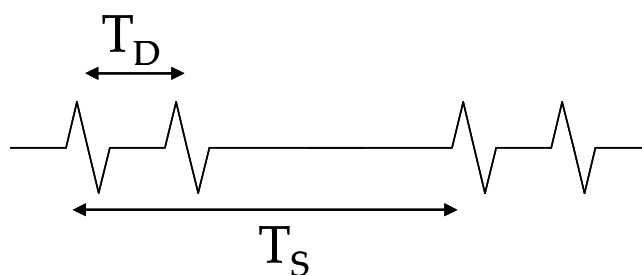


Figura 9. Esquema de dobletes de impulsos a construir para nuestro sistema

Primero vamos a presentar el estudio hecho en la tarjeta de recepción, el circuito emisor será descrito al final del capítulo.

2.1. Circuito receptor

El receptor se compone principalmente por el chip TDSC, que entrega datos analógicos en sus salidas A y B, para una señal UWB presente en su entrada RF_{in} . Así pues, además del chip TDSC, el receptor posee un conversor analógico/digital Texas ADS931 para cada salida analógica A y B. Estas salidas digitales de los conversores se aplican a las entradas de un microcontrolador Maxim DS89C450, que tiene por función generar las señales de control necesarias para el buen funcionamiento del chip TDSC, así como transmitir los datos recibidos a un ordenador mediante el puerto serie para el cálculo de la correlación y de la sincronización. La conexión entre la salida del microcontrolador y el puerto serie RS-232 del PC está asegurada por el circuito Maxim MAX232. Todo esto aparece ilustrado en la figura 10.

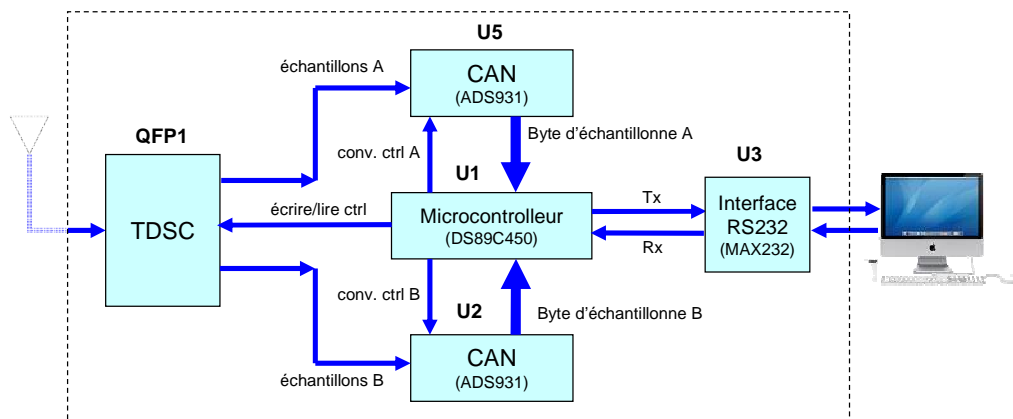


Figura 10. Esquema del principio del receptor TDSC

Es necesario efectuar diversas pruebas para verificar el correcto funcionamiento de los diferentes componentes elegidos a integrar en nuestro receptor definitivo antes de conectarlos al chip TDSC. Es indispensable verificar la forma y la amplitud de las señales de control, generadas por el microcontrolador, y que van a ser aplicadas al chip.

Presentamos primero los resultados de estas pruebas sin el chip y después aquellos del receptor completo.

2.2. Test y diseños sobre el circuito receptor

En un primer momento, el receptor ha sido probado sin el chip TDSC con el fin de comprobar el funcionamiento de los conversores y del microcontrolador por separado.

Daremos primero los resultados de las simulaciones y las medidas de las señales de control resultantes del microcontrolador que nos permiten gestionar los conversores y el chip TDSC. Es evidente que la construcción de la señal TR-UWB, es decir, el retardo T_D y en consecuencia el período T_S , dependen fuertemente de la capacidad del microcontrolador de realizar todas las operaciones en recepción y, por supuesto, de la velocidad del cuarzo. A continuación, verificaremos el buen funcionamiento del chip TDSC caracterizándolo en

términos de frecuencia de muestreo y linealidad. Para concluir, será presentada la tarjeta de recepción final.

El microcontrolador DSC89C450 necesita una tensión de 5V para funcionar y un cuarzo de 30 MHz, por lo que la duración de un ciclo de instrucción es de 33.33 ns.

Señales de control del microcontrolador Maxim DS89C450

Este primer test es realizado con el objetivo de observar las señales de control del chip TDSC y de los conversores analógico/digitales, permitiendo así verificar la compatibilidad de los niveles de salida del microcontrolador con nuestro sistema. Las señales de control para la escritura y la lectura de los dos canales A y B del chip TDSC y las señales de reloj de los conversores analógico/digitales asociados aparecen representadas en la figura 11. El programa del microcontrolador ha sido escrito en lenguaje ensamblador.

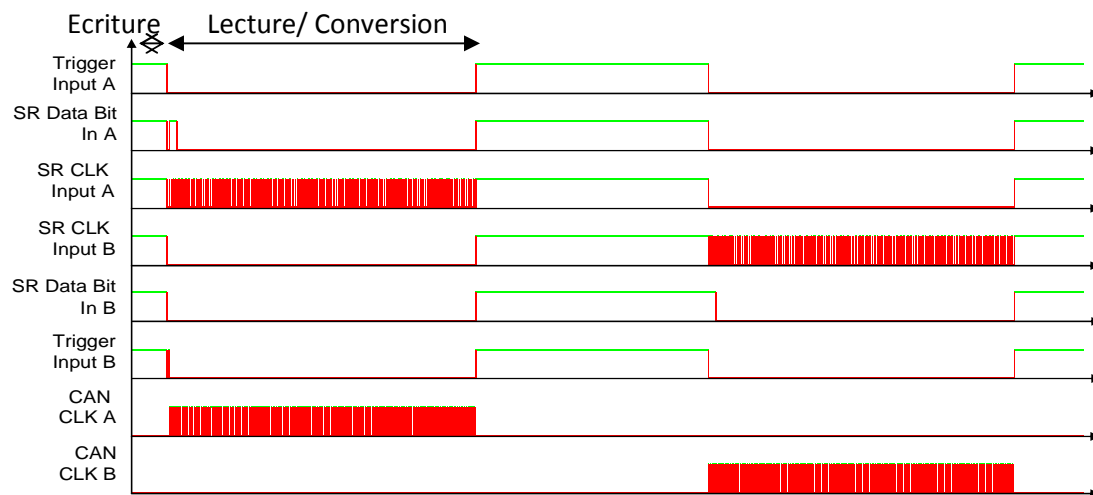


Figura 11. Señales de control del microcontrolador

- Las señales Trigger Input A y Trigger Input B son las señales de escritura de los canales A y B, respectivamente. Hay que señalar que la señal Trigger Input B esta retrasada T_D con respecto a la señal Trigger Input A.
- Las señales SR data Bit A y SR data Bit B son las señales de inicio de la lectura de los canales A y B, respectivamente.
- Las señales SR CLK input A y SR CLK input B son las señales de lectura de las 128 muestras de los muestreadores de los canales A y B, respectivamente.
- Las señales CAN CLK A y CAN CLK B son las señales de reloj de los conversores de los canales A y B, respectivamente.

Las medidas de la figura 12 permiten comprobar las señales generadas por el microcontrolador para la gestión del canal A del receptor (muestreador analógico A del chip y conversión a la salida de este).

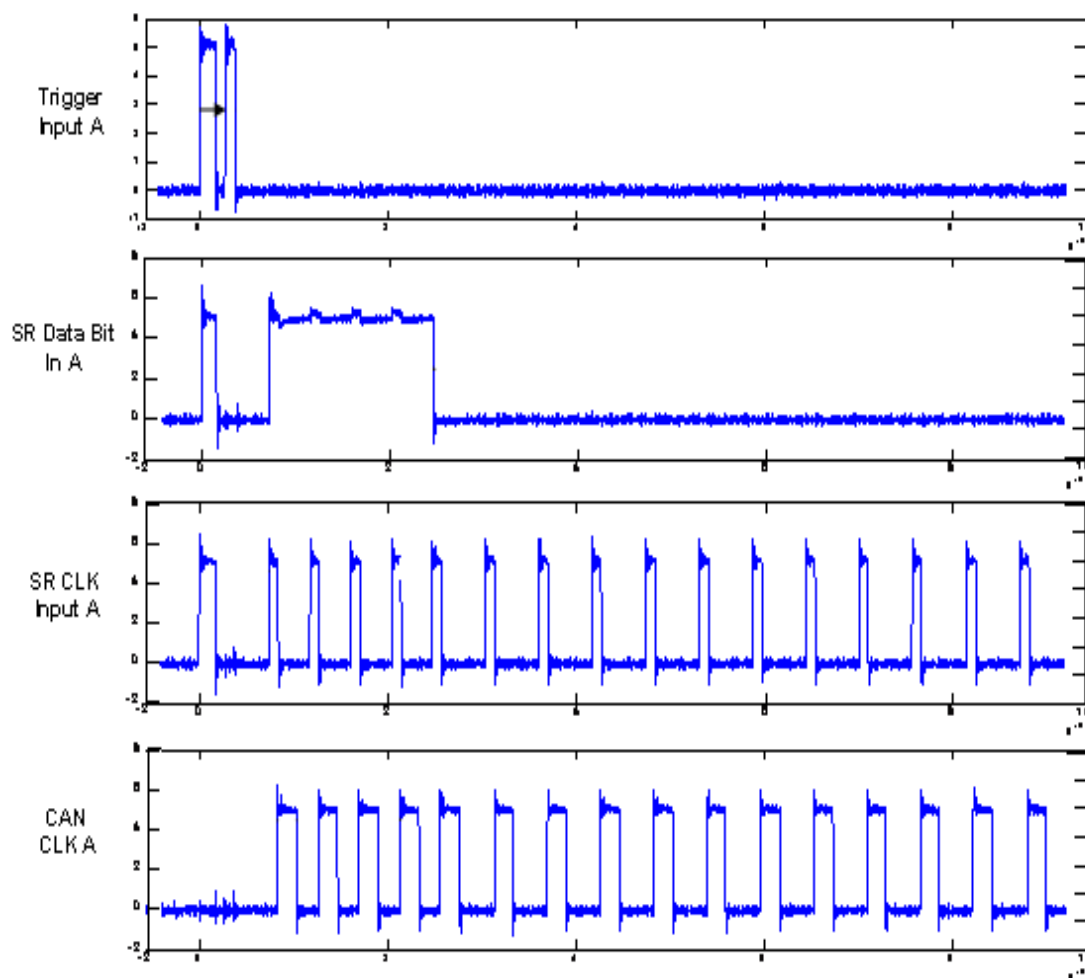


Figura 12. Señales de control a la salida del microcontrolador

El canal B es idéntico y, por tanto, podremos dimensionar la señal TR-UWB, que podrá ser emitida y detectada por nuestro sistema, utilizando los componentes elegidos. Teniendo en cuenta las diferentes anchuras de las señales y el número de impulsos necesarios de la señal de control, elegiremos la duración T_D .

Test de la conexión serie RS-232

Este test tiene por finalidad encontrar la mejor configuración de la conexión serie RS-232 entre el ordenador y las características de transmisión de nuestro sistema. Para ello se han efectuado varias pruebas transmitiendo un conjunto de valores ASCII ya conocido hacia el ordenador con diferentes velocidades de transmisión. En el ordenador, los datos han sido recibidos con la aplicación *Hyper Terminal* de Windows en un primer momento y, después, se ha escrito un programa en Matlab para gestionar la conexión serie. El conjunto de valores ASCII definidos para este test y transmitidos por el microcontrolador han sido recibidos correctamente por el ordenador a una velocidad fija de 38400 baudios. Los parámetros de la transmisión serie utilizada han sido: 38400 baudios, 1 bit de "start", 8 bits de datos, 1 bit de "stop" y sin paridad o control de flujo.

Test de los conversores analógico/digitales Texas ADS931

Este test permite comprobar el funcionamiento de los conversores. La transmisión puede hacerse, bien muestra a muestra, o bien por bloques de 128 muestras. Nosotros hemos testado los dos métodos y hemos decidido retener el segundo, ya que las prestaciones en términos de velocidad de transmisión parecen mejores.

En efecto, registrar las muestras en la memoria interna del microcontrolador permite hacer un muestreo más rápido con la misma velocidad de transmisión de datos.

La señal es muestreada a una velocidad máxima de 2.2 MHz y las muestras son grabadas en la memoria interna del microcontrolador. Después de la obtención de las 128 muestras, todas son transmitidas al ordenador y un nuevo bloque de muestras es grabado y transmitido después del envío del bloque precedente. La velocidad de transmisión al ordenador es de 39602 baudios. La señal cuadrada tiene una frecuencia de 4.54 KHz. La señal cuadrada que ha sido muestreada y la señal resultante de este muestreo aparecen representadas en las figuras 13 y 14 respectivamente. Podemos remarcar en esta última que la señal ha sido muestreada correctamente.

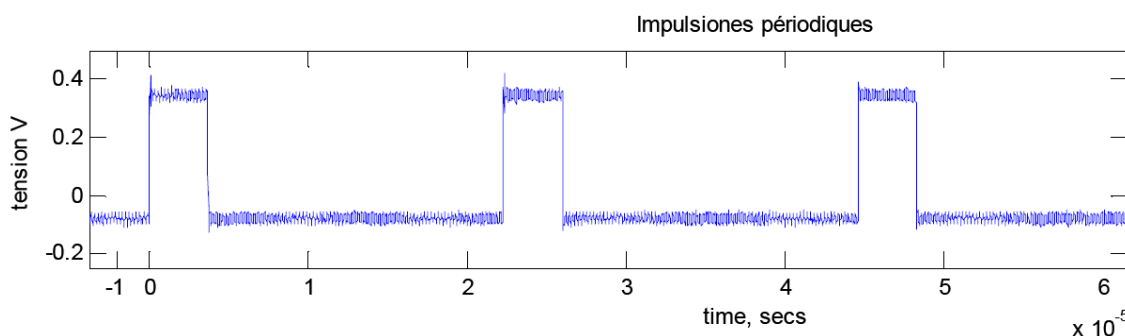


Figura 13. Señal cuadrada de 4.54 KHz

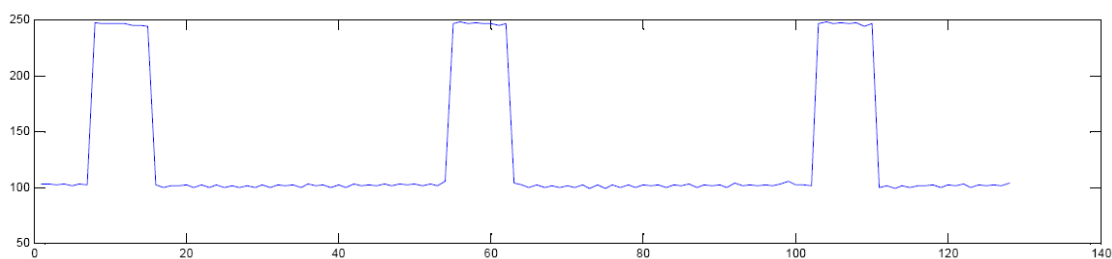


Figura 14. Señal cuadrada de 4.54KHz muestreada

Test para verificar el comportamiento de los relojes que controlan los conversores

El último test concierne la verificación del comportamiento de los relojes que controlan la conversión realizada por los conversores analógico/digitales.

En la figura 15 están presentes las señales correspondientes a los relojes de los conversores. El microcontrolador genera una ráfaga cada 0.064 segundos para controlar el

conversor conectado al canal analógico A del chip. Una segunda ráfaga es generada por el microcontrolador, a la frecuencia de 0.064 segundos pero retrasada 0.032 segundos con respecto a la primera, para controlar el segundo conversor conectado al canal analógico B del chip. Este retardo en tiempo tiene en cuenta T_D y T_S , pero también todos los cálculos efectuados por el microcontrolador.

Un zoom sobre la ráfaga de la señal de control de los conversores es presentado en la figura 16. Se puede observar en ésta que una ráfaga se compone de 128 impulsos que permiten leer las 128 muestras disponibles en los registros A o B del chip.

Debemos señalar que es necesario esperar 5 impulsos de reloj para obtener un valor convertido a la salida del conversor.

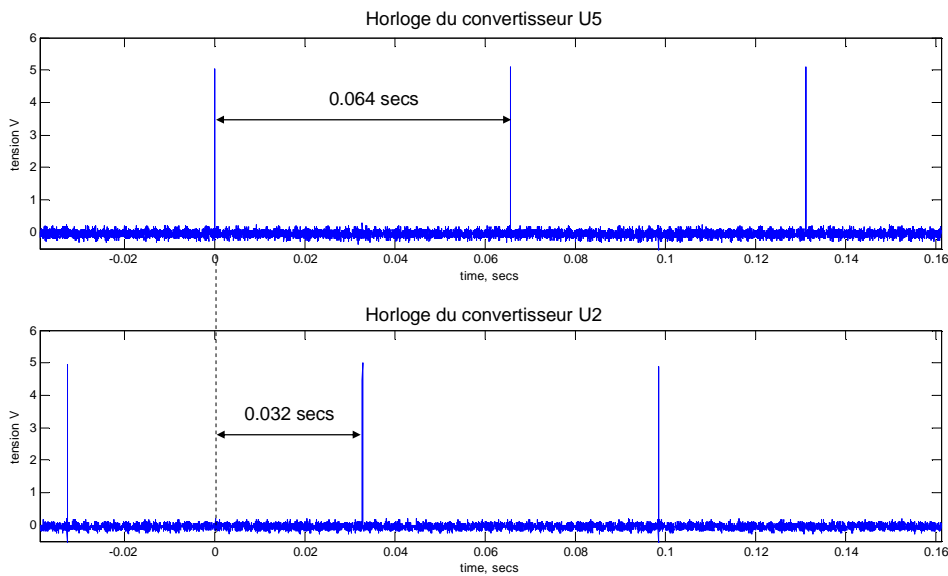


Figura 15. Señal de reloj de los conversores

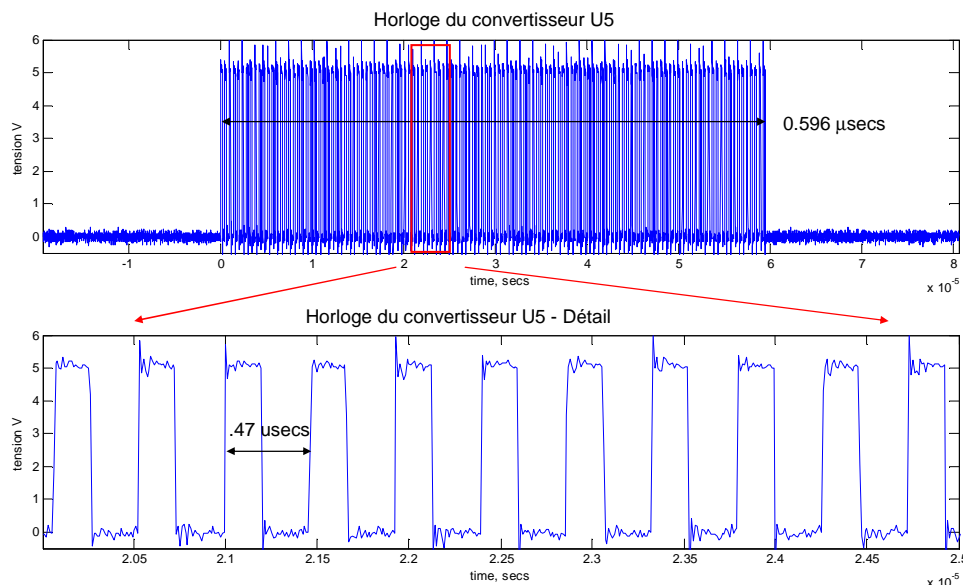


Figura 16. Zoom de un de los impulsos de la señal de reloj de los conversores

Vamos a ver ahora la integración del chip TDSC en el receptor final.

- Tensión de alimentación de 3.3 V.
- Dos muestreadores con una capacidad de 128 muestras cada uno.
- Frecuencia de muestreo en torno a los 5.5 GHz (variable entre 1.45 GHz y 7.5 GHz).

Verificación de las principales funciones del chip TDSC

21

La variación de la frecuencia de muestreo del chip TDSC ha sido comprobada modificando la tensión de control « Delay adjust » a la entrada del circuito de entrelazamiento y midiendo el retardo obtenido sobre la señal a la salida de este circuito. Hemos elegido algunos valores de tensión a este efecto y presentamos aquí dos ejemplos. Para una tensión de 0.96 V el retardo es de 88 ns, lo que corresponde a una frecuencia de $128/88 = 1.45$ GHz, y para una tensión de 2 V el retardo es de 24 ns, lo que corresponde en este caso a una frecuencia de $128/24 = 5.3$ GHz. Todo esto es ilustrado con la ayuda de la figura 18.

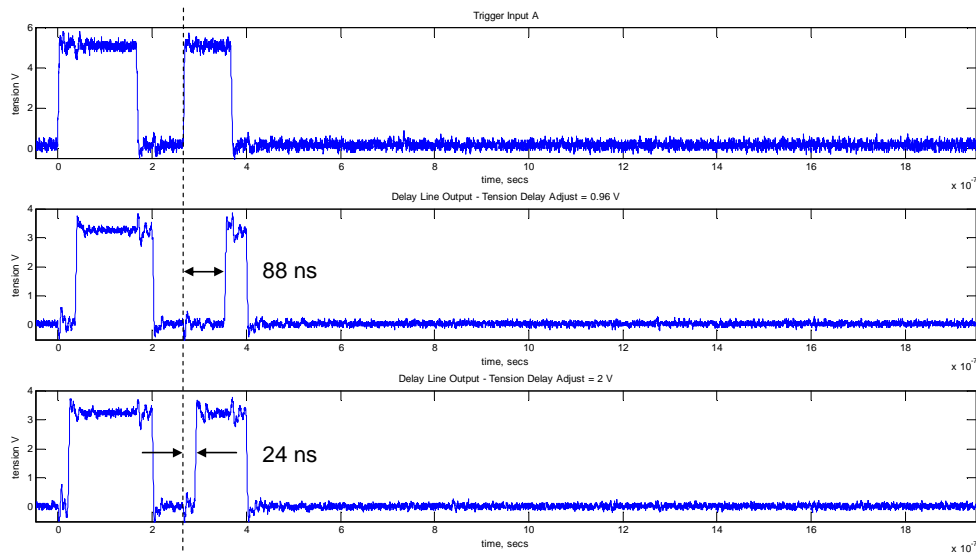


Figura 18. Variación de la frecuencia de muestreo del chip TDSC variando la tensión de control « Delay adjust »

El segundo test permite estudiar el grado de linealidad de la respuesta de los muestreadores en el chip TDSC. Para esto el receptor es excitado por una señal continua que varía entre 0.3 V y 2 V. Los resultados obtenidos sobre la ganancia asociada están resumidos en la figura 19, que presenta la evolución de la tensión de salida en función de la tensión de entrada.

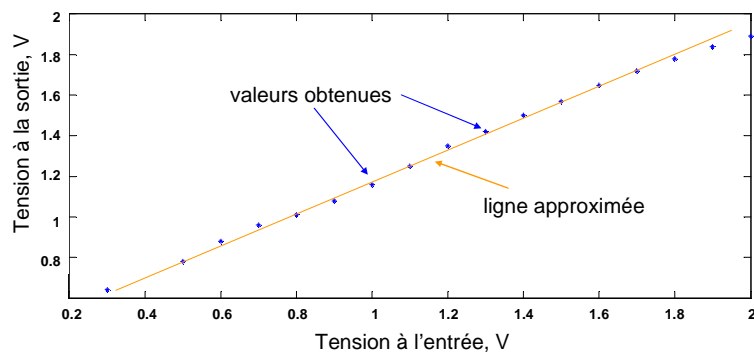


Figura 19. Evolución de la tensión de salida en función de la tensión de entrada

Ésta parece corresponder a los valores obtenidos durante la concepción del chip.

Diseño de la tarjeta de recepción

Para la concepción y la realización de la tarjeta receptora, hemos utilizado el programa de diseño de circuitos impresos PROTEUS. Este programa es un entorno integrado concebido para la realización completa de proyectos electrónicos: diseño, simulación, depuración y construcción. Se compone de cuatro elementos perfectamente integrados entre ellos: ISIS, ARES, PROSPICE y VS. Nosotros hemos utilizado solamente los dos primeros.

El esquema eléctrico de la tarjeta ha sido obtenido con la herramienta ISIS, como podemos ver en la figura 20. En esta tarjeta están integrados el chip TDSC (CQFP1), el microcontrolador (U1) y los dos convertidores (U2 y U5), así como el circuito MAX232 (U3) y los elementos necesarios (resistencias, capacidades, cuarzo, puerto serie) para el buen funcionamiento de los circuitos activos.

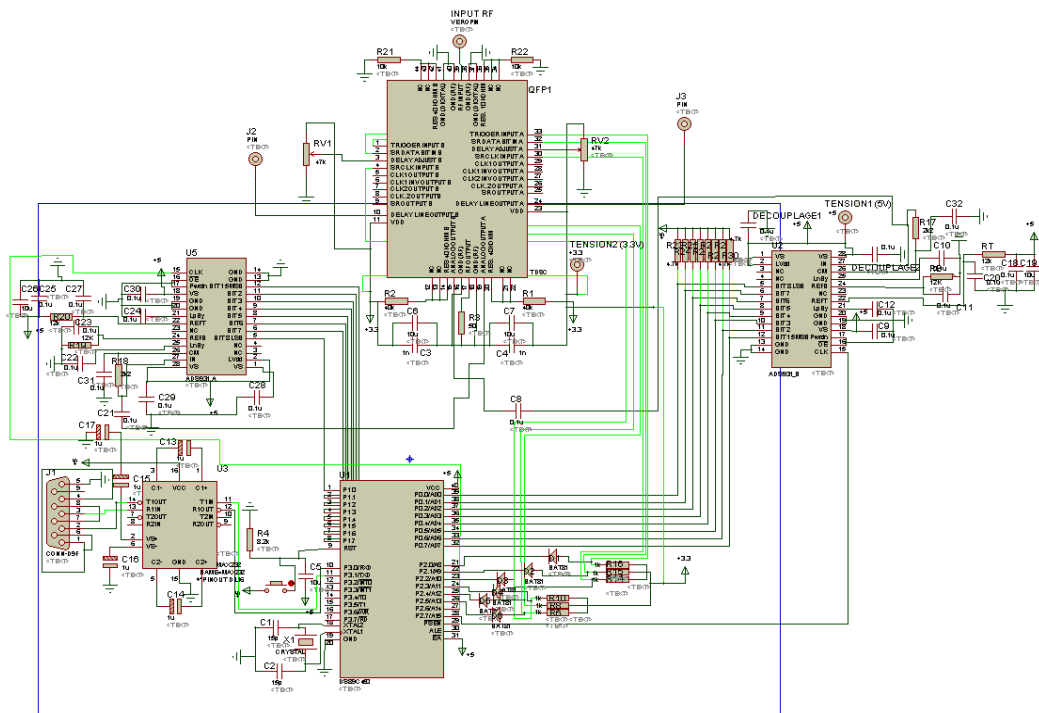


Figura 20. Esquema eléctrico completo del receptor

La implementación de los componentes sobre el circuito impreso es presentada en la figura 21. Esto ha sido obtenido gracias a la herramienta ARES, que permite la concepción de circuitos impresos con el posicionamiento automático de los elementos y la generación automática de las pistas.

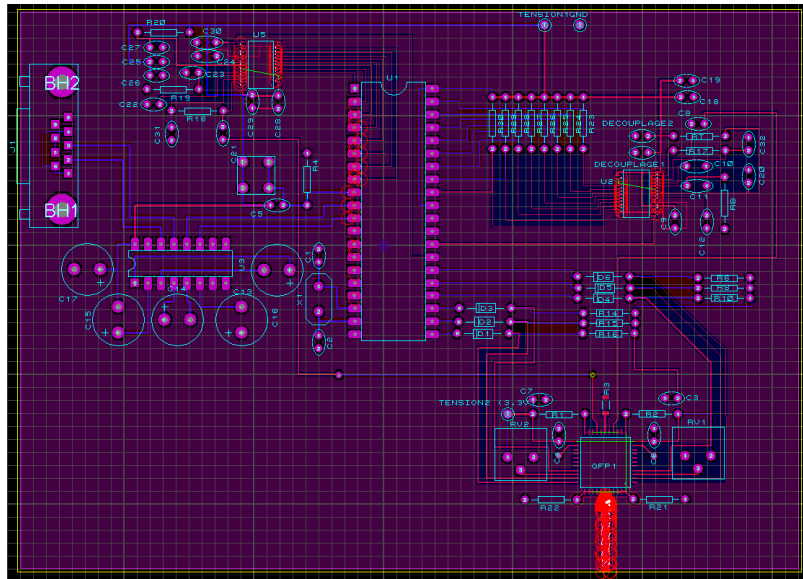


Figura 21. Implementación de los componentes del receptor completo

En la figura 22 presentamos una fotografía del receptor de test completo:

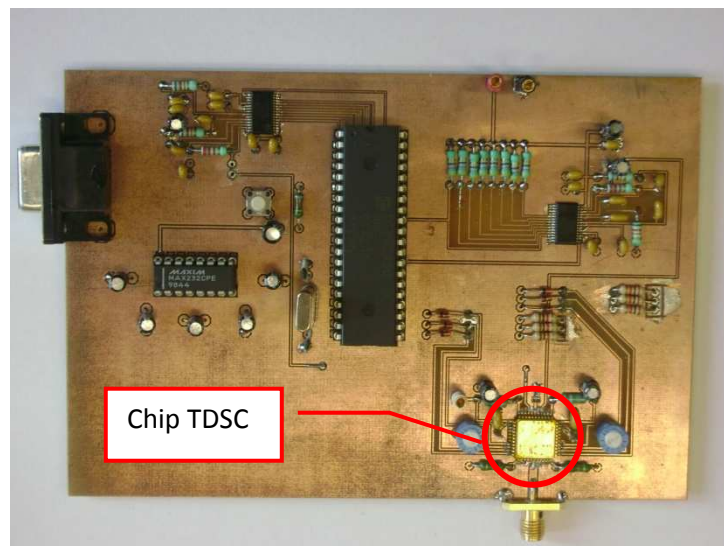


Figura 22. Fotografía del receptor completo

Conclusiones

En esta parte se ha comprobado en correcto funcionamiento de los convertidores analógico/digitales y del enlace a través del puerto serie de la tarjeta receptora con el ordenador, así como la generación de las señales de control proporcionadas por el microcontrolador.

Después de haber presentado el circuito de conversión de tensión entre el microcontrolador (5 V) y el chip TDSC (3.3 V), hemos comprobado que éste funciona correctamente con respecto a las prestaciones esperadas.

El programa del microcontrolador puede ser mejorado para optimizar la transmisión de los datos muestreados y digitalizados, gestionando el espacio de memoria de dicho controlador.

Es posible aumentar la velocidad de transmisión del microcontrolador si utilizamos los registros de configuración específicos del microcontrolador DSC89C450. Esta posibilidad podrá ser explotada por el proceso de sincronización entre el emisor y el receptor.

2.3. Circuito emisor

El emisor cuyo esquema es presentado en la figura 23, es una tarjeta que se compone de un microcontrolador que genera señales cuadradas. Estas señales atacan a un amplificador operacional funcionando como comparador. La salida de este último está conectada a un circuito de microondas que se compone de una línea cortocircuitada y de un diodo SRD (Step Recovery Diode). Este diodo entrega los impulsos UWB a emitir.

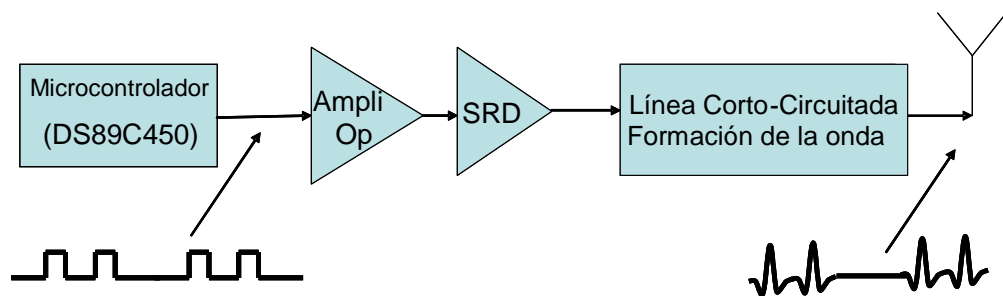


Figura 23. Esquema del principio de funcionamiento del emisor

Un ejemplo de un impulso proporcionado por el microcontrolador, controlando el amplificador operacional que polariza el diodo SRD, se presenta en la figura 24.

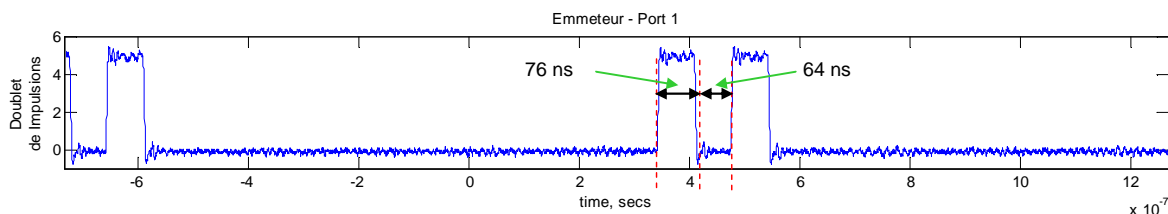


Figura 24. Impulsos o dobletes generados por el emisor

Se pueden generar diferentes trenes de impulsos en los que T_D y T_S son variables. Podemos ver en el ejemplo de la figura 24 una posible construcción del tren de impulsos con el material utilizado, cuyo ciclo de reloj es de 33.3 ns y, en este caso, T_D es 64 ns. Teniendo en cuenta el receptor, la señal utilizada para la versión final del emisor tendrá un valor de T_D de 1 μ s.

Las figuras 25 y 26 representan, respectivamente, el esquema eléctrico del emisor realizado con ISIS y su circuito impreso realizado con ARES.

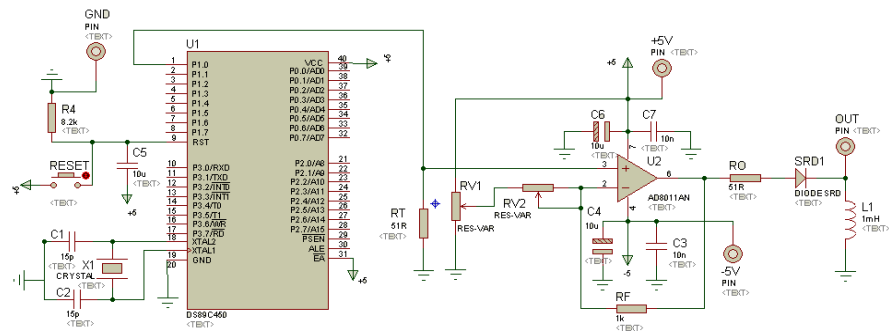


Figura 25. Esquema eléctrico del emisor

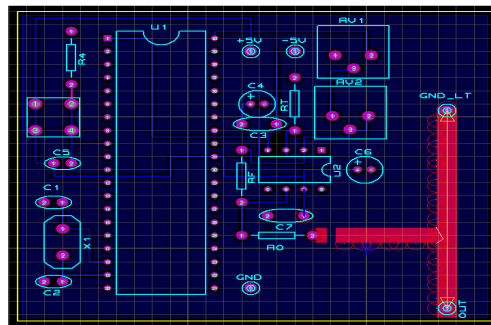


Figura 26. Circuito impreso del emisor

La foto del circuito emisor se presenta aquí debajo en la figura 27:

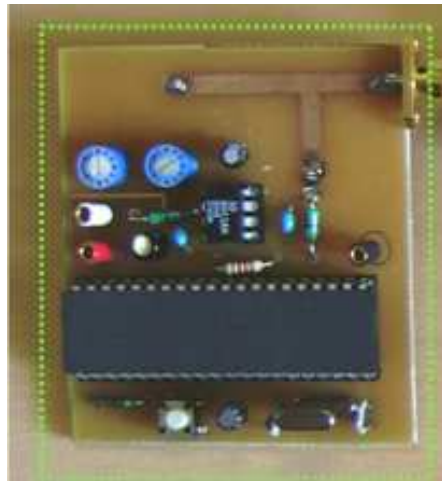


Figura 27. Fotografía del emisor

2.4. Resultados obtenidos para validar el funcionamiento de las dos tarjetas elaboradas

Las medidas presentadas aquí son obtenidas a partir de los circuitos realizados durante este proyecto. La construcción del tren de impulsos elegido se presenta en la figura 28. El retardo T_D es de $1\ \mu\text{s}$ y el periodo de repetición es de $67\ \mu\text{s}$.

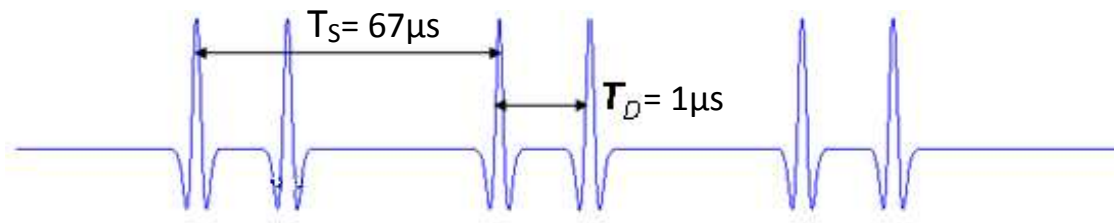


Figura 28. Formato del tren de impulsos construido

En la figura 29 podemos ver la forma de onda de uno de los impulsos del doblete, a la salida del circuito emisor, capturado con un osciloscopio rápido. El impulso obtenido tiene una amplitud de $1.16\ \text{V}$ y una anchura a media altura de $1\ \text{ns}$. Esto valida el funcionamiento del emisor que genera los impulsos de manera independiente de todo generador externo.



Figura 29. Medida del impulso a la salida del emisor

Las tarjetas de emisión y recepción han sido conectadas entre ellas, como se muestra en la figura 30, con el fin de registrar impulsos reales mediante el ordenador.

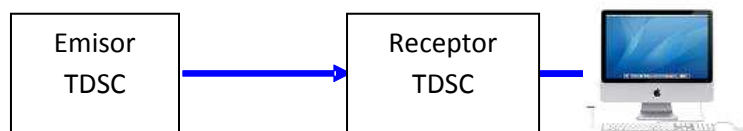


Figura 30. Test de emisión/recepción del sistema TDSC

El emisor envía un tren de impulsos compuesto por dobletes de impulsos separados 1 μ s. El chip TDSC del receptor permite desplazar los dos impulsos, uno con respecto al otro, y muestrearlos para ser convertidos y grabados finalmente en la memoria interna del microcontrolador, antes de ser transferidos hacia el ordenador mediante el puerto serie y después ser tratados por Matlab.

La figura 31 presenta un primer registro de dobletes transmitidos durante este test.

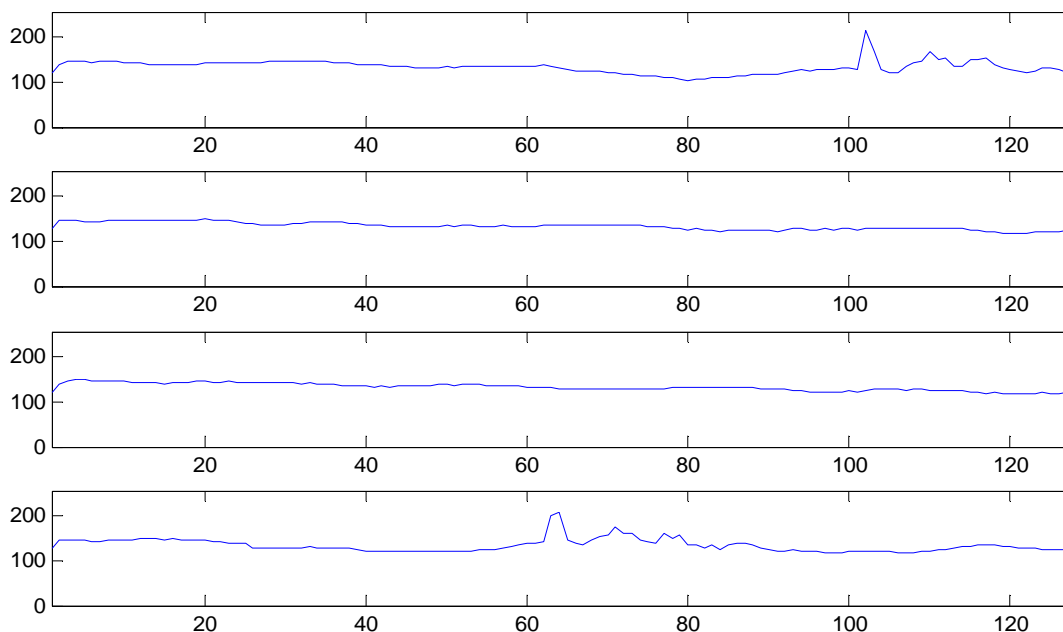


Figura 31. Diferentes registros de impulsos

Como podemos ver en esta figura 31, sin sincronización es difícil de « ver » un doblete completo en la ventana de captura.

Hemos realizado un programa en Matlab que realiza la correlación entre los dos impulsos grabados y permite conservar las dos formas de onda, dando el resultado correcto de la correlación, con el fin de permitirnos representar los resultados presentes en la figura 32, en los cuales podemos observar los dos impulsos de un doblete real transmitido.

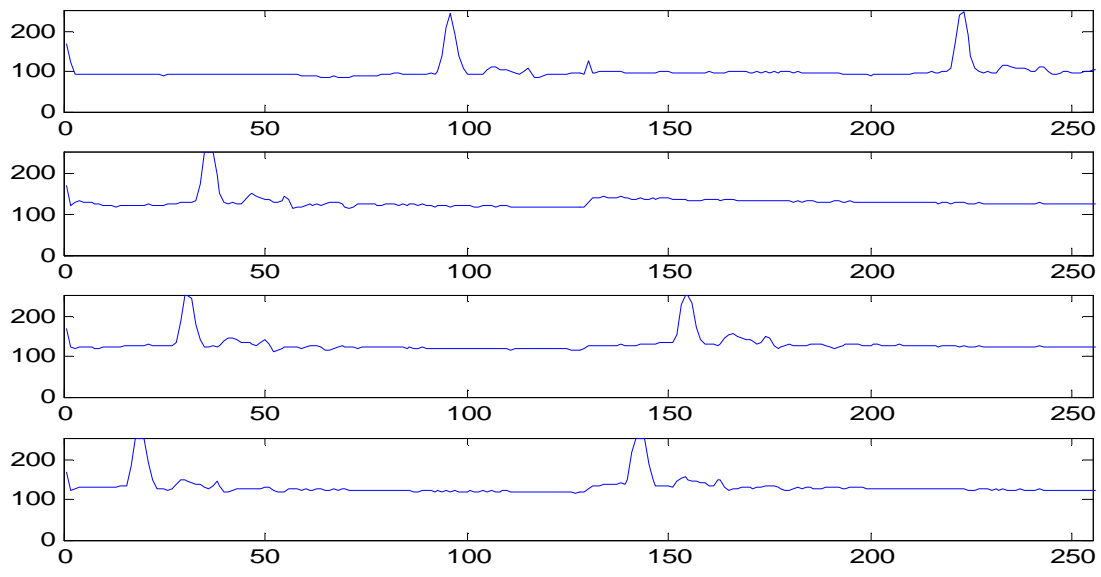


Figura 32. Algunos símbolos registrados en el ordenador, después de la transmisión

Conclusión

Este trabajo nos ha permitido realizar las tarjetas de emisión y de recepción con el objetivo de transmitir impulsos gracias a nuestro sistema TDSC, y de obtener la captura de trenes de impulsos reales para poder trabajar sobre la sincronización emisor/receptor, con el objetivo de implementar un algoritmo y hacer funcionar el sistema de manera autónoma.

En efecto, con el fin de estudiar el potencial de este sistema sobre aplicaciones como la localización en interiores, por ejemplo, debemos poseer un prototipo que nos permita trabajar con datos que se aproximen lo mejor posible a la realidad. Para ello, se dispone de antenas que serán utilizadas durante la continuación de este proyecto.

Capítulo 3

3. Estudio de la compatibilidad del estándar IEEE 802.15.4a con la técnica TDSC

El receptor descrito en el capítulo anterior permite la detección de señales TR-UWB utilizando la técnica TDSC, cuyo principio reposa en la utilización de un retardo digital programable. Este receptor puede ser utilizado en redes inalámbricas de baja velocidad para aplicaciones de localización, telemedicina y redes de sensores, por ejemplo, en las cuales se utilizan señales impulsionales UWB. Una de las normas UWB indicada para redes inalámbricas de baja velocidad y aparecida recientemente es la IEEE 802.15.4a. Nosotros hemos decidido simular este estándar en Matlab y hemos propuesto dos posibles modificaciones a aplicar sobre su trama con el fin de hacerla compatible con nuestra técnica de detección. En lo que sigue será descrito el estándar 802.15.4a, del cual algunos complementos son dados en los anexos. Después serán presentados los resultados de las simulaciones en Matlab.

3.1. Descripción del estándar IEEE 802.15.4a

La norma IEEE 802.15.4 o « Zigbee » fue definida en 2003 para las redes inalámbricas de baja velocidad, con una complejidad sencilla y un bajo consumo. Sin embargo, no permite la localización de individuos u objetos con una gran precisión. Con el objetivo de remediar esto, en 2007 fue aprobada una nueva norma, la IEEE 802.15.4a, en la cual fue modificada la capa PHY de Zigbee, permitiendo satisfacer las funciones de localización en las redes inalámbricas de baja velocidad utilizando señales impulsionales UWB. En 2009 se añadieron complementos, principalmente para integrar las nuevas bandas de frecuencias (< 1 GHz) de China y Japón, y aparecieron las normas 802.15.4c y 802.15.4d [14].

El grupo IEEE ha dividido el espectro en 16 sub-bandas de 499.2 MHz, pero algunas bandas son más anchas y se solapan con otras sub-bandas. Las bandas 0 y 9 son obligatorias, por lo que el sistema deberá, por tanto, utilizar estas bandas. Por otro lado, también podemos señalar una división en tres bandas: banda sub-gigahercio ($f < 1$ GHz), bandas bajas (3.1 y 4.8 GHz) y bandas altas (entre 6 y 10.6 GHz).

Las bandas y sus frecuencias asociadas aparecen precisadas en la tabla 1. Las frecuencias centrales son elegidas de manera que correspondan a un múltiplo entero de las frecuencias de los osciladores existentes: 13 MHz o 19.2 MHz.

Banda	Frecuencia (MHz)	Ancho de banda (MHz)	Tipo de banda
0	499.2	499.2	Sub-gigahercio
1	3494.4	499.2	Banda baja
2	3993.6	499.2	Banda baja
3	4492.8	499.2	Banda baja (obligatoria)
4	3993.6	1331.2	Banda baja
5	6489.6	499.2	Banda alta
6	6988.8	499.2	Banda alta
7	6489.6	1081.6	Banda alta
8	7488.0	499.2	Banda alta
9	7987.2	499.2	Banda alta (obligatoria)
10	8486.4	499.2	Banda alta
11	7987.2	1331.2	Banda alta
12	8985.6	499.2	Banda alta
13	9484.8	499.2	Banda alta
14	9984.0	499.2	Banda alta
15	9484.8	1354.97	Banda alta

Tabla 1. Bandas de frecuencia del estándar IEEE 802.15.4a.

Estructura de la trama

La figura 33 ilustra el proceso de formación de los datos al nivel de la capa física del estándar 802.15.4a [8; 15 ; 16 ; 17]. En lo que concierne a la emisión, podemos señalar los pasos: formación, codificación, modulación, inserción del preámbulo, generación de los impulsos UWB mediante un generador de impulsos (Pulse Shaper) y emisión mediante la antena tras efectuar la transposición en frecuencia (bloque RF).

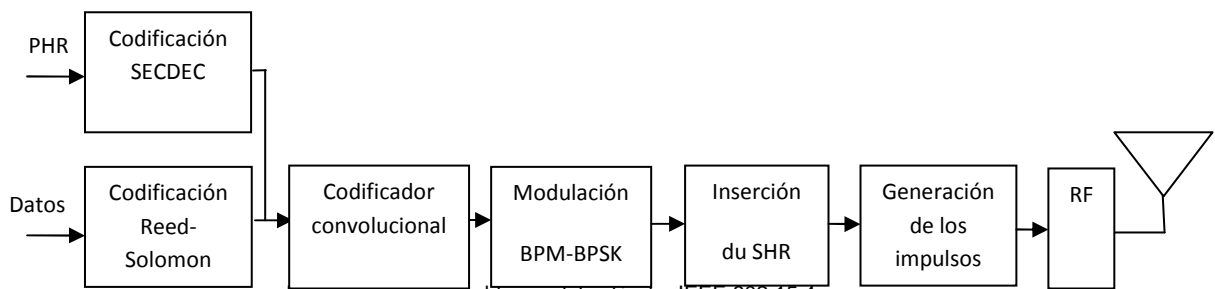


Figura 33. Esquema bloque del estándar IEEE 802.15.4a

Después de la codificación, la trama del estándar 802.15.4a se compone de tres partes: la cabecera de sincronización SHR (Synchronization Header), la cabecera de la capa física PHR (Physical Header) y los datos EPSDU (Encoded Physical Service Data Unit), como se muestra en la figura 34. El SHR es transmitido primero, seguido del PHR y finalmente del PSDU.

SHR 16, 64, 1024 o 4096 símbolos	PHR 19 símbolos	EPSDU 0 a 1209 símbolos
-------------------------------------	--------------------	----------------------------

Figura 34. Formato de una trama 802.15.4a

La cabecera SHR permite la ejecución de los algoritmos del receptor: control automático de ganancia, sincronización gruesa y fina, estimación del canal, deriva de los relojes y medida de distancia. Está formada por el preámbulo y el delimitador de inicio de trama SFD (Start of Frame Delimiter). Existen cuatro posibles cuatro longitudes para el preámbulo (16, 64, 1024 o 4096 símbolos), permitiendo especificar los paquetes para la localización (preámbulo largo) o para la transmisión de datos (preámbulo corto). Cada subred, funcionando en una de las bandas entre la 0 la 15, es identificada por un « código de preámbulo ». Este código se compone de una secuencia de 31 o 127 impulsos ternarios {-1, 0 et 1}, teniendo muy buenas propiedades de autocorrelación (para limitar los malos enganches) y de intercorrelación (para diferenciar las redes funcionando en una misma banda). Esta secuencia de 31 o 127 impulsos es repetida 16, 64, 1024 o 4096 veces. La velocidad del SHR es la velocidad del símbolo de base.

La longitud de la trama y la velocidad de los datos son precisados con la ayuda del PHR, que contiene siempre 19 símbolos. La modulación utilizada es una BPM (Burst Position Modulation) asociada a una modulación de fase BPSK (Binary Phase Shift Keying) y los datos PHR son codificados por un codificador SECDEC (Single Error Correct Double Error Detect). La velocidad empleada es de 110 o 850 Kbps.

Los datos se codifican mediante un código Reed Solomon (tipo $(K+8, K)$), donde se añaden 48 bits de paridad a los datos. Estos son seguidos por un código convolucional sistemático de tipo $R = \frac{1}{2}$ que genera un bit de posición y un bit de polaridad, formando así el EPSDU. Estos dos códigos forman la codificación de canal se modulan en BPM-BPSK. El EPSDU puede contener entre 0 y 1209 símbolos y la velocidad, la cual puede ser variable, aparece especificada en el PHR.

La utilización de estos dos tipos de modulación permite utilizar una misma señal emitida con un receptor coherente (modulación BPSK) o no coherente (modulación BPM). Con la intención de ilustrar esto, la figura 35 representa un símbolo BPM-BPSK con una duración T_{dsym} y constituido por N_C chips (un chip puede contener un impulso o estar vacío) de duración T_C ($T_{dsym} = N_C \cdot T_C$). Cada símbolo se divide en dos intervalos de duración $T_{BPM} = T_{dsym}/2$. Cada uno de estos dos intervalos se divide a su vez en dos sub-intervalos: el primero define las posibles posiciones de la ráfaga y el segundo es un intervalo de guarda que limita las interferencias intersimbólicas. Una ráfaga de duración T_{burst} compuesta por N_{cpb} impulsos ($T_{burst} = N_{cpb} \cdot T_C$) es colocada en uno de los dos intervalos. Esta es modulada en posición y

después en fase, donde la polaridad de los impulsos puede estar invertida. Un símbolo contiene una sola ráfaga de duración T_{burst} muy inferior a T_{BPM} , permitiendo así el acceso multiusuario mediante la utilización de un código de salto temporal (TH : Time Hopping) entre las diferentes posibles posiciones de una ráfaga para el ensanchamiento del espectro. Esto aumenta la robustez de la comunicación de cara a las interferencias. Se definen dos frecuencias medias de repetición de los impulsos (PRF = Pulse Repetition Frequency) obligatorias: 15.6 MHz (High PRF) y 3.9 MHz (Low PRF). La PRF media fija la amplitud de los pulsos y por tanto la potencia de transmisión. Los impulsos son repetidos dentro de una ráfaga con una frecuencia PRF_{pic} de 499.2 MHz para las bandas 0 a 3, 5, 6, 8 a 10 y 12 a 14, y por tanto: $T_c = 1/PRF_{pic} = 2$ ns. Para las bandas 4 y 11, 7 y 15, $T_c = 0.75$ ns, 0.92 ns y 0.74 ns respectivamente.

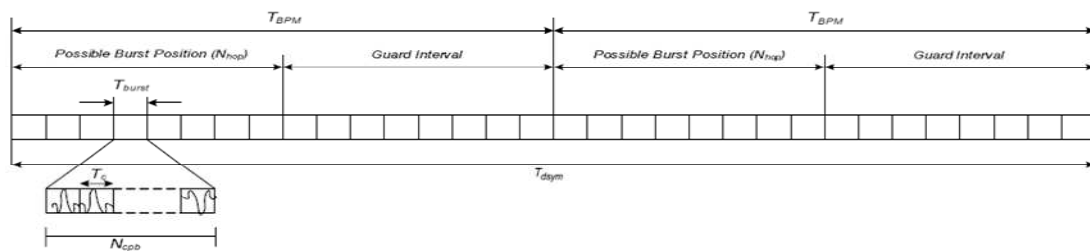


Figura 35. Cabecera EPSDU de la trama IEEE 802.15.4a

Todos estos parámetros que especifican la trama IEEE 802.15.4a para las diferentes bandas están resumidos en las tablas 39a, 39b y 39c de la norma [8].

3.2. Compatibilidad con el sistema TDSC

Hemos estudiado la compatibilidad del sistema TDSC con la capa física del estándar IEEE 802.15.4a. Como recuerdo, la trama utilizada está compuesta de una ráfaga de impulsos con una duración ≈ 2 ns. Cada impulso puede ser positivo, negativo o nulo para representar el « +1 », el « -1 » o el « 0 » respectivamente. Este estudio es necesario, ya que para la recepción de señales UWB utilizando el chip TDSC, cada símbolo debe estar compuesto de un doblete de impulsos, lo cual no es así en el caso de la trama del estándar 802.15.4a. Es por tanto indispensable adaptar esta trama al sistema TDSC: cada impulso de la trama debe estar precedido por el impulso de referencia con el fin de formar el doblete. Hemos estudiado por ello la posibilidad de añadir este segundo impulso al nivel del emisor que genera la trama 802.15.4a.

A la salida del bloque « Preamble Insertion », la trama está compuesta de símbolos ternarios {+1, -1, 0}. Para formar el doblete TR-UWB, debemos preceder un impulso de referencia (positivo) con una separación temporal T_D respecto al impulso de datos. El microcontrolador se encargará de añadir estos impulsos como se muestra en la figura 36. A continuación, el « formador de onda » compuesto por el amplificador operacional, el diodo SRD y la línea cortocircuitada, generará las formas de onda a transmitir. Para emitir en una de las

bandas especificadas por el estándar, conviene utilizar un mezclador con el fin de centrar la señal en la frecuencia central de la banda a utilizar.

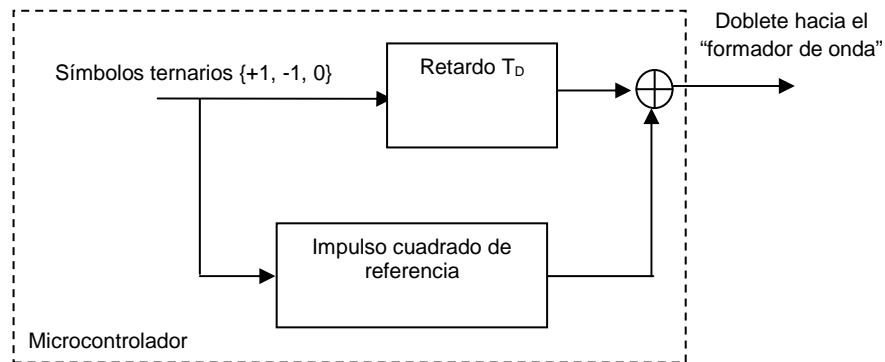


Figura 36. Esquema bloque para la generación de dobletes

La inserción de los impulsos de referencia puede hacerse de dos maneras diferentes:

- 1) El impulso de referencia y el impulso de datos situados en dos chips diferentes. El retardo T_D es fijado a 2 ns.
- 2) El impulso de referencia situado en el mismo chip que el chip de datos. Un doblete estará presente, por tanto, en cada chip. Así, la duración de un impulso ha sido reducida a 1 ns, al igual que T_D .

A continuación, gracias a las simulaciones Matlab [18], será generada una trama del estándar IEEE 802.15.4a y, después, la adaptaremos al sistema TDSC. Así, los espectros de la trama original y de la trama modificada podrán ser comparados.

3.3. Simulación en Matlab

Para llevar a cabo estas simulaciones, hemos elegido la banda nº 3 con una PRF media de 15.6 MHz, una frecuencia central F_C de 4.4928 GHz y una anchura de banda de 499.2 MHz.

La formación de una trama IEEE 802.15.4a requiere la generación de las cabeceras SHR, PHR y EPSDU. A continuación, éstas son concatenadas para formar la trama conforme al estándar.

Con respecto al bloque SHR, hemos utilizado un código de longitud 31 que satisface las condiciones de autocorrelación y de intercorrelación [8], así como una longitud de $N_{\text{sync}} = 16$, por lo que esta secuencia de 31 impulsos ternarios será repetida 16 veces. El código utilizado es el siguiente:

$$S_i = +1+100+100-1-1-1+1-10+1+1-1000+10+10-1+10+10000$$

Este corresponde al preámbulo 6 que aparece precisado en el documento del estándar [8]. Este código es extendido insertando 15 ceros después de cada elemento ternario del

código Si (extensión realizada utilizando el operador de Kronecker δ_L con $L = 16$). Del mismo modo, el campo SFD ha sido elegido a partir del documento de la norma. Tiene como longitud $N_{SFD} = 8$:

$$SFD = 0 \text{ Si } 0 \text{ -Si Si } 0 \text{ 0 -Si}$$

La duración del preámbulo es $T_{sync} = 15.9 \mu s$ y la del SFD es $T_{SFD} = 7.9 \mu s$. Así, la duración total de la cabecera SHR es $T_{SHR} = T_{sync} + T_{SFD} = 23.8 \mu s$. Hay un total de $N_{sync} + N_{SFD}$ ($16 + 8 = 24$) símbolos en el SHR, y la velocidad del SHR es de 1.01 Msímbolos/s.

El bloque PHR contiene siempre 19 símbolos modulados en BPM-BPSK con $N_c = 512$ chips/símbolo. La velocidad es de 0.85 Mb/s y la duración del PHR es $T_{PHR} = 19.48 \mu s$.

Para la parte de datos, hemos elegido transmitir 5 símbolos, que también están modulados en BPM-BPSK. Hay siempre $N_c = 512$ chips/símbolo y la velocidad es de 0.85 Mb/s. Para simplificar la programación, hemos considerado en este ejemplo que los datos han pasado ya por todas las etapas de codificación, a saber: codificación Reed-Solomon, añadidura de la cabecera PHR, codificación convolucional, modulación BPM-BPSK y extensión. La duración total del campo de datos EPSDU es $T_{EPSDU} = 5.12 \mu s$ (5 símbolos x 512 chips x 2 ns).

Una vez formada la trama, ésta es aplicada al generador de impulsos. Este último ha sido modelado por un filtro en raíz cuadrada del coseno realzado de orden 40 con un factor de roll-off de 0.25.

Simulación de una trama IEEE 802.15.4a

La primera simulación concierne la generación de una trama IEEE 802.15.4a y la observación del espectro asociado. La figura 37 presenta la evolución temporal de la trama a la salida del generador de impulsos. La trama está compuesta, por tanto, de impulsos ternarios $\{+1, -1, 0\}$ y tiene una duración: $T_{TRAMA} = T_{SHR} + T_{PHR} + T_{EPSDU} = 48.4 \mu s$.

La figura 38 permite observar con más precisión la forma de los impulsos generados por el filtro en raíz cuadrada del coseno realzado, cuya amplitud es de 0.3 V y tienen una duración de 2 ns (duración especificada por el estándar).

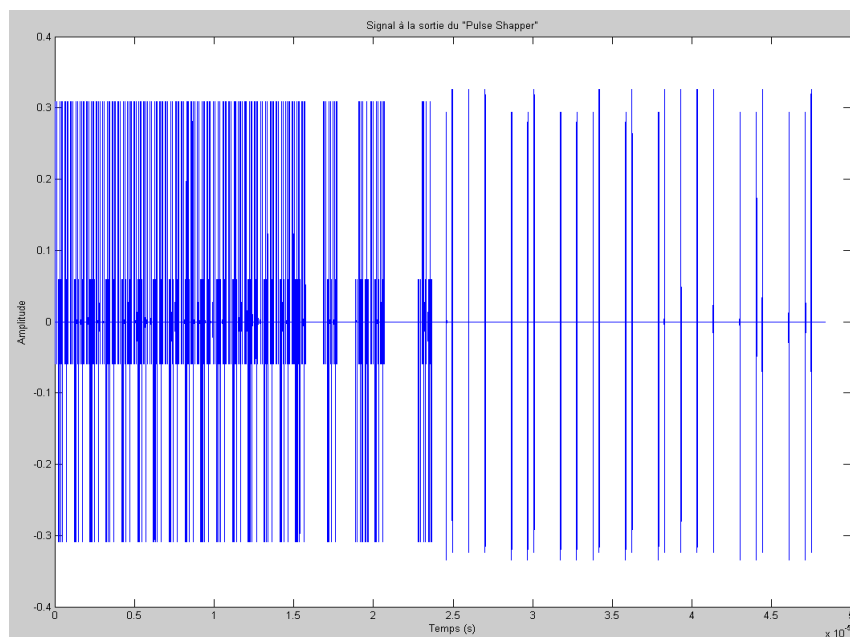


Figura 37. Evolución temporal de la trama a la salida del « pulse shaper »

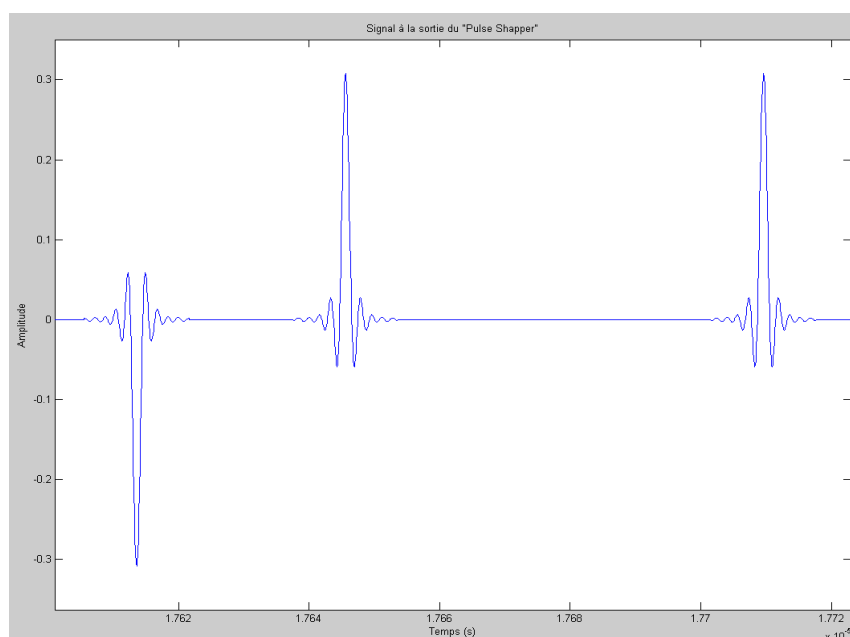


Figura 38. Impulsos generados por el filtro en raíz cuadrada del coseno realzado (pulse shaper)

En la figura 39 se presenta el espectro de la trama a la salida del « pulse shaper ». La anchura de banda es de alrededor de 600 MHz.

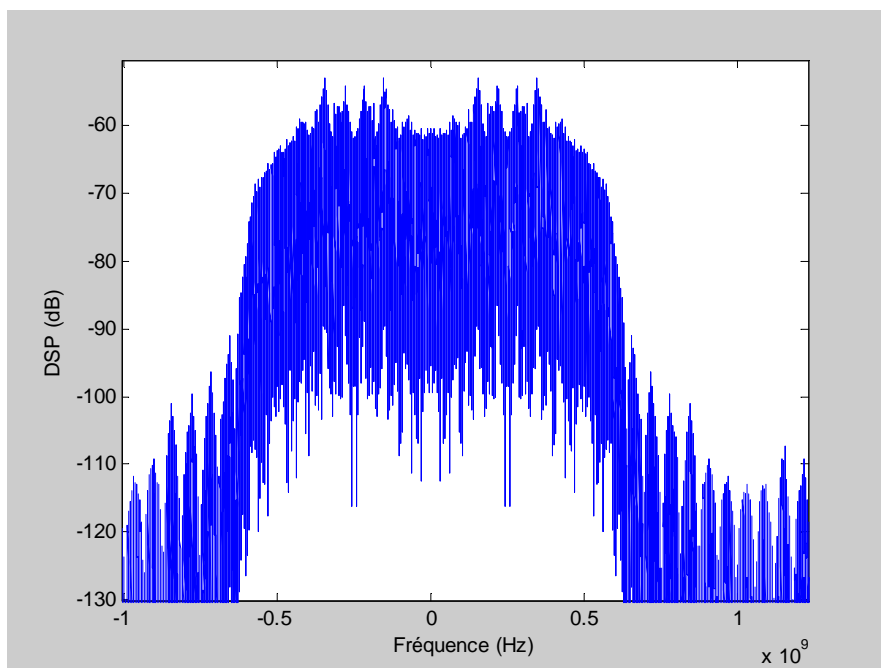


Figura 39. Espectro de la trama 802.15.4a a la salida del « pulse shaper ».

Siendo la frecuencia central del canal elegido de 4.4928 GHz, la señal en banda base es desplazada hasta esta frecuencia utilizando un mezclador (mixer). La figura 40 presenta el espectro de la trama 802.15.4a alrededor de esta frecuencia central. Sería necesario tener en cuenta todos los parámetros físicos de los circuitos utilizados para la generación de los impulsos para evaluar realmente la amplitud del espectro de emisión.

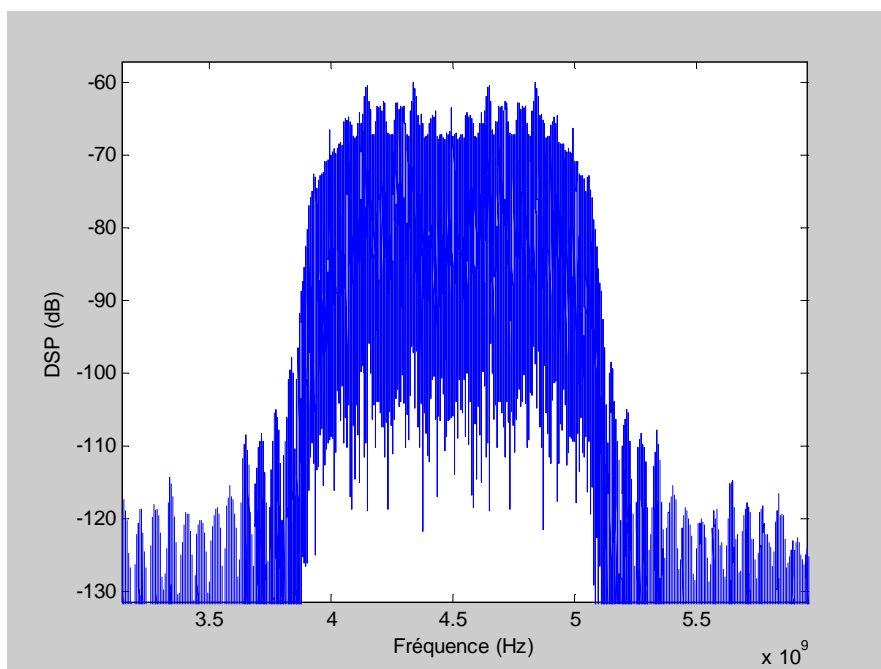


Figura 40. Espectro de la trama 802.15.4a a la salida del mezclador

Simulación de una trama 802.15.4a con dobletes

El doblete en 2 chips diferentes

En las siguientes simulaciones, los impulsos de referencia son insertados en un nuevo chip con el fin de formar los dobletes, por lo que el tiempo de transmisión aumenta. Los impulsos utilizados están en conformidad con la norma. En nuestra simulación consideramos el tiempo T_D entre los impulsos del doblete de 2 ns (la duración de un chip).

La figura 41 presenta la evolución temporal de la trama con dobletes a la salida del « pulse shaper ». La figura 42 permite observar los impulsos más claramente.

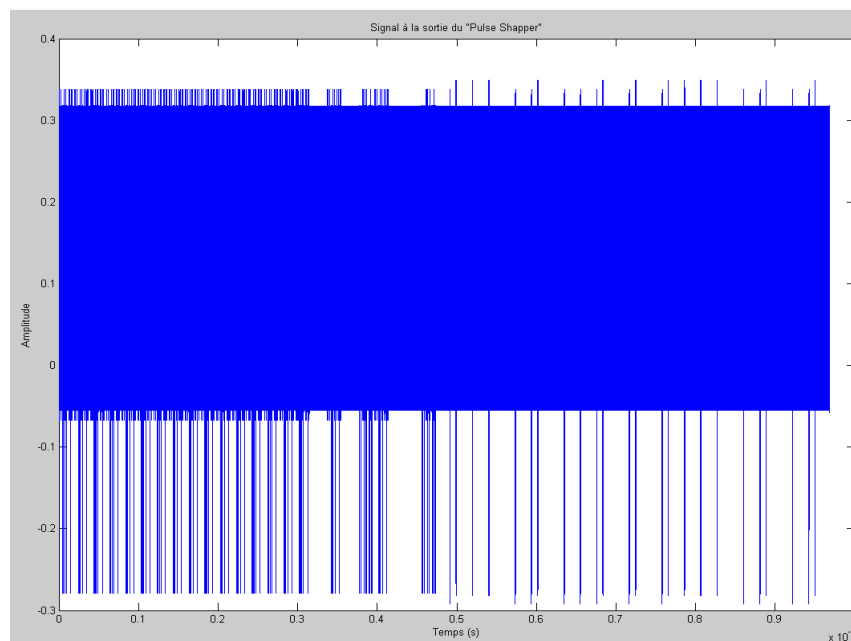


Figura 41. Trama 802.15.4a con dobletes

La duración total de la trama es el doble que la de la trama sin ellos. La velocidad de los datos se reduce por tanto a la mitad.

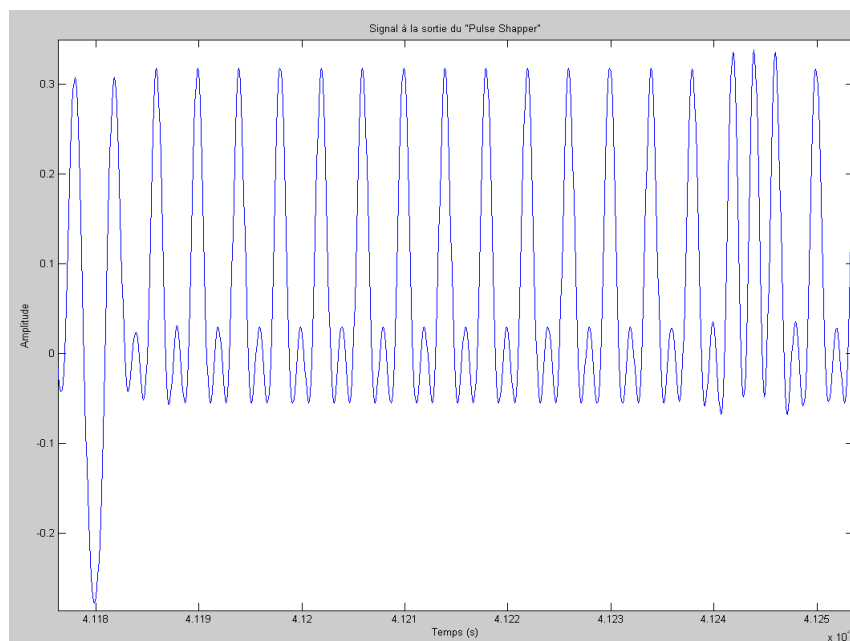


Figura 42. Detalle de los dobletes en la trama 802.15.4a

El número de impulsos ha sido doblado, lo que conduce a la aparición de picos parásitos más importantes en el dominio espectral. El espectro presentado en la figura 43 permite constatar este efecto: el ancho de banda no se modifica, pero los picos parásitos se ven claramente. En este caso convendría alisar el espectro con vistas a satisfacer las especificaciones del estándar.

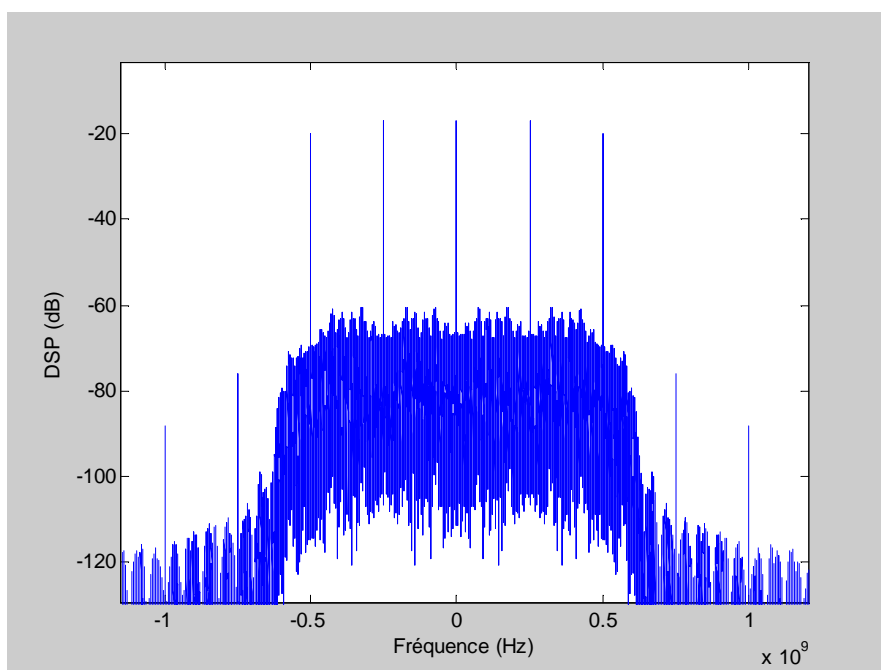


Figura 43. Espectro de la trama con dobletes a la salida del « pulse shaper »

El doblete en un mismo chip

En las simulaciones presentadas a continuación, el doblete se sitúa en la duración de un chip. La duración de cada impulso debe ser reducida, pues tenemos en un solo chip dos impulsos separados por un tiempo T_D sin olvidar el tiempo de separación entre dobletes consecutivos. Así, la duración de los impulsos ya no es de 2 ns sino de 1 ns. Las figuras 44 y 45 presentan la evolución temporal de la trama, modificada de esta manera, a la salida del « formador de onda » y una ampliación de ésta, respectivamente.

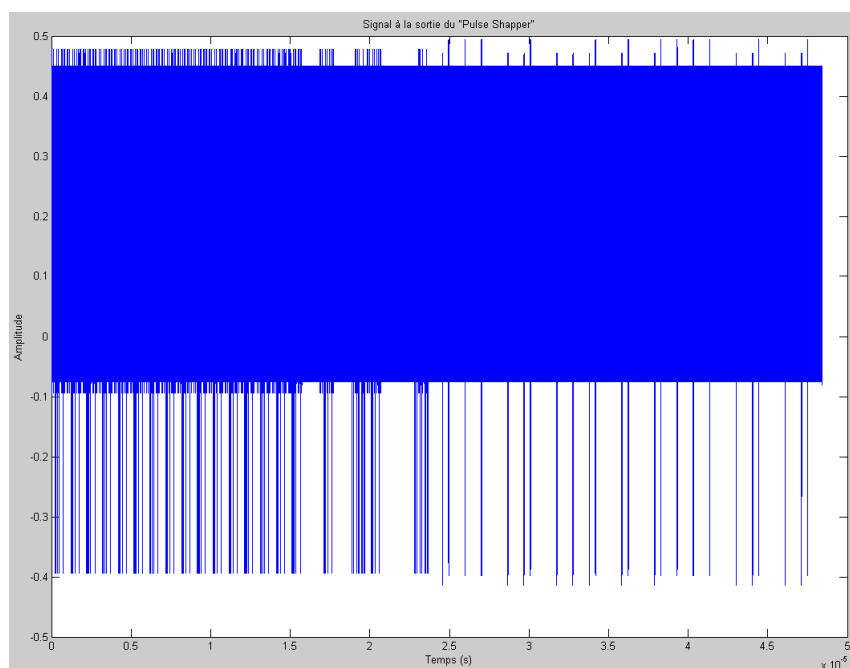


Figura 44. Trama 802.15.4a con cada doblete en un solo chip

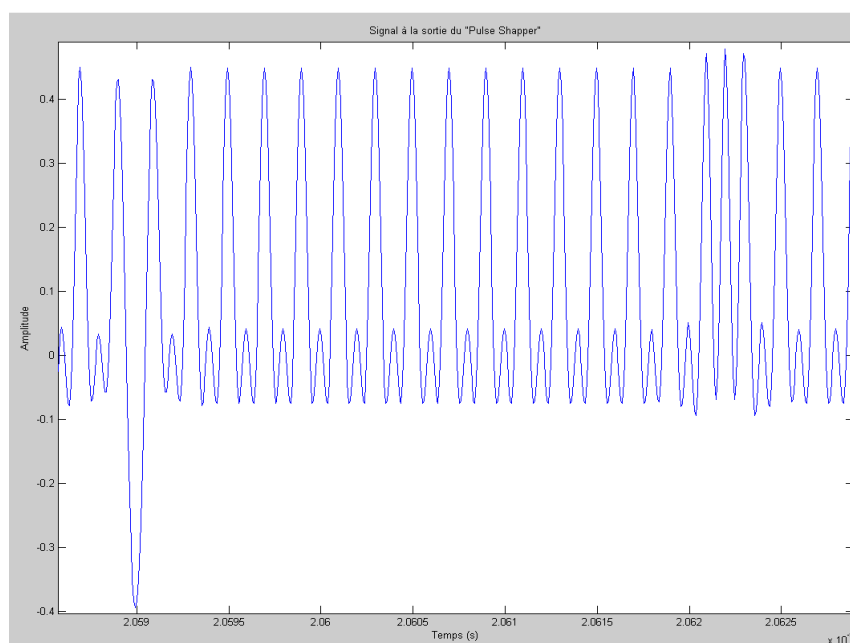


Figura 45. Ampliación de la trama 802.15.4a con cada doblete en un solo chip

El espectro de esta trama a la salida del « formador de onda » se presenta en la figura 46. En este caso también es conveniente alisar el espectro, pues también aparecen en él los picos parásitos. Observamos también que el ancho de banda ha sido doblado con respecto a los de las simulaciones precedentes. Esto se debe a la reducción del ancho de los impulsos.

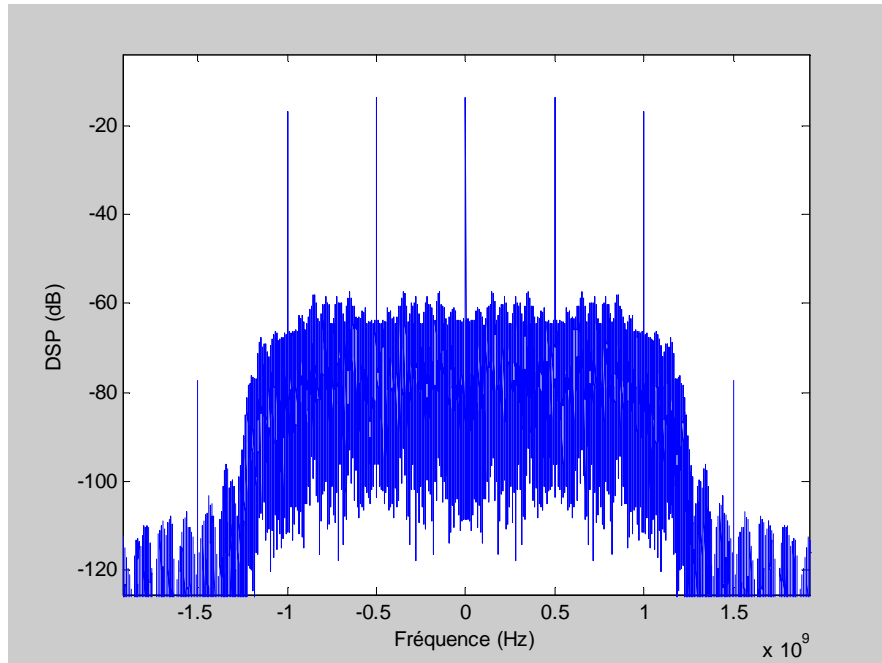


Figura 46. Espectro de la trama con cada doblete en un solo chip

Conclusión

En este capítulo hemos adaptado la trama del estándar IEEE 802.15.4a con el fin de hacerlo compatible con nuestro sistema TDSC. Esto ha sido posible insertando un impulso de referencia antes de cada impulso de datos. Esta inserción puede hacerse en dos chips diferentes, pero esto conduce al alargamiento de la trama y por tanto a la reducción de la velocidad útil a la mitad. La otra posibilidad era insertar este impulso de referencia en un solo chip de la trama de la norma pero con un impulso más estrecho (de anchura 1 ns en lugar de 2 ns). De esta manera, la longitud de la trama no se modifica y la velocidad se mantiene igual pues es transmitido el mismo número de datos durante la misma duración temporal. Esta última solución es la que ha sido retenida con el objetivo de ser estudiada de cara a su implementación en el banco experimental para los trabajos posteriores.

Conclusión general

Este proyecto de fin de estudios concernía la realización de un receptor para señales TR-UWB (Transmitted Reference UWB) cuya base es un chip CMOS, que permite la detección de estas señales con la ayuda de la técnica TDSC (Time Delayed Sampling & Correlation), desarrollada en el seno del equipo « COMMunications de PROximité » (COMunicaciones de PROximidad) de Télécom SudParis. La señal TR-UWB utilizada se compone de « dobletes » o parejas de impulsos: un impulso de referencia precede siempre al impulso de datos (positivo o negativo), con un retardo temporal T_D entre ambos. Con el fin de detectar el dato transmitido, la correlación se calcula entre estos dos impulsos y permite determinar si el dato transmitido es un '1' o un '0'.

Este trabajo se componía de dos partes principales: la primera se centraba en la concepción y la realización del receptor UWB completo alrededor del chip TDSC, y la segunda en la simulación de una trama UWB propuesta por el estándar 802.15.4a y en su adaptación a nuestro receptor, pues este estándar no propone el envío de los datos utilizando el formato de dobletes de impulsos.

Después de haber citado diferentes tipos de receptores UWB-LDR se ha presentado el método de detección TDSC. A continuación, se ha propuesto y probado un receptor que incluía el chip TDSC, dos conversores analógico/digitales y un microcontrolador. El chip TDSC permite muestrear las dos formas de onda analógicas y después las muestras son digitalizadas por los conversores. Los datos son guardados en la memoria interna del microcontrolador que gestiona todas las señales de control. Estos datos son transferidos al ordenador, utilizando el puerto serie RS-232, en el que el programa Matlab controla todo el sistema.

Con el objeto de testar esta técnica de detección, se ha concebido un emisor que genera las señales TR-UWB. Este permite generar un doblete de impulsos gracias a la utilización de un microcontrolador asociado al circuito de hiperfrecuencia, conteniendo además un diodo SRD y una línea de retardo cortocircuitada. La señal a la salida está formada por dobletes de impulsos TR-UWB. El retardo T_D , que es exactamente el mismo que el del receptor, es programado por medio del microcontrolador.

Se ha efectuado un test conectando directamente el emisor al receptor. Con la ayuda del programa Matlab, se ha podido grabar un tren de impulsos que nos permite así validar la primera parte del proyecto.

La etapa siguiente trataba sobre la generación de una trama UWB de la norma 802.15.4a. En efecto, el método de detección TDSC ha sido desarrollado para las redes inalámbricas a baja y media velocidad: localización, telemedicina, redes de sensores, etc... Nosotros hemos decidido simular este estándar en Matlab y hemos propuesto dos modificaciones posibles a aplicar a su trama con el objeto de hacerla compatible con nuestra técnica de detección. Esto ha sido posible insertando impulsos suplementarios en la trama con

el fin de incluir en ella los dobletes y, por tanto, haciéndola detectable por nuestro receptor. Una de las soluciones interesantes a retener era la inserción del impulso de referencia justo antes del impulso de datos, lo que nos llevaba a reducir la anchura del impulso impuesto por la norma de 2 ns a 1 ns, lo que no genera problemas al nivel del emisor, pero hace que aparezcan picos parásitos en el dominio espectral. Esto se debe a la añadidura de los impulsos de referencia, siempre idénticos a los impulsos de datos a los que acompañan.

Para la continuación de este trabajo, se debe proponer un algoritmo de sincronización con el objeto de establecer una comunicación entre el emisor y el receptor, y debe ser implementado independientemente del ordenador, algo que a nuestro conocimiento no es realizado en la actualidad por ningún sistema TR-UWB, y sin embargo es recomendado para este tipo de comunicaciones. Según lo que hemos podido constatar, será necesario reemplazar el microcontrolador del receptor por un componente con mejores prestaciones como un DsPic, que no es más que un microcontrolador que incluye un procesador de tratamiento de señal, siendo aún de bajo consumo y coste reducido. Los cálculos de correlación y de sincronización serían realizados por este componente, evitándonos así el uso de un ordenador.

Otra perspectiva concerniría la determinación de secuencias de código utilizadas en la trama 802.15.4a y que estarían compuestas por dobletes (+1+1 o -1+1). Con el objeto de reducir la amplitud de los picos parásitos que aparecen en el dominio espectral, los impulsos de referencia a insertar pueden ser aleatoriamente iguales a « +1 » o « -1 », pero esto impondría estudiar un nuevo tipo de modulación. Otros tipos de modulaciones pueden ser también considerados, entre los que podemos citar, por ejemplo, la modulación DTR (Differential Transmitted Reference).

Referencias bibliográficas

- [1] An Energy-Efficient OOK Transceiver for Wireless Sensor Networks. Denis C. Daly, Student Member, IEEE, and Anantha P. Chandrakasan, Fellow, IEEE. IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 42, NO. 5, MAY 2007.
- [2] Pseudocoherent Detection of OOK/PPM Signals as Zero-Delay Transmitted-Reference Signals With Bandpass Downsampling for UWB Communications. Shan Cui, Student Member, IEEE, Yong Liang Guan, Member, IEEE, IEEE, Kah Chan Teh, Senior Member, IEEE, and Kwok Hung Li, Senior Member, IEEE. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 58, NO. 8, OCTOBER 2009.
- [3] Charbel Saber, Roger Lamberti, Muriel Muller, Yang Ni, Francisco Hirata-Flores, "Optimal correlator size for the TDSC-UWB detection method," The International Conference on Mobile Technology, Applications and Systems, Singapore, 10-12 September 2007.
- [4] Francisco Hirata-Flores, M. Muller, Y. Ni and Claude Gimenes, "CMOS Implementation of a TR-UWB Receiver – based on Time Delayed Sampling and Correlation Method", INT 2007
- [5] Francisco I. Hirata-Flores, Muriel Muller, Yang Ni, and Claude Gimenes, "CMOS Implementation of a TR-UWB Receiver Based on Time Delayed Sampling and Correlation Method," European Wireless 2008, Prague, Czech Republic, 22-25 June 2008.
- [6] M. Muller, Y. Ni, R. Lamberti, C. Saber, "A new signal detection method for TR-UWB - by time delayed Sampling & Correlation (TDSC)," WINSYS, August 2006, Setubal Portugal.
- [7] Hirata Flores Francisco Iwao, 'Design and Implementation of a CMOS Prototype for a TDSC-UWB Receiver Based on TR Detection Scheme'. Thèse de Doctorat. Novembre 2008. Telecom SudParis – Université Pierre et Marie Curie
- [8] IEEE Std 802.15.4a – 2007, "Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-rate Wireless Personal Area Networks (LR-WPANS)", 2007
- [9] Klaus Witrisal, Geert Leus, Gerard J. M. Janssen, Marco Pausini, Florian Troesch, Thomas Zasowski, and Jac Romme, "Noncoherent Ultra-Wideband Systems", IEEE Signal Processing Magazine, July 2009
- [10] Win, M.Z.; Scholtz, R.A.; "[Impulse radio: how it works](#)". [Communications Letters, IEEE](#) Volume: 2 , Issue: 2, pp. 36 - 38 , 1998
- [11] Performance of An Improved Autocorrelation Receiver for TR-UWB Systems. Jun Zheng, Lingyuan Li, Yan Fan, Zhengqiang Wang, Xuehui Xu.
- [12] R. Hocht and H. Tomlinson, "Delay-hopped transmitted-reference RF communications," IEEE Conf. Ultra Wideband Systems and Techno., pp. 265-269, May 2002.
- [13] Goeckel D. L., Qu Zhang, "Slightly frequency-shifted reference ultra-wideband (UWB) radio: TR-UWB without the delay element," IEEE Military Communications Conference, vol. 5, pp. 3029-3035, 17-20 Oct. 2005.
- [14] <http://standards.ieee.org/getieee802/802.15.html>
- [15] : Emeric Guéguen, « Etude et optimisation des techniques UWB haut débit multibandes OFDM ». Thèse de Doctorat, INSA de Rennes, 2009.
- [16] David Marchaland, « Architectures et circuits dédiés aux émetteurs ultra large bande bas débit ». Thèse de Doctorat, Université de Marne-La-Vallée, 2007.

[17] Samuel Dubouloz, « Développement d'architectures avancées pour communications ultra large bande (UWB) dans des applications bas débit ». Thèse de Doctorat, ENST, 2008.

[18] Matlab Communications Toolbox™ 4 User's Guide

Anexos

A.1. Trama en la capa MAC

A.1.1. Esquema general de la trama

Cada trama MAC está compuesta de:

- Un campo MHR (MAC Header), que incluye el control de trama, el número de secuencia, las informaciones sobre la dirección y las informaciones relacionadas con la seguridad.
- Un campo MAC Payload, de longitud variable, que contiene las informaciones específicas del tipo de trama. Las tramas de confirmación (Acknowledgement) no contienen esta parte.
- Un campo MFR (MAC Footer), que contiene un FCS (16-bit Frame Check Sequence).

El formato general de una trama en la capa MAC está representado en la tabla 2:

Octets: 2	1	0/2	0/2/8	0/2	0/2/8	0/5/6/10/ 14	variable	2
Frame Control	Sequence Number	Destination PAN Identifier	Destination Address	Source PAN Identifier	Source Address	Auxiliary Security Header	Frame Payload	FCS
		Addressing fields						
MHR							MAC Payload	MFR

Tabla 2. Formato general de una trama en la capa MAC

El estudio se ha centrado principalmente en las tramas de tipo “beacon” (baliza), utilizadas por un coordinador para enviar las balizas. Ya sea en la capa MAC del estándar IEEE 802.15.4 o del estándar IEEE 802.15.4a, esta trama tiene la misma estructura, ya que el estándar IEEE 802.15.4a para las transmisiones UWB está basado en el estándar IEEE 802.15.4, pero adaptado a las especificaciones UWB de las transmisiones inalámbricas.

Sub-campos del campo MHR de la trama « beacon »

Podemos ver con la ayuda de la tabla 2 que el campo MHR contiene los sub-campos: control de trama, número de secuencia, campos de direccionamiento (solamente los campos identificador de la “Source PAN Identifier” y “Source Address”) y cabecera de seguridad auxiliar.

Sub-campos del campo MAC Payload de la trama « beacon »

La tabla 3 nos muestra que el campo MAC Payload contiene los sub-campos: especificación de la supertrama, campos GTS (Guaranteed Time Slot), campos de direcciones pendientes y datos o carga del beacon.

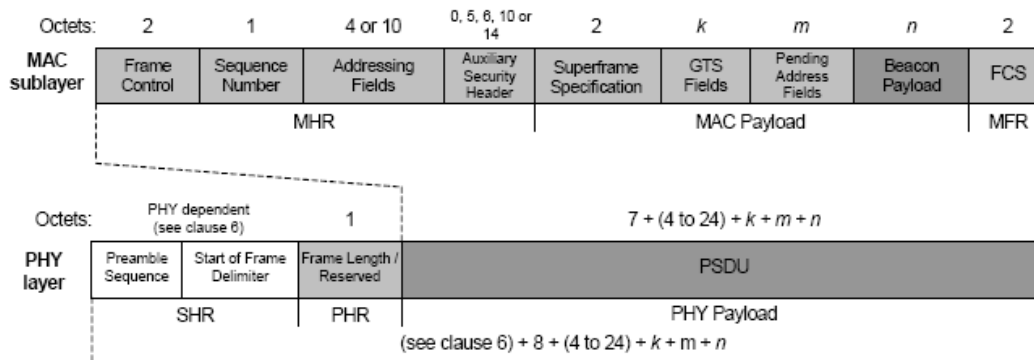


Tabla 3. Estructura de la trama beacon (baliza)

A.1.2. Número de octetos de una trama “beacon” en la capa MAC

El número de octetos de una trama beacon en la capa MAC es presentado con la ayuda de la tabla 4:

- Frame Control : 2 octetos
- Sequence Number : 1 octeto
- Addressing Fields : 4 o 10 octetos
 - Source PAN Identifier field : 2 octetos
 - Source Address field : 2 o 8 octetos
- Auxiliary Security Header : 0, 5, 6, 10 o 14 octetos
 - Security Control field : 1 octeto
 - Frame Counter field : 4 octetos
 - Key Identifier field : 0, 1, 5 o 9 octetos
- Superframe Specification : 2 octetos
- GTS Fields : k octetos (entre 1 y 23 octetos)
- Pending Address Fields : m octetos (entre 1 y 57 octetos)
- Beacon Payload : n octetos (entre 0 y 52 octetos)
- FCS : 2 octetos

	Frame Control	Sequence Number	Addressing Fields	Auxiliary Security Header	Super Frame Spec.	GTS Fields	Pending Address Fields	Beacon Payload	FCS	TOTAL
Min	2	1	4	0	2	1	1	0	2	13
Max	2	1	10	14	2	23	57	52	2	163

Tabla 4. Número de octetos de una trama beacon en la capa MAC

A.2. Trama de la capa PHY IEEE 802.15.4a (UWB)

A.2.1. Especificación PHY UWB

Se ha elegido una combinación de las modulaciones BPM (Burst Position Modulation) y BPSK (Binary Phase-Shift Keying), permitiendo utilizar a la vez los receptores coherentes y los no coherentes con un esquema de señalización común. La combinación BPM-BPSK es utilizada para modular los símbolos, los cuales se componen de una ráfaga de impulsos UWB. Son utilizadas diversas velocidades de datos, de ahí las ráfagas de longitud variable.

A.2.2. Formato de la trama PHY UWB

La trama UWB se compone de tres elementos principales: el SHR (Synchronization Header), el PHR (PHY Header) y el PSDU (PHY Service Data Unit), como se indica en la tabla 5:

		Bits	Octets
		19	variable
Preamble	SFD	PHR (see 6.8a.7)	PSDU
SHR		PHR	PHY payload

Tabla 5. Composición de una trama UWB

El preámbulo SHR es enviado siempre con una velocidad de base asociada a un código preámbulo cuyos parámetros están definidos en la tabla 6:

Bands	Preamble						
Channel Number	C_i Code Length	Peak PRF (MHz)	Mean PRF (MHz)	Delta Length δ_L	#Chips Per Symbol	Symbol Duration T_{psym} (ns)	Base Rate Msymbol/s
{0:15}	31	31.20	16.10	16	496	993.59	1.01
{0:3, 5:6, 8:10, 12:14}	31	7.80	4.03	64	1984	3974.36	0.25
{0:15}	127	124.80	62.89	4	508	1017.63	0.98

Tabla 6. Parámetros del preámbulo SHR

Cada periférico compatible con UWB debe poder aceptar los códigos de preámbulo de longitud 31 y dos velocidades de base correspondientes a las dos PRF (Pulse Repetition Frequencies) obligatorias para esta longitud de código. Las velocidades de base obligatorias para el preámbulo SHR son, en consecuencia, 1.01 Msímbolos/s y 0.25 Msímbolos/s.

El PHR es enviado a una velocidad nominal de 850 Kb/s para todas las velocidades de datos a partir de 850 Kb/s y a una velocidad nominal de 110 Kb/s para la velocidad nominal de 110 Kb/s. El PSDU es enviado a una velocidad definida por la tabla 7:

Channel Number	Peak PRF MHz	Bandwidth MHz	Preamble Code Length	Modulation & Coding			Data Symbol Structure						Data		
				Viterbi Rate	RS Rate	Overall FEC Rate	#Burst Positions per Symbol N_{burst}	#Hop Bursts N_{hop}	#Chips Per Burst N_{qb}	#Chips Per Symbol	Burst Duration T_{burst} (ns)	Symbol Duration T_{sym} (ns)	Symbol Rate (MHz)	Bit Rate Mb/s	Mean PRF (MHz)
{0,3,5,6,8,10,12,14}	499.2	499.2	31	0.5	0.87	0.44	32	8	128	4096	256.41	8205.13	0.12	0.11	15.60
	499.2	499.2	31	0.5	0.87	0.44	32	8	16	512	32.05	1025.64	0.98	0.85	15.60
	499.2	499.2	31	0.5	0.87	0.44	32	8	2	64	4.01	128.21	7.80	6.81	15.60
	499.2	499.2	31	1	0.87	0.87	32	8	1	32	2.00	64.10	15.60	27.24	15.60
{0,3,5,6,8,10,12,14}	499.2	499.2	31	0.5	0.87	0.44	128	32	32	4096	64.10	8205.13	0.12	0.11	3.90
	499.2	499.2	31	0.5	0.87	0.44	128	32	4	512	8.01	1025.64	0.98	0.85	3.90
	499.2	499.2	31	0.5	0.87	0.44	128	32	2	256	4.01	512.82	1.95	1.70	3.90
	499.2	499.2	31	1	0.87	0.87	128	32	1	128	2.00	256.41	3.90	6.81	3.90
{0,3,5,6,8,10,12,14}	499.2	499.2	127	0.5	0.87	0.44	8	2	512	4096	1025.64	8205.13	0.12	0.11	62.40
	499.2	499.2	127	0.5	0.87	0.44	8	2	64	512	128.21	1025.64	0.98	0.85	62.40
	499.2	499.2	127	0.5	0.87	0.44	8	2	8	64	16.03	128.21	7.80	6.81	62.40
	499.2	499.2	127	0.5	0.87	0.44	8	2	2	16	4.01	32.05	31.20	27.24	62.40
{4,11}	499.2	1331.2	31	0.5	0.87	0.44	32	8	128	4096	256.41	8205.13	0.12	0.11	15.60
	499.2	1331.2	31	0.5	0.87	0.44	32	8	16	512	32.05	1025.64	0.98	0.85	15.60
	499.2	1331.2	31	0.5	0.87	0.44	32	8	2	64	4.01	128.21	7.80	6.81	15.60
	499.2	1331.2	31	1	0.87	0.87	32	8	1	32	2.00	64.10	15.60	27.24	15.60
{4,11}	499.2	1331.2	127	0.5	0.87	0.44	8	2	512	4096	1025.64	8205.13	0.12	0.11	62.40
	499.2	1331.2	127	0.5	0.87	0.44	8	2	64	512	128.21	1025.64	0.98	0.85	62.40
	499.2	1331.2	127	0.5	0.87	0.44	8	2	8	64	16.03	128.21	7.80	6.81	62.40
	499.2	1331.2	127	0.5	0.87	0.44	8	2	2	16	4.01	32.05	31.20	27.24	62.40
7	499.2	1081.6	31	0.5	0.87	0.44	32	8	128	4096	256.41	8205.13	0.12	0.11	15.60
	499.2	1081.6	31	0.5	0.87	0.44	32	8	16	512	32.05	1025.64	0.98	0.85	15.60
	499.2	1081.6	31	0.5	0.87	0.44	32	8	2	64	4.01	128.21	7.80	6.81	15.60
	499.2	1081.6	31	1	0.87	0.87	32	8	1	32	2.00	64.10	15.60	27.24	15.60
7	499.2	1081.6	127	0.5	0.87	0.44	8	2	512	4096	1025.64	8205.13	0.12	0.11	62.40
	499.2	1081.6	127	0.5	0.87	0.44	8	2	64	512	128.21	1025.64	0.98	0.85	62.40
	499.2	1081.6	127	0.5	0.87	0.44	8	2	8	64	16.03	128.21	7.80	6.81	62.40
	499.2	1081.6	127	0.5	0.87	0.44	8	2	2	16	4.01	32.05	31.20	27.24	62.40
15	499.2	1354.97	31	0.5	0.87	0.44	32	8	128	4096	256.41	8205.13	0.12	0.11	15.60
	499.2	1354.97	31	0.5	0.87	0.44	32	8	16	512	32.05	1025.64	0.98	0.85	15.60
	499.2	1354.97	31	0.5	0.87	0.44	32	8	2	64	4.01	128.21	7.80	6.81	15.60
	499.2	1354.97	31	1	0.87	0.87	32	8	1	32	2.00	64.10	15.60	27.24	15.60
15	499.2	1354.97	127	0.5	0.87	0.44	8	2	512	4096	1025.64	8205.13	0.12	0.11	62.40
	499.2	1354.97	127	0.5	0.87	0.44	8	2	64	512	128.21	1025.64	0.98	0.85	62.40
	499.2	1354.97	127	0.5	0.87	0.44	8	2	8	64	16.03	128.21	7.80	6.81	62.40
	499.2	1354.97	127	0.5	0.87	0.44	8	2	2	16	4.01	32.05	31.20	27.24	62.40

Tabla 7. Velocidades del PSDU

Proceso de codificación de un PPDU (PHY Protocol Data Unit)

El proceso de codificación de un PPDU es ilustrado en la figura 47:

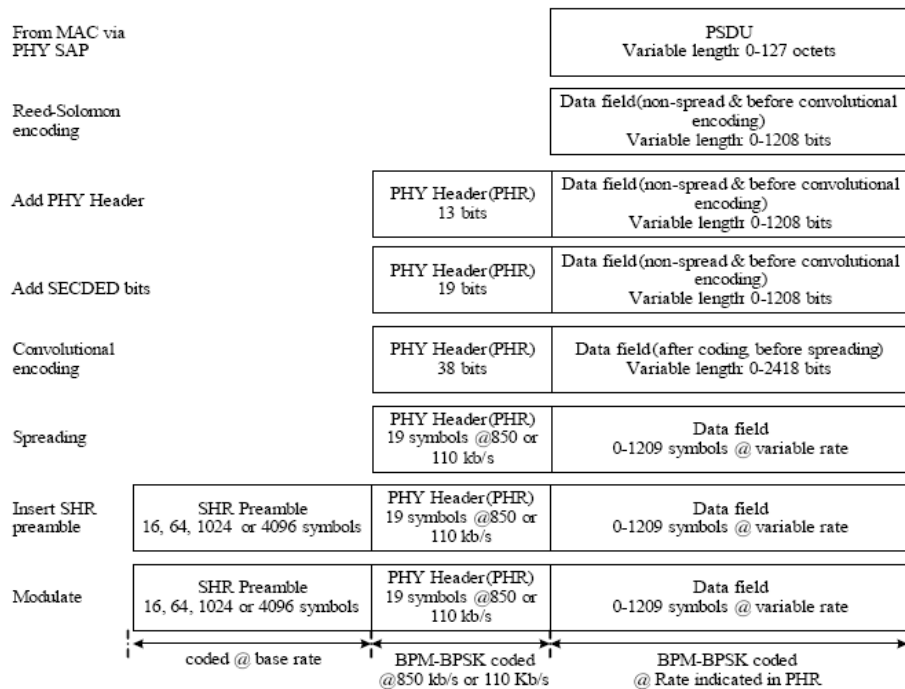


Figura 47. Proceso de codificación de un PPDU

Las etapas de codificación son:

- Efectuar una codificación Reed-Solomon sobre el PSDU.
- Generar el PHR.
- Añadir los bits de control SECDEC al PHR y colocarlo al comienzo del PSDU.
- Efectuar más códigos convolucionales. En ciertos casos, para una velocidad de datos de 27 Mb/s, la codificación convolucional del campo de datos es eludida y se codifican dos bits de datos para cada símbolo BPM-BPSK.
- Modular y extender el PSDU. El PHR es modulado utilizando BPM-BPSK, ya sea a 850 Kb/s o a 110 Kb/s, y el campo de datos es modulado a la velocidad especificada en el PHR.
- Generar el campo preámbulo SHR a partir del campo SYNC (utilizado para la convergencia AGC, la selección de la diversidad, la adquisición del tiempo y la adquisición de frecuencias secundarias) y del campo SFD (Start-of-Frame Delimiter), utilizado para indicar el comienzo de trama.

El flujo de señal en la capa PHY es mostrado en la figura 48:

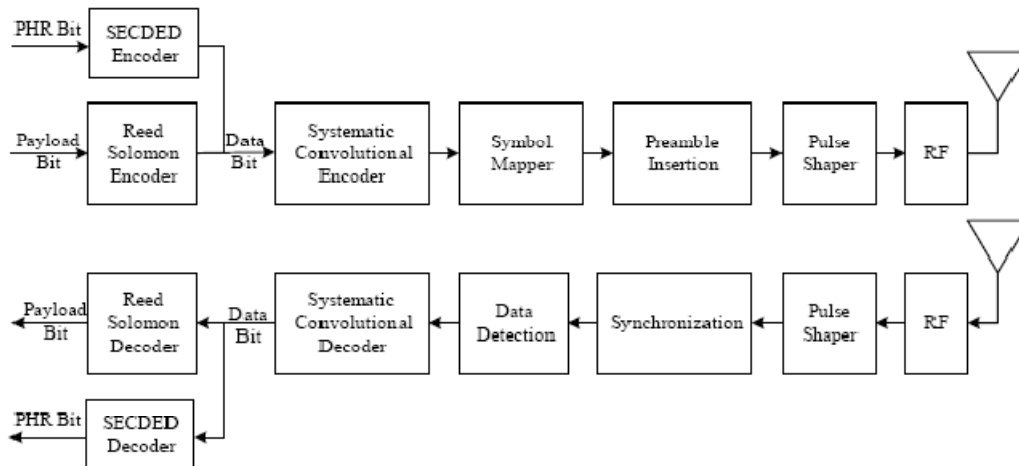


Figura 48. Flujo de señal en la capa física (PHY)

A.2.3. Estructura del símbolo PHY UWB

En el esquema de modulación BPM-BPSK, un símbolo UWB-PHY es capaz de transportar dos bits de información: un bit es utilizado para determinar la posición de una ráfaga de impulsos, mientras que el otro bit suplementario es utilizado para modular la fase (polaridad) de esta misma ráfaga. Se transmite una sola ráfaga en cada intervalo de símbolo PHY UWB. La estructura de un símbolo PHY UWB es ilustrada con la ayuda de la figura 49.

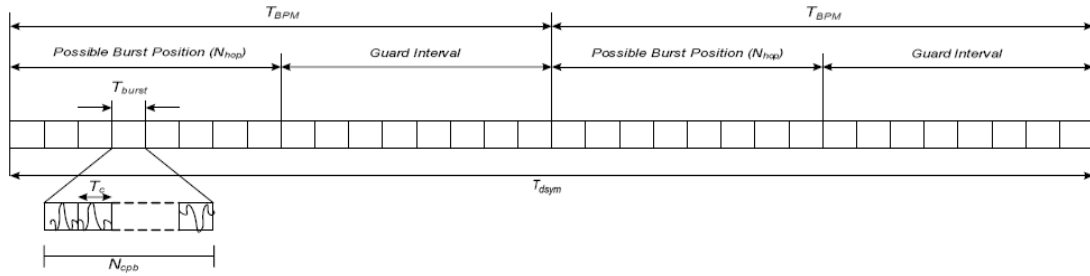


Figura 49. Estructura de un símbolo PHY UWB

N_C = número entero de posiciones de « chip » posibles de un símbolo.

T_C = duración de cada « chip » de un símbolo.

T_{dsym} = duración global de símbolo = $N_C \cdot T_C$.

Intervalo BPM = una de las dos partes en las cuales un símbolo es dividido para permitir la modulación BPM.

T_{BPM} = duración de un intervalo BPM = $T_{dsym}/2$.

N_{cpb} = número de chips consecutivos que forman una ráfaga.

T_{burst} = duración de una ráfaga = $N_{cpb} \cdot T_C$.

N_{burst} = número total de las duraciones de ráfaga por símbolo = T_{dsym}/T_{burst} .

El emplazamiento de la ráfaga, sea en la primera mitad o en la segunda mitad del símbolo, indica un bit de información. Además, la fase de la ráfaga (sea -1 o +1) es utilizado para indicar un segundo bit de información.

T_{burst} es más pequeño que T_{BPM} , esto permite rechazar la interferencia de acceso multiusuario utilizando TH (Time Hopping).

Con el fin de limitar las interferencias intersimbólicas causadas por los trayectos múltiples, solo la primera mitad de cada período T_{BPM} debe contener una ráfaga. Así, solo las primeras $N_{hop} = N_{burst}/4$ posiciones posibles de la ráfaga pueden contener una ráfaga en cada intervalo BPM.

A.2.4. Parámetros del PSDU

Existen dos longitudes posibles del código de preámbulo (31 o 127) y tres PRF medias posibles (15.6 MHz, 3.9 MHz et 62.4 MHz). Un dispositivo compatible debe aceptar la longitud 31 del código de preámbulo, así como las PRF medias de 15.6 MHz y 3.9 MHz para el PSDU. La utilización del código de longitud 127 es opcional, y cuando es aplicado, la PRF media del PSDU debe ser de 62.4 MHz.

Los canales UWB {4, 7, 11, 15} son canales opcionales y se diferencian de los otros canales UWB por una banda pasante más ancha (> 500 MHz) de las señales transmitidas.

A.2.5. Parámetros del preámbulo

Un símbolo de preámbulo se define como la forma de onda constituida por un código de preámbulo modulado (sea de longitud 31 o 127) y repetido. Para cada PRF, el preámbulo se construye a partir de un código de preámbulo C_i , insertando un cierto número de duraciones de « chip » entre los símbolos de código. El número de duraciones de « chip » a insertar es indicado por δ_L .

El preámbulo es enviado a una PRF media ligeramente superior a la de los datos. Esto se debe al hecho de que los códigos ternarios de longitud 31 o 127 son utilizados en el SHR y el número de « chips » en el SHR ya no es potencia de 2.

La velocidad de símbolo de base es definida como la velocidad a la cual son enviados los símbolos de preámbulo. Las velocidades de base que corresponden a las dos PRF medias obligatorias de 16.10 MHz y 4.03 MHz son de 1 Msímbolos/s y 0.25 Msímbolos/s, respectivamente. Estas corresponden a una duración de símbolo de preámbulo T_{psym} de 993.59 ns et 3974.36 ns para las dos PRF obligatorias.

Finalmente, para cada trama UWB compuesta por el SHR, el SFD, el PHR y un campo de datos, existen cuatro duraciones posibles del SHR. Estos se debe a las cuatro longitudes posibles del campo SYNC en el SHR. El campo SYNC se compone de repeticiones del símbolo de preámbulo. El número de repeticiones del símbolo de preámbulo puede ser 16, 64, 1024 o 4096. Estas diferentes longitudes del campo SYNC producen duraciones diferentes de la trama UWB.

N_C es función de la PRF utilizada en el canal y, por consiguiente, tiene dos o tres valores. Para cada valor de N_C , se definen las duraciones de símbolo de preámbulo admisibles T_{psym} , y la duración de la porción SYNC del SHR para cada longitud (16, 64, 1024 o 4096) es designada como T_{sync} . Tras la inserción del SFD (el SFD puede tener una longitud de 8 o 64 símbolos de preámbulo), la longitud total (en símbolos de preámbulo) del SHR puede ser uno de los N_{pre} valores indicados en la tabla 8:

Parameter	Description	Value		
<i>Channel</i>	UWB PHY Channel Number	{0:15}		
<i>PRF_{mean}</i>	Mean PRF (MHz)	16.10	62.89	4.03
<i>N_c</i>	Number of chips per preamble symbol	496	508	1984
<i>T_{psym}</i>	Preamble Symbol Duration (ns)	993.6	1017.6	3974.4
<i>N_{sync}</i>	Number of symbols in the packet sync sequence	Short	16	
		Default	64	
		Medium	1024	
		Long	4096	
<i>T_{sync}</i>	Duration of the packet sync sequence (μs)	Short	15.9	16.3
		Default	63.6	65.1
		Medium	1017.4	1042.1
		Long	4069.7	4168.2
<i>N_{sfd}</i>	Number of symbols in the SFD	8 or 64		
<i>T_{sfd}</i>	Duration of the frame sequence (μs)	7.9 or 63.6	8.1 or 65.1	31.8 or 254.4
<i>N_{pre}</i>	Number of symbols in the SHR Preamble	Short	24 or 80	
		Default	72 or 128	
		Medium	1032 or 1088	
		Long	4104 or 4160	
<i>T_{pre}</i>	Duration of the SHR Preamble (μs)	Short	23.8 or 79.5	24.4 or 270.6
		Default	71.5 or 127.2	73.3 or 319.5
		Medium	1025.4 or 1081.0	1050.2 or 1296.4
		Long	4077.7 or 4133.3	4176.3 or 4422.6
<i>N_{hdr}</i>	Number of symbols in the PHY Header	16		
<i>T_{hdr}</i>	Duration of the PHY Header field (μs)	16.4	16.8	65.6
<i>N_{data}</i>	Number of symbols in the data field	16 x LENGTH + 96		
<i>T_{data}</i>	Duration of the Data Field (μs)	$N_{data} \times T_{dsym}$		
<i>N_{CCA_PHR}</i>	Number of multiplexed preamble symbols in PHR	4 or 32		
<i>N_{CCA_data}</i>	Number of multiplexed preamble symbols in data field	$T_{data} / (4 \times T_{dsym}M)$		

Tabla 8. Longitud total (en símbolos de preámbulo) del SHR

Esto conduce a las posibles duraciones del SHR señaladas como T_{pre} . Tras la creación del SHR, la trama es añadida al PHR, cuya longitud, N_{hdr} , es de 16 símbolos y la duración es señalada como T_{hdr} .

A.2.6. Preámbulo SHR

El preámbulo puede ser dividido en dos partes diferentes como podemos ver en la figura 50: el SYNC (sincronización de los paquetes, estimación de canal y secuencia de alcance) y el SFD (secuencia delimitadora de trama).

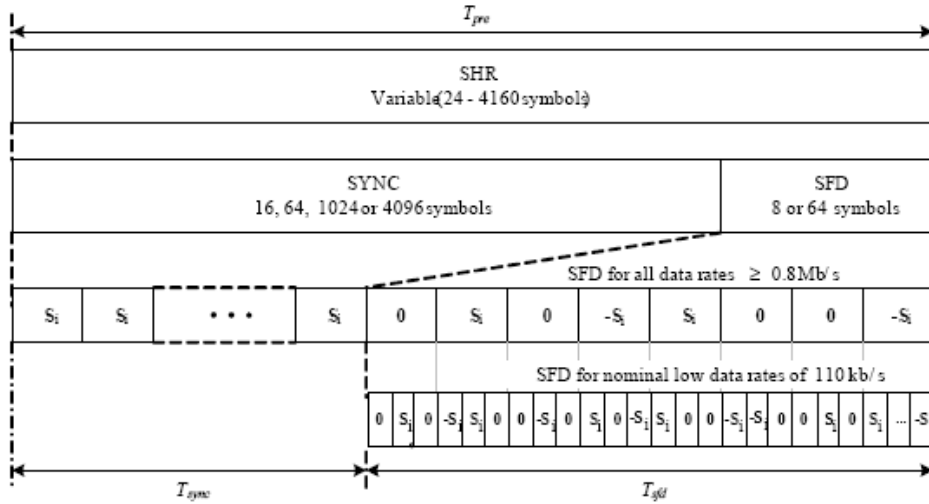


Figura 50. Composición del preámbulo SHR

A.2.6.1. Campo SYNC del SHR

Cada PAN funcionando sobre uno de los canales PHY UWB {0-15} está también identificada por un código de preámbulo. El código de preámbulo es utilizado para crear los símbolos que constituyen la parte SYNC del SHR. La capa PHY UWB sostiene dos longitudes de código de preámbulo: un código de longitud 31 y un código de longitud opcional 127. Cada código de preámbulo es una secuencia de símbolos de código elaborados a partir de un alfabeto ternario {-1, 0, 1} y seleccionados para el uso en PHY UWB debido a sus propiedades de autocorrelación periódica perfecta. Hay 8 códigos de longitud 31 y 16 de longitud 127. Los códigos que pueden utilizarse en cada uno de los canales PHY UWB están restringidos. Esta restricción de códigos es para asegurar que los códigos que tienen la más baja correlación cruzada son utilizados en el mismo canal PHY UWB. Además, 8 de los códigos de longitud 127 están reservados para una utilización solamente con el protocolo de alcance privado y no son utilizados durante el funcionamiento WPAN normal.

La atribución de los códigos de preámbulo a los canales ha sido realizada para permitir la comunicación intercanal. En otros términos, es posible que el funcionamiento de un periférico en un canal de banda ancha {4, 7, 11, 15} pueda comunicarse con otro periférico en un canal con el cual se superpone.

Para una WPAN utilizando el código ternario indexado por i , el campo SYNC está compuesto de N_{sync} repeticiones del símbolo S_i , donde S_i es el código C_i extendido por la función δ_L de longitud L . La operación de extensión, donde el código C_i es extendido a la duración de los símbolos de preámbulo indicada, está descrita matemáticamente por la ecuación siguiente:

$$S_i = C_i \otimes \delta_L(n)$$

$$\delta_L(n) = \begin{cases} 1 & n = 0 \\ 0 & n = 1, 2, \dots, L-1 \end{cases}$$

El operador \otimes indica un producto de Kronecker. Tras la operación de Kronecker, se forma un símbolo de preámbulo como en la figura 51, donde $L-1$ ceros han sido insertados entre cada elemento ternario de C_i , con N igual a 31 o 127.

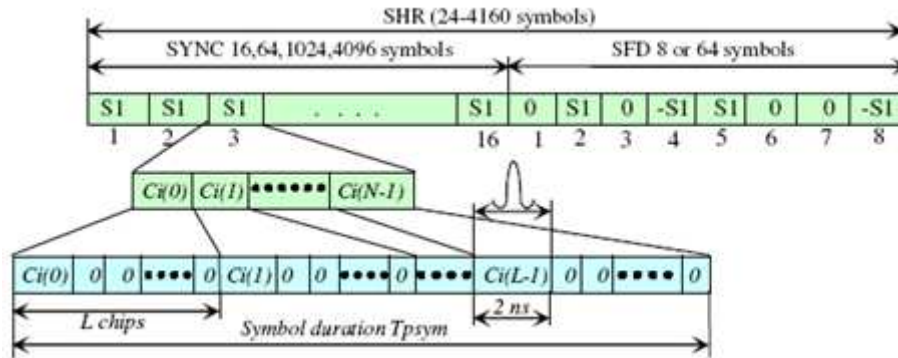


Figura 51. Formación del preámbulo SHR

A.2.6.2. Campo SFD del SHR

La capa PHY UWB sustenta un SFD corto obligatorio por defecto, y velocidades de datos medias y un SFD largo opcional para los datos a la velocidad nominal baja de 110 Kb/s. El SFD largo es ocho veces más largo que el SFD corto y está compuesto de 64 símbolos de preámbulo, de los cuales solamente 32 están activos y los otros 32 son ceros.

A.2.7. Cabecera PHY (PHR)

El PHR (tabla 9) debe ser añadido después del preámbulo SHR. El PHR se compone de 19 bits y contiene las informaciones necesarias para la correcta decodificación del paquete en el receptor. El PHR contiene información sobre la velocidad de datos utilizada para transmitir el PSDU, la duración del preámbulo de la trama en concreto y la longitud del « payload » de la trama. Además, se utilizan seis bits de control de paridad para proteger el PHR de los errores del canal.

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
R1	R0	L6	L5	L4	L3	L2	L1	L0	RNG	EXT	P1	P0	C5	C4	C3	C2	C1	C0	
Data Rate	Frame Length								Ranging Packet	Header Extension	Preamble Duration	SECDED Check Bits							

Tabla 9. Formato de la cabecera PHR

El PHR debe ser transmitido utilizando la modulación BPM-BPSK a una velocidad nominal de 850 Kb/s para todas las velocidades de datos por encima de 850 Kb/s y a una velocidad nominal de 110 Kb/s para los datos a la velocidad nominal baja de 110 Kb/s. El

campo de velocidad de datos se compone de dos bits (R1, R0) que indican la tasa de datos del PSDU recibido (tabla 10). Estos bits deben ser fijados dependiendo de la PRF media. El valor por defecto de estos bits debe ser fijado a “01” porque es la única tasa de datos obligatoria que es soportada por una implementación conforme a PHY UWB. Soportar otras velocidades de datos es opcional.

R1-R0	Mean PRF 15.60 or 62.40 MHz	Mean PRF 3.90 MHz
00	0.11	0.11
01	0.85	0.85
10	6.81	1.70
11	27.24	6.81

Tabla 10. Valores de los bits R1 y R0 del PHR

El campo de longitud de trama, L6-L0, debe ser un número entero sin signo de 7 bits que indique el número de octetos en el PSDU que la subcapa MAC solicita transmitir a la subcapa PHY. El bit de alcance del paquete, RNG, indica que la trama actual es una RFRAME si está definido a ‘1’, sino es definido a ‘0’. El bit de extensión de la cabecera, EXT, está reservado para la futura extensión del PHR y debe ser fijado a ‘0’. EL campo de duración del preámbulo (tabla 11), P1-P0, representa la longitud (en símbolos de preámbulo) de la parte SYNC del SHR. El ajuste por defecto de la duración del preámbulo es “01”, que corresponde a un campo SYNC con una longitud de 64 símbolos de preámbulo.

P1-P0	SYNC length (symbols) (Si)
00	16
01	64
10	1024
11	4096

Tabla 11. Valores de los bits P1 y P0 del PHR

El campo de duración del preámbulo está destinado a ser utilizado durante las operaciones de alcance y es utilizado por un receptor de la trama PHY para ayudar a determinar en qué símbolo de preámbulo la PHY UWB adquirió y comenzó el seguimiento del preámbulo. Un receptor puede utilizar el campo de duración del preámbulo para definir el valor de su duración de preámbulo basándose en el valor obtenido durante la comunicación de un paquete ACK.

El campo SECDEC (Single Error Correct, Double Error Detect), C5-C0, es un conjunto de bits de control de paridad que es utilizado para proteger el PHR de los errores ocasionados por el ruido y las degradaciones del canal. Los bits SECDEC son un simple código Hamming que permite la corrección de un solo error y la detección de dos errores al nivel del receptor. El valor de los bits SECDEC depende de los bits 0-12 del PHR.

A.3. Trama de la capa PHY IEEE 802.15.4 (ZigBee)

A.3.1. Especificaciones 868/915 MHz BPSK PHY

A.3.1.1. Tasa de datos

La tasa de datos de la banda 868/915 MHz BPSK PHY debe ser de 20 Kb/s durante las operaciones en la banda 868 MHz y 40 Kb/s durante las operaciones en la banda 915 MHz.

A.3.1.2. La modulación y el ensanchamiento

La banda 868/915 MHz BPSK PHY debe emplear directamente la modulación de “ensanchamiento de espectro por secuencia directa” (DSSS, Direct-Sequence Spread Spectrum) con una modulación de « chip » BPSK y una codificación diferencial utilizada para la codificación de los símbolos de datos.

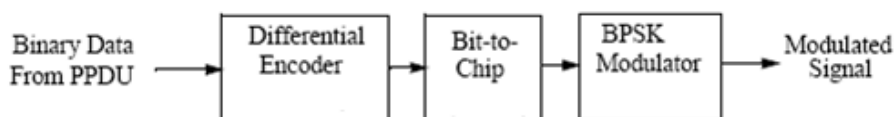


Figura 52. Modulación BPSK de los datos binarios

Codificación diferencial

La codificación diferencial es la suma en modulo 2 (XOR) de un bit de datos brutos con el bit codificado precedentemente. Esto es efectuado por el emisor y se puede describir como:

$$E_n = R_n \oplus E_{n-1}$$

donde R_n es el bit de datos brutos que esta siendo codificado, E_n es el bit correspondiente codificado diferencialmente y E_{n-1} es el bit codificado previamente.

Para cada paquete transmitido, E_1 es el primer bit de datos brutos que debe ser codificado y E_0 se supone igual a cero. Inversamente, el proceso de decodificación, tal y como se realiza al nivel del receptor, puede describirse como:

$$R_n = E_n \oplus E_{n-1}$$

Para cada paquete recibido, E_1 es el primer bit en ser decodificado y E_0 se supone igual a cero.

Integración bit-a-chip

Cada bit de entrada debe ser integrado en una secuencia PN (PseudoNoise) de 15 « chips » como se especifica en la tabla 12:

Input bits	Chip values ($c_0 c_1 \dots c_{14}$)
0	111101011001000
1	000010100110111

Tabla 12. Integración bit-a-chip

Modulación BPSK

Las secuencias de « chips » son moduladas sobre la portadora utilizando BPSK con formación de pulso de tipo coseno realzado (factor de roll-off = 1) donde un valor de « chip » de '1' corresponde a un impulso positivo y un valor de « chip » de '0' corresponde a un impulso negativo. La tasa de « chips » es de 300 Kchips/s en la banda 868 MHz y de 600 Kchips/s en la banda 915 MHz.

Forma del impulso

La forma del impulso de tipo coseno realzado (factor de roll-off = 1) utilizada para representar cada « chip » banda base es descrita por:

$$p(t) = \begin{cases} \frac{\sin \pi t / T_c \cos \pi t / T_c}{\pi t / T_c \sqrt{1 - 4t^2 / T_c^2}}, & t \neq 0 \\ 1, & t = 0 \end{cases}$$

A.3.2. Formato de trama

Como se muestra en la tabla 13, cada paquete PPDU se compone de los elementos de base siguientes:

- Una cabecera de sincronización (SHR), que permite a un receptor sincronizarse y bloquearse en el flujo de bits.
- Una cabecera PHY (PHR), que contiene la información de longitud de la trama.
- Un « payload » de longitud variable, que transporta la trama de la subcapa MAC.

		Octets		
		1	variable	
Preamble	SFD	Frame length (7 bits)	Reserved (1 bit)	PSDU
SHR		PHR		PHY payload

Tabla 13. Composición del paquete PPDU

A.3.2.1. Campo Preámbulo del SHR

El campo Preámbulo es utilizado por el emisor-receptor para obtener la sincronización de « chip » y de símbolo con un mensaje entrante. La longitud del preámbulo para las diferentes PHY es dada en la tabla 14:

PHY	Length		Duration (uS)
868-868.6 MHz BPSK	4 octets	32 symbols	1600
902-928 MHz BPSK	4 octets	32 symbols	800
868-868.6 MHz ASK	5 octets	2 symbols	160
902-928 MHz ASK	3.75 octets	6 symbols	120
868-868.6 MHz O-QPSK	4 octets	8 symbols	320
902-928 MHz O-QPSK	4 octets	8 symbols	128
2400-2483.5 MHz O-QPSK	4 octets	8 symbols	128

Tabla 14. Campo Preámbulo del SHR

Solo la modulación BPSK será utilizada en la continuación de este estudio.

A.3.2.2. Campo SFD del SHR

El SFD es un campo que indica el final del SHR y el comienzo del paquete de datos. La longitud del SFD para las diferentes PHY es dada en la tabla 15:

PHY	Length	
868-868.6 MHz BPSK	1 octet	8 symbols
902-928 MHz BPSK	1 octet	8 symbols
868-868.6 MHz ASK	2.5 octets	1 symbol
902-928 MHz ASK	0.625 octets	1 symbol
868-868.6 MHz O-QPSK	1 octet	2 symbols
902-928 MHz O-QPSK	1 octet	2 symbols
2400-2483.5 MHz O-QPSK	1 octet	2 symbols

Tabla 15. Campo SFD del SHR

El formato del SFD BPSK es el presentado en la tabla 16:

Bits: 0	1	2	3	4	5	6	7
1	1	1	0	0	1	0	1

Tabla 16. Formato del SFD BPSK

A.3.2.3. Campo de longitud de trama

El campo de longitud de trama tiene una longitud de 7 bits y especifica el numero total de octetos contenidos en el PSDU (PHY « payload »). Se trata de un valor comprendido entre 0 y *MaxPHYPacketSize* (127 octetos).

A.3.2.4. Campo PSDU

El campo PSDU tiene una longitud variable (entre 0 y 1016 bits) y contiene los datos del paquete PHY.

A.4. Gama de frecuencias

Las tablas 17 y 18 presentan la gama de frecuencias para la capa PHY de los estándares 802.15.4 y 802.15.4a, respectivamente.

PHY (MHz)	Frequency band (MHz)	Spreading parameters		Data parameters		
		Chip rate (kchip/s)	Modulation	Bit rate (kb/s)	Symbol rate (ksymbol/s)	Symbols
868/915	868–868.6	300	BPSK	20	20	Binary
	902–928	600	BPSK	40	40	Binary
868/915 (optional)	868–868.6	400	ASK	250	12.5	20-bit PSSS
	902–928	1600	ASK	250	50	5-bit PSSS
868/915 (optional)	868–868.6	400	O-QPSK	100	25	16-ary Orthogonal
	902–928	1000	O-QPSK	250	62.5	16-ary Orthogonal
2450	2400–2483.5	2000	O-QPSK	250	62.5	16-ary Orthogonal

Tabla 17. Gama de frecuencias de la capa PHY del estándar 802.15.4

Band group ^a (decimal)	Channel number (decimal)	Center frequency, f_c (MHz)	Band width (MHz)	Mandatory/Optional
0	0	499.2	499.2	Mandatory below 1 GHz
1	1	3494.4	499.2	Optional
	2	3993.6	499.2	Optional
	3	4492.8	499.2	Mandatory in low band
	4	3993.6	1331.2	Optional
2	5	6489.6	499.2	Optional
	6	6988.8	499.2	Optional
	7	6489.6	1081.6	Optional
	8	7488.0	499.2	Optional
	9	7987.2	499.2	Mandatory in high band
	10	8486.4	499.2	Optional
	11	7987.2	1331.2	Optional
	12	8985.6	499.2	Optional
	13	9484.8	499.2	Optional
	14	9984.0	499.2	Optional
	15	9484.8	1354.97	Optional

Tabla 18. Gama de frecuencias de la capa PHY del estándar 802.15.4a

El grupo de banda 0 es el canal sub-gigahercio, el grupo de banda 1 posee los canales UWB de la banda baja y finalmente el grupo de banda 2 posee los canales de la banda alta.

A.5. Código de simulación Matlab

Código de la trama UWB sin dobletes

close all;

%TRAME UWB (sans doublets)

%On a dessiné une trame UWB (IEEE 802.15.4a) comme exemple. On a considéré dans cette

%exemple que le PPDU a passé par tous les pas d'encodage:

%1) Codage Reed-Solomon

%2) Ajouter l'en-tête PHR

%3) Codage convolutionnel

%4) Modulation BPM-BPSK et élargissement

%5) Produire le préambule SHR

%

%On a considéré le canal 3 (N=31, Mean PRF=15.60 MHz)

%Regarder Tables 39a, 39b, 39c dans les pages 67, 70 et 71 respectivement

%pour voir les paramètres de configuration utilisées

%Preamble SHR: SYNC (short) + SFD (short)

%Code preamble 6: [++00+00---+-0++-000+0+0-+0+0000] (regarder Table 39d dans la page 72 de la norme IEEE 802.15.4a)

%Short SFD: [0 +1 0 -1 +1 0 0 -1] (regarder 6.8a.6.2 dans la page 74 de la norme IEEE 802.15.4a)

%

%Nsync=16 symboles, Tsync=15.9 µs, Nsfd=8 symboles, Tsfd=7.9 µs, DeltaL=16,

%Nc=496 chips/symbole, Débit SHR=1.01 Msymbols/s, #Chips SHR=11904,

%Npre=24 symboles, Tpre=Tshr=23.8 µs,

Si=[1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) -1

zeros(1,15) -1 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 1

zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1

zeros(1,15) 0 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0

zeros(1,15) 0 zeros(1,15) 0 zeros(1,15)];

SYNC=[Si Si Si Si Si Si Si Si Si Si Si Si Si Si Si];

SFD=[zeros(1,496) Si zeros(1,496) -Si Si zeros(1,496) zeros(1,496) -Si];

SHR=[SYNC SFD];

%PHR (19 symboles) + Data field (5 symboles) (BPM-BPSK)

%Nc=512 chips/symbole, Débit PHR=0.85 Mb/s, #Chips PHR=9728, Tphr=19.48716 µs

%

%S0 avec symbole 1 (burst chip position 48) et S1 avec symbole 2 (burst

%chip position 368) (regarder Table I.2 dans la page 186 de la norme IEEE 802.15.4a)

S0=[zeros(1,16) zeros(1,16) zeros(1,16) +1 +1 -1 +1 -1 -1 +1 +1 -1 +1 +1 -1 -1 +1 +1 -1 zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16)];

S1=[zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) -1 -1 -1 +1 -1 +1 -1

-1 +1 -1 +1 -1 +1 -1 -1 +1 zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)

zeros(1,16) zeros(1,16)];

PHR=[S1 S0 S0 S0 S1 S1 S0 S1 S1 S1 S0 S1 S0 S1 S0 S0 S0 S1];

%Exemple avec 5 symboles dans le champ de données

%Nc=512 chips/symbole, Débit Données=0.85 Mb/s, Tdonnées=5.1282 µs, #Chips Données=2560

Data=[S1 S0 S1 S1 S0];

Trame=[SHR PHR Data]; % #Chips Trame=24192 chips; Ttrame=48.41536 µs

Trame1=upsample(Trame,2); % On introduit un '0' entre chaque deux échantillons pour pouvoir différencier deux '1' ou deux '-1' ensembles

%Pulse shapping

nsamp=12; % number of samples/symbol

```

% Définition du filtre (paramètres)
filtorder = 40; % Ordre du filtre
delay = 8; %filtorder/(nsamp*2); % Délai de group (# of input samples)
rolloff = 0.25; % Rolloff factor du filtre

% Créer un filtre "square root raised cosine"
rrcfilter = rcosine(1,nsamp,'fir/sqrt',rolloff,delay); %rcosine(Fd,Fs,type_flag,r,delay)
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %'fir/sqrt'=Square-root raised cosine FIR
                %rolloff=it's a real number in the range [0, 1]
                %delay=specifies the filter's group delay, measured in input
samples. It's a positive integer.
                % The actual group delay in the filter design is delay/Fd
seconds.
% Signal transmise
ytx = rcosflt(Trame1,1,nsamp,'filter',rrcfilter); %RCOSFLT(X, Fd, Fs, TYPE_FLAG, NUM)==>
Upsample and apply square root raised cosine filter
                %X=input sequence
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %TYPE_FLAG='filter' ==> a filter specified by the user
                %NUM=the name of the filter specified by the user

Tshr=23.8e-6;
Tphr=19.48716e-6;
Tdata=5.1282e-6;
Ttrame=48.41536e-6;%durée de la trame sans les doublets

puntos_ytx=580800; %20 points/chip
Fs=puntos_ytx/Ttrame; %10 GHz
Dtiempo1=Ttrame/puntos_ytx; %0.1 ns
tTrame1=0:Dtiempo1:Ttrame-Dtiempo1; %580800 points

figure;
plot(tTrame1,ytx);
title('Signal à la sortie du "Pulse Shapper"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre signal à la sortie du Pulse Shaper)
Ts=Dtiempo1;
len=length(ytx);
ssf=(-len/2:len/2-1)/(Ts*len);
fx=fft(ytx);
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Pulse Shaper');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

%RF block: mixer + filte passe bande
%Upconversion (mixer)
Fc=4.4928e9;
OL=cos(2*pi*Fc*tTrame1);

```

```

signal=ytx'.*OL;
figure;
plot(tTrame1,signal);
title('Signal à la sortie du "Mixer" du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%Filtre Passe Bande (Fc = 4.4928 GHz, BW Filtre=499.2 MHz)
n=4;
Wn=[4.2432e9,4.7424e9]/6e9;
[B,A]=butter(n,Wn);

y_antena=filter(B,A,signal);
figure;
plot(tTrame1,y_antena);
title('Signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre)
Ts=Dtiempo1;
len=length(y_antena);
ssf=(-len/2:len/2-1)/(Ts*len);
fx=fft(y_antena);
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

%h = spectrum.welch;
%figure;
%psd(h,y_antena,'Fs',Fs);

```

Código de la trama UWB con cada impulso del doblete en un « chip » diferente

close all;

%On a considéré que $T_d = 2$ ns et la séparation entre un doublet et le
%suivant est aussi 2 ns dans cet exemple

%TRAME UWB

%On a dessiné une trame UWB (IEEE 802.15.4a) comme exemple. On a considéré dans cette
%exemple que le PPDU a passé par tous les pas d'encodage:

%1) Codage Reed-Solomon

%2) Ajouter l'en-tête PHR

%3) Codage convolutionnel

%4) Modulation BPM-BPSK et élargissement

%5) Produire le préambule SHR

%

%On a considéré le canal 3 ($N=31$, Mean PRF=15.60 MHz)

%Regarder Tables 39a, 39b, 39c dans les pages 67, 70 et 71 respectivement

%pour voir les paramètres de configuration utilisées

%Preamble SHR: SYNC (short) + SFD (short)

%Code preamble 6: $[++00+00---+0+-000+0+0-+0+0000]$ (regarder Table 39d dans la page 72 de
la norme IEEE 802.15.4a)

%Short SFD: $[0 +1 0 -1 +1 0 0 -1]$ (regarder 6.8a.6.2 dans la page 74 de la norme IEEE 802.15.4a)

%

%Nsync=16 symboles, Tsync=15.9 μ s, Nsf=8 symboles, Tsfd=7.9 μ s, DeltaL=16,

%Nc=496 chips/symbole, Débit SHR=1.01 Msymbols/s, #Chips SHR=11904,

%Npre=24 symboles, Tpre=Tshr=23.8 μ s

Si=[1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) -1
zeros(1,15) -1 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 1
zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1
zeros(1,15) 0 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0
zeros(1,15) 0 zeros(1,15) 0 zeros(1,15)];

SYNC=[Si Si Si Si Si Si Si Si Si Si Si Si Si Si Si Si];

SFD=[zeros(1,496) Si zeros(1,496) -Si Si zeros(1,496) zeros(1,496) -Si];

SHR=[SYNC SFD];

%PHR (19 symboles) + Data field (5 symboles) (BPM-BPSK)

%Nc=512 chips/symbole, Débit PHR=0.85 Mb/s, #Chips PHR=9728, Tphr=19.48716 μ s

%

%S0 avec symbole 1 (burst chip position 48) et S1 avec symbole 2 (burst

%chip position 368) (regarder Table I.2 dans la page 186 de la norme IEEE 802.15.4a)

S0=[zeros(1,16) zeros(1,16) zeros(1,16) +1 +1 -1 +1 -1 -1 +1 +1 -1 +1 +1 -1 -1 +1 +1 -1 zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16)];

S1=[zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) -1 -1 -1 +1 -1 +1 -1
-1 +1 -1 +1 -1 +1 -1 -1 +1 zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16)];

PHR=[S1 S0 S0 S0 S1 S1 S0 S1 S1 S1 S0 S1 S0 S1 S0 S0 S0 S0 S1];

%Exemple avec 5 symboles dans le champ de données

%Nc=512 chips/symbole, Débit Données=0.85 Mb/s, Tdonnées=5.1282 μ s, #Chips Données=2560

Data=[S1 S0 S1 S1 S0];

Trame=[SHR PHR Data]; % #Chips Trame=24192 chips; Ttrame=48.41536 μ s

%On introduit les impulsions de référence après chaque chip de la trame

%pour avoir les doublets


```

Trame_mod=ones(1,2*length(Trame));
i=1;
Trame_mod(1)=Trame(1);
for j=3:2:length(Trame_mod)
    Trame_mod(j)=Trame(j-i);
    i=i+1;
end

Trame1=upsample(Trame_mod,2); % On introduit un '0' entre chaque deux échantillons pour
pouvoir différencier deux '1' ou deux '-1' ensembles sous Matlab

%Pulse shapping
nsamp=12; % number of samples/symbol

% Définition du filtre (paramètres)
filtorder = 40; % Ordre du filtre
delay = 8; %filtorder/(nsamp*2); % Délai de group (# of input samples)
rolloff = 0.25; % Rolloff factor du filtre

% Créer un filtre "square root raised cosine"
rrcfilter = rcosine(1,nsamp,'fir/sqrt',rolloff,delay); %rcosine(Fd,Fs,type_flag,r,delay)
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %'fir/sqrt'=Square-root raised cosine FIR
                %rolloff=it's a real number in the range [0, 1]
                %delay=specifies the filter's group delay, measured in input
samples. It's a positive integer.
                % The actual group delay in the filter design is delay/Fd
seconds.
% Signal transmise
ytx = rcosflt(Trame1,1,nsamp,'filter',rrcfilter); %RCOSFLT(X, Fd, Fs, TYPE_FLAG, NUM)==>
Upsample and apply square root raised cosine filter
                %X=input sequence
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %TYPE_FLAG='filter' ==> a filter specified by the user
                %NUM=the name of the filter specified by the user

Tshr=23.8e-6;
Tphr=19.48716e-6;
Tdata=5.1282e-6;
Ttrame=96.83072e-6;%durée de la trame est le double parce qu'on ajoute les doublets

puntos_ytx=1161408; %48 points/chip
Fs=puntos_ytx/Ttrame; %12 GHz
Dtiempo1=Ttrame/puntos_ytx; %83.37 ps
tTrame1=0:Dtiempo1:Ttrame-Dtiempo1; %1161408 points

figure;
plot(tTrame1,ytx);
title('Signal à la sortie du "Pulse Shapper"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre signal à la sortie du Pulse Shaper)
Ts=Dtiempo1;
len=length(ytx);
ssf=(-len/2:len/2-1)/(Ts*len);

```

```

fx=fft(ytx);
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Pulse Shaper');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

%RF block: mixer + filtre passe bande
%Upconversion (mixer)
Fc=4.4928e9;
OL=cos(2*pi*Fc*tTrame1);
signal=ytx'.*OL;
figure;
plot(tTrame1,signal);
title('Signal à la sortie du "Mixer" du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%Filtre Passe Bande (Fc = 4.4928 GHz, BW Filtre=499.2 MHz)
n=4;
Wn=[4.2432e9,4.7424e9]/6e9;
[B,A]=butter(n,Wn);

y_antena=filter(B,A,signal);
figure;
plot(tTrame1,y_antena);
title('Signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre)
Ts=Dtiempo1;
len=length(y_antena);
ssf=(-len/2:len/2-1)/(Ts*len);
fx=fft(y_antena);
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

% Nouveau spectre;
% time=tTrame1;
% sample=y_antena;
% nechant=size(sample,2);
% fe=nechant/time(nechant);
% nbpoint=floor(nechant/2);
% f=fe/nechant*(1:nbpoint);
% spe=abs(fft(sample))/nechant*2;
% figure;
% plot(f,spe(1:nbpoint))

```

Código de la trama UWB con cada doblete en un solo « chip »

close all;

%On a considéré que dans chaque chip (2 ns) il y a un doublet et le temps de séparation
%entre doublets

%TRAME UWB

%On a dessiné une trame UWB (IEEE 802.15.4a) comme exemple. On a considéré dans cette
%exemple que le PPDU a passé par tous les pas d'encodage:

%1) Codage Reed-Solomon

%2) Ajouter l'en-tête PHR

%3) Codage convolutionnel

%4) Modulation BPM-BPSK et élargissement

%5) Produire le préambule SHR

%

%On a considéré le canal 3 (N=31, Mean PRF=15.60 MHz)

%Regarder Tables 39a, 39b, 39c dans les pages 67, 70 et 71 respectivement

%pour voir les paramètres de configuration utilisées

%Preamble SHR: SYNC (short) + SFD (short)

%Code preamble 6: [++00+00---+0+++000+0+0+0+0000] (regarder Table 39d dans la page 72 de
la norme IEEE 802.15.4a)

%Short SFD: [0 +1 0 -1 +1 0 0 -1] (regarder 6.8a.6.2 dans la page 74 de la norme IEEE 802.15.4a)

%

%Nsync=16 symboles, Tsync=15.9 µs, Nsf=8 symboles, Tsfd=7.9 µs, DeltaL=16,

%Nc=496 chips/symbole, Débit SHR=1.01 Msymbols/s, #Chips SHR=11904,

%Npre=24 symboles, Tpre=Tshr=23.8 µs

Si=[1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) -1
zeros(1,15) -1 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 1
zeros(1,15) -1 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1
zeros(1,15) 0 zeros(1,15) -1 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 1 zeros(1,15) 0 zeros(1,15) 0
zeros(1,15) 0 zeros(1,15) 0 zeros(1,15)];

SYNC=[Si Si Si Si Si Si Si Si Si Si Si Si Si Si Si Si];

SFD=[zeros(1,496) Si zeros(1,496) -Si Si zeros(1,496) zeros(1,496) -Si];

SHR=[SYNC SFD];

%PHR (19 symboles) + Data field (5 symboles) (BPM-BPSK)

%Nc=512 chips/symbole, Débit PHR=0.85 Mb/s, #Chips PHR=9728, Tphr=19.48716 µs

%

%S0 avec symbole 1 (burst chip position 48) et S1 avec symbole 2 (burst

%chip position 368) (regarder Table I.2 dans la page 186 de la norme IEEE 802.15.4a)

S0=[zeros(1,16) zeros(1,16) zeros(1,16) +1 +1 -1 +1 -1 -1 +1 +1 -1 +1 +1 -1 -1 +1 +1 -1 zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16)];

S1=[zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) -1 -1 -1 +1 -1 +1 -1
-1 +1 -1 +1 -1 +1 -1 -1 +1 zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16) zeros(1,16)
zeros(1,16) zeros(1,16)];

PHR=[S1 S0 S0 S0 S1 S1 S0 S1 S1 S1 S0 S1 S0 S1 S0 S0 S0 S0 S1];

%Exemple avec 5 symboles dans le champ de données

%Nc=512 chips/symbole, Débit Données=0.85 Mb/s, Tdonnées=5.1282 µs, #Chips Données=2560

Data=[S1 S0 S1 S1 S0];

Trame=[SHR PHR Data]; %Chips Trame=24192 chips; Ttrame=48.41536 µs

%On introduit les impulsions de référence après chaque chip de la trame

%pour avoir les doublets

Trame_mod=ones(1,2*length(Trame));

```

i=1;
Trame_mod(1)=Trame(1);
for j=3:2:length(Trame_mod)
    Trame_mod(j)=Trame(j-i);
    Trame_mod(j+1)=1;
    i=i+1;
end

Trame1=upsample(Trame_mod,2); % On introduit un '0' entre chaque deux échantillons pour
pouvoir différencier deux '1' ou deux '-1' ensembles sous Matlab

%Pulse shapping
nsamp=6; % number of samples/symbol

% Définition du filtre (paramètres)
filtorder = 40; % Ordre du filtre
delay = 8; %filtorder/(nsamp*2); % Délai de group (# of input samples)
rolloff = 0.25; % Rolloff factor du filtre

% Créer un filtre "square root raised cosine"
rrcfilter = rcosine(1,nsamp,'fir/sqrt',rolloff,delay); %rcosine(Fd,Fs,type_flag,r,delay)
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %'fir/sqrt'=Square-root raised cosine FIR
                %rolloff=it's a real number in the range [0, 1]
                %delay=specifies the filter's group delay, measured in input
samples. It's a positive integer.
                % The actual group delay in the filter design is delay/Fd
seconds.
% Signal transmise
ytx = rcosflt(Trame1,1,nsamp,'filter',rrcfilter); %RCOSFLT(X, Fd, Fs, TYPE_FLAG, NUM)==>
Upsample and apply square root raised cosine filter
                %X=input sequence
                %Fd=sampling frequency of the input
                %Fs=sampling frequency of the output
                % The ratio Fs/Fd must be a positive integer greater than 1
                %TYPE_FLAG='filter' ==> a filter specified by the user
                %NUM=the name of the filter specified by the user

Tshr=23.8e-6;
Tphr=19.48716e-6;
Tdata=5.1282e-6;
Ttrame=48.41536e-6;%durée de la trame avec un doublet dans chaque chip

puntos_ytx=580704; %60 points/chip
Fs=puntos_ytx/Ttrame; %30 GHz
Dtiempo1=Ttrame/puntos_ytx; %33.35 ps
tTrame1=0:Dtiempo1:Ttrame-Dtiempo1; %1451760 points

figure;
plot(tTrame1,ytx);
title('Signal à la sortie du "Pulse Shapper"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre signal à la sortie du Pulse Shaper)
Ts=Dtiempo1;
len=length(ytx);
ssf=(-len/2:len/2-1)/(Ts*len);

```

```

fx=fft(ytx);
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Pulse Shaper');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

%RF block: mixer + filtre passe bande
%Upconversion (mixer)
Fc=4.4928e9;
OL=cos(2*pi*Fc*tTrame1);
signal=ytx'.*OL;
figure;
plot(tTrame1,signal);
title('Signal à la sortie du "Mixer" du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%Filtre Passe Bande (Fc = 4.4928 GHz, BW Filtre=499.2 MHz)
n=4;
Wn=[4.2432e9,4.7424e9]/6e9;
[B,A]=butter(n,Wn);

y_antena=filter(B,A,signal);
figure;
plot(tTrame1,y_antena);
title('Signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Temps (s)');
ylabel('Amplitude');

%DFT (représentation spectre)
Ts=Dtiempo1;
len=length(y_antena);
ssf=(-len/2:len/2-1)/(Ts*len);
fx=fft(y_antena(1:len));
fxs=fftshift(fx);
figure;
plot(ssf,abs(fxs));
title('Spectre du signal à la sortie du Filtre Passe Bande du "RF Block"');
xlabel('Fréquence (Hz)');
ylabel('Amplitude');

%h = spectrum.welch;
%figure;
%psd(h,y_antena,'Fs',Fs);

```