



## **Protocolo de Comunicación para AMR**

Memoria del Proyecto Fin de Carrera  
de Ingeniería Informática realizado  
por Víctor Ramírez Cubero y dirigido  
por Jordi Carrabina Bordoll y  
Guillermo Talavera Velilla

Bellaterra, 1 de Febrero.de 2011

*A mi padre*



Los abajo firmantes, Jordi Carrabina Bordoll, profesor del Departament de Microelectrònica i Sistemes Electrònics, y Guillermo Talavera Velilla de la Universitat Autònoma de Barcelona

## Certifican

Que el trabajo correspondiente a la presente memoria ha sido realizado bajo nuestra dirección por Víctor Ramírez Cubero. Y para que así conste firmamos

Jordi Carrabina Bordoll

Guillermo Talavera Velilla

Bellaterra, 1 de Febrero de 2011



## AGRADECIMIENTOS

Quisiera agradecer a Jordi Carrabina el haber apostado por este proyecto y haberme proporcionado las herramientas necesarias para su realización, al igual que a Guillermo Talavera por haberme proporcionado su ayuda cuando se la he pedido.

A mis padres por todo el apoyo mostrado a lo largo de todos estos años de carrera, y sobre todo este último año. Sin ellos no estaría donde estoy ahora.

Agradecerle a mi abuela, a mi hermana y a mi novia su constante apoyo cuando no me veían muy animado y buscaban diferentes maneras para seguir adelante.

Por último darle las gracias a Aroa porque sin ella no podría haber realizado todas las gestiones del proyecto, al encontrarme fuera de Barcelona. Y en general a todos mis compañeros y amigos que han estado interesados en mi proyecto.

# Índex

Capítulo 1. Introducción y análisis.....	- 1 -
1.1. AMR .....	- 1 -
1.2. Introducción al Protocolo MBus y Wireless MBus .....	- 2 -
1.2.1. Comunicaciones Básicas.....	- 4 -
1.2.2. Componentes de una Red MBus.....	- 4 -
1.2.3. Cuestiones Básicas de los Sistemas Bus y Modelo OSI.....	- 5 -
1.3. Estudio de mercado y justificación de la elección .....	- 8 -
1.3.1. Estudio de mercado.....	- 8 -
1.3.2. Justificación de la elección .....	- 9 -
1.4. Motivación y objetivos del proyecto .....	- 10 -
Capítulo 2. Estado del arte de Wireless MBus.....	- 11 -
2.1. Estado del arte.....	- 11 -
2.2. Tipos de comunicación .....	- 12 -
2.2.1. Modo S.....	- 14 -
2.2.2. Modo T .....	- 19 -
2.2.3. Modo R2 .....	- 20 -
Capítulo 3. Arquitectura del Sistema .....	- 22 -
3.1. Arquitectura del sistema .....	- 22 -
3.1.1. ¿Qué es MSP430?.....	- 22 -
3.1.2. Placas Experimentales .....	- 24 -
3.1.3. Stack de comunicaciones .....	- 28 -
3.2. Ámbito Sistema.....	- 28 -
3.3. Ámbito Cliente.....	- 31 -
3.4. Metodología de trabajo .....	- 32 -
Capítulo 4. Implementación del protocolo .....	- 33 -

4.1. Modo de implementación .....	- 33 -
4.1.1. Modo transmisión .....	- 34 -
4.1.2. Modo recepción .....	- 35 -
Capítulo 5. Test y Resultados .....	- 38 -
5.1. Entorno de test .....	- 38 -
5.2. Resultados a nivel de sistema .....	- 39 -
5.3. Validación a nivel de sistema .....	- 41 -
Capítulo 6. Conclusiones.....	- 42 -
6.1. Conclusiones.....	- 42 -
6.2. Experiencia personal y profesional.....	- 42 -
6.3. Evolución futura .....	- 43 -
Referencias y Bibliografía .....	- 45 -
Anexos .....	- 48 -
Anexo A.....	- 48 -
Anexo B .....	- 49 -
Anexo C .....	- 51 -
Anexo D.....	- 52 -
Anexo E .....	- 54 -





# Capítulo 1.

## Introducción y análisis

### 1.1. AMR

AMR (Automatic Meter Reading), en español, lectura automática de contadores, es una tecnología que se encarga de la recolección automática de consumo, diagnóstico y estado de los diferentes contadores que tenemos en nuestros domicilios y del envío de estos datos a las sedes centrales para la creación de las facturas, informes de errores y análisis del sistema. Entre las diferentes ventajas que tienen los AMR cabe destacar el ahorro que constituye para las empresas a la hora de recopilar lecturas y la de lecturas en tiempo real, dejando las estimaciones entre lecturas anteriores y estimaciones posteriores. Gracias a esto proporcionamos a los usuarios y proveedores un mejor control del consumo en los hogares. [WIKIAMR]

Existen diferentes tipos de tecnologías para realizar este tipo de lecturas, a continuación se listarán algunas de ellas:

- Radio frecuencia: Es una de las más utilizadas y obtiene los resultados gracias a un sistema de doble sentido. Esta tecnología realiza a través de dispositivos móviles la captura de los datos.
- Red fija: Esta tecnología está permanentemente conectada a una red para la lectura de los datos. A través de antenas, repetidores, concentradores u otro tipo de infraestructuras que permitan el envío de datos, se obtienen los datos en un ordenador central.

En 1972 Theodore George Paraskevakos creó el primer sistema de monitoreo de sensores que utilizan transmisión digital por seguridad. En 1974 este sistema fue patentado y en 1977 se fabricaron los primeros lectores comerciales.

## 1.2. Introducción al Protocolo MBus y Wireless MBus

MBus es un estándar europeo desarrollado para satisfacer la necesidad de crear un sistema de redes y lectura de contadores de consumo, por ejemplo, los contadores de agua, luz y gas que tenemos en nuestras casas [MBUS][MBUS2]. Este tipo de tipología Bus cumple los requisitos necesarios para obtener un bajo consumo. Cuando los contadores reciben una petición estos envían datos a un sistema master, el cual puede ser otro contador o una simple PDA.

Tradicionalmente la lectura de los contadores se ha realizado de manera manual por los proveedores y la implementación de MBus ayuda con eficacia y rapidez a la recopilación de datos. A su vez Mbus y Wireless Mbus se pueden llegar a utilizar para el resto de instalaciones que tengamos en nuestros hogares, control iluminación, sensor de temperatura, sistema de apertura de ventanas.

Aquí tenemos algunas de las características que este protocolo puede traer a las lecturas de AMR:

- Los datos se leen de manera electrónica
- A través de un solo cable, o una interconexión Wifi, podemos conectar todos los contadores de una casa o edificio
- Cada contador tiene una dirección individual
- Posibilidad de lectura remota, por ejemplo, servicios centrales de los proveedores.

Tras esto podemos enumerar algunas de las ventajas que proporciona tanto a los usuarios como a las empresas:

- Lecturas rápidas y sin errores,
- Datos estructurados para una rápida interpretación
- Lecturas a distancia por lo que se ahorra en personal, se evita entrar en el entorno privado del cliente y permite poner contadores en lugares de difícil acceso.
- Intervalos cortos de lectura, para una mejor tarificación y control del gasto.
- Gracias a intervalos breves de lectura, el desgaste de pilas y de la pieza es menor y se optimiza el uso de la red.

En la *Figural* podemos ver un esquema general del funcionamiento de una red MBus y sus ramas, como el caso que nos atañe, Wireless MBus

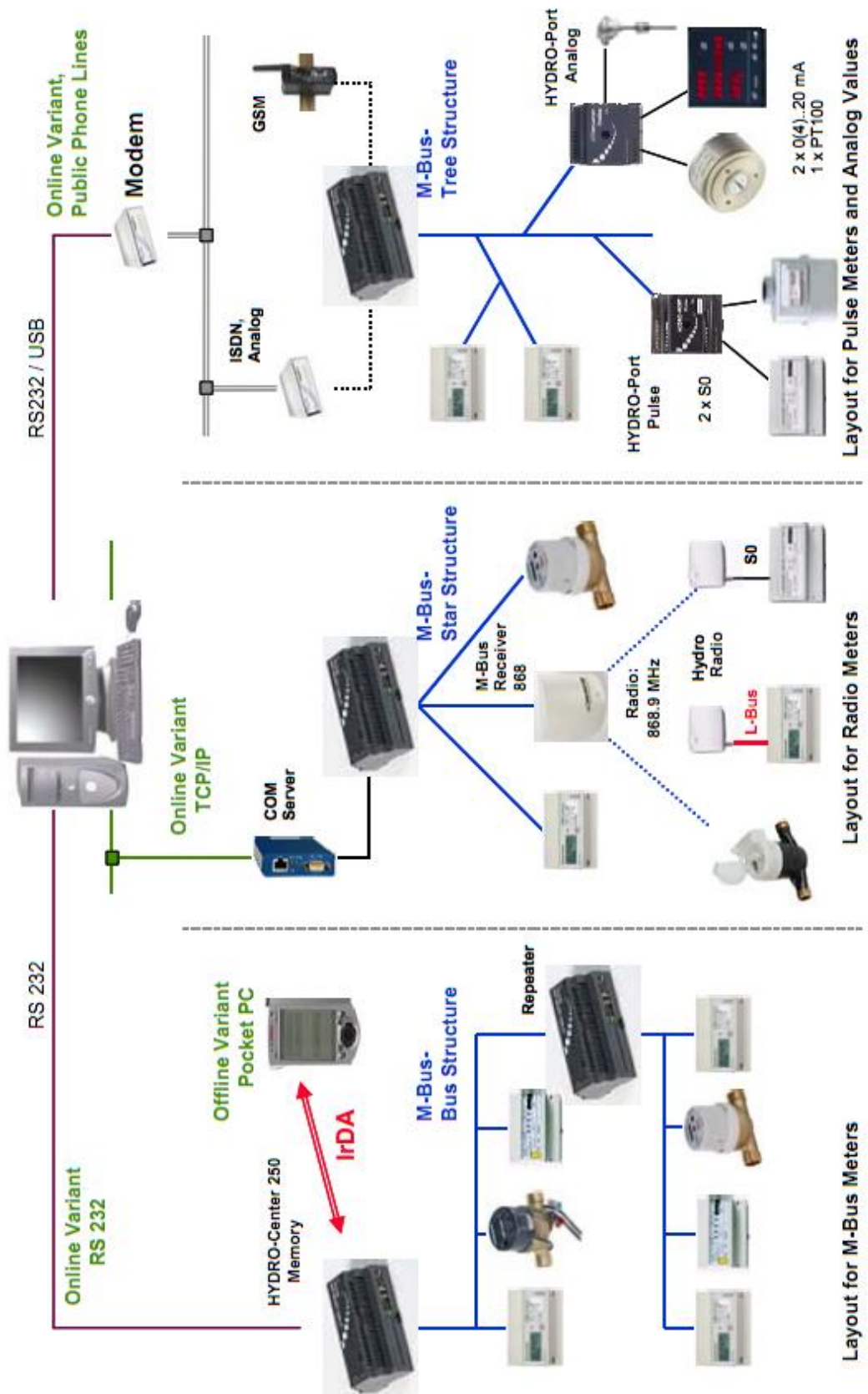


Figura 1. Funcionamiento Red Bus [GSWEB]

### **1.2.1. Comunicaciones Básicas**

MBus está optimizado para trabajar en ambientes descentralizados, ya que los valores se obtienen a través de un proceso maestro-esclavo.

El master de MBus inicializa las comunicaciones y asigna una dirección individual a los diferentes esclavos (de 1 a 250), en el caso de existir más terminales se utiliza una dirección especial 253.

En el caso de una comunicación normal el master reclama la información al terminal específico, este esclavo responde con la medida del ARM, la medida media, el tipo de dispositivo, el número de serie y el código de facturación.

Cuando tenemos más de 250 dispositivos, el master hace uso de la dirección especial 253, una especie de broadcast, por la cual todos los dispositivos responden con los datos estándar citados anteriormente.

### **1.2.2. Componentes de una Red MBus**

A continuación se enumerarán y se hará una pequeña descripción de los componentes que puede tener una red MBus:

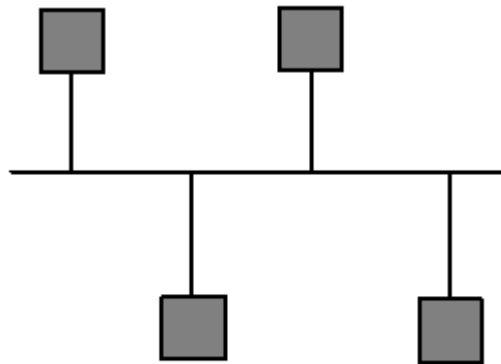
- Maestro
- Esclavo
- Conversores
- Repetidores
- Lectores Remotos
- Lectores Manuales
- Central de Facturación
- Sistemas de Radio Frecuencia
- Bluetooth

### 1.2.3. Cuestiones Básicas de los Sistemas Bus y Modelo OSI

#### a) Sistemas Bus

Para comprender un poco más la tecnología MBus y Wireless debemos analizar los sistemas Bus que nos rodean.

Una red con topología Bus es aquella cuyos componentes están conectados a una misma línea de transmisión, por lo que solo puede transmitir un equipo en cada instante. Este tipo de tipología es muy rentable y si uno de los componentes falla la red sigue funcionando sin problemas. La transmisión de los datos se puede hacer a todos los componentes conectados o a un grupo específico, tal y como podemos observar en la *Figura 2*.

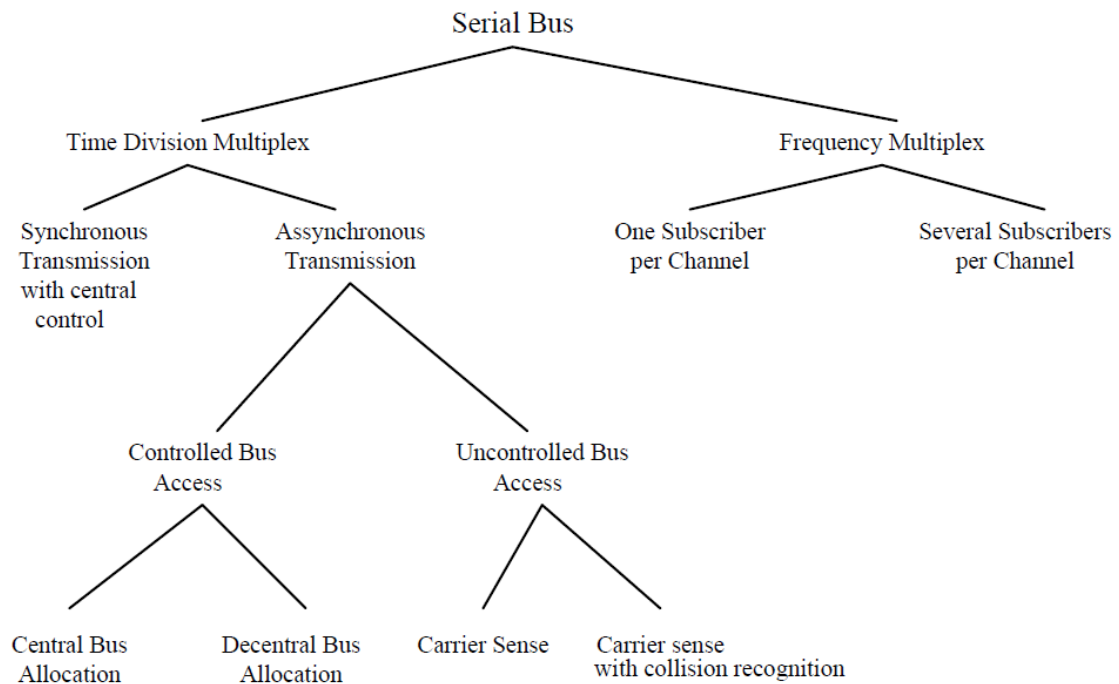


**Bus**

*Figura 2. Topología Bus*

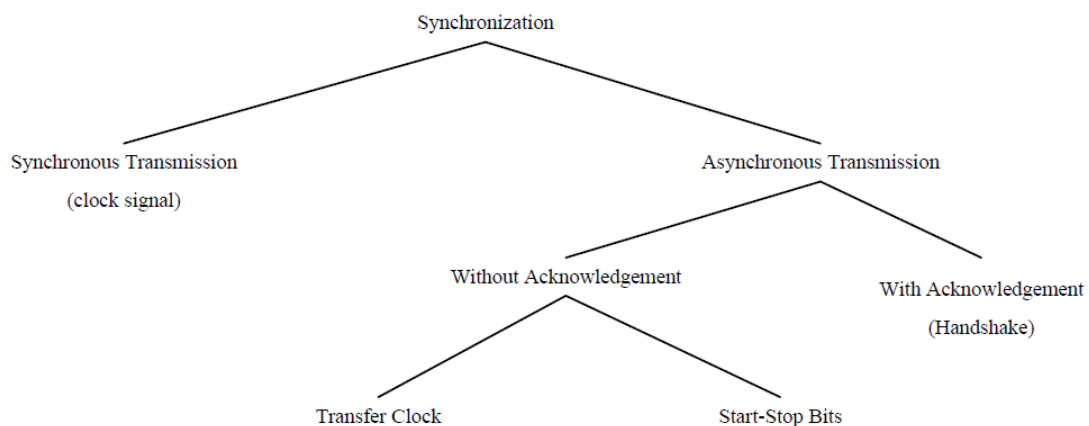
Un sistema Bus puede funcionar de diferentes maneras, una de ellas es la multiplexación por frecuencia la cual funciona a través de un rango de frecuencias. A través de estas frecuencias podemos subscribir uno o varios terminales por canal.

El funcionamiento más clásico es la multiplexación por división de tiempo, en el cual nos encontramos diferentes tipos de configuraciones. Hay que tener en cuenta que las estaciones no pueden transmitir simultáneamente ya que de esta manera evitamos posibles errores y colisiones. Por una parte tenemos un sistema centralizado, en el cual el sistema central recibe una petición de envío de datos y es él el que decide cuando el terminal puede enviar los datos por el canal. Y por otra parte tenemos los sistemas descentralizados, los cuales proveen a los terminales de varios métodos para que sean capaces de saber cuándo pueden utilizar el canal. En la *figura 3* podemos ver los diferentes usos de la multiplexación.



*Figura 3. Clasificación de los Sistemas Bus según el tipo de acceso*

Otro punto importante dentro de la multiplexación por división de tiempo es la sincronización de los terminales. Con las sincronizaciones síncronas los mensajes son enviados tras la una señal de reloj la cual es estable y sirve de medida para los tiempos de transmisión. Por otro lado con una sincronización asíncrona necesitamos métodos que implementen ACKs los cuales informan de que los datos han llegado correctamente al destino. En la siguiente *figura 4* podemos observar de manera esquemática los métodos de sincronización.



*Figura 4. Clasificación de Técnicas de sincronización*

Los sistemas Bus deben estar preparados para reconocer errores producidos en la transmisión y corregirlos. Los principales sistema de seguridad que pueden implementar los sistemas Bus son los bits de paridad, checksum. A la misma vez con los procesos de sincronización, los acks informan de si existe error o no. Si no se ha recibido ACK en un tiempo límite se considera error.

## b) Modelo OSI

El Modelo OSI (Open System Interconnection) es un modelo creado por el ISO (International Organization for Standardization) para la interconexión de equipos a través de la red.

Este modelo no es el más utilizado actualmente, ya que fue desbancado por el modelo TCP, aún así sigue sirviendo como referencia para diferentes protocolos, como MBus, y en la enseñanza.

Este modelo se divide en 7 capas diferentes:

7	Capa de aplicación	Capas orientadas a aplicación
6	Capa de presentación	
5	Capa de sesión	
4	Capa de transporte	Capas orientadas a transporte
3	Capa de red	
2	Capa de enlace de datos	
1	Capa física	

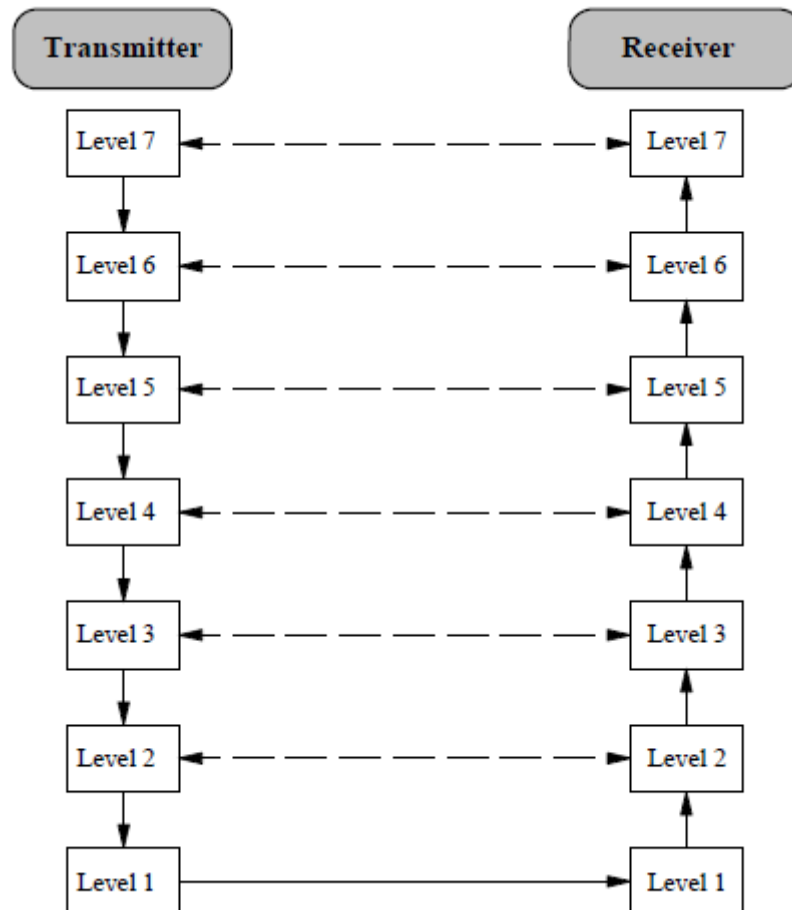
*Tabla 1. Capas modelo OSI*

1. Capa física: Es la encargada de la conexión física entre los diferentes los diferentes terminales
2. Capa de enlace de datos: Esta capa se encarga del direccionamiento físico, la topología de la red, distribución de tramas, control de flujo y acceso a la red
3. Capa de red: Es la encargada de hacer que los datos lleguen a su destino
4. Capa de transporte: Es la encargada de efectuar el transporte de datos del terminal de origen al de destino
5. Capa de sesión: Es la que se encarga de mantener y controlar el enlace establecido entre los dos terminales
6. Capa de presentación: Es la encargada de la representación de la información



7. Capa de aplicación: Ofrece a las aplicaciones la posibilidad de acceder a diferentes servicios y define los protocolos a utilizar.

A continuación, podemos observar en la *figura 5* cómo se realiza la comunicación entre las diferentes capas del modelo OSI



*Figura 5. Transmisión de datos según el modelo OSI*

## 1.3. Estudio de mercado y justificación de la elección

### 1.3.1. Estudio de mercado

En los últimos años se han desarrollado múltiples protocolos y tecnologías en el mundo de las telecomunicaciones y a partir de estas se han creado sistemas de control, gestión y monitorización de parámetros.

El objetivo de las empresas ha sido la creación de sistemas que permitan la interconectividad entre diferentes productos, en nuestro caso existen diversos fabricantes que disponen de contadores que implementan los estándares MBus y

Wireless MBus (Elster, EMS, Flonidan, L+G) los cuales deben ayudarse de los diferentes fabricantes de circuitos integrados, que proveen de microcontroladores orientados a entornos de ultra bajo consumo (ULP) y entornos Wireless (TI, Freescale, ST, Renesas, Nordic...) y que disponen de kits de diseño y notas de aplicación para la realización de prototipos.

Estos sistemas potencian los sistemas productivos ya que nos ofrecen un amplio abanico de posibilidades, monitorizar los diferentes contadores y en caso de error en los contadores poder efectuar diferentes gestiones telemáticas. A su vez se pueden actualizar rutas de trabajo, cambiar consumo contratado sin necesidad del cambio del contador y obtener datos en tiempo real, entre otras funciones. En cuanto a los datos en tiempo real pueden ayudar a la detección de problemas y por tanto ser más eficaces en cuanto a la previsión y planificación de problemas futuros.

Este tipo de tecnología tiene una gran implantación en Estados Unidos por la demografía que tiene, por ejemplo, los domicilios particulares pueden encontrarse en grandes explanadas y muy separadas unas de otras. A su vez en Europa la implementación de este tipo de tecnologías es menor, en comparación con Estados Unidos, por lo que la necesidad de un cambio en los modelos de telemetría en nuestro continente es necesario.

Como hemos podido observar existe una importante oportunidad de negocio en la implementación de protocolos y herramientas eficaces para el ahorro de costes y la mejora de los servicios. Son las empresas como Endesa, GasNatural y similares los clientes potenciales para que implementen este proyecto por ser una solución estandarizada.

### **1.3.2. Justificación de la elección**

La elección de esta tecnología se debe a su bajo coste, tanto en material como en consumo. La instalación de este tipo de tecnología está implementada en muchos hogares europeos por lo que su extensión puede ser rápida y eficaz.

Gracias a que MBus es un estándar, dispositivos de diferentes fabricantes pueden funcionar en el mismo Bus y los usuarios tienen libertad de elegir el fabricante. Por otro lado existe una alta estimulación de este mercado, incluso para otros tipos de contadores MBus, por lo que con diferentes configuraciones se pueden resolver problemas actualmente difíciles.

En el desarrollo del protocolo MBus se han tenido en cuenta los aspectos económicos y técnicos de la interfaz, ya que son importantes para el uso diario de estos. Los más relevantes son:

- Se pueden conectar un gran número de dispositivos
- Posibilidad de expansión de la red
- Robusto y a prueba de fallos
- Coste mínimo

- Mínimo consumo de energía en los medidores
- Aceptable velocidad de transmisión

De los muchos sistemas de Bus ya existente ninguno fue capaz de cumplir con todas estas limitaciones. Ahora, con MBus, como una nueva interfaz estándar para la lectura de contadores de consumo, se puede ofrecer un compromiso óptimo entre precio y rendimiento.

A su vez, el mercado actual de microcontroladores nos ofrece una cantidad enorme de productos los cuales tienen un consumo mínimo, siendo una excelente elección para este protocolo. Uno de los proveedores de microcontroladores es Texas Instruments, que a su vez fue participe en la creación de este protocolo, tiene un amplio catálogo que nos ayudará en el desarrollo del proyecto. El microcontrolador estrella que nos ofrece es el MSP430, cuyas características se estudiarán durante el desarrollo del proyecto.

## **1.4. Motivación y objetivos del proyecto**

El objetivo de este proyecto es la implementación del protocolo Wireless MBus para la comunicación entre diferentes tipos de sensores inalámbricos y aplicaciones de lectura remota. Con la implementación de protocolo se pretende estandarizar los métodos de comunicación haciéndolos más fiables y abiertos a los desarrolladores. En nuestro caso nos centraremos más en las lecturas de contadores de manera remota (AMR, Automatic Meter Reading)

Se busca que la implementación de este protocolo tenga un coste mínimo, sobre todo para las distribuidoras de servicios eléctricos, de agua o gas. Con este proyecto el coste de mantenimiento sería mínimo ya que con las baterías implantadas en un primer momento debe mantenerse en funcionamiento durante años.

La motivación principal de este proyecto es debida a la idea de simplificación de costes y hacer más cómoda la vida a los usuarios. La idea inicial del proyecto consistía en la creación de contadores automáticos, tras investigar sobre el tema se descartó esta idea ya que este producto ya se había inventado y funcionaba en diferentes partes del mundo.

Tras ver diferentes opciones, se observó que lo que había que mejorar era la comunicación entre los contadores, de aquí surgió el proyecto de implementar un protocolo MBus en este tipo de comunicación. Este proyecto está englobado en una idea mayor, el cual engloba la implementación de este tipo de tecnologías en las viviendas particulares y crear un sistema desde el cual se pueda controlar desde el consumo hasta la climatización del hogar.

La tecnología MBus nos da la posibilidad de hacer esto posible a un bajo coste y realizando el menor intrusismo posible al domicilio del particular. A demás podremos obtener información en tiempo real y la posibilidad de que el usuario pueda observar sus hábitos de consumo y mejorarlos.

## Capítulo 2.

### Estado del arte de Wireless MBus

#### 2.1. Estado del arte

El protocolo fue desarrollado por el equipo formado por Texas Instruments, TECHEM y la universidad de Paderborn (Alemania)

Cuando se desarrolló el protocolo MBus la idea inicial era usarlo en redes de topología Bus. Esta idea se desarrolla bien en entornos cerrados donde se puede controlar todo el sistema a través de un PLC, por ejemplo el control de una fábrica. A la hora de desarrollar este sistema para uso público se tuvo que hacer una evaluación de los costes necesarios para implementar el sistema.

A parte del coste necesario de infraestructura, había que tener en cuenta la privacidad y el respeto hacia la propiedad privada de los usuarios, por lo que se tuvo que buscar soluciones a este problema. Como evolución y solución natural el protocolo avanzó hacia un sistema Wireless, lo que implica menor cableado e intrusismo en la propiedad privada.

Gracias a la tecnología Wireless, el precio de la infraestructura es mucho menor y el consumo que tiene es ínfimo. Los fabricantes y proveedores se han centrado en este tipo de tecnología por lo que su desarrollo está en auge. Esta nueva concepción nos lleva a opción de transmitir los valores de los datos deseados de manera remota, eliminando la lectura in-situ actual.

Las soluciones de lectura remota tienen una penetración del 25% en EEUU mientras que en el resto de países es solo del 5%. En el caso de Europa los planes piloto se llevan a cabo en zonas pequeñas y no se aprecia realmente su impacto global. Una de las causas de la elevada implementación de estos sistemas en EE.UU es por la situación de los domicilios, ya que en muchas ocasiones se encuentran en medio de grandes explanadas y muy separadas unas de otras por lo que la lectura in-situ es muy costosa.

En el contexto internacional se tenía previsto que para el año 2010 el 25% de los contadores fueran de tipo AMR. En el caso de España se están realizando diferentes pruebas piloto en Cataluña, Andalucía y País Vasco. A su vez Endesa ha desarrollado el Programa Cervantes, el cual quiere realizar el cambio de 13 millones de contadores tradicionales por contadores AMR para 2015.

Nuestra visión con este proyecto va más allá de la lectura de contadores eléctricos a través de AMR, gracias a la tecnología Wireless MBus podremos gestionar todos los contadores y sensores de nuestro domicilio y controlar sus funciones y obtener sus datos a través de un dispositivo master, que puede ser otro contador. Con este

sistema la competitividad de las empresas suministradoras y los servicios que ofrecen pueden ser ampliados ya que trataremos de tener todos nuestros servicios en una única factura.

## 2.2. Tipos de comunicación

Como hemos comentado anteriormente Wireless MBus se basa en diferentes tipos de comunicaciones según el interés y la funcionalidad que se busque.

Principalmente se cuenta con 3 Modos de trabajo, los cuales pueden contener submodos. [STD1]

- Modo S “Estacionario”. Está pensado para comunicaciones uni o bidireccionales. Dispone de varios submodos (S1, Sm y S2) para la optimización de los envíos según el tipo de dispositivo a usar.
- Modo T “Transmisión Frecuente”. En este modo el contador transmite una trama muy corta cada pocos segundos. El submodo T1 es solo de transmisión y envía la mínima información. El submodo T2 es bidireccional.
- Modo R “Recepción Frecuente”. En este modo el dispositivo escucha cada pocos segundos a la espera de recibir un mensaje de activación. Cuando recibe dicho mensaje queda preparado durante unos segundos para la comunicación entre ambos dispositivos.

A continuación se mostrará en la *tabla 2* los diferentes submodos y su funcionamiento específico:

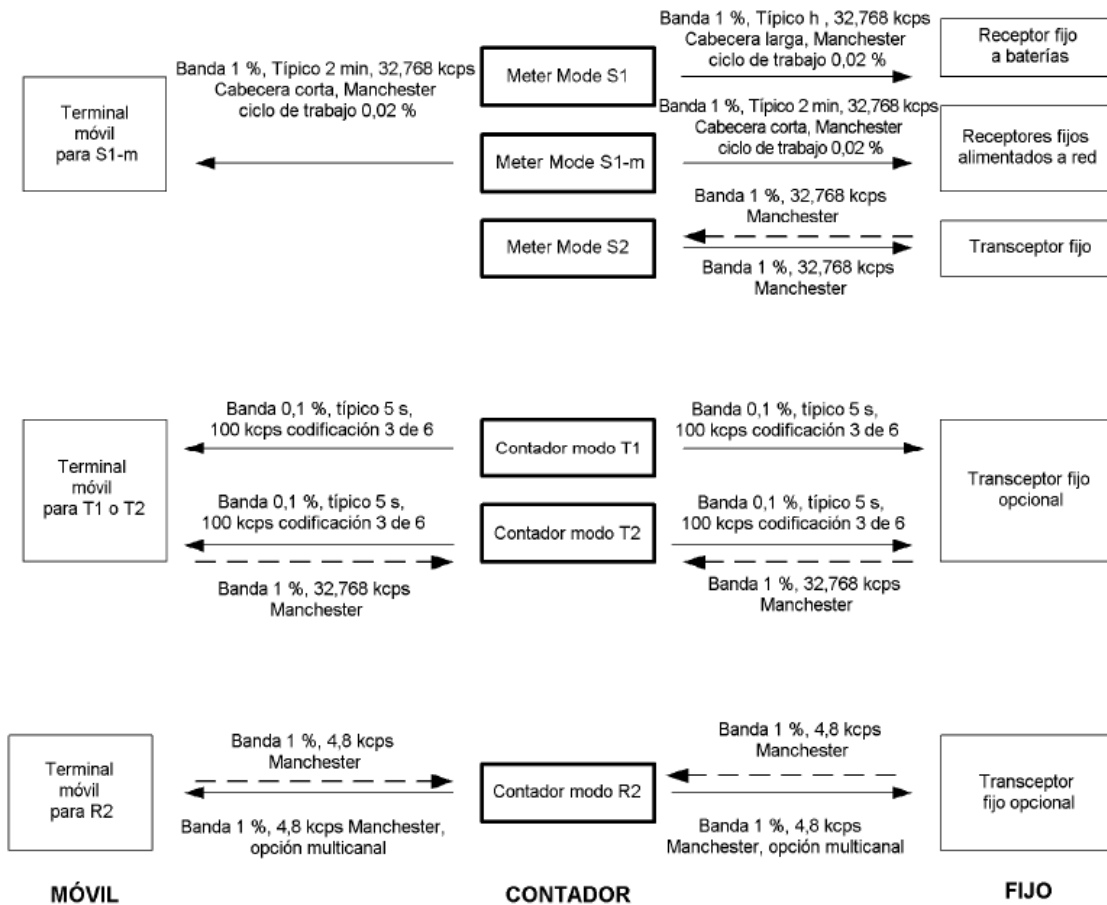
Modo	Vía	Aplicación Típica	Velocidad de chip	Clico de trabajo <sup>a</sup>	Máximo ciclo de trabajo <sup>b</sup>	Codificación de datos + cabecera	Descripción
S1	1	Contador de solo transmisión para lectura de recepción fija	32,768	1%	0,02%	Manchester + cabecera larga	Sólo transmisión; transmite un número de veces al día a un punto de recepción fijo. Transmite a la banda de frecuencia de 1% de ciclo de trabajo. Debido a su larga cabecera es apropiado para receptor de batería economizada
S1-m	1	Contador de solo transmisión para lectura móvil o fija	32,768	1%	0,02%	Manchester + cabecera corta	Sólo transmisión; transmite con una limitación del ciclo de trabajo del 0,02% por hora a un punto de recepción móvil o fijo. Trasmite en la banda de frecuencia de 1% del ciclo de trabajo. Requiere un receptor continuamente habilitado
S2	2	Todos los tipos de contadores. Lectura fija	32,768	1%		Manchester + cabecera corta u opción de cabecera larga	Contador con un receptor, ambos continuamente habilitados o sincronizados no requiriendo un preámbulo extenso para despertar. También es útil para nodos transpondedores o concentradores. Cabecera larga opcional.
T1	1	Transmisión frecuente (contadores de tramar corta)	100	0,1%		3 de 6 + cabecera corta	Transmite sólo cortas ráfagas de datos, <5ms cada pocos segundos, opera en la banda de frecuencia de 0,01% del ciclo de trabajo
T2	2	Transmisión frecuente (trama corta con capacidad bidireccional)	Contador: Tx: 100  Contador: Rx: 32,768	0,1%  1%		3 de 6 + cabecera corta  Manchester + cabecera corta	El contador transmite de forma regular como el Modo T1, su receptor está habilitado durante un corto periodo tras el final de cada transmisión y lo mantiene si recibe una confirmación (a 32,768 kcps)  Además, puede seguir una comunicación bidireccional en la banda de frecuencia del 0,1% usando 100 kcps (transmisión del contador) y 32,768 kcps (recepción del contador). Téngase en cuenta que la comunicación del contador al "otro" componente usa la capa física del modo T1, mientras que los parámetros de la capa física para la dirección contraria son idénticos a los del modo S2.
R2	2	Recepción frecuente	4,8	1%		Manchester + cabecera media	Receptor de contador con posible economizador de batería que requiere un preámbulo extenso para despertar. Opcionalmente, puede haber hasta 10 canales de división de frecuencia multiplexados. La respuesta del contador con 4,8 kcps de despertar seguidos pun una cabecera de 4,8 kcps.
Todos		Opción multimodo					Un componente del sistema puede operar simultáneamente, secuencialmente o mediante comandos en más de un modo mientras complete todos los requerimientos de cada uno de los modos.

<sup>a</sup> Se debe ajustar la limitación del ciclo de trabajo a la asignación de la banda de frecuencias definida para la operación en las bandas de 686 MHz a 870MHz, según CEPT/ERC/REC 70-03 E.

<sup>b</sup> Se debe limitar el ciclo de trabajo por contador al 0,02% por hora para limitar la ocupación total del canal a <10% con 500 contadores instalados dentro de un rango de transmisión

*Tabla 2. Tipos de comunicación en contadores*

La *figura 6* nos ayudará a entender algo mejor las operaciones que se realizan entre los diferentes modos y componentes.



*Figura 6. Tipos de comunicación en contadores*

A su vez este protocolo contiene diferentes clases de funcionamiento según la potencia que irradia ya sea transmisor o receptor. [STD1]

Tras una visión general de los diferentes modos, analizaremos independientemente cada uno de los modos y su implementación.

### 2.2.1. Modo S

Tal y como se ha visto en la tabla inicial de modos el modo S dispone de varios submodos pero vamos a analizar su funcionamiento de una manera genérica.

## a) Capa física

El modo S trabaja en una Banda de frecuencia entre 868MHz y 868,6MHz, ya que es lo que establece la norma europea.

- Modo Transmisor: Los parámetros de los transmisores deben ser como se listan a continuación:

Característica	Modo	Símbolo	Min.	Típico	Máx.	Unidad	Nota
Frecuencia central (contador de solo transmisión, submodo S1)			868,25	868,30	868,35	MHz	$\sim 60 \times 10^{-6}$ (ppm)
Frecuencia central (otro y modo S2)			868,278	868,300	868,322	MHz	$\sim 25 \times 10^{-6}$ (ppm)
Desviación FSK			$\pm 40$	$\pm 50$	$\pm 80$	kHz	
Velocidad de chip de transmisión		$f_{\text{chip}}$		32,768		kcps	
Tolerancia de velocidad de chip					$\pm 1,5$	%	
Fluctuación (Jitter) de bit digital <sup>a</sup>					$\pm 3$	us	
Velocidad de datos (Manchester) <sup>b</sup>				$f_{\text{chip}} \times \frac{1}{2}$		bps	
Longitud del preámbulo incluyendo en bit/byte de sincronismo, en ambas direcciones	S2 S1-M		48			chips	
Longitud del preámbulo incluyendo en bit/byte de sincronismo	S1	PL	576			chips	Opcional para S2
Longitud del epílogo (cola) <sup>c</sup>			2		8	chips	
Retardo de respuesta <sup>d</sup>		$t_{\text{RO}}$	3		50	ms	

<sup>a</sup> La fluctuación del bit se debe medir a la salida del microcontrolador, o circuito codificador.  
<sup>b</sup> Se debe codificar cada bit como dos chips (codificación Manchester).  
<sup>c</sup> El epílogo (cola) debe consistir en n=1 a 4 "unos", por ejemplo la secuencia de chip es n\*(01).  
<sup>d</sup> Retardo de respuesta: después de transmitir una trama en modo S2, el receptor debe estar listo para la recepción de una respuesta en un tiempo menor que el mínimo retardo de respuesta, y debe estar recibiendo al menos lo que dure el máximo retardo de respuesta.

Tabla 3. Modo S transmisor

- Modo Receptor: El receptor se debe adaptarse a los siguientes parámetros.

Característica	Clase	Símbolo	Min.	Típico	Máx.	Unidad	Nota
Sensibilidad (BER < $10^{-2}$ ) o (Tasa de aceptación de bloques > 80%)	H <sub>R</sub>	P <sub>o</sub>	-100	-105		dBm	
Prestación de bloqueo <sup>a</sup>	L <sub>R</sub>		3			Clase	
Prestación de bloqueo <sup>a,c</sup>	M <sub>R</sub>		2			Clase	
Prestación de bloqueo <sup>a,b,c</sup>	H <sub>R</sub>		2			Clase	
Tolerancia aceptable de velocidad de chip		D <sub>chip</sub>			$\pm 2$	%	
Velocidad de chip (contador)		$f_{\text{chip}}$		32,768		kcps	

<sup>a</sup> Clase de receptor según el apartado 9.3 de la Norma ETSI EN 300 200-1, V1.3.1:2000.  
<sup>b</sup> Requisito adicional para receptores H<sub>R</sub>: la selectividad de banda adyacente debe ser > 40 dB cuando se mida según el apartado 9.2 de la Norma ETSI EN 300 200-1, V1.3.1:2000.  
<sup>c</sup> Requisito adicional para receptores de clase M<sub>R</sub> y H<sub>R</sub>: el equipo debe cumplir los requisitos de inmunidad especificados en el apartado 9.2 de la Norma ETSI EN 301 489-1, V1.4.1:2002.

Tabla 4. Modo S receptor



## b) Capa de datos y de aplicación

En este apartado estudiaremos como se codifican y como se deben transmitir los datos en el Modo S.

- Codificación de los datos:

La codificación realizada durante la retransmisión es la Manchester, la cual permite una codificación/decodificación simple y ocupa una banda base más estrecha. Cada bit se codifica de modo que cada secuencia de “10” se representa con un “0” y “01” se representa con “1”.  
**[WIKIMNC]**

- Orden de transmisión:

Se deben transmitir primero el bit más significativo de cada byte de datos, según norma EN 13757-3 **[STD2]**. En la secuencia de CRC primero se transmite el byte más significativo. En el caso de la ID de fabricante, primeramente se envía en byte menos significativo. El resto de datos envían primero el byte menos significativo

- Secuencias del preámbulo, palabra de sincronización y epílogo:

Esta secuencia es introducida por el propio microcontrolador. En nuestro caso el preámbulo y epílogo deben estar formados por  $n \times (01)$  y la palabra de sincronización sería 000111011010010110.

Para el preámbulo  $n \geq 279$  en el caso de submodo S1 y S2 (con cabecera larga opcional), y  $n \geq 15$  para el submodo S2 (cabecera corta)

Hay que indicar que esta codificación de datos se usa igualmente en los modos T2 y R2.

- Enlace de datos:

Para el enlace de datos este protocolo se basa en la capa de enlace de la norma EN 60873-5-1 con la clase de formato FT3.

- Formato trama:

Por lo general los diferentes modos en los que se basa Wireless MBus tienen el mismo formato de trama. Esta trama está compuesta por 2 bloques + 1 opcional

- Bloque 1

Campo L	Campo C	Campo M	Campo A	Campo CRC
1 byte	1 byte	2 bytes	6 bytes	2 bytes

*Figura 7. Estructura Bloque 1*

○ Bloque 2

Campo CI	Campo de datos	Campo CRC
1 byte	15 bytes o, si es el último bloque, $((L-9) \text{ módulo } 16)-1$ bytes	2 bytes

*Figura 8. Estructura Bloque 2*

○ Bloque Opcional

Campo de datos	Campo CRC
16 bytes o, si es el último bloque, $((L-9) \text{ módulo } 16)$ bytes	2 bytes

*Figura 9. Estructura Bloque Opcional*

• Definición de los campos:

La definición de estos campos se explica de manera específica en las Normas EN 60870-5-1 y EN 60870-5-2

- Campo L: Se trata de la longitud del frame, excluyendo los campos L y CRC
- Campo C: El campo de control especifica el tipo de trama que es. Se deben usar los códigos especificados en la Norma EN 60870-5-2.

En el caso del Modo S se utilizan los siguientes códigos:

- C = 44h si el modo es S1 solo transmisión (Enviar/No responder)
- C = 06h (Modo S2 Bidireccional) o C = 46h (Modo S1 unidireccional), si nos encontramos en modo instalación.
- En el caso de modo S2 normal, podremos utilizar todos los valores definidos en la Norma EN 60870-5-2
- Campo M: ID del fabricante. Estos dos bytes deben estar formados por tres letras (A...Z) las cuales forman un identificado único del fabricante. Esta identificación se debe codificar tal y como se especifica en la Norma EN 13757-3:2004 apartado 5.5.

Si el bit más significativo de estos 2 bytes es 0 entonces la dirección A es única y está especificada por el fabricante

En el caso de ser un 1, la dirección debe ser única dentro del rango de actividad del sistema. Este tipo de dirección se asigna una vez instalado el dispositivo.

- Campo A: Este campo contiene la información de la dirección del dispositivo y tal y como se ha dicho anteriormente debe ser única. Si se usa junto con la capa de aplicación de la norma EN 13757-3 y el campo CI vale 72h, 78h o 7Ah se debe generar este campo

como una concatenación de “número de identificación”, “número de versión” e “información del tipo del dispositivo” tal y como se especifica en los apartados 5.4, 5.6, 5.7 de la norma anterior.

- Campo CI: Información de Control. Este campo especifica el tipo de protocolo y la naturaleza de la información que le sigue. El valor de este campo debe ser el siguiente:

Valor CI	Designación	Observaciones
51h	Datos enviados por el dispositivo de lectura al contador sin una cabecera fija (por definir)	Para compatibilidad con la Norma de Capa de Aplicación EN 13757-3
71h	Reservado para Informe de Alarma	Para compatibilidad con la Norma de Capa de Aplicación EN 13757-3
72h	Capa de Aplicación EN 13757-3 con cabecera completa	Para compatibilidad con la Norma de Capa de Aplicación EN 13757-3
78h	Capa de Aplicación EN 13757-3 sin cabecera (a definir)	Para compatibilidad con la Norma de Capa de Aplicación EN 13757-3
7Ah	Capa de Aplicación EN 13757-3 con cabecera corta	Para compatibilidad con la Norma de Capa de Aplicación EN 13757-3
81h	Capa de Aplicación de Repetidores	Para futuro desarrollo
82h	Para uso futuro	Para compatibilidad con las Normas CENELEC TC 205
A0h-b7h	Capa de aplicación específica del fabricante	

*Tabla 5. Valores CI*

- Campo CRC: Chequeo de Redundancia Cíclica [X]. Este chequeo se debe realizar sobre la información que contiene el bloque y debe generarse según el FT3 de la norma EN 60870-5. La fórmula es:

- Polinomio CRC =  $x^{16} + x^{13} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^2 + 1$
- El valor inicial es 0
- El CRC final está complementado

En el anexo D se puede ver parte de la generación de código destinado a la implementación de este modo de conexión

Las estructuras del modo S son muy similares a las que se implementan en el Modo T y modo R por lo que en el análisis de los siguiente modos destacaremos sus diferencias y las partes más técnicas serán anexadas.

## 2.2.2. Modo T

### a) Capa física

La estructura y los requerimientos del modo T en funcionamiento general, transmisor y T2 Receptor se adjuntan en los anexos A y E

### b) Capa de datos y aplicación

En este punto caso vamos a diferenciar la capa de datos y la parte de codificación de los receptores y transmisores ya que difieren en un par de puntos. En un primer momento vamos a ver las características del transmisor:

- **Codificación contador:**  
Para los modos T1 y T2 el contador debe usar una codificación “3 de 6” para conseguir una eficiencia mejorada comparada con la codificación Manchester. En el caso de las placas utilizadas no soportan este tipo de codificación, por lo que el transmisor CC1101 debe configurarse en modo NRZ y la codificación/decodificación de los datos la debe realizar el microcontrolador.
- **Orden de transmisión de datos:**  
Los datos codificados como “3 de 6”, deben transmitirse con el bit más significativo primero y con el nibble más significativo primero. El resto de datos deben ser enviados como se realiza en el modo S, CRC byte más alto primero, resto de datos byte menos significativo.
- **Preámbulo, palabra de control y epílogo:**  
En el caso del preámbulo sigue el mismo patrón que el Modo S,  $n \geq 19$ . En este caso la palabra de control es 00001111001 y el epílogo tiene el mismo tamaño que el Modo S

Ahora veamos las características necesarias para la Transmisión:

- **Codificación:**  
En el caso del Modo T2 utilizaremos la codificación Manchester como se hace en el Modo S
- **Transmisión de datos:**  
Se transmite el bit más significativo de cada byte. En el caso del CRC primero se envía el byte más significativo. Con el resto de los datos se envía primero el byte menos significativo
- **Preámbulo, palabra de control y epílogo:**  
En este caso se trata del mismo sistema que se realiza en el Modo S2, con  $n \geq 15$  en el preámbulo y misma palabra de control y mismo epílogo.

Tras un vistazo a la codificación observaremos que la capa de datos es similar a al modo S

- Enlace de datos:

Tal y como ocurre en el Modo S, se debe utilizar la capa de enlace de datos de la Norma EN 60870-5-2 con la clase de formato FT3

- Formato trama:

El formato de la trama y el valor de sus campos es el mismo que en el Modo S. Los únicos cambios que se pueden apreciar entre el Modo S y el Modo T son los campos C y CI. El campo C sigue manteniendo los valores que deseamos cuando se trata de envío de información sin respuesta y comunicación uni/bidireccional, pero para el submodo T2 el contador debe enviar con frecuencia el mensaje “Petición de acceso” que corresponde con C=48h y debe esperar una confirmación, la cual sería C=00h, a esa petición. En el caso del campo IC, eliminamos el valor 81h ya que no hay aplicación para repetidores. El método CRC es general para todo el protocolo.

- Polinomio CRC =  $x^{16} + x^{13} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^2 + 1$
- El valor inicial es 0
- El CRC final está complementado

### 2.2.3. Modo R2

#### a) Capa física

La estructura y los requerimientos del modo R2 en funcionamiento general, transmisor y R2 Receptor se adjuntan en Anexo B

#### b) Capa de datos y aplicación

Este caso es muy similar al del Modo S. Remarcaremos las principales diferencias y en el caso de no haberlas se indicará el método utilizado

- Codificación:

Al igual que en el Modo S la codificación utilizada es la Manchester

- Transmisión de los datos:

Se transmite el bit más significativo de cada byte. En el caso del CRC primero se envía el byte más significativo. Con el resto de los datos se envía primero el byte menos significativo

- Preámbulo, palabra de control y epílogo:  
El preámbulo sigue la misma estructura de siempre, nx(01), siendo en esta ocasión  $n \geq 39$ . La palabra de control y el epílogo es igual que en el Modo S
- Enlace de datos:  
Tal y como ocurre en el Modo S, se debe utilizar la capa de enlace de datos de la Norma EN 60870-5-2 con la clase de formato FT3
- Formato trama:  
Al igual que todos los modos la trama no se modifica, únicamente el contenido de sus campos C y CI.  
En el caso de C, si el contador usa el valor 06h para indicar que está en modo instalación. Si no está en modo instalación puede utilizar cualquiera de los valores recogidos en la Norma EN 60870-5-2, como por ejemplo C =4Bh (Petición/Respuesta), C=08h (Respuesta), C=44h (Enviar/No responder).  
En cuanto a CI, hemos visto que en el Modo T la configuración cambia un poco, pero en el Modo R2 se mantiene igual que en el Modo S
- CRC:  
Mantiene el mismo modelo que en los Modos anteriores
  - Polinomio CRC =  $x^{16} + x^{13} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^2 + 1$
  - El valor inicial es 0
  - El CRC final está complementado

## Capítulo 3.

### Arquitectura del Sistema

#### 3.1. Arquitectura del sistema

Tras analizar diferentes plataforma para la implementación de MBus observamos que los procesadores MSP430 de Texas Instruments (de aquí en adelante TI) nos ofrecía una buena alternativa de implementación y es una plataforma en la cual se está empezando a desarrollar la tecnología MBus.

##### 3.1.1. ¿Qué es MSP430?

Se trata de microcontroladores de muy-bajo-consumo, cuya estructura es RISC de 16Bit, que proporcionan una solución adecuada para los sistemas de medición que funcionan con baterías. Este tipo microcontroladores usa señal mixta y señal digital por lo que permite conectar simultáneamente interfaces analógicas, digitales y sensores sin que exista un consumo excesivo de energía. [TIMSP430]

##### a) Características

Cada dispositivo cuenta con un sistema de reloj flexible el cual le permite llevar hasta 7 modos de baja potencia (LPMs), por lo que se incrementa la optimización. Cuando se producen los periodos de interrupción, MSP430 se asegura de que nuestra aplicación solo utilice la periferia y relojes adecuados para la tarea a realizar.

##### b) Medidas Claves para el bajo consumo

- Diferentes modos de baja potencia
  - Modo Activo: por debajo de 120 $\mu$ A/MHz a 2.2V
  - Modo Espera/auto despertar y retención RAM (LMP3): por debajo de 0.7 $\mu$ A a 2.2V
  - Modo Espera/auto despertar (LMP4): por debajo de sub-100nA a 2.2V
  - Modo Apagado/Retención RAM (LMP3.5)\_ por debajo de sub-100nA a 2.2V

- Tiempo instantáneo de sub1- $\mu$ s para despertar desde los modos de potencia

### c) Arquitectura

A continuación podemos observar la estructura general del microcontrolador MSP430.

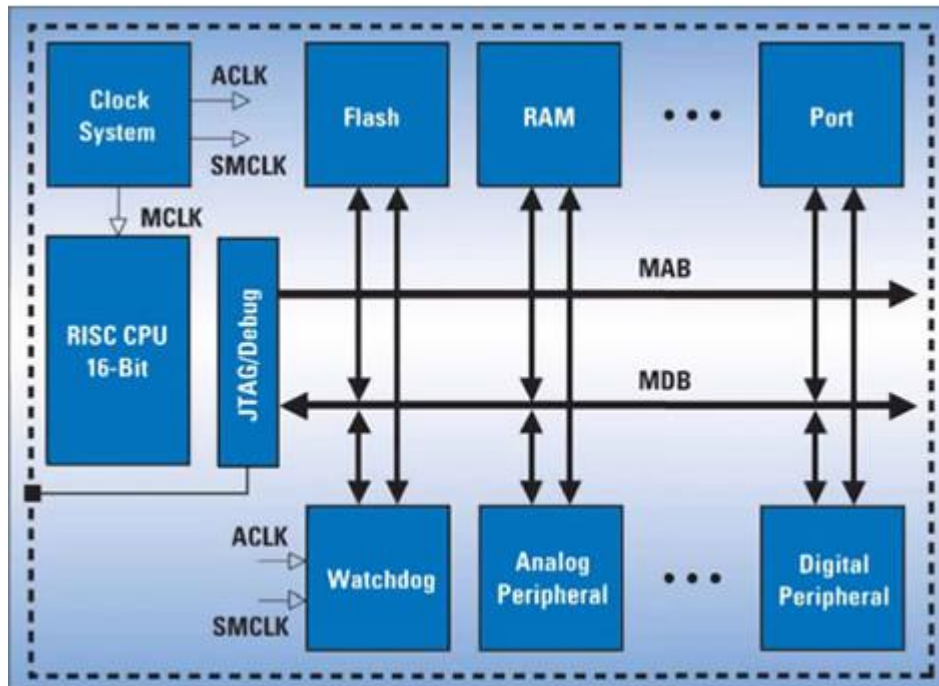


Figura 10. Estructura General Microcontrolador

Y las especificaciones del microcontrolador a utilizar

Series	Part Number	Flash (KB)	SRAM (KB)	I/O	16-Bit Timers			Watchdog and RTC	PMM: BOR, SVS, SVM, LDO	USCI		DMA	MPY	Comp B	Temp Sensor	ADC Ch/Res	Additional Features	Package(s)	1ku Price <sup>1</sup>	Common Features
					Total	A'	B'			Ch A: UART/LIN/IrDA/SPI	Ch B: I <sup>2</sup> C/SPI									
F51xx	CC430F5133	8	2	30	2	5	3	●	●	1	1	3ch	32x32	●	●	6ch ADC12	—	48RGZ	\$4.15	- AES HW Encryption - Max RF Data Rate 500 kbps - Best Sensitivity <sup>2</sup> - 110 dBm
	CC430F5135	16	2	30	2	5	3	●	●	1	1	3ch	32x32	●	●	6ch ADC12	—	48RGZ	\$4.30	
	CC430F5137	32	4	30	2	5	3	●	●	1	1	3ch	32x32	●	●	6ch ADC12	—	48RGZ	\$5.00	
F61xx	CC430F6125	16	2	44	2	5	3	●	●	1	1	3ch	32x32	●	—	—	96seg LCD	64RGC	\$4.35	- Frequency Ranges: 300-348 MHz 389-464 MHz 779-929 MHz
	CC430F6126	32	2	44	2	5	3	●	●	1	1	3ch	32x32	●	—	—	96seg LCD	64RGC	\$4.60	
	CC430F6127	32	4	44	2	5	3	●	●	1	1	3ch	32x32	●	—	—	96seg LCD	64RGC	\$5.05	
	CC430F6135	16	2	44	2	5	3	●	●	1	1	3ch	32x32	●	●	8ch ADC12	96seg LCD	64RGC	\$4.65	
	CC430F6137	32	4	44	2	5	3	●	●	1	1	3ch	32x32	●	●	8ch ADC12	96seg LCD	64RGC	\$3.35	

Figura 11. Especificaciones Familia CC430

Se tratan del modelo CC430F6137 (Marcado en amarillo) el cual combina un procesador MSP430 y un transmisor CC1101, tal y como explicaremos más adelante.



### 3.1.2. Placas Experimentales

Texas Instruments ofrece una larga colección de productos con los que utilizar MSP430. En nuestro caso nos hemos declinado por unas placas basadas en MSP430, específicamente la familia CC430. Se tratan de placas que incluyen un procesador MSP430 basado en la generación 5xx, conocido por su alto nivel de integración, su funcionamiento analógico y su fácil uso. A su vez están equipadas con un sistema de RF sub 1GHz que garantiza los requerimientos necesarios para una buena comunicación en cualquier ambiente. Esta integración nos proporciona la disminución de la potencia sde RF, el tamaño y el coste, obteniendo así un gran rendimiento. [TIWIKICC]

En nuestro caso la placa experimental a utilizar es el modelo CC430F6137 el cual ofrece 64 pins QFN, ADC y la opción de conectar un sistema LCD. A su vez funciona con una memoria de 32KB + 512B de Memoria Flas y 4KB de RAM

Las placas experimentales podemos dividir las en 2. Una de las partes sería el transmisor CC1101. En la *figura 12* podemos apreciar su esquema de funcionamiento

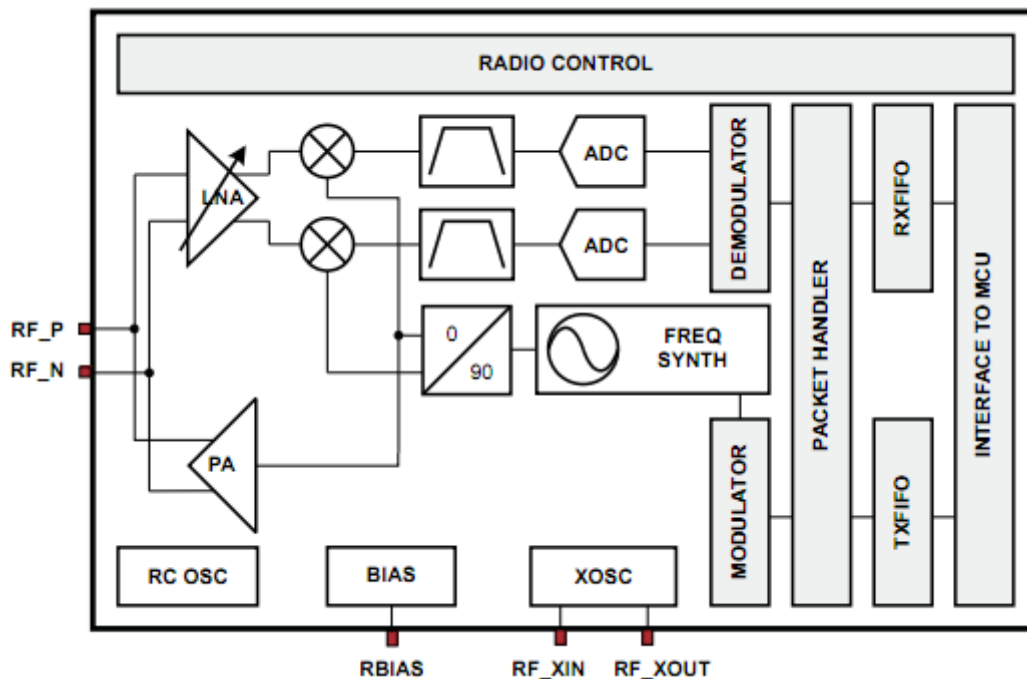
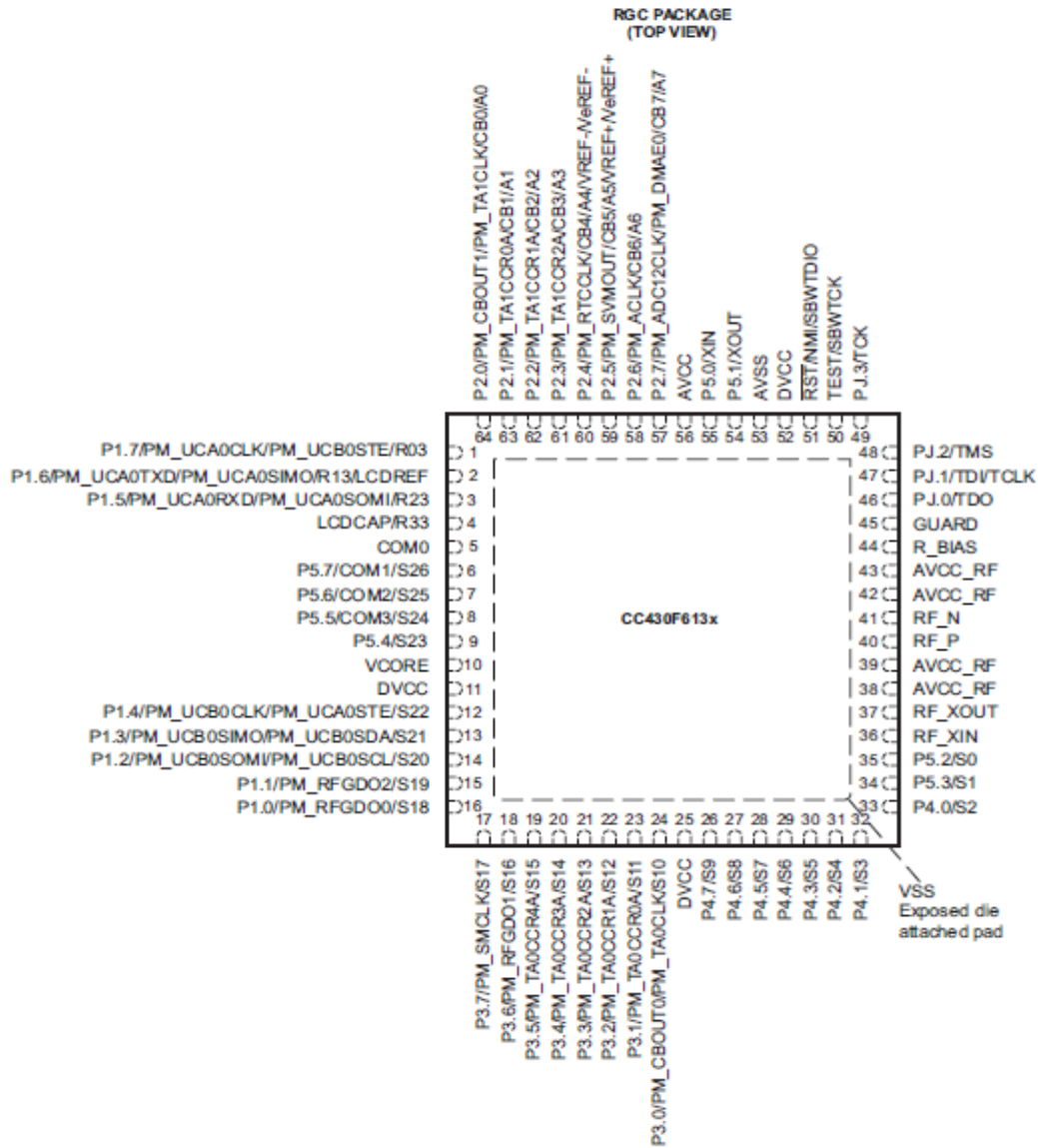


Figura 12. Esquema transmisor

La otra parte de las placas experimentales es la parte del microcontrolador. En la *figura 13* podemos ver un esquema del microcontrolador CC430.

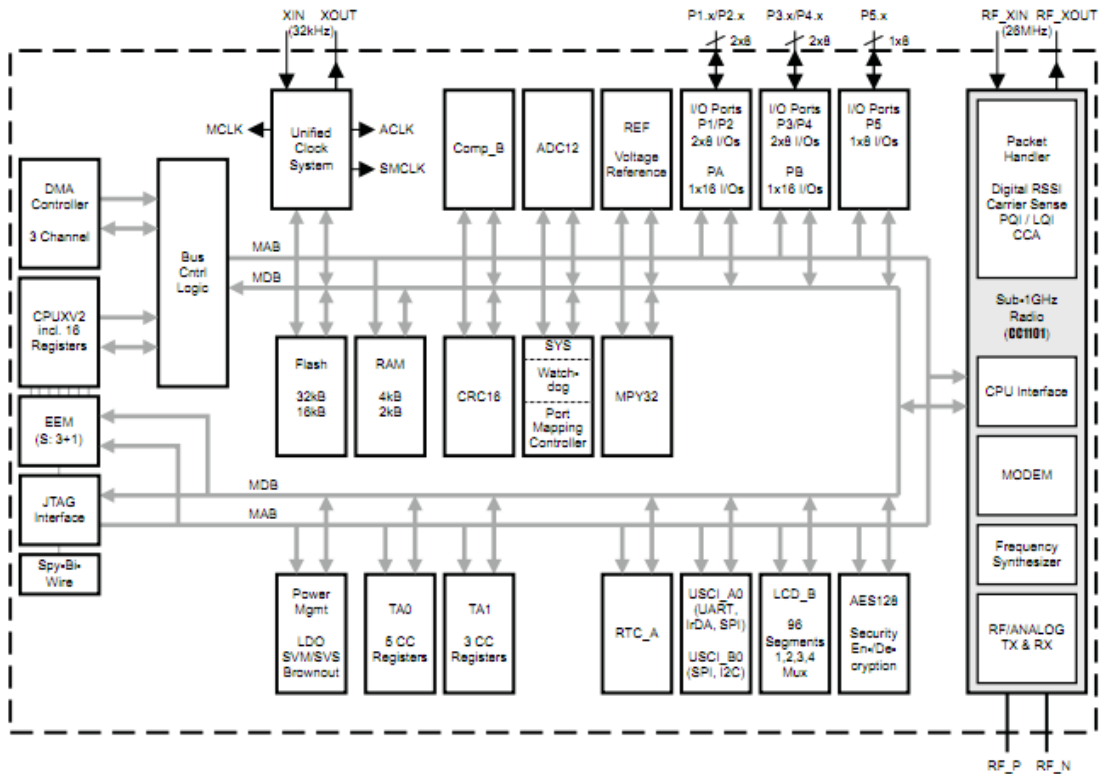


The secondary digital functions on ports P1, P2, and P3 are fully mappable. Pinout above shows only the default mapping. See [Table 7](#) for details.

**CAUTION:** the LCDCAPI/R33 must be connected to VSS if not used.

*Figura 13. Esquema CC430*

Por último nos quedaría mostrar el diagrama de bloque funcional de las placas experimentales CC430F6137, es decir, uniendo la parte de transmisor y microcontrolador.



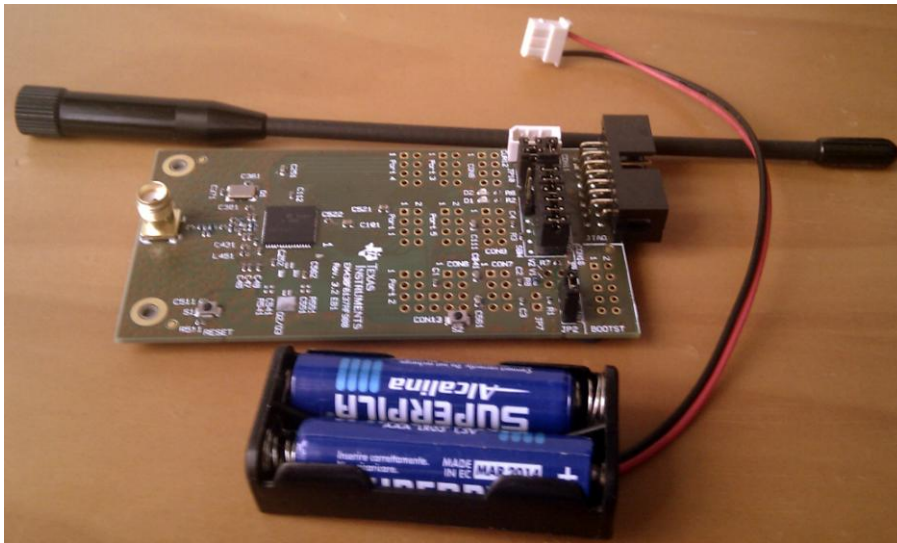
*Figura 14. Diagrama Placas CC430F6137*

A continuación vamos a ver una imagen del debugger, el cual es encargado de conectar las placas al ordenador y poder ejecutar el código programado en estas.



*Figura 15. MSP-FET430UIF*

Por último mostraremos una imagen de una de las placas:



*Figura 16. EM430F6137RF900*

A la hora de realizar el diseño se cuenta con dos placas experimentales (*Figura 16*), las cuales hacen funciones de emisor-receptor según se haya configurado su modo de funcionamiento (S, T, R2).

La familia de microcontroladores MSP430 proporciona una gran variedad de opciones y herramientas para poder realizar y gestionar el trabajo del desarrollador, para

obtener más información sobre la familia del microcontrolador utilizado, consultar referencia [TIRF900]

### 3.1.3. Stack de comunicaciones

Este sistema principalmente está formado por 3 capas de comunicación, física, de datos y aplicación, las cuales se basan en el modelo OSI, tal y como se ha especificado en el estado del arte.

La capa física viene dada por los requerimientos del protocolo Wireless MBus, es decir, velocidades de transmisión, amplitud de señal, frecuencia de transmisión, etc. Para obtener más datos de la capa física revisar estado del arte y la referencia [STR1]

En cuanto a la capa de datos sigue la norma EN 60870-5-2 con la clase de formato FT3. A su vez el formato de la trama se programa tal y como aparece en [STR1], separamos cada uno de los bloques para completarlos con la información necesaria y una vez completos se unen en el buffer TX para su envío. Una vez recibida la trama esta se vuelve a separar en bloques para su análisis y realizar las acciones permitentes.

Por último la capa de aplicación se encarga de general el campo CI y de generar el CRC para más adelante confirmar la integridad de la información, entre otros. En el caso de que el sistema estuviese configurado como receptor esta capa trata los datos recibidos para saber que acción debe realizar a continuación.

## 3.2. Ámbito Sistema

A la hora de empezar con el proyecto, la propia compañía (TI) nos ofrece dos entornos de desarrollo. El primero que ofrece es la plataforma libre Eclipse con un plugin específico para la programación del procesador MSP-430. Esta opción fue descartada tras una serie de pruebas ya que el método de configuración es complicado y no conectaba correctamente con el dispositivo.

La segunda plataforma, y finalmente elegida, ha sido IAR Embedded Workbench, proporcionada por IAR Systems, empresa la cual está especializada en herramientas para la programación de microcontroladores. Esta plataforma ofrece diferentes versiones, entre ellas una gratuita, la cual es la seleccionada en este proyecto. [IAR]

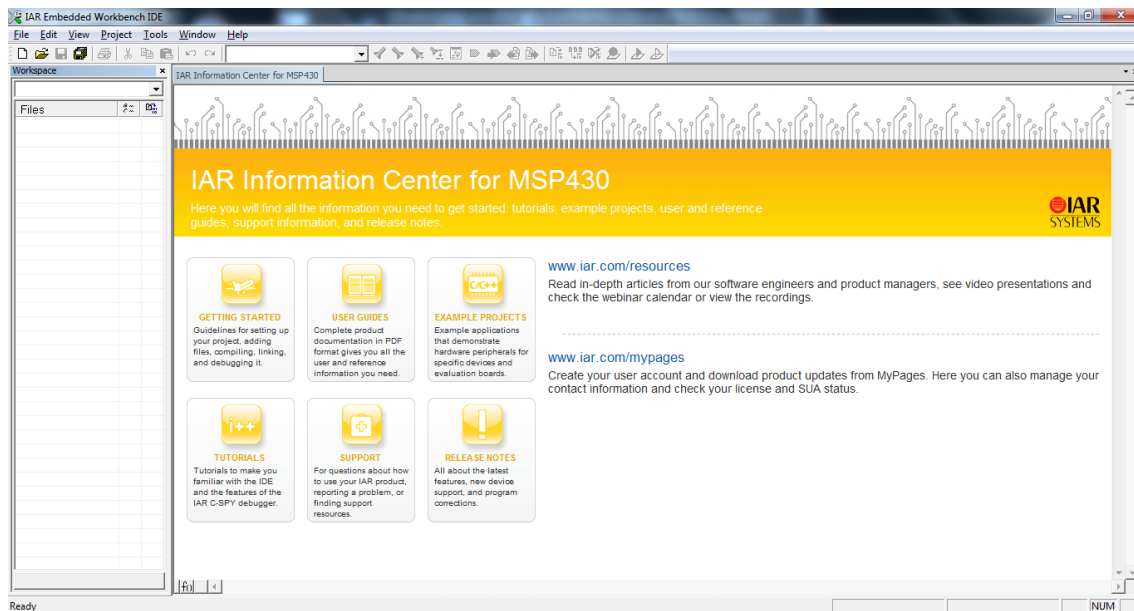
Esta plataforma nos ofrece la posibilidad de programar tanto en ensamblador, C y C++. A su vez podemos usar su simulador o el hardware deseado para realizar nuestros proyectos. Igualmente proporciona una interfaz sencilla e intuitiva, la cual es independiente al microcontrolador utilizado.

A continuación se van a enumerar algunas de las principales ventajas de este entorno de desarrollo:

- Compilador muy optimizado para C y C++

- Archivos de configuración para todos los dispositivos compatibles
- Simulador C-SPY y sistema de debug en hardware
- Reubicación de Ensamblador
- Librerías en tiempo real
- Soporte de debug tipo FET

Por último mostraremos algunas capturas de pantalla del ámbito del sistema seleccionado. En esta primera captura podemos observar la página de inicio que nos muestra IAR. Vemos que nos proporciona diferentes guías y ejemplos de cómo utilizar nuestro MSP430



*Figura 17. Vista Principal IAR*

Aquí podemos ver el entorno de desarrollo en funcionamiento y su visión en modo debug

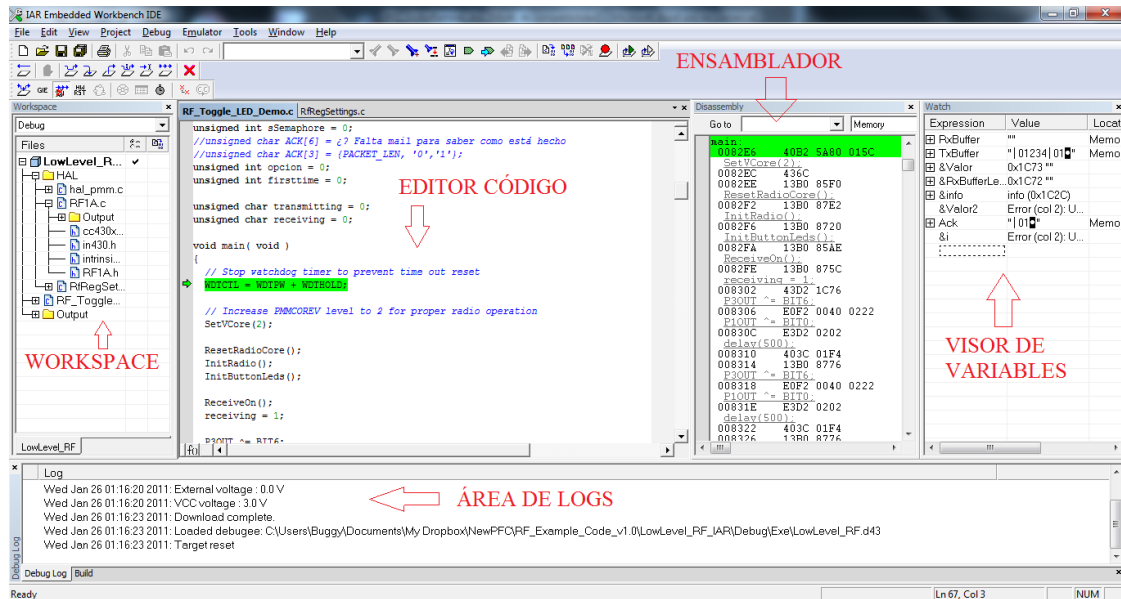
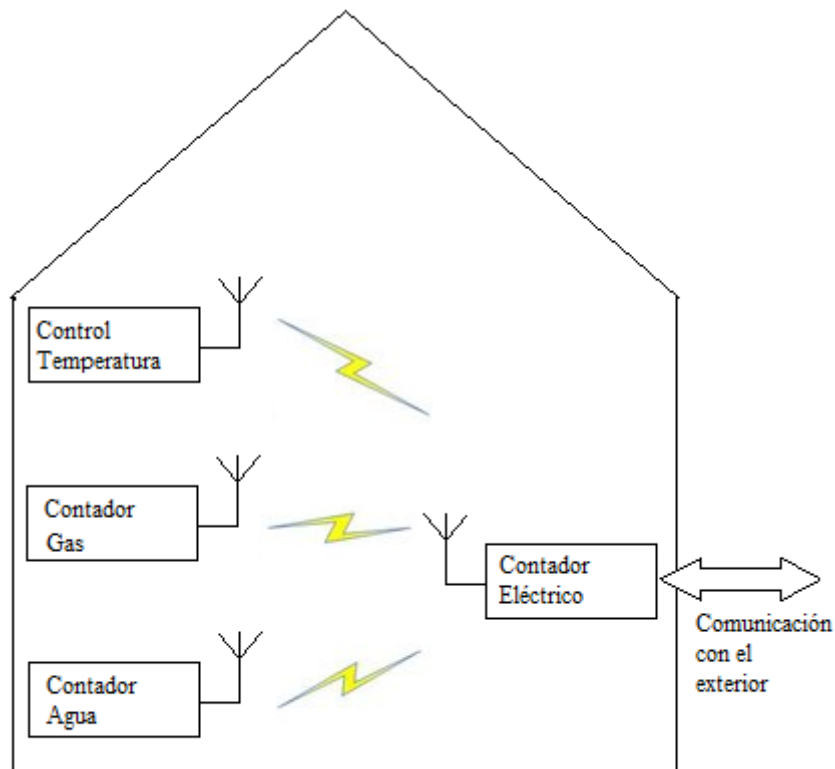


Figura 18. Vista área de trabajo IAR

### 3.3. **Ámbito Cliente**

El ámbito del cliente puede ser muy variado, desde particulares a fábricas, en nuestro caso nos vamos a centrar en el cliente particular.

Es en el ámbito doméstico donde hay que realizar una mayor inversión en la actualización de los utensilios de medición para darle una comodidad al cliente y poder ofrecerle casi de manera instantánea el consumo que lleva una vez cumplido el periodo de facturación. Principalmente nos encontraremos el siguiente esquema en los domicilios:



*Figura 19. Esquema de situación real*

Como podemos observar en la *figura 19* la idea es distribuir en el domicilio los diferentes contadores y sensores de control y que estos se comuniquen con un contador principal, en nuestro caso el eléctrico ya que siempre podrá estar alimentado y disponible.

Gracias al contador eléctrico se realizarán todas las comunicaciones al exterior, enviando la información necesaria a los distribuidores cuando estos la requieran.



### 3.4. Metodología de trabajo

La metodología seguida ha sido bastante lineal ya que ha pasado por diferentes fases de aprendizaje hasta la implementación final del proyecto. En el anexo C se puede ver el diagrama de Gantt para la organización de este.

Durante los primeros meses se ha estado trabajando en el estudio del protocolo, consultando diferentes fuentes, pero en los dos últimos meses de trabajo se ha podido contar con documentación oficial del estándar por lo que se ha podido entender mucho mejor el funcionamiento de este.

El paso siguiente fue comprender como funcionaban las placas experimentales y el micro-procesador. Hasta ese momento nunca me había adentrado en el mundo de los microcontroladores y este tipo de tecnología por lo que entender muchos de los términos asociados me ha costado bastante. A la misma vez he aprendido una nueva forma de trabajar en C, la cual desconocía por completo. El trabajo con registros y microcontroladores es más amplio de lo que me podía imaginar en un principio.

Tras el estudio de los elementos con los cuales se iba a trabajar se realizaron pruebas simples viendo ejemplos proporcionados por TI, enviando datos pequeños de información e intentando hacer cambios en estos para ver como respondía el sistema. Tras estas pruebas lo siguiente ha sido empezar a implementar las características del protocolo para que cumpla su cometido.

Por último, cuando ya se disponía de información suficiente se implementó el protocolo en las placas EM430F6137RF900, ajustando las configuraciones necesarias, tanto en las capas superiores de comunicación como en las intermedias.

## **Capítulo 4.**

### **Implementación del protocolo**

A la hora de implementar el protocolo se deben tener en cuenta diferentes aspectos. Wireless MBus funciona con diferentes modos de trabajo y es el fabricante o proveedor el que decide qué tipo de modo desea implementar. En nuestro caso se proporciona la posibilidad de configurar los transmisores según la preferencia deseada.

#### **4.1. Modo de implementación**

Tras ver el soporte proporcionado por TI para la implementación del protocolo Wireless MBus usaremos como base y guía para nuestra programación los códigos proporcionados por M. Morales y D. Dang. Esta base nos proporciona las comunicaciones básicas entre las dos placas experimentales y el envío y recepción de los datos. El funcionamiento es simple, se envía un paquete de datos y se enciende una luz confirmando la recepción o el envío, según proceda.

Para configurar la capa de datos y física de las placas debemos hacer uso el programa proporcionado por TI, SmartRF Studio [**TISMART**]. Con este programa, una vez conectada la placa al ordenador el software la reconoce y nos da la posibilidad de configurarla a nuestro gusto, desde los modos de transmisión de radio hasta la posibilidad de la codificación Manchester. A su vez podemos configurar los registros que deseamos, pero esto también se puede hacer directamente desde el código creado. Una vez configurada la placa a nuestro gusto exportaremos los archivos creados y los insertaremos en el proyecto. En el caso de este proyecto se ha tenido problemas a la hora de configurar las placas con SmartRF Studio por incompatibilidad del software con el sistema operativo y el firmware entre el programa y las placas experimentales.

Tras varias consultas y búsquedas en los foros de soporte de TI se ha podido implementar el archivo de configuración de RF de forma manual. Se ha procedido a implantar dos tipos de configuración, una para el modo S y otra para el modo T, ya que estas dos son las más utilizadas. En los anexos H e I se puede consultar dicha configuración.

Si se consultan los anexos anteriormente nombrados podemos ver que la codificación Manchester no está implementada, aunque MSP430 de soporte a esta configuración. En su caso se ha creado una codificación Manchester vía software, respetando las normas del protocolo Wireless MBus, la cual permite que en el caso de que las plataformas donde se incorpore este protocolo no tengan soporte de codificación Manchester puedan hacer uso de esta.

Igualmente pasa con la codificación “3 de 6” que se utiliza en el Modo T. Como se ha comentado en el estado del arte, en el apartado de Modo T, el MSP430 no soporta este tipo de codificación, por lo que se debe realizar vía software. Esta codificación codifica un valor de 16 bits en uno de 24bits, en el caso de que se codifiquen solamente 8 bits, se le añade un epílogo a los datos codificados.

Tanto el código diseñado para codificación Manchester y “3 de 6” están preparadas para ser decodificadas y su funcionamiento se basa en tablas conversoras. Este tipo de codificación, al no tener un apoyo hardware puede ralentizar el envío de los datos.

Otro problema con el que nos encontramos es la ejecución del CRC. Las placas experimentales ofrecen un sistema CRC integrado en el hardware por lo que su velocidad de ejecución es mucho mayor en hardware que en software. La pega de este CRC integrado es que no corresponde con el necesario en el protocolo Wireless MBus, por lo que hay que realizarlo vía software, la cual puede ser hasta un 25% más lenta que la implementación hardware. En nuestro caso al tratarse de datos pequeños la diferencia no sería apreciable en cuanto al tiempo de respuesta deseado. Un CRC vía software ofrece varios modelos de implementación, se ha optado por una implementación rápida, la cual se basa en tablas de codificación, igual que ocurre con la codificación Manchester y “3 de 6”, de esta forma los cálculos y resultados se obtienen más rápido.

Una vez vista la implementación de las configuraciones de la capa de datos y física, a continuación se analizarán los modos de transmisión y recepción. [TIWMBUS]

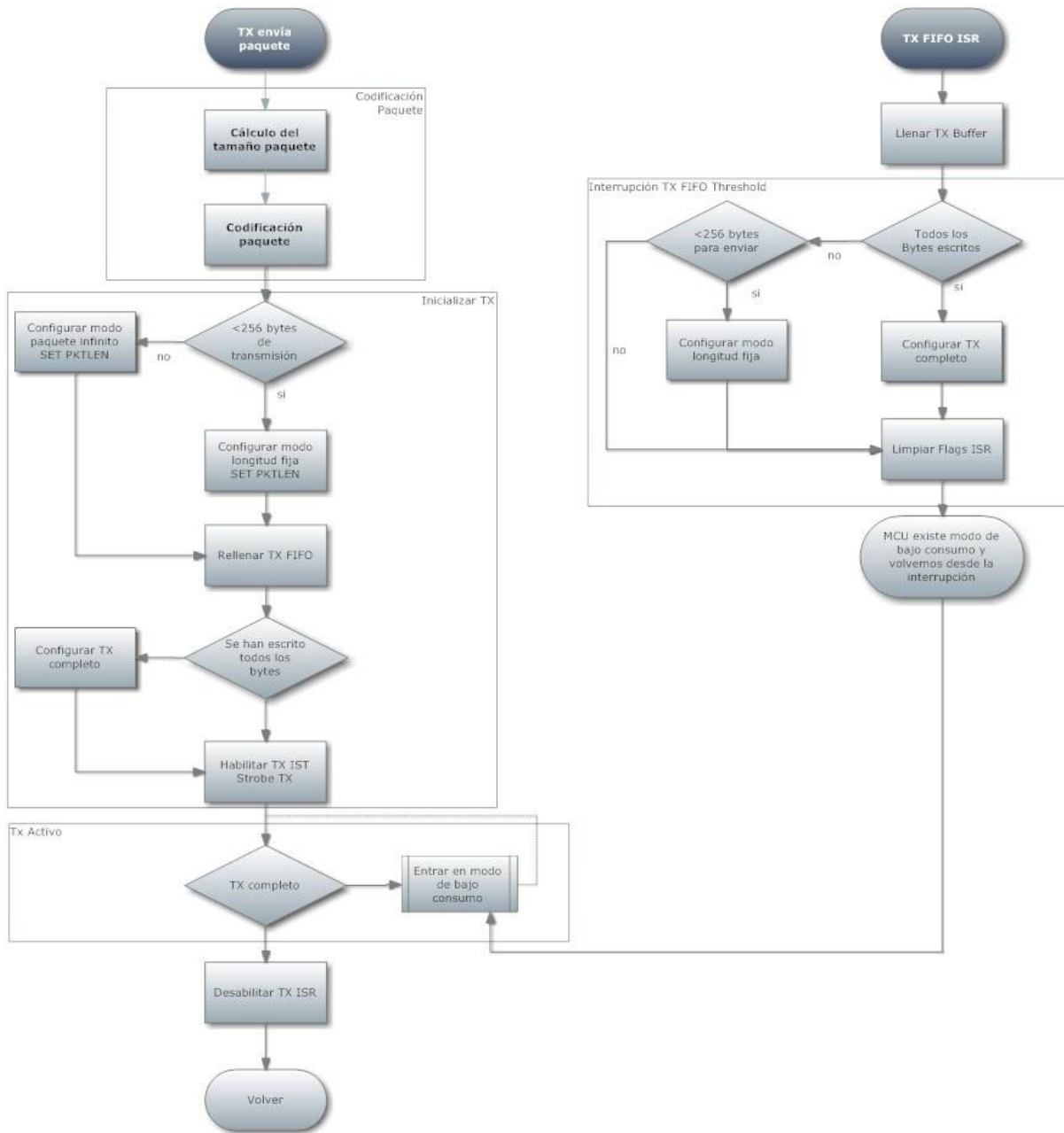
#### **4.1.1. Modo transmisión**

La transmisión de los datos a través de nuestro protocolo se realiza de manera sencilla y lo podemos dividir en cuatro bloques. Indicar que todos los datos enviados se gestionan a través de un buffer TX, el cual está preparado para la comunicación a través de las placas.

- Codificación del paquete
  - Calculamos la cantidad de bytes a enviar
  - Codificamos el paquete, ya sea en codificación Manchester o “3 de 6”
- Inicializamos TX
  - Importamos los datos deseados a TX
  - Habilitamos TX
- TX activo
  - Si no se han escrito todos los bytes a la cola TX FIFO, el MCU se configurará en modo de bajo consumo
- Threshold de TX FIFO interrumpido – poniendo datos en TX FIFO
  - Rellenamos TX FIFO

- Modo de tamaño fijo cuando son menos de 255 bytes a transmitir
- Activar flag cuando todos los datos se hayan escrito

En la *figura 20* podemos ver el diagrama de flujo del modo de transmisión de datos



*Figura 20. Diagrama de flujo TX*

### 4.1.2. Modo recepción

Al igual que se ha dividido el modo de transmisión en 4 bloques, se realizará de manera similar con la recepción de los datos.

- Inicializar RX
  - Configuramos el modo de tamaño infinito
  - Configuramos el Threshold de FIFO en 4 bytes
- Activar RX
  - Si no se han leído todos los datos de RX FIFO, la MCU se configura en bajo consumo
- Interrupción Threshold en RX FIFO de 4 bytes: comienzo del paquete
  - Se lee el tamaño de RX FIFO
  - Se calcula el tamaño total de bytes a recibir
  - Se configura el modo de tamaño fijo si son menos de 255 bytes
  - Se calcula y configura PKLEN
- Interrupción Threshold en RX FIFO de 32 bytes: lectura RX FIFO
  - Se lee RX FIFO
  - Se configura el modo de tamaño fijo si son menos de 255 bytes
- Paquete RX recibido – paquete completo
  - Se terminan de leer los últimos bytes
  - Señalizar la lectura completa de los datos
- Decodificación
  - Se decodifica el paquete y se realiza la comprobación CRC

Indicar que como se aprecia en el esquema anterior no se tienen en cuenta en la recepción del paquete el preámbulo, la palabra de sincronización ni el epílogo, eso son datos de los cuales hace uso el receptor RF pero no afectan de modo alguno en los datos, simplemente ayuda al receptor a saber cuándo empieza un nuevo paquete. A continuación podemos ver el diagrama de flujo del funcionamiento del receptor.

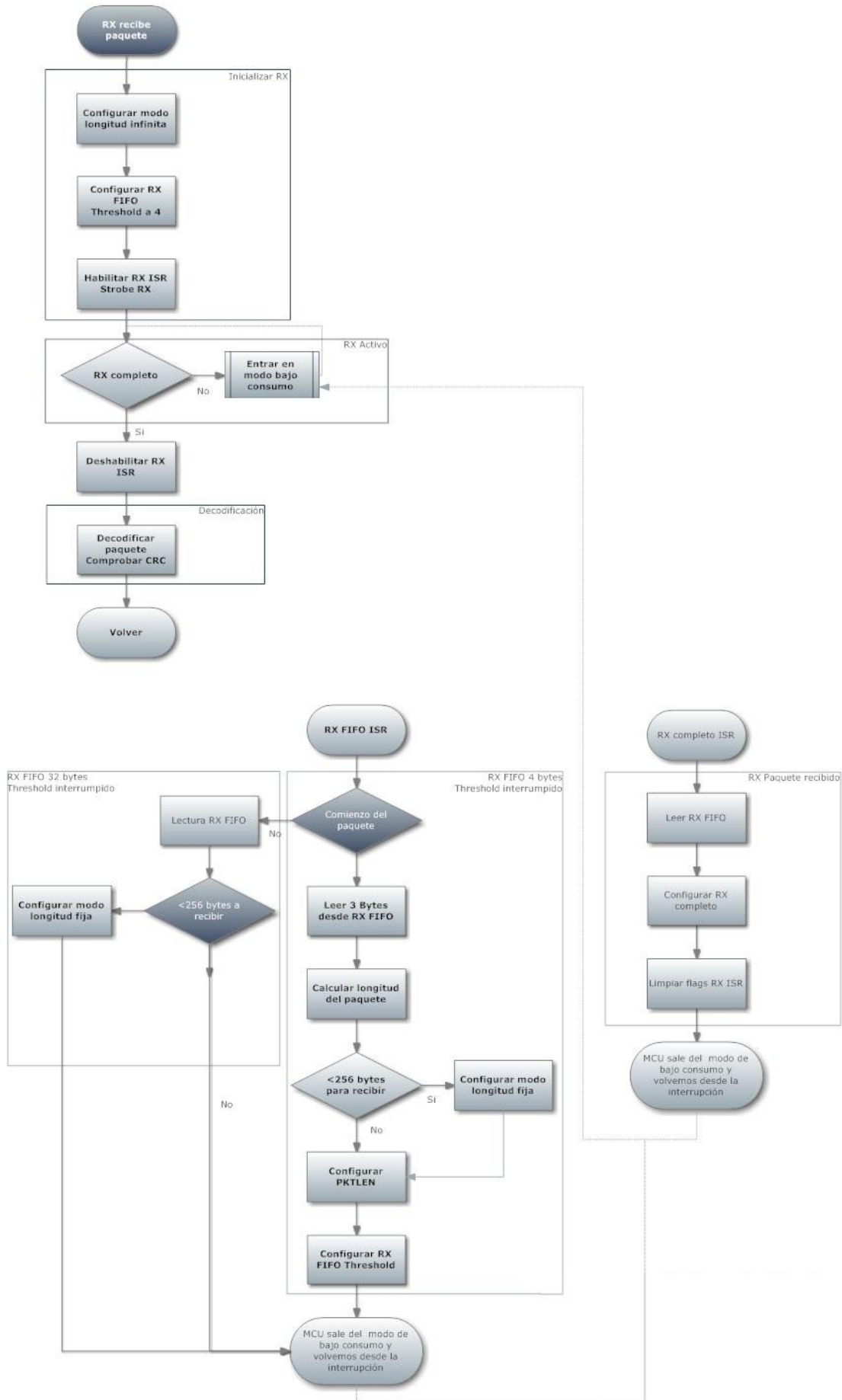


Figura 21. Diagrama de flujo RX

## **Capítulo 5. Test y Resultados**

### **5.1. Entorno de test**

Los test se han desarrollado en un entorno real, un bloque de pisos de 5 plantas más cochera. Cada planta tiene una altura de 3 metros aproximadamente. Las placas experimentales se sitúan en las zonas donde se encuentran los contadores de cada planta y el contador general que recoge los datos de todas las casas. Por lo general estos contadores están por la zona del hueco del ascensor, por lo que la cantidad de metal y aluminio a atravesar es importante. La placa situada en la zona general de contadores está conectada al PC para ver si los resultados que obtiene son correctos. La segunda placa es la que se va situando en las diferentes plantas del edificio.

A través de los botones integrados en las placas se realiza la comunicación entre estas. Una vez pulsado el botón S1 en la placa “móvil” se transforma en emisor y la placa conectada a PC se transforma en receptor. La placa emisora emite una luz roja cuando ha enviado el paquete y la placa receptora emite o apaga una luz verde cada vez que recibe un paquete. Para la utilización de la placa móvil he contado con el apoyo de un compañero el cual iba cambiando de planta cada vez que hacía falta.

Las pruebas se han realizado midiendo el tiempo que tarda en llegar la señal entre las diferentes plantas y se ha llegado a un máximo de 5 plantas, desde la planta 0 de entrada a la 5ª planta. Como se puede apreciar en la descripción la señal ha ido verticalmente y el sistema responde sin problemas sin ningún tipo de repetidor. A su vez se ha realizado el envío y recepción de datos desde el exterior, dejando una de las placas en la zona de contadores general.

El caso experimental que se trata es el envío de los datos tomados por un contador eléctrico. Los campos de cada bloque están completos con información obtenida a través de un contador ficticio siguiendo el protocolo Wireless MBus. La cantidad de datos enviados son 14bytes + prologo, epílogo y palabra de control. Se realizan 6 envíos de datos y se realiza la media, los datos que se muestran en las simulaciones son la media obtenida de las 5 muestras realizadas.

## 5.2. Resultados a nivel de sistema

Se han realizado dos tipos de test. Los primeros test realizados hicieron uso exclusivo del CRC hardware que aporta el MSP430 y los segundos se llevan a cabo con el CRC software.

La transmisión de los datos se realiza siempre en los mismos tiempos ya que el proceso es el mismo en cada momento. Introducción de los datos en el buffer TX y en envío de estos. A continuación podemos ver los datos obtenidos:

Planta Origen	Planta Destino	Tiempo(ms)
5°	0°	168
4°	0°	168
3°	0°	183
2°	0°	177
1°	0°	176
Media		174

*Tabla 6. Envío TX sin CRC software*

Como podemos observar la media de ejecución son 174 milisegundos cuando se ejecuta el código sin CRC software.

En cambio cuando se hace uso del CRC vía software el tiempo de ejecución se incrementa en un 25% aproximadamente. A continuación podemos ver las medias obtenidas para la transmisión de los datos a partir de las mediciones con CRC software.

Planta Origen	Planta Destino	Tiempo(ms)
5°	0°	218
4°	0°	218
3°	0°	218
2°	0°	218
1°	0°	218
Media		218

*Tabla 7. Envío TX con CRC software*



En cuanto a la recepción de los datos, las medias suelen ser un poco más variadas, ya que dependiendo de la distancia a la que se encuentren el transmisor y el receptor la señal tarda algo más en obtenerse. Siguiendo el esquema anterior, primeramente vamos a ver los tiempos necesarios para la recepción de los datos con CRC hardware

Planta Origen	Planta Destino	Tiempo(ms)
5°	0°	180
4°	0°	185
3°	0°	180
2°	0°	172
1°	0°	167
Media		176

*Tabla 8. Recepción RX sin CRC software*

Podemos observar que los tiempos medios de envío y recepción a través de CRC hardware son muy similares, ya que es la propia MCU la que se encarga de comprobar la integridad de los datos enviados.

Al igual que pasaba en la transmisión de datos con CRC software, la recepción de datos con este método de comprobación de integridad hace que esta sea más lenta en otro 25% aproximadamente. En la siguiente tabla podemos ver los datos obtenidos.

Planta Origen	Planta Destino	Tiempo(ms)
5°	0°	223
4°	0°	221
3°	0°	227
2°	0°	228
1°	0°	220
Media		223

*Tabla 9. Recepción RX con CRC software*

Los resultados son muy similares, pero no se pueden extrapolar a edificaciones de mayor tamaño, ya que a mayor altura y sin repetidores la señal tardará más en llegar o se perdería por el camino. La similitud en los datos se debe al rango de distancias testeadas. Por norma general entre las 5 medidas obtenidas por planta siempre se

obtiene una que se sale de la media, esto se debe a que la comprobación del CRC no se encuentra en la memoria RAM y su ejecución es más lenta.

Por último indicar que las medidas realizadas desde el exterior son prácticamente similares a las realizadas entre la planta baja y la 1º planta del edificio. El transmisor externo se ha situado a unos 5 metros de la entrada del domicilio y a 10 de la zona de contadores general.

En cuanto al tamaño de código utilizado no es muy extenso, unos 270K de información, lo cual entra dentro de los límites soportados por el MSP430.

### **5.3. Validación a nivel de sistema**

Tras un análisis de los resultados podemos decir que las medidas de tiempo entran dentro del resultado deseado. En un principio existía la duda de que el método CRC vía software ralentizara en exceso el envío y recepción de los datos, pero se mantiene dentro de unos límites aceptables.

La potencia de las señales enviadas por las placas es la óptima para su implementación en domicilios particulares y mayormente en zonas residenciales ya que la señal no se ve interrumpida o alterada, al menos dentro del rango estudiado (15-20 metros). En el caso de existir algún problema de retransmisión se puede hacer uso de repetidores, o los mismos dispositivos situados en las casas colindantes pueden servir de repetidor.

## **Capítulo 6.**

### **Conclusiones**

#### **6.1. Conclusiones**

Tras la realización de esta memoria podemos sacar diferentes conclusiones que llevan al apoyo de un desarrollo completo de este proyecto.

Primeramente indicar que se han cumplido los objetivos propuestos para este proyecto. Se ha conseguido implementar el protocolo Wireless MBus con algunas dificultades, sobre todo en lo que respecta a la capa física ya que se depende de 3º partes y no depende 100% del programador.

Cabe destacar el funcionamiento, prestaciones y fiabilidad de los microcontroladores MSP430. Su diseño es óptimo para las necesidades de este proyecto y lo más importante es su bajo consumo. Gracias a esto una sola batería puede aguantar hasta 20 años funcionando, disminuyendo drásticamente el mantenimiento de estos sistemas. A su vez es capaz de administrar diferentes periféricos y hacerlos trabajar de manera eficiente.

En cuanto al protocolo Wireless MBus, es un protocolo muy interesante. Su implementación, una vez conocido su funcionamiento, es relativamente rápido y no exige mucha cantidad de código. Los requisitos para las señales RF son fáciles de obtener y además tienen una respuesta bastante buena en lugares cerrados como bloques de vecinos y comunidades. Se ha comprobado que tanto en distancia vertical como horizontal no ha tenido ningún problema para la recepción y transmisión de los datos, además de obtener resultados bastante aceptables

En conclusión, Wireless MBus en combinación con los microcontroladores de bajo consumo MSP430 son una de las mejores opciones del mercado para implementación de lecturas remotas de contadores. Actualmente en España tiene un duro competidor, la tecnología PRIME. Esta tecnología está siendo utilizada por Iberdrola para sus contadores.

#### **6.2. Experiencia personal y profesional**

Este proyecto ha presentado un antes y un después en mi manera de ver parte de la informática. Siempre he estado interesado en la parte software pero gracias a este proyecto me he adentrado en un mundo completamente nuevo para mí. El estudio del funcionamiento de los microcontroladores ha sido muy tedioso, ya que todo son

conceptos nuevos y la mayoría de veces me he pasado horas intentando entender los ejemplos y conceptos.

Igualmente he aprendido a realizar búsquedas en la red más específicas para obtener una respuesta a mis dudas. He descubierto una comunidad de desarrolladores enorme que disfrutan de su trabajo y están dispuestos a ayudarte ofreciéndote todo su conocimiento.

Aún me queda mucho que descubrir y entender del mundo de los microcontroladores, y del mundo hardware en general, pero tras este proyecto se aprecia mucho más el trabajo que hay por debajo del software.

En cuanto al uso de protocolos y estándares también ha sido algo nuevo, ya que durante la carrera solo se estudian y te dicen cómo deben funcionar, pero realmente no implementas ninguno. La única pega que le pongo a trabajar con protocolos y estándares es la multitud de referencias que se tienen y a las que te es difícil acceder para completar la información que necesitas.

Por último, y creo que lo más importante, es que tras este proyecto te das cuenta de lo que es sacar un proyecto adelante, no solo basta con tener una idea o saber programar sino las horas de investigación que llevas a cabo y la inversión que hay que realizar. A su vez aprendes que llevar una programación de trabajo y cumplirla hace que llegues a las fechas de entrega sin problemas

### **6.3. Evolución futura**

Este proyecto tiene muchas posibilidades de evolución. Una de las principales es la configuración de los terminales. El único método disponible con las placas experimentales es a través de los botones que llevan, siendo esta una tarea un poco incomoda. La idea sería poder configurar los modos de funcionamiento de manera remota. Igualmente habría que mejorar los sistemas de interrupción y activación del modo de bajo consumo

Una de las evoluciones importante es la implementación de periféricos tipo bluetooth en el caso de querer una comunicación con sistemas PDA. En muchas ocasiones el sistema de Wifi de estos terminales no están adaptados a la frecuencia que utiliza Wireless MBus, por lo que tenemos que dotar al sistema de un modo eficaz de comunicación con el exterior.

En cuanto a nivel de usuario, con la información contenida en los contadores principales, el usuario puede tener a su disposición aplicaciones de sobremesa o móviles para conocer sus hábitos de consumo y mejorarlos, en el caso de no saber cómo hacerlo el propio sistema le daría consejos para hacerlo.



## Referencias y Bibliografía

1. <http://www.m-bus.com/> [**MBUS**]
2. The M-Bus: A documentation Rev 4.8 by F.Miehlisch [**MBUS2**]
3. <http://en.wikipedia.org/wiki/Meter-Bus> [**WIKIMBUS**]
4. [http://en.wikipedia.org/wiki/Automatic\\_meter\\_reading](http://en.wikipedia.org/wiki/Automatic_meter_reading) [**WIKIAMR**]
5. [http://es.wikipedia.org/wiki/Codificación\\_Manchester](http://es.wikipedia.org/wiki/Codificación_Manchester) [**WIKIMNC**]
6. <http://e2e.ti.com/> [**TI**]
7. <http://processors.wiki.ti.com/index.php/MSP430> [**TIMSP430**]
8. [http://e2e.ti.com/support/microcontrollers/msp43016-bit\\_ultra-low\\_power\\_mcus/f/166.aspx](http://e2e.ti.com/support/microcontrollers/msp43016-bit_ultra-low_power_mcus/f/166.aspx) [**TIMSP430-2**]
9. <http://focus.ti.com/docs/toolsw/folders/print/smartrftm-studio.html>  
[**TISMART**]
10. Wireless MBUS Implementation with CC1101 and MSP430, By Patrick Seem [**TIWMBUS**]
11. CC430 RF Examples. Application Report SLAA465 by Miguel Morales y Dung Dag [**TICC430**]
12. EM430F6137RF900 Reference Design Guide SLAA462 by Miguel Morales [**TIRF900**]
13. MSP430 Hardware Tools User's Guide SLAU278F [**TIUG430**]
14. CRC Implementation Design Note DN502 By Jan-Bjørnar Lund [**TICRC**]
15. MSP430x5xx/MSP430x6xx Family User's Guide SLAU208G [**TIUG6xx**]
16. [http://e2e.ti.com/blogs\\_/b/cc430/archive/2009/12/12/wireless-m-bus-and-the-cc430.aspx](http://e2e.ti.com/blogs_/b/cc430/archive/2009/12/12/wireless-m-bus-and-the-cc430.aspx) [**TIWMBUS-2**]

17. <http://processors.wiki.ti.com/index.php/MSP-FET430UIF> [**TIWIKIFET**]
18. <http://processors.wiki.ti.com/index.php/CC430F613x, CC430F612x, CC430F513x> [**TIWIKICC**]
19. [http://processors.wiki.ti.com/index.php/CC430\\_Sampling\\_Page](http://processors.wiki.ti.com/index.php/CC430_Sampling_Page) [**TIWIKICC430**]
20. <http://focus.ti.com/docs/prod/folders/print/cc430f6137.html> [**TIF6137**]
21. [http://processors.wiki.ti.com/index.php/CC430\\_General\\_Overview](http://processors.wiki.ti.com/index.php/CC430_General_Overview) [**TIWIKICC430G**]
22. <http://www.iar.com/website1/1.0.1.0/1595/1/> [**IAR**]
23. Remote Meter Reading Via M-Bus by Klaus Richter, Energy Management Product Manager. GOSSEN METRAWATT GMBH [**GSRMR**]
24. <http://www.gossenmetrawatt.com/english/seiten/m-bussystemforremotemeterreading.htm> [**GSWEB**]
25. Lectura remota de contadores y sus sistemas de comunicación, Parte 4. UNE-EN 13757-4:2005 [**STR1**]
26. Communication systems for and remote Reading of meters, Part 3. EN 13757-3:2004 [**STR2**]





## Anexos

### Anexo A

#### Requerimientos del modo T Transmisor

Característica	Modo	Simbolo	Mín.	Típico	Máx.	Unidad	Nota
Frecuencia central: (contador a otro)	T1, T2		868,90	868,95	869,00	MHz	$\sim 60 \times 10^{-6}$ (ppm)
Frecuencia central: (otro a contador)	T2		868,278	868,300	868,322	MHz	$\sim 25 \times 10^{-6}$ (ppm)
Desviación FSK: (contador a otro)	T1, T2		$\pm 40$	$\pm 50$	$\pm 80$	kHz	
Desviación FSK: (otro a contador)	T2		$\pm 40$	$\pm 50$	$\pm 80$	kHz	
Velocidad de chip de transmisión (contador a otro)	T1, T2	$f_{\text{chip}}$	90	100	110	kcps	
Variación de la velocidad dentro de la cabecera + trama: (contador)	T1, T2	$D_{\text{chip}}$		0	$\pm 1$	%	
Velocidad de datos: <sup>a</sup> (contador a otro, codificación 3 de 6)	T1, T2			$f_{\text{chip}} \times \frac{2}{3}$		bps	
Velocidad de chip de transmisión: (otro a contador)	T2			32,768		kcps	
Tolerancia de velocidad de chip: (otro a contador)	T2				$\pm 1,5$	%	
Fluctuación (Jitter) de bit digital <sup>b</sup>	T2				$\pm 3$	$\mu\text{s}$	
Velocidad de datos (otro a contador, codificación Manchester)	T2			$f_{\text{chip}} \times \frac{1}{2}$		bps	
Longitud del preámbulo incluyendo en bit/byte de sincronismo, en ambas direcciones.	T1, T2	PL	48			chips	
Longitud del epílogo (cola) <sup>c</sup>	T1, T2		2		8	chips	
Retardo para confirmación <sup>d</sup>	T2	$t_{\text{ACK}}$	2		3	ms	

<sup>a</sup> Se debe codificar cada nibble (4 bits) como 6 chips, véase tabla 11.

<sup>b</sup> La fluctuación del bit se debe medir a la salida del microprocesador, o circuito codificador.

<sup>c</sup> El epílogo (cola) debe consistir en, al menos dos chips alternados. Si el último chip del CRC fue un cero, entonces el mínimo epílogo debe ser "10", si no, debe ser "01".

<sup>d</sup> Retardo para confirmación: después de transmitir una trama incluyendo el epílogo, el contador debe estar listo para la recepción de una confirmación en un tiempo menor que el mínimo retardo de confirmación. El receptor del contador debe, después de que se haya transmitido una trama, escuchar para una posible confirmación durante, al menos, la duración del máximo retardo de confirmación.

### Requerimiento del modo T2 Receptor

Característica	Modo	Símbolo	Min.	Típico	Máx.	Unidad	Nota
Sensibilidad (BER <math>< 10^{-3}</math>) o (Tasa de aceptación de bloques > 80%)	H <sub>R</sub>	P <sub>0</sub>	-100	-105		dBm	
Prestación de bloqueo <sup>a</sup>	L <sub>R</sub>		3			Clase	
Prestación de bloqueo <sup>a c</sup>	M <sub>R</sub>		2			Clase	
Prestación de bloqueo <sup>a b c</sup>	H <sub>R</sub>		2			Clase	
Rango aceptable de velocidad de chip de cabecera: (otro)	T1, T2	f <sub>chip</sub>	88	100	112	kcps	~±12%
Variación aceptable de velocidad de chip durante la cabecera o la trama: (otro)	T1, T2	D <sub>fchip</sub>		0	±2	%	
Velocidad de chip (contador)	T2	f <sub>chip</sub>		32,768		kcps	
Tolerancia aceptable de velocidad de chip (contador)	T2	D <sub>fchip</sub>		0	±2	%	

<sup>a</sup> Clase de receptor según el apartado 9.3 de la Norma ETSI EN 300 200-1, V1.3.1:2000.

<sup>b</sup> Requisito adicional para receptores H<sub>R</sub>: la selectividad de banda adyacente debe ser > 40 dB cuando se mida según el apartado 9.2 de la Norma ETSI EN 300 200-1, V1.3.1:2000.

<sup>c</sup> Requisito adicional para receptores de clase M<sub>R</sub> y H<sub>R</sub>: el equipo debe cumplir los requisitos de inmunidad especificados en el apartado 9.2 de la Norma ETSI EN 301 489-1, V1.4.1:2002.

## Anexo B

### Requerimientos Modo R2 Transmisor

Característica	Símbolo	Min.	Típico	Máx.	Unidad	Nota
Frecuencia central (otro)			868,330		MHz	
Frecuencia central (contador)			868,030 +n × 0,06		MHz	
Tolerancia de frecuencia (contador/otro)			0	±17	kHz	~20 × 10 <sup>-6</sup> (ppm)
Desviación FSK:		±4,8	±6	±7,2	kHz	
Velocidad de chip (despertar y comunicaciones)			4,8		kcps	
Tolerancia de velocidad de chip (despertar y comunicaciones)			0	±1,5	%	
Fluctuación (Jitter) de bit digital <sup>a</sup>				±15	μs	
Velocidad de datos (codificación Manchester) <sup>b</sup>			f <sub>chip</sub> × ½		bps	
Longitud del preámbulo incluyendo el bit/byte de sincronismo	PL	96			chips	
Longitud del epilogo (cola) <sup>c</sup>		2		8	chips	
Retardo de respuesta <sup>d</sup>	t <sub>RO</sub>	3		50	ms	
Retardo de respuesta <sup>d</sup>	t <sub>RM</sub>	10		10 000	ms	

<sup>a</sup> La fluctuación del bit se debe medir a la salida del microprocesador, o circuito codificador.

<sup>b</sup> Se debe codificar cada bit como 2 chips (codificación Manchester).

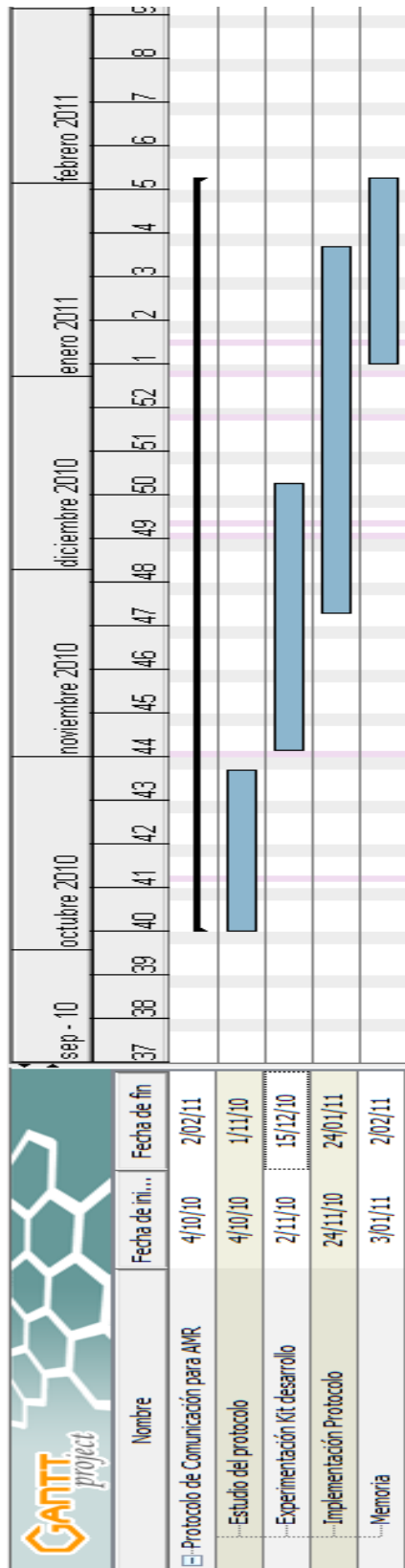
<sup>c</sup> El epilogo (cola) debe consistir en 1 ≤ n ≤ 4 "unos", por ejemplo, la secuencia de chip debe ser n × (01).

<sup>d</sup> Retardo de respuesta: después de transmitir una trama, el receptor del contador debe estar listo para la recepción de una respuesta en un tiempo más corto, que el mínimo retardo de respuesta y debe estar recibiendo, al menos, lo que dure el máximo retardo de respuesta. Se debe usar el retardo de respuesta t<sub>RO</sub> si el campo CI de la trama recibida es 81<sub>b</sub>, de otro modo se debe usar el retardo de respuesta t<sub>RM</sub>.

## Requerimientos R2 Receptor

Característica	Clase	Símbolo	Min.	Típico	Máx.	Unidad	Nota
Sensibilidad ( $BER < 10^{-2}$ ) o (Tasa de aceptación de bloques $> 80\%$ )	H <sub>R</sub>	P <sub>o</sub>	-105	-110		dBm	
Prestación de bloqueo <sup>a</sup>	L <sub>R</sub>		3			Clase	
Prestación de bloqueo <sup>a c</sup>	M <sub>R</sub>		2			Clase	
Prestación de bloqueo <sup>a b c</sup>	H <sub>R</sub>		2			Clase	
Rango aceptable de velocidad de chip		fchip	4,7	4,8	4,9	kcps	~±2%
Variación aceptable de velocidad de chip durante la cabecera o la trama		Dfchip		0	±0,2	%	
<p><sup>a</sup> Clase de receptor según el apartado 9.3 de la Norma ETSI EN 300 200-1, V1.3.1:2000.</p> <p><sup>b</sup> Requisito adicional para receptores H<sub>R</sub>: la selectividad de banda adyacente debe ser <math>&gt; 40</math> dB cuando se mida según los apartados 9.1 y 9.2 de la Norma ETSI EN 300 200-1, V1.3.1:2000.</p> <p><sup>c</sup> Requisito adicional para receptores de clase M<sub>R</sub> y H<sub>R</sub>: el equipo debe cumplir los requisitos de inmunidad especificados en el apartado 9.2 de la Norma ETSI EN 301 489-1, V 1.4.1:2002 y en la Norma EN 301 489-3, V 1.3.1.</p>							

## Anexo C



## Anexo D

Aquí podemos observar las configuración de radio del Modo S

```
#ifndef SMODE_RF_SETTINGS
#define SMODE_RF_SETTINGS

// Product = CC1101
// Chip version = A (VERSION = 0x04)
// Crystal accuracy = 10 ppm
// X-tal frequency = 26 MHz
// RF output power = + 10 dBm
// RX filterbandwidth = 270 kHz
// Deviation = 47 kHz
// Datarate = 32.73 kBaud
// Modulation = (0) 2-FSK
// Manchester enable = (0) Manchester disabled
// RF Frequency = 868.299866 MHz
// Channel spacing = 199.951172 kHz
// Channel number = 0
// Optimization = -
// Sync mode = (5) 15/16 + carrier-sense above threshold
// Format of RX/TX data = (0) Normal mode, use FIFOs for RX and TX
// CRC operation = (0) CRC disabled for TX and RX
// Forward Error Correction = (0) FEC disabled
// Length configuration = (0) Fixed length packets, length configured in PKTLEN register.
// Packetlength = 255
// Preamble count = (2) 4 bytes
// Append status = 0
// Address check = (0) No address check
// FIFO autoflush = 0
// Device address = 0

// GDO0 signal selection = ( 6) Asserts when sync word has been sent / received, and de-asserts at the
end of the packet
// GDO2 signal selection = (41) CHIP_RDY
```

Y aquí los registros usados en el Modo S

```
RF_SETTINGS sModeRfConfig = {
```

```
0x08, // FSCTRL1 Frequency synthesizer control.
0x00, // FSCTRL0 Frequency synthesizer control.
0x21, // FREQ2 Frequency control word, high byte.
0x65, // FREQ1 Frequency control word, middle byte.
0x6A, // FREQ0 Frequency control word, low byte.
0x6A, // MDMCFG4 Modem configuration.
0x4A, // MDMCFG3 Modem configuration.
0x05, // MDMCFG2 Modem configuration.
0x22, // MDMCFG1 Modem configuration.
0xF8, // MDMCFG0 Modem configuration.
0x00, // CHANNR Channel number.
0x47, // DEVIATN Modem deviation setting (when FSK modulation is enabled).
0xB6, // FRENDD1 Front end RX configuration.
0x10, // FRENDD0 Front end RX configuration.
0x18, // MCSM0 Main Radio Control State Machine configuration.
0x2E, // FOCCFG Frequency Offset Compensation Configuration.
0x6D, // BSCFG Bit synchronization Configuration.
0x04, // AGCCTRL2 AGC control.
0x09, // AGCCTRL1 AGC control.
0xB2, // AGCCTRL0 AGC control.
0xEA, // FSCAL3 Frequency synthesizer calibration.
0x2A, // FSCAL2 Frequency synthesizer calibration.
0x00, // FSCAL1 Frequency synthesizer calibration.
0x1F, // FSCAL0 Frequency synthesizer calibration.
0x59, // FSTEST Frequency synthesizer calibration.
0x81, // TEST2 Various test settings.
0x35, // TEST1 Various test settings.
0x09, // TEST0 Various test settings.
0x06, // IOCFG2 GDO2 output pin configuration.
0x00, // IOCFG0 GDO0 output pin configuration. Refer to SmartRF® Studio User Manual for
detailed pseudo register explanation.
0x00, // PKTCTRL1 Packet automation control.
0x00, // PKTCTRL0 Packet automation control.
0x00, // ADDR Device address.
0xFF // PKTLEN Packet length.
};
```

## Anexo E

Aquí podemos observar la configuración de radio del Modo T

```
#ifndef TMODE_RF_SETTINGS
#define TMODE_RF_SETTINGS

// Product = CC1101
// Chip version = A (VERSION = 0x04)
// Crystal accuracy = 10 ppm
// X-tal frequency = 26 MHz
// RF output power = + 10 dBm
// RX filterbandwidth = 325.000000 kHz
// Deviation = 38 kHz
// Datarate = 32.630920 kBaud
// Modulation = (0) 2-FSK
// Manchester enable = (0) Manchester disabled
// RF Frequency = 868.9497 MHz
// Channel spacing = 199.951172 kHz
// Channel number = 0
// Optimization = -
// Sync mode = (5) 15/16 + carrier-sense above threshold
// Format of RX/TX data = (0) Normal mode, use FIFOs for RX and TX
// CRC operation = (0) CRC disabled for TX and RX
// Forward Error Correction = (0) FEC disabled
// Length configuration = (0) Fixed length packets, length configured in PKTLEN register.
// Packetlength = 255
// Preamble count = (2) 4 bytes
// Append status = 0
// Address check = (0) No address check
// FIFO autoflush = 0
// Device address = 0
```

Y aquí los registros usados en el Modo T

```
RF_SETTINGS tModeRfConfig = {
    0x08, // FSCTRL1 Frequency synthesizer control.
    0x00, // FSCTRL0 Frequency synthesizer control.
    0x21, // FREQ2 Frequency control word, high byte.
    0x6B, // FREQ1 Frequency control word, middle byte.
```

```
0xD0, // FREQ0   Frequency control word, low byte.
0x5C, // MDMCFG4  Modem configuration. - 103 kBaud
0x04, // MDMCFG3  Modem configuration.
0x05, // MDMCFG2  Modem configuration.
0x22, // MDMCFG1  Modem configuration.
0xF8, // MDMCFG0  Modem configuration.
0x00, // CHANNR   Channel number.
0x44, // DEVIATN   Modem deviation setting (when FSK modulation is enabled).
0xB6, // FRENDD1   Front end RX configuration.
0x10, // FRENDD0   Front end RX configuration.
0x18, // MCSM0     Main Radio Control State Machine configuration.
0x2E, // FOCCFG    Frequency Offset Compensation Configuration.
0xBF, // BSCFG     Bit synchronization Configuration.
0x43, // AGCCTRL2  AGC control.
0x09, // AGCCTRL1  AGC control.
0xB5, // AGCCTRL0  AGC control.
0xEA, // FSCAL3    Frequency synthesizer calibration.
0x2A, // FSCAL2    Frequency synthesizer calibration.
0x00, // FSCAL1    Frequency synthesizer calibration.
0x1F, // FSCAL0    Frequency synthesizer calibration.
0x59, // FSTEST    Frequency synthesizer calibration.
0x81, // TEST2     Various test settings.
0x35, // TEST1     Various test settings.
0x09, // TEST0     Various test settings.
0x06, // IOCFG2    GDO2 output pin configuration.
0x00, // IOCFG0D   GDO0 output pin configuration. Refer to SmartRF® Studio User Manual for
detailed pseudo register explanation.
0x00, // PKTCTRL1  Packet automation control.
0x00, // PKTCTRL0  Packet automation control.
0x00, // ADDR     Device address.
0xFF // PKTLEN   Packet length.
};
```



## **RESUM**

Aquest projecte tracta d'aportar una idea al futur dels comptadors automàtics, on tant el usuari, com les empreses, tindran un control continu sobre les dades.

A més a més s'han buscat sistemes i protocols que realitzin una bona comunicació entre els comptadors automàtics, en aquest punt on entra la tecnologia W-Mbus.

Aquesta tecnologia juntament amb els microcontroladors de baix consum, proporcionats per Texas Instruments, són la millor combinació per aconseguir el nostre objectiu

## **RESUMEN**

Este proyecto trata de aportar una idea al futuro de los contadores automáticos, donde los usuarios, como las empresas, tendrán un control continuo sobre los datos.

A su vez hay que buscar sistemas y protocolos que realicen una buena comunicación entre los contadores automáticos, aquí es donde entra la tecnología W-MBUS.

Esta tecnología junto a los microcontroladores de bajo consumo, proporcionados por Texas Instruments, son la mejor combinación para conseguir nuestro objetivo

## **SUMMARY**

This project aims to provide an idea about the future of automatic metering readers, where users and companies will have a continuous control over the data

It also necessary to look for systems and protocols that make a good communication between AMR, this is where Wireless MBus technology enters the story.

This technology with low power microcontrollers, provided by Texas Instruments, are the best combination to achieve our goal.