



epsc

**Escola Politècnica Superior
de Castelldefels**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TRABAJO FINAL DE CARRERA

TÍTULO DEL TFC: Nodo autónomo para la medida de CO₂

TITULACIÓN: Ingeniería Técnica de Telecomunicación, especialidad Telemática

AUTOR: Carlos Aparicio León

DIRECTOR: Manel Gasulla Forner

FECHA: 20 de septiembre de 2010

Título: Nodo autónomo para la medida de CO₂

Autor: Carlos Aparicio León

Director: Manel Gasulla Forner

Fecha: 20 de septiembre de 2010

Resumen

Se realiza el diseño, desarrollo e implementación de un nodo sensor para la medición de CO₂. Las características principales son la transmisión de datos inalámbrica, la comunicación digital microcontrolador-sensor y el bajo consumo del sistema, permitiendo al nodo trabajar de forma autónoma. El nodo tiene capacidad para integrarse dentro de la red de sensores inalámbrica REALnet del Campus Baix Llobregat.

En los primeros capítulos se explica cada una de las partes que forman el nodo, tales como el microcontrolador, el sensor y el transceptor inalámbrico. Se detallan las interfaces de comunicación entre los diferentes componentes:

- I²C entre sensor y microcontrolador.
- UART entre transceptor y microcontrolador.
- Zigbee entre transceptor y REALnet.

Se definen 2 modos de trabajo para el nodo sensor. Un modo realiza las mediciones continuas con períodos entre lecturas de 1 minuto o 5 minutos. En este modo el sensor siempre permanece activo. El segundo modo de trabajo permite un ahorro energético importante, realizando lecturas cada 30 minutos o 120 minutos.

En los capítulos finales se explica el sistema de alimentación adoptado y la caracterización de los dispositivos que forman el nodo obteniendo los consumos teóricos. También se muestra el sistema de validación y el diseño definitivo.

Title: Autonomous node for the measurement of CO₂

Author: Carlos Aparicio León

Director: Manel Gasulla Forner

Date: September, 20th 2010

Overview

An autonomous sensor has been designed and implemented of a sensor node in order to measure CO₂. The node has the capability to be integrated into the wireless sensor network REALnet of the Campus del Baix Llobregat.

First chapters explain each of the constituent parts of the node, such as the microcontroller, sensor and wireless transceiver. The communication interfaces between the different components are detailed:

- I2C between sensor and microcontroller.
- UART between transceiver and microcontroller.
- Zigbee between the transceiver and REALnet.

Two modes of operation for the autonomous sensor are defined. One mode performs continuous measurements with periods between sensor readings of 1 minute or 5 minutes. In this mode the sensor always remains active. The second working mode allows significant energy savings, taking readings every 30 minutes or 120 minutes.

The final chapters explain the power supply system adopted and the characterization of the devices that make up the node obtaining the theoretical consumption. The system validation and the final design are also shown.

Dedicatoria

Tras varios años con una espina clavada, he podido realizar el proyecto que tanto tiempo me ha costado empezar. Quiero hacer mención especial a Natalia por concienciarme para que acabara aquello que empecé hace mucho. A mi familia por el apoyo ofrecido durante todo este tiempo.

A David y Sergio por haber soportado todo este tiempo mis ausencias y a Ronald, porque gracias a su flexibilidad he podido llevar a cabo el proyecto, que en otras circunstancias no hubiese sido posible. Muchas gracias a Víctor, con quien he compartido muchas tardes de estudio .

Agradecer a mi tutor, Manel, toda la ayuda prestada y a Francis su colaboración inestimable.

Gracias.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1.EL SENSOR	3
1.1. Introducción	3
1.1.1. El dióxido de carbono[3]	3
1.1.2. Tipos de sensores de CO ₂	4
1.2. Características del sensor utilizado	4
1.3. Comunicación	5
1.3.1. Comunicación con el microcontrolador	7
CAPÍTULO 2. EL MICROCONTROLADOR	13
2.1. Introducción	13
2.1.3. Selección del microcontrolador	13
2.2. Características MSP430F2132.....	16
2.3. Código	21
CAPÍTULO 3: EL TRANSEPTOR ZIGBEE.....	25
3.1. Especificación Zigbee [13]	25
3.1.1. Comparativa con otros sistemas inalámbricos.....	26
3.2. Características módulo ETRX2	26
3.1.2. Órdenes AT	26
3.1.3. Registros S	27
3.1.4. Interrupciones y modo de bajo consumo	28
3.3. Configuración	29
3.3.1. Configuración dispositivo final (nodo sensor).....	29
3.3.2. Configuración coordinador	30
CAPÍTULO 4. ALIMENTACIÓN Y CONSUMO.....	31
4.1. Sistema de alimentación.....	32
4.2. Consumo	34
4.2.1. Consumo del sensor.....	35
4.2.2. Consumo del transceptor Zigbee	36
4.2.3. Consumo del regulador	37
4.2.4. Consumo del microcontrolador.....	38
CAPÍTULO 5. MONTAJE FINAL Y VALIDACIÓN	43
5.1. Montaje final.....	43
5.2. Validación e Interfaz gráfica	45

CAPÍTULO 6. CONCLUSIONES	47
6.1. Objetivos alcanzados	47
6.2. Impacto medioambiental	47
6.3. Líneas de trabajo futuras	48
REFERENCIAS.....	49
BIBLIOGRAFÍA	51
ANEXO A. DATOS RELEVANTES DE SENSOR E+E89.....	53
ANEXO B. DATOS RELEVANTES DATASHEET MSP430F2132.....	62
ANEXO C. DATOS RELEVANTES DATASHEET REGULADOR MAX 1797.	64
ANEXO D. DATOS RELEVANTES DATASHEET ETRX2 Y KIT DE DESARROLLO ETRXDVKA.....	68
ANEXO E. COMUNICACIÓN I²C [8] Y UART	77
ANEXO F. DATOS RELEVANTES TRANSISTOR BSS123 UTILIZADOS EN ADAPTACIÓN DE NIVEL BUS I²C.....	85
ANEXO G. ENTORNO DE DESARROLLO, PLACA DE PRUEBAS MSP430FG4618/F2013 Y HERRAMIENTA DE DEPURACIÓN MSP-FET430UIF.....	88
ANEXO H. CAPTURAS DURANTE TRANSMISIÓN DE DATOS VARIANDO RESISTENCIAS DE PULL-UP EN BUS I²C	93
ANEXO I. CAPTURAS DURANTE LA TRANSMISIÓN ZIGBEE	95
ANEXO J LISTADO DE COMPONENTES DEL NODO SENSOR Y PRECIOS [23].....	99

ÍNDICE FIGURAS

Fig. 0.1 Diagrama de bloques del sensor autónomo.....	2
Fig. 1.1 Byte de control.....	6
Fig. 1.2 Formato de trama de datos completa.....	7
Fig. 1.3 Nivel físico I ² C	8
Fig. 1.4 Adaptación de tensiones en bus I ² C	9
Fig. 1.5 Nivel bajo en transmisión del maestro sección 1.....	11
Fig. 2.1 Nomenclatura para las diferentes familias y modelos de microcontroladores MSP430.....	13
Fig.2.2 Diagrama de bloques funcionales.....	16
Fig. 2.3 Esquemático jumpers conectado a entradas digitales del microcontrolador.....	18
Fig. 2.4 Diagrama de bloques del código del programa.....	23
Fig.3.1 Formato órdenes AT.....	26
Fig. 3.2 Orden para transmisión de datos al SINK.....	26
Fig. 3.3 Pines interrupciones módulo ETRX2.....	27
Fig. 4.1 Diseño en Diseño del nodo sensor en la placa de pruebas	31
Fig. 4.2 Diagrama del sistema de alimentación.....	32
Fig. 4.3 Captura del consumo del sensor durante medición.....	33
Fig 4.4 Captura del consumo del sensor cuando no realiza medición.....	33
Fig. 4.5 Consumo durante medida.....	35
Fig. 4.6 Consumo transceptor durante transmisión.....	36
Fig. 4.7 Gráfica de eficiencia según carga de corriente[16].....	37
Fig. 4.8 Tiempo transmisión I ² C	35
Fig. 4.9 Evolución temporal de activaciones en modo 1 alto consumo.....	39
Fig. 4.10 Evolución temporal de activaciones en modo 2 bajo consumo.....	40
Fig. 5.2 Diseño final en PCB	42
Fig. 5.1 Esquemático nodo sensor	43
Fig. 5.3 Diagrama de bloques del sistema de validación.....	44
Fig. 5.4 Interfaz gráfica LabVIEW.....	44
Fig. E.1 Condición de inicio y condición de fin.....	75
Fig. E.2 Transmisión completa datos en bus I ² C.....	76
Fig. G.1 Placa de experimentación MSP430F4618/F2013 [17].....	86
Fig. G.2 Diagrama bloques placa pruebas.....	88
Fig. G.3 MSP-FET430 UIF.....	89
Fig. G.4 Conexión JTAG.....	89
Fig. G.5 Interfaz aplicación Visual Basic 6.....	90
Fig. H.1 Transmisión I ² C con resistencia pull-up 3,3 kΩ.....	91
Fig. H.2 Transmisión I ² C con resistencia pull-up 10 kΩ.....	91
Fig. H.3 Transmisión I ² C resistencias pull-up de 20 kΩ.....	92
Fig H.4 Señal con resistencias pull-up de 100 kΩ.....	92
Fig. I.1 Tiempo de actividad en transmisión Zigbee.....	93
Fig. I.2 Ancho pico 1.....	93
Fig. I.3 Ancho pulso 2.....	94
Fig. I.4 Ancho pulso 3.....	94
Fig. I.5 Ancho pulso 4.....	94
Fig. I.6 Ancho pulso 5.....	95
Fig. I.7 Detalle activación modo bajo consumo.....	95

Fig. I.8 Perfil de consumo de transceptor sin transmisión en modo activo.....96

INTRODUCCIÓN

Dentro del proyecto para la medida de parámetros ambientales en el Campus del Baix Llobregat mediante sensores autónomos (REALnet) se implementa un nuevo nodo con capacidad para ser integrado dentro de esta red y que permite la medición de CO₂. El sensor utilizado, a diferencia de la mayoría de proyectos anteriores [1], [2] es del tipo NDIR. Esta tecnología tiene mayor precisión en las medidas que los sensores electroquímicos. En este proyecto se realiza la implementación del nodo pero no se integra dentro de REALnet. Se habilita una interfaz que permita la modificación del código del microcontrolador para su integración en futuros proyectos.

La medida de CO₂ puede obedecer a diferentes objetivos, tales como el muestreo en tiempo real de la concentración en una zona o la activación de una alarma en caso de fugas, entre otras aplicaciones. En base a este criterio y teniendo en cuenta que el sensor viene determinado, el nodo desarrollado permite la selección de varios períodos de lectura en dos modos de trabajo. El primer modo de trabajo comprende los períodos de 1 y 5 minutos y el sensor siempre está activo. En el segundo modo, los períodos son de 30 y 120 minutos y el sensor entra en reposo tras realizar una lectura.

Una de las máximas en los sensores integrados en una WSN (Wireless sensor network) es su autonomía derivada de bajos consumos en los componentes que lo forman. Esta ha sido la principal característica tenida en cuenta para el desarrollo del nodo, tanto en su parte hardware, utilizando dispositivos que permitan ser desconectados o dormidos; como en su parte software, desarrollando un código lo más óptimo posible para tener el menor consumo posible. Según el modo de trabajo escogido, el tiempo de actividad variará, aumentando o disminuyendo la vida útil de las baterías. En todos los casos, se presenta un estudio del consumo del nodo.

El sensor está compuesto por diferentes elementos hardware (Fig. 0.1) que se explican en los diferentes capítulos. En el primer capítulo se detalla el sensor utilizado, sus características eléctricas y su comportamiento. En el segundo capítulo se entra en el detalle del microcontrolador, explicando la elección dentro de la familia de microcontroladores de bajo consumo MSP430. También se detallará el código utilizado. En el tercer capítulo se entra de lleno en el transceptor Zigbee, que permite la comunicación con un dispositivo punto a punto o la posterior integración en una red de sensores como REALnet. Se detallan las características principales del protocolo Zigbee y del transceptor. Finalmente se detalla la configuración utilizada.

En el cuarto capítulo se calcula de forma teórica los consumos de cada elemento que forma el nodo. Los datos utilizados han sido extraídos durante la caracterización del consumo. Finalmente se realiza una valoración del tiempo de vida con unas baterías convencionales como son las pilas alcalinas AA. En el quinto capítulo, se muestra el esquemático y la PCB realizada. Se explica el sistema de validación utilizado. En el último capítulo, se realizan las conclusiones, posibles mejoras futuras e impacto medioambiental.

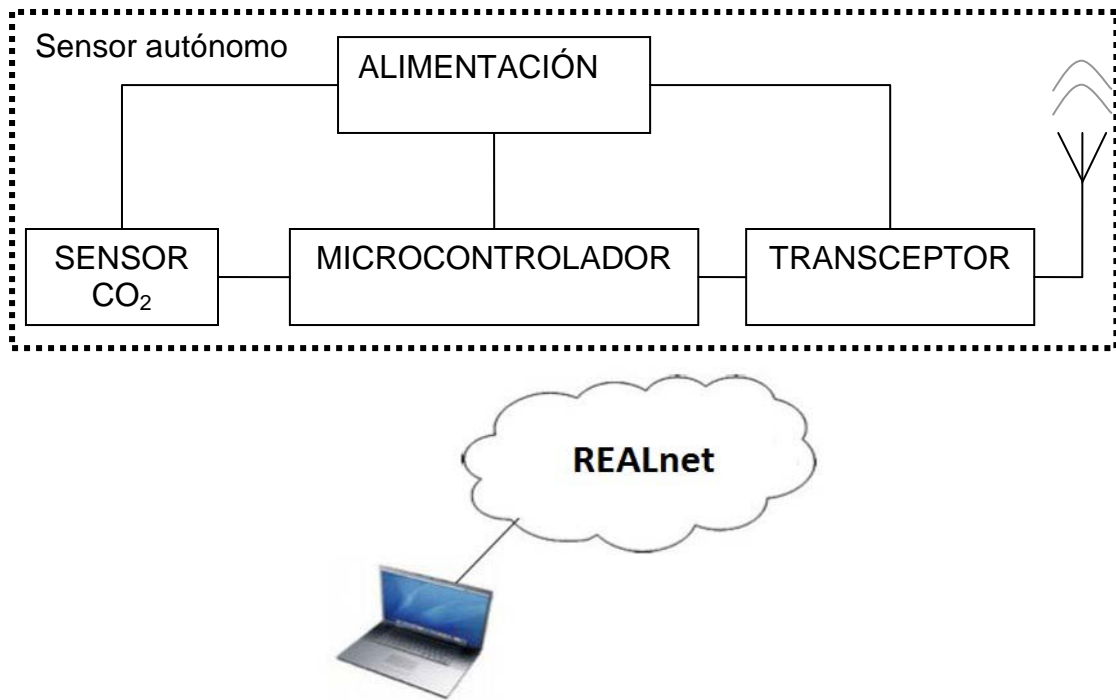


Fig. 0.1 Diagrama de bloques del sensor autónomo

CAPÍTULO 1. EL SENSOR

1.1. Introducción

El sensor utilizado en este proyecto es el modelo EE89 del fabricante E+E Elektronik. El sensor es de tipo NDIR, del acrónimo en inglés “non-dispersive infrared”. El sensor viene dado en el proyecto, por lo tanto no se realiza ningún estudio para la selección del dispositivo ni comparativa alguna con otros modelos existentes en el mercado.

A modo introductorio se explica la tecnología NDIR y los sensores electroquímicos. También se presentan las consecuencias que se pueden producir para la salud humana con diferentes concentraciones de dióxido de carbono.

Así mismo, se detallan en profundidad las características técnicas y de funcionamiento del dispositivo ya que han sido determinantes en la lógica de funcionamiento del nodo.

1.1.1. El dióxido de carbono[3]

El dióxido de carbono es un gas incoloro e inodoro que se encuentra en una concentración aproximada de 350 ppm en la troposfera (capa de la atmósfera más cercana a la superficie de la Tierra). El CO₂ es emitido a la atmósfera principalmente en la combustión y descomposición de material carbónico como el gas natural, el petróleo, el carbón, la madera y sus derivados. También los seres humanos y animales al expirar emitimos CO₂, incluso los excrementos producidos por los animales en las explotaciones ganaderas lo producen.

Actualmente existe una gran concienciación sobre los perjuicios que produce en el clima el aumento de CO₂ causando el fenómeno de efecto invernadero [4] y provocando el cambio climático, es por ello que han tomado mucha relevancia los sistemas para medir las emisiones producidas por las industrias, vehículos, plantas de generación de electricidad, etcétera. No solo en la afectación en la atmósfera se tiene hoy en día en cuenta el CO₂, los sistemas de climatización en muchas ocasiones incorporan sensores de dióxido de carbono con la finalidad de renovar el aire de las estancias donde se encuentran instalados y disminuir la concentración.

La concentración del dióxido de carbono se mide en partes por millón (moléculas de CO₂ por millón de moléculas). La tabla 1.1 detalla cuales serían las consecuencias que producirían en un ser humano diferentes concentraciones en un recinto cerrado.

Tabla 1.1 Concentración CO₂. Modificado de [1], [2]

Concentración CO ₂ (ppm)	Consecuencia
150000	Provoca la muerte
60000 a 80000	Aparición de parálisis
~40000	Proporción aproximada en el aliento durante una expiración
30000	Problemas de respiración incipiente
15000	Ocasiona dolor de cabeza e incluso hiperventilación
5000	Límite de concentración de CO ₂ en un puesto de trabajo
4000	Límite de concentración de CO ₂ en un colegio
Superior a 1000	Produce fatiga y reducción en la concentración
1000	Nivel máximo recomendado en colegios
800	Nivel máximo recomendado en oficinas
350	Concentración aproximada en la atmósfera

1.1.2. Tipos de sensores de CO₂

Existen varios tipos de sensores de CO₂, los más utilizados son los NDIR y los electroquímicos. Este tipo de sensores pueden ser utilizados para la medición de diferentes tipos de gases. Para cada tipo, tendrá unos filtros específicos en el caso del NDIR y un material de electrodo concreto en el caso de los electrolíticos.

- **Sensores NDIR:** Es capaz de medir la concentración de un gas en el interior de un tubo de muestra mediante una fuente de infrarrojo, un filtro de longitud de onda y un detector de infrarrojos. Se basa en el principio de la absorción de cierta longitud de onda por parte de las moléculas del gas. La luz infrarroja se difunde en el tubo de muestra hasta llegar al detector. El detector tiene un filtro óptico frente a él que elimina toda la luz excepto la procedente de la longitud de onda que absorben las moléculas del gas medido. Según la luz que reciba el detector se determinará la concentración del gas .[5]
- **Sensores electroquímicos:** Se basan en el principio de la reacción química producida al exponer el sensor a un gas determinado. El sensor esta compuesto por un electrodo sensor (cátodo) y un contra electrodo (ánodo) separados por un electrolito. El gas, al entrar en contacto con los electrodos, produce una reacción de oxidación o reducción. Los materiales que componen los electrodos catalizan esta reacción y se genera una tensión proporcional a la concentración de gas [1].

1.2. Características del sensor utilizado

El sensor EE89 realiza la medición del CO₂ mediante la tecnología NDIR y transmite los datos digitalmente a través de una interfaz I²C. Esta interfaz es la

utilizada para comunicarse con el microcontrolador. En la tabla 1.2 están las principales características del sensor.

Tabla 1.2 Características del sensor EE89 [6]

Alimentación	4,75 – 7,5 VDC
Consumo	Típica 0,7 mA; Máximo 0,5 A durante 0.3s
Margen de medida	0- 2000
Exactitud	a 25°C y 1013mbar $<\pm 50$ ppm + 3% de la medida
Tiempo de calentamiento	< 5 min
Tiempo de muestreo	15 segundos aproximadamente
Tiempo de respuesta	< 195 segundos

La alimentación del sensor varia entre 4,75 y 7,5 VDC teniendo un consumo típico de 7 mA. El fabricante explicita que hay un consumo máximo de 0,5 A durante 0,3 segundos. Más adelante, en el capítulo de consumo se explica en detalle el perfil de consumo del sensor.

El margen de medida del sensor utilizado es de 2000 ppm. Otros modelos del mismo fabricante pueden llegar hasta 10000 ppm manteniendo similares características. El margen del sensor utilizado en el proyecto es una limitación según la aplicación a la cual sea destinada el sensor, pero como el objetivo principal de este nodo es la medición en el exterior, con este margen es suficiente. El diseño del nodo permite acoplar un sensor con mayor margen sin necesidad de cambiar ni el hardware ni el software.

Se entiende como tiempo de calentamiento, el tiempo hasta que los elementos internos del sensor estén en las condiciones óptimas para realizar una lectura. El tiempo de calentamiento determinado por el fabricante es menor de 5 minutos. Cualquier lectura realizada antes de este tiempo puede ser incorrecta.

El tiempo de muestreo se refiere al tiempo interno entre mediciones, siendo independiente de cuando el microcontrolador consulta el valor. Como se puede ver en el capítulo 4, en el consumo del sensor se observa un aumento cada 15 segundos asociado a la medición.

El tiempo de respuesta esta referido a un cambio en la magnitud física. Este tiempo, según el fabricante es de menor a los 195 segundos.

1.3. Comunicación

El sensor tiene una interfaz que permite la comunicación I²C, pero el protocolo que implementa es una variante basada en sus especificaciones, diseñado por el fabricante y denominado "E2 interface" [7]. En el anexo A esta la especificación completa, pero se detallarán los puntos más importantes.

Dicho protocolo mantiene todas las especificaciones a nivel físico de I²C variando en el nivel de enlace y en el nivel de red (protocolo). Otra

característica de E2 es que el esclavo no puede enviar datos de forma independiente y no hay posibilidad de direccionamiento, siendo obligatoria la comunicación punto a punto entre sensor y microcontrolador.

A nivel físico se requiere una línea de datos y otra línea de reloj conectadas a resistencias de pull-up. Ambas líneas tienen que estar configuradas como colector/drenador abierto. La capa de enlace define la forma en la que se accede al bus y las operaciones básicas, tales como condición de inicio, condición de fin, transmisión de datos y reconocimientos. En el anexo E se detallan las operaciones básicas. En I²C, después de una condición de inicio el maestro envía el byte de dirección. En E2, este byte tiene que tener el formato de la figura 1.1 y se denomina "byte de control". A diferencia de I²C, en E2 no hay direccionamiento ya que es obligatorio enviar los bits 1, 2, 3 a 0. En este byte el maestro especifica la orden que debe realizar el sensor. En la tabla 1.3 podemos ver las órdenes.

Control Byte							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Main Command				Device Address			R/W

Fig. 1.1 Byte de control

El bit 0 se corresponde al sentido de la transmisión, en nuestro caso el maestro ordena al sensor que devuelva el valor de la medida y por tanto R/W siempre es 1. Los bit 1, 2 y 3 siempre se envían a 0. Los bits 4, 5, 6 y 7 se utilizan para las órdenes.

Tabla 1.3 Órdenes E2

Byte de control	Número de bit				Tipo de orden
	7	6	5	4	
0x11	0	0	0	1	Tipo sensor (grupo)
0x21	0	0	1	0	Tipo sensor (subgrupo)
0x31	0	0	1	1	Medidas disponibles
0x41	0	1	0	0	Reservado
0x51	0	1	0	1	Reservado
0x61	0	1	1	0	Reservado
0x71	0	1	1	1	Byte de estado
0x81	1	0	0	0	1 byte bajo medida 1
0x91	1	0	0	1	1 byte alto medida 1
0xA1	1	0	1	0	1 byte bajo medida 2
0xB1	1	0	1	1	1 byte alto medida 2
0xC1	1	1	0	0	1 byte bajo medida 3
0xD1	1	1	0	1	1 byte alto medida 3
0xE1	1	1	1	0	1 byte bajo medida 4
0xF1	1	1	1	1	1 byte alto medida 4

Las tres primeras órdenes permiten consultar el tipo de sensor y las magnitudes físicas disponibles en el sensor. El fabricante ha reservado 3

órdenes para uso interno o futuro. La orden 0x71 permite al programador verificar si los datos leídos con anterioridad han sido correctos y se denomina byte de estado. No es obligatorio en la comunicación, simplemente permite la consulta. Las órdenes desde 0x81 a 0xF1 devuelven el valor leído de cada magnitud física. En la tabla 1.4 se detallan los valores de cada orden en el sensor utilizado.

Tabla 1.4 Respuesta a órdenes en el sensor EE89 [8]

command:	return-value	kind	format	measuring-range	output:
Group	0x89				
Sub-Group	0x09				
Available measuring-values	0x08				
Statusbyte:	0x0x				
Measuring value 1:		Not defined			
Measuring value 2:		Not defined			
Measuring value 3:		Not defined			
Measuring value 4:		CO2	unsigned int.	0 – 2000 or 0 – 5000	ppm

En todos los casos, el valor medido se transmite en 2 bytes, enviándose dos tramas con datos y checksum como la que se observa en la figura 1.2. Para la lectura de CO₂ esta reservada la medida 4, por tanto se utilizan las órdenes 0xE1 y 0xF1. El formato del dato transmitido es un entero sin signo, que son 32 bits, 16 de los cuales se utilizan para datos y 16 para checksum en 2 tramas.

Estas órdenes son específicas del nivel de red de E2 y tienen un formato de trama concreto tal y como se muestra en la figura 1.2. La diferencia básica reside en que en I²C el segundo byte no es obligatorio y puede ser un dato. En el caso de E2, es obligatorio y se corresponde con el checksum de los datos enviados en la misma trama.

Start	Control byte									Data byte									Checksum									Stop
	7	6	5	4	3	2	1	0	ACK	7	6	5	4	3	2	1	0	ACK	7	6	5	4	3	2	1	0	ACK	
Start								1	A	x	x	x	x	x	x	x	x	A	x	x	x	x	x	x	x	x	NA	Stop

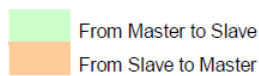


Fig. 1.2 Formato de trama de datos completa

En el anexo E se detallan los aspectos más importantes de I²C.

1.3.1. Comunicación con el microcontrolador

La figura 1.3 muestra un esquema de comunicación a nivel físico entre dispositivos en un bus I²C.

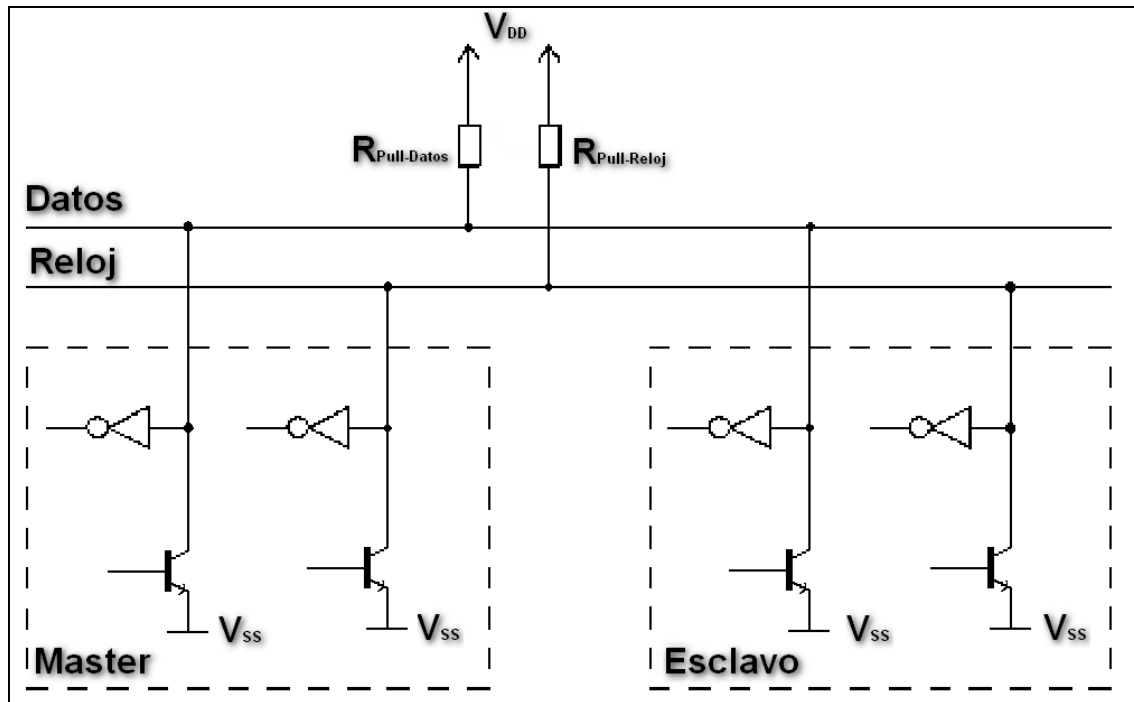


Fig. 1.3 Nivel físico I²C

En nuestro caso, en la comunicación del sensor con el microcontrolador existe un problema de alimentaciones, ya que el sensor puede trabajar entre 4,5 y 7,5 VDC y el microcontrolador, como se verá en el siguiente capítulo, lo hace entre 1,8 y 3,6 VDC.

Para la correcta comunicación en el bus I²C, las líneas no pueden trabajar a diferentes tensiones. No es posible trabajar a 5 VDC, ya que los puertos del microcontrolador no lo soportan y tampoco se puede trabajar a 3 VDC porque el sensor para detectar un nivel alto tiene que recibir como mínimo el $0,8 \times V_{DD} = 4V$. En la tabla 1.5 podemos ver los valores máximos y mínimos que requiere el sensor para detectar un nivel alto y nivel bajo.

Tabla 1.5 Valores de detección de niveles en sensor [7]

Parámetro	Mínimo	Máximo
Nivel alto en entrada [V]	$0,8 V_{DD}$	$V_{DD} + 0,3$
Nivel bajo en entrada[V]	$V_{SS} - 0,3$	$0,2 \times V_{DD}$ ó $0,8 V$ (se utiliza el valor mínimo de estos dos)

Los niveles de entrada (detección) del puerto I²C del microcontrolador se muestran en la tabla 1.6.

Tabla 1.6 Valores de detección de niveles en microcontrolador [9]

Parámetro	Mínimo	Máximo
Nivel en entrada [V]	1,35	2,25

Para resolver el problema de los diferentes niveles de detección y poder comunicar el microcontrolador y el sensor, se realiza una adaptación de niveles mediante transistores MOSFET de canal n enriquecido. El transistor seleccionado es el BSS123 de Fairchild. En el anexo F están los detalles del transistor y los motivos de su elección. Podemos ver en la figura 1.4 como quedaría el bus con la adaptación. La sección 1 se corresponde con el maestro y la sección 2 con el esclavo.

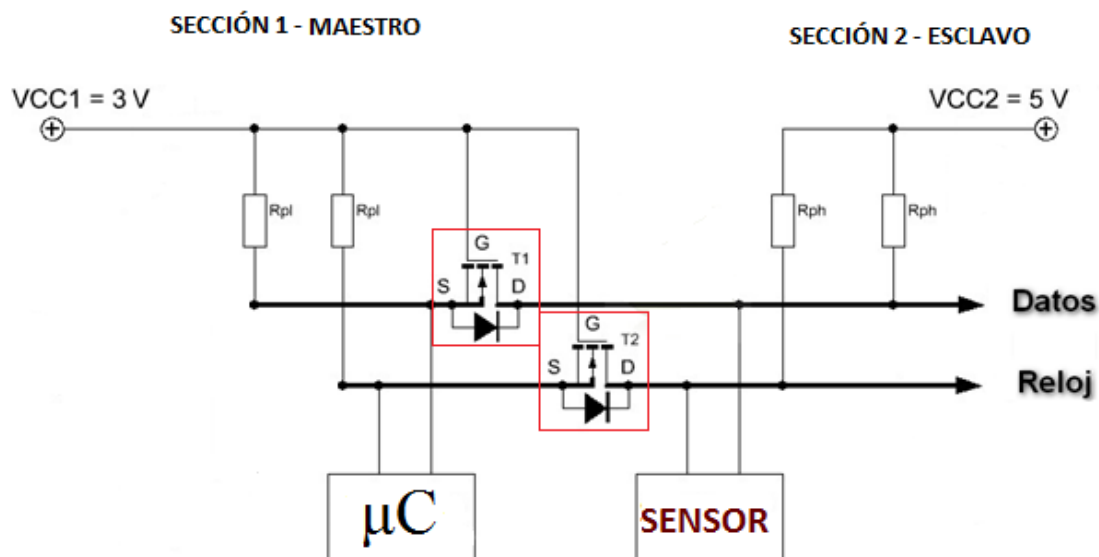


Fig. 1.4 Adaptación de tensiones en bus I²C. Modificado de [10]

Este tipo de transistores actúan como interruptores lógicos activados por tensión positiva superior a 0V. Cuando se produce una diferencia de tensión concreta entre la puerta (G) y la fuente (S), denominada tensión umbral (como se observa en el Anexo F, con el transistor escogido es de 0,8 a 2 V) el transistor entra en saturación y se produce continuidad entre la fuente (S) y el drenador (D), en caso contrario la impedancia es muy alta y no hay continuidad entre fuente y drenador.

El funcionamiento de este interruptor lógico permite separar las alimentaciones en cada parte del bus manteniendo las tensiones de referencia en cada lado. El transistor en este montaje tiene tres estados posibles [10]:

- El primer estado posible se da cuando la línea se encuentra a nivel alto (ningún dispositivo transmite) y no hay diferencia de potencial entre puerta (G) y fuente (S). En este caso el diodo de protección que incorpora el transistor impide la conducción entre las secciones de tensión alta (esclavo) hasta las zonas de baja tensión (maestro), dando protección a los dispositivos de baja tensión.
- El segundo estado posible es cuando el maestro (sección 1) activa la línea a nivel bajo. En este caso hay una diferencia de potencial de 3V entre la puerta (G) y la fuente (S), $G=3\text{ V}$ y $S=0\text{ V}$, produciendo el

cambio de estado del transistor e iniciando la conductividad entre la fuente (S) y el drenador (D). En este estado la sección 2, será inducida a nivel bajo debido a la conductividad del transistor. Por lo tanto ambas secciones se encuentran a nivel bajo a la misma tensión.

- El tercer estado posible se produce cuando el esclavo (sección 2) activa la línea a nivel bajo. En primer lugar, el diodo interno entrará en conducción desde la línea del maestro a la línea del esclavo, de esta forma la fuente (S) pasa a nivel bajo inducido por el diodo, produciéndose de nuevo una diferencia de tensión entre la puerta (G) y la fuente (S) y comenzando la conducción a través del mismo. Se asegura el nivel bajo en ambas secciones del bus. Cuando es polarizado correctamente, el transistor se comporta como un resistor variable entre el drenador (D) y la fuente (S).

Para las resistencias de pull-up, en el caso del nodo implementado, se han utilizado unos valores de resistencias obtenidos tras varias pruebas, ya que el rango de valores permitido por el sensor es de 1 a 100 k Ω [7] y por el microcontrolador de 20 a 50 k Ω [9].

Se detalla en la tabla 1.7 los diferentes valores de resistencias probados y los resultados obtenidos. La comparativa muestra las tensiones a nivel alto y bajo en la línea de datos. Para el nivel bajo se distinguen dos casos según quien actúa en la línea. La línea de reloj es siempre inducida a nivel bajo por el maestro, por tanto los valores se corresponden con los obtenidos en la activación del maestro.

Tabla 1.7 Resistencias de pull-up probadas

Resistencia pull-up [k Ω]	Nivel bajo. Activación Maestro: Sec.1/Sec. 2 [mV]	Nivel bajo. Activación Esclavo: Sec.1/Sec. 2 [mV]	Nivel alto. Sec.1/Sec. 2 [V]
3,3	40/100	1560/1520	2,86/5,16
10	0/35	600/660	2,70/5,08
20	0/80	360/400	2,68/3,96
43	35/40	192/192	2,72/3,84
100	0/22	108/110	2,40/3,68

En el anexo H se muestran las capturas obtenidas en cada una de las pruebas con los diferentes valores de resistencias. La alimentación de la sección 1 (maestro) durante las pruebas ha sido de 2,8 V

Hay una relación directa entre los valores de resistencia y el nivel de voltaje de la señal. Cuando más pequeño es el valor de las resistencias de pull-up, menor atenuación hay en los niveles altos. En los niveles bajos, hay dos comportamientos diferenciados según quien transmite:

- Durante la transmisión del maestro los valores del nivel bajo son 0 o muy cercanos, aunque en algún caso ha apreciado valores negativos en algunas pruebas. La explicación de esta situación no se ha estudiado en profundidad, pero se observa que la señal no es completamente

estable ya que en la transmisión de un mismo bit adquiere varios valores de tensión. Esto podría ser debido a un problema en la calibración del osciloscopio o por la caída de tensión introducida por el diodo interno del transistor. No se indaga más en este fenómeno debido a que la transmisión funciona correctamente y se determina que los niveles bajos en estos casos son 0. En la figura 1.5 se muestra un caso, con una resistencia de 20 k Ω donde se produce un nivel bajo por debajo de 0 cuando transmite el maestro. A parte de esta situación, los valores son bastante uniformes y no se ven afectados por el cambio del valor de la resistencia.

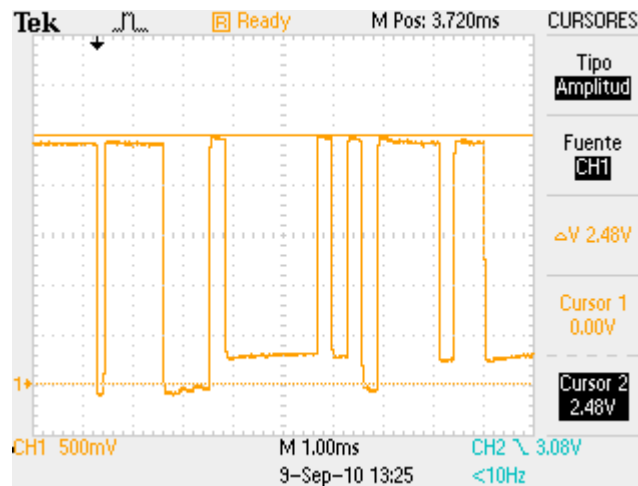


Fig. 1.5 Nivel bajo en transmisión del maestro (sección 1)

- Cuando transmite el esclavo, el nivel bajo se encuentra por encima del nivel bajo cuando transmite el maestro. Con la resistencia más pequeña el nivel bajo está más lejos de 0, a un voltaje de 1,5V.

En conclusión:

- Resistencia pequeña → Problema en niveles bajos.
- Resistencia grande → Problema en niveles altos.

El valor escogido es de 20 k Ω siendo el valor de resistencia con el cual los niveles altos y bajos están más cerca de los recomendados por el fabricante del sensor (tabla 1.6)

CAPÍTULO 2. EL MICROCONTROLADOR

2.1. Introducción

El microcontrolador es el elemento principal en el nodo, contiene la lógica de comunicación con el sensor y el transceptor, habilita y deshabilita los modos de bajo consumo y desconecta el regulador de forma que el sensor queda desactivado. Con dos de sus entradas digitales se puede configurar el modo de trabajo del sensor. Con todas estas acciones se pretende minimizar el consumo del nodo y alargar la vida útil de las baterías al máximo.

Se detalla en este capítulo los criterios escogidos para la selección del microcontrolador, las características más importantes y el código implementado.

2.1.3. Selección del microcontrolador

Un requisito del proyecto es trabajar con un microcontrolador de la familia MSP430 (Texas Instruments) por la gran variedad de microcontroladores disponibles y por tener varios modos de bajo consumo.

Dentro de la familia existen decenas de microcontroladores, la mayoría de 16b y arquitectura RISC. Las premisas básicas para la elección del microcontrolador es que tuviese una interfaz de comunicación I²C y UART, para poder comunicar con los periféricos.

Antes de la elección del microcontrolador, se ha realizado un estudio previo donde se han desarrollado varios programas de prueba con una placa de experimentación para conocer en profundidad el funcionamiento del microcontrolador y determinar todas las necesidades. En el anexo G se detallan las características del kit de desarrollo y las funcionalidades del código desarrollado.

Tras el estudio previo se determina que las necesidades completas del microcontrolador son las siguientes:

- Puerto UART
- Puerto I²C
- 1 Timer con posibilidad de realizar interrupción
- Watchdog
- 2 entradas digitales
- 4 salidas digitales
- Modos de bajo consumo
- Bajo coste

2.1.3.1. Comparativa microcontroladores

Tras el análisis de las necesidades se realiza la búsqueda de un microcontrolador que posea todas las características comentadas en el apartado anterior. La nomenclatura utilizada para catalogar las diferentes familias y modelos de microcontroladores MSP430 es la que se muestra en la figura 2.1.

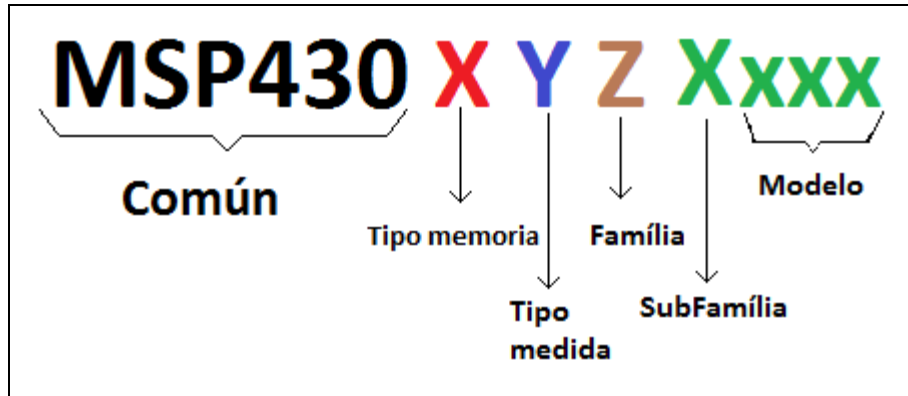


Fig. 2.1 Nomenclatura para las diferentes familias y modelos de microcontroladores MSP430

El primer dígito tras MSP430 indica el tipo de memoria que utiliza. Puede ser F para Flash y C para ROM. El segundo dígito no está presente en todos los modelos y nos indica el tipo de medida para el cual está diseñado. Puede ser E para electricidad, W para agua y G para señales que necesitan un nivel de ganancia proporcionada por amplificadores operacionales. El tercer dígito determina la familia y el quinto a la subfamilia a la cual pertenece. En las últimas posiciones, podemos encontrar 2 o 3 dígitos que determinan el dispositivo específico.

A continuación se detallan las características generales de las diferentes familias [11].

- MSP430x1xxx: Con una CPU de 8 MHz proporciona un amplio rango de dispositivos para uso general. Incluyen múltiples entradas y salidas de uso general para conexión de periféricos e incluso algunos multiplicadores hardware que pueden ser usados como rudimentarios procesadores de señal digital.
- MSP430x2xx: La CPU puede trabajar a 16 MHz, el doble de velocidad que sus predecesores teniendo un consumo de solo la mitad a la misma velocidad. Incluye algunos modelos con encapsulado del tipo PDIP permitiendo realizar circuitos en placa de pruebas. No requiere de la conexión de un cristal externo para trabajar con bajas frecuencias ya que incorpora un oscilador interno de muy bajo consumo (VLO). Incluye en sus salidas resistencias de pull-up y pull-down reduciendo los componentes externos. Tiene múltiples opciones para entradas analógicas.
- MSP430x3xx: La familia original, con CPU de 8MHz. Incluye driver para LCD. Actualmente es obsoleto.
- MSP430x4xx: Puede trabajar con LCD de hasta 160 segmentos.

- MSP430X: La arquitectura original de MSP430 ha sido extendida en 2006 con la versión X. Tiene más memoria para direcciones. No se vende como una familia separada, sus dispositivos son incluidos en las familias MSP430x2xx y MSP430x4xx sin un número que los distinga. Serán de esta familia las que tengan más de 64KB de memoria.
- MSP430x5xxx/6xxx: La CPU puede trabajar a 20/25 MHz incluyendo nuevos modos de gestión del consumo. Integra multitud de periféricos como USB, comparador analógico, DMA, multiplicador Hardware, RTC, etc.

Teniendo en cuenta que la placa de pruebas integra el microcontrolador MSP430FG4618 y para reutilizar parte del código implementado se decide usar un microcontrolador de 16 MHz. Por tanto, se debe escoger entre la familia 2 y 4. Lo más sencillo hubiese sido continuar con el mismo microcontrolador, pero el encapsulado es del tipo TQFP y esto complica mucho la integración en el laboratorio con las herramientas disponibles. Además la familia 4 permite conectar un LCD y multitud de periféricos que para este proyecto no son necesarios. Por estos motivos se decide buscar entre microcontroladores más sencillos de la familia 2.

Dentro de la familia 2, existen 6 subfamilias con características muy variadas. Se detalla en la tabla 2.1 las principales características de 5 dispositivos preseleccionados, uno de cada subfamilia. La sexta subfamilia se descarta por su elevado precio y características sobredimensionadas. Se debe tener en cuenta que dentro de cada subfamilia el tamaño de la memoria RAM y Flash varía. Como el código es ligero, es suficiente con trabajar con memorias del orden 8 KBytes para flash y 256 Bytes para RAM.

El precio varía en función del tipo de encapsulado y se corresponde con el precio para mil unidades.

Tabla 2.1 Comparativa de 5 microcontroladores de la familia 2 [11]

	MSP430F2003	MSP430F2132	MSP430F2234	MSP430F2330	MSP430F2410
Memoria RAM/Flash	128B/1KB	512B/8KB	512B/8KB	1KB/8KB	4KB/56KB
Puertos	USI: - UART - SPI I ² C	USCI A: - UART - IrDA - SPI USCI B: - I ² C - SPI	USCI A: - UART - IrDA - SPI USCI B: - I ² C - SPI	USCI A: - UART - IrDA - SPI USCI B: - I ² C - SPI	USCI A: - UART - IrDA - SPI USCI B: - I ² C - SPI
Temporizadores	Watchdog, Basic clock, TimerA, DCO	Watchdog, Basic clock, TimerA_0, TimerA_1, D CO	Watchdog, Basic clock, TimerA, TimerB, DCO	Watchdog, Basic clock, TimerA, Timer B, DCO	Watchdog, Basic clock, TimerA, TimerB, DCO
I/O	P1 / P2	P1/P2/P3	P1/P2/P3/P4	P1/P2/P3/P4	P1/P2/P3/P4/P 5/P6
Extras		ADC 10 bits	ADC 10 bits, 2 Amplificador	Comparador analógico,	Multiplicador hardware, ADC

			operacional	Multiplicador hardware	12 Bits
Optimizado para:		Sensores inalámbricos			Comunicación Wireless
Modos de bajo consumo	5	5	5	5	5
Alimentación	1,8 – 3,6 V	1,8 – 3,6 V	1,8 – 3,6 V	1,8 – 3,6 V	1,8 – 3,6 V
Encapsulado	14TSSOP/ 14DIP	28TSSOP /32QFN	38TSSOP/ 40QFN	49QFN/ 49DSBGA	64QFN/ 64QFP
Precio[\$]	1,10	1,50	1,80	1,75	4,60

La subfamilia MSP430x20xx se descarta porque solo incorpora un puerto de comunicación serie para conectar periféricos ya sea UART o I²C, incumpliendo una necesidad básica.

Las otras cuatro subfamilias si que incorporan 2 puertos para comunicación serie y cumplen todos los requisitos. La subfamilia MSP430F24XX esta sobredimensionada porque incorpora hasta 6 puertos de entradas y salidas digitales y un multiplicador hardware. Estas características no tienen uso en este diseño. Finalmente se escoge un dispositivo de la subfamilia MSP430x21x2 porque el propio fabricante indica que esta optimizado para sensores inalámbricos. A parte es el más económico y tiene I/O suficientes por si se quiere añadir algún periférico más. El encapsulado es TSSOP, que facilita la adaptación a la PCB.

Entre el MSP430F2112, el MSP430F2122 y el MSP430F2132 se escoge este último porque tiene más memoria RAM /Flash y el precio es muy similar.

Tabla 2.2 Comparativa entre dispositivos familia 2 subfamilia 2

Dispositivo	Memoria RAM[B]/Flash[KB]	Precio [\$]
MSP430F2112	256/2	1,10
MSP430F2122	512/4	1,40
MSP430F2132	512/8	1,50

2.2. Características MSP430F2132

Se detalla a continuación las características principales del microcontrolador escogido. Se muestra en la figura 2.2 el diagrama de bloques funcionales de los dispositivos MSP430F21x2.

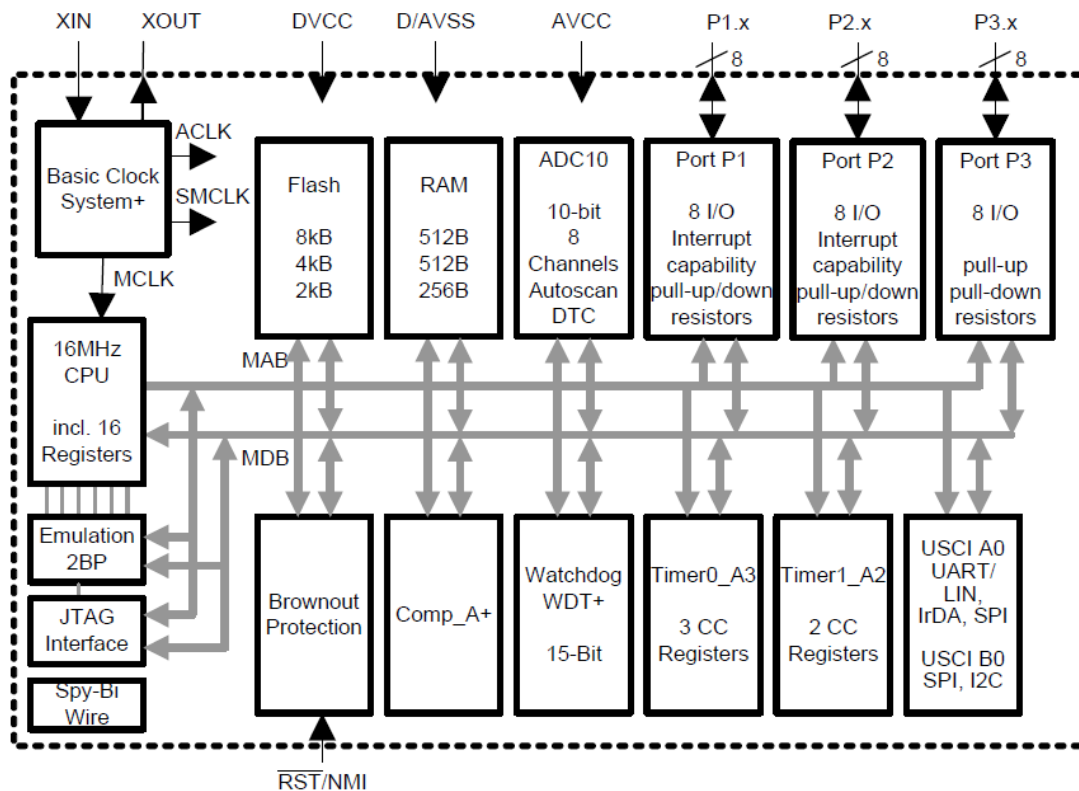


Fig.2.2 Diagrama de bloques funcionales

2.2.3.1. Puertos UART e I²C

Los puertos utilizados para la comunicación serie son del tipo USCI (Universal Serial Communication Interface) debido a que soporta múltiples modos de comunicación serie. En nuestro caso es indispensable, para trabajar simultáneamente con el sensor (I²C) y el transceptor (UART). Cada módulo USCI es identificado con una letra, USCI_A para UART/IrDA/SPI y USCI_B para I²C/SPI. En el caso de I²C el puerto USCI incorpora una máquina de estados hardware que facilita mucho el desarrollo. Ambas características no están disponibles en los puertos del tipo USI.

La máquina de estados hardware puede utilizarse tanto trabajando como maestro o esclavo (en el caso de comunicar 2 microcontroladores por ejemplo). Cuando se utiliza como maestro es capaz de generar las condiciones de inicio sin que el programador tenga que detectar ningún flanco ni realizar ninguna comprobación sobre la disponibilidad del bus. Envía los datos que se introducen en un registro determinado y gestiona el envío y recepción automáticamente de los bits de reconocimiento. Cuando se reciben los datos se almacenan y posteriormente se genera la condición de fin. En la sección 2.3 se proporcionan más detalles.

El puerto USCI_A se utiliza para comunicar el transceptor con el microcontrolador con la configuración para UART. El puerto USCI_B se utiliza para comunicar el microcontrolador con el sensor con la configuración para I²C.

El puerto UART se ha configurado para trabajar de forma asíncrona, a 9600 baudios, sin paridad, 8 bits de datos, 1 bit de stop y señal de reloj ACLK con divisor por 8. Se transmite primero el bit de menor peso (LSB). Los registros configurados para ambos puertos se detallan en el Anexo E.

2.2.3.2. Señales de reloj, fuente de reloj y timers

La familia MSP430 de microcontroladores tiene las siguientes señales de reloj disponibles para su uso [12].

- MCLK: Master clock. Usado por la CPU y algunos periféricos.
- SMCLK: Subsystem Master clock. Normalmente trabaja a la misma frecuencia que MCLK y es usado por algunos periféricos.
- ACLK: Auxiliary clock. Usado por algunos periféricos.

Estas señales de reloj pueden obtener la fuente de reloj de señales externas o osciladores internos.

- LFXT1: *Low or high-frequency crystal oscillator*. Disponible en todos los dispositivos. Normalmente es usada con un cristal externo de 32 KHz pero también puede usarse con cristales de alta frecuencia.
- XT2: *High-frequency crystal oscillator*. Similar a LFXT1 excepto que esta restringida para cristales o resonadores que trabajen a altas frecuencias.
- VLO: *Internal Very low-power, low-frequency oscillator*. Proporciona un LFXT1 alternativo cuando la exactitud de un cristal no es necesaria.
- DCO: *Digital controller oscillator*. Disponible en todos los dispositivos. Es básicamente un oscilador interno controlable por circuitos RC que trabaja a altas frecuencias hasta 1,1 MHz en la familia 2. Se utiliza en aplicaciones que no requieren gran precisión. Es escalable para trabajar a multitud de frecuencias.

Por defecto, la señal del ACLK es obtenida de un cristal externo y MCLK/SMCLK obtienen la señal de DCO con una frecuencia de 1 MHz.

El microcontrolador MSP430F2132 incorpora 2 timers de 16 bits que permiten la captura de 2 o 3 registros según si es el timer 0 o el timer 1. También incorpora un watchdog. En este proyecto se ha utilizado el timer 0, con la señal de reloj ACLK, para realizar las interrupciones y salir del modo de bajo consumo. El watchdog timer es utilizado en caso de entrar en un bucle infinito por ejemplo si no recibe un byte por parte del sensor. Cuando se detecta esta situación, el watchdog timer se activa y vuelve a ejecutar el código desde el principio. El timer 1 no se ha utilizado porque no ha sido necesario.

2.2.3.3. Puertos digitales

Todos los puertos digitales del microcontrolador pueden ser utilizados como entradas o salidas. Cada puerto tiene un total de 8 bits, que están asociados a un pin del microcontrolador, con la siguiente nomenclatura:

- PX.Y
 - P indica que es un puerto digital.
 - X indica el puerto, entre 1 y 3 para MSP430F2132.
 - Y indica el bit.

Las salidas digitales se han utilizado para actuar sobre los periféricos conectados, se han utilizado tres. Dos se conectan directamente al transceptor y activan las interrupciones que permiten entrar o salir de su modos de bajo consumo. La tercera salida digital se conecta al regulador y activa la desconexión (en el capítulo 4 se introducen los datos relacionados con el regulador).

Las entradas digitales nos permiten gestionar señales externas. Se han utilizado 2 entradas digitales conectadas a dos jumpers y con dos resistencias de pull-up que nos permiten seleccionar los modos y períodos de medición del nodo. Cuando el jumper no esta conectado, la línea esta a nivel alto por la resistencia de pull-up, en el momento que se conecta el jumper, pasa a nivel bajo. Se muestra en la figura 2.3 como quedan conectados los jumpers.

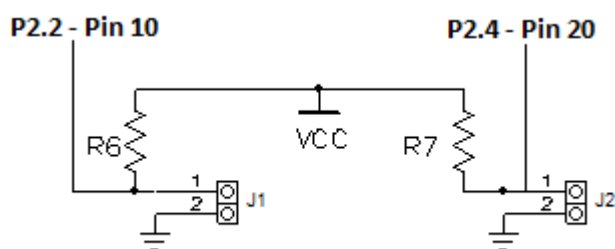


Fig. 2.3 Esquemático jumpers conectado a entradas digitales del microcontrolador

Los puertos para comunicación serie también utilizan un puerto digital. I²C utiliza el puerto 3, bits 2 y 3. UART utiliza el puerto 3, bits 4 y 5.

2.2.3.4. Modos de trabajo. Bajo consumo y activo.

Todos los microcontroladores de la familia MSP430 incorporan 5 modos de bajo consumo que deshabilitan la CPU y las señales de reloj para tener un mayor o menor ahorro energético. En la tabla 2.3 se detallan todos los modos de bajo consumo indicando el estado de cada reloj. LPM es el acrónimo en inglés de *Low Power Mode*.

Tabla 2.3 Detalle señales de reloj en modos de bajo consumo [9]

Modo de bajo consumo	ACLK	SMCLK	MCLK	DCO
Modo 0 LPM0	ON	ON	OFF	ON
Modo 1 LPM	ON	ON	OFF	OFF
Modo 2 LPM2	ON	OFF	OFF	ON
Modo 3 LPM3	ON	OFF	OFF	OFF

Modo 4 LPM4	OFF	OFF	OFF	OFF
-------------	-----	-----	-----	-----

Para salir de los modos de bajo consumo es necesario una interrupción. Esta interrupción puede ser interna (por ejemplo el timer 0 o timer 1) o bien externa (I/O). En este caso, no hay opción a interrupciones externas ya que el sensor trabaja siempre en modo esclavo y solo transmite datos cuando el microcontrolador lo solicita y el transceptor actúa de la misma forma. Por ello, se trabaja con el timer 0 que utiliza la señal ACLK, que realizará las interrupciones. El modo de bajo consumo utilizado es el LPM3 ya que mantiene activo el ACLK y desactiva el resto, teniendo un ahorro energético mayor tal y como se muestra en la tabla 2.4.

Tabla 2.4 Consumo microcontrolador MSP430F2132 [9]

Modo de trabajo	Condiciones	Consumo típico	Consumo máximo	Unidades
LPM0 (1 MHz)	F _{DCO} =1MHz F _{ACLK} = 32 kHz	-	83	µA
LPM0 (100 kHz)	F _{DCO} =100kHz F _{ACLK} = 0 Hz	-	46	µA
LPM2	F _{DCO} =1MHz F _{ACLK} = 32 kHz	-	27	µA
LPM3 Reloj externo	F _{DCO} =0Hz F _{ACLK} = 32 kHz	0,9	1,9	µA
LPM3 oscilador interno	F _{DCO} =0 Hz F _{ACLK} = oscilador interno baja frecuencia VLO	0,7	0,8	µA
LPM4	F _{DCO} =0 Hz F _{ACLK} = 0 Hz	0,1	0,5	µA

La CPU utiliza el MCLK que obtiene la señal de reloj del DCO. El ACLK se utiliza en las comunicaciones con los periféricos por los puertos UART e I²C y para el timer 0, que obtiene la señal de reloj de un cristal externo de 32KHz. En el caso del timer 0, se utiliza el divisor por 8, obteniendo una frecuencia de trabajo de 4 KHz que permite introducir retardos mayores en la ejecución de los tiempos de espera sin necesidad de despertar el microcontrolador. Podemos mantener parado el microcontrolador durante 16 segundos continuados. Para períodos de inactividad más grandes, el microcontrolador encadena reposos de 16 segundos hasta llegar al tiempo marcado, activándose y desactivándose tras 16 segundos. Otras familias de microcontroladores tiene una fuente de reloj denominada *Real time clock* que permite mantener en reposo el microcontrolador hasta pasado un tiempo establecido en segundos.

En la tabla 2.5 se muestra el consumo en función de la señal de reloj escogida. El consumo también varía en función de la temperatura de operación y el voltaje de alimentación [9]. En todos los casos será 25 °C y 3 VDC respectivamente.

Tabla 2.5 Consumo microcontrolador MSP430F2132 en modo activo según fuente de reloj [9]

Parámetro	Condiciones	Consumo típico	Consumo máximo	Unidades
Modo activo (1 MHz)	f _{DCO} =1Mhz f _{ACLK} =32768Hz Programa ejecutado en flash	350	450	μA
Modo activo (1 MHz)	f _{DCO} =1Mhz f _{ACLK} =32768Hz Programa ejecutado en RAM	220	300	μA
Modo activo (4 kHz)	f _{DCO} =0 Hz f _{ACLK} =32768Hz/8=4096Hz Programa ejecutado en flash	3	7	μA
Modo activo (100 kHz)	f _{DCO} =100 kHz f _{ACLK} =0Hz Programa ejecutado en flash	72	95	μA

En el caso del modo activo, el programa se ejecuta en flash y por lo tanto el valor a utilizar es 350mA. En el capítulo 4 se utilizan estos datos para determinar el consumo del microcontrolador.

2.3. Código

El lenguaje de programación escogido para desarrollar del código fuente es C por tener mayores conocimientos que en lenguaje ensamblador.

Durante el desarrollo se ha tenido en cuenta que cualquier lectura realizada antes del tiempo de calentamiento establecido por el fabricante (5 minutos) no es fiable. En todos los casos, se espera al menos 5 minutos desde la alimentación del sensor para realizar la lectura y transmitir el dato. Esta limitación nos ha llevado a plantear diferentes períodos de lectura. En aplicaciones donde la medición debe ser más frecuente se requiere tomar una muestra en períodos de tiempo cortos. En todo caso, el tiempo mínimo entre lecturas debe ser superior a 15 segundos, por el tiempo de muestreo del propio sensor. Se considera 1 minuto y 5 minutos tiempos adecuados para este tipo de aplicaciones. En aplicaciones menos críticas, donde se la frecuencia de muestreo sea mayor, pueden utilizarse tiempos de 25 minutos o 115 minutos.

Con los períodos de 1 minuto y 5 minutos, el sensor siempre se mantiene activo ya que una tras una desconexión se necesita 5 minutos para obtener un valor correcto. En este caso las baterías tendrán una vida útil muy corta. En el capítulo 4 se entra en detalle en los consumos. Con períodos superiores, es posible mantener desconectado el sensor teniendo un ahorro energético importante. Se muestra en la tabla 2.5 la configuración de jumpers para seleccionar cada período de trabajo. Conectado significa que el jumper esta

puesto y detecta un nivel bajo. Desconectado significa que el jumper no esta puesto y detecta un nivel alto.

Tabla 2.5 Configuración jumpers para selección de modo y período

Período	Jumper 1 (modo)	Jumper 2 (período)
P1 - 1 minuto	Conectado	Conectado
P2 - 5 minutos	Conectado	Desconectado
P3 - 25 minutos	Desconectado	Conectado
P4 - 115 minutos	Desconectado	Desconectado

Se detallan a continuación los diferentes bloques que forman el diagrama de flujo.

- *Configuración Inicial:* Tras recibir alimentación, se procede a la inicialización de las variables. A continuación se configura el watchdog timer, el timer 0 y los puertos digitales, definiéndolos según son entradas o salidas. Se activa el regulador poniendo a nivel bajo la salida digital 2.1 del microcontrolador. Mediante la salida digital 2.3 se activa la interrupción IRQ0 del transceptor con un nivel alto y este queda en modo de bajo consumo. Se configuran como entradas digitales los puertos 2.2 y 2.4 utilizados para la selección de modo y período. En este punto se configuran los puertos UART e IC2 mediante los registros pertinentes utilizando la señal de reloj ACLK. Para finalizar se habilitan las interrupciones para el timer 0.
- *OFF Micro – Timer 5 minutos:* El microcontrolador entra en modo de bajo consumo y queda a la espera de una interrupción del timer 0 transcurridos 5 minutos permite que Mientras pasan los 5 minutos, el regulador permanece activo.
- *ON Micro:* El microcontrolador sale del modo de bajo consumo por una interrupción producida por el timer 0 una vez han pasado los 5 minutos y comienza la lectura de CO₂.
- *Lectura CO₂:* El sensor transmite la medida en dos bytes, en el formato de trama visto en el capítulo 1, teniendo esto en cuenta el micro envía la orden al sensor para obtener el primer byte y se almacena en una variable. El checksum también se guarda pero no se realiza la comprobación. Al finalizar la recepción del primer byte, se envía la orden para obtener el segundo byte y se realiza el mismo proceso. Una vez se tiene los dos bytes se finaliza la lectura.
- *Comprobación J1, J2:* Se comprueban el estado de los jumpers 1 y 2. Se almacena en dos variables, el estado (jumper 1) y el período (jumper 2).
- *Estado?:* Se comprueba la variable de estado y se entrará en un modo de alto consumo o bajo consumo. Si el estado es 1, el regulador no se para y se procede a la transmisión Zigbee. Si el estado es 2, se envía la señal de “shutdown” al regulador con un nivel alto en la salida digital 2.1 del microcontrolador y a continuación se transmiten los datos Zigbee. Se hace en este momento para que en el modo de bajo consumo el regulador y sensor estén activos el menor tiempo posible.
- *ON Zigbee – Transmisión Zigbee – OFF Zigbee:* La salida digital 2.3 del microcontrolador se pone a nivel bajo para que detecte el siguiente flanco y pueda activarse la interrupción. La salida 2.0 del

microcontrolador se pone a nivel alto activando la interrupción IRQ1 y cambiando el estado del transceptor a activo. Se procede a la transmisión desde el microcontrolador al transceptor por la UART. El registro donde se almacenan los datos a transferir por la UART en el microcontrolador permite la transmisión byte a byte. Se envían un total de 12 bytes. Cada byte se corresponde con un carácter ASCII y se transmite lo siguiente: **AT+SCAST="byte alto&byte bajo" <CR>**. Tras esta transmisión, el transceptor ensambla los bytes recibidos en su buffer, interpreta y transmite los datos al nodo coordinador de la red Zigbee. Al finalizar la transmisión la salida digital 2.0 se pone a nivel bajo y la 2.3 a nivel alto, de forma que detecta el flanco en la salida 2.3 y activa la interrupción IRQ0 del transceptor que lo deja en modo de bajo consumo.

- *Estado? – Período?:* Se comprueban las variables de estado y período almacenadas anteriormente según la posición de los jumpers para determinar el período hasta la siguiente lectura.
 - o Si el estado = 1, entramos en la opción de alto consumo. Se ejecuta el período de espera según la variable de período (1 indica un minuto y 2 indica 5 minutos) con el micro desconectado. Al finalizar se activa el microcontrolador y se vuelve al inicio, realizando de nuevo otra lectura.
 - o Si el estado = 2 entramos en la opción de bajo consumo. Según el período (3 indica 25 minutos y 4 indica 115 minutos), se ejecutará un temporizador y al finalizarlo, como el regulador esta parado, se activa. Se desconecta el microcontrolador y se vuelve a esperar 5 minutos hasta volver a realizar la lectura de CO₂. Transcurrido los 5 minutos, se activa el microcontrolador y se vuelve al inicio, para realizar otra lectura.

La figura 2.4 muestra el diagrama de flujo del código implementado.

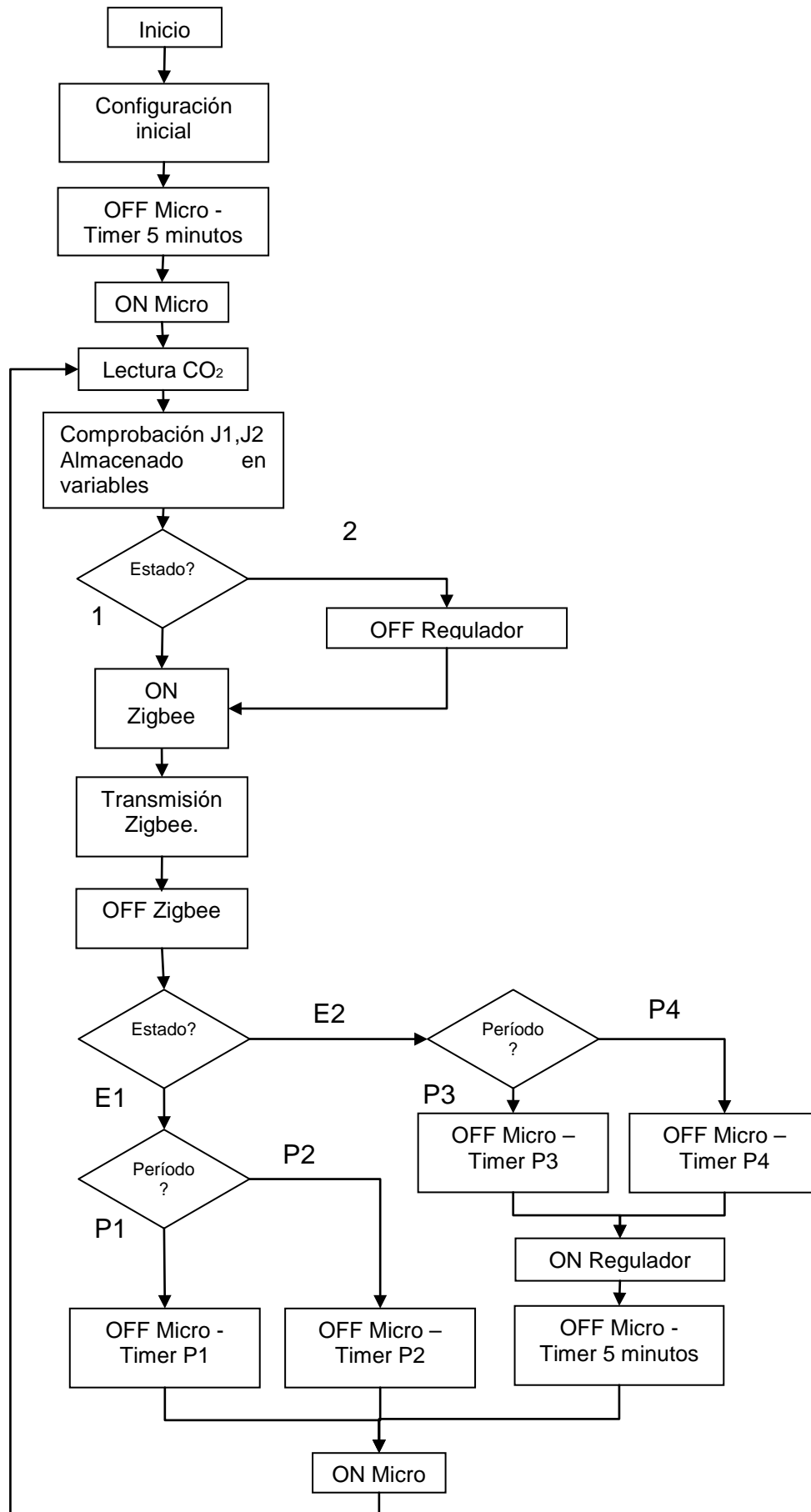


Fig. 2.4 Diagrama de flujo del código del programa

CAPÍTULO 3: El transceptor Zigbee

Para el desarrollo del nodo sensor se ha optado por utilizar un transceptor Zigbee con la capacidad de transmitir de forma inalámbrica los datos a otros dispositivos interconectados en una red PAN (Personal Area Network).

Se utiliza esta tecnología y el dispositivo ETRX2 porque en proyectos anteriores [1], [2] relacionados con REALnet ya ha sido utilizado y porque permite la integración del nodo en dicha red en proyectos futuros.

En el nodo sensor se sitúa el módulo Zigbee ETRX2 que establecerá, en este proyecto, comunicación con un módulo ETRX2-USB conectado a un ordenador. La comunicación se establece entre ambos módulos, siendo el nodo sensor el dispositivo final y el USB el coordinador de la red. Se entra en más detalle de estos conceptos a continuación.

3.1. Especificación Zigbee [13]

Zigbee es una especificación para comunicaciones inalámbricas en redes de ámbito personal (PAN) con transmisiones de baja velocidad donde es necesario también un bajo consumo. Es un protocolo de alto nivel que completa la especificación IEEE 802.15.4.

Sus principales características son:

- Banda de frecuencia:
 - o 868 MHz en Europa (1 canal).
 - o 915 MHz en Norte America, Australia y otros países (10 canales).
 - o 2,4 GHz en todo el mundo (16 canales).
- Permite varias topologías de red y direccionamiento de 16 bits.
 - o Topología mallada: Al menos uno de los nodos tiene más de una conexión.
 - o Topología estrella: El coordinador se sitúa en el centro.
 - o Topología de árbol: El coordinador es la raíz del árbol.
- Permite 3 tipos de dispositivos dentro de la red.
 - o Zigbee Coordinator (ZC): Debe existir uno por red. Se encarga de crear la red y encaminar. Solo existe uno por red. Debe estar conectado siempre.
 - o Zigbee Router(ZR): Se encarga de interconectar diferentes dispositivos o enlazarlos según la topología de la red. Debe estar conectado la mayor parte del tiempo.
 - o Zigbee End Device (ZED): Transmite la información a su nodo padre (coordinador o encaminador asociado). Puede permanecer la mayor parte del tiempo en reposo y conectarse solo cuando tiene necesidad de transmitir.
- Hasta 500 metros de alcance según el entorno.
- Bajo consumo energético.

3.1.1. Comparativa con otros sistemas inalámbricos

En comparación con otros sistemas inalámbricos [14] como puede ser Bluetooth, Zigbee nos ofrece:

- Mayor número de nodos conectados a la red. 65535 en subredes de 255 para Zigbee frente a 8 de Bluetooth.
- Menor consumo en reposo y en transmisión. 40 mA en transmisión y 0,2 mA en reposo de Bluetooth frente a 32 mA en transmisión y 1 μ A en reposo de Zigbee.
- Como contrapartida, la velocidad de comunicación es menor, de 250 Kb/s en Zigbee frente a 3 Mb/s en Bluetooth.

Para las redes de sensores inalámbricas, Zigbee es ideal porque permite un consumo muy reducido, capacidad de dormir los dispositivos, reintegrarse en la red al despertar de forma transparente y la integración de multitud de nodos en topologías de red muy variadas.

3.2. Características módulo ETRX2

El módulo ETRX2 es un dispositivo que integra un microcontrolador y un transceptor Zigbee. El microcontrolador es un EM250 que incorpora la pila EmberZNet2 [15].

La pila EmberZNet2 define tres tipos de dispositivos no definidos en Zigbee.

- MED (Moving End Device): Es un dispositivo final del tipo ZED que se encuentra en movimiento y se desvincula de la red cuando pierda la conexión y puede unirse a otra red a la cual tenga permiso.
- SED (Sleppy End Device): Es un dispositivo final del tipo ZED que no se desvincula de la red cuando pierda la conexión con los nodos. Permanece la mayor parte del tiempo en modo de bajo consumo "dormido" y cuando se activa se vuelve a conectar a la red.
- SINK: Es un dispositivo tipo ZC que realiza las funciones de almacén de datos.

La comunicación con el transceptor en este proyecto se realiza a través de UART (incorpora de forma opcional I²C y SPI) utilizando instrucciones AT. El microcontrolador tiene una memoria flash donde almacena las variables de su configuración, son los registros S y son modificables mediante ordenes AT, de forma local o remota. Algunos de estos registros están protegidos por una clave y las comunicaciones pueden encriptarse. El transceptor incorpora varias entradas/salidas digitales y temporizadores configurables mediante los registros. Permite además activar diferentes modos de bajo consumo mediante estas entradas digitales.

3.1.2. Órdenes AT

Se explica en este apartado el formato para las órdenes AT y se detallan utilizadas para la transmisión de datos y la configuración realizada.

En la figura 3.1 vemos el formato de órdenes AT.

AT	+	FUNCIÓN
----	---	---------

Fig.3.1 Formato órdenes AT

La orden utilizada se muestra en la figura 3.2 y es la que transmite al SINK. Se utiliza el envío unicast al SINK y no el broadcast porque el broadcast tiene un consumo mayor al enviar 3 veces el mensaje en la red Zigbee y porque en este proyecto la comunicación es punto a punto.

AT	+	SCAST:<DATOS>
----	---	---------------

Fig. 3.2 Orden para transmisión de datos al SINK

Otras órdenes utilizadas son las relativas a la asociación, creación y desasociación de la PAN [15]. No se detallan en profundidad porque solo se utilizaron la primera vez mediante el kit de desarrollo.

En este proyecto se ha trabajado directamente la comunicación entre un dispositivo final (sensor) y un coordinador (ETRX2USB) y toda la configuración esta enfocada a la interconexión entre ambos nodos. Se ha utilizado el kit de desarrollo mostrado en el anexo D para configurar los registros S necesarios en el transceptor del nodo. Mediante la orden "AT+DASSL" se ha desasociado de cualquier PAN a la cual estuviese previamente conectado el módulo del sensor. Los registros S se configuran mediante la orden "ATSXX=<valor>", en el siguiente apartado se entra en detalle.

3.1.3. Registros S

Existen un total de 52 registros que son utilizados para configurar el comportamiento del módulo ETRX2 [12]. Muchos de estos registros no son utilizados en este proyecto, por lo tanto detallaremos sólo los que tienen relevancia.

- S00: Máscara del canal. Permite definir una máscara para que el nodo solo se conecte a un canal en concreto.
- S01: Identificador de la PAN preferida. Define el identificador de una PAN a la cual conectarse al iniciar. El valor por defecto es FFFF, indicando que no hay PAN preferida y permitiendo la conexión a cualquier PAN detectada.

- S06: Función principal. Define el modo de trabajo del dispositivo dentro de la PAN y la forma de acceder a la esta.
- S0B: Configuración del puerto UART. Define los baudios, bits de datos, stop y paridad configurados para este puerto.
- S15: Función a realizar tras IRQ0. Define la actuación del nodo tras recibir una interrupción. Permite configurar multitud de acciones, pero solo se utilizan las referentes a los modos de bajo consumo.
- S16: Función a realizar tras IRQ1. Define la actuación del nodo tras recibir una interrupción. Permite configurar multitud de acciones, pero solo se utilizan las referentes a los modos de bajo consumo
- S29: Modo de energía. Definen el modo energético en el que se encuentra en cada momento.
- S2A: Modo de energía inicial. Define el modo energético en el que se iniciará el dispositivo.
- S2E: Define el comportamiento del dispositivo ante ciertos eventos externos como pueden ser las interrupciones. Permite configura si la activación de las interrupciones será con flanco de subida o bajada.

En el anexo D se detallan los valores que se pueden configurar en cada registro descrito anteriormente.

3.1.4. Interrupciones y modo de bajo consumo

El módulo ETRX2 soporta interrupciones externas (IRQ0 e IRQ1) que son utilizadas para entrar y salir de los diferentes modos de bajo consumo.

Las interrupciones IRQ0 e IRQ1 están disponibles en el pin 15 y 16 del conector EM250 GPIO del módulo ETRX2, tal y como se muestra en la figura 3.3. Estos pines se conectan directamente a los puertos digitales 2.0 y 2.3 del microcontrolador. El pin 17 se utiliza para la recepción de datos del microcontrolador y esta conectado al puerto 3.4, pin 15. En el capítulo 5 se muestra el esquemático del nodo y se pueden observar todas las conexiones.

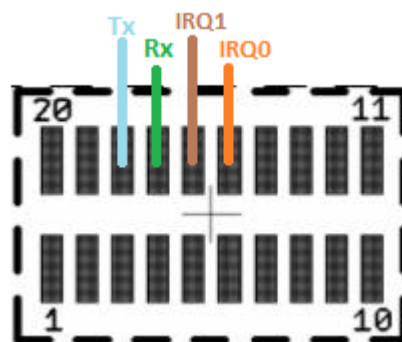


Fig. 3.3 Pines interrupciones y UART módulo ETRX2

Mediante los registros S15 y S16 se configura la acción a realizar tras la interrupción. La IRQ0 activará el nivel de bajo consumo y la IRQ1 volverá al estado normal.

El módulo tiene hasta 4 modos de trabajo, se utilizará el modo 0 y el modo 3 detallados en la tabla 3.1. Se considera el modo 0 como estado normal y el modo 3 como el de bajo consumo.

Tabla 3.1 Consumo según modos de trabajo módulo ETRX2

Mode	Router,COO				MED,SED			
	MCU	Radio	Timers	I	MCU	Radio	Timers	I
0	Awake	Awake	User defined	29mA	Awake	Asleep	User defined	11mA
1	Idle	Awake	User defined	23mA	Idle	Asleep	User defined	6mA
2	Awake	Awake	User defined	1 mA	Asleep	Asleep	User defined	1 mA
3	Asleep	Asleep	off	15µA	Asleep	Asleep	Off	15µA

Como se observará más adelante en el capítulo 4, en el modo normal de trabajo, (modo 0) el transceptor tiene un consumo de 35 mA en picos producidos cada segundo. El valor de 11mA se interpreta que es el valor medio, entre el consumo típico y los picos observados durante la actividad. En la figura 1.8 del anexo 8 se observa el comportamiento en el modo 0. Se deduce que el sistema de radio por defecto esta parado y cada segundo se activa para comprobar si hay algún dato a procesar o debe transmitir.

3.3. Configuración

Se debe realizar una configuración específica en cada uno de los nodos de la red. El nodo sensor se conecta como opción preferente al canal y PAN establecida por el coordinador, y se configuran las interrupciones IRQ0 e IRQ1 para que habiliten y deshabiliten el modo de ahorro de energía al recibir un flanco de subida. El ETRX2USB es el coordinador de la red porque es el creador de esta y se configura como SINK porque es quien recibe los datos. En el anexo D se pueden ver los valores que pueden tomar los registros más importantes y a continuación se definen los valores adoptados para cada dispositivo.

3.3.1. Configuración dispositivo final (nodo sensor)

En la tabla 3.2 se muestra la configuración final del nodo sensor.

Tabla 3.2 Valores configurados en registros S más importantes en el nodo sensor [15]

Registro	Valor configurado	Significado
S00	0001	Solo puede conectarse al canal 11
S01	3EC0	ID de la PAN a conectarse
S06	0202	SED- dispositivo final estático
S0B	0300	Configuración UART. 9600, 8N1

S15	0004	Interrupción IRQ0 pasa a modo 4
S16	0001	Interrupción IRQ0 pasa a modo 0
S2A	00	Modo de energía inicial por defecto a modo 0.
S2E	000A	Detección de flanco para interrupciones IRQ0 e IRQ1 en nivel alto.

3.3.2. Configuración coordinador

En la tabla 3.3 se muestra la configuración final del coordinador

Tabla 3.3 Valores configurados en registros S más importantes en el coordinador [15]

Registro S	Valor configurado	Significado
S00	0001	Solo puede crear una PAN en el canal 11
S01	3EC0	ID de la PAN creada
S06	0102	SINK de la red.
S0B	0300	Configuración UART. 9600, 8N1
S15	0001	Interrupción IRQ0 pasa a modo 0
S16	0001	Interrupción IRQ0 pasa a modo 0
S2A	00	Modo de energía inicial por defecto a modo 0.
S2E	0005	Por defecto, detección de flanco para interrupciones IRQ0 e IRQ1 en nivel bajo.

CAPÍTULO 4. Alimentación y consumo

Desde el primer momento, el objetivo principal de este proyecto ha sido diseñar un nodo para la medida de CO₂ con el menor consumo posible. Para conseguir este objetivo se ha tenido en cuenta tanto la elección del hardware como el diseño del código que ejecuta el microcontrolador.

Se expone en este capítulo el sistema de alimentación. También se añaden los resultados de las pruebas de campo realizadas con el diseño en la placa de pruebas (figura 4.1) midiendo los consumos de los diferentes componentes.

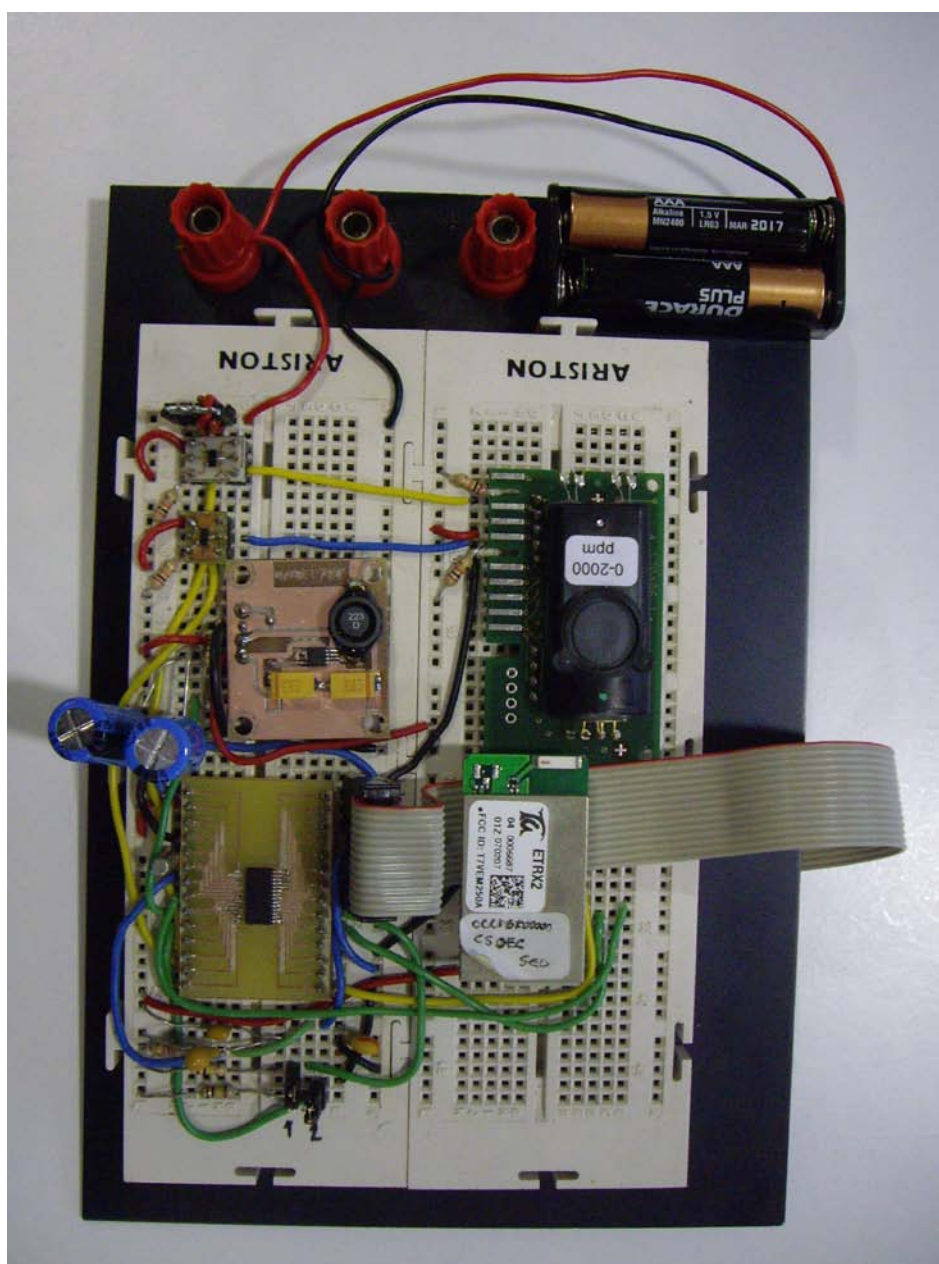


Fig. 4.1 Diseño del nodo sensor en la placa de pruebas

4.1. Sistema de alimentación

El sistema de alimentación tiene que utilizar pilas como fuente de energía. En la tabla 4.1 se muestran los márgenes de la tensión de alimentación de cada uno de los dispositivos del nodo.

Tabla 4.1 Márgenes de alimentación de los dispositivos que integran el nodo

Dispositivo	Margen de alimentación [V]
Microcontrolador MSP430F2132	2,2 – 3,6
Sensor EE89	4,75 -7,5
Transceptor ERTX2	2,1 – 3,6

Teniendo en cuenta estos parámetros se decide trabajar con 2 pilas alcalinas de 1,5 V del tipo AA conectadas en serie para alimentar el microcontrolador y el transceptor. La tensión de entrada oscilará entre 3V (a plena carga) y 2 V (carga baja). Para la alimentación del sensor, se introduce un regulador DC-DC del tipo elevador que nos dará una salida de 5 V. En la figura 4.2 podemos ver como quedaría el diseño en un diagrama de bloques.

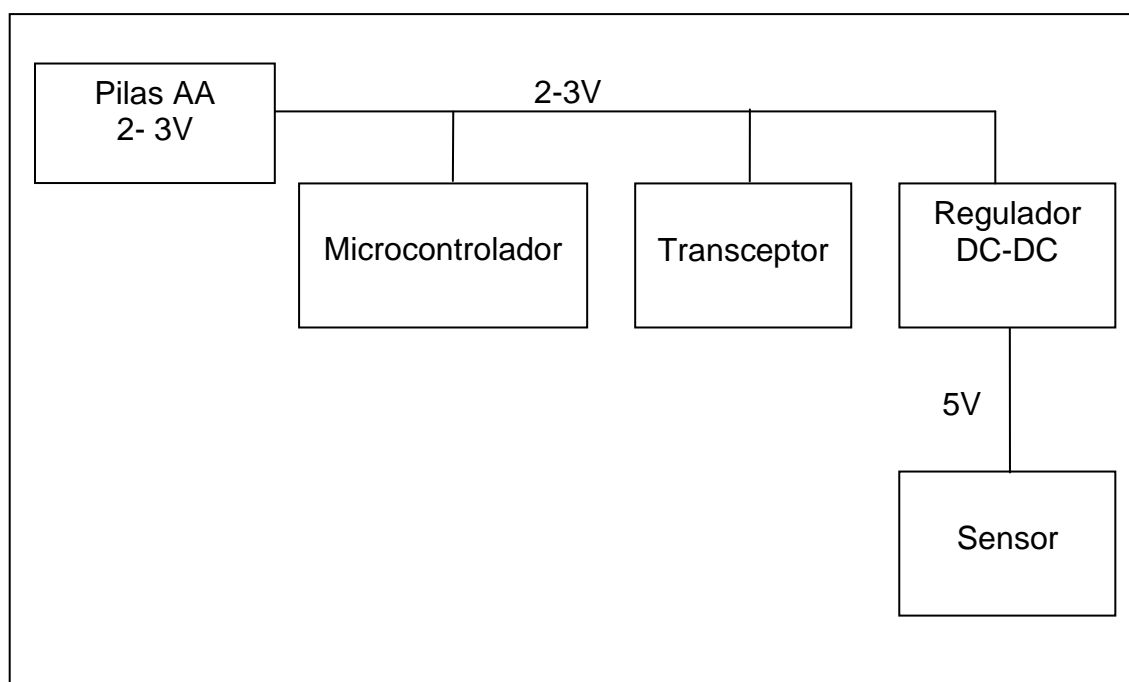


Fig. 4.2 Diagrama de bloques del sistema de alimentación

El regulador de tensión tiene que cumplir con unas especificaciones determinadas derivadas de las características del sensor a conectar. Para proceder a la selección del regulador se explican primero las características del consumo del sensor.

El sensor, cada 15 segundos cuando realiza una medida, tiene un pico de consumo que puede llegar hasta los 600 mA. A continuación, decrece con una descarga similar a la de un sistema de primer orden y pasados 350 ms baja el

consumo hasta unos 5 mA estables durante los 14,65 segundos restantes hasta la próxima medición.

La figura 4.3 muestra en una captura de osciloscopio la corriente que absorbe el sensor en el momento de la medición cuando se alimenta a 5V. La figura 4.4 muestra el consumo durante los 14,65 segundos hasta la siguiente medida. Estas medidas de corriente y todas las siguientes se han realizado con una sonda inductiva Tektronix TCPA300.

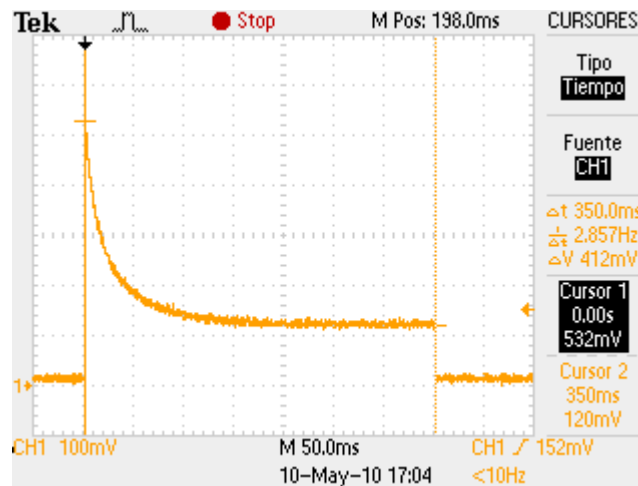


Fig. 4.3 Captura del consumo del sensor durante medición

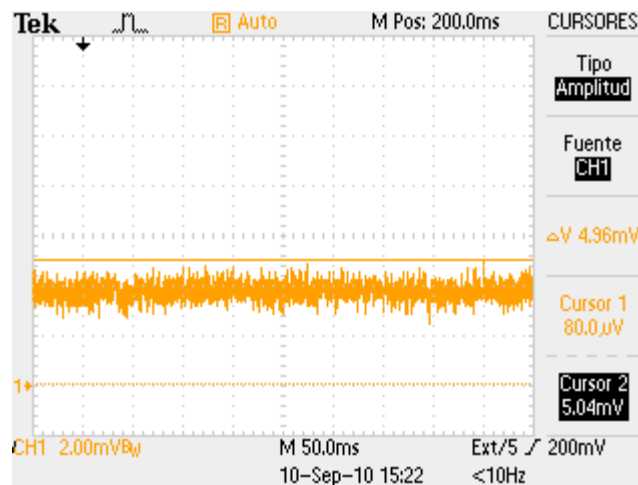


Fig 4.4 Captura del consumo del sensor cuando no realiza medición

Se realiza, pues, la búsqueda de un regulador de tensión que cumpla con las siguientes características:

- Salida de 5V
- Entrada de 2-3 V
- Intensidad de salida de hasta 1 A
- Permite desconexión
- Pocos componentes externos
- Alto rendimiento
- SMD

Hay dos tipos de reguladores adecuados, los conmutados que utilizan inductancias y las bombas de carga (charge pump) que utilizan condensadores para almacenar y transferir la energía. Teniendo en cuenta que el sensor produce picos de consumo muy altos, se fija como intensidad de salida para el regulador 1 A. Este valor descarta la mayoría de reguladores charge pump, que suministran intensidades menores en la salida. Por lo tanto nos decantamos por los conmutados aunque sean un poco más complejos de implementar. Se preseleccionan dos reguladores que cumplen los requisitos, el MAX1797 y el MAX8815A. La tabla 4.2 muestra sus características principales.

Tabla 4.2 Características reguladores de tensión [16], [17]

Modelo	MAX1797	MAX8815A
Tensión de entrada	1 – 5,5V	1,2 – 5,5V
Tensión de salida	Regulable mediante resistencias. 3 ó 5 V	Regulable mediante resistencias. 3 ó 5 V
Consumo desconectado/sin carga	2 μ A / 25 μ A	0,1 μ A /30 μ A
Eficiencia	> 95 %	97 %
Salida de intensidad máxima	Hasta 1 A	Hasta 1 A
Entrada para desconexión / extras	Sí / Detección de nivel bajo de batería	Sí / Detección nivel bajo batería, incorpora rectificador, desconexión térmica
Empaquetado	8TSSOP	10TDFN-EP
Precio	10,5 €/unidad	1,14 \$/1000 unidades

El escogido es el MAX1797 aunque el precio sea superior porque es más simple al tener menos pines y el tipo de conexión es 8TSSOP, que facilita la integración en PCB con las herramientas que se disponen. Otra razón es que ha sido utilizado en proyectos anteriores [18] y hay disponibilidad inmediata en el laboratorio.

Al conectar el sensor al regulador con los condensadores que recomienda el fabricante el sensor no funciona. Se observa que el regulador en su salida no soporta el pico inicial de 500-600mA. Se han introducido diferentes valores de capacidad tanto en la entrada y la salida. Inicialmente, se conecta en paralelo a los recomendados un condensador electrolítico de 1mF de capacidad y el sensor funciona correctamente. Se busca un condensador con estas características pero SMD. Se encuentran en [15] condensadores electrolíticos de aluminio que cumplen los requisitos y se utilizan en vez de de los de tántalo 47 μ F recomendados por el fabricante.

4.2. Consumo

A parte del sistema de alimentación escogido para el nodo y los criterios seguidos, es muy importante conocer los consumos de cada uno de los

elementos que incorpora el nodo. Se realiza la caracterización del sensor y el transceptor.

4.2.1. Consumo del sensor

El sensor, como se ha detallado anteriormente, tiene un consumo regular durante la mayor parte del tiempo y un pico de consumo cada 15 segundos, coincidiendo con el momento de efectuar la medida. Para caracterizar el consumo del sensor, se ha decidido aislarlo conectándolo solo a una fuente de alimentación de 5V y medir el consumo con la sonda de corriente. Según las capturas obtenidas, hay dos zonas diferenciadas de consumo. El consumo durante la medida y el consumo durante el tiempo hasta la siguiente medida, lo denominaremos tiempo de espera. Para obtener el consumo durante la medida, se realiza un media de los valores tal y como se muestra la figura 4.5.

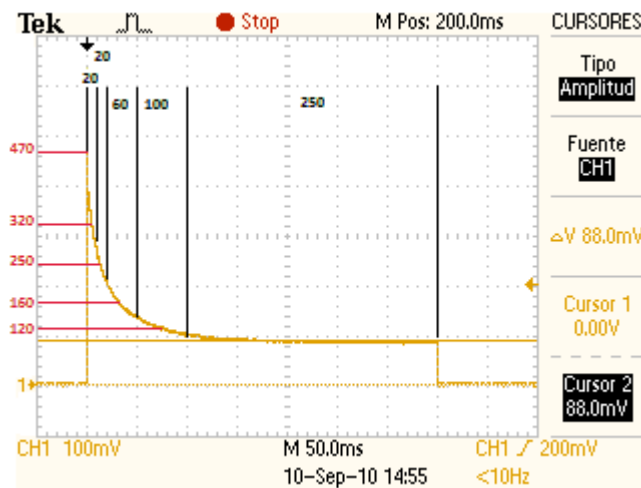


Fig. 4.5 Consumo durante medida

Mediante la fórmula 4.1 se calcula el consumo medio durante la medición.

$$I_{MEDIDA} = \frac{I_{\Delta t1} \cdot T_{\Delta t1} + I_{\Delta t2} \cdot T_{\Delta t2} + I_{\Delta t3} \cdot T_{\Delta t3} + I_{\Delta t4} \cdot T_{\Delta t4} + I_{\Delta t5} \cdot T_{\Delta t5}}{T_{TOTAL}} =$$

$$\frac{320mA \cdot 20ms + 250mA \cdot 20ms + 160mA \cdot 60ms + 120mA \cdot 100 + 88mA \cdot 250ms}{350ms} = 157,14mA$$

(4.1)

La fórmula 4.2 permite calcular el consumo medio del sensor.

$$I_{SENSOR} = \frac{I_{MEDIDA} \cdot T_{MEDIDA} + I_{ESPERA} \cdot T_{ESPERA}}{T_{TOTAL}} = \frac{157,14 \cdot 350ms + 5 \cdot 1465ms}{15} = 8,55mA$$

(4.2)

El fabricante indica que el consumo medio es de 7mA. El valor obtenido es 8,55 mA y es un 20 % superior. Se utiliza el valor de 7mA para los cálculos que se realizan a continuación.

4.2.2. Consumo del transceptor Zigbee

Según la especificación del fabricante el transceptor tiene un consumo de 35,5 mA durante la transmisión con la potencia configurada por defecto y de 1 µA cuando esta en modo de bajo consumo. El transceptor pasa la mayor parte del tiempo en modo de bajo consumo excepto cuando se realiza una transmisión.

Se realiza una caracterización del consumo del transceptor Zigbee donde se obtiene datos relevantes para su consumo. Esta caracterización, se ha realizado desconectando el regulador y sensor. Se introduce una modificación en el código para que el microcontrolador sólo se comunice con el transceptor y fuerce a enviar un valor y así poder caracterizar el comportamiento.

Se realizan varias capturas conectando la sonda de corriente al osciloscopio y situándola en la entrada de alimentación del transceptor. Se observa que los picos de consumo se realizan cada segundo con una duración de 5ms y un consumo medio de 36 mA. En la figura 4.6 se muestra el comportamiento del consumo durante la transmisión con el coordinador.

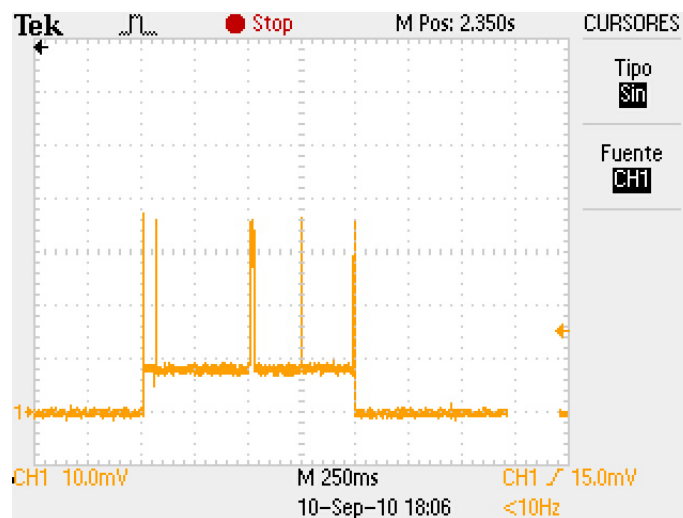


Fig. 4.6 Consumo transceptor durante transmisión

En el anexo I se muestran las capturas realizadas mediante las cuales se han obtenidos los tiempos y consumos durante la transmisión de una lectura.

El tiempo total que se mantiene activo el transceptor es de 990ms, de los cuales 43,9ms está consumiendo 36mA. El resto del tiempo, 946,1ms tiene un consumo medio de 7,6mA.

La fórmula 4.3 permite calcular el consumo medio.

$$I_{Zigbee} = \frac{I_{MAX} \cdot T_{MAX} + I_{MEDIA} \cdot T_{MEDIA}}{T_{TRANSMISIÓN}} = \frac{36mA \cdot 43,9ms + 7,6mA \cdot 946,1ms}{990ms} = 8,86mA \quad (4.3)$$

El consumo medio del transceptor durante la transmisión es de **8,86 mA**.

Tras la transmisión el transceptor pasa al modo de bajo consumo. En este modo el consumo se reduce y es de **1 μ A**.

4.2.3. Consumo del regulador

El regulador de tensión en modo desconectado "Shutdown" tiene un consumo de 2 μ A. Cuando esta en reposo, es decir sin una carga conectada, el consumo pasa a ser de 25 μ A. Mientras tiene conectada una carga, el consumo variará en función del voltaje en la entrada y la corriente en la salida tal y como se observa en grafica de la figura 4.7.

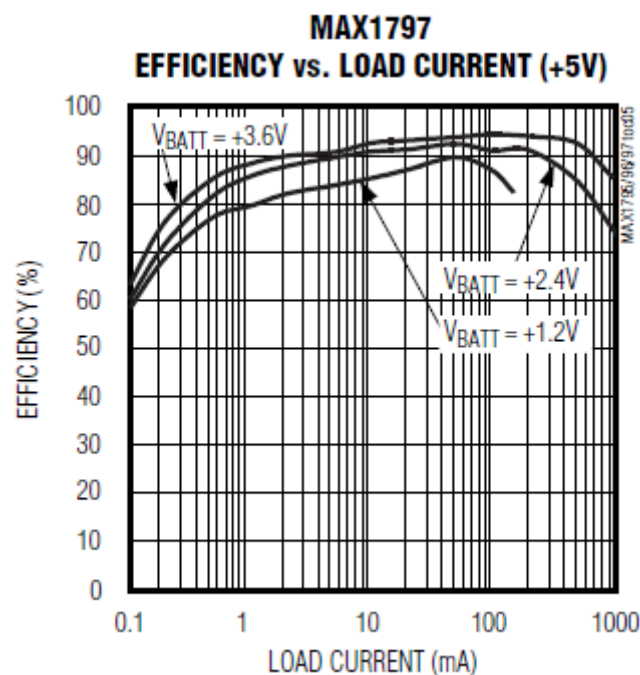


Fig. 4.7 Gráfica de eficiencia según carga de corriente[16]

Se toma como referencia los datos de VBATT= 2,4 V, ya que el sistema de alimentación puede variar entre los 2 y 3 V. Como se ha visto con anterioridad, el sensor tiene varias franjas de consumo. La mayor parte del tiempo tiene un consumo sobre los 7-8 mA y a estos niveles, la eficiencia es del 90 %. Durante el pico de consumo, podemos llegar hasta los 500mA, donde la eficiencia es del 85 %. Durante los 350ms que tiene un consumo mayor, la media es de 90 mA, donde la eficiencia vuelve a ser 90%. Se utiliza el valor del 90 % para todos los cálculos.

Con la fórmula 4.5 se calcula el consumo del regulador con el sensor conectado. Se utiliza el valor medio entre 2 y 3 V para el voltaje de entrada.

$$I_{\text{Regulador}} = \frac{V_{\text{OUT}} \cdot I_{\text{OUT}}}{\eta \cdot V_{\text{IN}}} = 5V \cdot \frac{7\text{mA}}{0,9 \cdot 2,5V} = 15,55 \text{ mA} \quad (4.4)$$

Debido a que el sensor se alimenta a través del regulador, el consumo durante la actividad utilizado para los cálculos es **15,55mA**.

El consumo en “shutdown” del regulador es **2 μ A**.

4.2.4. Consumo del microcontrolador

Los microcontroladores de la familia MSP430 tiene varios modos de bajo consumo que permiten al desarrollador modificar el estado del microcontrolador según sus necesidades en cada momento mejorando el ahorro de energía. Las diferencias existentes entre cada modo consisten en las señales de reloj que permanecen activas o no durante el reposo. El consumo también varía durante la fase de actividad según la frecuencia de trabajo seleccionada y el uso del oscilador interno. En el capítulo 2 se ha explicado con detalle.

Los valores utilizados para los cálculos son los consumos típicos aportados por el fabricante:

- Activo: 350 μ A.
- Bajo consumo: 0,9 μ A.

El microcontrolador esta activo durante el tiempo de lectura (Figura 4.8) y transmisión Zigbee. Para averiguar el tiempo de actividad del microcontrolador durante la lectura se insertan varios puntos de parada en el código y se captura la transmisión del primer y segundo byte con la sonda conectada en el canal de datos de I²C.

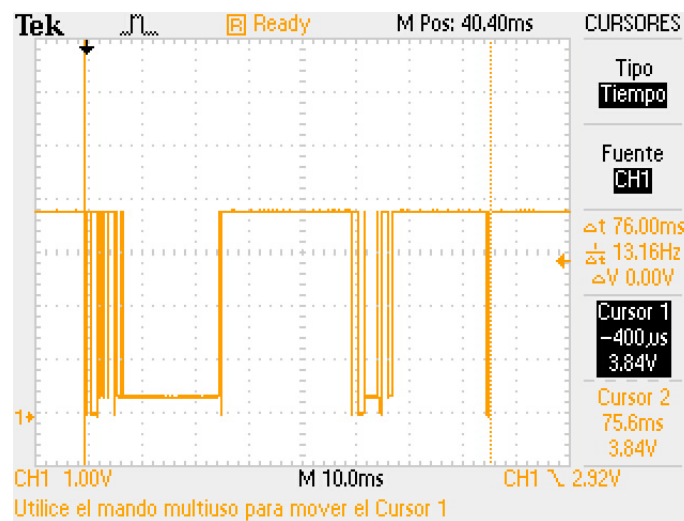


Fig. 4.8 Tiempo transmisión I²C

El tiempo de lectura CO₂ son 76 ms.

El tiempo de transmisión Zigbee son 990 ms.

La fórmula 4.5 muestra el tiempo total de actividad del microcontrolador por cada lectura de CO₂ realizada.

$$T_{\text{micro ACTIVO}} = T_{\text{TXZIGBEE}} + T_{\text{LECTURACO}_2} = 990\text{ms} + 76\text{ms} = 1066\text{ms} \quad (4.5)$$

4.2.4.1. Consumo teórico

Para realizar los cálculos de consumo teóricos se utilizan los datos obtenidos anteriormente. Con la idea de poder comparar el consumo entre los diferentes modos se realizarán los cálculos sobre 120 minutos.

En la tabla 4.6 mostramos los valores con los que realizaremos los cálculos.

Tabla 4.6 Consumos y tiempos en activo y reposo

Dispositivo	Consumo Activo[mA]	Consumo Reposo [μ A]
Microcontrolador	0,350	0,9
Regulador + Sensor	12,96	2
Transceptor	8,86	1

En la figura 4.9 se muestra la evolución temporal de la activación de los dispositivos para el modo de alto consumo y en la figura 4.10 para el modo de bajo consumo.

- T1: Tiempo para realizar la configuración inicial. 250 milisegundos.
- T2: 5 minutos del calentamiento del sensor.
- T3: Tiempo de lectura del CO₂, 76ms
- T4: Tiempo de transmisión datos mediante transceptor, 990ms.
- En el modo 1, el valor T5 puede ser 1 ó 5 minutos.
- En el modo 2, el valor T5 puede ser 25 ó 115 minutos.

Para el cálculo en el modo de alto consumo, solo se tiene en cuenta T3, T4 y T5, que son los que forman parte del régimen estacionario que se va a repetir continuamente. No se tiene en cuenta T1 ni T2 ya que solo se realiza la primera vez. Para el cálculo en el modo de bajo consumo, no se tiene en cuenta T1 que solo se realiza la primera vez.

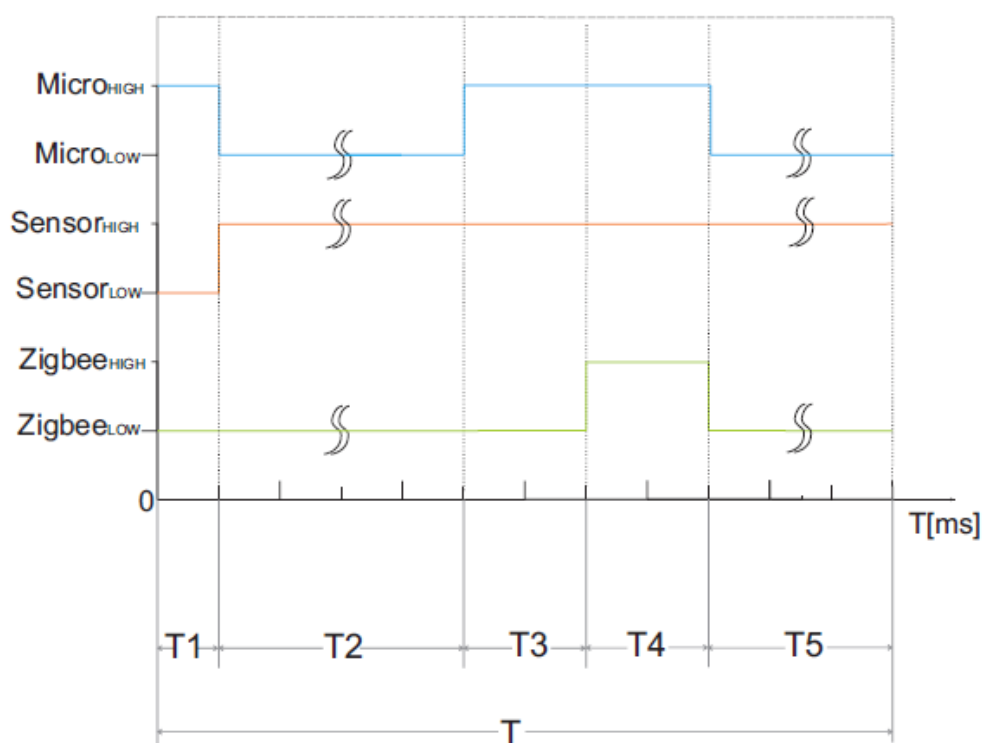
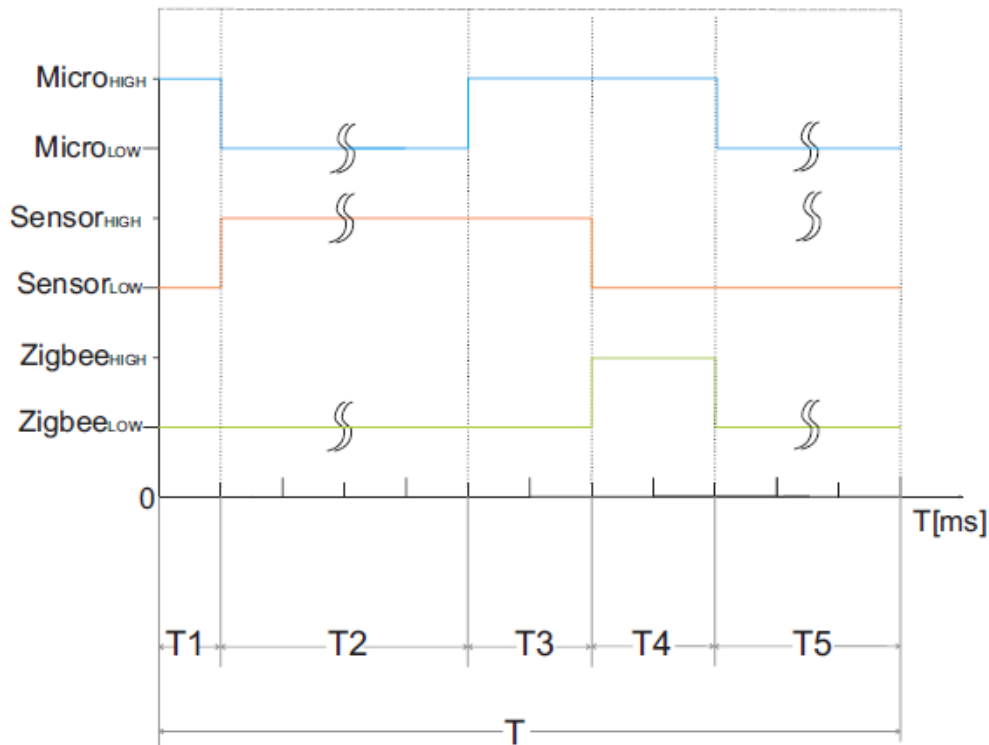


Fig. 4.9 Evolución temporal de activaciones en el modo de alto consumo**Fig. 4.10** Evolución temporal de activaciones en el modo de bajo consumo

Para realizar el cálculo del consumo en el modo 1 se utiliza la fórmula 4.6.

$$I = \frac{I_{\mu\text{CON}} \cdot T_{\mu\text{CON}} + I_{\mu\text{COFF}} \cdot T_{\mu\text{COFF}} + I_{\text{ZigON}} \cdot T_{\text{ZigON}} + I_{\text{ZigOFF}} \cdot T_{\text{ZigOFF}} + I_{\text{Re gON}} \cdot T_{\text{Re gON}} + I_{\text{Re gOFF}} \cdot T_{\text{Re gOFF}}}{T_{\text{TOTAL}}} \quad (4.6)$$

Para el modo de alto consumo período 1

$$I = \frac{350\mu\text{A} \cdot 1\text{s} + 1,4\mu\text{A} \cdot 60\text{s} + 8,86\text{mA} \cdot 1\text{s} + 1\mu\text{A} \cdot 60\text{s} + 15,5\text{mA} \cdot 61\text{s}}{61\text{s}} = 15,70\text{mA} \quad (4.7)$$

Para el modo de alto consumo período 2:

$$I = \frac{350\mu\text{A} \cdot 1\text{s} + 1,4\mu\text{A} \cdot 300\text{s} + 8,86\text{mA} \cdot 1\text{s} + 1\mu\text{A} \cdot 300\text{s} + 15,55\text{mA} \cdot 301\text{s}}{301\text{s}} = 15,58\text{mA} \quad (4.8)$$

Para el modo de bajo consumo período 3:

$$I = \frac{350\mu A \cdot 1s + 1,4\mu A \cdot 1560s + 8,86mA \cdot 1s + 1\mu A \cdot 1560s + 15,55mA \cdot 300s + 2\mu A \cdot 1261}{1561s} = 2,99mA \quad (4.9)$$

Para el modo de bajo consumo período 4:

$$I = \frac{350\mu A \cdot 1s + 1,4\mu A \cdot 7200s + 8,86mA \cdot 1s + 1\mu A \cdot 7200s + 15,55mA \cdot 300s + 2\mu A \cdot 6901}{7201s} = 0,65mA \quad (4.10)$$

El tiempo de vida de las baterías se calcula utilizando la fórmula 4.11:

$$Tiempodevida = \frac{Capacidad_{batería}}{I_{media}} \quad (4.11)$$

Se aplica el cálculo de vida útil para cada modo/período con unas pilas de capacidad de 3Ah. En la tabla 4.7 podemos ver el resultado en horas.

Tabla 4.7 Tiempo de vida de las baterías según el modo.

Modo – Período	Tiempo de vida. Horas / Días
Modo alto consumo – Período 1	191/8
Modo alto consumo – Período 2	192/8
Modo bajo consumo – Período 3	1003/41
Modo bajo consumo – Período 4	4615/192

Hay una gran diferencia entre el modo de alto consumo y el modo de bajo consumo. Para el primero, se recomienda mantenerlo conectado a una fuente de alimentación permanente, ya que el recambio de baterías debería realizarse cada 8 días y esto no es viable. Para el modo de bajo consumo, la vida útil es bastante mayor. Se recomienda utilizar estos modos.

CAPÍTULO 5. Montaje final y validación

En este capítulo se muestra el esquemático del montaje final del nodo sensor y el diseño de la PCB. Se detalla el sistema montado para la validación.

5.1. Montaje final

En la figura 5.1 se muestra el diseño final de la PCB.

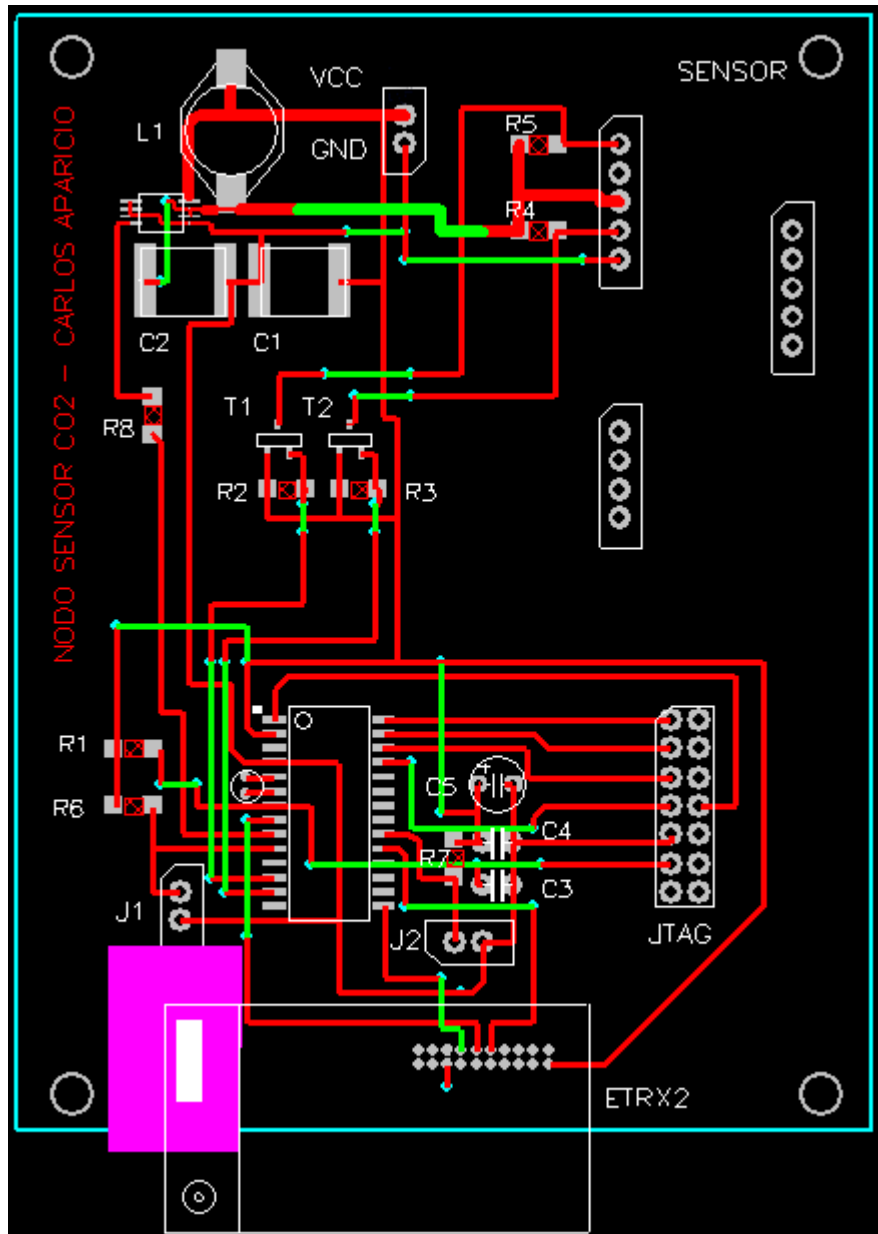


Fig. 5.1 Diseño final en PCB

La figura 5.2 muestra el esquemático del nodo sensor donde se aprecian todas las conexiones y componentes. Cabe destacar que se ha incorporado el puerto

JTAG a través de un conector de 14 pines para conectar el MSP430UIF y poder descargar el código y modificarlos si fuese necesario.

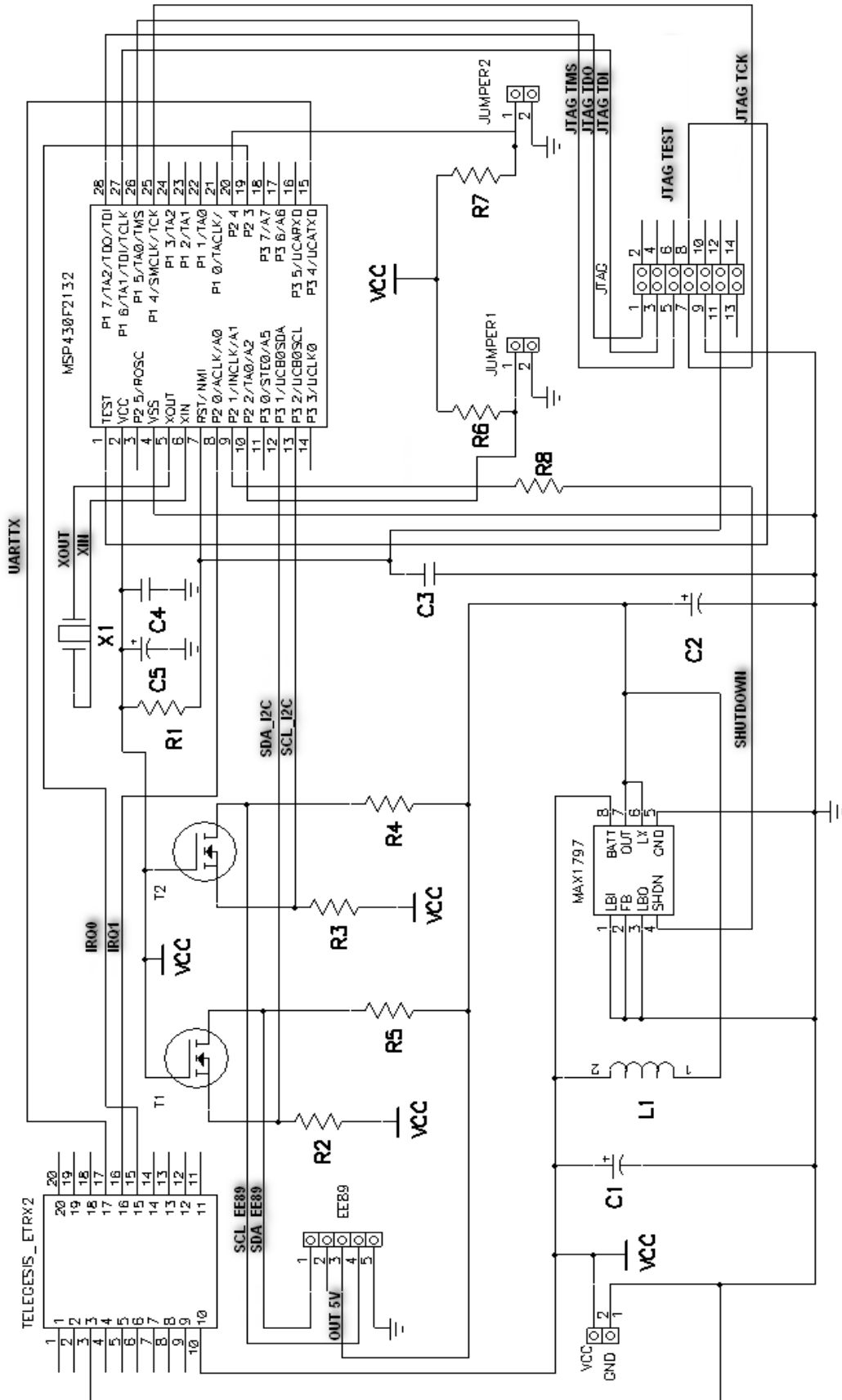
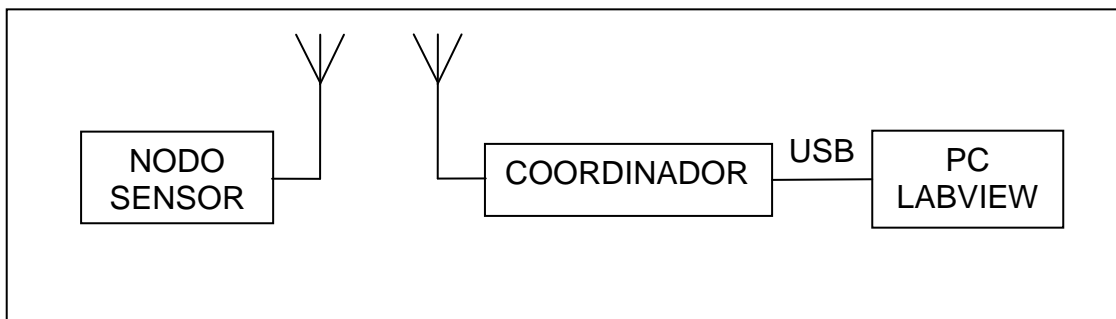
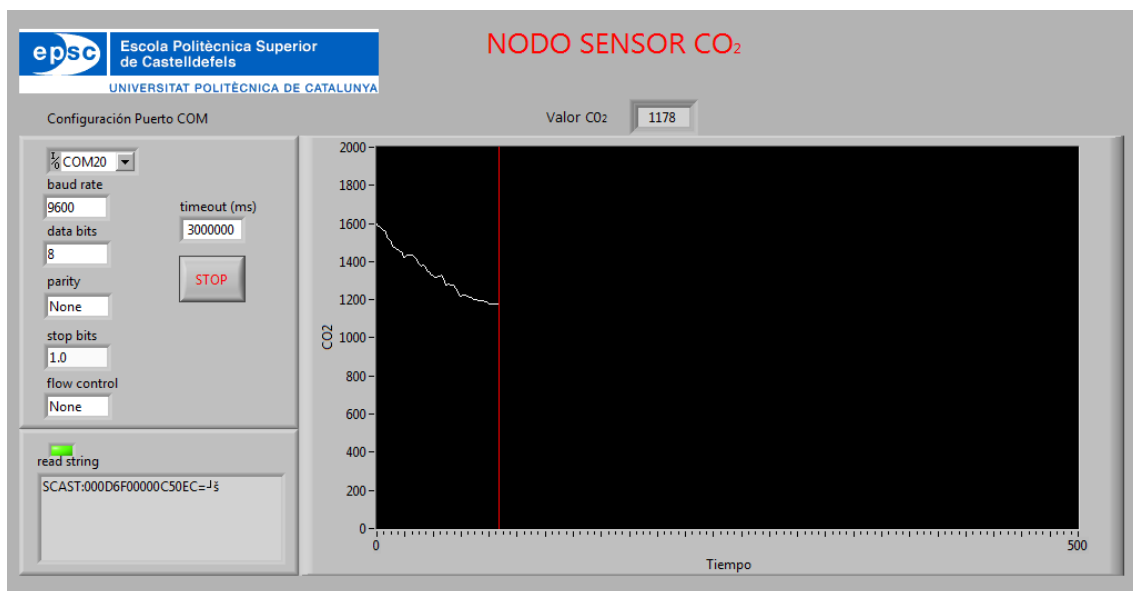


Fig. 5.2 Esquemático nodo sensor

5.2. Validación e Interfaz gráfica

Para validar el sistema y mostrar los datos de forma intuitiva se realiza el montaje de la figura 5.3 y se elabora mediante LabVIEW una interfaz que muestra una gráfica de la evolución de CO₂

Los datos se transmiten desde el nodo sensor al coordinador de forma inalámbrica. El coordinador las transmite a través del USB los datos recibidos al PC y la aplicación de LabVIEW realiza el tratamiento de estos datos y los muestra por pantalla en una gráfica, podemos ver la interfaz diseñada en la figura 5.4.

**Fig. 5.3** Diagrama de bloques del sistema de validación**Fig. 5.4** Interfaz gráfica LabVIEW durante medición de CO₂ en el laboratorio

CAPÍTULO 6. CONCLUSIONES

6.1. Objetivos alcanzados

Se ha alcanzado el objetivo del proyecto al diseñar y validar un nodo sensor autónomo con modo de alto y bajo consumo, con comunicación digital a través de bus I²C y Zigbee para transmitir datos y ser tratados por un PC.

El mayor reto ha sido conseguir minimizar el consumo teniendo un sensor con unos picos de alimentación muy elevados cada 15 segundos. Para ello ha sido necesario desconectar todos los dispositivos y optimizar al máximo el código implementado. También ha sido un reto la adaptación de niveles en el bus I²C y la conseguir un funcionamiento adecuado del regulador con un consumo muy elevado en un instante muy pequeño. Sin experiencia previa se ha desarrollado una interfaz para LabVIEW que realiza el tratamiento de los datos ofreciendo de forma intuitiva la evolución. A nivel de desarrollo, durante 3 meses se han realizado varias implementaciones de aplicaciones y ejemplos de la bibliografía, desarrollando incluso un programa en Visual Basic 6 que almacena en un archivo CSV los valores de CO₂ y comunicarlo a través del puerto RS-232 que incorpora la placa de experimentación MSP430FG4618/F2013 al ordenador y extraer gráficas en Excel. Gracias a este periodo previo, se han obteniendo los conocimientos oportunos para desarrollar el código definitivo.

Pensando en diferentes aplicaciones, el nodo se ha diseñado permitiendo la configuración manual de dos modos de trabajo, siendo posible cambiar el modo y los períodos de reposo modificando los jumpers 1 y 2 y reiniciando eléctricamente el dispositivo.

6.2. Impacto medioambiental

La red de sensores REALnet esta orientada al conocimiento del entorno del Campus del Baix Llobregat y a la mejora de la sostenibilidad realizando la medición de ciertos parámetros ambientales. Este proyecto aporta un nuevo dispositivo que permite la medición de CO₂.

En la actualidad se están produciendo fenómenos ambientales provocados por la acción del hombre que están cambiando el modo de actuar de los países industrializados, siendo cada vez más respetuosos y teniendo mayor concienciación con nuestro entorno. Las emisiones de dióxido de carbono a la atmosfera son la punta de lanza en este cambio y un sensor como el desarrollado en este proyecto puede unirse a los equipamientos existentes que nos permiten controlar de una forma más eficiente y barata dichas emisiones.

El sensor desarrollado en este proyecto tiene como principal objetivo el mínimo consumo, siendo más respetuoso con el medio ambiente. Por si sólo, el sensor tiene un impacto ambiental derivado de los componentes electrónicos que lo componen.

6.3. Líneas de trabajo futuras

En el diseño del sensor se ha tenido en cuenta posibles mejoras futuras y es por ello que se ha incorporado el puerto JTAG que permite la programación y depuración del código del microcontrolador mediante el MSP430FET. Gracias a esta opción, se podría modificar el código para integrar este nodo dentro de la red REALnet entre otras posibilidades.

Una mejora interesante en el código consiste en permitir la modificación remota de parámetros mediante la interfaz Zigbee, tales como los modos de trabajo y tiempos de reposo. La incorporación de una placa fotovoltaica y unas baterías recargables, permitiría al nodo ser autosuficiente. Una actualización del firmware del sensor EE89, publicada el 07/2010 permitiría modificar el intervalo de medida, con la consecuente reducción de consumo asociada a cada medida y optimizando el consumo [19],[20].

Actualmente la red de sensores REALnet se dedica exclusivamente a la medición de parámetros ambientales para su análisis.

Una mejora de calado para esta red implicaría la actuación sobre el medio según el comportamiento instantáneo. Con la implementación de una red de actuadores se podría, por ejemplo, activar el sistema de aire acondicionado si se detecta una concentración determinada de CO₂ en una sala o activar una alarma si se sobrepasa el límite permitido en un punto concreto.

REFERENCIAS

- [1] Nodo autónomo para la medida ambiental de CO₂, temperatura y humedad. Luciano Cárdenas Antognazzi.
<http://hdl.handle.net/2099.1/5001>
- [2] Nodo sensor para la medida ambiental de CO₂, temperatura y humedad. Marcos Jara Burgos.
<http://hdl.handle.net/2099.1/4703>
- [3] Dióxido de carbono
http://es.wikipedia.org/wiki/%C3%93xido_de_carbono_%28IV%29
- [4] Efecto invernadero
http://es.wikipedia.org/wiki/Efecto_invernadero
- [5] Características sensor NDIR.
http://es.wikipedia.org/wiki/Sensor_de_Infrarrojo_No_Dispersivo
- [6] Datasheet Sensor EE89.
http://www.epluse.com/uploads/tx_EplusEprDownloads/DB_EE89_2_e_03.pdf
- [7] Especificación E2 interface.
http://www.epluse.com/uploads/tx_EplusEprDownloads/Specificati on_E2_Interface_15.pdf
- [8] Respuesta a órdenes EE89. E2 interface _v1_0_01.
http://www.epluse.com/uploads/tx_EplusEprDownloads/E2_INTER FACE_EE03_V1_1.PDFLink
- [9] Datasheet microcontrolador MSP430F2132.
<http://focus.ti.com/lit/ds/symlink/msp430f2132.pdf>
- [10] Especificación Bus I²C.
http://www.nxp.com/acrobat_download2/literature/9398/39340011.pdf
- [11] Web Texas Instruments con características MSP430F2XXX.
<http://focus.ti.com/paramsearch/docs/parametricsearch.tsp?familyId=912§ionId=95&tabId=1528&family=mcu&startIdx=1&endIdx=10#rt>
- [12] Guía de usuario de la familia MSP430F2xxx.
<http://focus.ti.com/lit/ug/slau144e/slau144e.pdf>
- [13] Especificación Zigbee
<http://140.116.72.245/~zak/ZigBee/Docs/TSC-ZigBee-Specification.pdf>

- [14] REALnet: xarxa de sensors sense fils per la monitorització ambiental del Campus del Baix Llobregat. Enric Olmos Martínez.
<http://hdl.handle.net/2099.1/5347>
- [15] TG-ETRX-R212-AT-Commands.
<http://www.telegesis.com/downloads/general/TG-ETRX-R212-Commands.pdf>
- [16] Datasheet regulador MAX1797.
<http://datasheets.maxim-ic.com/en/ds/MAX1795-MAX1797.pdf>
- [17] Datasheet regulador MAX8815A.
<http://datasheets.maxim-ic.com/en/ds/MAX8815A.pdf>
- [18] Sensor autónomo para la medida de CO₂, temperatura y humedad relativa. Enrique Gil de Avalue Muñoz.
https://mitra.upc.es/SIA/PFC_PUBLICA.DADES_PFC?w_codipfc=6311
- [19] Optimización del consumo en sensor EE89.
http://www.epluse.com/uploads/tx_EplusEprDownloads/AN0101_Rev1_0.pdf
- [20] Cambio del intervalo de medida y efectos en el tiempo de respuesta en sensor EE89.
http://www.epluse.com/uploads/tx_EplusEprDownloads/AN0102_Rev1_0.pdf
- [21] Guía de usuario placa experimentación MSP430FG4618/F2013.
<http://focus.ti.com/lit/ug/slau213a/slau213a.pdf>
- [22] Guía de usuario herramienta desarrollo MSP-FET430UIF.
<http://focus.ti.com/lit/ug/slau278e/slau278e.pdf>
- [23] Distribuidor de electrónica y componentes.
www.farnell.com
- [24] Distribuidor bobina Coilcraft.
<http://www.coilcraft.com/pa/order.cfm?ID=5447303>
- [25] Proveedor sensores OEM en España.
Contacto comercial: comercial@sensovant.com

BIBLIOGRAFÍA

- [26] MSP430 Microcontrollers Basics, John H. Davies, Newnes, Oxford, 2008.

ANEXO A. DATOS RELEVANTES DE SENSOR E+E89



EE89 Series

Miniature CO₂ Module for OEM / HVAC Applications

alterations according to customer specifications possible

The EE89 series CO₂ miniature module is designed especially for mass production and OEM applications.

The measuring principle is based on infrared technology (NDIR).

The EE89 series is maintenance free because of the patented E+E auto-calibration feature, compensating for the effects of aging and therefore warrants outstanding long term stability.

The output of the digital interface allows easy signal processing.

Contact pads at the edge or contact pins provide multiple ways of mounting the board.

With the EE89 testboard it is possible to display and save the measured values of the EE89 on the PC. An E2-RS232 adapter, a power supply unit and the EE89 evaluation software are included in the scope of supply.



Typical Applications

OEM
building management
demand HVAC installations

Features

maintenance free
autocalibration
highest accuracy
excellent long term stability
small dimensions
digital interface

Technical Data

Measuring values

CO ₂			
Measurement principle	Non-Dispersive Infrared Technology (NDIR)		
Sensor	E+E Dual-Source Infrared System		
Working range	0...2000 / 5000 / 10000ppm		
Accuracy at 25°C (77°F) and 1013mbar	0...2000ppm:	< ± (50ppm +2% of measuring value)	
	0...5000ppm:	< ± (50ppm +3% of measuring value)	
	0...10000ppm:	< ± (100ppm +5% of measuring value)	
Response time t_{90}	< 185s		
Temperature dependence	typ. 2ppm CO ₂ /°C (0...50°C / 32...122°F)		
Long term stability	typ. 20ppm / year		
Sampling rate	approx. 15s		

Output

0...2000 / 5000 / 10000ppm	digital E2 interface (details: www.epluze.com)
----------------------------	---

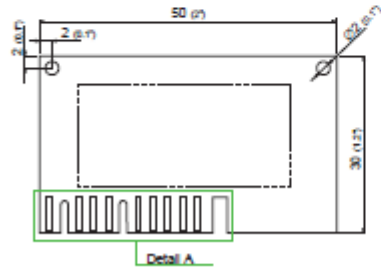
General

Supply voltage	4.75 - 7.5V DC		
Current consumption	typ. 7mA max. 0.5A for 0.3s		
Warm up time ¹⁾	< 5 min		
Electrical connection	contact pads or contact pins, grid 2.54mm (100mil)		
Working conditions	-20...60°C (-4...140°F)	5...85% RH (not condensating)	85...110kPa
Storage conditions	-20...60°C (-4...140°F)	5...85% RH (not condensating)	70...110kPa
Dimensions	50 x 30mm (2 x 1.2")		
Weight	approx. 15g (0.5oz)		

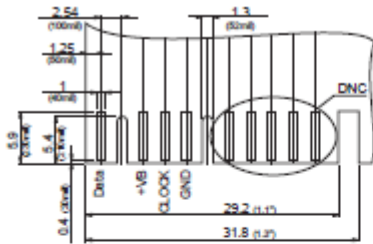
¹⁾ warm up time to reach specification values

Connection Diagram / Dimensions (mm)

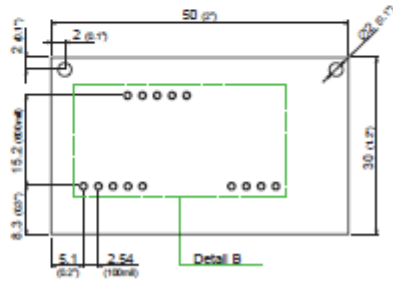
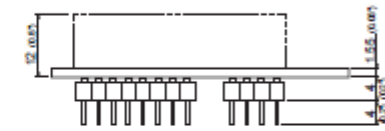
Mounting X (Contact Pads)



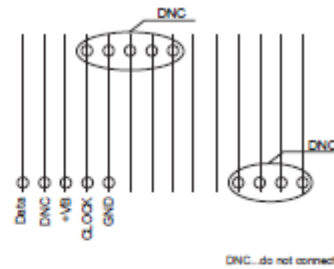
Detail A / Connection Diagram:



Mounting Y (Contact Pins)
designed for 28 pin socket or PCB soldering



Detail B / Connection Diagram:



Ordering Guide

MEASURING RANGE	TYPE	OUTPUT	MOUNTING
0...2000ppm (2)	CO ₂	(C)	digital interface (8)
0...5000ppm (6)			
0...10000ppm (10)			
EE89-			

Order Example

EE89-2C9Y
measuring range: 0...2000ppm
type: CO₂
output: digital interface
mounting: contact pins

Accessories

EE89 testboard (HA011010)

Especificación E2



1 Introduction

1.1 Overview

The E2 interface was specified by E+E Elektronik in 2004 and represents a subset of the E2 bus protocol. The E2 bus is based on similar principles as the I²C Bus¹ or the SMBus¹ introduced by Phillips Semiconductors in 1982. The most significant differences to the I²C Bus are the slower transmission rate, the slightly divergent addressing mechanism and the error detection (checksum). The E2 interface is used for the digital, bi-directional data transmission between a master module (e.g.: climate controller, microcontroller...) and a slave module (commonly an E+E transmitter). The data transmission takes place synchronous and serial, whereby the master is responsible for the generation of the clock pulse. The slave cannot send any data independently.

1.2 Agreements

The technical specifications in this document are to be regarded as recommendations and are applicable for all E+E transmitters with E2 interface, insofar as nothing to the contrary is specified in their data sheets.

Terms:

Master: Module with E2 interface that initiates a communication and can drive a clock signal.

Slave: Module with E2 interface that cannot drive a clock signal.

2 Characteristics

In the following, the E2 interface will be specified in accordance with the ISO-OSI Layer Model up to Layer 3. In addition, it should also be stated, that due to the great similarity with the SM Bus or the I²C Bus, the E2 interface is hardware-compatible with popular microcontrollers with interfaces of that type. E2-slaves can therefore be connected directly to the corresponding pins of such controllers.

2.1 Layer 1: Physical Layer

The E2 interface consists of two active lines (Clock and Data) as well as a grounding line as reference potential. The Clock and Data lines are connected with the operating voltage over pull-up resistors. In the idle state, both lines are at High-Level (positive logic). The respective inputs and outputs of the modules are designed as Open Drain (or Open Collector), and thus can be connected directly as „wired AND“.

¹ All brands, names, product names and logos listed are registered trade marks or brands of their respective owners, that we hereby explicitly acknowledge.

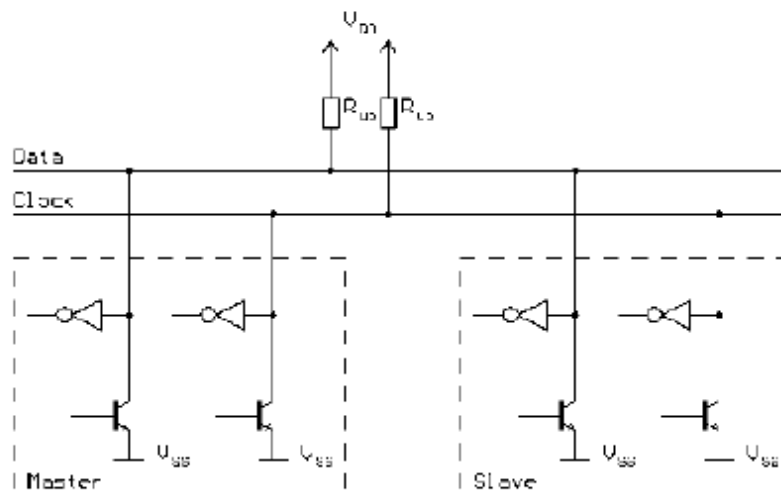


Fig. 1: Principle joining together of Master- and Slave Module

Parameters

Symbol	Parameter	Minimum	Maximum	Unit	Remark
V_{DD}	Operating voltage			V	See module description
V_{IH}	Input High Level	$0.8 V_{DD}$	$V_{DD} + 0.3$	V	
V_{IL}	Input Low Level	$V_{SS} - 0.3$	$0.2 * V_{DD}$ or 0.8	V	Smallest value is valid
V_{OL}	Output Low Level		0.7	V	$I_{in} = 0.5 \text{ mA}$
C_{max}	Line capacity to ground		1	nF	at $R_{up} = 22 \text{ k}\Omega$ see remark 1
f_{CLK}	Clock frequency (data rate)	500	5000	Hz	See remark 1
t_{CLKH}	Clock-High time	100		μs	
t_{CLKL}	Clock-Low time	100		μs	
R_{up}	Pull-up resistor	1	100	$\text{k}\Omega$	See remark 1

For further parameters not specified here, the specifications in the data sheets of the modules used are applicable.

Remark 1: The maximum data rate that can be achieved is dependent on the combination of the line capacity and the pull-up resistors.



2.2 Layer 2: Data Link Layer

2.2.1 Bit Transfer

The data transmission fundamentally takes place serial by bit and synchronous, whereby the clock line is used as synchronization line, and always controlled from the master. With the exception of the start- and stop conditions, a change of the level on the data line is always only permissible during a LOW-phase of the clock line. At the beginning of every communication there is a start condition. This is realized by a negative edge on the data line, with simultaneous sustained high-level on the clock line. After a delay of at least 4 μ s the clock line is drawn to low-level, and the first data bit (MSB) can be placed on the data line. The data transfer takes place during the high-phase of the clock line. After all data bits (incl. ACK/ NACK) have been sent, the communication is terminated with a stop condition. One terms a stop condition as a positive edge on the data line with simultaneous high-level on the clock line.

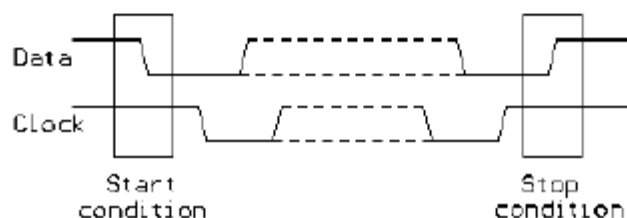


Fig. 2: Bit Transfer

Clock Low Extension

The slave has the possibility of holding the clock line at low level for up to 25ms after every transmitted data bit, in order to gain a little time for internal operations. The total transmission time for a complete byte may not however exceed 35ms.

2.2.2 Data Transfer Formats

The data are transmitted from the transmitter in bytes, and confirmed by the receiver individually (for each byte) with ACK (data line = Low) or NACK (data line = High) as ninth bit. The first data byte (control byte) is always sent from the master to the slave. Thereby the first seven bits represent the „Address“ of the slave module. The eighth bit (R/W) specifies the direction of the data transfer (R/W=0 data from master to slave; R/W=1 data from slave to master).

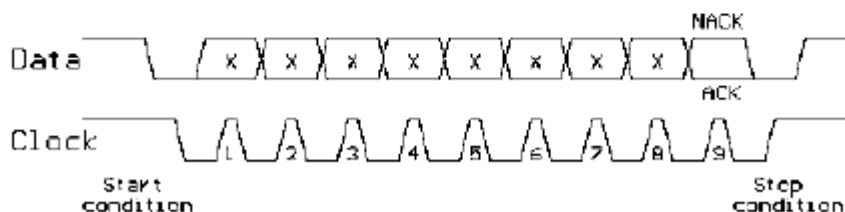


Fig. 3: Byte Transfer



2.2.3 Control Byte

The control byte of the E2 interface is used (in contrast to the E2 bus protocol) merely for the differentiation of various command modes (which are defined in the so-called Main Command) and the data flow direction (R/W). A differentiation of several slave modules by means of a different device address is not provided for with the E2 interface. The device address for all slaves is always 0, but for reasons of compatibility must also be transmitted.

The control byte is comprised as follows:

Control Byte							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Main Command				Device Address			R/W

Fig. 4: Control Byte Structure

Bit	Meaning
0 (LSB)	R/W
1	Device Address low bit (always 0)
2	Device Address (always 0)
3	Device Address high bit (always 0)
4	Main Command low bit
5	Main Command
6	Main Command
7 (MSB)	Main Command high bit

2.2.4 PEC Packet Error Code (Checksum)

For the detection of transmission errors a checksum byte is transmitted as the last data byte of every transmission. The checksum byte corresponds with the low byte of the sum (unsigned char) of all transmitted bytes.

Read: Checksum byte = (Control byte + Data byte) MOD 0x100

Write: Checksum byte = (Control byte + Address byte + Data byte) MOD 0x100



2.3 Layer 3: Network Layer (Protocol)

In this section, all defined commands of the E2 interface are described. For the data format of the transmitted data bytes, refer to the data sheets of the modules used.

2.3.1 Read Byte from Slave

This command is supported by all modules with E2 interface and is used for the reading out of individual data bytes.

Command structure:

Start	Control byte								ACK	Data byte								ACK	Checksum								ACK	Stop			
	7	6	5	4	3	2	1	0		7	6	5	4	3	2	1	0		7	6	5	4	3	2	1	0			7	6	5
Start							1	A	x	x	x	x	x	x	x	x	A	x	x	x	x	x	x	x	x	NA	Stop				

From Master to Slave
 From Slave to Master

Attention: For the Read command, the bit 0 of the control byte is always 1 (High). The Read command is terminated by a **NACK** and a stop condition of the master.

For Read Byte from Slave, the Main Commands are organized in such a form, that important data- and information bytes (all measured values, status byte, sensor type...) can be read out with one single command.

The following Main Commands are defined:

Control byte	Bit number				Type of command
	7	6	5	4	
0x11	0	0	0	1	Output sensor type (group)
0x21	0	0	1	0	Output sensor type (subgroup)
0x31	0	0	1	1	Read out available measured variables
0x41	0	1	0	0	Reserve
0x51	0	1	0	1	Reserve
0x61	0	1	1	0	Reserve
0x71	0	1	1	1	Status byte
0x81	1	0	0	0	Measured variable 1 low byte
0x91	1	0	0	1	Measured variable 1 high byte
0xA1	1	0	1	0	Measured variable 2 low byte
0xB1	1	0	1	1	Measured variable 2 high byte
0xC1	1	1	0	0	Measured variable 3 low byte
0xD1	1	1	0	1	Measured variable 3 high byte
0xE1	1	1	1	0	Measured variable 4 low byte
0xF1	1	1	1	1	Measured variable 4 high byte



2.3.1.1 Sensor Type (Group)

Identifies the current sensor type.

Example: Group = 3 hex for EE03.
Group = 7 hex for EE07.

2.3.1.2 Sensor Type (Subgroup)

Identifies the sub-type of the current sensor (upper 4 bits) and the method of the output (lower 4 bits).

Example: Subgroup = 19 hex for EEEx-1 in FT9 execution (E2 interface).
Subgroup = 46 hex for EEEx-4 in FT6 execution (4-20mA).

For further information see the module description.

2.3.1.3 Available Measured Variables

This field specifies which measured variables can be acquired with the current sensor. The following assignment to the individual bits is thereby used:

Available Measured Variables							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	Reserved	T-passive	Reserved	CO ₂	Flow	Temperature	Humidity

1...supported

0...not supported

2.3.1.4 Interrogate Status Byte

Through the interrogation of the status byte, the start of a measurement (in the slave) can be forced. After the waiting of the module-specific measuring time, the new measured values can then be read out. The status byte provides information about the validity of the last measurement. The assignment of the bits to the measured variables is equivalent to the assignment in the measured variables byte.

1...Measured value faulty (see module description)

0... Measured value in order

2.3.1.5 Read out Measured Variables

When reading out a 16 bit measured variable, it is expedient to read out the low byte first and then the associated high byte. This method of procedure ensures, that two associated bytes are always read out (when reading the low byte, the high byte is „captured“ in the slave). For the measured values data format used, refer to the data sheets of the corresponding modules.

ANEXO B. DATOS RELEVANTES DATASHEET MSP430F2132

MSP430F21x2 MIXED SIGNAL MICROCONTROLLER

SLAS578G - NOVEMBER 2007 - REVISED DECEMBER 2009

- Low Supply-Voltage Range, 1.8 V to 3.6 V
- Ultra-Low Power Consumption:
 - Active Mode: 250 μ A at 1 MHz, 2.2 V
 - Standby Mode: 0.7 μ A
 - Off Mode (RAM Retention): 0.1 μ A
- Ultra-Fast Wake-Up From Standby Mode in Less Than 1 μ s
- 16-Bit RISC Architecture, 62.5-ns Instruction Cycle Time
- Basic Clock Module Configurations:
 - Internal Frequencies up to 16 MHz With Four Calibrated Frequencies to \pm 1%
 - Internal Very Low-Power Low-Frequency Oscillator
 - 32-kHz Crystal
 - High-Frequency Crystal up to 16 MHz
 - Resonator
 - External Digital Clock Source
 - External Resistor
- 16-Bit Timer0_A3 With Three Capture/Compare Registers
- 16-Bit Timer1_A2 With Two Capture/Compare Registers
- On-Chip Comparator for Analog Signal Compare Function or Slope Analog-to-Digital (A/D) Conversion
- 10-Bit 200-kbps A/D Converter With Internal Reference, Sample-and-Hold, Autoscan, and Data Transfer Controller
- Universal Serial Communication Interface
 - Enhanced UART Supporting Auto Baudrate Detection (LIN)
 - IrDA Encoder and Decoder
 - Synchronous SPI
 - I²C™
- Brownout Detector
- Serial Onboard Programming, No External Programming Voltage Needed, Programmable Code Protection by Security Fuse
- Bootstrap Loader in Flash Devices
- On-Chip Emulation Module
- Family Members Include:
 - MSP430F2132
 - 8KB + 256B Flash Memory
 - 512B RAM
 - MSP430F2122
 - 4KB + 256B Flash Memory
 - 512B RAM
 - MSP430F2112
 - 2kB + 256B Flash Memory
 - 256B RAM
- Available in 28-Pin TSSOP and 32-Pin QFN Packages (See Available Options)
- For Complete Module Descriptions, See the *MSP430x2xx Family User's Guide*, Literature Number SLAU144

description

The Texas Instruments MSP430 family of ultra-low-power microcontrollers consists of several devices featuring different sets of peripherals targeted for various applications. The architecture, combined with five low-power modes, is optimized to achieve extended battery life in portable measurement applications. The device features a powerful 16-bit RISC CPU, 16-bit registers, and constant generators that contribute to maximum code efficiency. The digitally controlled oscillator (DCO) allows wake-up from low-power modes to active mode in less than 1 μ s.

The MSP430F21x2 series is an ultra-low-power microcontroller with two built-in 16-bit timers, a fast 10-bit A/D converter with integrated reference and a data transfer controller (DTC), a comparator, built-in communication capability using the universal serial communication interface, and up to 24 I/O pins.



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage. ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications. These devices have limited built-in ESD protection.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 2009, Texas Instruments Incorporated

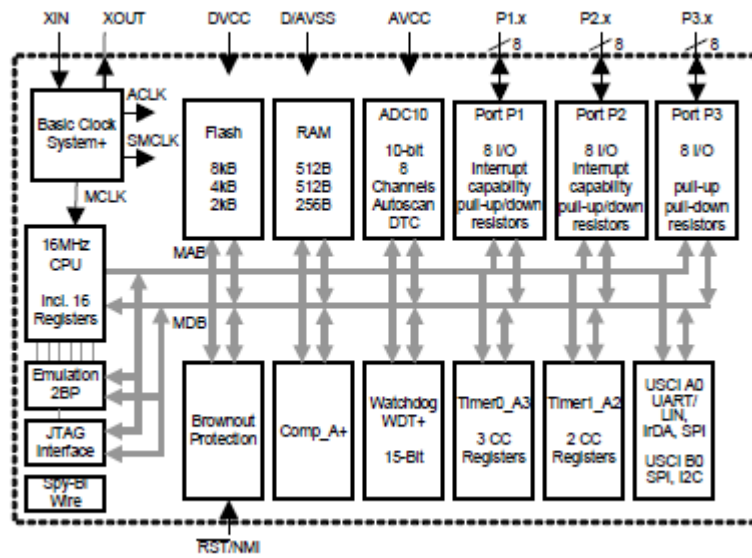
 **TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

pin designation, PW package

TEST/SBWTCK	1	28	P1.7/TA0.2/TDO/TDI
DV _{DD}	2	27	P1.6/TA0.1/TDV/CLK
P2.5/R _{TEST} /CA5	3	26	P1.5/TA0.0/TMS
DV _{DD}	4	25	P1.4/SMCLK/TCK
XOUT/P2.7/CA7	5	24	P1.3/TA0.2
XIN/P2.6/CA6	6	23	P1.2/TA0.1
RST/NMI/SBWDIO	7	22	P1.1/TA0.0/TA1.0
P2.0/ACLK/A0/CA2	8	21	P1.0/TACLK/ADC10CLK/CAOUT
P2.1/TAINCLK/SMCLK/A1/CA3	9	20	P2.4/TA0.2/A4/V _{DD1} /V _{DD2} /CA1
P2.2/TA0.0/A2/CA4/CAOUT	10	19	P2.3/TA0.1/A3/V _{DD1} /V _{DD2} /CA0
P3.0/UCB0STE/UCA0CLK/A5	11	18	P3.7/TA1.1/A7
P3.1/UCB0SIMO/UCB0SDA	12	17	P3.6/TA1.0/A6
P3.2/UCB0SOMI/UCB0SCL	13	16	P3.5/UCA0RXD/UCA0SOMI
P3.3/UCB0CLK/UCA0STE	14	15	P3.4/UCA0TXD/UCA0SIMO

pin designation, RHB package



ANEXO C. DATOS RELEVANTES DATASHEET REGULADOR MAX 1797

19-1798; Rev 0; 12/00

Low Supply Current, Step-Up DC-DC Converters with True-Shutdown

General Description

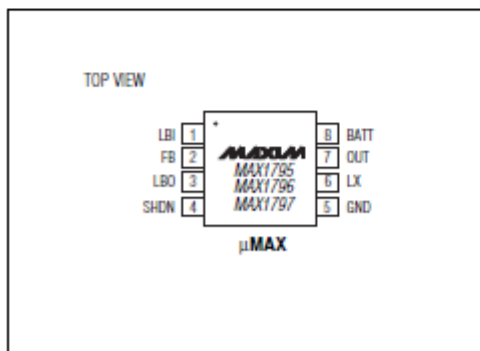
The MAX1795/MAX1796/MAX1797 are high efficiency step-up DC-DC converters intended for small portable hand-held devices. These devices feature Maxim's True-Shutdown™ circuitry, which fully disconnects the output from the input in shutdown, improves efficiency, and eliminates costly external components. All three devices also feature Maxim's proprietary LX-damping circuitry for reduced EMI in noise-sensitive applications. For additional in-system flexibility, a battery monitoring comparator (LBI/LBO) remains active even when the DC-DC converter is in shutdown.

The input voltage range is +0.7V to V_{OUT}, where V_{OUT} can be set from +2V to +5.5V. Startup is guaranteed from +0.85V. The MAX1795/MAX1796/MAX1797 have a preset, pin-selectable 5V or 3.3V output. The output can also be adjusted to other voltages, using two external resistors. The three devices differ only in their current limits, allowing optimization of external components for different loads: The MAX1795, MAX1796, and MAX1797 have current limits of 0.25A, 0.5A, and 1A, respectively. All devices are packaged in a compact 8-pin μ MAX package that is only 1.09mm tall and half the size of an 8-pin SO.

Applications

Portable Digital Audio Players
PDAs/Palmtops
Wireless Handsets
Portable Terminals

Pin Configuration



Features

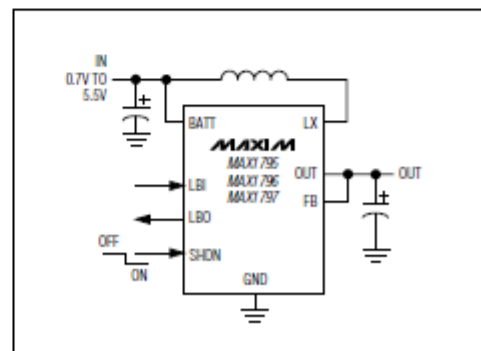
- ◆ >95% Efficiency
- ◆ True-Shutdown Circuitry
Output Disconnects from Input in Shutdown
No External Schottky Diode Needed
- ◆ 25 μ A Quiescent Supply Current
- ◆ Low-Noise Antiringing Feature
- ◆ LBI/LBO Comparator Enabled in Shutdown
- ◆ 2 μ A Shutdown Current
- ◆ 8-Pin μ MAX Package

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX1795EUA	-40°C to +85°C	8 μ MAX
MAX1796EUA	-40°C to +85°C	8 μ MAX
MAX1797EUA	-40°C to +85°C	8 μ MAX

True-Shutdown is a trademark of Maxim Integrated Products.

Typical Operating Circuit



Maxim Integrated Products 1

For price, delivery, and to place orders, please contact Maxim Distribution at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

Low Supply Current, Step-Up DC-DC Converters with True-Shutdown

ABSOLUTE MAXIMUM RATINGS

OUT, LX, SHDN, LBI, LBO, BATT to GND-0.3V to +6V	Operating Temperature Range-40°C to +85°C
FB-0.3V to (V _{OUT} + 0.3V)	Junction Temperature+150°C
I _{LX} , I _{OUT}±1.5A	Storage Temperature Range-65°C to +150°C
Output Short-Circuit Duration5s	Lead Temperature (soldering, 10s)+300°C
Continuous Power Dissipation			
8-Pin μ MAX (derate 4.1mW/°C above +70°C)330mW		

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V_{BATT} = +2V, OUT = FB (V_{OUT} = +3.3V), SHDN = LBI = GND, T_A = 0°C to +85°C, unless otherwise noted. Typical values are at T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Minimum Input Voltage		After startup		0.7		V
Operating Voltage	V _{BATT}	(Note 1)	1.0		5.5	V
Startup Voltage		T _A = +25°C, R _L = 3k Ω		0.85	1.0	V
Startup Voltage Tempco				-2.2		mV/°C
Output Voltage	V _{OUT}	FB = OUT	3.17	3.3	3.43	V
		FB = GND	4.80	5.0	5.20	
Adjustable Output Voltage Range			2.0		5.5	V
Steady-State Output Current	I _{OUT}	BATT = +2V, FB = OUT (V _{OUT} = +3.3V)	MAX1795	100	180	mA
			MAX1796	200	300	
			MAX1797	400	550	
		BATT = +2V, FB = GND (V _{OUT} = +5.0V)	MAX1795	50	120	
			MAX1796	100	200	
			MAX1797	250	370	
Feedback Set-Point Voltage (Adjustable Mode)	V _{FB}	V _{OUT} = +2V to +5.5V	1.20	1.24	1.28	V
Feedback Input Current	I _{FB}	V _{FB} = +1.24V		4	100	nA
Internal NFET, PFET On-Resistance	R _{DS(ON)}	V _{OUT} = +3.3V, I _{LX} = 100mA	NFET	0.17	0.3	Ω
			PFET	0.27	0.45	
LX Switch Current Limit (NFET only)	I _{LIM}	MAX1795	0.2	0.25	0.35	A
		MAX1796	0.4	0.5	0.625	
		MAX1797	0.8	1.0	1.25	
LX Leakage Current	I _{LEAK}	V _{LX} = 0 and +5.5V, V _{OUT} = +5.5V		0.2		μ A
Synchronous Rectifier Turn-Off Current Limit				25		mA
Damping Switch On-Resistance	R _{DAMP}		100	200	400	Ω
Operating Current into OUT (Note 2)		V _{FB} = +1.4V		25	45	μ A

Low Supply Current, Step-Up DC-DC Converters with True-Shutdown

ELECTRICAL CHARACTERISTICS (continued)

(V_{BATT} = +2V, OUT = FB (V_{OUT} = +3.3V), SHDN = LBI = GND, T_A = 0°C to +85°C, unless otherwise noted. Typical values are T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Operating Current into BATT		V _{FB} = +1.4V, V _{LBI} = +1V		2	4	μA
Shutdown Current into BATT		SHDN = BATT, V _{LBI} = +1V		2	4	μA
LX Switch Maximum On-Time	t _{ON}	V _{FB} = +1V, if current limit not reached	3	4	5	μs
LX Switch Minimum Off-Time	t _{OFF}	V _{FB} = +1V	0.8	1	1.2	μs
LBI Threshold Voltage Falling	V _{LBI}	V _{BATT} = +2V	0.8	0.85	0.90	V
		V _{BATT} = LBI	0.875	0.925	0.975	
LBI Hysteresis				25		mV
LBI Input Current	I _{LBI}	V _{LBI} = +0.8V		9	100	nA
LBO Low Output Voltage		V _{BATT} = V _{LBI} = +0.975V, sinking 20μA (50Ω typ)			0.1	V
		V _{BATT} = V _{LBI} = +1.1V, sinking 100μA (25Ω typ)			0.1	
LBO Off-Leakage Current		V _{LBO} = +5.5V		1	100	nA
SHDN Input Voltage	V _L				0.2 × V _{BATT}	V
	V _H		0.8 × V _{BATT}			
Shutdown Input Current		V _{SHDN} = 0 and +5.5V			100	nA

ELECTRICAL CHARACTERISTICS

(V_{BATT} = +2V, OUT = FB (V_{OUT} = +3.3V), SHDN = LBI = GND, T_A = -40°C to +85°C, unless otherwise noted.) (Note 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	MAX	UNITS
Operating Voltage	V _{BATT}	Note 1	1.0	5.5	V
Output Voltage	V _{OUT}	FB = OUT	3.13	3.47	V
		FB = GND	4.75	5.25	
Adjustable Output Voltage Range			2.0	5.5	V
Steady-State Output Current (Note 1)	I _{OUT}	FB = OUT (V _{OUT} = +3.3V)	MAX1795	100	mA
			MAX1796	200	
			MAX1797	400	
		FB = GND (V _{OUT} = +5.0V)	MAX1795	60	
			MAX1796	125	
			MAX1797	250	
Feedback Set-Point Voltage (Adjustable Mode)	V _{FB}	V _{OUT} = +2V to +5.5V	1.19	1.29	V
Feedback Input Current	I _{FB}	V _{FB} = +1.25V		100	nA

Low Supply Current, Step-Up DC-DC Converters with True-Shutdown

ELECTRICAL CHARACTERISTICS (continued)

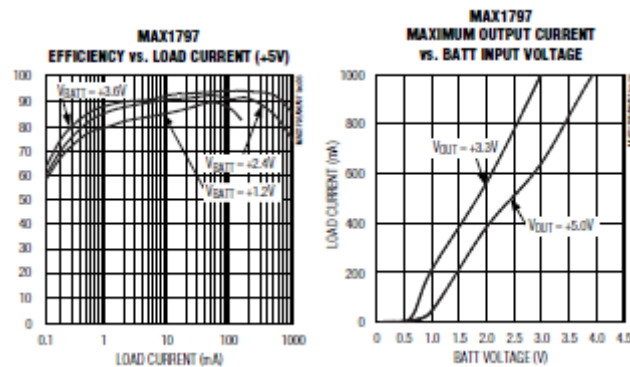
($V_{BATT} = +2V$, $OUT = FB$ ($V_{OUT} = +3.3V$), $SHDN = LBI = GND$, $T_A = -40^{\circ}C$ to $+85^{\circ}C$, unless otherwise noted.) (Note 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	MAX	UNITS
Internal NFET, PFET On-Resistance	$R_{DS(ON)}$	$V_{OUT} = +3.3V$, $I_{LX} = 100mA$	NFET	0.3	Ω
			PFET	0.45	
LX Switch Current Limit (NFET only)	I_{LIM}	MAX1795	0.19	0.37	A
		MAX1796	0.35	0.7	
		MAX1797	0.8	1.32	
LX Leakage Current	I_{LEAK}	$V_{LX} = 0$ and $+5.5V$, $V_{OUT} = +5.5V$			μA
Damping Switch On-Resistance	R_{DAMP}		100	400	Ω
Operating Current into OUT (Note 2)		$V_{FB} = +1.4V$		45	μA
Operating Current into BATT		$V_{FB} = +1.4V$, $V_{LBI} = +1V$		4	μA
Shutdown Current into BATT		$SHDN = BATT$, $V_{LBI} = +1V$		4	μA
LX Switch Maximum On-Time	t_{ON}	$V_{FB} = +1V$, if current limit not reached	2.75	5.25	μs
LX Switch Minimum Off-Time	t_{OFF}	$V_{FB} = +1V$	0.7	1.3	μs
LBI Threshold Voltage	V_{LBI}	$V_{BATT} = +2V$	0.8	0.90	V
		$V_{BATT} = LBI$	0.875	0.975	
LBI Input Current	I_{LBI}	$V_{LBI} = +0.8V$		100	nA
LBO Low Output Voltage		$V_{BATT} = V_{LBI} = +0.975V$, sinking $20\mu A$ (50 Ω typ)		0.1	V
		$V_{BATT} = V_{LBI} = +1.1V$, sinking $100\mu A$ (25 Ω typ)		0.1	
LBO Off-Leakage Current		$V_{LBO} = +5.5V$		100	nA
SHDN Input Voltage	V_{IL}			$0.2 \times V_{BATT}$	V
	V_{IH}		$0.8 \times V_{BATT}$		
Shutdown Input Current		$V_{SHDN} = 0$ and $+5.5V$		100	nA

Note 1: Operating Voltage: Since the regulator is bootstrapped to the output, once started it will operate down to a 0.7V input.

Note 2: Device is bootstrapped (power to IC comes from OUT). This correlates directly with the actual battery supply current.

Note 3: Specifications to $-40^{\circ}C$ are guaranteed by design, not production tested.



ANEXO D. DATOS RELEVANTES DATASHEET ETRX2 Y KIT DE DESARROLLO ETRXDVKA



ETRX2

ETRX2 Summary



Image not shown actual size; enlarged to show detail.

Module Features

- Small form factor, SMT module 37.5 x 20.5 x 3.2 mm
- Optional board-to-board or board-to-cable connector
- 3 RF output options: Integrated ceramic antenna, Hirose U.FL coaxial connector or single port 50Ω pad
- XAP2b microcontroller with non intrusive debug interface (SIF)
- 128k flash and 5kbytes of SRAM
- UART interface with DMA, hardware I²C and SPI accessible with custom firmware
- Wide supply voltage range (2.1 to 3.6V)
- Module ships with standard Telegesis AT-style software interface based on the EmberNet meshing stack.
- Can act as ZigBee End Device, Router or Coordinator
- 12 general-purpose I/O lines and 2 analogue inputs (all 17 GPIOs of the EM250 are accessible)
- Supports 4 different power modes for extended battery life
- Current consumption below 1μA in deep sleep mode with self wakeup
- Firmware upgrades via RS232 or over the air (password protected)
- Hardware supported encryption (AES-128)
- Tested for CE and FCC compliance (with integrated antenna), FCC modular approval
- Operating temperature range: -40 C to +85 C
- Options include: On board low power voltage regulator, DC/DC regulator and watch crystal

Radio Features

- Based on the Ember EM250 single chip ZigBee®/IEEE802.15.4 solution
- 2.4GHz ISM Band
- 250kbit/s over the air data rate – NB: actual usable data throughput with ZigBee is about 20kbps
- 16 channels (802.15.4 Channel 11 to 26)
- +3dBm output power (+5dBm in boost mode)
- High sensitivity of -98dBm typ. at 1% packet error rate
- Hardware acceleration for IEEE 802.15.4 compliant transmissions

The Telegesis ETRX2 module is a low power 2.4GHz ISM band transceiver based on the Ember EM250 single chip ZigBee®/IEEE802.15.4 solution. It has been designed to be integrated into any device without the need for RF experience and expertise. Utilizing the EmberZNet meshing and self-healing stack, the ETRX2 enables you to add powerful wireless networking capability to your products and quickly bring them to market. The module's unique AT-style command line interface allows you to quickly integrate meshing radio technology without complex software engineering.

Suggested Applications

- AMR – Automatic Meter Reading
- Wireless Alarms and Security
- Home/Building Automation
- Wireless Sensor Networks
- M2M Industrial Controls
- Future ZigBee® systems
- PC Peripherals
- IEEE 802.15.4 Systems
- Item Tracking

Development Kits

- Two complementary development kits consisting of two or three modules and a single development board with USB connectivity and I/O breakouts.
- AT-style software interface command dictionary can be modified for high volume customers.
- Custom software development available upon request.

Example AT-Style Commands

AT+BCAST	Sends a Broadcast
AT+UCAST:<address>	Sends a Unicast
AT+EN	Establish PAN network
AT+JN	Join PAN

At power-up the last configuration is loaded from non volatile S-Registers, which can eliminate the need for an additional host controller.

3 Module Pinout

The ETRX2 is pin compatible with the ETRX1, (NB: it has additional pins to the ETRX1). For all new designs using either ETRX1 or ETRX2 it is recommended that you use the ETRX2 footprint to ensure the option of future upgrading is guaranteed.



Figure 1: ETRX2 Module Pinout

The table below gives details about the 38 module pin signals for direct SMD soldering of the ETRX2 to the application board. The pin numbers shown in brackets () are the related pins of the EM250. In order to use the ETRX2 as a plug-in solution a Harwin 1.27mm pitch connector can be fitted on the bottom of the ETRX2 (Harwin part number M50-3601042). Other connectors that use the same footprint may also be used, such as a Samtec TFML-110-02-S-D.

All GND pads are connected within the module, but for best RF performance all of them should be grounded externally.

ETRX2 Pad	Function	EM250 GPIO	ETRX2 Harwin Pin
1	GND	GND	
2	Antenna		
3	GND	GND	
4	I/O9	GPIO 0 (21)	1
5	Vreg (1)		2
6	GND	GND	3
7	Vcc		10
8	GND	GND	3
9	A/D1	GPIO 4 (26)	4
10	A/D2	GPIO 5 (27)	5
11	I/O7	GPIO 3 (25)	6
12	I/O6	GPIO 2 (24)	7
13	I/O5	GPIO 1 (22)	8
14	I/O4 or RTS (3)	GPIO 12 (20)	9
15	GND	GND	
16	SIF CLK	SIF CLK	
17	SIF MISO	SIF MISO	
18	SIF MOSI	SIF MOSI	
19	SIF LOADB	SIF LOADB	
20	GND	GND	
21	I/O8	GPIO 6 (29)	11
22	I/O2 or CTS (3)	GPIO 11 (19)	12
23	I/O3	GPIO 13 (43)	13
24	Reset	(13)	14
25	I/O1	GPIO 14 (42)	15
26	I/O0	GPIO 8 (31)	16
27	TXD (2)	GPIO 9 (32)	18
28	RXD (2)	GPIO 10 (33)	17
29	GND	GND	3
30	I/O10	GPIO 15 (41)	19
31	I/O11	GPIO 16 (40)	20
32	GND	GND	
33	RXTXSW	(11)	
34	GND	GND	
35	GND	GND	
36	GND	GND	
37	GND	GND	
38	NC (4)	GPIO 7 (30)	

Table 1: Pin Information

Notes:

- {1} Where the onboard regulator option is mounted this pin is connected to the output voltage of the onboard regulator option and **NOT** to the output voltage VREG_OUT of the EM250
- {2} TXD is the UART output, RXD is the input
- {3} RTS/CTS handshaking is selectable in firmware. See the AT Command Manual.
- {4} Pad 38 is connected to GPIO7 but is not used by the Telegesis firmware

Valores de los registros más importantes.

S00 – Channel Mask	
Description The 802.15.4 channel mask.	Parameters XXXX
Operations R/W LOCAL R/W REMOTE	Where XXXX represents a 16-bit decimal number enabling IEEE802.15.4 channel numbers 11 to 26. Writing a bit to 1 enables a channel and subsequently writing a bit to 0 disables a channel for scanning, joining and establishing networks. e.g. when setting S00 to 0001, only channel 11 will be used for all following operations.
Becomes effective When Joining, Scanning or establishing a PAN	Range 0001 - FFFF
Note The channel mask does not affect the AT+JPAN command	Factory Default FFF0
Storage Non-Volatile	
SW release	Default changed in R210

S01 – Preferred PAN ID	
Description The 802.15.4 PAN ID.	Parameters XXXX
Operations R/W LOCAL R/W REMOTE	Where XXXX represents a 16-bit decimal number
Becomes effective When Joining or establishing a PAN	Range 0000 – 3FFF or FFFF
Storage Non-Volatile	When establishing a PAN the coordinator will pick a random PAN ID if S01 is set to FFFF. If set to any value between 0000 and 3FFF this number will be used as PAN ID instead, unless trying to establish a PAN which already exists on the same channel. In this case a random PAN ID will be used instead. When joining only PANs with the ID stored in S01 will be joined unless S01 is set to FFFF. In this case the next best PAN which allows joining is joined.
	Factory Default FFFF
SW release	R200

S15 – Immediate Functionality At IRQ0	
Description Describes the immediate action taken on IRQ0.	Parameters XXXX
Operations R/W LOCAL R/W REMOTE	If set to 0 the functionality is disabled. Please see section 5 for a list of the functionality.
Becomes effective Instantly	
Storage Non-Volatile	Factory Default 0001 (Wakeup to power mode 0)
SW release	R200

S16 – Immediate Functionality At IRQ1	
Description Describes the immediate action taken on IRQ1.	Parameters XXXX
Operations R/W LOCAL R/W REMOTE	If set to 0 the functionality is disabled. Please see section 5 for a list of the functionality.
Becomes effective Instantly	
Storage Non-Volatile	Factory Default 0001 (Wakeup to power mode 0)
SW release	R200

S06 – Main Function	
Description Defines the behaviour of the Device.	Parameters XXXX
Operations R/W LOCAL R/W REMOTE	Where XXXX represents a 16-bit hexadecimal number.
Becomes effective Instantly	Range 0000 to FFFF
Write operation ATS06=XXXX:<Password> ATSREM06:<EUI64>=xxxx: <Password>	Bit F: Read only. Set: Brownout Fuse is not set. (ETRX1 only, reserved on ETRX2) Bit E: Set: Remote S-Register Reads do not wait for ACK. Multiple nodes can therefore be interrogated in quick succession Bit D: Set: UCASTs and SCASTs wait for ACK Bit C: Set: Prompt user if a channel has been requested. <u>Unset:</u> Automatically accept channel Bit B: Set: Joining not permitted (overrides bits 0-3) Bit A: Set: Moving End Device (MED). <u>Unset:</u> Sleepy End Device (SED) (note: bit 9 must also be set) Bit 9: Set: Device is an End Device (bit A specifies Sleepy or Moving End Device) Bit 8: Device is Data Sink Bit 7: Set: Writing to S03 takes effect immediately (device will do a quick leave/re-init) Bit 6: Set: Display RSSI and LQI when devices report to AT+SN and AT+REMSN or AT+PING in Bit 5: Set: Attach Vcc reading to transmissions for functionalities 0100-0103, 0110-0113 and 2000 Bit 4: Set: When trying to join a network use network key in S03 (secured joining) Bit 3: <u>Set:</u> Allow unsecured joining via local node Bit 2: Set: Allow secured joining via local node Bit 1: <u>Set:</u> Allow unsecured joining via Trust Centre Bit 0: Set: Allow secured joining via Trust Centre
Note <ul style="list-style-type: none"> - Default settings are underlined - For security reasons this register is password protected. The default password on both ETRX1 and ETRX2 is "TG-ETRX1". - Changes to Bits 9 and 10 require leaving and joining the network again to take effect. - An end device cannot be a sink 	
Storage Non-Volatile	Factory Default 000C
SW release	R200

S0B – UART Setup	
<p>Description The device's RS232 Baudrate and mode. The default setting of 0500 results in: 19200bps, no parity, 1 stop bit, 8 data bits.</p> <p>Operations R/W LOCAL R/W REMOTE</p> <p>Becomes effective Instantly</p> <p>Note For the ETRXn Ember only guarantees proper operation of the stack up to 19200 baud. It is not recommended to use continuous data rates above 38400, even using hardware flow control as using higher data rates does not increase the actual throughput. If bit 5 is set, bi-directional Hardware Flow Control is used instead of XON/XOFF flow control. If using Hardware flow control I/O4 becomes the RTS output and the CTS input is assigned to either</p> <p>I/O3 (ETRX1) I/O2 (ETRX2)</p> <p>On the ETRX1 the CTS input can be moved to I/O2 for compatibility reasons by setting bit 7 of this register.</p> <p>Access to these I/Os via S0D,S0F is blocked whilst Hardware Flow control is active. Note that in case the 128-byte output buffer of the ETRXn is full data will be dropped.</p> <p>The parity settings do not affect the bytes transmitted over the air.</p>	<p>Parameters XXXX</p> <p>Where XXXX represents a 16-bit hexadecimal number.</p> <p>Range of the most significant byte 00 to 0C</p> <p>00: 1200 baud 01: 2400 baud 02: 4800 baud 03: 9600 baud 04: 14400 baud 05: 19200 baud 06: 28800 baud 07: 38400 baud 08: 50000 baud 09: 57600 baud 0A: 76800 baud 0B: 100000 baud 0C: 115200 baud (ETRX2 only)</p> <p>Range of the least significant byte 00 to FF</p> <p>bit 7 set: CTS is on I/O2 (ETRX1 only, ETRX2: reserved) bit 6 set: enable error prompts for parity error, rx-frame-error and rx-buffer overflow bit 5 set: H/W flow control enable bit 4 set: no command echo bit 3 set: 7 data bits instead of 8 bit 2 set: 2 stop bits instead of one bit 1 set: odd parity enabled bit 0 set: even parity enabled</p>
<p>Storage Non-Volatile</p> <p>SW release</p>	<p>Factory Default 0500</p> <p>Modified in R208</p>

S2E – Device Specific	
Description Defines the behaviour of the specific device. Functionality differs between ETRX1 and ETRX2.	Parameters XXXX Where XXXX represents a 16-bit hexadecimal number.
Operations R/W LOCAL R/W REMOTE	ETRX1: Bit F (MSB): Set: Behaviour of I/O3 is defined by S2F. <u>Unset:</u> Standard I/O pin. Bit E: Set: A/D readings from S12 and S13 represent the 10-bit reading rather than the voltage Bit D: Set: Pull-ups enabled globally Bit C – Bit 5: Reserved Bit 4: Set: Enable debouncing on IRQ pins(100ms) Bit 3: Set: IRQ1 on rising edge Bit 2: Set: IRQ1 on falling edge Bit 1: Set: IRQ0 on rising edge Bit 0: Set: IRQ0 on falling edge
Becomes effective Instantly	ETRX2: Bit F (MSB): Set: I/O3 is PWM as defined by S2F/S31. <u>Unset:</u> Standard I/O pin. Bit E – Bit D: Reserved Bit C: Set: Transmitter adds PAN ID to raw data packets. Receiver ignores packets with wrong ID <u>Unset:</u> PAN ID not used with raw data, devices in all PANs on the same radio channel receive data Bit B – Bit 9: Reserved Bit 8: Set: Enable Boost mode regardless of setting in S02 Bit 7: Set: I/O8 turns into A/D3, which can be read from S14 Bit 6: Set: Present A/D-Reference at I/O0 during measurement Bit 5: Set: Enable wakeup on UART activity Bit 4: Set: Enable debouncing on IRQ pins(100ms) Bit 3: Set: IRQ1 on rising edge Bit 2: <u>Set:</u> IRQ1 on falling edge Bit 1: Set: IRQ0 on rising edge Bit 0: <u>Set:</u> IRQ0 on falling edge
Note Default settings are underlined	
Storage Non-Volatile	Factory Default 0005
SW release	Last modified in R209

1 ETRX2DVKA – Development Kit Functional Summary



The Telegesis ETRX2DVKA Development Kit has been designed to enable fast and simple evaluation and development of the low cost, low power, meshing solution provided by the ETRX2 module.

Development Kit Features

This ETRX2 Development Kit provides genuine quick and easy, out-of-the-box construction of a working mesh network. Using the comprehensive AT-style Command Dictionary and the simple to use Telegesis Terminal software package provided, it should take less than half an hour to set up a three-way mesh network and to begin detailed, in-depth assessment of the latest EmberZNet meshing technology.

Included in the kit are both a full Development Board and two Module Carrier Boards with the following features:

- 2 LEDs
- 2 buttons
- On board 3V voltage regulator
- Operating Voltage 4 - 6V
- Temperature Sensor
- Light Sensor
- SIF Interface for programming and real time debugging of custom firmware

The carrier boards allow you to quickly experiment with sensors and actuators and are ideal for demonstrations and prototyping. They can either be battery or mains powered. The module's serial port can be accessed via pads or by plugging the Module Carrier Board onto the development board.

Module Carrier Board Pin/Function Table

I/O0 – BUTTON
 I/O1 – BUTTON
 I/O3 – BEEPER
 I/O5 – GREEN LED
 I/O7 – RED LED

Development Kit Contents

- 1 x ETRX2DV Development Board
- 1 x ETRX2HW Module with a Harwin 1.27mm pitch, 2x10 connector
- 2 x ETRX2MCB Module Carrier Boards fitted with ETRX2 Modules
- 2 x AA Battery Holders with leads
- 1 x USB Cable

The ETRX2DV Development Board

- Board Dimensions 100mm x 80mm
- USB to Serial Bridge
- On Board 3.3V Voltage regulator, powered from the USB bus
- Virtual COM port on the PC created by driver, allowing easy access to the AT-Command interface
- Breakout of selected pins of the ETRX2 modules
- 4 LEDs, 4 buttons, 2 potentiometers and a beeper which can be connected to the I/O of the ETRX2 module
- SIF Interface for programming and real time debugging of custom firmware
- The ETRX2 Module is connected to the Development board by the Harwin 1.27mm pitch surface mount header allowing a plug-in solution
- Reset and Bootload button

ANEXO E. COMUNICACIÓN I²C [8] Y UART

Se muestra en este anexo los detalles más relevantes del Bus I²C y puerto UART. El sensor utiliza un protocolo basado en I²C para la comunicación entre microcontrolador y sensor. Para la comunicación entre microcontrolador y transceptor se utiliza una comunicación a través del puerto UART.

E.1. Bus I²C

El bus I²C define en su nivel físico 2 líneas de transmisión. En una se transmiten los datos en serie (**S**erial **D**ata- SDA) y la otra línea es de reloj (**S**erial **C**lock – SCL). Ambas líneas requieren resistencias de pull-up ya que los dispositivos solo pueden poner la línea a nivel bajo. Permite las comunicaciones bidireccionales y tiene topología maestro – esclavo.

El nivel de enlace define la forma en la que se accede al bus, se transmiten los datos y se libera. Por defecto la línea siempre se encuentra a nivel alto y para comenzar una comunicación es necesaria una condición de inicio y al acabar la transmisión de datos una condición de fin.. Al tratarse de un bus es necesario el direccionamiento, por ello cada dispositivo debe tener una dirección.

Vemos a continuación en la figura E.1 la condición de inicio que ejecuta el maestro para empezar una comunicación y la condición de fin para finalizar.

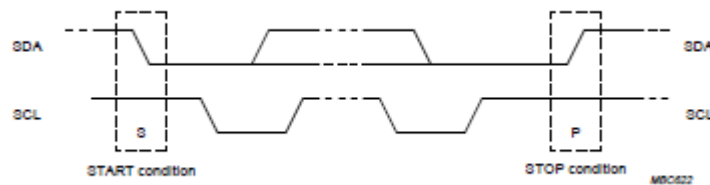


Fig. E.1 Condición de inicio y condición de fin

La transmisión de datos tiene un formato específico, tal y como se observa en la figura E.2. Como se puede ver, primero se envía el byte de dirección, donde el último byte se corresponde con el sentido de la transmisión (R/W – 0 escritura en esclavo, 1 lectura se corresponde con la solicitud de un dato al esclavo). Cada byte enviado de tener una respuesta en forma de ACK por parte de quien recibe el byte. A continuación se transmite byte a byte los datos y siempre primero el más significativo (MSB). No hay una restricción en el número de bytes a transmitir. Si un esclavo no puede recibir o transmitir otro byte puede mantener la línea SCL a nivel bajo obligando al maestro a esperar. Cuando este listo y puede transmitir, libera la línea de reloj.

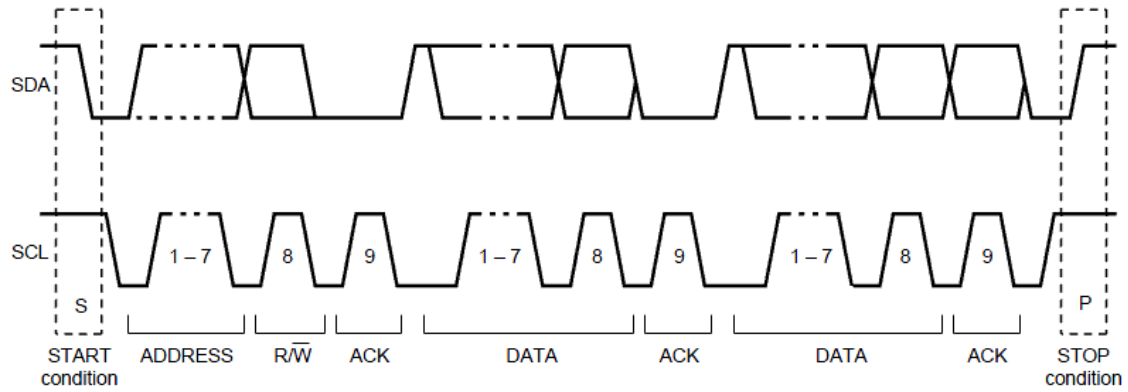
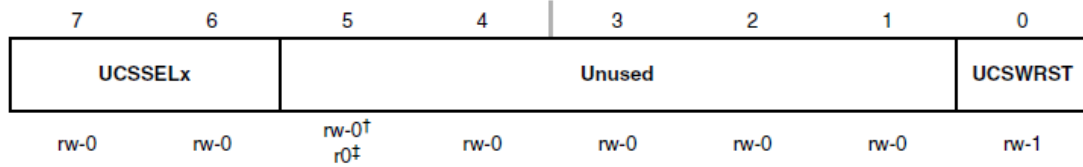


Fig. E.2 Transmisión completa datos en bus I²C

El bus I²C tiene características más complejas que permiten tener varios maestros en el bus y mediante arbitraje se determina quien transmite. También permite modos de transmisión de alta velocidad. No se entra en detalle de estas características ya que no tienen implicaciones dentro del proyecto.

Los registros utilizados en la configuración del puerto I²C en el microcontrolador son los siguientes:

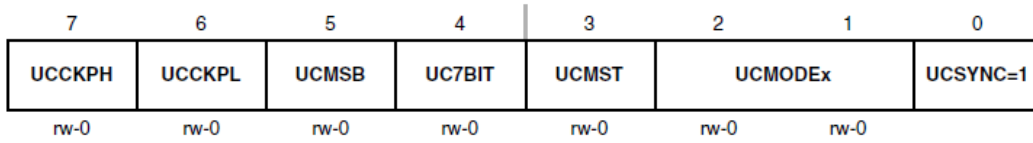
UCBxCTL1, USCI_Bx Control Register 1



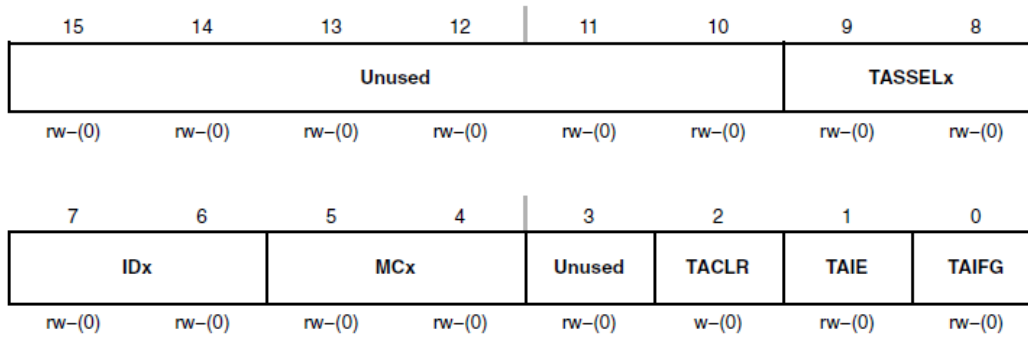
† UCAxCTL1 (USCI_Ax)

‡ UCBxCTL1 (USCI_Bx)

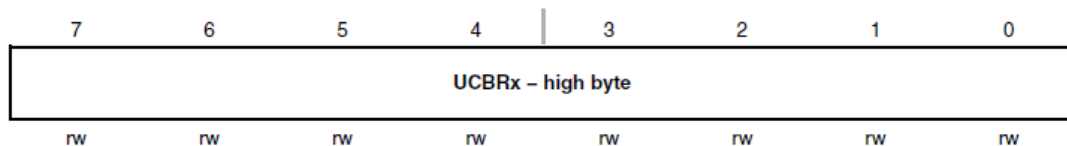
UCSSELx	Bits 7-6	USCI clock source select. These bits select the BRCLK source clock in master mode. UCxCLK is always used in slave mode. 00 NA 01 ACLK 10 SMCLK 11 SMCLK
Unused	Bits 5-1	Unused
UCSWRST	Bit 0	Software reset enable 0 Disabled. USCI reset released for operation. 1 Enabled. USCI logic held in reset state.

UCBxCTL0, USCI_Bx Control Register 0

UCCKPH	Bit 7	<p>Clock phase select.</p> <p>0 Data is changed on the first UCLK edge and captured on the following edge.</p> <p>1 Data is captured on the first UCLK edge and changed on the following edge.</p>
UCCKPL	Bit 6	<p>Clock polarity select.</p> <p>0 The inactive state is low.</p> <p>1 The inactive state is high.</p>
UCMSB	Bit 5	<p>MSB first select. Controls the direction of the receive and transmit shift register.</p> <p>0 LSB first</p> <p>1 MSB first</p>
UC7BIT	Bit 4	<p>Character length. Selects 7-bit or 8-bit character length.</p> <p>0 8-bit data</p> <p>1 7-bit data</p>
UCMST	Bit 3	<p>Master mode select</p> <p>0 Slave mode</p> <p>1 Master mode</p>
UCMODEx	Bits 2-1	<p>USCI mode. The UCMODEx bits select the synchronous mode when UCSYNC = 1.</p> <p>00 3-Pin SPI</p> <p>01 4-Pin SPI with UCxSTE active high: slave enabled when UCxSTE = 1</p> <p>10 4-Pin SPI with UCxSTE active low: slave enabled when UCxSTE = 0</p> <p>11 I²C Mode</p>
UCSYNC	Bit 0	<p>Synchronous mode enable</p> <p>0 Asynchronous mode</p> <p>1 Synchronous Mode</p>

TACTL, Timer_A Control Register

Unused	Bits 15-10	Unused
TASSELx	Bits 9-8	Timer_A clock source select 00 TACLK 01 ACLK 10 SMCLK 11 INCLK
IDx	Bits 7-6	Input divider. These bits select the divider for the input clock. 00 /1 01 /2 10 /4 11 /8
MCx	Bits 5-4	Mode control. Setting MCx = 00h when Timer_A is not in use conserves power. 00 Stop mode: the timer is halted. 01 Up mode: the timer counts up to TACCR0. 10 Continuous mode: the timer counts up to 0FFFFh. 11 Up/down mode: the timer counts up to TACCR0 then down to 0000h.
Unused	Bit 3	Unused
TACLr	Bit 2	Timer_A clear. Setting this bit resets TAR, the clock divider, and the count direction. The TACLr bit is automatically reset and is always read as zero.
TAIE	Bit 1	Timer_A interrupt enable. This bit enables the TAIFG interrupt request. 0 Interrupt disabled 1 Interrupt enabled
TAIFG	Bit 0	Timer_A interrupt flag 0 No interrupt pending 1 Interrupt pending

UCBxBR1, USCI_Bx Bit Rate Control Register 1

UCBRx Bit clock prescaler setting.
The 16-bit value of (UCxxBR0 + UCxxBR1 × 256) forms the prescaler value.

E.2. Comunicación UART

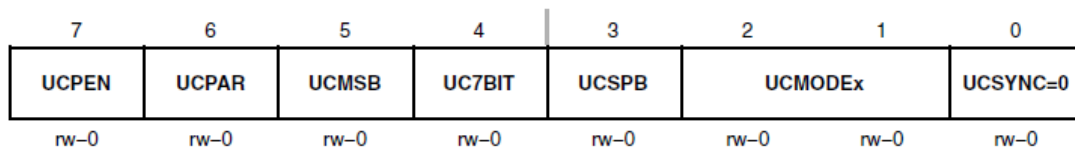
UART son las siglas de “Universal Asynchronous Receiver Transmitter”. Es una comunicación serie que se establece entre dispositivos. Los datos se transmiten como una secuencia de bits individuales y se recomponen en el otro lado de la comunicación para ser interpretados. La transmisión puede ser síncrona o asíncrona. Si soporta las comunicaciones síncronas se denomina USART.

- La transmisión serie síncrona requiere una señal de reloj compartida entre emisor y receptor o bien que el emisor proporcione una señal de temporización.
- La transmisión serie asíncrona es la más común y no se necesita ninguna señal de reloj. El emisor y receptor deben de ponerse de acuerdo en los parámetros de temporización.

Los parámetros básicos de configuración son los baudios (bits por segundo), tamaño de trama, paridad y bit de stop.

Para el microcontrolador MSP430F2132 los registros utilizados en la configuración del puerto serie son los siguientes.

UCAxCTL0, USCI_Ax Control Register 0



UCPEN	Bit 7	Parity enable 0 Parity disabled. 1 Parity enabled. Parity bit is generated (UCAxTXD) and expected (UCAxRXD). In address-bit multiprocessor mode, the address bit is included in the parity calculation.
UCPAR	Bit 6	Parity select. UCPAR is not used when parity is disabled. 0 Odd parity 1 Even parity
UCMSB	Bit 5	MSB first select. Controls the direction of the receive and transmit shift register. 0 LSB first 1 MSB first
UC7BIT	Bit 4	Character length. Selects 7-bit or 8-bit character length. 0 8-bit data 1 7-bit data
UCSPB	Bit 3	Stop bit select. Number of stop bits. 0 One stop bit 1 Two stop bits
UCMODEx	Bits 2-1	USCI mode. The UCMODEx bits select the asynchronous mode when UCSYNC = 0. 00 UART Mode. 01 Idle-Line Multiprocessor Mode. 10 Address-Bit Multiprocessor Mode. 11 UART Mode with automatic baud rate detection.
UCSYNC	Bit 0	Synchronous mode enable 0 Asynchronous mode 1 Synchronous Mode

Table 15–4. Commonly Used Baud Rates, Settings, and Errors, UCOS16 = 0

BRCLK frequency [Hz]	Baud Rate [Baud]	UCBRx	UCBRsX	UCBRFx	Max. TX Error [%]	Max. RX Error [%]		
32,768	1200	27	2	0	-2.8	1.4	-5.9	2.0
32,768	2400	13	6	0	-4.8	6.0	-9.7	8.3
32,768	4800	6	7	0	-12.1	5.7	-13.4	19.0
32,768	9600	3	3	0	-21.1	15.2	-44.3	21.3
1,048,576	9600	109	2	0	-0.2	0.7	-1.0	0.8
1,048,576	19200	54	5	0	-1.1	1.0	-1.5	2.5
1,048,576	38400	27	2	0	-2.8	1.4	-5.9	2.0
1,048,576	56000	18	6	0	-3.9	1.1	-4.6	5.7
1,048,576	115200	9	1	0	-1.1	10.7	-11.5	11.3
1,048,576	128000	8	1	0	-8.9	7.5	-13.8	14.8
1,048,576	256000	4	1	0	-2.3	25.4	-13.4	38.8
1,000,000	9600	104	1	0	-0.5	0.6	-0.9	1.2
1,000,000	19200	52	0	0	-1.8	0	-2.6	0.9
1,000,000	38400	26	0	0	-1.8	0	-3.6	1.8
1,000,000	56000	17	7	0	-4.8	0.8	-8.0	3.2
1,000,000	115200	8	6	0	-7.8	6.4	-9.7	16.1
1,000,000	128000	7	7	0	-10.4	6.4	-18.0	11.6
1,000,000	256000	3	7	0	-29.6	0	-43.6	5.2
4,000,000	9600	416	6	0	-0.2	0.2	-0.2	0.4
4,000,000	19200	208	3	0	-0.2	0.5	-0.3	0.8
4,000,000	38400	104	1	0	-0.5	0.6	-0.9	1.2
4,000,000	56000	71	4	0	-0.6	1.0	-1.7	1.3
4,000,000	115200	34	6	0	-2.1	0.6	-2.5	3.1
4,000,000	128000	31	2	0	-0.8	1.6	-3.6	2.0
4,000,000	256000	15	5	0	-4.0	3.2	-8.4	5.2
8,000,000	9600	833	2	0	-0.1	0	-0.2	0.1
8,000,000	19200	416	6	0	-0.2	0.2	-0.2	0.4
8,000,000	38400	208	3	0	-0.2	0.5	-0.3	0.8
8,000,000	56000	142	7	0	-0.6	0.1	-0.7	0.8
8,000,000	115200	69	4	0	-0.6	0.8	-1.8	1.1
8,000,000	128000	62	4	0	-0.8	0	-1.2	1.2
8,000,000	256000	31	2	0	-0.8	1.6	-3.6	2.0

ANEXO F. DATOS RELEVANTES TRANSISTOR BSS123 UTILIZADOS EN ADAPTACIÓN DE NIVEL BUS I²C



June 2003

BSS123

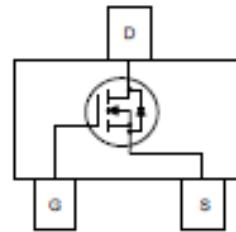
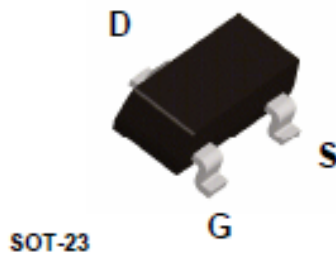
N-Channel Logic Level Enhancement Mode Field Effect Transistor

General Description

These N-Channel enhancement mode field effect transistors are produced using Fairchild's proprietary, high cell density, DMOS technology. These products have been designed to minimize on-state resistance while provide rugged, reliable, and fast switching performance. These products are particularly suited for low voltage, low current applications such as small servo motor control, power MOSFET gate drivers, and other switching applications.

Features

- 0.17 A, 100 V. $R_{DS(ON)} = 6\Omega @ V_{GS} = 10\text{ V}$
 $R_{DS(ON)} = 10\Omega @ V_{GS} = 4.5\text{ V}$
- High density cell design for extremely low $R_{DS(ON)}$
- Rugged and Reliable
- Compact industry standard SOT-23 surface mount package



Absolute Maximum Ratings

$T_A = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Ratings	Units
V_{DS}	Drain-Source Voltage	100	V
V_{GS}	Gate-Source Voltage	± 20	V
I_D	Drain Current – Continuous	0.17	A
	– Pulsed	0.68	
P_D	Maximum Power Dissipation	0.36	W
	Derate Above 25°C	2.8	
T_A, T_{STG}	Operating and Storage Junction Temperature Range	-55 to $+150$	$^\circ\text{C}$
T_L	Maximum Lead Temperature for Soldering Purposes, $1/16"$ from Case for 10 Seconds	300	

Thermal Characteristics

$R_{\theta JA}$	Thermal Resistance, Junction-to-Ambient	(Note 1) 350	$^\circ\text{C/W}$
-----------------	---	--------------	--------------------

Package Marking and Ordering Information

Device Marking	Device	Reel Size	Tape width	Quantity
SA	BSS123	7"	8mm	3000 units

Electrical CharacteristicsT_A = 25°C unless otherwise noted

Symbol	Parameter	Test Conditions	Min	Typ	Max	Units
Off Characteristics						
BV _{DSS}	Drain–Source Breakdown Voltage	V _{GS} = 0 V, I _D = 250 μA	100			V
$\frac{\Delta BV_{DSS}}{\Delta T_J}$	Breakdown Voltage Temperature Coefficient	I _D = 250 μA, Referenced to 25°C		97		mV/°C
I _{DSS}	Zero Gate Voltage Drain Current	V _{DS} = 100 V, V _{GS} = 0 V			1	μA
		V _{DS} = 100 V, V _{GS} = 0 V, T _J = 125°C			60	μA
		V _{DS} = 20 V, V _{GS} = 0 V			10	nA
I _{DSS}	Gate–Body Leakage	V _{GS} = ±20 V, V _{DS} = 0 V			±50	nA
On Characteristics (Note 2)						
V _{GS(th)}	Gate Threshold Voltage	V _{DS} = V _{GS} , I _D = 1 mA	0.8	1.7	2	V
$\frac{\Delta V_{GS(th)}}{\Delta T_J}$	Gate Threshold Voltage Temperature Coefficient	I _D = 1 mA, Referenced to 25°C		–2.7		mV/°C
R _{DS(on)}	Static Drain–Source On–Resistance	V _{GS} = 10 V, I _D = 0.17 A V _{GS} = 4.5 V, I _D = 0.17 A V _{GS} = 10 V, I _D = 0.17 A, T _J = 125°C		1.2 1.3 2.2	6 10 12	Ω
I _{D(on)}	On–State Drain Current	V _{GS} = 10 V, V _{DS} = 5 V	0.68			A
g _{FS}	Forward Transconductance	V _{DS} = 10V, I _D = 0.17 A	0.08	0.8		S
Dynamic Characteristics						
C _{iss}	Input Capacitance	V _{DS} = 25 V, V _{GS} = 0 V, f = 1.0 MHz		73		pF
C _{oss}	Output Capacitance			7		pF
C _{rss}	Reverse Transfer Capacitance			3.4		pF
R _G	Gate Resistance	V _{GS} = 15 mV, f = 1.0 MHz		2.2		Ω
Switching Characteristics (Note 2)						
t _{d(on)}	Turn–On Delay Time	V _{DD} = 30 V, I _D = 0.28 A, V _{GS} = 10 V, R _{GEN} = 6 Ω		1.7	3.4	ns
t _r	Turn–On Rise Time			9	18	ns
t _{d(off)}	Turn–Off Delay Time			17	31	ns
t _f	Turn–Off Fall Time			2.4	5	ns
Q _g	Total Gate Charge	V _{DS} = 30 V, I _D = 0.22 A, V _{GS} = 10 V		1.8	2.5	nC
Q _{gs}	Gate–Source Charge			0.2		nC
Q _{gd}	Gate–Drain Charge			0.3		nC
Drain–Source Diode Characteristics and Maximum Ratings						
I _S	Maximum Continuous Drain–Source Diode Forward Current				0.17	A
V _{SD}	Drain–Source Diode Forward Voltage	V _{GS} = 0 V, I _S = 0.34 A (Note 2)		0.8	1.3	V
t _{rr}	Diode Reverse Recovery Time	I _F = 0.17 A,		11		nS
Q _{rr}	Diode Reverse Recovery Charge	d _I /d _t = 100 A/μs		3		nC

NOTE:

1. R_{thJA} is the sum of the junction-to-case and case-to-ambient thermal resistance where the case thermal reference is defined as the solder mounting surface of the drain pins. R_{thJC} is guaranteed by design while R_{thCA} is determined by the user's board design.



a) 350°C/W when mounted on a minimum pad.

Scale 1 : 1 on letter size paper

2. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2.0%

El transistor escogido es del tipo MOSFET (Metal-oxid semiconductor Field effect transistor) ya que conmutan mediante diferencia de tensión. En nuestro caso esta variación se produce entre la puerta **G** (donde esta conectado VCC) y la fuente **S** (donde esta conectada la señal de datos del microcontrolador). El utilizado es del tipo enriquecido porque el umbral de conmutación siempre será superior a 0 y de canal n porque trabaja con tensiones positivas.

Los dispositivos conectados al bus I²C tienen la tecnología de circuito colector abierto. Este tipo de circuito permite múltiples dispositivos con comunicación bidireccional en un solo cable. Operan con resistencias de pull-up externas para mantener la señal en alto hasta que un dispositivo dentro del bus pone la línea a 0 y todos los dispositivos conectados son inducidos a este nivel.

Los dispositivos normalmente integran transistores BJC, que a diferencia de los MOSFET conmutan con diferencia de intensidad. Se puede decir que este tipo de circuito se comporta como un AND lógico. Así que teniendo un único dispositivo a 0, todo el bus irá a ese nivel de tensión.

ANEXO G. ENTORNO DE DESARROLLO, PLACA DE PRUEBAS MSP430FG4618/F2013 Y HERRAMIENTA DE DEPURACIÓN MSP-FET430UIF

Con la finalidad de determinar todas las necesidades y al disponer de una placa de prueba en el laboratorio, se comenzó el desarrollo del código sobre la placa de experimentación MSP430FG4618/F2013.

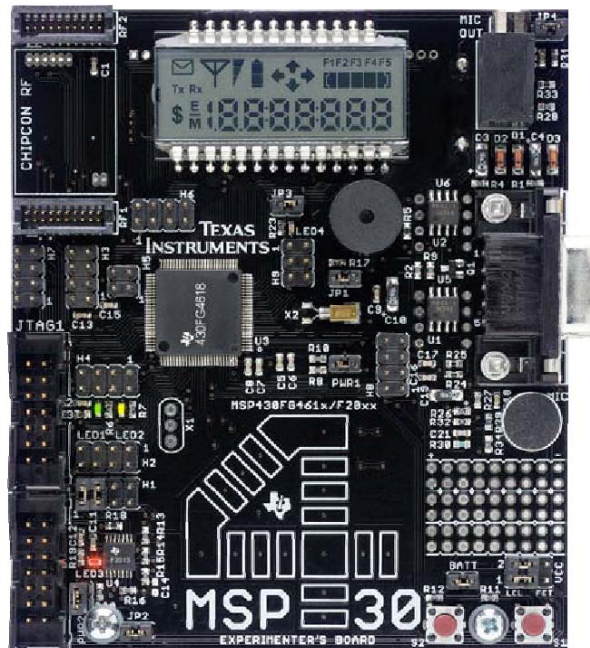


Fig. G.1 Placa de experimentación MSP430F4618/F2013 [21]

El kit de desarrollo contiene, aparte de la placa, una herramienta para cargar el código al microcontrolador y depurar, el MSP-FET430UIF. Con ambos dispositivos se han podido desarrollar aplicaciones de prueba que realizan funciones similares al código final, tales como la comunicación I²C, UART, uso de interrupciones, mostrar mensajes en display, reset mediante botón, lectura de entradas digitales o activación de salidas (encendido y parando LEDs). Con esta experiencia previa ya se pueden determinar todas las características que debe cumplir el microcontrolador. Se han implementado algunas de los programas recomendados en libro citado en la bibliografía con la finalidad de obtener las nociones básicas en el desarrollo de aplicaciones con la familia de microcontroladores MSP430. El entorno de desarrollo utilizado es el IAR Embedded Workbench, distribuido junto a la placa de pruebas y que permite la selección del tipo de microcontrolador utilizado. Junto al entorno de desarrollo se suministran librerías que facilitan la utilización de los registros del microcontrolador tales como IO430X21X2.h. Esta librería es una interfaz intermedia que define los registros estandarizados y bits de forma que su utilización en el código sea más intuitiva y sencilla.

Ejemplos de definición de bits y registros:

```
/*-----
 * Standard Bits
 *-----*/
```

```
#define BIT0      (0x0001)
#define BIT1      (0x0002)
#define BIT2      (0x0004)
#define BIT3      (0x0008)
#define BIT4      (0x0010)
#define BIT5      (0x0020)
#define BIT6      (0x0040)
#define BIT7      (0x0080)
#define BIT8      (0x0100)
#define BIT9      (0x0200)
#define BITA      (0x0400)
#define BITB      (0x0800)
#define BITC      (0x1000)
#define BITD      (0x2000)
#define BITE      (0x4000)
#define BITF      (0x8000)
```

```
/*-----
 * Status register bits
 *-----*/
```

```
#define C          (0x0001)
#define Z          (0x0002)
#define N          (0x0004)
#define V          (0x0100)
#define GIE        (0x0008)
#define CPUOFF     (0x0010)
#define OSCOFF     (0x0020)
#define SCG0       (0x0040)
#define SCG1       (0x0080)
```

/* Low Power Modes coded with Bits 4-7 in SR */

```
#define LPM0_bits  (CPUOFF)
#define LPM1_bits  (SCG0+CPUOFF)
#define LPM2_bits  (SCG1+CPUOFF)
#define LPM3_bits  (SCG1+SCG0+CPUOFF)
#define LPM4_bits  (SCG1+SCG0+OSCOFF+CPUOFF)
```

La placa de pruebas incorpora ciertos elementos que facilitan el desarrollo. En las aplicaciones desarrolladas a modo de aprendizajes se ha utilizado:

- Los pulsadores como interrupciones externas o entradas digitales y los LEDs como salidas.

- El LCD para mostrar los datos (son necesarias 2 librerías para facilitar la implementación, lcdutils2.h y lcdutils2.c).
- El puerto RS-232 para transmitir al PC.
- El puerto I²C para comunicar entre microcontroladores.
- Cristal externo para timer 0.

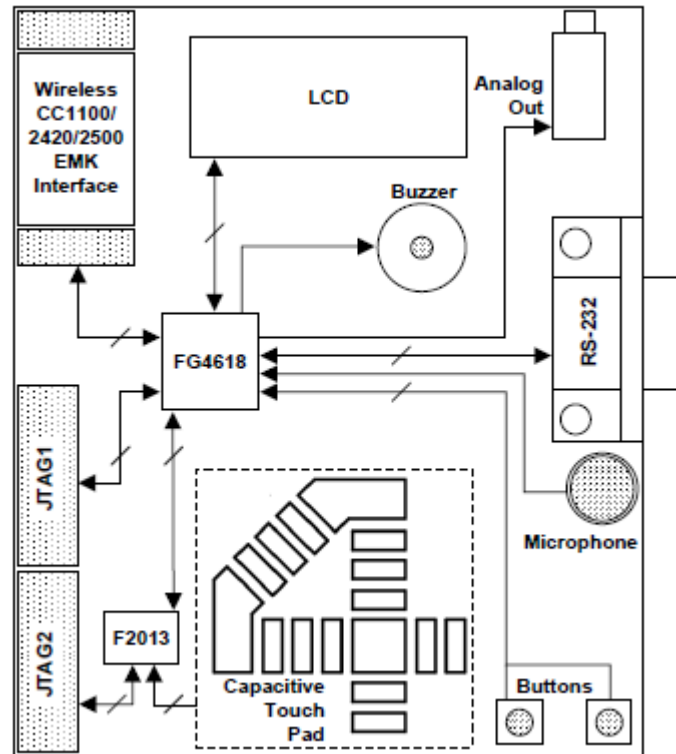
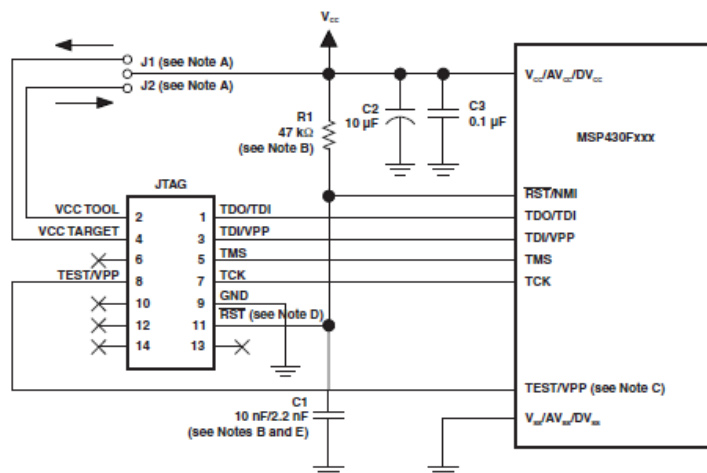


Fig. G.2 Diagrama bloques placa pruebas

Mediante el puerto JTAG se conecta el MSP-FET430UIF. Este dispositivo, conectado mediante interfaz USB al PC permite cargar el código en el microcontrolador y depurar durante la ejecución. Facilita enormemente el desarrollo de aplicaciones. Es por este motivo que se incorpora el puerto JTAG en el diseño final de la PCB.



Fig. G.3 MSP-FET430 UIF



- A Make either connection J1 in case a local target power supply is used or connection J2 to power target from the debug/programming adapter.
- B The RST/NMI pin R1/C1 configuration is device family dependent. See the respective MSP430 family user's guide for the recommended configuration.
- C The TEST pin is available only on MSP430 family members with multiplexed JTAG pins. See the device-specific data sheet to determine if this pin is available.
- D The connection to the JTAG connector RST pin is optional when using 4-wire JTAG communication mode capable-only devices and not required for device programming or debugging. However, this connection is required when using 2-wire JTAG communication mode capable devices in 4-wire JTAG mode.
- E When using 2-wire JTAG communication capable devices in 4-wire JTAG mode, the upper limit for C1 should not exceed 2.2 nF. This applies to both TI FET interface modules (LPT/USB FET).

Figure 2-1. Signal Connections for 4-Wire JTAG Communication

Fig. G.4 Conexión JTAG [22]

Se muestra un imagen de la aplicación realizada en Visual Basic 6 para leer los valores de CO2 por UART y extraer gráficos Excel. Con estos datos, ya en el inicio del proyecto se pudo determinar la variabilidad de los datos obtenidos durante el warm-up.

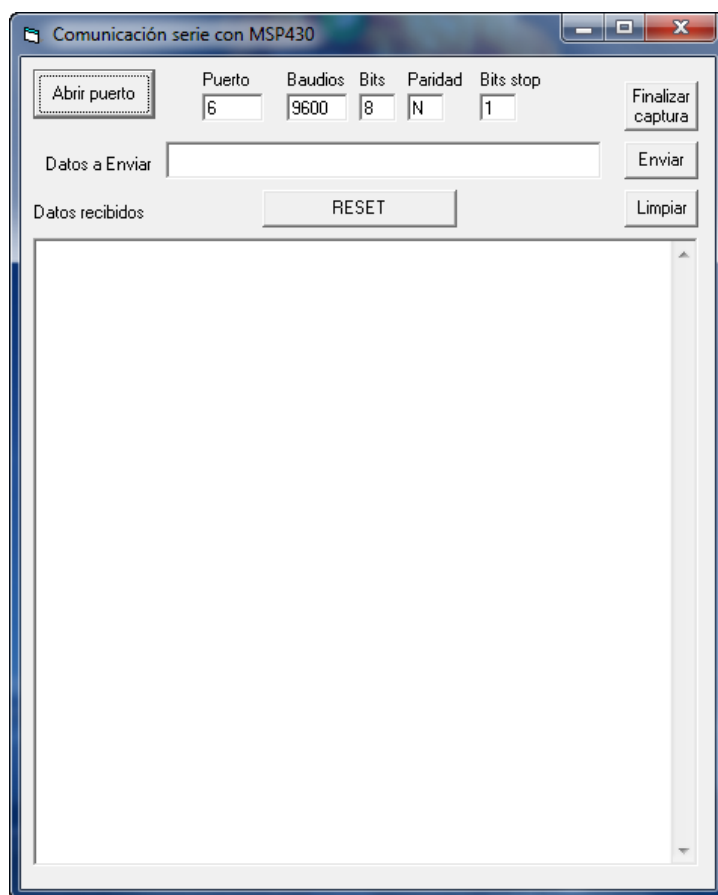


Fig. G.5 Interfaz aplicación Visual Basic 6

ANEXO H. CAPTURAS DURANTE TRANSMISIÓN DE DATOS VARIANDO RESISTENCIAS DE PULL-UP EN BUS I²C

En este anexo se muestran las capturas obtenidas durante las pruebas para obtener los valores de resistencias de pull-up adecuadas.

Los datos obtenidos en la tabla 1.5 han sido extraídos de las capturas que se muestran a continuación. Para todas las capturas mostradas, el canal 1 se corresponde con la señal medida en la sección 1 de bajo voltaje (3V) y el canal 2 con la señal de la sección 2 de alto voltaje (5V).

Comenzando con las resistencias más bajas, podemos observar la figura H.2, con 3,3 k Ω y donde el nivel bajo puesto por el sensor no es detectado por el microcontrolador.

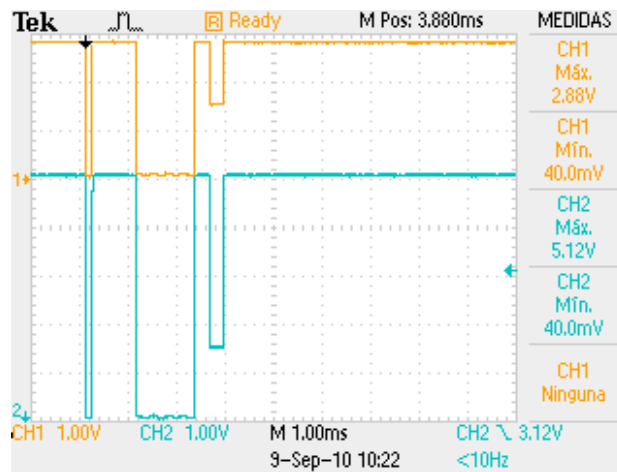


Fig. H.1 Transmisión I²C con resistencia pull-up 3,3 k Ω

Continuando con 10 k Ω , la comunicación es correcta y detectada por ambos dispositivos. Como podemos observar hay diferencia en los niveles bajos según transmita el maestro o el esclavo. Esto puede ser debido a los transistores internos de cada dispositivo y no se ha podido analizar en profundidad.

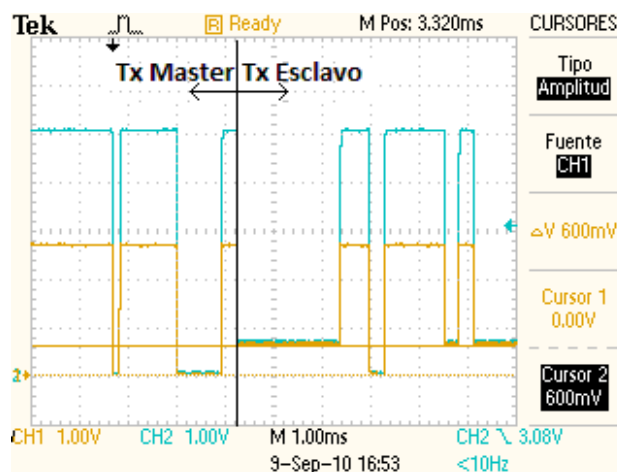


Fig. H.2 Transmisión I²C con resistencia pull-up 10 kΩ

Se muestra en la figura H.3 una captura donde se aprecia la diferencia entre los niveles bajos al transmitir el maestro y el esclavo con las resistencias de 20kΩ.

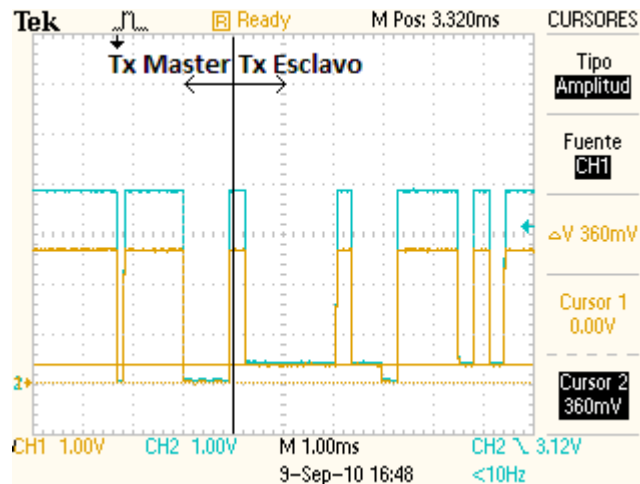


Fig H.3 Transmisión I²C resistencias pull-up de 20 kΩ

En la siguiente captura se observa la señal en la zona del maestro (canal 1) y del esclavo (canal 2). Se observa que el nivel alto en la zona del esclavo está en los 3,72 V.

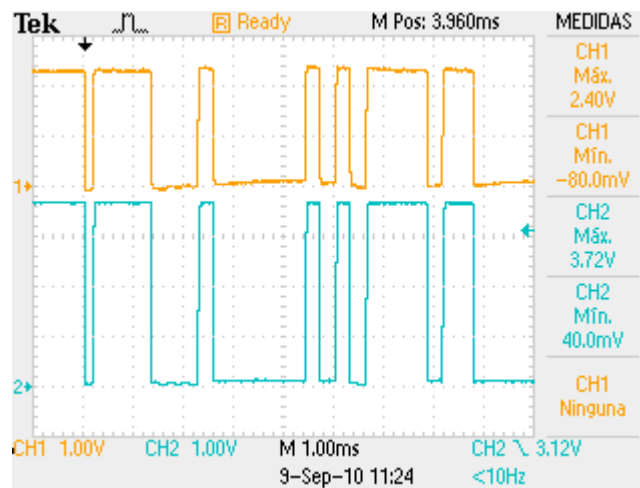


Fig H.4 Señal con resistencias pull-up de 100 kΩ

ANEXO I. CAPTURAS DURANTE LA TRANSMISIÓN ZIGBEE

Se muestra a continuación el consumo del transceptor durante la transmisión de datos al ordenador a través de Zigbee. Con la sonda de corriente conectada a la entrada del transceptor se realizan las siguientes capturas y se detallan los tiempos de cada pulso a máximo consumo. En la figura 4.7 se muestra el perfil completo del consumo. Repetimos la captura pero midiendo el ancho total del pulso en la captura I.1, obteniendo un tiempo de actividad de 990ms. Hay un total de 5 picos, cada uno con un tiempo diferente. En las figuras I.2, I.3, I.4, I.5 y I.6 se observa el detalle del ancho del pulso. Los resultados se exponen en el capítulo 4.2.2.

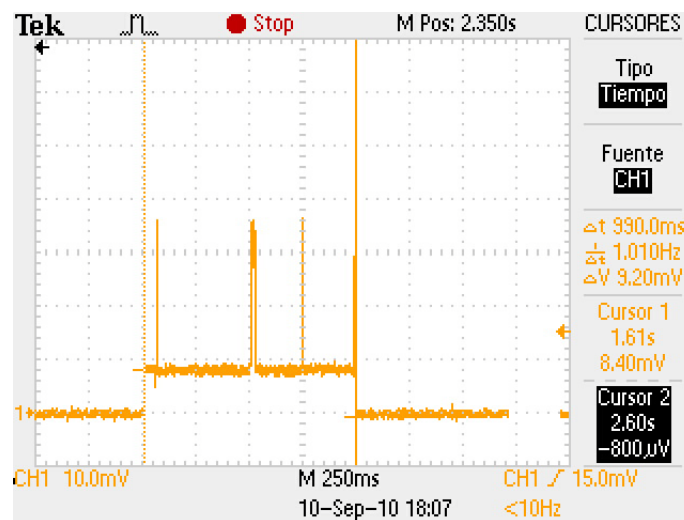


Fig. I.1 Tiempo de actividad en transmisión Zigbee

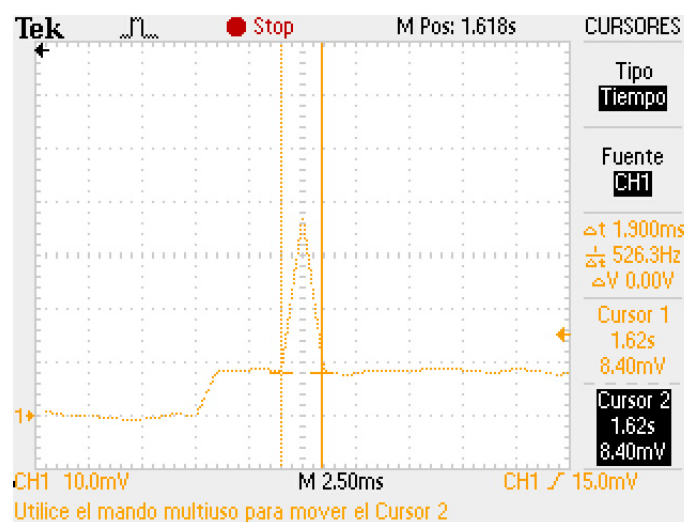


Fig. I.2 Ancho pico 1

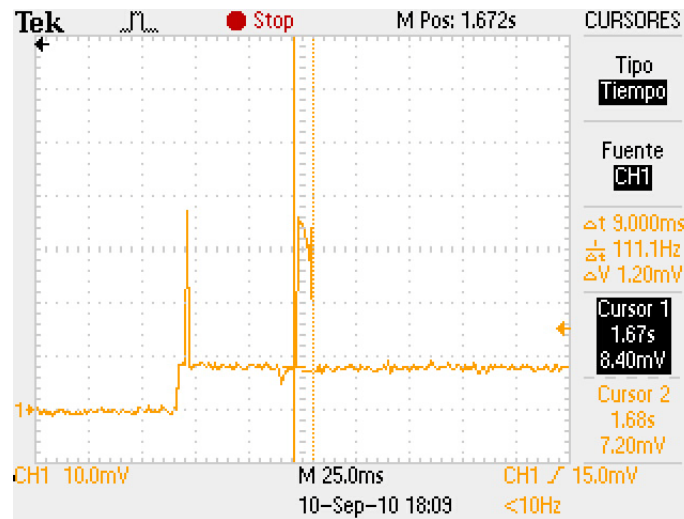


Fig. I.3 Ancho pulso 2

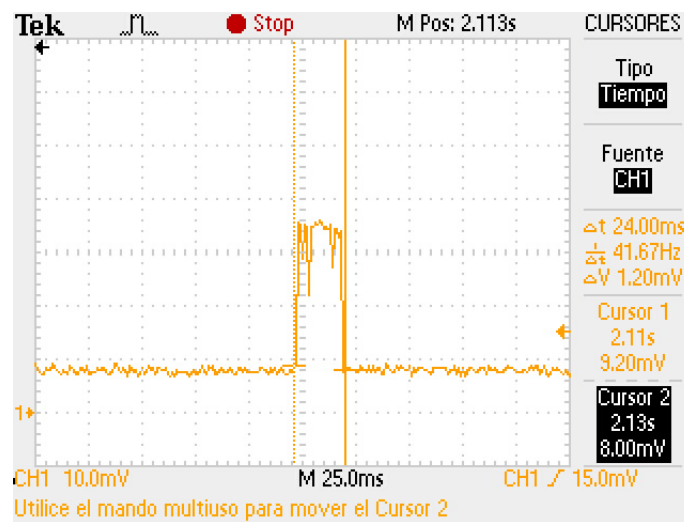


Fig. I.4 Ancho pulso 3

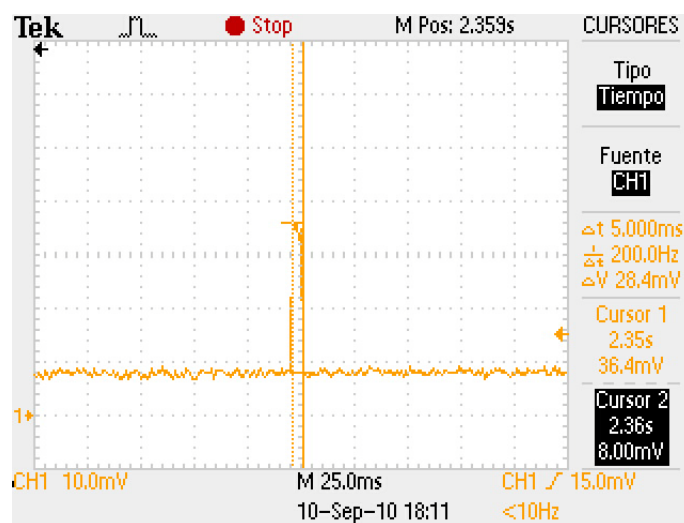


Fig. I.5 Ancho pulso 4

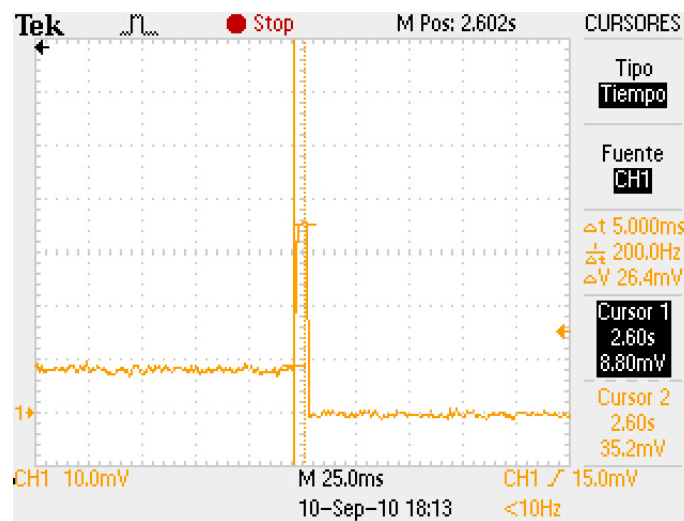


Fig. I.6 Ancho pulso 5

Finalmente, vemos en la captura I.7 el descenso del consumo en el momento de la interrupción que activa el modo de bajo consumo en el transceptor. En el canal 2, en azul, se observa la activación de la salida digital. En el canal 1, de amarillo, se observa como decrece el consumo.

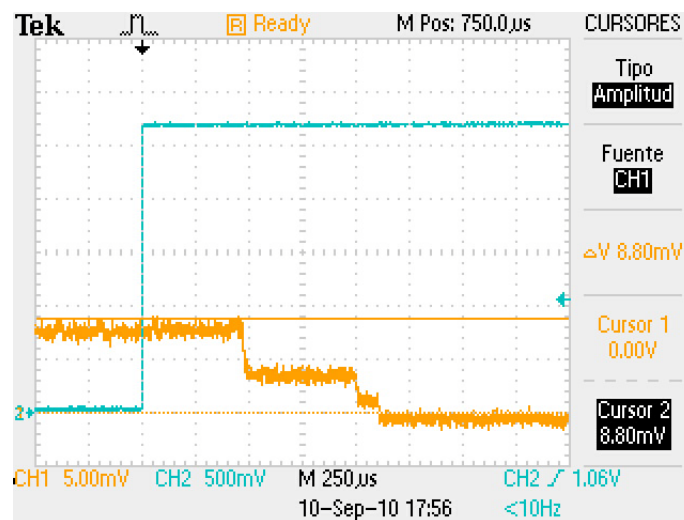


Fig. I.7 Detalle activación modo bajo consumo

En la figura I.8 se observa el perfil de consumo del transceptor Zigbee sin realizar transmisión de datos durante el modo activo. Cada segundo se produce un pico de consumo de 36 mA y el resto del tiempo se mantiene en unos 8 mA. Al entrar el modo de bajo consumo, se observa un perfil completamente plano y no se producen estos picos.

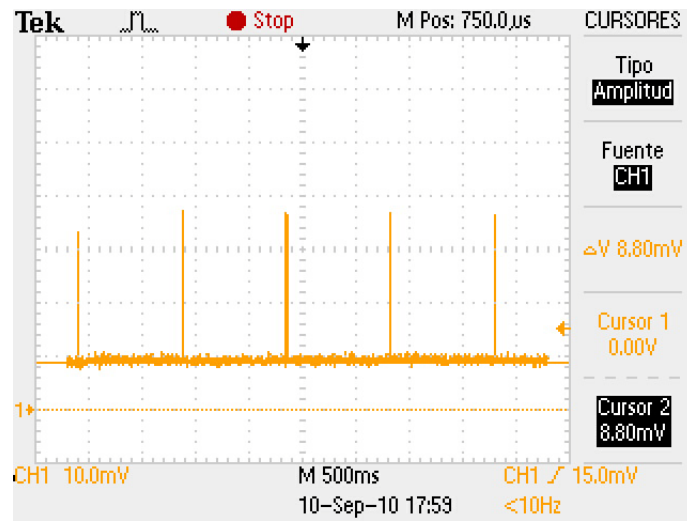


Fig. I.8 Perfil de consumo de transceptor sin transmisión en modo activo

ANEXO J LISTADO DE COMPONENTES DEL NODO SENSOR Y PRECIOS [23]

Tabla J.1 Listado de precios

Componente	Cantidad	Subtotal	Total
Microcontrolador MSP430F2132	1	4,27	4,27
Cristal 32 KHz	1	1,63	1,63
Resistencia R6,R7 100K Ω	2	0,0125	0,025
Regulador MAX1797	1	10,5	10,50
Bobina Coilcraft DS3316p-223 [24]	1	1,09	1,09
Condensador C1, C2, Rubycon 6.3TZV1000M8X10.5	2	1,26	2,52
Resistencia R8 1M Ω	1	0,0125	0,0125
Sensor EE89 [25]	1	139	139
Conector 1x5 pines	2	0,12	0,24
Conector 1x4 pines	1	0,12	0,12
ETRX2	1	22,81	22,81
Conector 2x10 pines	1	0,26	0,26
Transistor MOSFET BSS123	2	0,32	0,64
Resistencia R2, R3, R4, R5 20 K Ω	4	0,0125	0,05
Conector 2x7 pines	1	0,12	0,12
Resistencia R1 47k Ω	1	0,0125	0,0125
Condensador C3, 5pF	1	0,016	0,016
Condensador C4, 0,1 μ F	1	0,021	0,021
Condensador C5, 10 μ F	1	0,92	0,92
Conector 2x1 pines	3	0,12	0,36
Portapilas 2xAA	1	3,04	3,04
Pilas Energizer L91	2	3,44	6,88
IMPORTE TOTAL [€]			194,53