

**GATEWAY Y DE COMUNICACIÓN PARA REDES
DE SENsoRES INALÁMBRICOS CON EL USO
DE LA PLACA C.I.A.A.
(GATEWAY Y CIAA)
PROYECTO FINAL**

Version 1.3
<2019>

ESCRIBANO, LUCAS A.
Legajo: 32167
Email: lucasescribano@hotmail.com.ar

GONZALEZ, RODRIGO J.
Legajo: 27134
Email: rodjergon@gmail.com



Agradecimientos

Este pequeño apartado es para dar mi más humilde agradecimiento a todos aquellos que han hecho posible la finalización de este trabajo.

Quiero agradecer principalmente a las personas más importantes que tengo que son mis padres por ser los principales promotores de mis sueños, gracias a ellos por cada día confiar, creer, y querer el mejor futuro para mi.

Gracias a mi pareja por entenderme en todo, gracias a ella porque en todo momento fue un apoyo incondicional en mi vida.

A mi hijo, te agradezco por los momentos sacrificados en nuestra vida como familia, que requirió el cumplimiento de este proyecto.

También agradecer a cada uno de los profesores que colaboraron para que de alguna manera u otra nuestro proyecto final pudiera concretarse.

No debemos dejar de lado a muchos compañeros que también han aportado su gran ayuda para que se vea plasmado nuestro sacrificio en este proyecto final.

Agradecer a la Institución Universidad Tecnológica Facultad Regional Mendoza gracias por habernos permitido formarnos en ella, gracias a todas las personas que fueron participes de este proceso, ya sea de manera directa o indirecta, sin olvidarnos también de los profesionales en esta gran carrera que es la Ingeniería en Electrónica que a partir de ahora y para siempre va a formar parte de nuestra vida.

Desde siempre muchas gracias a todos.

Rodrigo Gonzalez



Agradecimientos

Primero mencionar a los verdaderos hacedores de este título, mis padres, los cuales incondicionalmente me apoyaron en todo el trayecto de mi carrera, comprendieron absolutamente todo y nunca dudaron de mí. Mi inspiración desde siempre.

Segundo a mis hermanos, quienes siempre estuvieron a mi lado, desde más cerca o desde más lejos, estando en los momentos difíciles, acompañando.

A mi pareja, mi compañía. Sin ella este sueño nunca se hubiera hecho realidad. No dejó que dudara de mí.

En lugar muy especial siempre van a estar mis amigos y compañeros, los que me siguieron en este camino y los de siempre, incondicionales.

A los profesores, que batallan la tan difícil lucha de la educación. Los que dan todo para que todos creamos que hay un mañana mejor, que se puede y que al final de este tan difícil camino la elección de emprender este viaje dará sus frutos.

A la Universidad Tecnológica Nacional Facultad Regional Mendoza, la institución pública que me formó, con el aporte de todos los habitantes de mi país.

A mi país, Argentina, que me permitió formarme como Ingeniero Electrónico.

Lucas Escribano



HISTORIAL DE VERSIONES

Version #	Implementado por	Fecha de Revisión	Aprobado Por	Fecha de Aprobación	Razón
0.1	Escribano, Lucas	01/12/2014	Gonzalez, Rodrigo	01/12/2014	Para comentarios
0.2	Escribano, Lucas	03/05/2015	Gonzalez, Rodrigo	04/05/2015	Para comentarios
0.3	Gonzalez, Rodrigo	15/02/2017	Escribano, Lucas	20/02/2017	Para comentarios
0.4	Gonzalez, Rodrigo	8/07/2017	Escribano, Lucas	30/11/2017	Para comentarios
0.5	Escribano, Lucas	01/12/2018	Gonzalez, Rodrigo	01/12/2018	Para comentarios
1.0	Gonzalez, Rodrigo	01/04/2019	Escribano, Lucas	23/04/2019	Para comentarios
1.1	Escribano, Lucas	08/05/2019	Gonzalez, Rodrigo	15/05/2019	Evaluación Económica
1.2	Escribano, Lucas	15/06/19	Gonzalez, Rodrigo	20/06/19	Completar Informe
1.3	Escribano, Lucas	22/07/19	Gonzalez, Rodrigo	26/07/19	Finalizar Informe



INFORMACIÓN DEL PROYECTO

Proyecto	GATEWAY DE COMUNICACIÓN PARA REDES DE SENSORES INALAMBRICOS CON EL USO DE LA PLACA C.I.A.A.
Empresa / Organización	GRIDTICS (1)
Fecha de preparación	22/05/2014
Cliente / Laboratorio	GRIDTICS
Patrocinador (Sponsor)	GRIDTICS
Gerente / Líder de Proyecto	Ing. Carlos Taffernaberry

INFORMACIÓN INTERNA DEL PROYECTO

Nombre del integrante 1	Escribano, Lucas Alberto
Legajo	32167
e-mail	lucasescrivano@hotmail.com
Nombre del integrante 2	Gonzalez, Rodrigo Jeremías
Legajo	27134
e-mail	rodjergon@gmail.com
Tutor	Ing. Taffernaberry, Carlos
Jurado	Ing. Taffernaberry, Carlos
Jurado	Ing. Diedrichs, Ana
Responsable de la cátedra	Ing. Merino, Nelson



ÍNDICE

1	INTRODUCCIÓN	8
1.1	EXECUTIVE RESUME	8
1.2	RESUMEN EJECUTIVO.....	8
2	DESCRIPCIÓN DEL PROYECTO	9
2.1	OBJETIVO GENERAL.....	9
2.2	OBJETIVO PARTICULAR.....	9
3	JUSTIFICACIÓN DEL PROYECTO (CASO DE NEGOCIO)	10
3.1	ANTENEDENTES DEL PROYECTO.....	10
3.2	ESTADO ACTUAL.....	10
3.3	NECESIDAD DEL NEGOCIO Y DEFINICIÓN DEL PROBLEMA	11
3.4	IDEA DEL PROYECTO.....	11
3.5	BENEFICIOS DEL PROYECTO	12
4	ALCANCE	13
4.1	LÍMITES O FUERA DE ALCANCE	13
4.2	REQUISITOS DE ALTO NIVEL	14
4.3	SOLUCIONES Y ENTREGABLES PRINCIPALES	14
5	PLANIFICACIÓN DEL PROYECTO	15
5.1	PROCESOS	15
5.2	CRONOGRAMA.....	16
5.3	HITOS.....	17
6	DESARROLLO DEL PROYECTO	18
6.1	PROYECTO EDU-CIAA (2)	18
6.1.1	¿QUÉ ES LA CIAA?.....	18
6.1.2	ORIGEN DEL PROYECTO CIAA.....	18
6.1.3	PROPÓSITO.....	20
6.1.4	ALCANCE.....	20
6.1.5	OBJETIVOS.....	20
6.1.6	REQUERIMIENTOS.....	20
6.1.7	DEFINICIÓN DE MÓDULOS A INCLUIR EN LA PLATAFORMA.....	21
6.1.8	HARDWARE DE LA EDU-CIAA.....	22
6.1.8.1	BLOQUES FUNCIONALES.....	22
6.1.8.2	DIAGRAMA EN BLOQUES DE LA PLATAFORMA.....	23
6.1.8.3	MÓDULOS DE LA EDU-CIAA	23
6.2	PROYECTO WSN GRIDTIC'S	24
6.2.1	PROTOCOLO WSN GRIDTIC'S.....	25
6.3	HARDWARE	27
6.3.1	HARDWARE GRIDTICs.....	33
6.3.2	HARDWARE REALIZADO.....	33
6.4	SOFTWARE	39
6.4.1	DIAGRAMA DE FLUJO DE FUNCIONAMIENTO	39
6.4.2	CONFIGURACIONES INICIALES.....	40
6.4.3	PROGRAMACIÓN EN C.....	42
6.4.3.1	SPI (ETHERNET).....	42
6.4.3.2	SPI (SD)	48
6.4.3.3	ESP8266(WIFI)	49
6.4.3.4	INTERRUPCIONES	49
6.4.3.5	RTC (REAL TIME COUNTER)	51
6.4.3.6	NTP(NETWORK TIME PROTOCOL)	51
6.4.3.7	MENÚ DE CONFIGURACIÓN	51
6.4.3.8	DISPLAY LCD	52



6.4.3.9	GPIO (GENERAL PURPOSE INPUT /OUTPUT).....	52
6.4.3.10	EFICIENCIA ENERGÉTICA	55
6.4.3.11	REPOSITORIO DEL SOFTWARE.....	55
6.5	TESTS	57
7	FACTIBILIDAD ECONÓMICA.....	61
7.1	APROXIMACIÓN AL VALOR ACTUAL NETO	61
7.2	TASA INTERNA DE RETORNO	61
7.3	PAYBACK O PLAZO DE RECUPERACIÓN.....	62
8	PRODUCTOS DE OTROS FABRICANTES.....	63
8.1	MONNIT	63
8.2	NATIONAL INSTRUMENTS	63
8.3	ADVANTECH.....	64
9	CONCLUSIÓN FINAL	64
10	APÉNDICE A: REFERENCIAS	68
11	APÉNDICE B: DICCCIONARIO.....	69



1 INTRODUCCIÓN

1.1 EXECUTIVE RESUME

The project consists of a gateway, a connection interface of the sensor network with computer networks for data acquisition, administration and control. Its purpose is the processing and analysis of them.

A Gateway that communicates an existing wireless sensor network with a server connected to the Internet through TCP / IP will be made in this project. The connections between the members of the network will be made through several ways, which are WiFi, GPRS, Ethernet and / or USB. When the unlink transmission, the information will be stored on an SD card, until a new reconnection.

1.2 RESUMEN EJECUTIVO

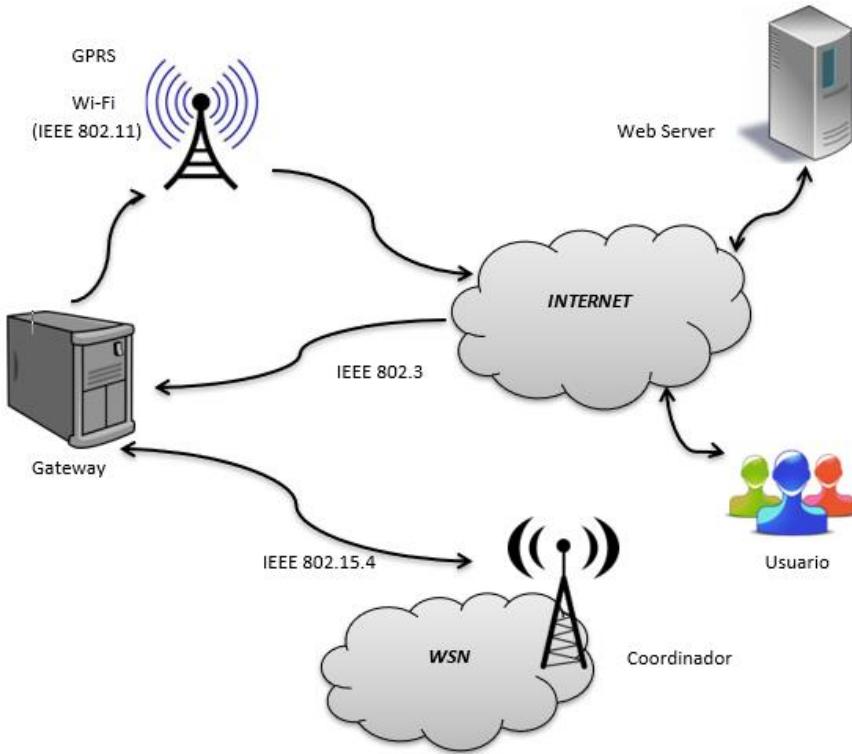
El proyecto consta de una puerta de enlace o Gateway, una interfaz de conexión de la red de sensores con redes de computadoras para la adquisición de datos, administración, control. Su propósito es el procesamiento y análisis de los mismos.

Se realizará, en este proyecto, un Gateway que comunique una red de sensores inalámbrico existente, con un servidor conectado a internet mediante TCP/IP. Las conexiones entre los integrantes de la red se harán mediante diversas maneras, las cuales son WiFi, GPRS, Ethernet y/o USB. Cuando la transmisión de estos enlaces se pierda, la información estará guardada en una tarjeta SD, hasta una nueva reconexión.



2 DESCRIPCIÓN DEL PROYECTO

Consiste en el diseño de un Gateway de comunicaciones a través de una Computadora Industrial Abierta Argentina (CIAA), segura y de bajo costo.



2.1 OBJETIVO GENERAL

Satisfacer las necesidades de los clientes en cuanto a seguridad en el traspaso de datos, facilidad de manejo, ahorro de energía y robusticidad, permitiéndoles mantener comunicación bilateral entre una red propietaria WSN e Internet.

2.2 OBJETIVO PARTICULAR

Como objetivo particular, buscamos captar paquetes de información (temperatura, humedad, etc.) de redes de sensores inalámbricos a través de un protocolo propietario y enviarlos a un servidor web por vía ethernet (IEEE 802.3), Wifi (IEE 802.11). Permitiendo también dicha transmisión de comunicación en ambas direcciones, es decir full duplex, sin pérdida de datos.



3 JUSTIFICACIÓN DEL PROYECTO (CASO DE NEGOCIO)

3.1 ANTENCEDENTES DEL PROYECTO

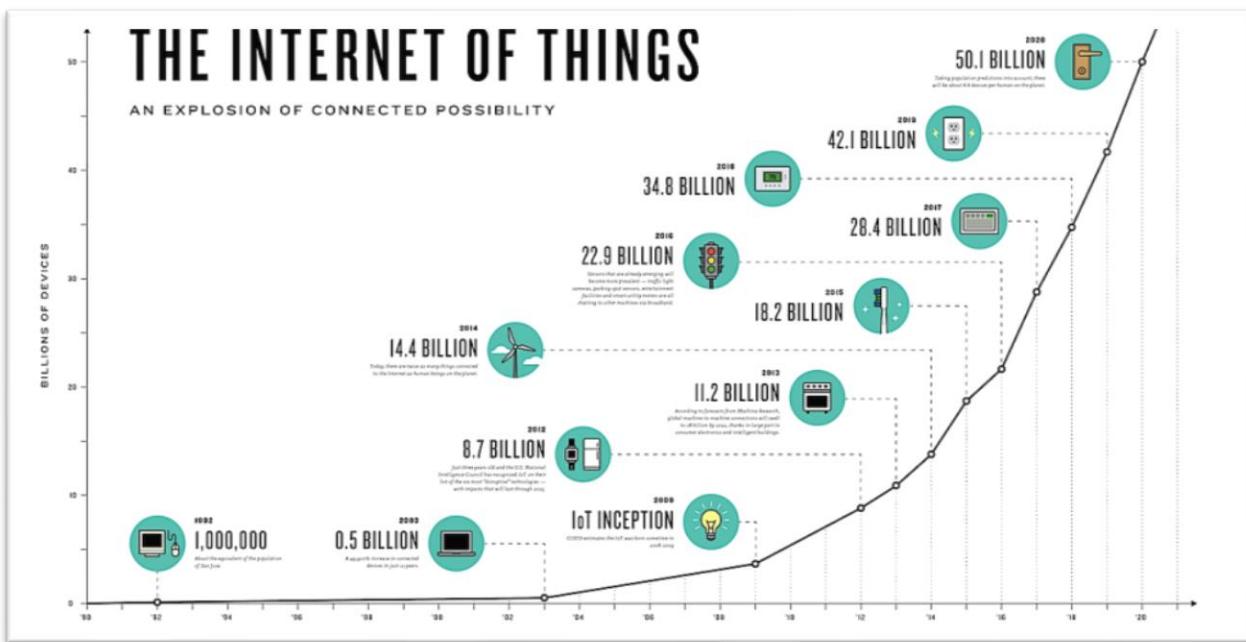
El origen del proyecto se basa en la necesidad de tener dispositivos industriales que se puedan comunicar con redes de sensores inalámbricos, de diseño argentino. En el mercado, estos, son escasos y se encuentran a un precio elevado, complicando la adquisición de repuestos, debido a que no están aptos para su uso industrial.

Nuestro proyecto intenta satisfacer esas necesidades, ofreciendo un producto a bajo costo, con tecnología nacional y de proyección mundial.

3.2 ESTADO ACTUAL



Actualmente los dispositivos de puerta de enlaces, tiene una demanda creciente debido a la integración del concepto de Internet de las Cosas (IoT, Internet of Things) en el campo de la industria (Agrícola, Petroquímica, Metalúrgica, etc.)



3.3 NECESIDAD DEL NEGOCIO Y DEFINICIÓN DEL PROBLEMA

Hoy en día, en el mercado estos equipos tienen un costo muy excesivo del cual estamos seguros que puede reducirse considerablemente. Además de poder adaptarlos a cualquier protocolo de comunicación diseñado.

El Gateway apunta a resolver la problemática de una red de sensores inalámbricas que se comunican a través de un protocolo de comunicaciones propietarios que desarrollaron en el grupo de investigación Gridtcs. De la cual los objetivos del mismo son:

- ✓ Envío de la información a un servidor web para su análisis.
- ✓ Vincular protocolos propietarios con protocolos estandarizados.

3.4 IDEA DEL PROYECTO

La idea del proyecto es enlazar los datos de una red de sensores inalámbricos, utilizados en vides, con un servidor a través de comunicación cableada (Ethernet), y/o comunicación inalámbrica (Wifi / GPRS).

Teniendo en cuenta la desconexión de las transmisiones, se considera el almacenamiento de la información a través de una tarjeta SD, hasta su próxima reconexión, donde se re establecerá el envío de los datos almacenados.

Su configuración resulta sencilla, donde se insertará los parámetros correspondientes en un archivo tipo txt, almacenados en la tarjeta SD.



3.5 BENEFICIOS DEL PROYECTO

- ✓ Capacidad de interconexión entre distintas redes.
- ✓ Capacidad de análisis de datos en la nube, en cualquier parte del mundo y en cualquier momento.
- ✓ Costos muy por debajo del precio internacional de mercado.
- ✓ Dimensiones menores a los productos similares que se encuentran disponibles.
- ✓ Seguimiento detallado de las funciones que realiza el Gateway mediante un log interno, disponible en cualquier momento, permitiendo que el usuario tenga una trazabilidad en el tiempo de lo que está sucediendo, dado la seguridad que el usuario final necesita.
- ✓ Muy fácil configuración.



4 ALCANCE

Los alcances del proyecto son:

- ✓ Portabilidad Energética y Tecnológica.
- ✓ Facilidad de utilización y configuración.
- ✓ Robusticidad.
- ✓ Producto de Utilización Industrial.
- ✓ Producto Económico.
- ✓ Ahorro de Energía.
- ✓ Incorporación de la EDU-CIAA, aportando al desarrollo de la misma.
- ✓ Realización de software de fácil entendimiento, con comentarios entendibles, e integrado a repositorios de fácil acceso, de modo que se pueda seguir con revisiones y versiones nuevas del mismo.
- ✓ Soporte tanto en la utilización del producto como en la programación del software.
- ✓ Diseño modular, de manera que el reemplazo o la actualización de algún componente sea sencillo.

4.1 LÍMITES O FUERA DE ALCANCE

Nuestro dispositivo se limitará solo vinculará el protocolo propietario diseñado por el grupo de investigación Gridtics. Esto es, sólo recibe tramas de dicho protocolo, dado que la cantidad de datos que se transmiten están dentro de la trama. Al cambiar de protocolo la trama posiblemente cambie, lo cual deja inutilizable al equipo. De cualquier forma el manejo del software permite revertir el producto fácilmente para que se adapte a las nuevas necesidades.

Las configuraciones que se realizarán del equipo son básicas. Se podrán configurar sólo 5 parámetros, los cuales serán el SSID de la red WiFi, la contraseña de la misma, la dirección IP del server y el puerto a utilizar, y la dirección del server NTP al cual se conectará el dispositivo.

El dispositivo solo se limitará la comunicación WiFi, Ethernet y GPRS a través de módulos, shields o ponchos compatibles con la placa CIAA.

Solo se admite tarjetas SD (NO micro SD) con formato de archivos FAT.



La comunicación con el servidor donde se relevan los datos sólo se puede realizar con el protocolo IPv4, no siendo compatible con el protocolo IPv6.

No se podrá modificar ningún parámetro del coordinador de la red propietaria WSN. Del mismo sólo se obtienen los datos por UART, y se envían los paquetes recibidos desde el servidor.

4.2 REQUISITOS DE ALTO NIVEL

La siguiente tabla presenta los requisitos que el resultado del proyecto (producto o servicio) debe cumplir como condición necesaria.

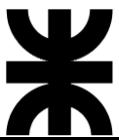
Req. #	Descripción del requisito
1	Permitir la comunicación entre protocolos (propietarios y estandarizados).
2	Transmisión Full-duplex.
3	En caso de interrupción de comunicación, asegurar que los datos no se pierdan.
4	Tener respuestas rápidas que permitan recibir datos con intervalos de hasta 3 segundos.
5	Utilizar los protocolos estándar de comunicación UART y TCP/IP.
6	Permitir la configuración de algunos parámetros.
7	Integración del coordinador de la red WSN y los shields a una misma placa compatible con la EDU-CIAA.

4.3 SOLUCIONES Y ENTREGABLES PRINCIPALES

El gateway se entregará en su respectivo empaque, el cual contendrá en su interior los entregables principales, listados en el artículo previo (Gateway, shields de comunicación, circuitos de conexiónados, cable de alimentación, manual de operación al usuario).

La siguiente tabla muestra un listado de los entregables del proyecto (productos o servicios)

Entregables principales	I Descripción del entregable
Gateway	
Manual de Operación	
Shields Comunicación	Ethernet, Wifi, GPRS
Circuito de Conexiónado	
Cable de alimentación	Cable micro USB-B 2.0



5 PLANIFICACIÓN DEL PROYECTO

5.1 PROCESOS

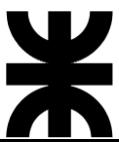
Elegimos la metodología de ISO 21500, esta norma nos proporciona orientación para dirección y gestión de proyectos y puede usarse para cualquier tipo de organización y tipo, con independencia de su complejidad, tamaño y duración.

Esta norma internacional proporciona una descripción de alto nivel de conceptos y procesos que se consideran que forma parte de las buenas prácticas en dirección y gestión de proyectos.



5.2 CRONOGRAMA

ID	Task Name	Duration	Start	Finish	H2	2015	H1	H2	2016	H1	H2	2017	H1	H2	2018	H1	H2	2019	H1	H2	2020	H1	H2
1	Espesar la llegada de EDU-CIAA	3 mons	Fri 01/08/14	Thu 23/10/14																			
2	Recopilar Información	3 mons	Fri 24/10/14	Thu 15/01/15																			
3	Pruebas con LPC Expresso LPC4337	6 mons	Fri 24/10/14	Thu 09/04/15																			
4	Buscar Entorno de Trabajo	7 days	Fri 10/04/15	Mon 20/04/15																			
5	Busqueda información de hardware adicional	3 mons	Fri 10/04/15	Thu 02/07/15																			
6	Adquisición de Materiales	6 mons	Fri 03/07/15	Thu 17/12/15																			
7	Diseño de prototipo (hardware)	3 mons	Fri 18/12/15	Thu 10/03/16																			
8	Programación Firmware	36 mons	Fri 11/03/16	Thu 13/12/16																			
9	Ensayos	2 mons	Fri 14/12/16	Thu 07/02/19																			
10	Análisis de Datos de Ensayos	2 mons	Fri 08/02/19	Thu 04/04/19																			
11	Redacción Manual	2 mons	Fri 05/04/19	Thu 30/05/19																			
12	Redacción y entrega de proyecto	3 mons	Fri 05/04/19	Thu 27/06/19																			



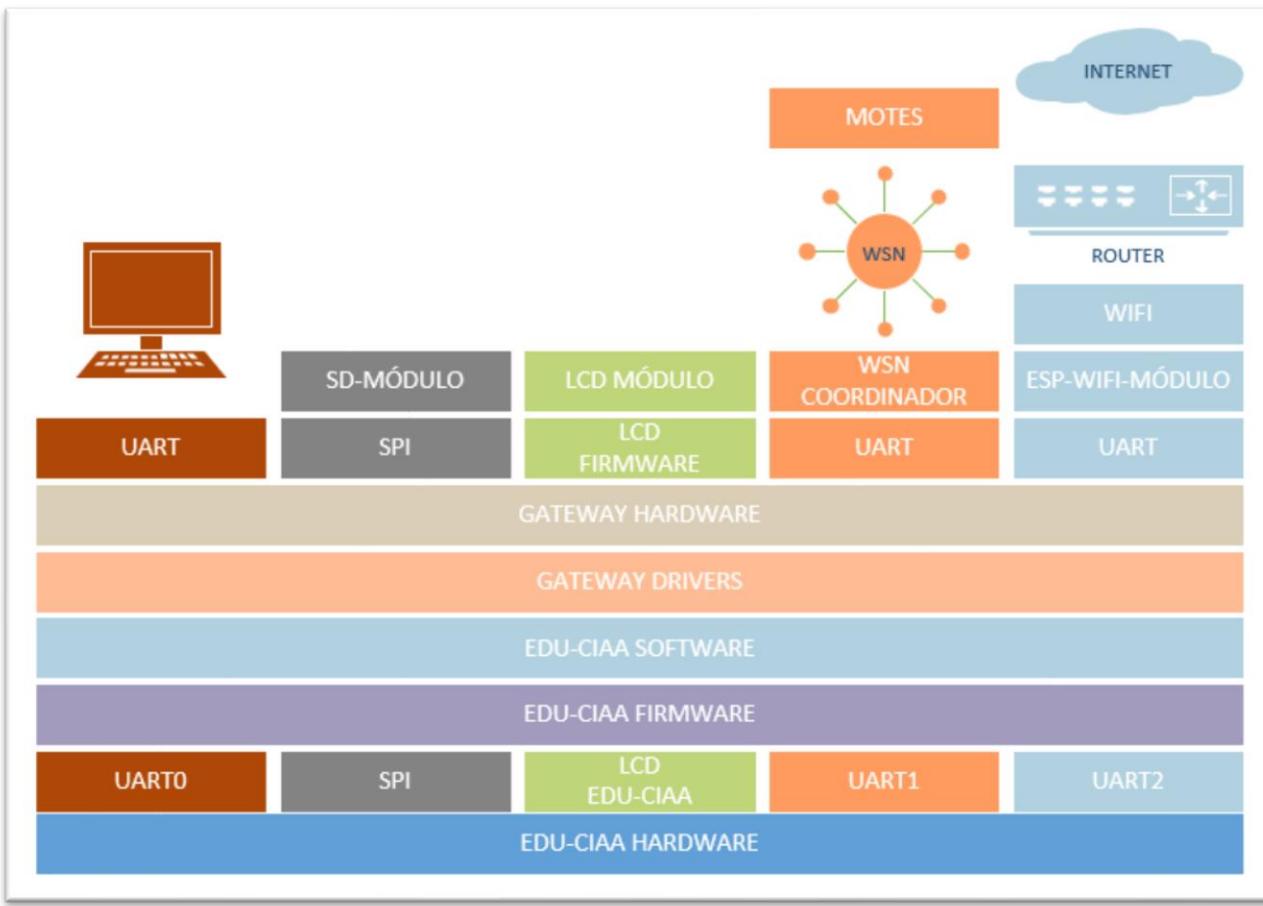
5.3 HITOS

La tabla muestra un listado de hitos generales del proyecto y el cronograma estimado de finalización.

Hitos	Fecha de finalización
<i>Inicio de Proyecto</i>	01/08/14
<i>Esperar la llegada de EDU-CIAA</i>	23/10/14
<i>Adquisición de Materiales</i>	17/12/15
<i>Diseño de prototipo (hardware)</i>	10/03/16
<i>Programación Firmware</i>	13/12/18
<i>Ensayos</i>	07/02/19
<i>Redacción Manual</i>	30/05/19
<i>Entrega de Proyecto Final</i>	



6 DESARROLLO DEL PROYECTO



6.1 PROYECTO EDU-CIAA (2)

6.1.1 ¿QUÉ ES LA CIAA?

La **CIAA** (Computadora Industrial Abierta Argentina) es una plataforma creada para impulsar el desarrollo tecnológico, generar conocimiento, y darle la posibilidad a **PyMEs** de poder basar sus trabajos en una plataforma testeada y con la mayor cantidad de funcionalidades posible. Reduciendo así el tiempo y la inversión necesaria para poder llegar a un producto final.

6.1.2 ORIGEN DEL PROYECTO CIAA

La Computadora Industrial Abierta Argentina (CIAA) se comenzó a gestar en julio de 2013, cuando la Secretaría de Planeamiento Estratégico Industrial del Ministerio de Industria de la Nación (SPEI) y la Secretaría de Políticas Universitarias del Ministerio de Educación de la Nación (SPU) convocaron a la Asociación Civil para la Investigación, Promoción y Desarrollo de los Sistemas Electrónicos Embebidos (ACSE) y a la Cámara de Industrias Electrónicas, Electromecánicas y Luminotécnicas (CADIEEL) a participar en el “Plan Estratégico Industrial 2020”.

El pedido inicial fue que desde el sector académico (ACSE) y desde el sector industrial (CADIEEL) se presenten propuestas para agregar valor en distintas ramas de la



economía (maquinaria agrícola, bienes de capital, forestal, textil, alimentos, etc.) a través de la incorporación de sistemas electrónicos en procesos productivos y en productos de fabricación nacional.

Entre agosto y noviembre de 2013 se desarrollaron varias reuniones de trabajo, en las que participaron además el Ministerio de Ciencia, Tecnología e Innovación Productiva y representantes de otros organismos públicos y de diversas empresas privadas.

Como resultado de esas reuniones se concluyó que muchas empresas argentinas de diversos sectores productivos no incorporaban electrónica en sus procesos productivos o en sus productos, otras utilizaban sistemas electrónicos obsoletos, muchas utilizaban sistemas importados y sólo unas pocas utilizaban diseños propios basados en tecnologías vigentes y competitivas.

También se observó que muchas empresas eran reticentes a invertir en desarrollos electrónicos, pero que al mismo tiempo sí realizaban inversiones de igual o mayor magnitud en ampliaciones edilicias. Es decir, se observó que el problema no estaba ligado a limitaciones económicas, sino más bien al riesgo tecnológico que las empresas veían en la incorporación de sistemas electrónicos en sus productos o procesos productivos debido a su desconocimiento de la temática.

Además, se encontró que muchas empresas tenían temor de pasar a depender de servicios o insumos electrónicos comercializados por empresas que utilizan políticas muy restrictivas de intercompatibilidad de sus sensores y aplicaciones, lo que implica un riesgo logístico y económico.

A partir de esta situación la ACSE y CADIEEL propusieron desarrollar un sistema electrónico abierto de uso general, donde absolutamente toda su documentación y el material para su fabricación estuviera libremente disponible en internet (diagramas esquemáticos, diseño del circuito impreso, códigos fuentes de los programas, etc.), que estuviera diseñado en base a criterios adecuados para su utilización en aplicaciones industriales, que no dependiera de una línea específica de procesadores, y que pudiera ser fabricado por la mayoría de las empresas PyMEs nacionales, lo que implicaba por ejemplo utilizar en el diseño circuitos impresos de no más de cuatro capas. De este modo a partir del 15 de noviembre de 2013 se comenzó a trabajar en la Computadora Industrial Abierta Argentina (CIAA).

Desde ese momento la iniciativa de la CIAA se desarrolla con éxito y en la actualidad decenas de universidades nacionales, empresas privadas e instituciones están involucradas.

Hoy en día la CIAA está disponible en la versión [CIAA-NXP](#) y otras seis versiones están en elaboración: CIAA-ATMEL, [CIAA-FSL](#), CIAA-PIC, CIAA-RX, CIAA-ST, CIAA-TI. Además, se está trabajando en el firmware y en el software, para que la CIAA se pueda programar en lenguaje C utilizando una API especialmente diseñada para ser compatible con los estándares POSIX y que sea portable a diversos sistemas operativos de tiempo real. También se está trabajando para que la CIAA se pueda programar en cualquiera de los lenguajes descriptos por la norma IEC 61131-3, como por ejemplo el Ladder.

A su vez la CIAA está soportada por una comunidad de más de 3.000 desarrolladores de sistemas embebidos (embebidos32@googlegroups.com) y ha concitado el interés de PyMEs de todo el país, e incluso de empresas del exterior, así como diversos medios de prensa, como ser la Televisión Pública, Telam, los diarios La Nación, Ámbito Financiero, el Cronista Comercial y Tiempo Argentino, ya que se trata de la primera iniciativa a nivel mundial que cumple con las características de ser industrial, abierta y estar basada en procesadores de distintas marcas.



6.1.3 PROPÓSITO

Los propósitos de la EDU-CIAA son:

- Proveer una plataforma de desarrollo moderna y económica basada en la CIAA que sirva a docentes y a estudiantes en los cursos de sistemas embebidos.
- Lograr una amplia inserción en el sistema educativo argentino.
- Realizar un aporte eficaz al desarrollo de vocaciones tempranas en electrónica, computación e informática.
- Demostrar que las universidades argentinas son capaces de realizar un desarrollo colaborativo exitoso en el área de los sistemas embebidos, cumpliendo con requerimientos de tiempo y forma.

6.1.4 ALCANCE

El alcance del desarrollo de esta primera etapa de la EDU-CIAA es el siguiente:

- Diseño, fabricación y validación de una primera versión del hardware de la plataforma, basado en el diseño de la CIAA-NXP, pero con menores prestaciones que la CIAA, con el fin de reducir su costo y complejidad.
- Diseño, implementación y validación de una primera versión de un entorno de desarrollo integrado (IDE) específico, pensado para que sea accesible a estudiantes que abordan por primera vez este tipo de plataformas.

6.1.5 OBJETIVOS

Los siguientes son los objetivos propuestos para la EDU-CIAA:

- Implementar una versión de bajo costo de la CIAA pensada para la enseñanza Universitaria, Terciaria y Secundaria.
- Realizar el desarrollo en forma colaborativa entre los docentes universitarios miembros de la Red RUSE.
- Lograr que el costo final sea menor a los 50 dólares por unidad en lotes de fabricación de 100 unidades, considerando los componentes y el circuito impreso.
- Desarrollar el circuito impreso en dos capas y en un tamaño igual o menor al de la CIAA.
- Contar con 20 prototipos operativos para la fecha de realización del SASE2014 (13/08/2014).
- Realizar 10 talleres de formación sobre la EDU-CIAA durante el segundo semestre de 2014.
- Alcanzar a 100 docentes de la Red RUSE y a 50 docentes terciarios y secundarios con actividades de capacitación durante el segundo semestre de 2014.
- Lograr que la EDU-CIAA se utilice para dar clases universitarias en 30 unidades académicas, 20 escuelas secundarias y 5 escuelas terciarias durante el primer semestre de 2015.

6.1.6 REQUERIMIENTOS

Los siguientes son los requerimientos adoptados para la EDU-CIAA:

- El sistema debe soportar el mismo firmware que la CIAA.
- El entorno de desarrollo integrado (IDE) debe ser una versión de más fácil manejo que el IDE de la CIAA.
- El sistema debe tener un pulsador de reset.



- El sistema debe tener el mismo microcontrolador que alguna de las versiones de la CIAA.
- El sistema debe tener un tamaño igual o menor al de la CIAA y estar diseñado en un PCB de dos capas.
- El sistema debe poder alimentarse mediante un puerto USB estándar.
- El sistema debe poseer un conector USB para programación y depuración desde una PC.
- El sistema debe tener conectividad mediante interfaz RS-485.
- El sistema debe poseer conectores de expansión que incluyan los siguientes elementos del microcontrolador:
 - i) interfaz RMII Ethernet
 - ii) Interfaz USB
 - iii) Interfaz CAN
 - iv) interfaz SPI
 - v) interfaz I2C
 - vi) salida D/A
 - vii) entradas A/D
- El sistema debe tener cuatro Leds (RGB) que sean activados por el microcontrolador.
- El sistema debe tener cuatro pulsadores conectados al microcontrolador.
- El sistema debe tener un display de siete segmentos que sea controlado por el microcontrolador.

6.1.7 DEFINICIÓN DE MÓDULOS A INCLUIR EN LA PLATAFORMA

Aquí se encuentra una tabla detallando las razones por las cuáles se agregaron o quitaron a la EDU-CIAA módulos originalmente disponibles en la CIAA:



Módulo	¿Se incluye?	¿Por qué?	Observaciones
FUENTE	Sí	Es mandatorio	Se alimentará a través de USB
CPU (LPC4337)	Sí	Es mandatorio	
JTAG	Sí	Para debugging	Componentes montaje opcional
SDRAM (8MB)	No	Uso poco frecuente	
FLASH (4MB)	No	Uso poco frecuente	
EEPROM	No	Uso poco frecuente	
Entradas Analóg.	Sí	Uso habitual	Sin front-end - pines en CON1
Salidas Analóg.	Sí	Uso habitual	Sin front-end - pines en CON1
Entradas Optoac.	No	Elevado costo	Pines disponibles en CON1
Salidas Op.Drain	No	Elevado costo	Pines disponibles en CON1
Salidas Relay	No	Elevado costo	Pines disponibles en CON1
RS485	Sí	Interconexión	Muy simple para armar redes
USB	Sí	Habitual en PCs	
Ethernet	"Ní"	Elevado costo	Pines disponibles en CON1
RS232	"Ní"	Obsoleto en PCs	Pines disponibles en CON1
CAN	"Ní"	Uso poco frecuente	Pines disponibles en CON1
microSD	No	Ausente en LPC4337	
Expansión SPI	Sí	Conectividad al uC	Pines disponibles en CON1
Expansión I2C	Sí	Conectividad al uC	Pines disponibles en CON1
Expansión GPIO	Sí	Conectividad al uC	Pines disponibles en CON1
LEDs RGB x4	Sí	P/práctica básica	
Pulsadores x4	Sí	P/práctica básica	
Int. Vaux 3.3V	Sí	Uso habitual	Tensión regulada del sistema
Int. Vaux 5V	Sí	Uso habitual	Tensión de entrada al sistema
Int. Vaux 24V	No	Ausencia de 24V	

6.1.8 HARDWARE DE LA EDU-CIAA



6.1.8.1 BLOQUES FUNCIONALES

La EDU-CIAA está basada en la CIAA-NXP, por ser la primera versión de la CIAA que se encuentra disponible. Por lo tanto su microcontrolador es también el LPC4337 (dual core ARM Cortex-M4F y Cortex-M0).

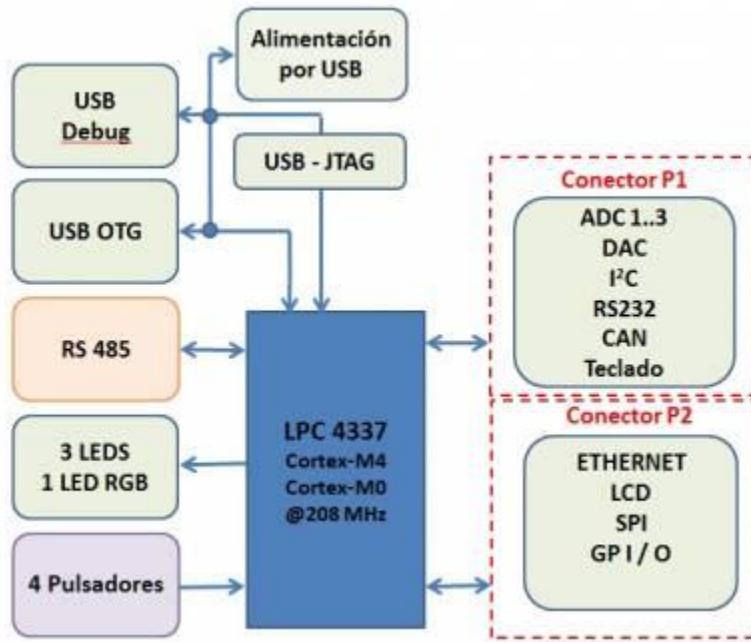
Sin embargo, con el objetivo de abaratar costos y reducir su complejidad la EDU-CIAA incorpora sólo algunas de las funcionalidades de la CIAA.



A su vez, con el fin de permitir el desarrollo de algunas prácticas sencillas sin que sea necesario recurrir a hardware adicional, incluye además algunos recursos que no están presentes en la CIAA.

6.1.8.2 DIAGRAMA EN BLOQUES DE LA PLATAFORMA

En la siguiente figura se observa un diagrama en bloques de la EDU-CIAA basada en LPC4337:



6.1.8.3 MÓDULOS DE LA EDU-CIAA

- 2 puertos micro-USB (uno para aplicaciones y debugging, otro para alimentación).
- 4 salidas digitales implementadas con leds RGB.
- 4 entradas digitales con pulsadores.
- 1 puerto de comunicaciones RS 485 con bornera.
- 2 conectores de expansión:

◦ P1:

- 3 entradas analógicas (ADC0_1,2y3),
- 1 salida analógica (DAC0),
- 1 puerto I2C,
- 1 puerto asincrónico full duplex (para RS-232).
- 1 puerto CAN,
- 1 conexión para un teclado de 3x4,

◦ P2:

- 1 puerto Ethernet,
- 1 puerto SPI,
- 1 puerto para Display LCD con 4 bits de datos, Enable y RS.
- 9 pines genéricos de I/O.

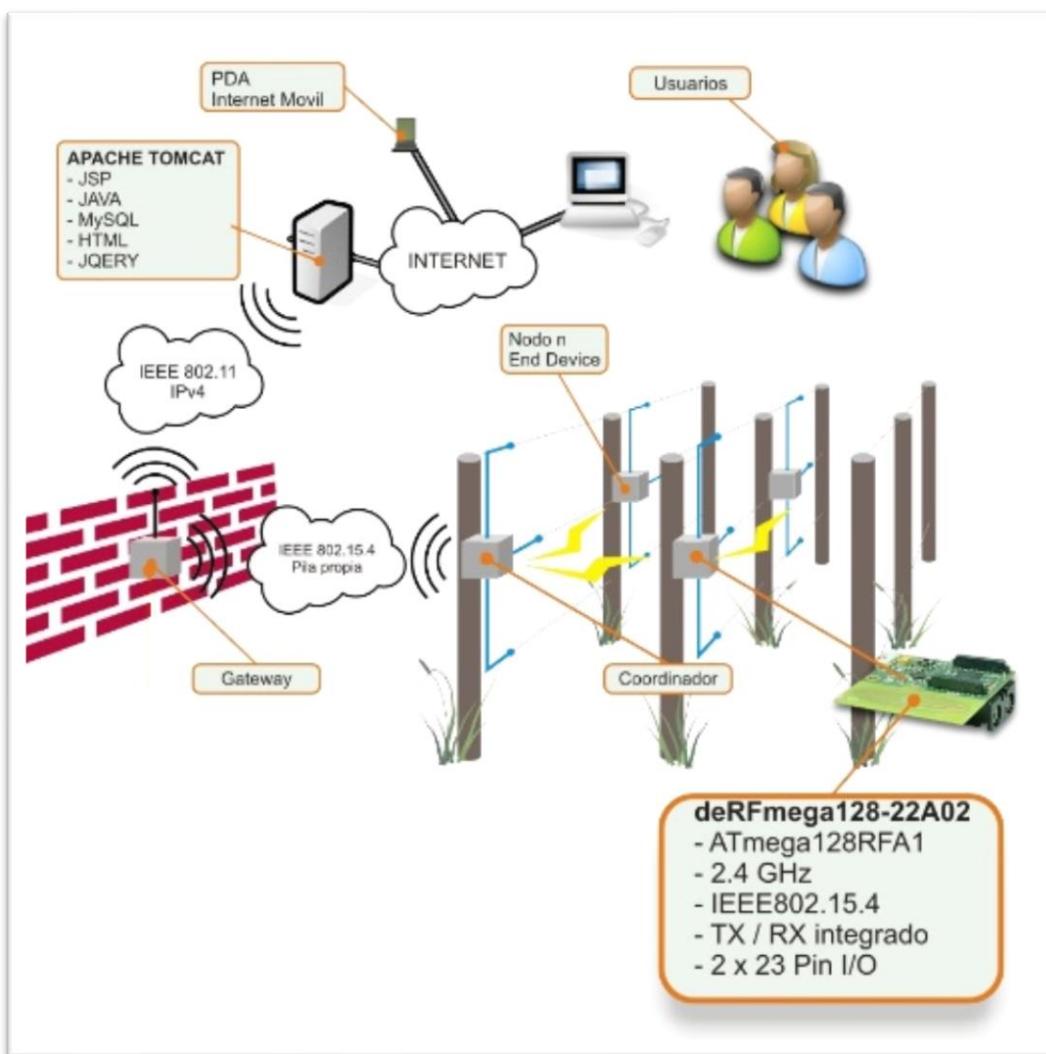


6.2 PROYECTO WSN GRIDTIC'S

El trabajo de GridTic's presenta el desarrollo de una red inalámbrica de sensores (WSN), basada en IEEE-802.15.4, para ser usada en la caracterización de las heladas en agricultura de precisión mediante la medición de temperatura.

El principal objetivo es reducir el consumo energético de la red al mínimo, permitiendo varios puntos de medición por nodo y el monitoreo remoto del comportamiento de los sensores.

Para la interfaz de comunicación entre el nodo de la WSN y los sensores, desarrollaron un protocolo de comunicación serie inspirado en SDI-12. Los resultados preliminares muestran una WSN de bajo costo y bajo consumo. El usuario puede acceder a los datos y utilizarlos para la investigación agronómica.





6.2.1 PROTOCOLO WSN GRIDTIC'S

Formato de la Trama

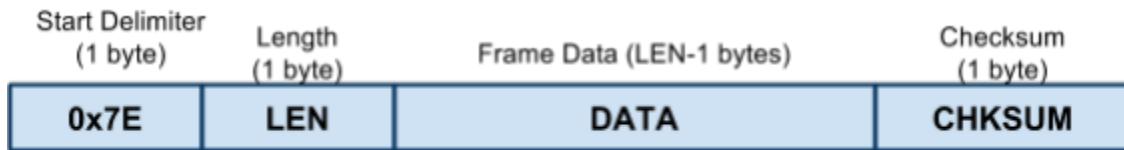


figura 1

- ✓ **Start Delimiter:** Indica el comienzo de una trama. Siempre toma el valor 0x7E.
 - ✓ **Length:** El largo de la trama en bytes. Tiene en cuenta todos los bytes de DATA más el byte de checksum.
 - ✓ **Frame Data:** Los datos del frame.
 - ✓ **Checksum:** Para calcular el checksum, se debe hacer una XOR entre todos los caracteres, incluido el delimitador de trama (0x7E).
- Para verificar, el receptor debe hacer el mismo cálculo sin incluir el byte de checksum y comparar el resultado con éste.

Caracteres Especiales

- 0x7E: Delimitador de trama
- 0x7D: Escape carácter

¿Cómo escapar los caracteres especiales?

Cada vez que uno de estos caracteres aparezca en la trama, se antepone 0x7D y se hace una XOR entre el valor original y 0x20. Por ejemplo, si 0x7E aparece en la trama, se hace la XOR con 0x20 y queda 0x5E, por lo tanto, el 0x7E de la trama original se reemplaza por 0x7D 0x5E.

Para recuperar el dato original, cuando aparece un 0x7D en la trama, se elimina y al carácter siguiente se le vuelve a hacer una XOR con 0x20. Por ejemplo, siguiendo el ejemplo anterior, si nos llega 0x7D0x5E, eliminamos el 0x7D y hacemos el XOR entre 0x5E y 0x20, obteniendo 0x7E original.

¿Qué se debe escapar? Todo lo que venga luego del indicador de comienzo de trama (del 0x7E)

Primitivas

Las primitivas especificadas en este protocolo se clasifican en:

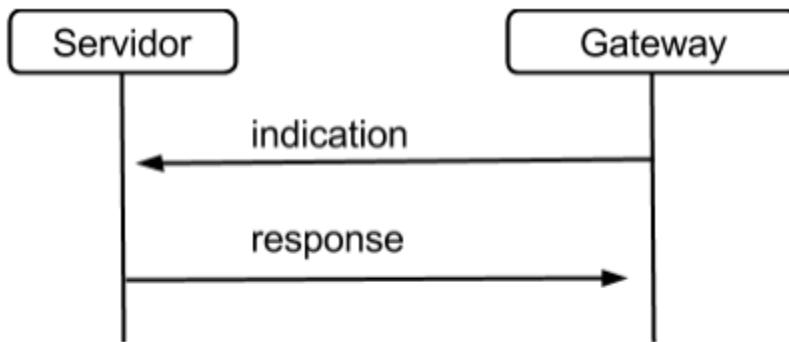
Comandos: empiezan con el prefijo COMMAND. Utilizadas para enviar y recibir comandos desde el server hacia la WSN y viceversa.

Datos: empiezan con el prefijo DATA. Utilizadas para recibir y pedir datos de los sensores.

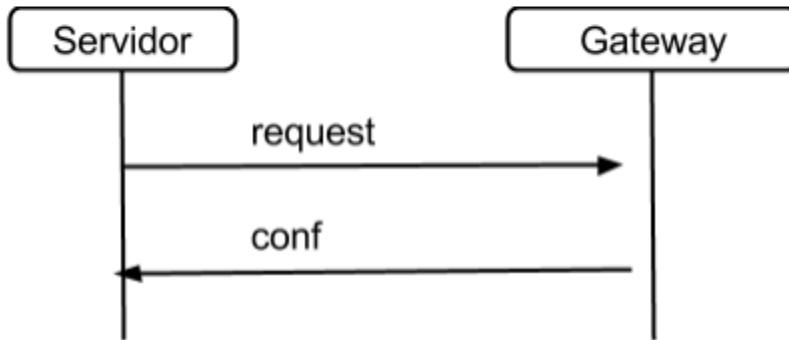


Desde el punto de vista del servidor se utilizará el sufijo REQ, CONF, IND y RESP según quién inicie el evento y quién es notificado. A modo de ejemplo mostraremos el envío y pedido de datos hacia la red.

1) Envío de datos, eventos al servidor



2) Petición del servidor hacia la red WSN



Listado de Primitivas del Protocolo Gateway

- **DATA.REQ (0x01):** Data request. Se usa para pedir un dato a un mote
- **DATA.CONF (0x02):** Data confirmation. Respuesta a DATA.REQ con el estado y los datos pedidos.
- **DATA.IND (0x03):** Data indication. Los datos que envía el mote de forma periódica según como sea configurado.
- **DATA.RESP 0X00**
- **COMMAND.REQ (0x04):** Command request. Se usa para enviar un comando.
 - INTERVAL_CMD (0x03)
 - RTC_CMD (0x04)
- **COMMAND.CONF (0x05):** Command confirmation. Respuesta a COMMAND.REQ con el estado de la ejecución del comando.
 - INTERVAL_CMD (0x03)
 - RTC_CMD (0x04)
- **COMMAND.IND (0x06):** Command indication. Avisa de cambios en la red. (Cambio de dirección corta de un nodo, se murió un nodo, etc).
 - MOTE_START(0X01)
 - ADDRESS_CHANGE (0x02)



- STATUS (0x05)
- **COMMAND.RESP (0307)**
- **KEEP ALIVE:** LONG = 0 y CMD=0. El paquete keep alive sólo contiene byte comienzo de trama y luego cero. Este paquete pequeño es enviado por el gateway periódicamente (cada 3 minutos) al servidor. El servidor le responde con el mismo paquete. La utilidad y necesidad de este comando se debe a que el Gateway desconoce cuando la conexión ha terminado por causas ajenas (ej, pérdida de wifi) De esta manera gracias a este comando puede notar cuando se ha perdido conexión para empezar a guardar los paquetes de datos de la red de sensores en su memoria flash. Con esta solución poco prolja se evita la pérdida de datos debido a caídas de conexión ajenas al servidor y al Gateway.

Como no influye nuestro proyecto en la lectura de todas las tramas, no se muestran a continuación todas las posibilidades de paquetes que el coordinador puede recibir. Esta información se encuentra disponible en los papers citados en las referencias del APÉNDICE A. (3)

6.3 HARDWARE

Lo primero que se realizó fue comparar los objetivos planteados, con el material propuesto, para comprobar que éstos últimos puedan ser llevado a cabo correctamente. A simple vista se comprueba que se necesita hardware adicional, ya que en los objetivos se considera, Ethernet, WiFi, GPRS y SD, interfaces que no están contempladas en la placa original.

Se plantearon dos opciones:

- Realizar el hardware correspondiente, comprando los componentes necesarios, creando las placas y soldando.
- Comprar hardware existente que cumpla las condiciones necesarias.

Se llegó a la conclusión que, lo más rápido y económico, gracias a los protocolos de comunicación existentes en la placa, era utilizar recursos disponibles en el mercado para cumplir con los objetivos.

Se buscó una solución a nivel hardware en la EDU-CIAA para comunicarnos con los módulos externos.

Se evaluaron 4 tipos de comunicación:

- SPI
- SERIAL (UART)
- I2C
- CAN

Y se concluyó que los tipos más universales son UART y SPI, siendo SPI la mejor opción por permitir tener varios esclavos conectados a un mismo bus.

Una vez leídos los datasheet de los componentes, se procede entonces a adquirir 3 módulos externos:



-Ethernet (SPI)



Shield Arduino Ethernet-SD:

El Arduino ethernet shield nos da la capacidad de conectar un Arduino a una red ethernet. Es la parte física que implementa la pila de protocolos TCP/IP.

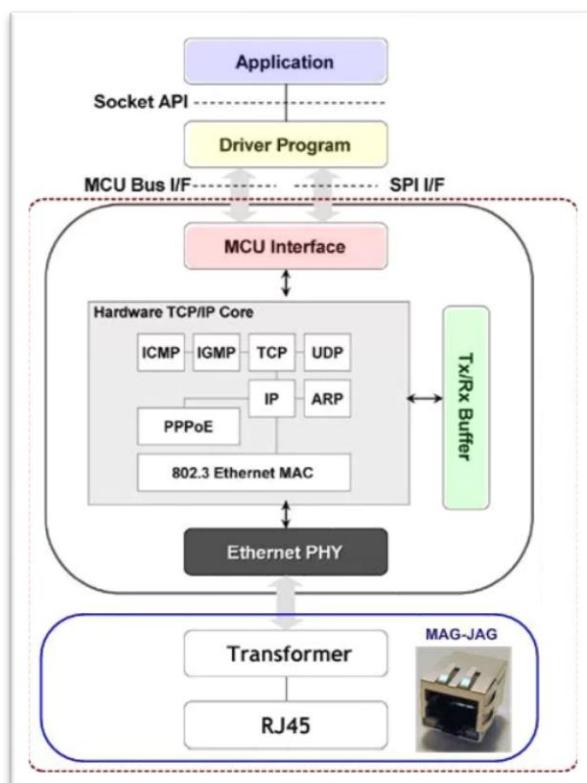
Está basada en el chip ethernet Wiznet W5100. El Wiznet W5100 provee de una pila de red IP capaz de soportar TCP y UDP. Soporta hasta cuatro conexiones de sockets simultáneas.

La shield contiene varios LEDs para información:

- ON: indica que la placa y la shield están alimentadas.

- LINK: indica la presencia de un enlace de red y parpadea cuando la shield envía o recibe datos
- 100M: indica la presencia de una conexión de red de 100 Mb/s (de forma opuesta a una de 10Mb/s)
- RX: parpadea cuando el shield recibe datos
- TX: parpadea cuando el shield envía datos

El jumper soldado marcado como "INT" puede ser conectado para permitir a la placa Arduino recibir notificaciones de eventos por interrupción desde el W5100.





Features

- Support Hardwired TCP/IP Protocols : TCP, UDP, ICMP, IPv4 ARP, IGMP, PPPoE, Ethernet
- 10BaseT/100BaseTX Ethernet PHY embedded
- Support Auto Negotiation (Full-duplex and half duplex)
- Support Auto MDI/MDIX
- Support ADSL connection (with support PPPoE Protocol with PAP/CHAP Authentication mode)
- Supports 4 independent sockets simultaneously
- Not support IP Fragmentation
- Internal 16Kbytes Memory for Tx/Rx Buffers
- 0.18 µm CMOS technology
- 3.3V operation with 5V I/O signal tolerance
- Small 80 Pin LQFP Package
- Lead-Free Package
- Support Serial Peripheral Interface(SPI MODE 0, 3)
- Multi-function LED outputs (TX, RX, Full/Half duplex, Collision, Link, Speed)

-WiFi (SPI-UART)



ESP8266 Features:

- 32 Bit CPU@80Mhz
- 64 Kb command Ram.
- 96 Kb data Ram.
- External QPI Flash usually 512 Kb up to 4Mbit.
- IEEE 802.11 WiFi 2.4Ghz, WEP/WPA/WPA2.
- Up to 16 GPIO Pins, SPI, I²C, I²S, UART.
- 10 Bit ADC.

-SD (SPI-UART)



Rev 1.0:

SD Features:

- Standard SD card, SDHC card and TF card compatible.
- UART Grove & I2C connection compatible.
- Fully supported SD Library.
- Minimal number of SPI port.
- Truly stackable.



También se obtuvo una pantalla LCD de 16x2 caracteres.



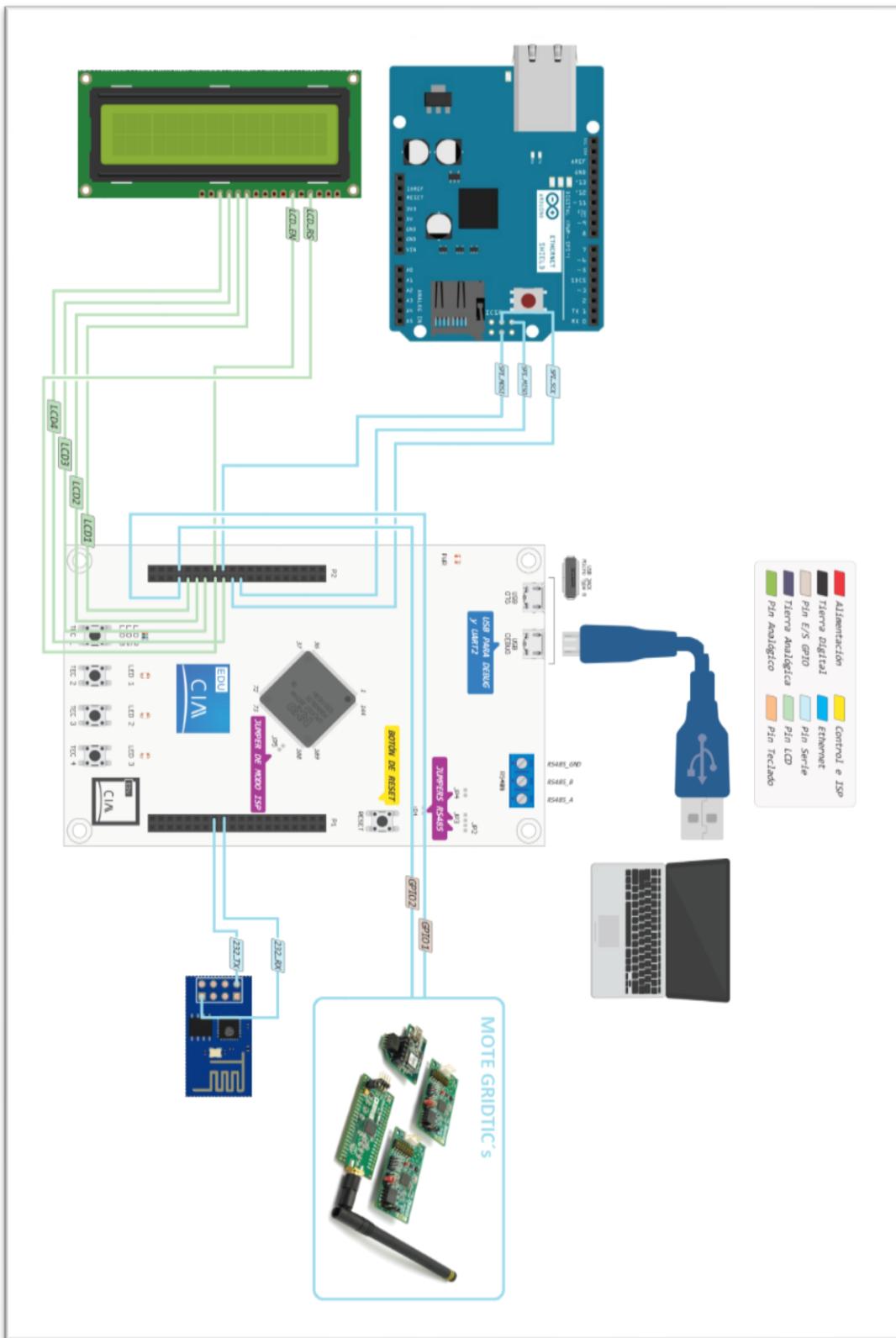
Se definieron las posibilidades de conexión de los artefactos electrónicos en la EDU-CIAA.

Se tuvieron en cuenta los siguientes puntos:

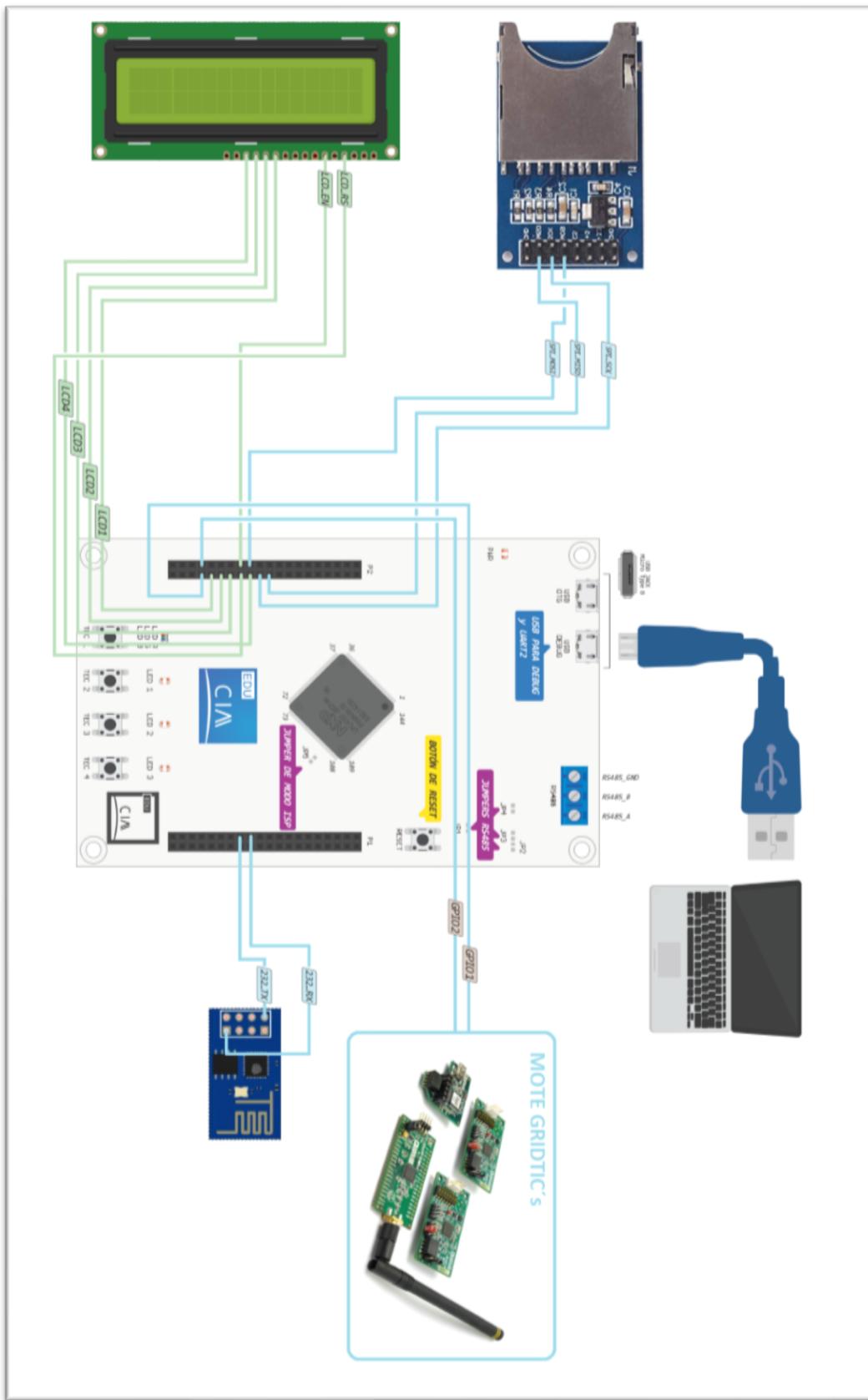
1. La EDU-CIAA tiene 3 UART's.
 - a. El "UART_USB", que se utiliza para conectar la EDU-CIAA a un terminal serie de la PC. Físicamente el mismo se encuentra en el conector micro-usb que se encuentra serigrafiado como "DEBUG". El mismo también sirve para grabar el firmware en la EDU-CIAA, por lo que fue muy importante no utilizarlo para otros propósitos.
 - b. El "UART_RS232". El mismo tiene serigrafiadas las siglas "232_RX" y "232_TX". El mismo se va a utilizar para la comunicación con el ESP8266.
 - c. El "UART_485" ("UART_GPIO"). Si bien tiene una adaptación de tensiones y niveles por medio de hardware especial para eso, mediante el conector serigrafiado como "RS_485", pero se puede obtener directo del LPC4337, sin modificaciones, por los pines serigrafiados como "GPIO1" y "GPIO2". El mismo se va a utilizar para la comunicación con los MOTES de la red WSN.
2. Hay pines dedicados para el manejo de pantallas LCD, 16x2. Los mismos fueron utilizados para su propósito. "LCD_EN", "LCD_RS", "LCD1", "LCD2", "LCD3" y "LCD4".
3. Hay pines de SPI para conectar la placa Ethernet. *Rev 1.0:* Este mismo puerto se utiliza para grabar la memoria SD.



La siguiente imagen muestra el resultado final:



La siguiente imagen muestra cómo queda el circuito en la Rev 1.0:





6.3.1 HARDWARE GRIDTICs

Desde el laboratorio se brinda el coordinador de la red de sensores WSN. Dicho coordinador, tal como se puede ver en las imágenes anteriores, está conectado mediante UART a la EDU-CIAA.

model: deRFmega128-22A00 (con el deRFtoRCB-Adapter incluido).



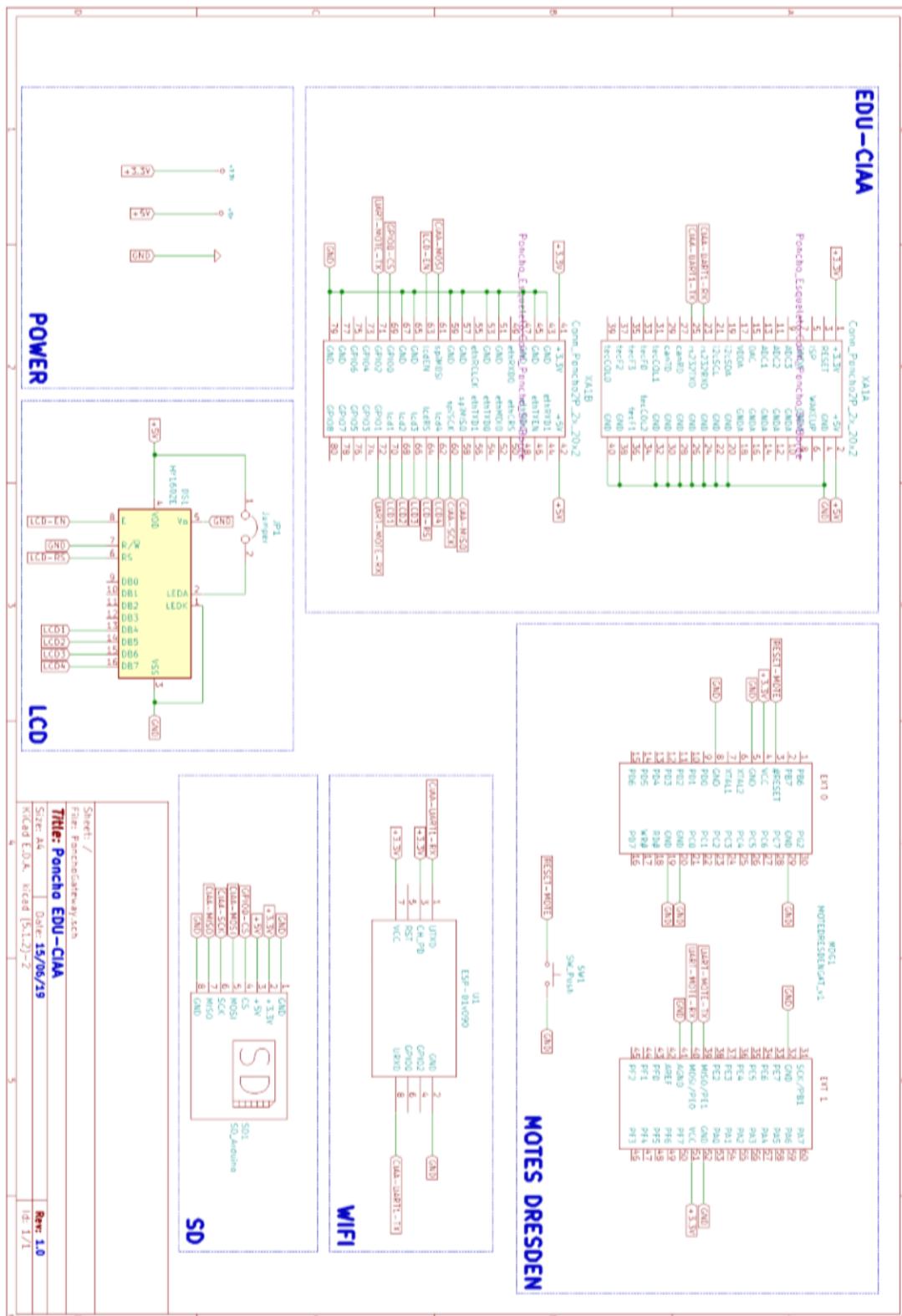
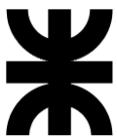
6.3.2 HARDWARE REALIZADO

Para integrar todos los componentes a la EDU-CIAA se decide realizar un “Poncho”, que es una placa que se acopla directamente a los conectores de la EDU-CIAA, sin necesidad de cables adicionales, equivalente en otras tecnologías a un SHIELD.

Dicho diseño se realizó en KiCAD, un software libre de diseño de PCB, reconocido mundialmente, siguiendo los lineamientos del proyecto completo en sí mismo, y para poder colaborar en él.

Para ver los lineamientos de los “Ponchos” de la EDU-CIAA se puede seguir la dirección que se muestra en la bibliografía. (4)

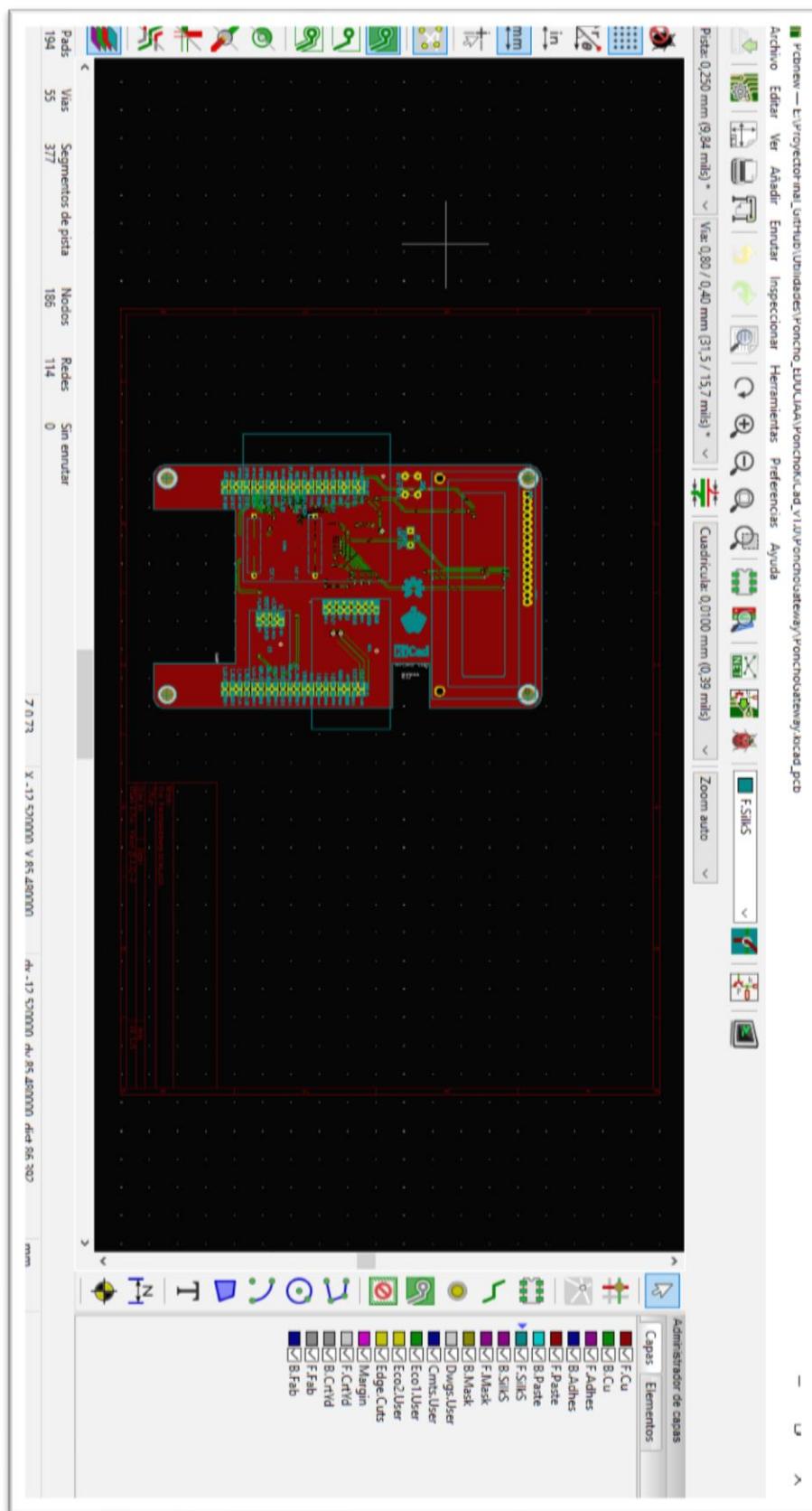
Siguiendo con las conexiones realizadas en la etapa de pruebas, se generó el “PonchoGateway”, cuyas imágenes se pueden apreciar a continuación.

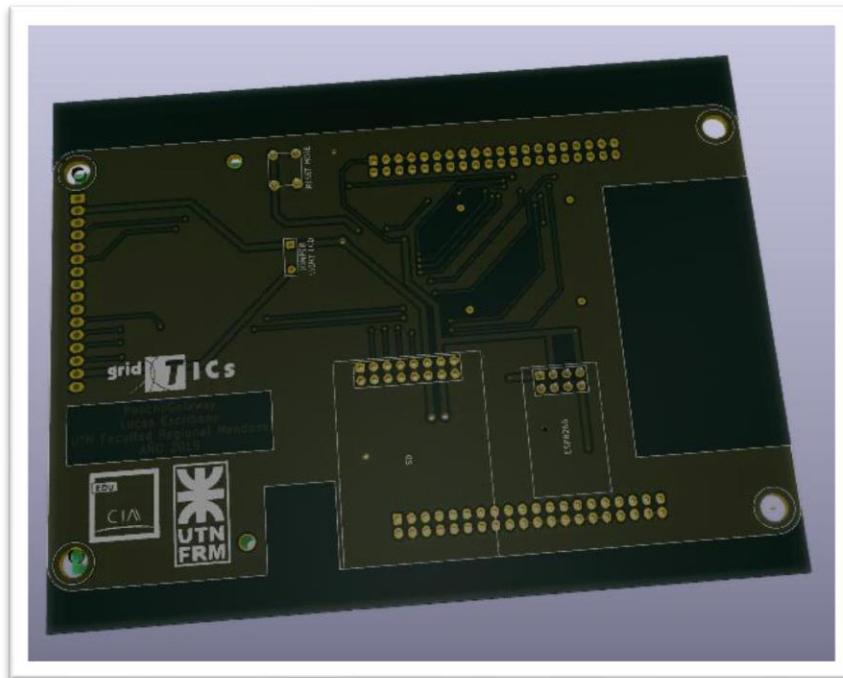
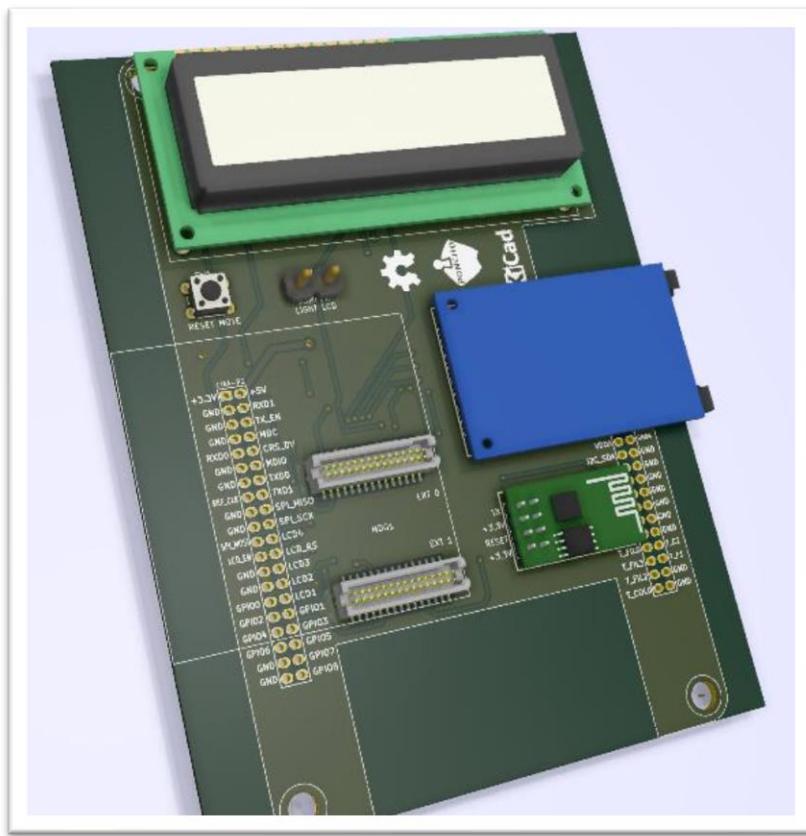


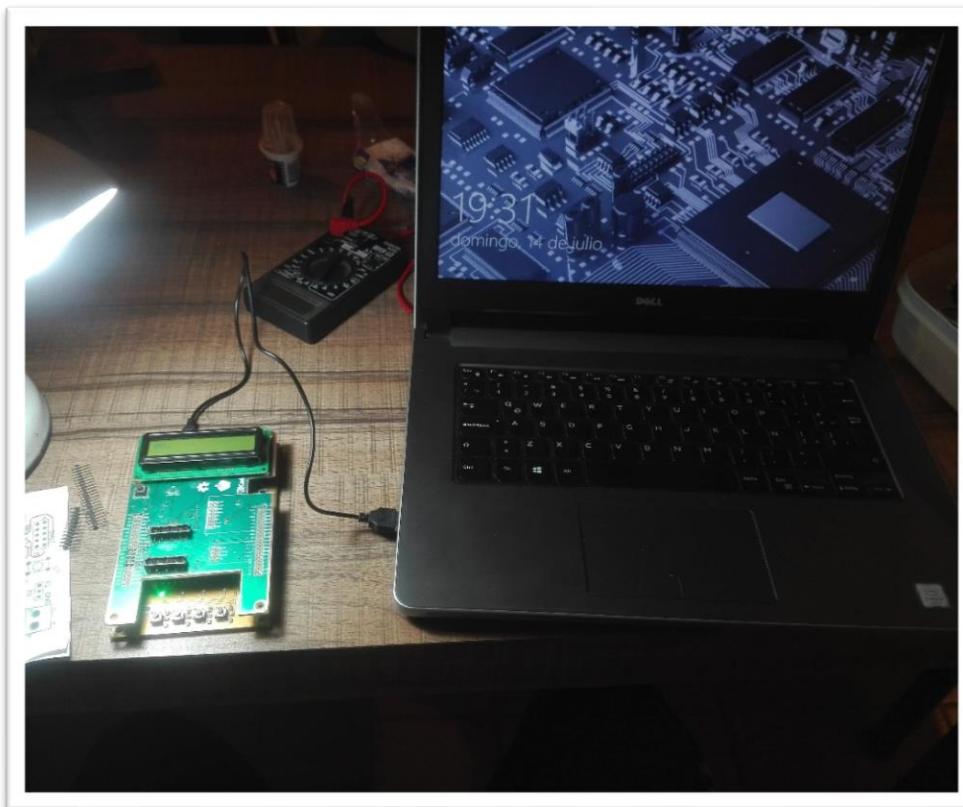


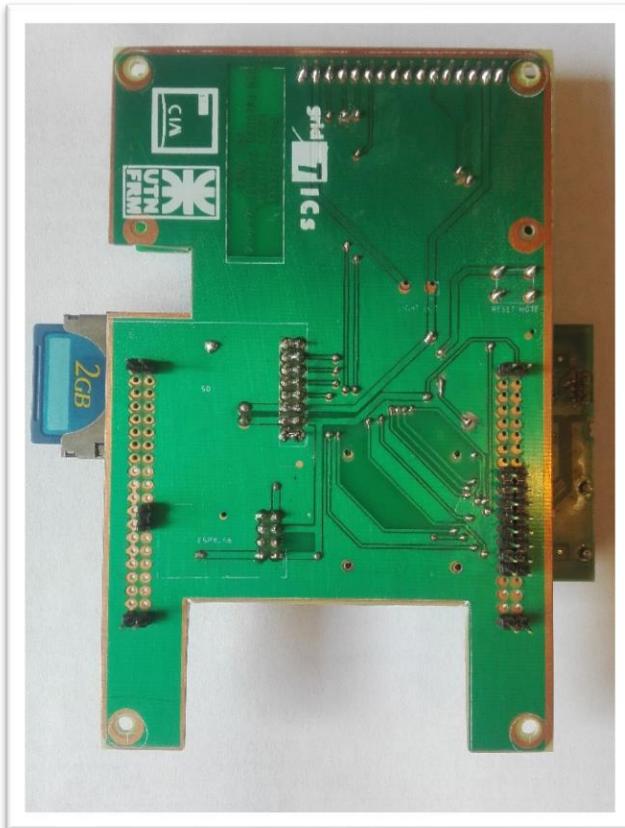
GATEWAY CIAA

Escribano - Gonzalez
Versión: 1.3









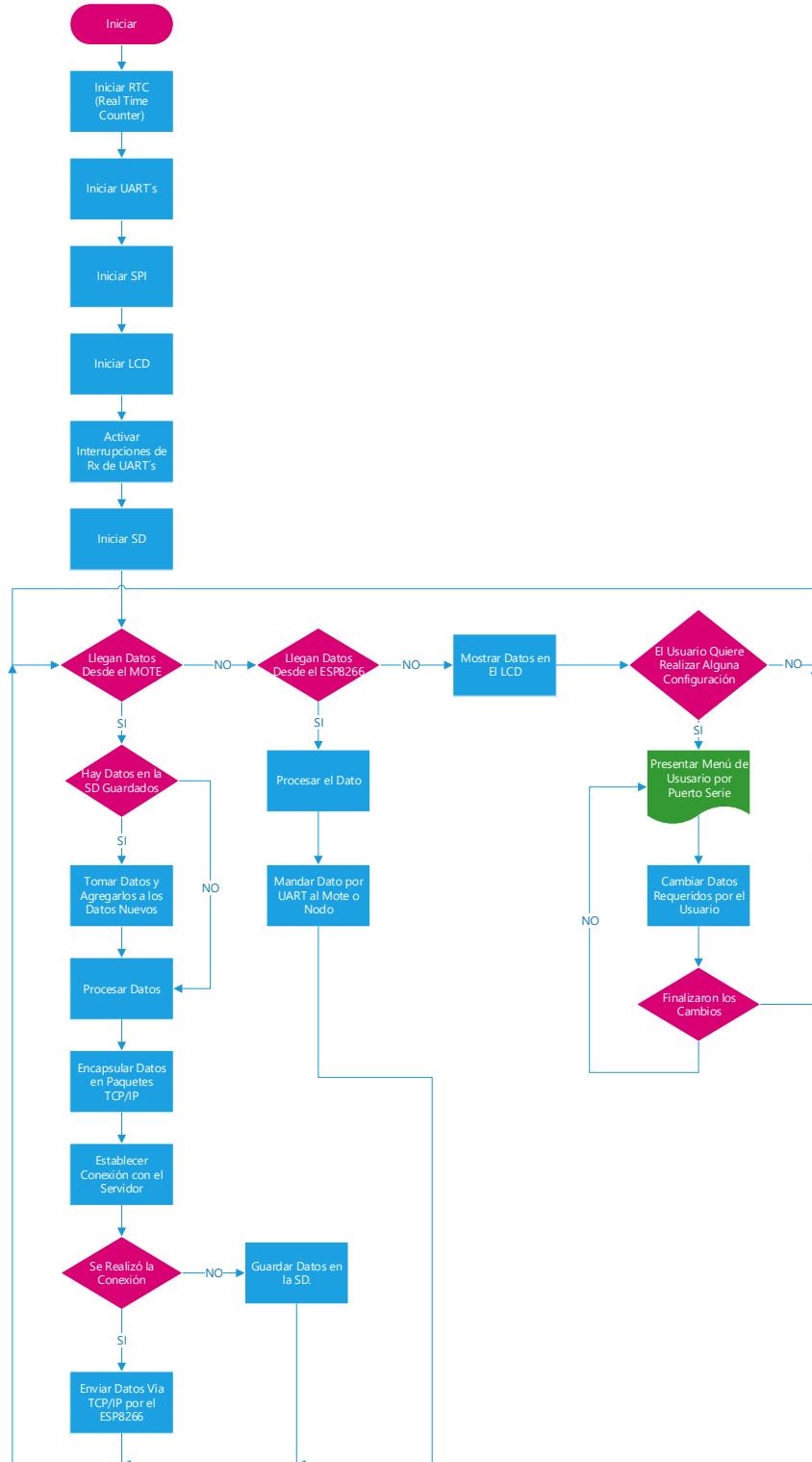


6.4 SOFTWARE

6.4.1 DIAGRAMA DE FLUJO DE FUNCIONAMIENTO

Antes de comenzar la programación del firmware es esencial establecer un diagrama de flujo que ayude a la comprensión del funcionamiento del mismo.

El mismo se presenta como sigue:





6.4.2 CONFIGURACIONES INICIALES

Como primera medida se procede a evaluar cómo se programará la EDU-CIAA. Desde la página principal del proyecto.

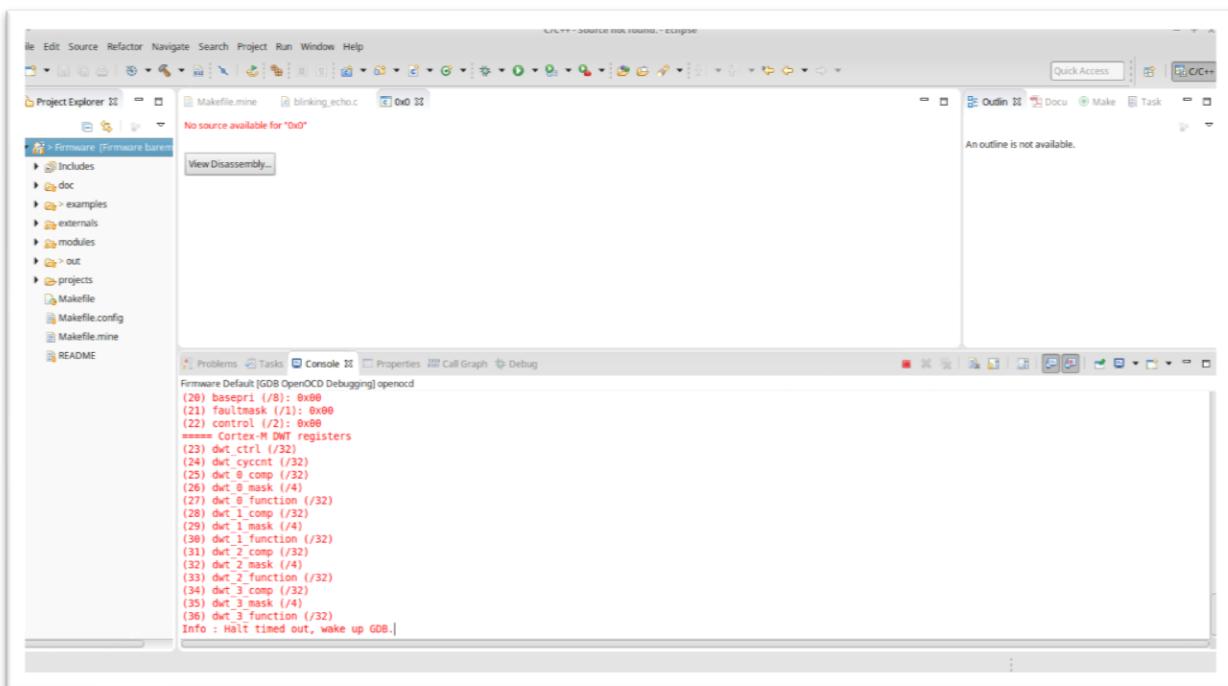
A simple vista se puede visualizar los distintos lenguajes de programación con los cuales se puede realizar el mismo:

- ✓ Lenguaje C
- ✓ Micropython
- ✓ Ladder
- ✓ CIAABOT (Es una plataforma de robótica educativa que permite programar de manera gráfica y está diseñada para la enseñanza de sistemas embebidos.)
- ✓ Java

Se elige Lenguaje C por poseer mayores conocimientos en él.

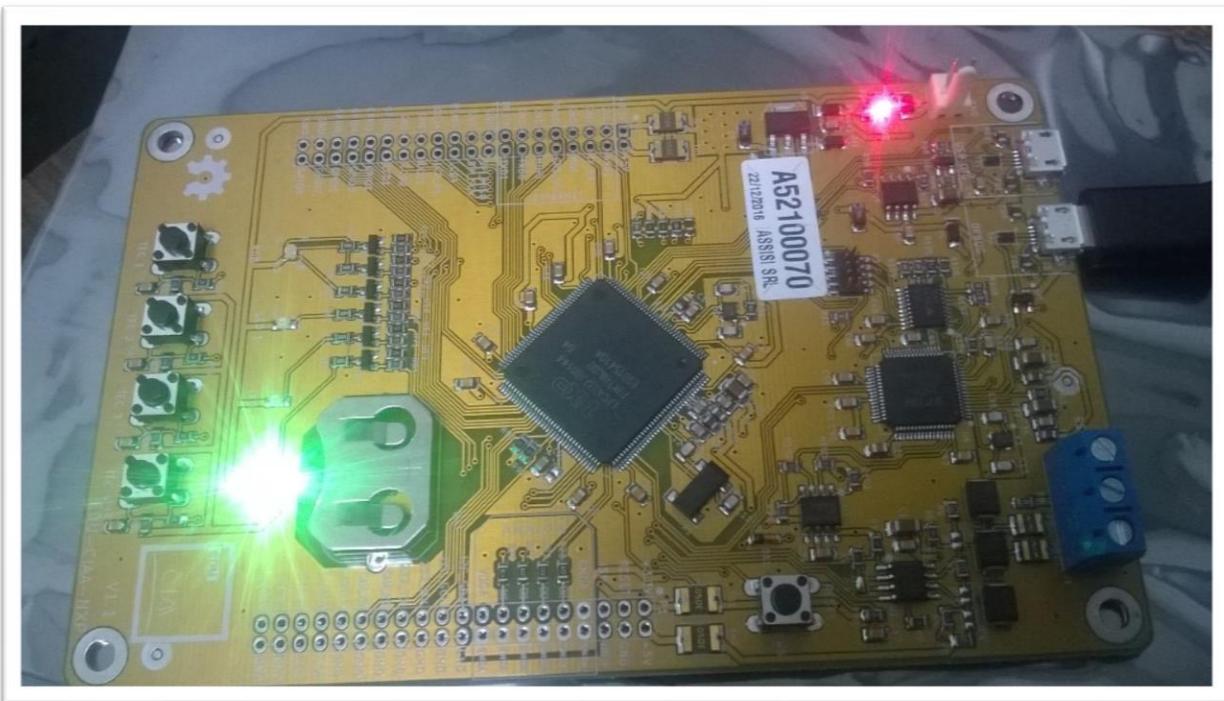
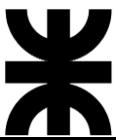
Luego se eligen las sugerencias de la página para dejar instalado el entorno de programación:

- 1- Se instala Cygwin (compilador gcc).
- 2- Se instala el IDE Eclipse.



- 3- Se instala el plugin para Eclipse: Plug-In GNU-ARM-OpenOCD.
- 4- Se configura el entorno tal cual las especificaciones de la página.

Una vez realizado lo anterior se procede a cargar el ejemplo "blinking" de la librería examples existente.



Tanto las configuraciones, como la carga del ejemplo y la visualización en la placa fueron correctos. Podemos decir que ya estamos listos para comenzar el proyecto.

Rev1.0

Se cambia el software de programación dado que los creadores de la EDU-CIAA, realizaron sobre la marcha un entorno gráfico (GUI) nuevo, basado también en Eclipse, pero mucho más amigable.

```
/* PROYECTO FINAL DE INGENIERÍA ELECTRÓNICA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL MENDOZA
Autor: Lucas Escribano - Rodrigo Gonzalez
Año: 2014

*/
// **** INCLUDES ****
#include "sapi.h"
#include <string.h>

//Inclusiones propias
#include "ESP8622_driver.h"           //Driver de la placa de WiFi
#include "RTC_Config.h"               //Funciones de el Real Time Counter
#include "FAT_SD.h"                  //Funciones de la SD
#include "Interrupciones_UART.h"       //Funciones de las Interrupciones de todos los UART's
#include "NTP_Config.h"              //Funciones de las conexiones con el Server NTP

// **** DECLARACIONES ****
// **** VARIABLES GLOBALES ****
```

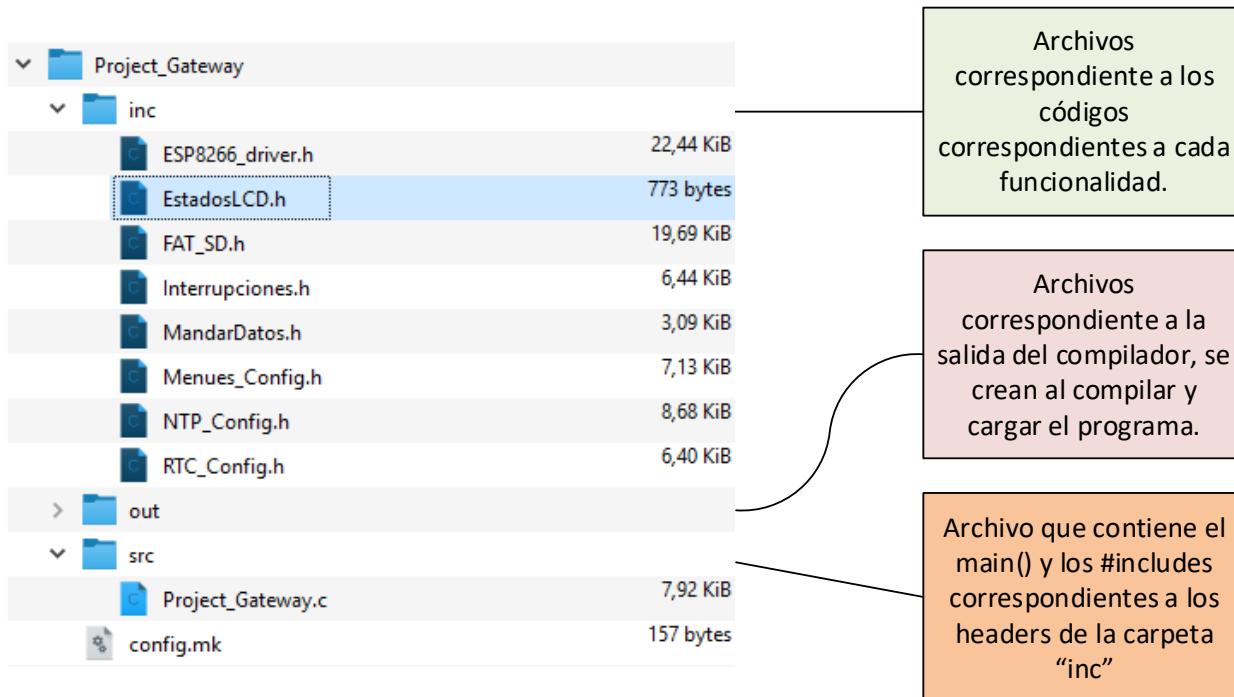


6.4.3 PROGRAMACIÓN EN C

Se procede a realizar el software que se implementará en la EDU-CIAA para controlar todos los dispositivos.

Como primera instancia se va a conservar una estructura de programación para que sea fácil hacer un seguimiento del mismo.

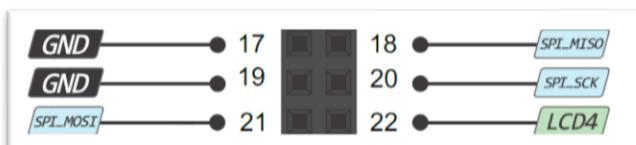
Dicha estructura, aunque no responde a lo habitual en programación de sistemas embebidos, se trata de la siguiente manera:



De esta manera se procura que sea fácil encontrar problemas que puedan surgir a lo largo de la programación.

6.4.3.1 SPI (ETHERNET)

Con ayuda del datasheet del micro LPC4337, podemos realizar pruebas con el SPI. Antes que nada hay que encontrar los pines en la EDU-CIAA que se utilizarán para tales propósitos, con la información citada en anexos¹ podemos verificar los pines de salida a utilizar:



¹ ANEXO – INFORMACIÓN LPC 4337 – EDU-CIAA PIN OUT



SPI_SCK	●	120	PF_4	SSP1_SCK	GP_CLKIN	TRACECLK	R	R	R	I2S0_TX_MCLK	I2S0_RX_SCK
SPI_MISO	●	44	P1_3	GPIO0[10]	CTOUT_8	SGPIO10	EMC_OE	USB0_IND1	SSP1_MISO	R	SD_RST
SPI_MOSI	●	47	P1_4	GPIO0[11]	CTOUT_9	SGPIO11	EMC_BLS0	USB0_IND0	SSP1_MOSI	R	SD_VOLT1

Rápidamente nos damos cuenta que en realidad no se utilizó en la conformación de la placa impresa el verdadero puerto SPI del LPC4337 sino que se utilizó el SSP1(aparentemente por poseer mayores prestaciones los pines que apuntan a dicho puerto). Esto está correctamente indicado en el datasheet del micro:

P1_3		P5	M2	J1	44	[2]	N;	PU	I/O	GPIO0[10] — General purpose digital input/output pin.
									O	CTOUT_8 — SCT output 8. Match output 0 of timer 2.
									I/O	SGPIO10 — General purpose digital input/output pin.
									O	EMC_OE — LOW active Output Enable signal.
									O	USB0_IND1 — USB0 port indicator LED control output 1.
									I/O	SSP1_MISO — Master In Slave Out for SSP1.
									-	R — Function reserved.
									O	SD_RST — SD/MMC reset signal for MMC4.4 card.
P1_4		T3	P2	J2	47	[2]	N;	PU	I/O	GPIO0[11] — General purpose digital input/output pin.
									O	CTOUT_9 — SCT output 9. Match output 3 of timer 3.
									I/O	SGPIO11 — General purpose digital input/output pin.
									O	EMC_BLS0 — LOW active Byte Lane select signal 0.
									O	USB0_IND0 — USB0 port indicator LED control output 0.
									I/O	SSP1_MOSI — Master Out Slave in for SSP1.
									-	R — Function reserved.
									O	SD_VOLT1 — SD/MMC bus voltage select output 1.
PF_4		D10	H4	172	120	[2]	O;	PU	I/O	SSP1_SCK — Serial clock for SSP1.
									I	GP_CLKIN — General purpose clock input to the CGU.
									O	TRACECLK — Trace clock.
									-	R — Function reserved.
									-	R — Function reserved.
									-	R — Function reserved.
									O	I2S0_TX_MCLK — I2S transmit master clock.
									I/O	I2S0_RX_SCK — I2S receive clock. It is driven by the master and received by the slave. Corresponds to the signal SCK in the I ² S-bus specification.

Como primera medida se configuran los pines del micro. El micro posee multiplexores en cada salida (en realidad son registros de memoria), en los cuales, dependiendo de la configuración que se haga de los pines, es la funcionalidad que se obtiene. Para ello nos valemos de una colección de bibliotecas, más específicamente, la de periferia. El mismo denominó la misma como sAPI.

Eso se configura en el programa de la siguiente manera:

Chip_SCU_PinMuxSet() es la función del cortex que asigna la funcionalidad de las patas del chip.



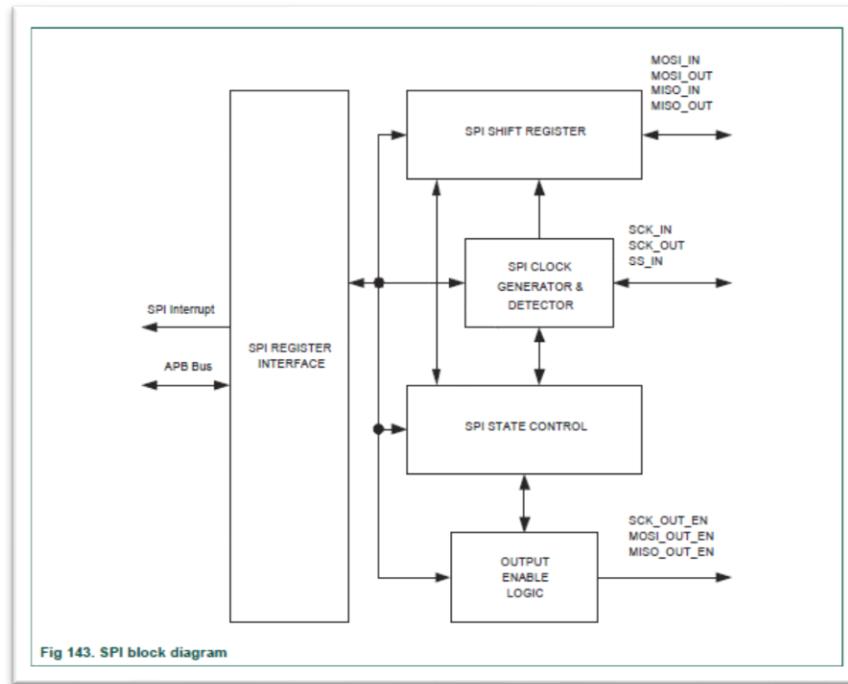
Su sintaxis es la siguiente:

Chip_SCU_PinMuxSet(Primer número del pin, Segundo número del pin, Función del Pin)
pudiendo agregar más de una característica en la función del pin.

Se preseñó una función en el programa, de la librería del propio procesador:

```
void Board_SSP_Init(LPC_SSP_T *pSSP)
{
    if (pSSP == LPC_SSP1) {
        Chip_SCU_PinMuxSet(0x1, 5, (SCU_PINIO_FAST | SCU_MODE_FUNC5)); /* P1.5 => SSEL1 */
        Chip_SCU_PinMuxSet(0xF, 4, (SCU_PINIO_FAST | SCU_MODE_FUNC0)); /* PF.4 => SCK1 */

        Chip_SCU_PinMuxSet(0x1, 4, (SCU_MODE_INACT | SCU_MODE_INBUFF_EN
| SCU_MODE_ZIF_DIS | SCU_MODE_FUNC5)); /* P1.4 => MOSI1 */
        Chip_SCU_PinMuxSet(0x1, 3, (SCU_MODE_INACT | SCU_MODE_INBUFF_EN
| SCU_MODE_ZIF_DIS | SCU_MODE_FUNC5)); /* P1.3 => MISO1 */
    }
    else {
        return;
    }
}
```



EL diagrama de tiempos dependiente de la configuración es el que sigue:

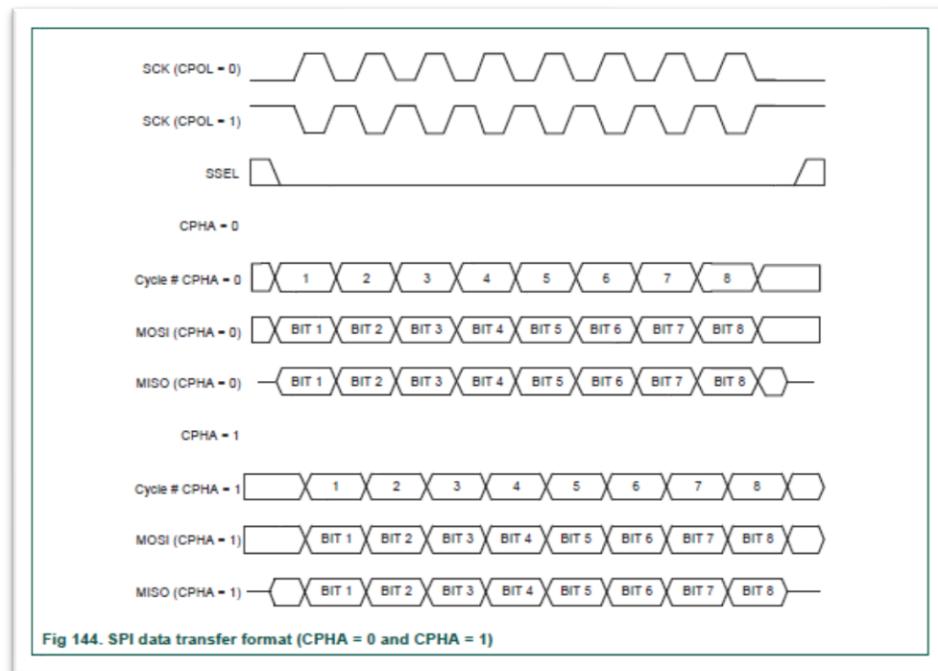


Fig 144. SPI data transfer format (CPHA = 0 and CPHA = 1)

Los pasos a seguir son proporcionados por el fabricante, y depende si será utilizado como maestro o como esclavo.

Las configuraciones de la placa ethernet según el fabricante son las siguientes:

Commands:

According to SPI protocol, there are only two data lines used between SPI devices. So, it is necessary to define OP-Code. W5100 uses two types of OP-Code - Read OP-Code and Write OP-Code. Except for those two OP-Codes, W5100 will be ignored and no operation will be started.

In SPI Mode, W5100 operates in “unit of 32-bit stream”.

The unit of 32-bit stream is composed of 1 byte OP-Code Field, 2 bytes Address Field and 1byte data Field.

OP-Code, Address and data bytes are transferred with the most significant bit(MSB) first and least significant bit(LSB) last. In other words, the first bit of SPI data is MSB of OP-Code Field and the last bit of SPI data is LSB of Data-Field. W5100 SPI data format is as below.

Command	OP-Code Field		Address Field	Data Field
Write operation	0xF0	1111 0000	2 bytes	1 byte
Read operation	0x0F	0000 1111	2 bytes	1 byte

1. Configure Input/Output direction on SPI Master device pins.

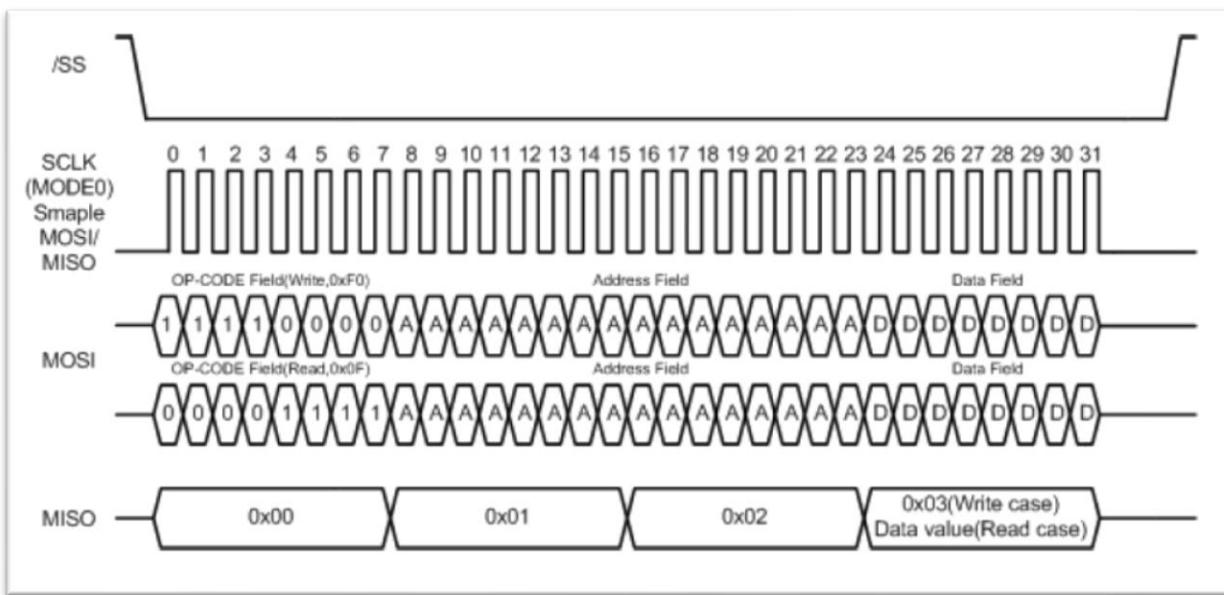
* /SS (Slave Select) : Output pin

* SCLK (Serial Clock) : Output pin

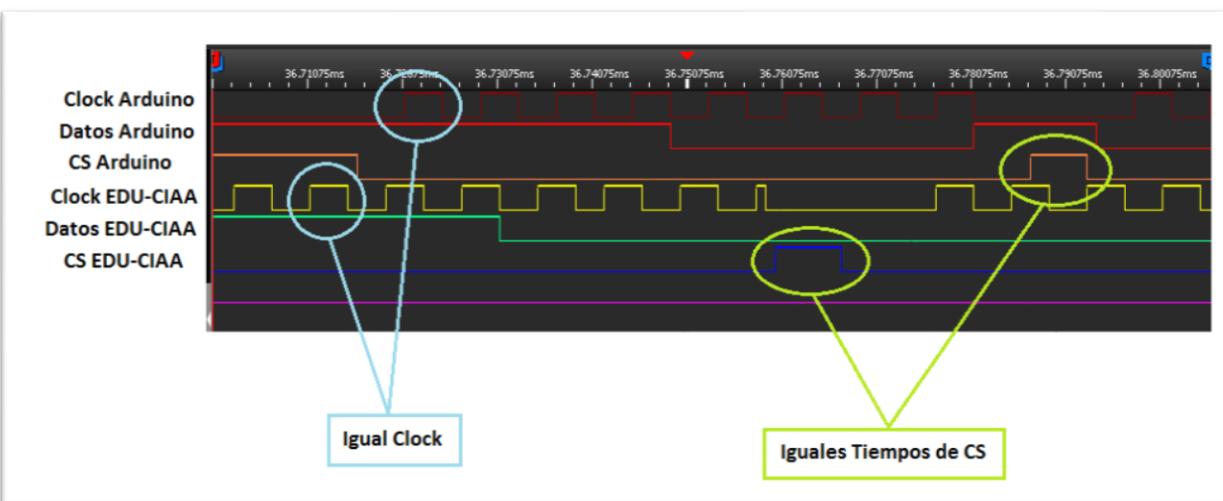
* MOSI (Master Out Slave In) : Output pin



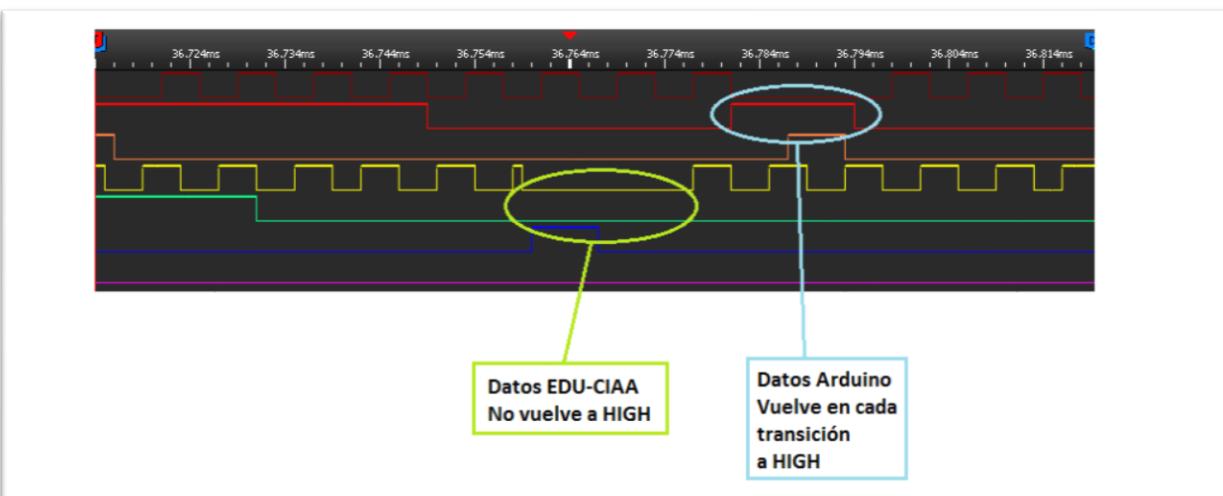
- * MISO (Master In Slave Out) : Input pin
- 2. Configure /SS as 'High'
- 3. Configure the registers on SPI Master device.
 - * SPI Enable bit on SPCR register (SPI Control Register)
 - * Master/Slave select bit on SPCR register
 - * SPI Mode bit on SPCR register
 - * SPI data rate bit on SPCR register and SPSR register (SPI State Register)
- 4. Write desired value for transmission on SPDR register (SPI Data Register).
- 5. Configure /SS as 'Low' (data transfer start)
- 6. Wait for reception complete
- 7. If all data transmission ends, configure /SS as 'High'



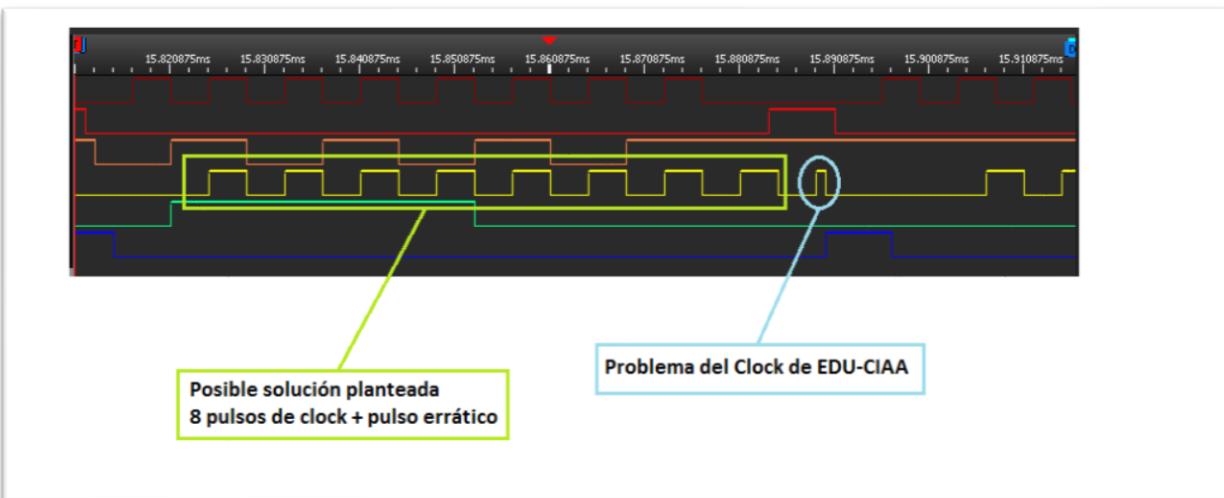
Dichos pasos fueron correctamente realizados en el programa.
Al probar la placa ethernet, nos dimos cuenta que la misma no toma las configuraciones que le enviamos mediante comandos por el puerto SPI.
Es por ello que la placa se somete a un TEST con equipamiento de adquisición de datos que el departamento GridTic's nos ofrece.
Se comparan el puerto SPI de una placa ARDUINO MEGA 2560, con el puerto SPI de la EDU-CIAA. Los resultados obtenidos son los que siguen:



Se observa que los tiempos son exactamente iguales, tanto los clocks como los chip select (CS).



Pero se pueden ver dos diferencias muy marcadas. La primera es que los DATOS en el ARDUINO vuelven a estado HIGH cada vez que termina de transmitir un dato. Los DATOS de la EDU-CIAA no vuelven a ese estado.



Otro punto a tener en cuenta y muy notable es que el último pulso de clock del puerto SPI de la EDU-CIAA es de mucho menor tiempo que los otros pulsos.

Como solución a esto último se procede a tomar 9 pulsos de clock en vez de 8, pero esto tampoco da buenos resultados. La placa Ethernet no se graba correctamente.

Dicha situación se plantea al grupo de investigación del proyecto EDU-CIAA. Se queda de acuerdo que estamos ante un problema del hardware del Micro LPC4337 de NXP (Phillips), en el puerto SPI.

Por lo mencionado anteriormente se decide:

EL PROYECTO GATEWAY QUEDARÁ SIN CONEXIÓN ETHERNET.

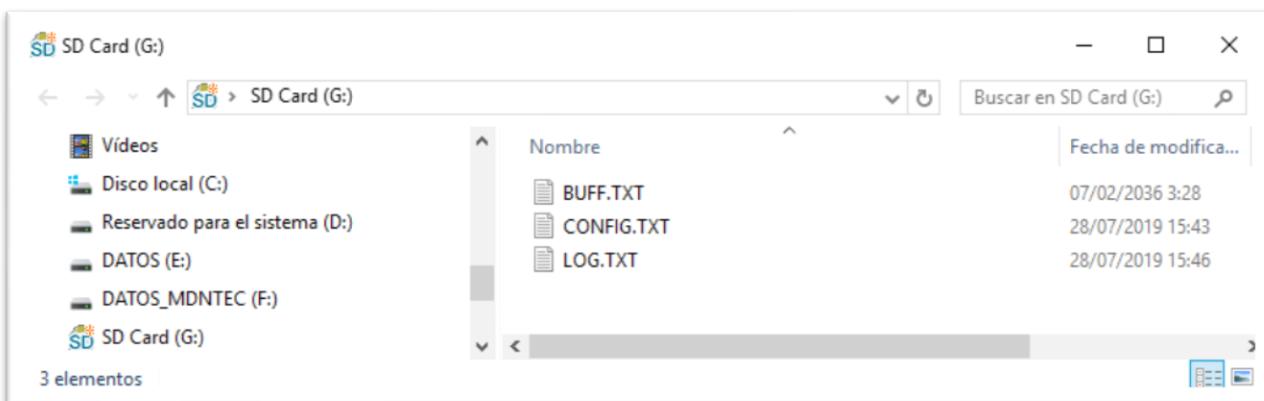
6.4.3.2 SPI (SD)

Aunque el puerto SPI no pudo utilizarse para grabar específicamente el módulo que vimos anteriormente, se comprueba que funciona perfectamente para la grabación de las tarjetas de memoria.

Para dicha tarea se utiliza una biblioteca realizada por el fabricante del LPC4337. La misma se denomina LPCUSBLib (5), la provee PHILLIPS.

El driver de la tarjeta SD se encuentra realizado en el archivo "FAT_SD.h".

En la tarjeta SD encontraremos 3 archivos que vale la pena mencionar:





Cada vez que se inicia la EDU-CIAA se comprueba la existencia de dichos archivos, de no encontrarse alguno, se crea.

1. BUFF.TXT: Este archivo sirve de buffer para guardar los paquetes que no se pueden transmitir cuando el server TCP/UDP/IP se desconecta por alguna razón. No se debe modificar.
2. CONFIG.TXT: Este es el más importante de los tres. En él se encuentran las configuraciones del Gateway, y tiene una sintaxis determinada que se muestra a continuación:

```
ssid_wifi =SSID;
clave_wifi =Clave;
server_tcpip =192.168.0.101;
puerto_tcpip =2222;
servidor_ntp =3.ar.pool.ntp.org;
```

Sobre todo lo más importante es no dejar espacios entre los signos “=” (igual) y los signos “;” (punto y coma). Son 5 parámetros, en orden, nombre de la red WiFi, clave de la red WiFi, dirección IP o http del servidor TCP, puerto por el cual se va a establecer la comunicación con el servidor TCP, y la dirección del servidor NTP que también puede ser IP o http.

3. LOG.TXT: Es un registro de las actividades que realiza la EDU-CIAA, también tiene una sintaxis determinada, pero no debe ser modificado por el usuario. Es para tener un respaldo de lo que sucede temporalmente.

```
Jul 20 2019 19:53:20 ; La EDU-CIAA es reiniciada por causas desconocidas.
Jul 20 2019 19:53:20 ; No hay conexión TCP con el servidor, se guarda el paquete en la SD.
Jul 20 2019 19:53:20 : La EDU-CIAA es reiniciada por causas desconocidas.
```

Se puede ver que en el archivo se encuentra la fecha, la hora y la causa del log.

6.4.3.3 ESP8266(WIFI)

El módulo wifi se configura por el puerto UART, denominado UART_ESP en el programa. El driver de configuración para todas las funcionalidades del módulo se encuentra en el archivo “ESP8266_driver.h”, y se realiza enteramente con comandos AT.

El datasheet de comandos y configuraciones se encuentra disponible en la web citada en las referencias (6).

6.4.3.4 INTERRUPCIONES

La mayoría de las operaciones en el software se realizan mediante interrupciones del MCU.



En el archivo “Interrupciones.h” se encuentran las interrupciones utilizadas en el software.

Se encuentran allí las interrupciones de los distintos UART’s, que se ejecutan cuando llega un dato al buffer de recepción del UART correspondiente.

También están las interrupciones de los TIMER’s, que se ejecutan cada cierto período seteado previamente.

Se muestra el manejo de la interrupción del UART correspondiente al coordinador de la red WSN:

```
'/Funcion de Interrupcion del MOTE, cada vez que llega un dato Ingresa Aqui.
void INT_GPIO_RX()
{
    *punt_rx_gpio = uartRxRead(UART_GPIO);
    uartWriteByte(UART_USB, *punt_rx_gpio);

    switch(estadoInt)
    {
        case MOTE_ESTADO_INICIAL:
            if(*punt_rx_gpio == 0x7E)estadoInt = MOTE_BUSCANDO_LONG;
            else    ResetGpioBuff();
        break;

        case MOTE_BUSCANDO_LONG:
            cantDatos = *punt_rx_gpio;
            estadoInt = MOTE_RECIBIENDO_DATO;
        break;

        case MOTE_RECIBIENDO_DATO:
            cuentaDatos++;
            if(cuentaDatos == (cantDatos))estadoInt = MOTE_ENVIANDO_TRAAMA;
        break;

        case MOTE_ENVIANDO_TRAAMA:
            bandera_mandar_datos = true;
        break;
    }

    //Solo aumentamos el puntero si no estoy en ESTADO_INICIAL
    if(estadoInt != MOTE_ESTADO_INICIAL)punt_rx_gpio++;
}
```

Se puede ver que las interrupciones (para que sean más eficientes), se manejan con una pseudo máquina de estados, en donde, dependiendo de los datos que se reciban es en qué estado de la recepción nos encontraremos.

Como comentario particular de esta función se puede ver que se habilita una bandera (“bandera_mandar_datos”), que se va a ejecutar en el main(), esto se debe a que el timer del micro, como es de esperarse, se detiene cada vez que ingreso en una interrupción,



por lo tanto si necesito referencias temporales no puedo utilizar la rutina mencionada anteriormente.

6.4.3.5 RTC (REAL TIME COUNTER)

Las configuraciones del RTC propio de la EDU-CIAA se encuentran en el archivo “RTC_Config.h”.

En dicho archivo podemos encontrar funciones para escribir y leer el RTC con distintos formatos.

6.4.3.6 NTP(NETWORK TIME PROTOCOL)

Network Time Protocol (NTP) es un protocolo de Internet para sincronizar los relojes de los sistemas informáticos a través del enrutamiento de paquetes en redes con latencia variable. NTP utiliza UDP como su capa de transporte, usando el puerto 123. Está diseñado para resistir los efectos de la latencia variable.

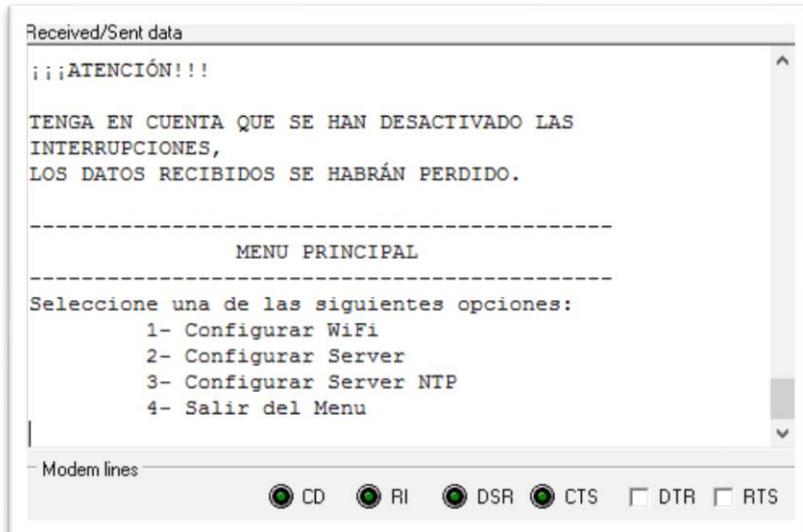
Todas las funciones necesarias para tomar la fecha y la hora actual desde un servidor NTP se encuentran el archivo “NTP_Config.h”.

La hora y fecha tomada del servidor NTP, se guardan automáticamente en el RTC de la EDU-CIAA, para que la hora en todo momento sea la correcta, y siempre esté sincronizado el sistema.

6.4.3.7 MENÚ DE CONFIGURACIÓN

A demás de la configuración que toma la EDU-CIAA desde la SD, también podemos modificarla desde un pequeño menú realizado para la consola serie que podemos conectar mediante el micro-usb serigrafiado con la palabra “DEBUG”, el cual debe ser seteado con una velocidad de 115200 baudios.

En la imagen siguiente se puede ver, mediante consola serie, el menú de configuración principal:





Hay que tener en cuenta, como bien se aprecia en la imagen, que cuando accedemos al menú, todas las interrupciones se deshabilitan, imposibilitando que el dispositivo reciba y mande paquetes como comúnmente lo hace.

Todos los menu's están incluidos en el archivo "Menues_Config.h".

6.4.3.8 DISPLAY LCD

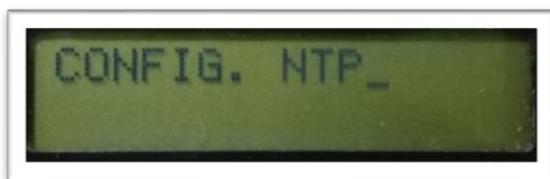
En el display LCD del Gateway podemos a simple vista visualizar distintas cosas.

Como primera instancia, se puede ver el estado del proceso que se realiza en la EDU-CIAA a medida que realiza las configuraciones primarias. Algunos mensajes se pueden ver en la siguiente imagen:

Realizando las configuraciones en el ESP8266:



Configurando el server NTP, tomando la fecha y hora:



Si finalizan las configuraciones y todas son correctas:

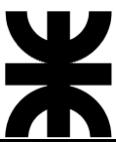


Todos los mensajes que se muestran en la pantalla, junto con las funciones para dicho propósito se encuentran en el header "EstadosLCD.h".

6.4.3.9 GPIO (GENERAL PURPOSE INPUT /OUTPUT)

La EDU-CIAA cuenta con 1 led RGB que muestra en cada instante el estado de la ejecución del programa.

Tiene 3 estados que son:

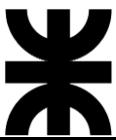


- RGB VERDE: El sistema está operando con normalidad.



- RGB AZUL: El sistema está en proceso de configuración. También se enciende cuando estoy en el menú.

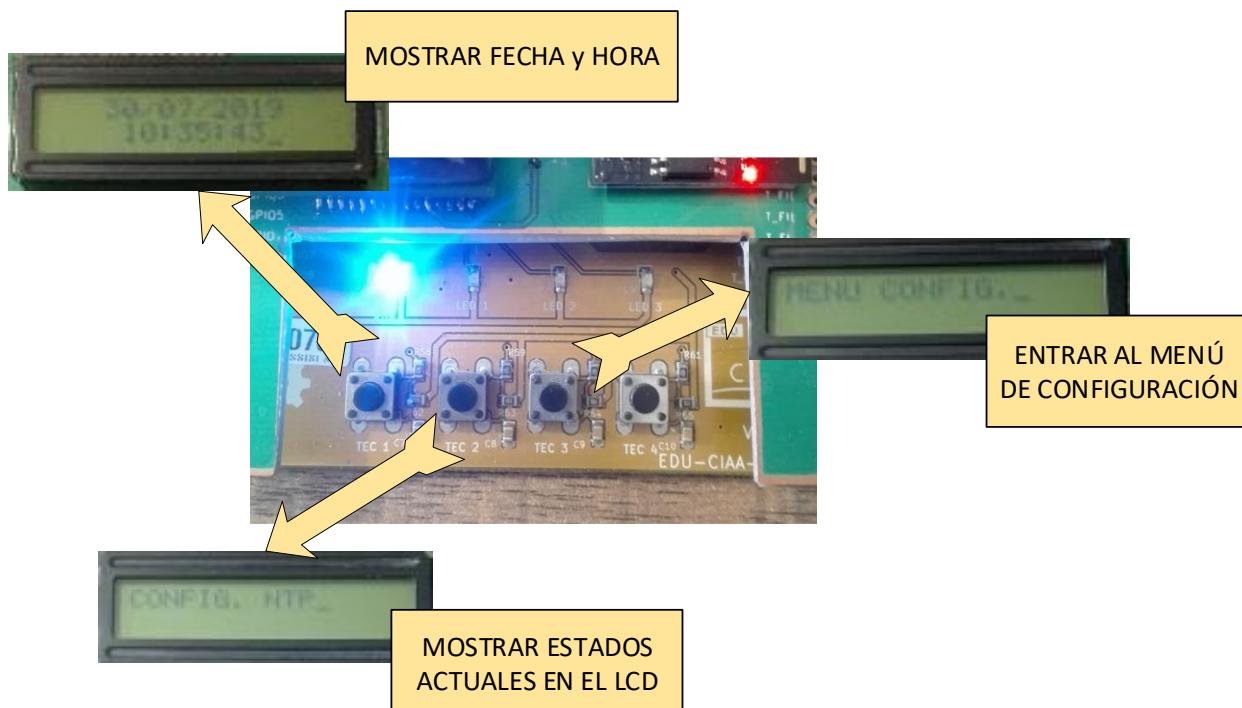




- RGB ROJO: Error el sistema se detuvo por alguna causa y no puede continuar. Hay que reiniciar la EDU-CIAA.



También contamos con 3 pulsadores en la parte inferior los cuales proporcionan las siguientes funcionalidades:





6.4.3.10 EFICIENCIA ENERGÉTICA

El gateway también está pensado para utilizar la menor energía posible. Es por ello que cada vez que se ejecuta el main(), la EDU-CIAA se duerme, o se pasiva. Esto quiere decir que el microprocesador pasa a un estado de Stand-By utilizando funciones muy abreviadas que consumen menor energía.

Esto se ve en el siguiente comando:

```
//Como trabajo todo por interrupciones duermo la EDU-CIAA hasta que la próxima llegue
sleepUntilNextInterrupt();
```

Como todo el programa ha sido planteado de forma que se maneje con interrupciones, la EDU-CIAA se va a despertar (pasar de estado Stand-By a ON) cuando llegue un dato, que es cuando se va a ejecutar alguna de las mencionadas.

Esto claramente reduce el consumo, dado que se puede apreciar directamente con la temperatura del micro, la cual desciende notablemente al utilizar esta práctica.

6.4.3.11 REPOSITORIO DEL SOFTWARE

Toda la programación del software fue trabajada con el concepto de repositorio almacenado en la nube.

El repositorio utilizado fue GitHub (7).

Esto posibilitó la identificación y resolución de problemas, hacer pruebas sin tener preocupaciones en cuanto al borrado de archivos o modificación de alguno, la portabilidad hacia cualquier dispositivo y lugar, la posibilidad de volver a estados anteriores, el seguimiento por parte del jurado del estado del proyecto en todo momento, etc.



A continuación, se muestran imágenes del repositorio del Gateway en GridTic's.

GridTICs / gateway-sipia-ciaa

Code Issues 4 Pull requests 0 Projects 0 Wiki Security Insights Settings

Gateway IoT con placa CIAA proyecto SIPIA Edit

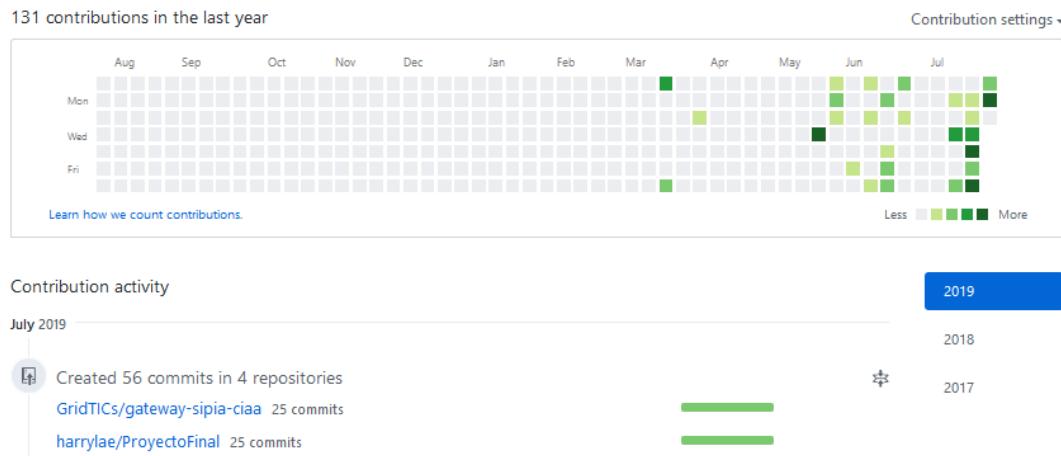
Manage topics

58 commits 2 branches 0 releases 1 contributor BSD-3-Clause

Branch: master New pull request Create new file Upload files Find File Clone or download

Commit	Message	Date
harrylae Trabajando en el Informe. Issue #8	Latest commit a234f40	yesterday
Informe_Proyecto_Final	Trabajando en el Informe. Issue #8	yesterday
Project_Gateway	Trabajando en el Informe. Issue #8	yesterday
Utilidades	Probando la Placa -PonchoGateway-, funciona todo correctamente. Hay q...	15 days ago
libs	First Commit by Lucas Escribano	5 months ago
sapi	Probando la Placa -PonchoGateway-, funciona todo correctamente. Hay q...	15 days ago
scripts/openocd	First Commit by Lucas Escribano	5 months ago
LICENSE	First Commit by Lucas Escribano	5 months ago
Makefile	First Commit by Lucas Escribano	5 months ago
README.md	Trabajando en el README.md	13 days ago

Se pueden ver las contribuciones al repositorio mencionado:



El repositorio se puede ver y descargar en el siguiente enlace:

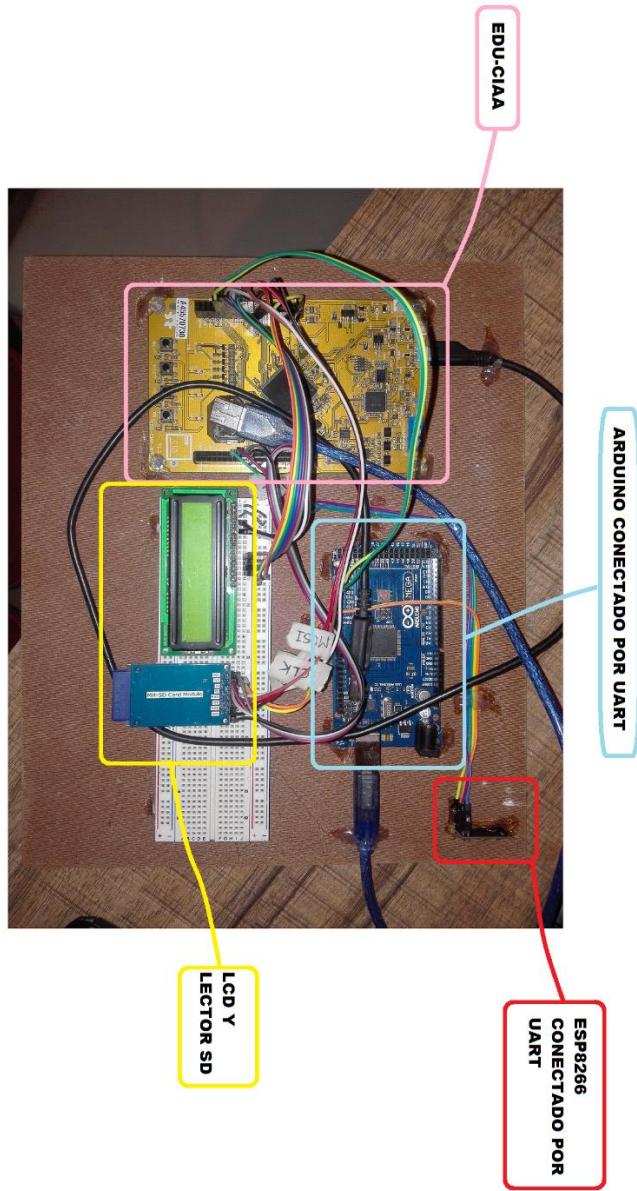
<https://github.com/GridTICs/gateway-sipia-ciaa>



6.5 TESTS

El Gateway ha sido sometido a distintas pruebas de estrés, mediante una configuración del sistema alternativa para dicho propósito.

En la imagen se ve claramente la disposición de los elementos para la realización de las pruebas:

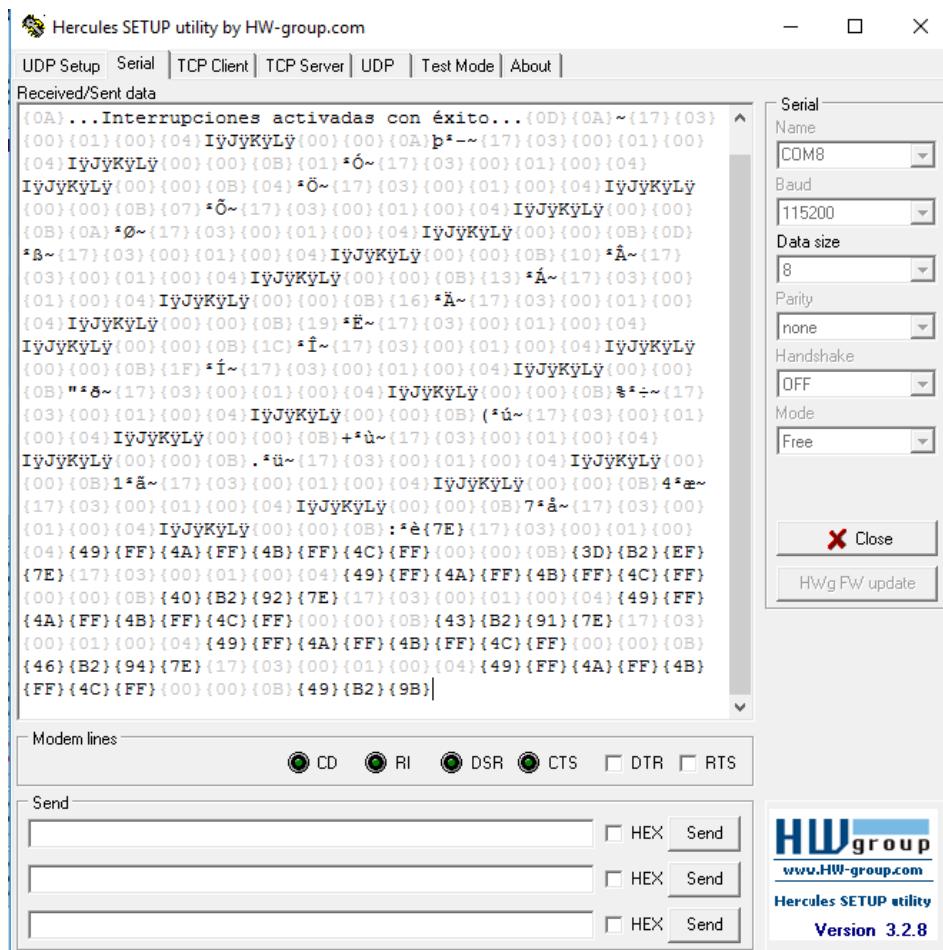


Mediante esta configuración de elementos se pudo evaluar el comportamiento de la placa ante velocidades mucho mas grandes que las de trabajo normal. Dichas pruebas se realizaron en ambas direcciones, esto es, desde el servidor hacia el coordinador de la red WSN, y desde el coordinador hacia el servidor.

En nuestro caso el coordinador fue reemplazado por una placa ARDUINO MEGA 2560, con la misma velocidad del UART que le coordinador (9600 baudios).



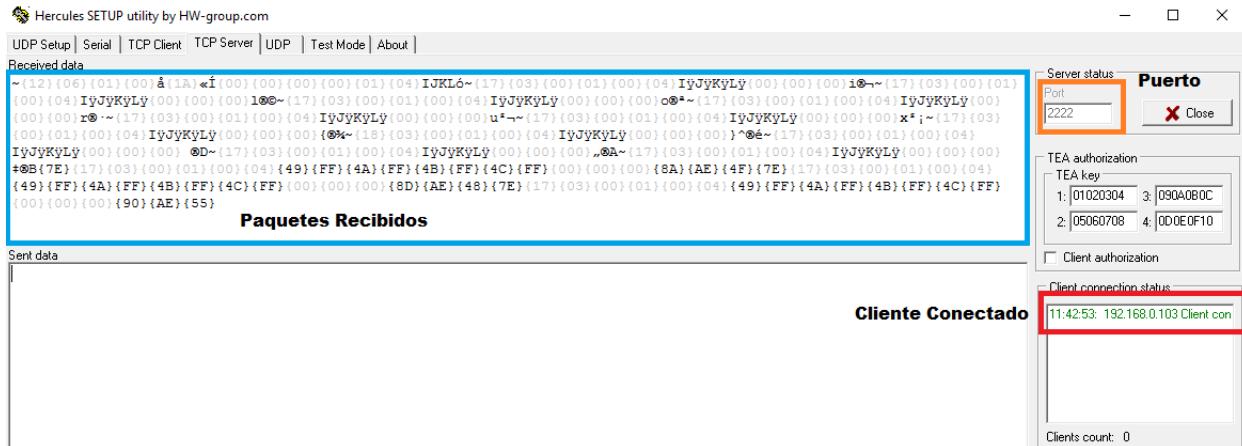
El servidor fue simulado con el programa “Hércules” de la empresa HW-Group (7), que es libre y de fácil configuración.



En la imagen se puede ver como se reciben datos por el UART de la EDU-CIAA, lo que se está haciendo es que cada paquete que se recibe mediante el Arduino, se muestra por pantalla. Hay que tener en cuenta que esto le quita tiempo al Gateway, por lo tanto, para realizar las pruebas esta funcionalidad se desactivó.



En la imagen anterior se muestra lo que se recibe del UART desde el Arduino y luego es enviado al server, que reside en la misma CPU mediante el puerto 2222 y que se está ejecutando en el mismo software, podemos ver lo que está recibiendo dicho server en la imagen siguiente:



Los valores se encuentran representados en dos formatos. La primera parte de los paquetes recibido y enviados está en formato ASCII, pero en la última parte están representados en formato hexadecimal. Comparando ambas imágenes se llega a la conclusión de que los datos enviados y recibidos son idénticos.

Las conclusiones a las pruebas finales fueron:

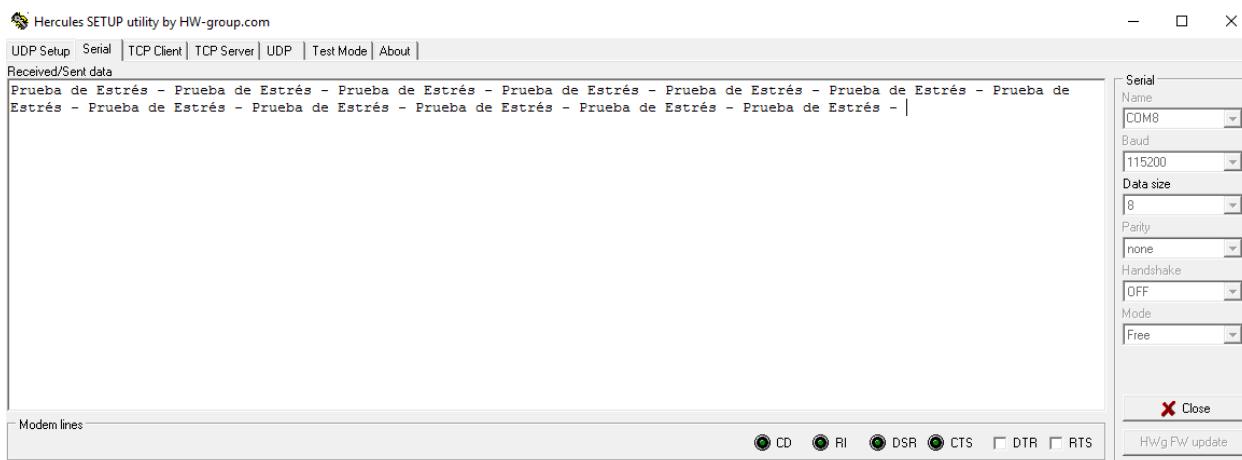
Tiempo entre paquetes recibidos por UART mínimo (CAMINO COORDINADOR → SERVER):

Se puede ver que, si el Gateway tiene conexión con el servidor, el mínimo tiempo entre paquetes, con un máximo de 1024 bytes de longitud, puede llegar a ser de **150ms** sin perder datos y sin errores.

Si el Gateway no tiene conexión con el server y, por lo tanto, los datos deben ser almacenados en la SD el tiempo asciende a **350ms**. sin pérdida de datos y sin errores.



Tiempo entre paquetes recibidos por SERVER mínimo (CAMINO SERVER → COORDINADOR):



En este caso las pruebas de estrés fueron más difíciles de determinar ya que el programa "Hércules" no permite enviar paquetes con una diferencia temporal seteable.

Se enviaron paquetes de 20 bytes manualmente, a razón de aproximadamente 10 paquetes por segundo, sin pérdida de información y sin errores.

Por lo tanto, se puede determinar que el sistema es capaz de recibir paquetes desde el servidor con un espacio temporal de menos de **100ms**.

Estos tiempos superan por mucho a las necesidades temporales de la transmisión de datos del Gateway, donde los paquetes son transmitidos cada intervalos que pueden llegar a ser mucho mayores que 1 minuto.

El programa de prueba de Arduino se encuentra en el repositorio que se mencionó en el apartado anterior.



7 FACTIBILIDAD ECONÓMICA

Todos los cálculos fueron realizados para una TASA del 12%, que corresponde a una tasa internacional.

También se realizó en dólares, por la variabilidad del peso argentino, con una cotización del mismo al momento de realizar dichos cálculos de:

$$1 \text{ peso} = 44,96 \text{ dólares}$$

Cálculo del VAN y TIR	Unidades a Vender	COTIZACIÓN DÓLAR									
TASA	12%	33,44	44,96								
FLUJO DE CAJA	0	1	2	3	4	5	6	7	8	9	10
Unidades Vendidas	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44
Ingresos	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22
Costos y gastos	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01
Amortización bienes de uso	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73
Utilidad antes de impuestos	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48
Impuesto a las ganancias (35%)	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82
Utilidad después de impuestos	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66
Inversión fija	-USS 3.925,71										
Inversión de capital de trabajo	-USS 2.446,62										
Recuperación de capital de trabajo											USS 1.112,10
Flujo de caja neto	-USS 6.372,33	USS 1.065,66	USS 2.177,76								

7.1 APROXIMACIÓN AL VALOR ACTUAL NETO

Para este ejercicio en particular, se calcula la cantidad de unidades para la cual el VAN se hace aproximadamente 0, lo cual es mas representativo en el momento de hacer la presentación de un proyecto real. Se podría interpretar como la cantidad mínima de unidades a vender por año durante el ejercicio para cubrir los costos mínimos.

Cálculo del VAN y TIR	Unidades a Vender	COTIZACIÓN DÓLAR									
TASA	12%	33,44	44,96								
FLUJO DE CAJA	0	1	2	3	4	5	6	7	8	9	10
Unidades Vendidas	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44
Ingresos	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22
Costos y gastos	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01
Amortización bienes de uso	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73
Utilidad antes de impuestos	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48
Impuesto a las ganancias (35%)	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82
Utilidad después de impuestos	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66
Inversión fija	-USS 3.925,71										
Inversión de capital de trabajo	-USS 2.446,62										
Recuperación de capital de trabajo											USS 1.112,10
Flujo de caja neto	-USS 6.372,33	USS 1.065,66	USS 2.177,76								
VAN	USS 6,97										
TIR	12,03%										

Como se aprecia con 33,44 ≈ 34 unidades vendidas por año vamos a obtener un valor VAN de aproximadamente 0.

7.2 TASA INTERNA DE RETORNO

Cálculo del VAN y TIR	Unidades a Vender	COTIZACIÓN DÓLAR									
TASA	12%	33,44	44,96								
FLUJO DE CAJA	0	1	2	3	4	5	6	7	8	9	10
Unidades Vendidas	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44	33,44
Ingresos	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22	USS 15.619,22
Costos y gastos	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01	-USS 13.913,01
Amortización bienes de uso	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73	-USS 66,73
Utilidad antes de impuestos	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48	USS 1.639,48
Impuesto a las ganancias (35%)	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82	-USS 573,82
Utilidad después de impuestos	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66	USS 1.065,66
Inversión fija	-USS 3.925,71										
Inversión de capital de trabajo	-USS 2.446,62										
Recuperación de capital de trabajo											USS 1.112,10
Flujo de caja neto	-USS 6.372,33	USS 1.065,66	USS 2.177,76								
VAN	USS 6,97										
TIR	12,03%										



Se puede ver que el TIR es de 12,03%, que concuerda con la tasa como corresponde, al ser el VAN 0.

7.3 PAYBACK O PLAZO DE RECUPERACIÓN

Periodo	Saldo Inversión	Rent. exigida	Recuperación de la inversión
1	U\$S 6.372,33	U\$S 764,68	U\$S 1.065,66
2	U\$S 6.071,35	U\$S 728,56	U\$S 1.065,66
3	U\$S 5.734,25	U\$S 688,11	U\$S 1.065,66
4	U\$S 5.356,69	U\$S 642,80	U\$S 1.065,66
5	U\$S 4.933,83	U\$S 592,06	U\$S 1.065,66
6	U\$S 4.460,23	U\$S 535,23	U\$S 1.065,66
7	U\$S 3.929,80	U\$S 471,58	U\$S 1.065,66
8	U\$S 3.335,71	U\$S 400,29	U\$S 1.065,66
9	U\$S 2.670,33	U\$S 320,44	U\$S 1.065,66
10	U\$S 1.925,11	U\$S 231,01	U\$S 2.177,76

Como se puede suponer, el plazo de recuperación va a ser la cantidad de años tenidos en cuenta en el ejercicio, o sea, 10 años.



8 PRODUCTOS DE OTROS FABRICANTES

Si bien se encuentran actualmente en la web varios Gateway, ninguno cumple con las especificaciones buscadas en este proyecto, por lo cual no habrá un punto de comparación económico real en este punto.

De todas formas se muestran a continuación productos de similares prestaciones:

8.1 MONNIT

The screenshot shows the Monnit product page for the Alta Ethernet Gateway 4. At the top, there's a navigation bar with links for Products, Applications, Services, Partners, and Support, along with a dropdown for 'Entire Site'. The main product image is a small black device with a blue label that reads 'ALTA' and 'Ethernet Gateway Long Range Wireless Sensors'. To its right, the product name 'ALTA ETHERNET GATEWAY 4' is displayed in bold blue text, followed by the price '\$220.00'. Below the price is the part number 'Part # MNG2-9-EGW-CCE'. A note says 'Region | Frequency Not Selected - Frequency not selected!' with a checkbox. Under 'Product Variations', there's a quantity selector set to '1'. At the bottom are two buttons: 'Add to Cart' (disabled) and 'Customize'.

El precio de venta al público es de u\$s220, que al valor del dólar actual da un total de \$10000 aproximadamente.

Este gateway NO almacena datos, y solo provee una interfaz ethernet para la conexión del server. A demás sólo permite sensores de su protocolo propietario iMonnit™. Esta tecnología NO es compatible con los objetivos de nuestro proyecto y ofrece soluciones muy por debajo del nivel de nuestro producto.

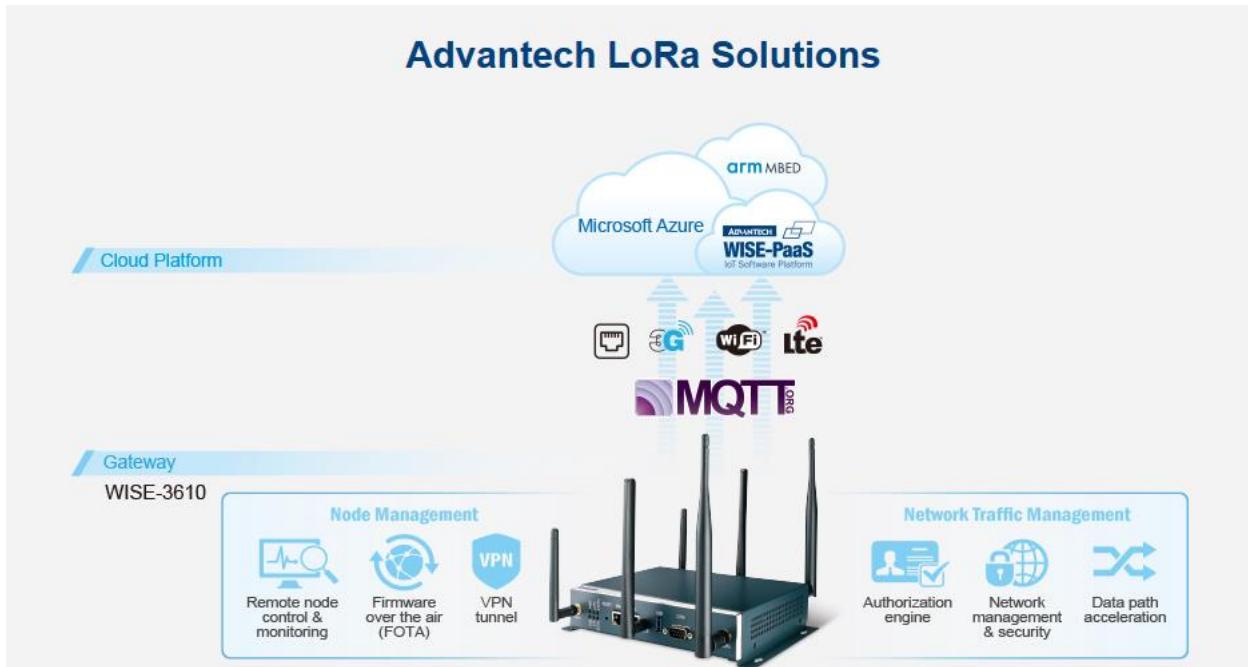
8.2 NATIONAL INSTRUMENTS

The screenshot shows the National Instruments product page for the WSN-9791. At the top, there's a navigation bar with links for INNOVACIONES, PRODUCTOS, SOPORTE, and COMUNIDAD, along with a search bar and a shopping cart icon showing '0'. The main product image is a grey device with a blue label that reads 'WSN-9791' and 'Dispositivo Ethernet Gateway'. To its right, the product name 'WSN-9791' is displayed in large blue text, followed by the description 'Dispositivo Ethernet Gateway'. Below the description is a section titled 'CONTÁCTENOS'. A detailed description of the product follows: 'Dispositivo Ethernet Gateway, Radio IEEE 802.15.4, 2.4 GHz—El WSN-9791 es un Ethernet gateway para dispositivos de red de sensores inalámbrica (WSN). El gateway tiene un radio IEEE 802.15.4 de 2.4 GHz basado en la tecnología ZigBee para adquirir datos de medida desde la red de sensores y un puerto Ethernet de 10/100 Mbits/s para proporcionar conectividad flexible con un controlador principal Windows o SO LabVIEW Real-Time. El software incluido con el WSN-9791 le ayuda a configurar rápidamente su red de sensores y extraer, analizar y presentar sus datos de medida usando el entorno de desarrollo gráfico LabVIEW.' A note at the bottom says 'La antena externa soporta un rango exterior de hasta 300 m con línea de vista y puede alojar 8 nodos finales en una topología en estrella o hasta 36 nodos en una topología de malla.' with a link to '- Mostrar Menos'.



National Instruments ofrece una alternativa más equivalente a nuestro proyecto. Esta solución tiene un costo de u\$s2600, pero acepta el protocolo ZigBee. Hoy esto representa un total de unos \$114000 aproximadamente. De todas formas, tampoco tiene soporte de pérdida de datos.

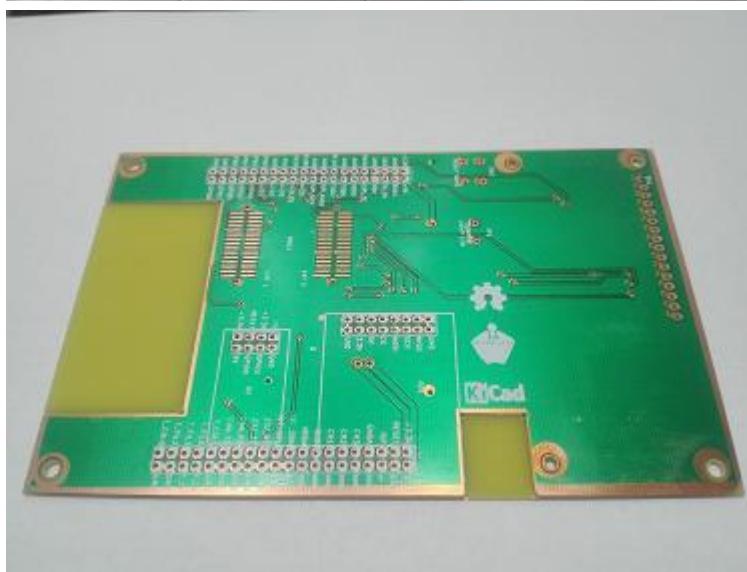
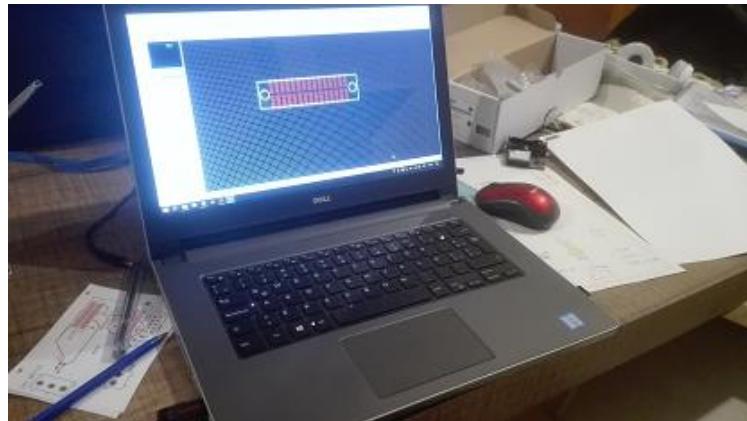
8.3 ADVANTECH

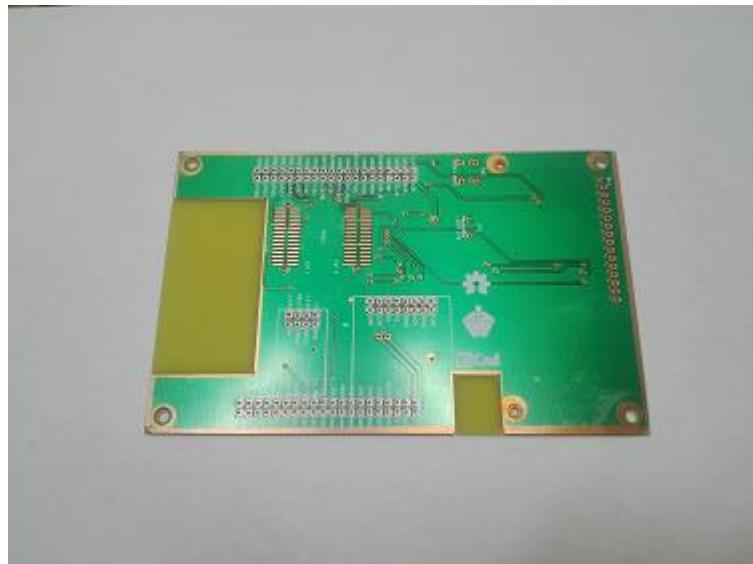


Advantech ofrece una solución con capacidades muy superiores a nuestro proyecto. Da soporte a un sinfín de protocolos y servidores IoT. También soporta GPRS con 3G, 4LTE, Ethernet. Soporta a demás tunelización por VPN, característica de seguridad indispensable para cualquier servicio que se conecte a internet para administrar datos. Dicho producto tiene un costo, por lo tanto, muy superior a los anteriores. Tiene un precio de u\$s 4100, lo cual significa un aproximado de \$180000.



9 IMÁGENES REALIZACIÓN DEL PROYECTO







10 CONCLUSIÓN FINAL

- ✓ Concluimos entonces, que nuestro producto con un precio de venta de alrededor de entre \$10000 y \$20000, se podría colocar en el mercado como competencia directa al sector al que afecta. Tiene una relación precio/calidad/prestaciones muy acordes a los productos que se encuentran en el mercado.
- ✓ Entendemos también que con muy poco desarrollo, se podría ofrecer el producto de forma abierta y con la posibilidad de agregar add-ons a la placa de manera que a un precio un poco superior el cliente adapte fácilmente el producto a sus necesidades, con posibilidad de soporte online directo.
- ✓ El Gateway realizado en este proyecto, ha proveído la solución buscada de manera correcta y ha cumplido todos los objetivos que se plantearon.



11 APÉNDICE A: REFERENCIAS

1. GridTICs. Grupo de Investigación. <http://gridtics.frm.utn.edu.ar/site/>.
2. Proyecto CIAA. <http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp>.
3. GridTICs, Paper de WSN. https://www.researchgate.net/publication/266557776_Low-Power_Wireless_Sensor_Network_for_Frost_Monitoring_in_Agriculture_Research.
4. Proyecto EDU-CIAA, Ponchos. http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:ciaa:ponchos_existentes.
5. PHILLIPS, BIblioteca FAT-SD. <http://www.lpcware.com/content/project/LPCUSBlib>.
6. Ai-Thinker, User Manual ESP-01. https://wiki.ai-thinker.com/_media/esp8266/esp8266_series_modules_user_manual_v1.1.pdf.



12 APÉNDICE B: DICCIONARIO

La siguiente tabla proporciona definiciones de palabras relevantes para este documento.

Palabra	Definición
CIAA	Computadora Industrial Argentina Abierta
PCB	Printed Circuit Board (Placa de Circuito Impreso)
SPI	Serial Peripheral Interface (Interfase Periférica Seriada)
UART	Universal Asynchronous Receiver-Transmitter (Receptor-Transmisor Asíncrono Universal)
GridTICs	Grupo UTN de Investigación y Desarrollo en Tecnologías de la Información y Comunicaciones.
WSN	Wireless sensor networks (Redes de Sensores No Cableadas)
EDU-CIAA	CIAA dedicada para la educación