*GATEWAY DE COMUNICACIÓN PARA REDES DE SENSORES INALÁMBRICOS CON EL USO DE LA PLACA C.I.A.A.*

*(gateway ciaa)*proyecto FINAL

Version 1.0

*<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 nuestros más humildes agradecimientos a todos aquellos que han hecho posible la finalización de este trabajo.

Queremos agradecer principalmente a las personas más importantes que tenemos que son nuestros padres por ser los principales promotores de nuestros sueños, gracias a ellos por cada día confiar, creer, y querer el mejor futuro para nosotros.

Gracias a nuestras parejas por entendernos en todo, gracias a ellas porque en todo momento son un apoyo incondicional en nuestras vidas.

A nuestro 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 novia, mi siempre, mi todo. Sin ella este sueño nunca se hubiera hecho realidad. Vivió conmigo este camino y no dejó que dudara nunca 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.

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 |

Información del Proyecto

|  |  |
| --- | --- |
| Proyecto | **GATEWAY DE COMUNICACIÓN PARA REDES DE SENSORE INALAMBRICOS CON EL USO DE LA PLACA C.I.A.A.** |
| Empresa / Organización | **GRIDTICS** |
| 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 | **lucasescribano@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 6](#_Toc7283978)

[1.1 Executive Resume 6](#_Toc7283979)

[1.2 Resumen ejecutivo 6](#_Toc7283980)

[2 Descripción del Proyecto 6](#_Toc7283981)

[2.1 Objetivo general 6](#_Toc7283982)

[2.2 Objetivo particular 6](#_Toc7283983)

[3 Justificación del proyecto (Caso de Negocio) 7](#_Toc7283984)

[3.1 Antencedentes del proyecto 7](#_Toc7283985)

[3.2 Estado actual 7](#_Toc7283986)

[3.3 Necesidad del negocio y definición del problema 7](#_Toc7283987)

[3.4 Idea del proyecto 7](#_Toc7283988)

[3.5 Beneficios del proyecto 7](#_Toc7283989)

[3.6 Alcance 8](#_Toc7283990)

[3.7 Límites o fuera de alcance 8](#_Toc7283991)

[3.8 Requisitos de alto nivel 8](#_Toc7283992)

[3.9 soluciones y Entregables principales 9](#_Toc7283993)

[4 Planificación del proyecto 9](#_Toc7283994)

[4.1 Procesos 9](#_Toc7283995)

[4.2 Cronograma 10](#_Toc7283996)

[4.3 Hitos 11](#_Toc7283997)

[5 Desarrollo DEL PROYECTO 12](#_Toc7283998)

[5.1 Proyecto Edu-Ciaa 12](#_Toc7283999)

[5.1.1 ¿Qué es la CIAA? 12](#_Toc7284000)

[5.1.2 Origen del Proyecto CIAA 12](#_Toc7284001)

[5.1.3 Propósito 13](#_Toc7284002)

[5.1.4 Alcance 14](#_Toc7284003)

[5.1.5 Objetivos 14](#_Toc7284004)

[5.1.6 Requerimientos 14](#_Toc7284005)

[5.1.7 Definición de módulos a incluir en la plataforma 15](#_Toc7284006)

[5.1.8 Hardware de la EDU-CIAA 16](#_Toc7284007)

[*5.1.8.1* Bloques funcionales 16](#_Toc7284008)

[*5.1.8.2* Diagrama en bloques de la plataforma 17](#_Toc7284009)

[*5.1.8.3* *Módulos de la EDU-CIAA* 17](#_Toc7284010)

[5.2 Hardware 18](#_Toc7284011)

[5.3 Software 24](#_Toc7284012)

[5.3.1 Diagrama de Flujo de Funcionamiento 24](#_Toc7284013)

[5.3.2 Configuraciones Iniciales 25](#_Toc7284014)

[5.3.3 Programación en C 27](#_Toc7284015)

[*5.3.3.1* SPI 27](#_Toc7284016)

[6 Factibilidad Económica 35](#_Toc7284017)

[7 Aproximación al valor actual neto 35](#_Toc7284018)

[8 Tasa interna de retorno 35](#_Toc7284019)

[9 Payback o plazo de recuperación 35](#_Toc7284020)

[10 APÉNDICE A: REFERENCIAS 37](#_Toc7284021)

[11 APéNDICE B: DICCIONARIO 38](#_Toc7284022)

# IntroDucción

## 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.

## 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.

# Descripción del Proyecto

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

El tiempo estimado del diseño del prototipo es de 12 meses con un costo aproximado de $50000.

## Objetivo general

Se desea encontrar solución a las redes de sensores, capturando la información que estos sensores proveen y colocarlas en un servidor web para su posterior análisis, a través de sistemas y protocolos de comunicación estándares.

## Objetivo particular

Como actividad 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.

# Justificación del proyecto (Caso de Negocio)

## 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.

## Estado actual

Actualmente los dispositivos de puerta de enlaces, tiene una demanda mayor 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.)

Estos tienen un costo muy elevado. (Desde $90.000 aprox. para la compra).

## Necesidad del negocio y definición del problema

Hoy en día, en el mercado estos equipos tienen un costo muy excelso 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 Gridtics. 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.

## 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 reestablecerá 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.

## Beneficios del proyecto

* Seguro.
* Confiable.
* Diseño a medida para cualquier protocolo propietario.
* Fácil Mantenimiento.
* Configuración simple.
* Industrial.
* Bajo costo.
* Excelente relación precio/calidad.
* Producto con proyección internacional.

## Alcance

El proyecto producirá:

* Seguridad y confiabilidad en el tratamiento de datos.
* Fácil configuración.
* Posibilidad de alimentación a través de paneles solares.
* Fácil operación.
* Alta Disponibilidad (con paneles solares)

## Límites o fuera de alcance

Nuestro dispositivo se limitará solo vinculará el protocolo propietario diseñado por el grupo de investigación Gridtics.

El equipo al encender, desde la tarjeta SD, leerá solo los parámetros descriptos el archivo tipo txt, para conectarse a la red inalámbrica detallada en la configuración.

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

El Gateway, dentro de la SD, almacenará un log de los distintos procesos u operaciones que va realizando el mismo. Este log tiene un formato tipo txt.

## Requisitos de alto nivel

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

| Req. # | I Descripción del requisito |
| --- | --- |
| 1 | Permitir la comunicación entre protocolos (propietarios y estandarizados). |
| 2 | Transmisión Full-duplex. |
| 3 | En caso de interrupción de comunicación, almacenar datos en SD. |
| 4 | Debe ser fácil de operación e intuitivo. |

## 

## 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 conexionados, 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 Conexionado |  |
| Cable de alimentación | Cable micro USB-B 2.0 |

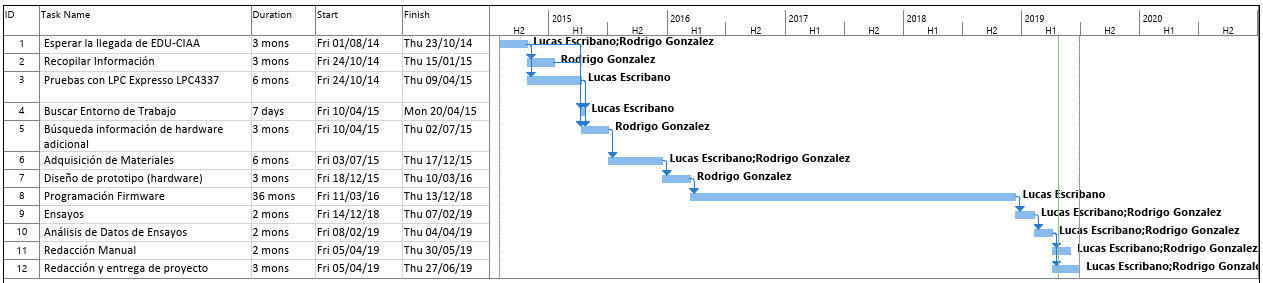
# Planificación del proyecto

## 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.

## Cronograma



## Hitos

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

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

# Desarrollo DEL PROYECTO

## Proyecto Edu-Ciaa

### ¿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.

### 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](http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:hardware:ciaa_nxp:ciaa_nxp_inicio) y otras seis versiones están en elaboración: CIAA-ATMEL, [CIAA-FSL](http://www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:hardware:ciaa_freescale:ciaa_freescale_inicio), 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](https://groups.google.com/group/embebidos32/subscribe)) 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.

### 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.

### 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.

### 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.

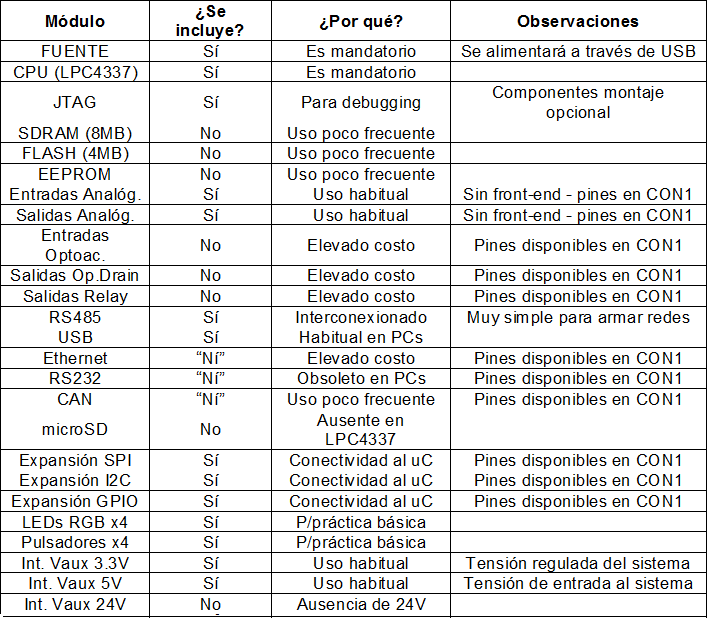
### Requerimientos

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.

### 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:



### Hardware de la EDU-CIAA



#### Bloques funcionales

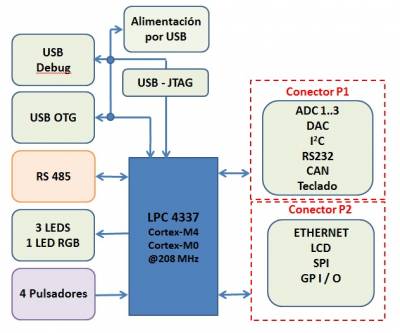
La EDU-CIAA está basada en la CIAA-NXP, por ser la primera versión de la CIAA que se encuentra disponible. Por lo tanto su microcontrolador es también el LPC4337 (dual core ARM Cortex-M4F y Cortex-M0).

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.

#### Diagrama en bloques de la plataforma

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

[](http://www.proyecto-ciaa.com.ar/devwiki/lib/exe/detail.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp&media=versiones:educiaav1_0:hardware:edu_ciaa_nxp:diagrama_en_bloques.jpg)

#### *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 3×4,
  + 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.

## 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.

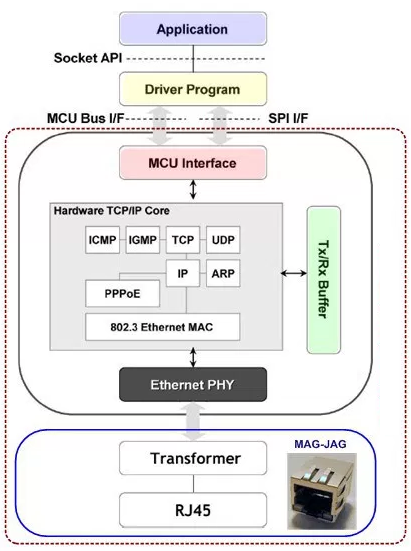
Una vez leídos los datasheet de los componentes, se procede entonces a adquirir 3 módulos externos:

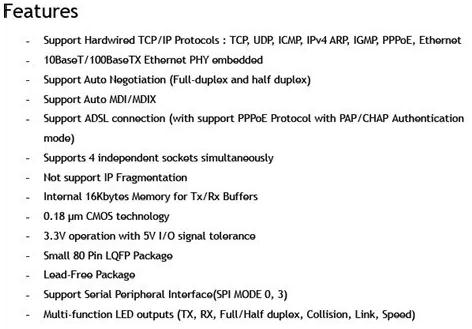
**-Ethernet (SPI)**

|  |  |
| --- | --- |
| Resultado de imagen para mÃ³dulo ethernet arduino | 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.





**-WiFi (SPI-UART)**

|  |  |
| --- | --- |
| Imagen relacionada | 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.AA WiFi 2.4Ghz, WEP/WPA/WPA2. * Up to 16 GPIO Pins, SPI, I2C, I2S, UART. * 10 Bit ADC. |
|  |  |

**-SD (SPI-UART)**

|  |  |
| --- | --- |
| Resultado de imagen para features shield sd arduino | *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 a se obtuvo una pantalla LCD de 16x2 caracteres.

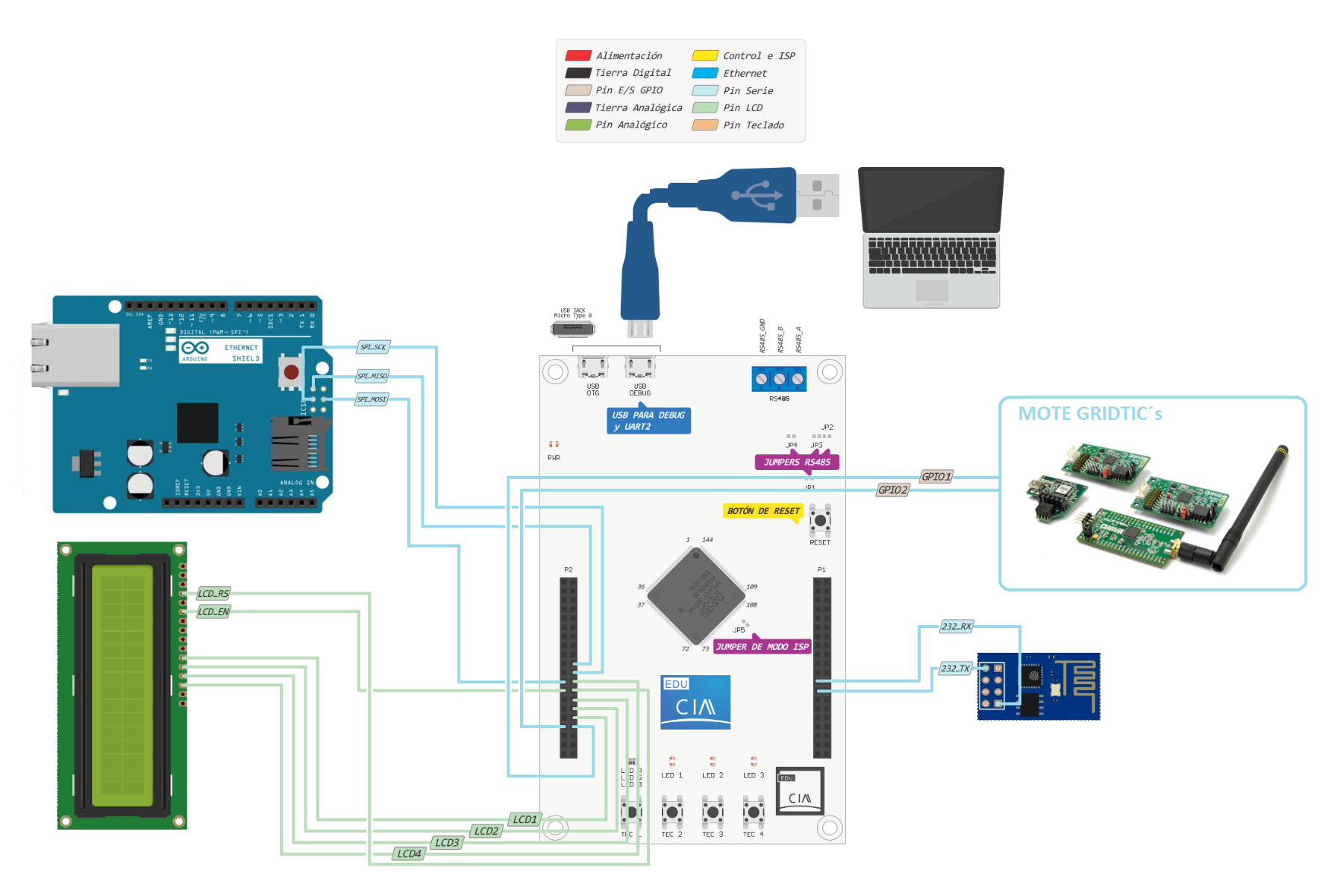


Se definieron las posibilidades de conexión de los artefactos electrónicos en la EDU-CIAA.

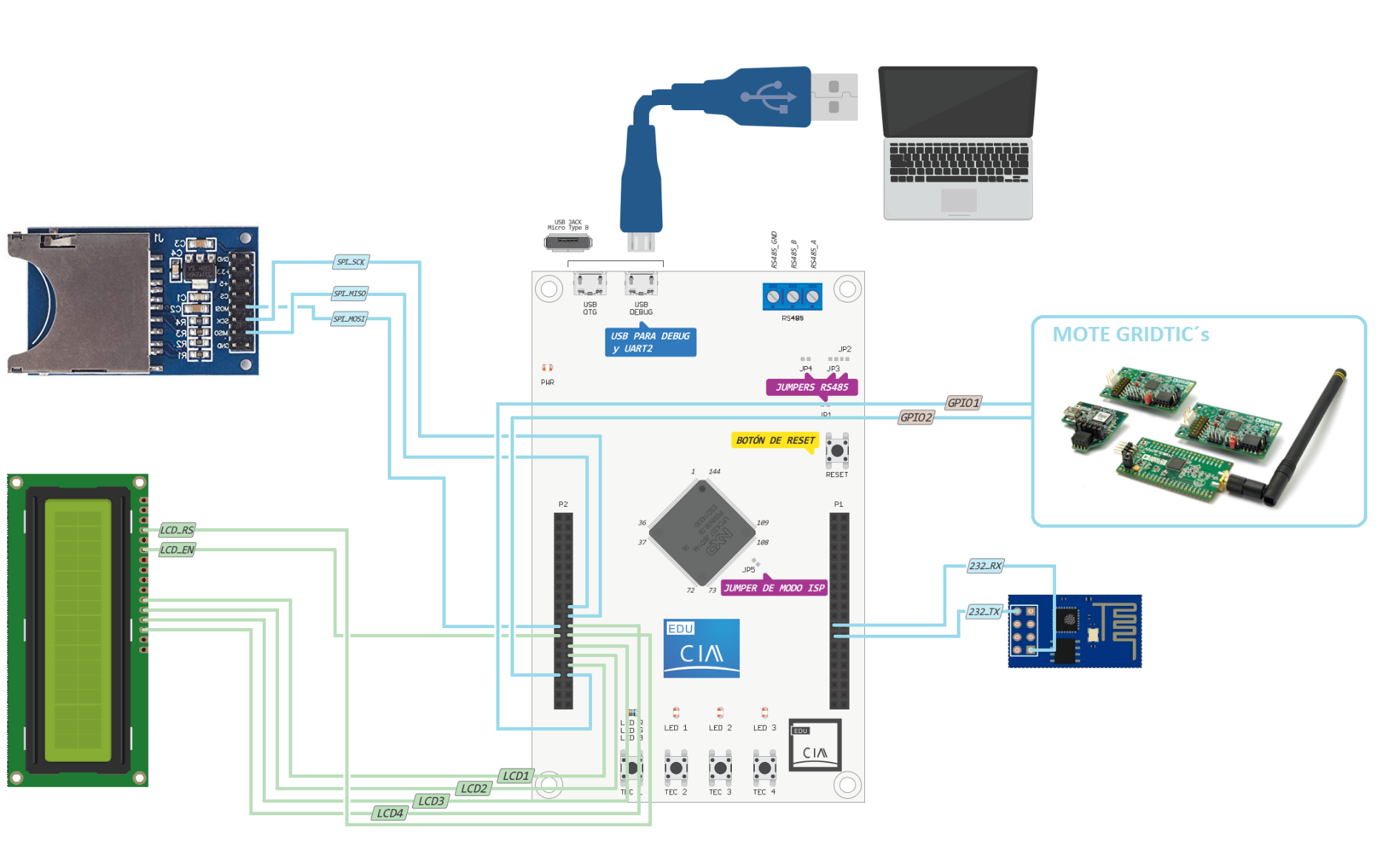
Se tuvieron en cuenta los siguientes puntos:

1. La EDU-CIAA tiene 3 UART´s.
   1. 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.
   2. 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.
   3. El “UART\_485”. 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*:

****

## Software

### 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:



### Configuraciones Iniciales

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

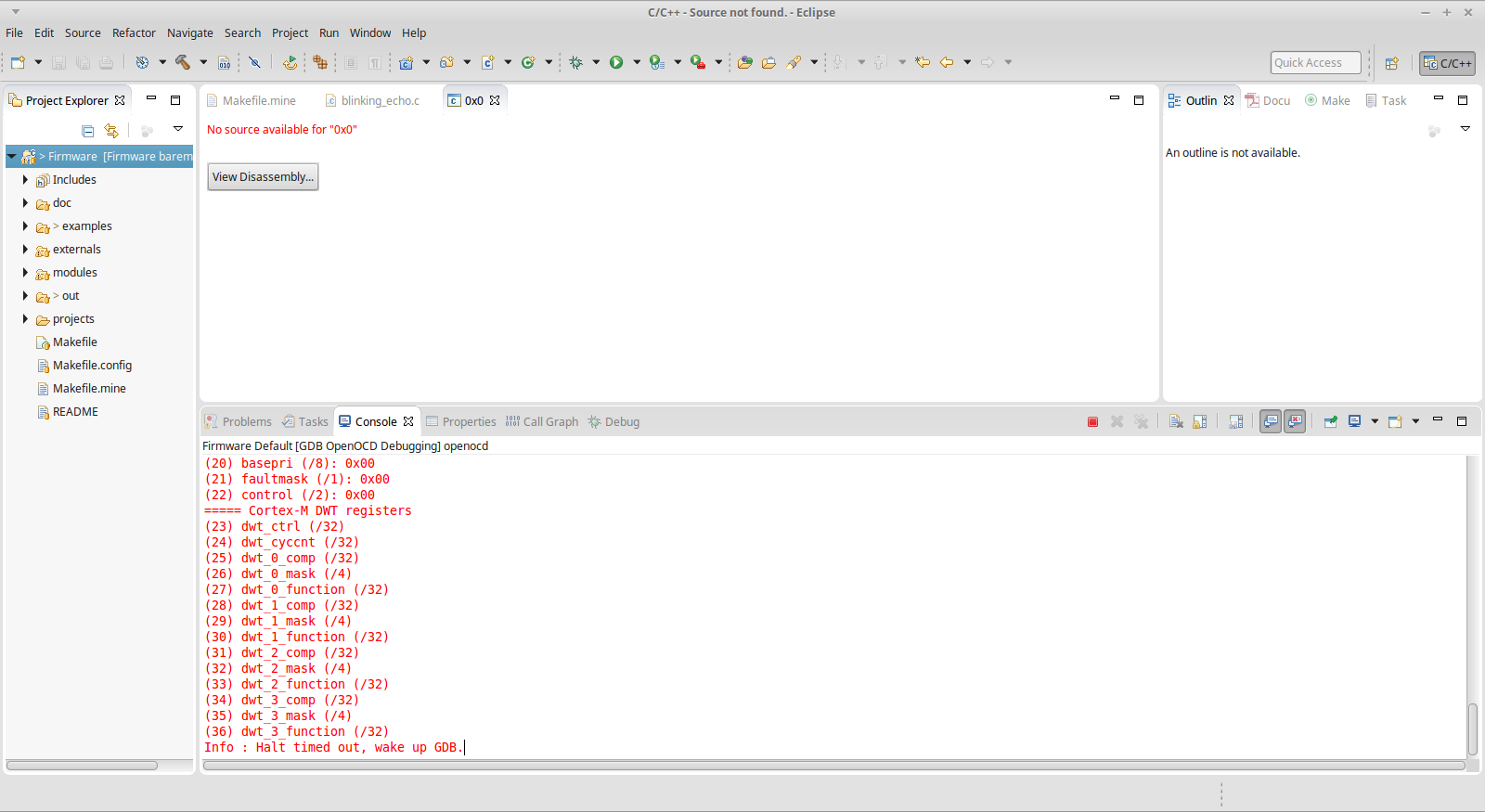
A simple vista se puede visualizar los distintos lenguajes de programación con los cuales se puede realizar el mismo:

* 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 elije Lenguaje C por poseer mayores conocimientos en él.

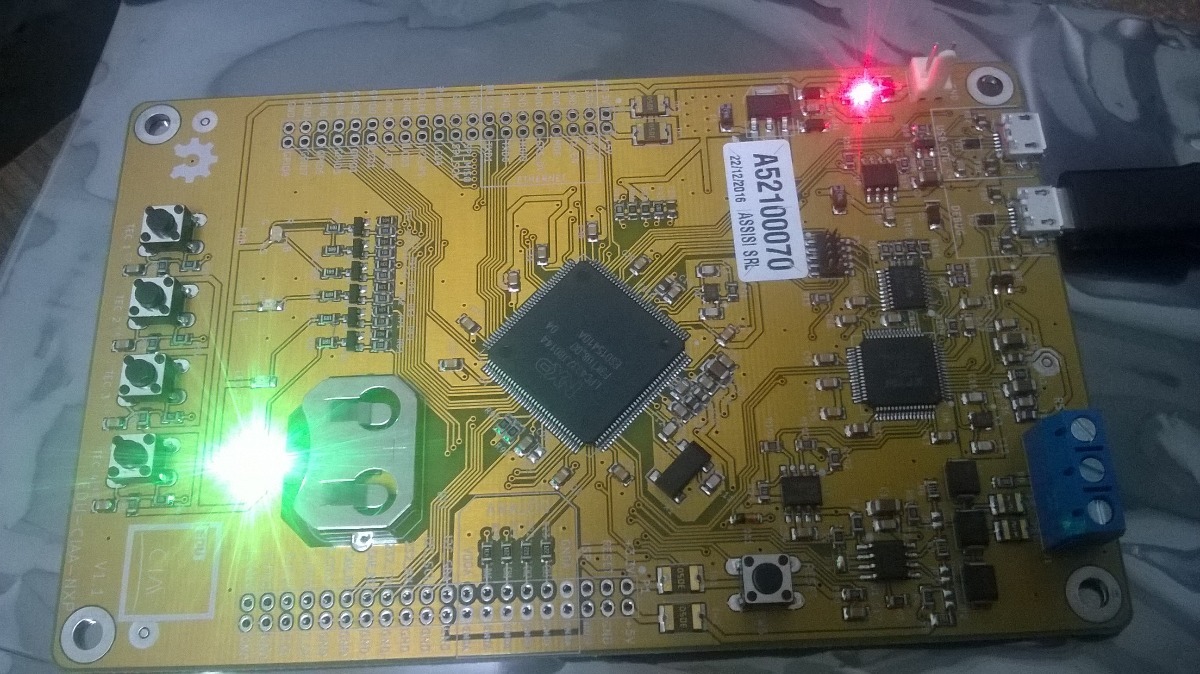
Luego se eligen las sugerencias de la página para dejar instalado el entorno de programación:

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



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

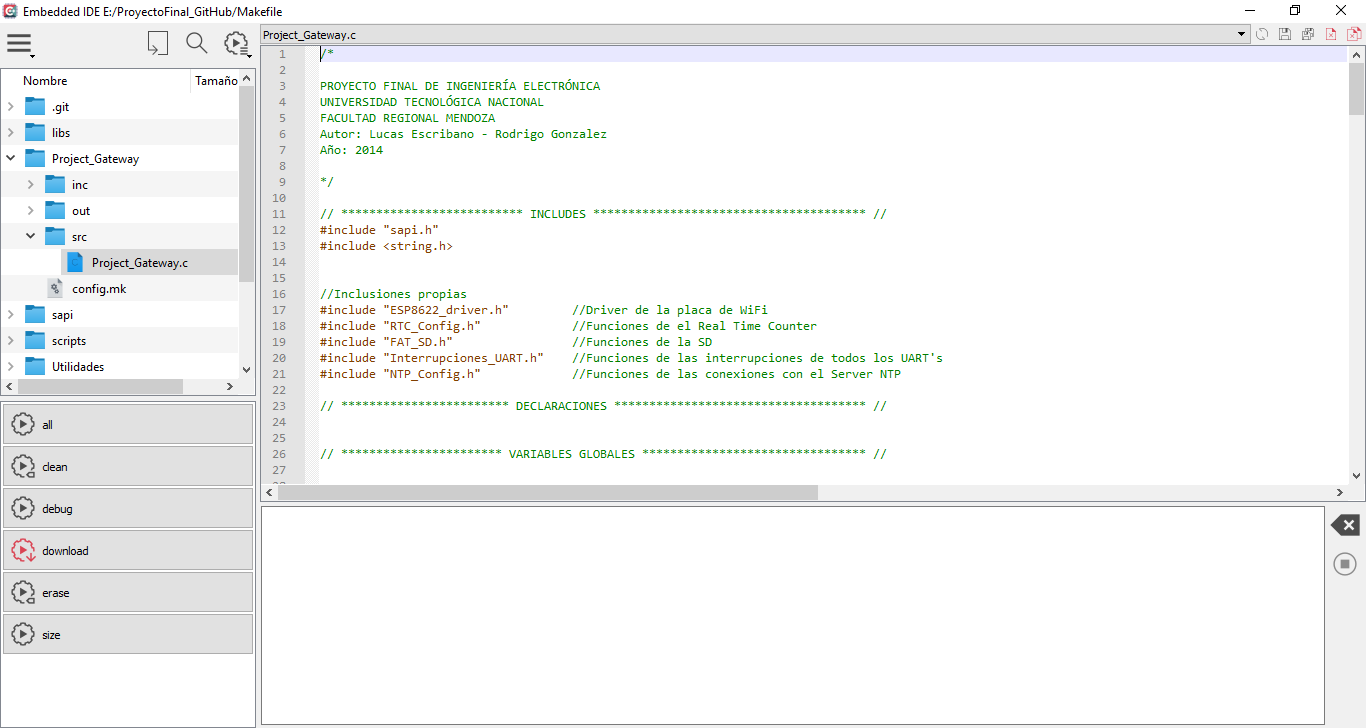
Una vez realizado lo anterior se procede a cargar el ejemplo “blinking” de la librería examples existente.



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.



### Programación en C

Luego de realizar varias pruebas con la EDU-CIAA, podemos apreciar a simple vista que el Driver para la utilización del puerto SPI del microprocesador no estaba realizado. Se encuentra un ejemplo muy pobre de lo mencionado.

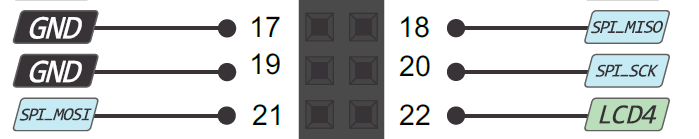
Para utilizar el UART hay una versión “beta” de un driver.

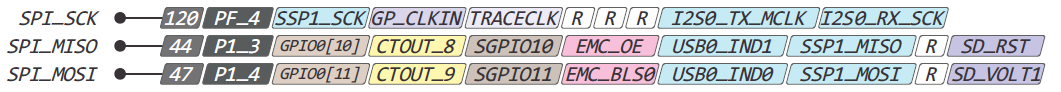
Nos disponemos a estudiar las librerías de la EDU-CIAA, para poder, como primera instancia, realizar un driver para el SPI. Este puerto de comunicaciones nos servirá de base para poder grabar y leer tanto la placa ethernet como la placa de la SD.

#### SPI (ETHERNET)

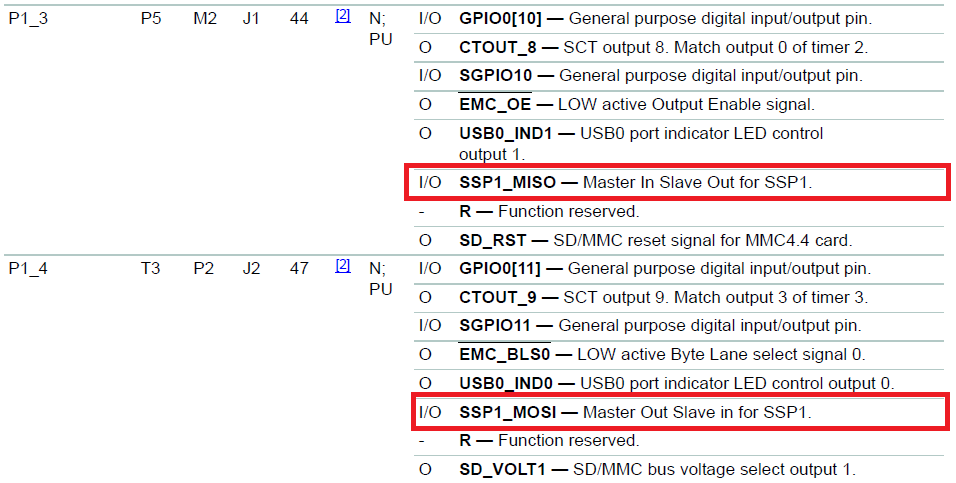
Con ayuda del datasheet del micro LPC4337, podemos realizar pruebas con el SPI.

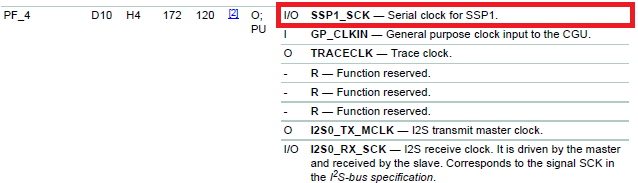
Antes que nada hay que encontrar los pines en la EDU-CIAA que se utilizarán para tales propósitos, con la información citada en anexos[[1]](#footnote-1) podemos verificar los pines de salida a utilizar:





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:





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 sitaxis 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 preseteó una funció 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 \*/*

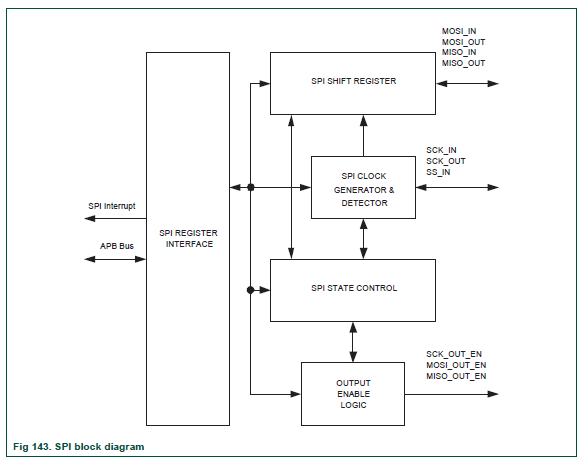
*}*

*else {*

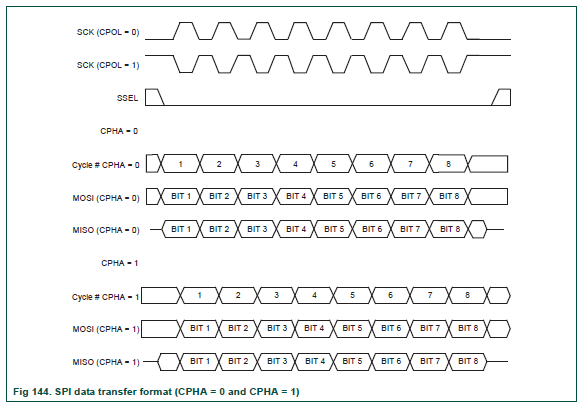
*return;*

*}*

*}*



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



Los pasos a seguir son proporcionados por el fabricante, y depende si será utilizado como maestro o como esclavo. Dicha información se encuentra en el anexo.[[2]](#footnote-2)

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

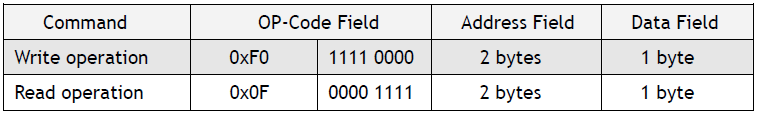
***Commands:***

*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.*

**

*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*

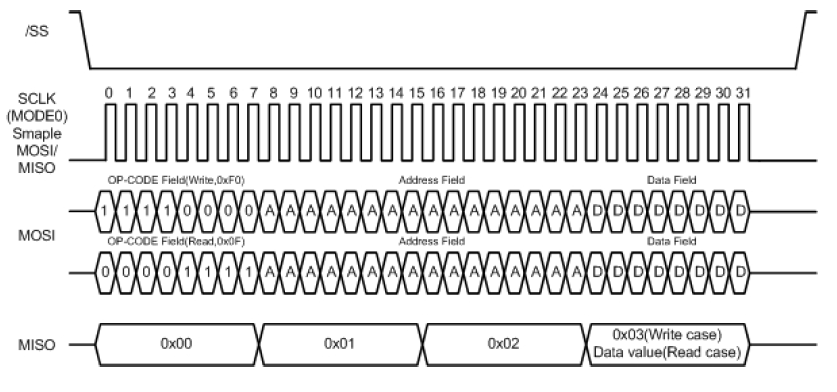
*\* SPI data rate bit on SPCR register and SPSR register (SPI State Register)*

*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’*

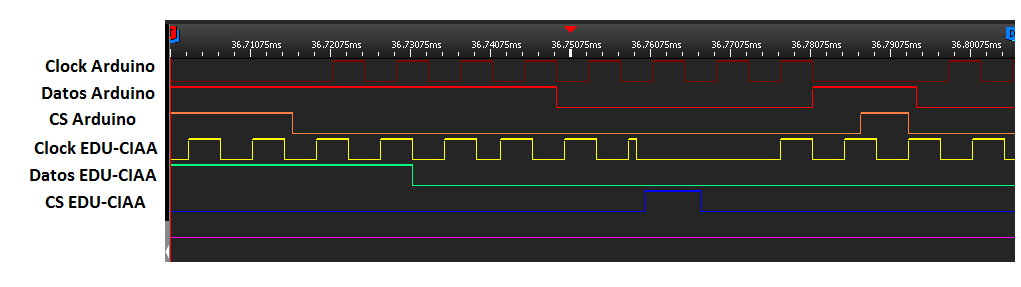


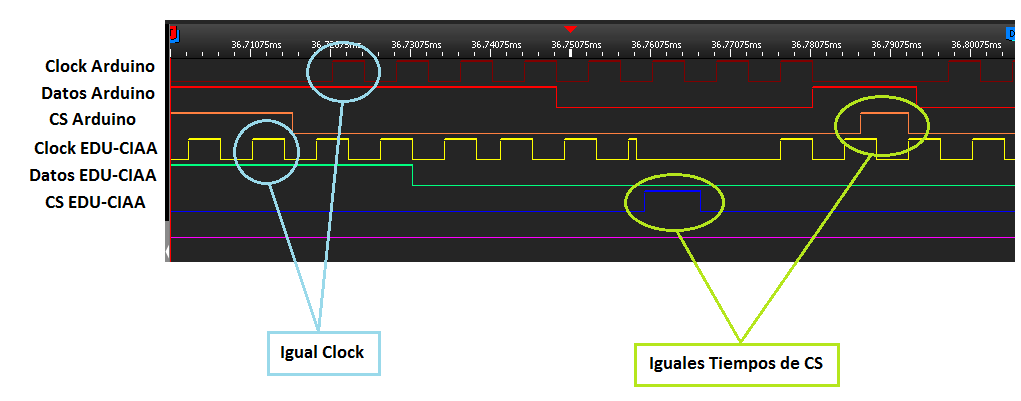
Dichos pasos fueron correctamente realizados en el programa.

Al probar la placa ethernet, nos dimos cuenta que la misma no toma las configuraciones que le enviamos mediante comandos por el puerto SPI.

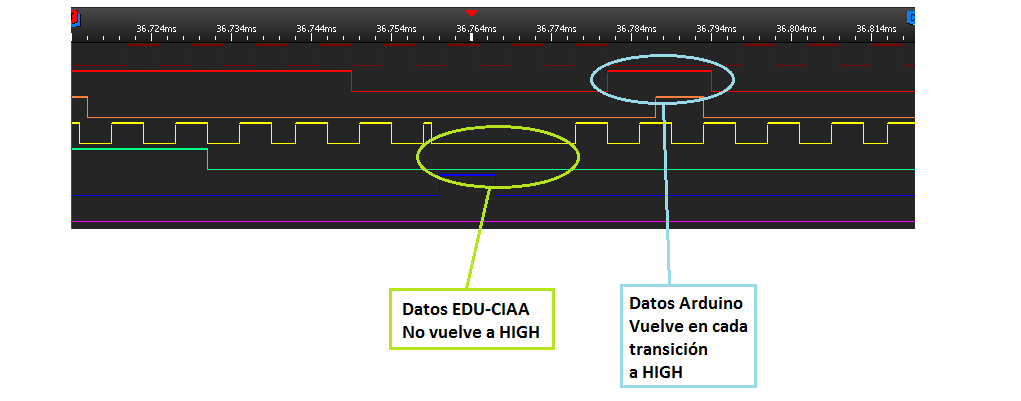
Es por ello que la placa se somete a un TEST con equipamiento de adquisición de datos que el departamento GridTic´s nos ofrece.

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:

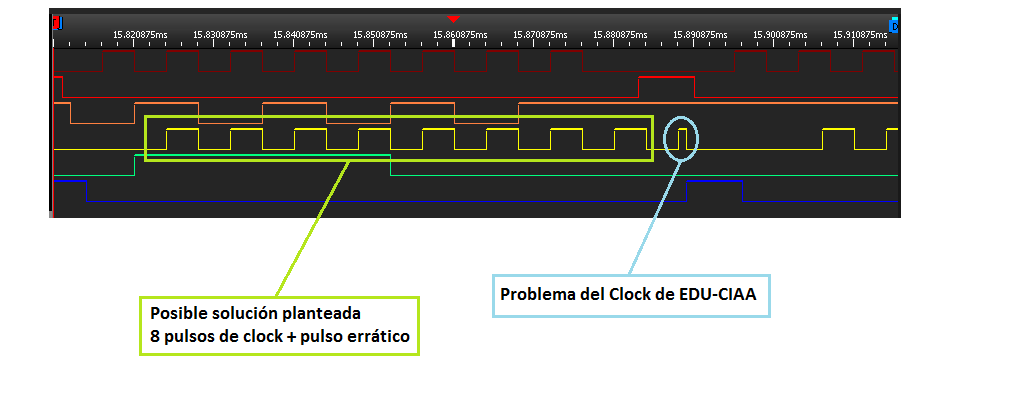




Se observa que los tiempos son exactamente iguales, tanto los clocks como los chip select (CS).



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 ETHERNE**T.

#### 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, la provee PHILLIPS

http://www.lpcware.com/content/project/LPCUSBlib

# 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 peso = 44,96 dólares



# Aproximación al valor actual neto

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.



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

# Tasa interna de retorno

Se puede ver que el TIR es de 28,13%

# Payback o plazo de recuperación

Podemos ver que nuestro plazo de recuperación es de 8 años.

# 

# APÉNDICE A: REFERENCIAS

La siguiente tabla contiene los documentos a los que se hace referencia en este documento

|  |  |  |
| --- | --- | --- |
| **Nombre del documento y versión** | **Descripción** | **Ubicación** |
|  |  | http://www.lpcware.com/content/project/LPCUSBlib |
|  |  |  |
|  |  |  |

# APéNDICE B: DICCIONARIO

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

|  |  |
| --- | --- |
| **Palabra** | **Definición** |
| *Nombre corto del proyecto (sigla)* | *Nombre completo del proyecto* |
|  |  |
|  |  |

1. ANEXO – INFORMACIÓN LPC 4337 – EDU-CIAA PIN OUT [↑](#footnote-ref-1)
2. ANEXO – INFORMACIÓN LPC 4337 - SPI - PASOS DE FUNCIONAMIENTO [↑](#footnote-ref-2)