

Informe final
Monitoreo de nivel de agua en depósitos
mediante tecnologías IoT



**UNIVERSIDAD
NACIONAL
DE LA PLATA**

Taller de Proyecto I - Ingeniería en computación

Alumnos	N° de legajo
Dehan Lucas	565/1
Duarte Victor	1055/7



Departamento de electrónica - Taller de proyecto I

ÍNDICE

Introducción	1
Objetivos del proyecto	2
Objetivo Principal	2
Objetivo Secundario	3
Análisis de requerimientos	4
Requerimientos funcionales	4
Requerimientos no funcionales	4
Diseño del hardware	5
Pulsador	6
Sensor Ultrasonido HC-SR04	7
RTC DS3231	7
ESP-01	9
LED	10
Alimentación del sistema	10
Diseño del PCB	12
Vías	12
Pads	13
Reglas del diseño	13
Conectores	14
Pulsador	15
Sensor Ultrasonido HC-SR04	15
RTC DS3231	16
ESP-01	16
LED	17
Diseño del software	19
Pulsador	19
hc_sr04	20
build_defs	20
rtc	20
esp01	21



Departamento de electrónica - Taller de proyecto I

main	22
Ensayos y mediciones	23
Prototipo conectado a la EDU-CIAA	23
Dificultades en proceso de diseño	26
Conclusiones	27
Tiempo dedicado y Presupuesto final	28
Bibliografía	31
Anexos	33
Esquemático.	33
Cómo funciona el sensor ultrasónico.	33
Cómo funciona un RTC	34
Thingspeak	35
PCB	37
Cronograma final	40
Lista de Materiales	41
Agradecimientos	42



Departamento de electrónica - Taller de proyecto I

Introducción

En ámbitos ganaderos existe la necesidad de ahorrar tiempo a la hora de verificar si los bebederos de los animales poseen agua y si su nivel es adecuado. Hoy en día esto lo realiza un empleado recorriendo grandes distancias entre cada bebedero, con el fin de saber si los mismos están en un estado de llenado u ocurre algún problema en dicho proceso.

Haciendo uso de los avances de las tecnologías aplicadas a IoT o Internet de las Cosas se busca producir dos productos: Servidor y Clientes. El Servidor estará ubicado en la estancia central, será el encargado de recibir toda la información recolectada mediante una conexión especializada de bajo consumo de energía para transmisión a largas distancias y enviarla a la nube mediante conexión wifi con el fin de que el usuario pueda visualizar toda la información en un dispositivo electrónico.

En cuanto al Cliente, estará ubicado en el bebedero y será el encargado de recolectar la siguiente información: nivel de agua e informando el estado del flotante de carga que posee el bebedero con el fin de saber si el bebedero se encuentra cargando cuando el nivel de agua no es máximo. Dicha información se enviará al Servidor.

También se desarrollará una aplicación web con una interfaz gráfica para que el usuario sea capaz de visualizar el estado actual de los bebederos utilizando la información recolectada por el Servidor.

Para la materia “Taller de Proyecto I”, se fabricará un prototipo del Cliente y la comunicación hacia una plataforma de análisis de datos para IoT. Las mediciones serán realizadas sobre un recipiente con agua el cual será una representación a escala menor de un bebedero.

Para dicho proyecto se requiere de un sensor de nivel de agua, un pulsador para conocer el estado del flotante, también se requerirá un reloj para informar la fecha de ocurrencia de cada sensado. Se dispondrá de una conexión inalámbrica para enviar la información a la nube para que el usuario pueda visualizar la información a través de una aplicación web.



Departamento de electrónica - Taller de proyecto I

Objetivos del proyecto

Objetivo Principal

Se realizará un prototipo a escala como prueba de concepto que sea verificable en el ámbito del aula con la placa EDU-CIAA de un sistema capaz de realizar distintas tareas como:

1. Medir carga completa del tanque con un pulsador activado mediante un flotante de carga.
2. Medir el nivel de agua a distintas alturas con un sensor de ultrasonido.
3. Para la comunicación inalámbrica, se usará un módulo wifi para conectar la placa EDU-CIAA a un servicio de IoT. En este proyecto se usará el servicio ThingSpeak con el protocolo MQTT para enviar los sensores.
4. Para el reloj, se hará uso de un RTC¹ conectado a la EDU-CIAA con el cual se podrá informar al usuario la fecha y hora a la que se producen los sensores.
5. Medir el nivel de batería del dispositivo para informar al usuario cuando es necesario realizar un cambio de batería.
6. La aplicación web será una interfaz de usuario proporcionada por ThingSpeak, la cual mediante gráficos y mensajes de texto informará al usuario del estado del recipiente.

Con estos objetivos en mente podemos hacer un esquema con el EDU-CIAA conectado a los siguientes componentes:

- Un pulsador para detectar la señal del flotante de carga.
- Un sensor de ultrasonido HC-SR04 para detectar el nivel de agua.
- Un RTC (Modelo específico a definir) como reloj de sistema.
- Modulo ESP-01 para la conexión wifi.

¹ RTC: Real Time Clock o Reloj de Tiempo Real. Es un reloj de computadora, generalmente incluido en un circuito integrado, cuyo objetivo es mantener la hora del sistema.



Departamento de electrónica - Taller de proyecto I

En la siguiente imagen (Ver Figura 1) se observa una representación del esquema mencionado anteriormente.

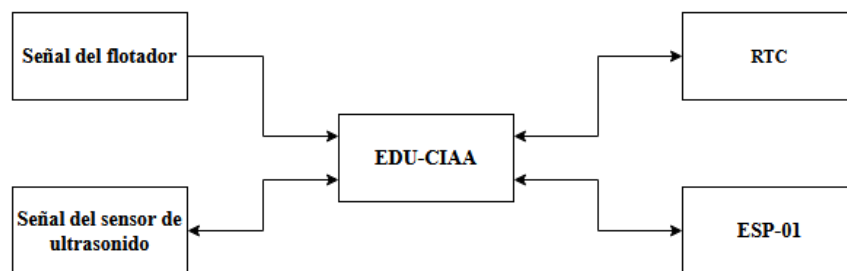


Figura 1: Diagrama en bloques del sistema.

Como se puede observar en la imagen, a la EDU-CIAA, del lado izquierdo superior podemos ver mediante una conexión unidireccional se conecta el pulsador que emitirá la señal del flotador.

Las demás conexiones serán bidireccionales esto quiere decir que recibirán una señal desde la EDU-CIAA y responderán a ella con la información pedida. Las conexiones son con:

- Sensor de ultrasonido (HC-SR04: naylampmechatronics, 2022)
- ESP-01 con soporte para protocolos SPI, UART, I2C. (naylampmechatronics, 2022)
- RTC con un protocolo I2C

Objetivo Secundario

Con vista al futuro la idea es poder implementar el Servidor y crear los Clientes con una conexión de mayor alcance (LoRa) respecto a la conexión wifi para conectarlo al Servidor el cual se conectará a la nube mediante wifi.

Para mejorar el rendimiento energético, dado que la EDU-CIAA consume más de lo necesario para este proyecto, en el futuro se podría reemplazar la EDU-CIAA por otra placa con los componentes necesarios para un funcionamiento con menor consumo.

Para no depender de servicios de terceros, se podría incorporar la aplicación web en un servidor propio con la capacidad para que usuarios puedan registrarse e incorporar todos sus dispositivos en su cuenta, para un mejor seguimiento de los mismos.



Departamento de electrónica - Taller de proyecto I

Análisis de requerimientos

Requerimientos funcionales

- Registrar la fecha y hora de cada sensado.
- Sensado de nivel máximo de agua en el tanque por medio de un pulsador conectado al flotante.
- Sensado de nivel de agua en el recipiente por medio de un sensor de ultrasonido.
- Sensar el nivel de batería del dispositivo para detectar el estado de la batería (Detalles a confirmar).
- Capacidad de conectarse a una red wifi existente a la plataforma ThingSpeak para enviar los datos sensados.

Requerimientos no funcionales

- Utilización de la placa EDU-CIAA.
- Utilización de la placa ESP-01 para la transmisión Wifi.
- Utilización del protocolo MQTT para un menor consumo de energía y ancho de banda.
- Programación de software en lenguaje C mediante las herramientas proporcionadas por el Proyecto CIAA.
- Desarrollo del PCB con los circuitos necesarios para conectar los sensores.
- Realizar el prototipo final para su presentación acorde a la fecha límite prevista por el cronograma, en este caso es para febrero de 2023.

Departamento de electrónica - Taller de proyecto I

Diseño del hardware

En base al diagrama en bloques explicado en la sección “Objetivos del proyecto” se diseñó un Esquemático en el programa Proteus.

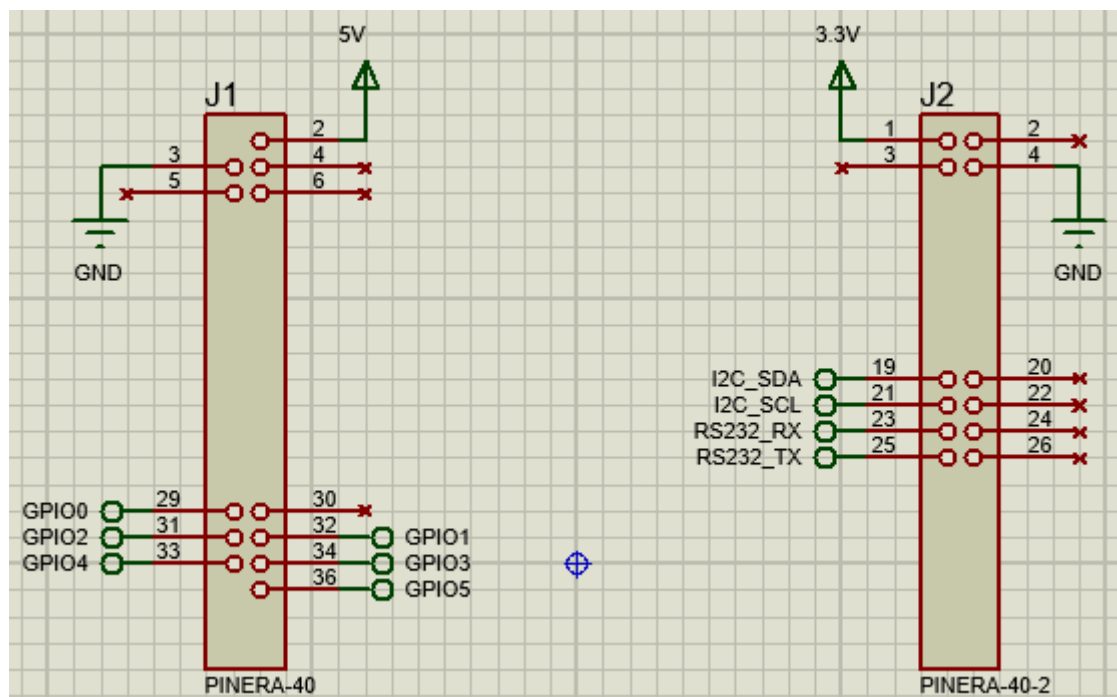


Figura 2: Diagrama de los conectores.

La imagen anterior es una representación de los conectores (J1 y J2) que irán encastrados en los conectores de la EDU-CIAA en Proteus. Cada conector contiene pines que pueden conectarse a otros dispositivos como sensores, pulsadores, LEDs, etc. Aquellos pines que serán utilizados para una comunicación con otros módulos están representados con un símbolo en color verde, junto con el nombre del pin en la EDU-CIAA y aquellos que no serán utilizados en este proyecto serán eliminados de la pinera para ganar espacio, para que pasen pistas o serán marcados con otro símbolo que indica que no está conectado a ningún otro lado pero se deja el agujero para usarse como sujeción mecánica del pcb a la placa. (ver bibliografía EDU-CIAA-NXP).



Departamento de electrónica - Taller de proyecto I

A continuación se describen los distintos componentes del hardware y sus conexiones correspondientes:

Pulsador

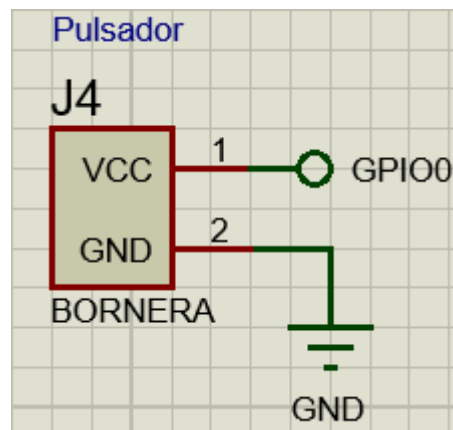


Figura 3: Diagrama de la bornera conectada al pulsador.

La comunicación entre el Pulsador y la EDU-CIAA se realizará por el pin GPIO0. Este pulsador hará uso de la resistencia de pull up interna del EDU-CIAA en el pin. La resistencia de pull up funciona de la siguiente manera: Cuando el pulsador no está presionado la tensión presente en el pin es de nivel alto (3.3V) y en el momento cuando se presiona el pulsador, se deriva toda la corriente a tierra (GND) ya que en ese momento es el camino de menor resistencia por lo que la tensión en el pin pasará a ser de nivel bajo (0V).



Departamento de electrónica - Taller de proyecto I

Sensor Ultrasonido HC-SR04

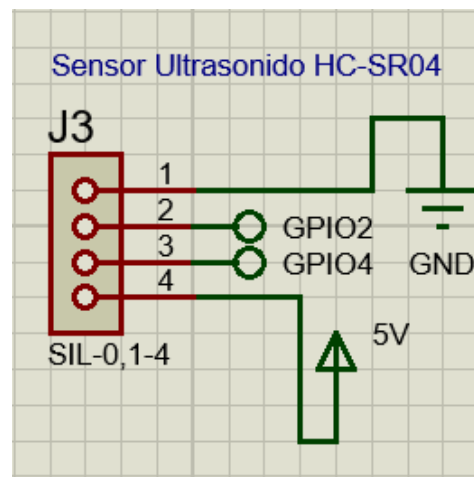


Figura 4: Diagrama del sensor ultrasónico.

En este proyecto, el sensor será alimentado por la EDU-CIAA con una tensión de 5V. Para emitir el sonido ultrasónico, se hará una conexión entre el pin GPIO4 del EDU-CIAA al pin Trigger del sensor para controlar cuándo realizar el sensado. La respuesta del sensor es enviado por el pin Echo, que estará conectado al pin GPIO2 del EDU-CIAA.

(Ver anexo pág. 33)

RTC DS3231

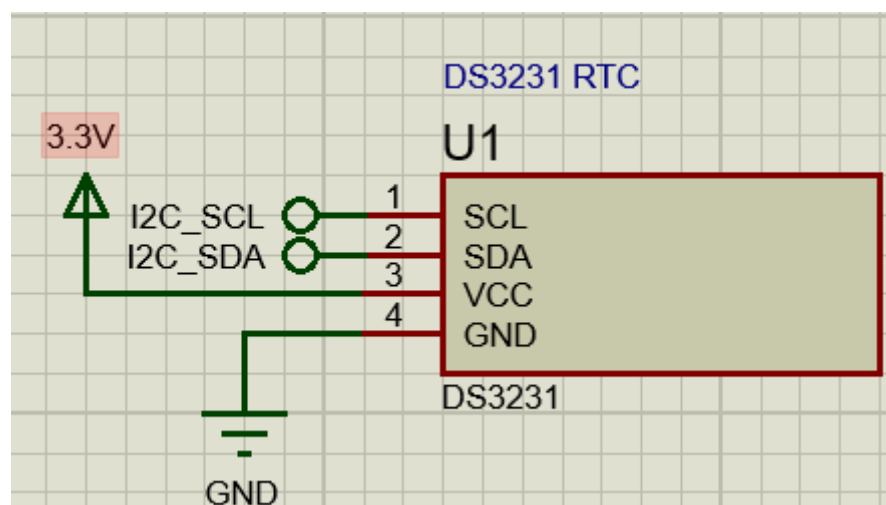


Figura 5: Diagrama del RTC.



Departamento de electrónica - Taller de proyecto I

El RTC seleccionado es el DS3231 que posee las características siguientes:

- Voltaje de alimentación de 3.0 a 5 volts.
- RTC de alta exactitud, maneja todas las funciones para el mantenimiento de fecha/hora.
- Exactitud de $\pm 2\text{ppm}$ operando a una temperatura de 0°C a $+40^{\circ}\text{C}$.
- Módulo cuenta con reloj DS3231 y memoria EEPROM I2C.
- Registro de segundos, minutos, horas, día de la semana, fecha, mes y año con compensación de años bisiestos hasta 2100.

Para conectar un DS3231 con la EDU-CIAA se requieren de 4 conexiones, una opcional para la señal de interrupción y otra opcional para una salida de referencia que puede ser utilizada para medir la precisión del reloj o para sincronizar otros dispositivos.

Los pines del DS3231 son los siguientes:

1. SCL – Señal de reloj del bus I2C.
2. SDA – Señal de datos del bus I2C.
3. VCC – Alimentación de 3.3V del módulo.
4. GND – Tierra o común.



Departamento de electrónica - Taller de proyecto I

ESP-01

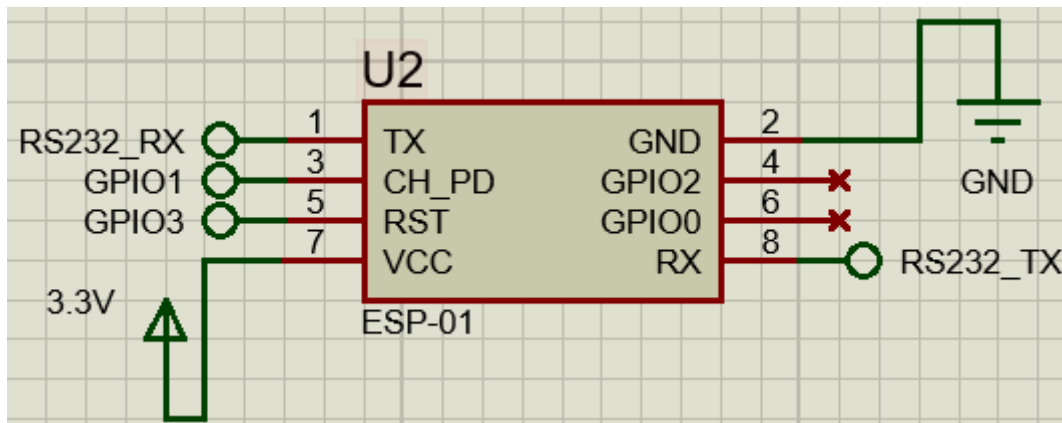


Figura 6: Diagrama del ESP-01.

El módulo ESP-01 está basado en el SoC(System on Chip) ESP8266. Es alimentado por la EDU-CIAA a 3.3V. El mismo provee de comunicación WiFi al sistema.

El ESP-01 tiene 8 pines:

1. TXD: Pin por donde se van a transmitir los datos del puerto serie.
2. GND: Tierra.
3. CH_PD: Pin para apagar y encender el ESP-01. Si lo ponemos a 0 V (Nivel bajo) se apaga, y a 3,3 V (Nivel alto) se enciende.
4. GPIO2: Pin de entrada-salida de propósito general.
5. RST: Pin para reiniciar el ESP-01. Si lo ponemos a 0 V, se reinicia.
6. GPIO0: Pin de entrada-salida de propósito general.
7. VCC: Pin de alimentación del ESP-01. Funciona a 3,3 V y admite un máximo de 3,6 V.
8. RXD: Pin por donde se van a recibir los datos del puerto serie.

NOTA: La comunicación con este módulo se realiza mediante protocolo UART, por lo tanto los terminales TXD y RXD del módulo se conectan a los terminales RXD_232 y TXD_232 respectivamente.



Departamento de electrónica - Taller de proyecto I

LED

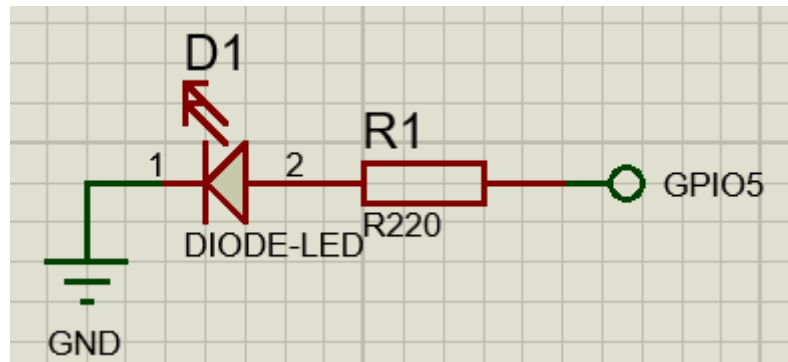


Figura 7: Diagrama de los conectores.

Se hará uso de un LED para indicar el estado de la conexión a Internet. El LED se encenderá para indicar que la EDU-CIAA está conectada a Internet, y se apagará cuando no haya conexión.

El LED estará conectado al pin GPIO5 con una resistencia de 440Ω con las siguientes condiciones: la corriente que pasará por el LED será de aproximadamente de 10 mA y la tensión entre los bornes del LED será de aproximadamente 2.2V.

Alimentación del sistema

Las formas para alimentar la EDU-CIAA son las siguientes:

- Una fuente de alimentación externa a 5V.
- Puerto USB dedicado para la alimentación.

La alimentación por medio del puerto USB otorga a la placa una tensión de 3.3V y otra tensión de 5V, pero tiene ciertas restricciones: la corriente máxima otorgada por este medio son de 500 mA, y además hay un fusible en ambos conectores tanto para 3.3V y 5V. Por lo tanto hay cuatro fusibles en total y el máximo de corriente que puede tolerar cada fusible es de 300 mA.

Para asegurarnos que sea posible alimentar el sistema por el puerto USB, es necesario realizar un cálculo aproximado de la corriente necesaria para la alimentación.



Departamento de electrónica - Taller de proyecto I

En conector J1:

Dispositivo	Consumo de corriente
Sensor Ultrasonido HC-SR04	15 mA
Pulsador	0.5 mA
LED	10 mA
Total J1	75 mA

En conector J2:

Conector	Consumo de corriente
RTC DS3231	200 uA
Modulo ESP-01	226 mA (Puede tener picos mayores)
Total J2	226.2 mA

Total de consumo:

Conector	Consumo de corriente
J1	75 mA
J2	226.2 mA
Total	301.2 mA



Departamento de electrónica - Taller de proyecto I

Diseño del PCB

En base a lo explicado en la sección del esquemático, se procede a realizar el PCB. Los componentes de PCB en Proteus se miden en milímetros (mm), pulgadas (in) y mils o thou (th). Para evitar confusiones en las medidas que se mostrarán más adelante, a continuación se presenta una tabla con las unidades de medida del sistema anglosajón de unidades y su equivalencia a milímetros.

Tabla 1. Unidades de medida anglosajona y su valor en milímetros.

Unidad	valor en mm
1 in	25,4 mm
1 th	0,0254 mm

Para este proyecto se diseñaron componentes de PCB:

Vías

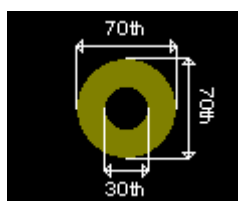


Figura 8A: Diagrama la vía “V70X30”

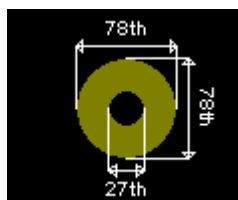


Figura 8B: Diagrama la vía “V78X27”

Las vías son usadas por los puentes para conectar sus extremos. En un principio, se usaron las vías “V70x30” en los puentes, luego fueron reemplazadas por la vía “V78x27”, para ajustarse a los requerimientos de la fabricación artesanal.



Departamento de electrónica - Taller de proyecto I

Pads

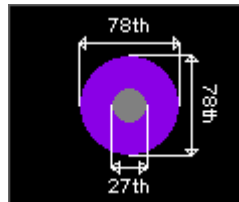


Figura 9: Diagrama del pad “M78X27”

Acorde a las sugerencias del ATEI, se implementó el pad M78X27 en todos los pines del PCB.

Reglas del diseño

- Pad - Pad: 20 th.
- Pad - Trace: 20 th.
- Graphic: 20 th
- Edge: 60 th



Departamento de electrónica - Taller de proyecto I

Conectores

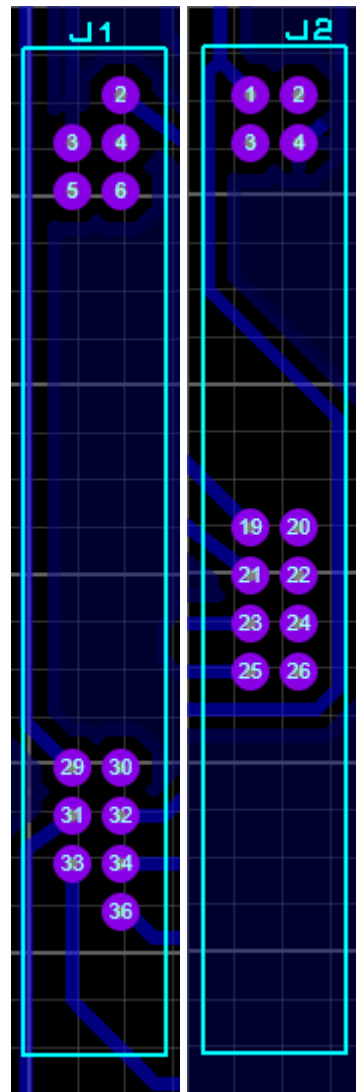


Figura 10: Diagrama las conectores en el PCB

La imagen anterior es una representación de las conectores (J1 y J2) en el PCB.



Departamento de electrónica - Taller de proyecto I

Pulsador

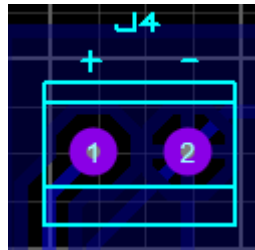


Figura 11: Diagrama de la bornera conectada al pulsador en el PCB.

El extremo positivo está conectado a GPIO2, el negativo a GND.

El diagrama del pad es de “M100X30” para un mejor agarre de la bornera, dado que esta posee unos tornillos para ajustar las conexiones, y el torque de los mismos al ajustarse puede ocasionar que se desuelda la bornera de la placa.

Sensor Ultrasonido HC-SR04



Figura 12: Diagrama del sensor ultrasónico.

Se realizó un puente para hacer una conexión del pin 1 a GND y se utilizó una separación estándar de los pines de 0.1in.



Departamento de electrónica - Taller de proyecto I

RTC DS3231

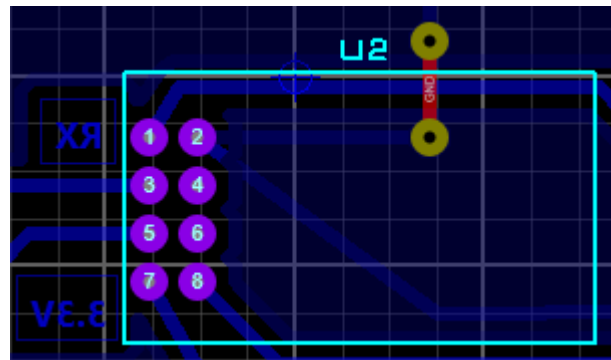


Figura 13: Diagrama del RTC.

Se realizó el footprint de acuerdo a las dimensiones del módulo.

ESP-01

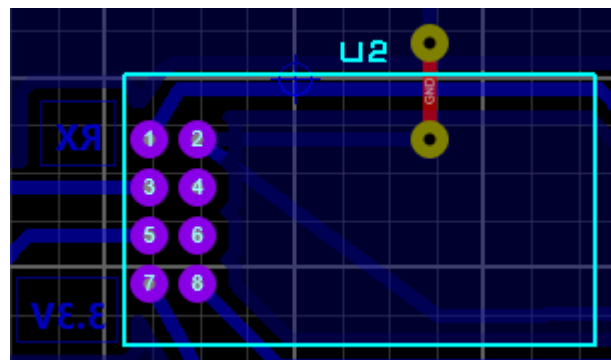


Figura 14: Diagrama del ESP-01.

Se realizó el footprint de acuerdo a las dimensiones del módulo, dado que el mismo no se encontraba en la librería de proteus. Se realizó un puente para conectar el pin a GND.



Departamento de electrónica - Taller de proyecto I

LED

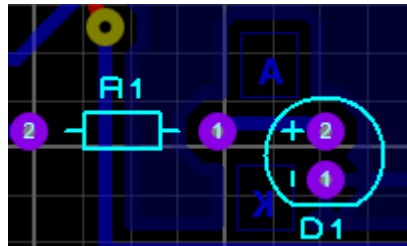


Figura 15: Diagrama del LED junto su resistencia R1.

El extremo positivo está conectado a la resistencia R1, el negativo a GND.

Notar que los símbolos del positivo y negativo son de vital importancia marcarlos en el cobre, para el momento de la colocación del LED al pcb, no confundir su polaridad.

Una vez realizado el circuito en proteus, el mismo fue impreso como se muestra en la siguiente imagen.

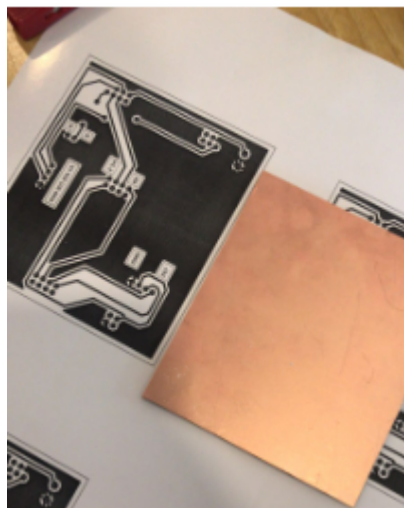


Figura 16: Impresión del circuito en papel Land.

Se recortó la placa de cobre a medida, y mediante un proceso de planchado, se adhiere la tinta de la impresión a la placa.



Departamento de electrónica - Taller de proyecto I

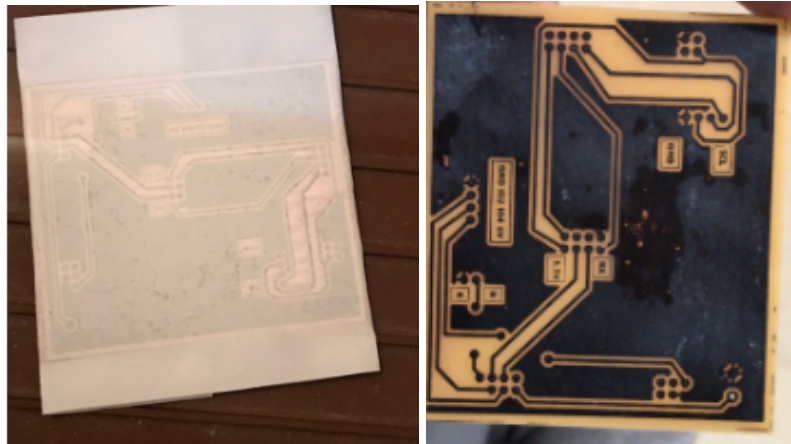


Figura 17: Impresión del circuito luego del planchado.

La misma se utiliza para proteger el cobre que se desea mantener, dado que luego la placa es sumergida en ácido, como muestra la siguiente imagen.

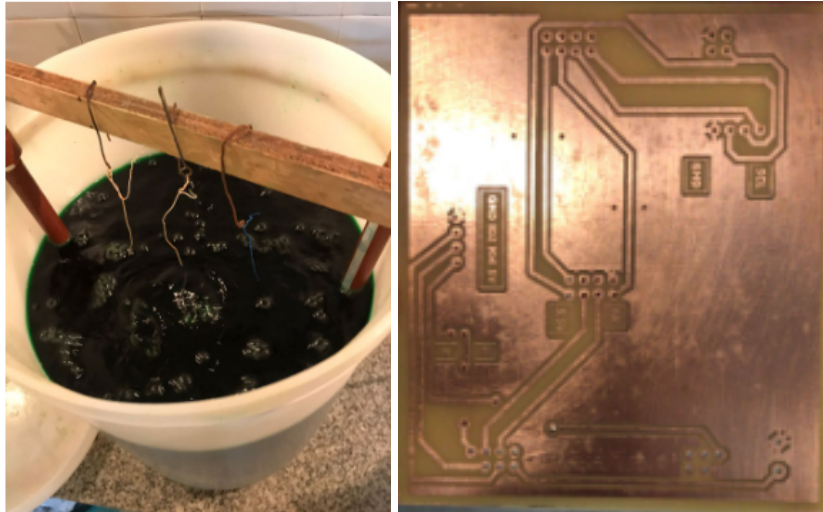


Figura 17: Impresión del circuito post ácido.

El proceso de sumergir en ácido la placa es para que el mismo ataque al cobre expuesto, y lo corroe dejando solamente el cobre con el dibujo de las pistas deseadas.

A continuación podemos ver la placa finalizada a la cual se le soldaron los componentes deseados y por último se obtuvo el “Poncho” para la EDU-CIAA correspondiente.

Departamento de electrónica - Taller de proyecto I

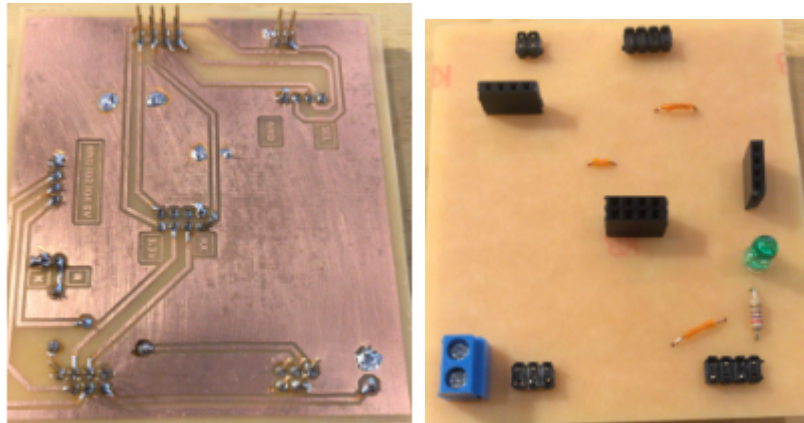


Figura 18: “Poncho” finalizado.

Diseño del software

La implementación del software necesario para el funcionamiento del sistema. Se basa en los siguientes módulos para la administración de los dispositivos:

- Pulsador
- HC_SR04
- BUILD_DEFs
- RTC
- ESP_01
- main

con sus respectivas librerías, como para obtener la fecha y hora de la última compilación y otra librería para el manejo de RTC, y un programa principal el cual utiliza una arquitectura superloop.

Pulsador

- void init_pulsador (void): Realiza la configuración del pin GPIO0 como un puerto de entrada con pull-up.



Departamento de electrónica - Taller de proyecto I

- `int flotantecarga(void)`: Devuelve el estado de la señal del flotante. Si devuelve 0, significa que el tanque está lleno y si devuelve 1 significa que el tanque está cargando.

hc_sr04

- `void HC_init(void)`: Inicializa el sensor ultrasónico.
- `float HC_Scan(void)`: Retorna la distancia en cm a la cual está ubicado el objeto.

build_defs

Esta librería se utiliza para otorgarle al RTC la fecha y hora inicial mediante las macros definidas en `C __DATE__` y `__TIME__` para devolver la fecha y hora en forma de campos separados en año, mes, día, hora, minutos y segundos apropiados para el uso del reloj.

`__TIME__` devuelve la hora en el momento de compilación en un formato “hh:mm:ss” en un formato de 24 hs (Ej. “22:29:12”).

`__DATE__` devuelve la fecha en el momento de compilación en un formato “mmm dd yyyy” (Ej. “31 Dec 2021”) en forma de tipo de dato string. En particular, la librería transforma el mes que es devuelto en caracteres a su correspondiente representación numérica (Ej. “Dec” se pasa a “12”).

rtc

Librería para la administración de un RTC DS3231, contiene funciones para la inicialización, de lectura y escritura de los registros del reloj. Dicha librería se implementó dado que utilizamos un RTC por comunicación I2C y no se encuentra una librería de estas características en EDU-CIAA.

Las funciones más usadas en el proyecto son las siguientes:

- `void RTC_read_control_registers(rtc3231_t* now, int32_t i2c, uint8_t address)`: Lee los registros de control del RTC. Guarda los contenidos en el registro `now`, para la lectura se necesita el parámetro `i2c` que para la placa EDU-CIAA es `I2C0` y la dirección del reloj, que es `0x68`.



Departamento de electrónica - Taller de proyecto I

- void RTC_Init(rtcDS3231_t* now, uint8_t year, uint8_t month, uint8_t month_day, uint8_t week_day, uint8_t hour, uint8_t minute, uint8_t second): Guarda en el registro now los valores iniciales para el RTC.
- void RTC_write_time(rtcDS3231_t* time, int32_t i2c, uint8_t address): Se escribe en los registros los valores especificados en time utilizando los parámetros i2c y address.
- void RTC_read_time(rtcDS3231_t* now, int32_t i2c, uint8_t address): Se lee el contenido de los registros del RTC y se lo guarda en el registro now.

esp01

Módulo basado en un ejemplo incluido en el firmware V3 para la administración del SoC ESP01 junto con el led que indica si el sistema está conectado o no a internet.

- bool_t esp01Init(uartMap_t uartForEsp, uartMap_t uartForDebug, uint32_t baudRate): Inicializa el modulo ESP01.
- void esp01CleanRxBuffer(void): Limpia el buffer RX
- bool_t esp01ShowWiFiNetworks(void): Muestra las redes Wi-Fi disponibles
- bool_t esp01ConnectToWifiAP(char* wiFiSSID, char* wiFiPassword): Se conecta a la red de Wi-Fi especificada con el wiFiSSID con la contraseña pasada en wiFiPassword
- bool_t esp01ConnectToServer(char* url, uint32_t port): Realiza la conexión con el servidor
- bool_t esp01SendTCPIPData(char* strData, uint32_t strDataLen): Envía los datos al servidor.
- bool_t esp01SendTCPDataToServer(char* url, uint32_t port, char* strData, uint32_t strDataLen): Envía lo que se encuentra en el parámetro data al servidor definido en url por el puerto port.
- int esp01_init(void): Inicializa la terminal UART_USB como salida para información del estado del chip, configura el pin GPIO5 como puerto de salida para el uso del LED, y realiza los pasos para la inicialización del ESP01 y su posterior



Departamento de electrónica - Taller de proyecto I

conexión a internet. Si los pasos anteriores se realizaron con éxito, se encenderá el LED. En caso contrario se quedará apagado.

- `int sendMessage(int64_t est_button, int64_t est_u_sensor, char time[100])`: Envía los datos a la plataforma ThingSpeak por medio de un mensaje que contendrá el estado del pulsador (`est_button`), la distancia medida por el sensor ultrasonico (`est_u_sensor`) y la hora del RTC (`time`).

main

El programa principal. Contiene el `setup`, el cual inicializa la placa y los componentes de nuestro sistema, y el `loop` el cual cada cierto periodo, el programa lee los valores del pulsador, el sensor ultrasónico y el RTC y los envía por Wi-Fi a Thingspeak para el usuario.

Pseudocódigo del `setup`:

Se inicializa la placa EDU-CIAA
Se inicializa la UART_USB a 115200 baudios
Se inicializa el pulsador
Se inicializa el sensor de ultrasonido
Se hace un retardo de 200 ms
Se inicializa el modulo ESP01
Se inicializa el RTC
Se hace un retardo de 100 ms

Pseudocódigo del `loop`:

Se lee los registros de control del RTC para ver el estado del bit `control_status`
Si el bit `control_status` es distinto de 0, es porque se inicia por primera vez o está mal la hora y fecha en el chip por lo que se carga la fecha y hora de compilación al RTC.
Se lee la fecha y hora del RTC
Se lee la distancia medida por el sensor de ultrasonido
Se lee la señal del pulsador
Se envía el mensaje a ThingSpeak a través del protocolo HTTP.
Se espera por un periodo de tiempo de aproximadamente 5 segundos.

Ensayos y mediciones

Prototipo conectado a la EDU-CIAA

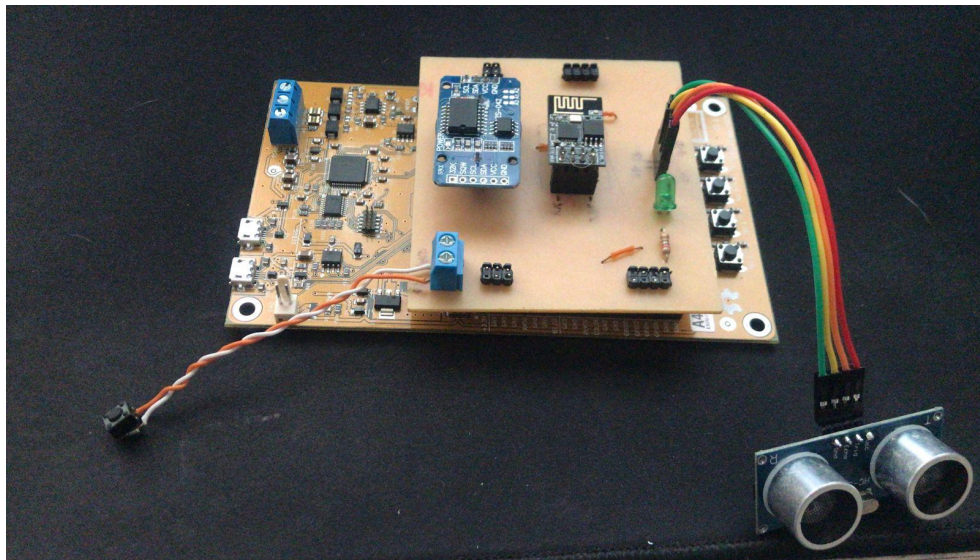


Figura 19: Prototipo en EDU-CIAA.

Una vez finalizada la impresión del PCB y colocados sus respectivos componentes. Colocamos el PCB sobre la EDU-CIAA y al terminar el desarrollo del software del sistema le cargamos el código correspondiente para resolver el problema en cuestión. Se hicieron pruebas para comprobar su funcionamiento. Como fue explicado en la sección del programa principal, se inicializan los componentes y en cada periodo se hace una lectura de los mismos y son transmitidos por Wi-Fi, en dicha placa se incorporó un led el cual indica que hay conexión a wifi para la transmisión de los datos a la plataforma Thingspeak. La plataforma recibe los datos y los muestra en gráficas como se observa a continuación:



Departamento de electrónica - Taller de proyecto I

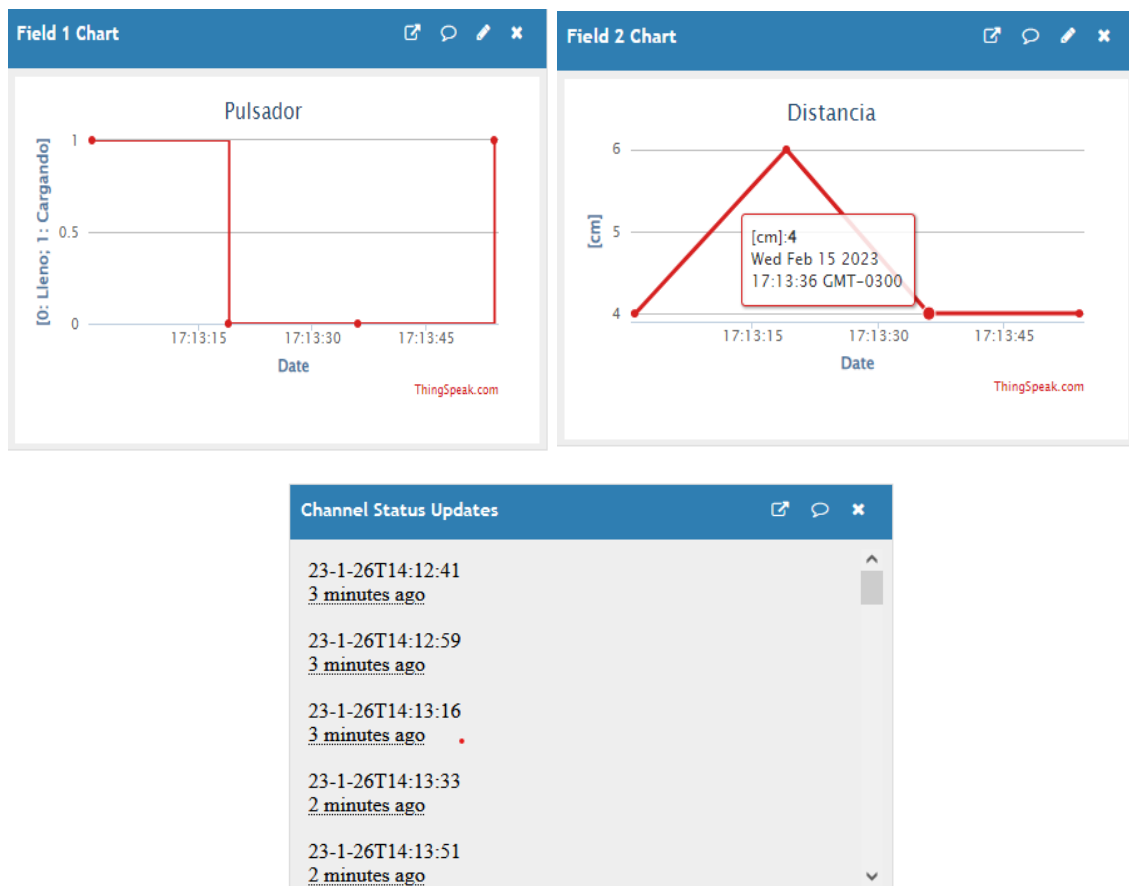


Figura 20: Gráficas obtenidas de Thingspeak.

El “Field 1 Chart” es la gráfica de la señal del pulsador, el eje X es la hora en la que llega el dato al servidor y el eje Y se mueve entre 0 y 1, donde 0 significa que el tanque está lleno y 1 que el tanque se encuentra cargando.

“Field 2 Chart” es la medición del sensor ultrasónico, el eje X es la hora en la que llega el dato al servidor y el eje Y es la distancia medida en centímetros.

La gráfica “Channel Status Updates” indica la fecha y hora del RTC en el momento que se envió el mensaje desde la placa en un formato especificado para poder comparar con el tiempo en que llegó al servidor.

Durante las mediciones se colocó un periodo entre cada mensaje de 17 segundos aproximadamente, pero para el proyecto finalizado estimamos que con un periodo de entre

Departamento de electrónica - Taller de proyecto I

10-30 minutos será acorde. Además, entre la hora de la llegada y la hora en la que se envía el mensaje hay una demora de 13 segundos aproximadamente, la cual es provocada por la latencia de la conexión del sistema hasta el servidor de ThingSpeak.

ThingSpeak recibe dichos datos con el siguiente formato: (Información del pulsador en entero, sensor ultrasonido en entero, fecha y hora formato texto).

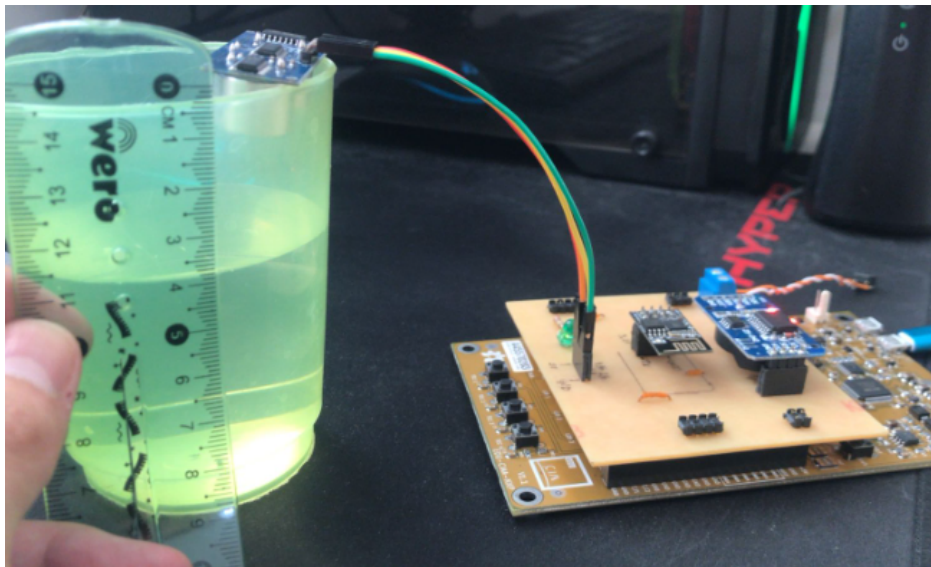


Figura 21: Prototipo en EDU-CIAA “Ensayo”.

Como se puede ver en la figura 17 en Field 2, el sensor envía una distancia de 4cm aproximadamente, dicha lectura corresponde a la figura 18, la cual para el ensayo se utiliza una lectura de distancia en centímetros desde el sensor al agua, con esto solo podemos demostrar que el sensor funciona correctamente pero para el sensado real de un bebedero de agua, se debería modificar el código en base a las dimensiones del bebedero, para poder determinar cuánta agua posee el mismo.



Departamento de electrónica - Taller de proyecto I

Dificultades en proceso de diseño

Durante el desarrollo del firmware se encontraron dificultades a la hora de desarrollar el firmware para el manejo del sensor ultrasónico y el RTC debido a la complejidad de desarrollar una librería desde cero. En el sensor ultrasónico, se dedicó más tiempo al estudio de la librería implementada por sAPI y encontrar la forma de modificarla para que sea compatible con nuestro sistema. En el caso del RTC, el problema principal fue que el tiempo estipulado en el cronograma de trabajo no fue lo suficientemente acorde, dado que no se tuvo en cuenta el estudio del chip zs-042 porque no contábamos con que no había una librería implementada por EDU-CIAA y se decidió implementar una librería externa.



Departamento de electrónica - Taller de proyecto I

Conclusiones

En los objetivos del proyecto se enumeraron varias metas para nuestro prototipo.

Objetivo definido	Grado de cumplimiento del objetivo
Medir carga completa del tanque con un pulsador activado mediante un flotante de carga.	Se cumplió con el objetivo.
Medir el nivel de agua a distintas alturas con un sensor de ultrasonido.	Se cumplió con el funcionamiento del sensor, pero se debería modificar el código para medidas reales de un cierto bebedero.
Para la comunicación inalámbrica, se usará un módulo wifi para conectar la placa EDU-CIAA a un servicio de IoT. En este proyecto se usará el servicio ThingSpeak con el protocolo MQTT para enviar los sensados.	Se realizó la comunicación a la plataforma Thingspeak exitosamente, pero no se llegó a implementar MQTT como protocolo de comunicación, se utilizó HTTP en su lugar.
Para el reloj, se hará uso de un RTC conectado a la EDU-CIAA con el cual se podrá informar al usuario la fecha y hora a la que se producen los sensados.	Se cumplió con el objetivo.
Medir el nivel de batería del dispositivo para informar al usuario cuando es necesario realizar un cambio de batería.	No se llegó a avanzar con este objetivo.



Departamento de electrónica - Taller de proyecto I

La aplicación web será una interfaz de usuario proporcionada por ThingSpeak, la cual mediante gráficos y texto informará al usuario del estado del recipiente.	Se cumplió con el objetivo.
--	-----------------------------

A partir de la tabla anterior, podemos decir que nuestro proyecto logró cumplir la mayoría de los objetivos propuestos y el producto final es capaz de sensar el nivel de agua de un tanque y enviar dicha información junto con el estado del pulsador y la fecha y hora del sensado a la plataforma Thingspeak para que el usuario pueda visualizar esa información de forma remota. A partir de ello, podemos concluir que se cumplió con la mayoría de los requerimientos descritos en la sección “Análisis de Requerimientos”, salvo con aquellos relacionados a la medición del nivel de la batería y la implementación del protocolo MQTT los cuales no fueron implementados aún, dado que el prototipo inicial se alimenta mediante usb, y la conexión a internet es mediante wifi porque aún no se adaptó el prototipo para zonas rurales que es el objetivo final.

Tiempo dedicado y Presupuesto final

Con el fin de demostrar las horas invertidas por los integrantes del grupo, se realizó una tabla con las tareas y horas dedicadas por cada uno.

Tabla 1. Actividades realizadas por los integrantes.

Tarea	Encargado	Horas dedicadas a la tarea
Desarrollo del software del pulsador	Dehan	4
Desarrollo del software de sensado de nivel	Duarte	8
Configuración de RTC	Dehan	16



Departamento de electrónica - Taller de proyecto I

Configuración de Wifi	Duarte	8
Comunicación con la nube	Dehan	8
Diseño del circuito esquemático	Duarte	16
Diseño del PCB	Dehan	14
Redaccion de documentacion	Duarte	16
Redacción de informes	Dehan	16
Adquisición de materiales	Duarte	8
Ensayos y validaciones	Dehan y Duarte	8
Fabricación de hardware	Dehan y Duarte	8

Integrante	Horas invertidas
Dehan	74
Duarte	72
Total	146

Se realizó un presupuesto aproximado de los componentes que conforman el proyecto:

Elemento	Precio
EDU-CIAA	\$30000
Pulsador	\$30
LED	\$43
Sensor Ultrasonico HC-SR04	\$630
ESP-01	\$800



Departamento de electrónica - Taller de proyecto I

RTC DS3231	\$1400
Placa Plaqueta Fenolico 10x15	\$500
Total	\$33403



Departamento de electrónica - Taller de proyecto I

Bibliografía

naylampmechatronics. SENSOR ULTRASONIDO HC-SR04 (2022). Disponible en internet:

<https://naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>

Consultado el 3/10/2022.

naylampmechatronics. MÓDULO ESP-01 ESP8266 WIFI-SERIAL (2022). Disponible en internet:

<https://naylampmechatronics.com/espressif-esp/48-modulo-esp-01-esp8266-wifi-serial.html>

Consultado el 3/10/2022.

maxim integrated. DS3231 Datasheet. Disponible en internet:

<https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>

Consultado el 17/10/2022.

Elecfreaks. Ultrasonic Ranging Module HC - SR04 Datasheet. Disponible en internet:

<https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>

Consultado el 17/10/2022.

AI Thinker. ESP-01 WiFi Module. Disponible en internet:

<https://www.microchip.ua/wireless/esp01.pdf>

Consultado el 17/10/2022.

Hernández, Luis del Valle. Guía para configurar un ESP-01, el módulo WiFi basado en ESP8266. Disponible en internet:

<https://programarfacil.com/podcast/como-configurar-esp01-wifi-esp8266/>

Consultado el 17/10/2022.



Departamento de electrónica - Taller de proyecto I

EDU-CIAA-NXP v1.1 Board - 2019-01-03 v5r0.pdf

Disponible en internet:

https://github.com/epernia/firmware_v3/blob/master/documentation/CIAA_Boards/NXP_LPC4337/EDU-CIAA-NXP/EDU-CIAA-NXP%20v1.1%20Board%20-%202019-01-03%20v5r0.pdf

Consultado el 17/10/2022.

Documentación de Thingspeak

Disponible en internet: <https://www.mathworks.com/help/thingspeak/>

Consultado el 17/10/2022



Departamento de electrónica - Taller de proyecto I

Anexos

Esquemático.

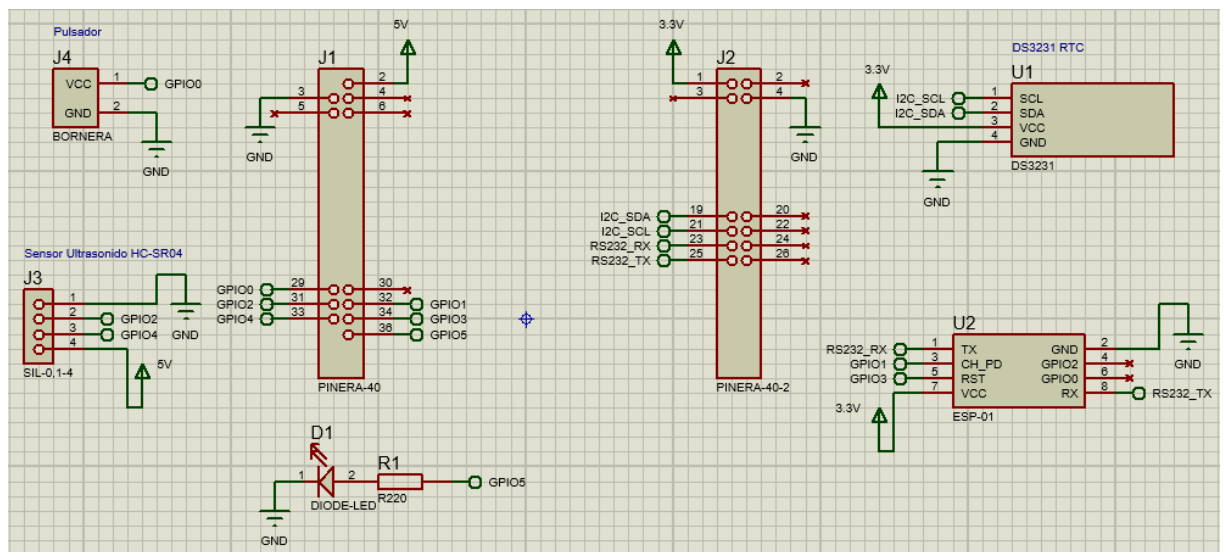
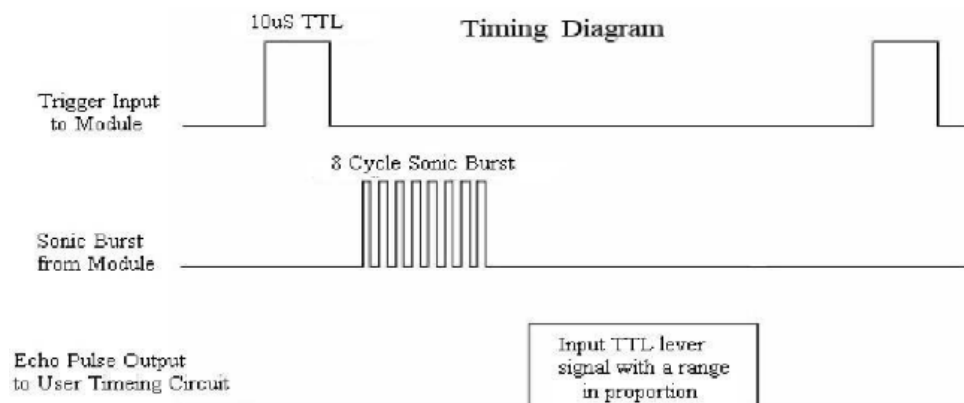


Figura A1. Esquemático completo del proyecto

Cómo funciona el sensor ultrasónico.

El funcionamiento del sensor ultrasónico consiste en emitir un sonido ultrasónico por uno de sus transductores, y esperar que el sonido rebote de algún objeto presente, el eco es captado por el segundo transductor. La distancia es proporcional al tiempo que demora en llegar el eco.





Departamento de electrónica - Taller de proyecto I

Figura A2. Diagrama de pulsos y salidas durante el sensado.

Se necesita un pulso de 10 us por el pin Trigger para iniciar el sondeo, entonces el sensor envía una rafaga de 8 ciclos de ultrasonido a 40kHz y coloca la salida del pin Echo en alto. Esta salida se mantendrá en alto hasta que se reciba un eco debido al rebote del sonido con algún objeto.

Cómo funciona un RTC

Un reloj de tiempo real (RTC) es un dispositivo electrónico que permite obtener mediciones de tiempo en las unidades temporales que empleamos de forma cotidiana. El DS3231 utiliza el protocolo I2C para la comunicación entre el reloj y la placa dado que la EDU-CIAA tiene pines definidos para una comunicación por medio de I2C.

En el DS3231 tenemos las siguientes direcciones de registros:

- Las direcciones 0x00 a 0x06 son los registros de fecha y hora
- Las direcciones 0x07 a 0x0D son los registros de alarma
- Las direcciones 0x0E a 0x10 son los registros de estado y control
- Las direcciones 0x11 y 0x12 son los registros de temperatura

Para escribir en el RTC DS3231 se realizan los siguientes pasos:

1. Transmitir la condición START del protocolo i2c
2. Transmitir la dirección del DS3231 (1001101) seguido de 0 para indicar una operacion de escritura.
3. Transmitir la dirección del campo al que se quiere acceder (escribir el valor del Register Pointer, este valor indica cual dirección será accedida).
4. Transmitir los datos necesarios.
5. Transmitir la condición STOP del protocolo i2c



Departamento de electrónica - Taller de proyecto I

Para leer desde el DS3231 se realizan los siguientes pasos:

1. Transmitir la condición START del protocolo i2c
2. Transmitir la dirección del DS3231(1001101) seguido de 1 para indicar una operación de lectura.
3. Transmitir la dirección del campo al que se quiere recibir los datos.
4. Esperar a recibir el mensaje del DS3231.
5. Transmitir la condición STOP del protocolo i2c

Thingspeak

Es una API y aplicación de código abierto para su uso en aplicaciones IoT que permite almacenar y recopilar datos de objetos conectados a través del protocolo HTTP o MQTT a través de Internet o una red local.

Para utilizar el servicio es necesario una cuenta de usuario, la plataforma ofrece varias licencias:

- Free: Cuenta gratuita para proyectos pequeños no-comerciales. En caso de uso comercial, la cuenta es de tiempo limitado.
- Standard: Cuenta paga para su uso en actividades comerciales o en organizaciones gubernamentales.
- Academic: Cuenta paga para su uso en instituciones académicas y/o de investigación.
- Student: Cuenta paga para estudiantes de cursos.
- Home: Cuenta paga para uso personal solamente.

Para realizar la conexión con el hardware, es necesario conocer la “API Key” del canal para las operaciones de lectura en caso que desee recibir datos de la nube o para operaciones de escritura en caso de enviar datos a la nube.

Como el proyecto se basa en enviar datos a la nube, el firmware envía un mensaje al servidor de Thingspeak con la “Write API Key” para identificar al canal al que tiene que escribir junto con los campos a enviar. Thingspeak acepta los siguientes parámetros en el mensaje:



Departamento de electrónica - Taller de proyecto I

- field<x>: Dato del campo X, donde el X es el ID del campo. El rango de X es del 1 al 8.
- lat: Latitud en grados, el rango de valores es de -90 a 90.
- long: Longitud en grados. Su rango de valores es de -180 a 180.
- elevation: Elevación en metros.
- status: Mensaje de estado del canal.
- twitter: Usuario de Twitter vinculado con ThingTweet
- tweet: Mensaje de estado de Twitter
- created_at: Fecha el cual fue creado el mensaje. Su formato es “AAAA-MM-DD HH:MM:SS”



Departamento de electrónica - Taller de proyecto I

PCB

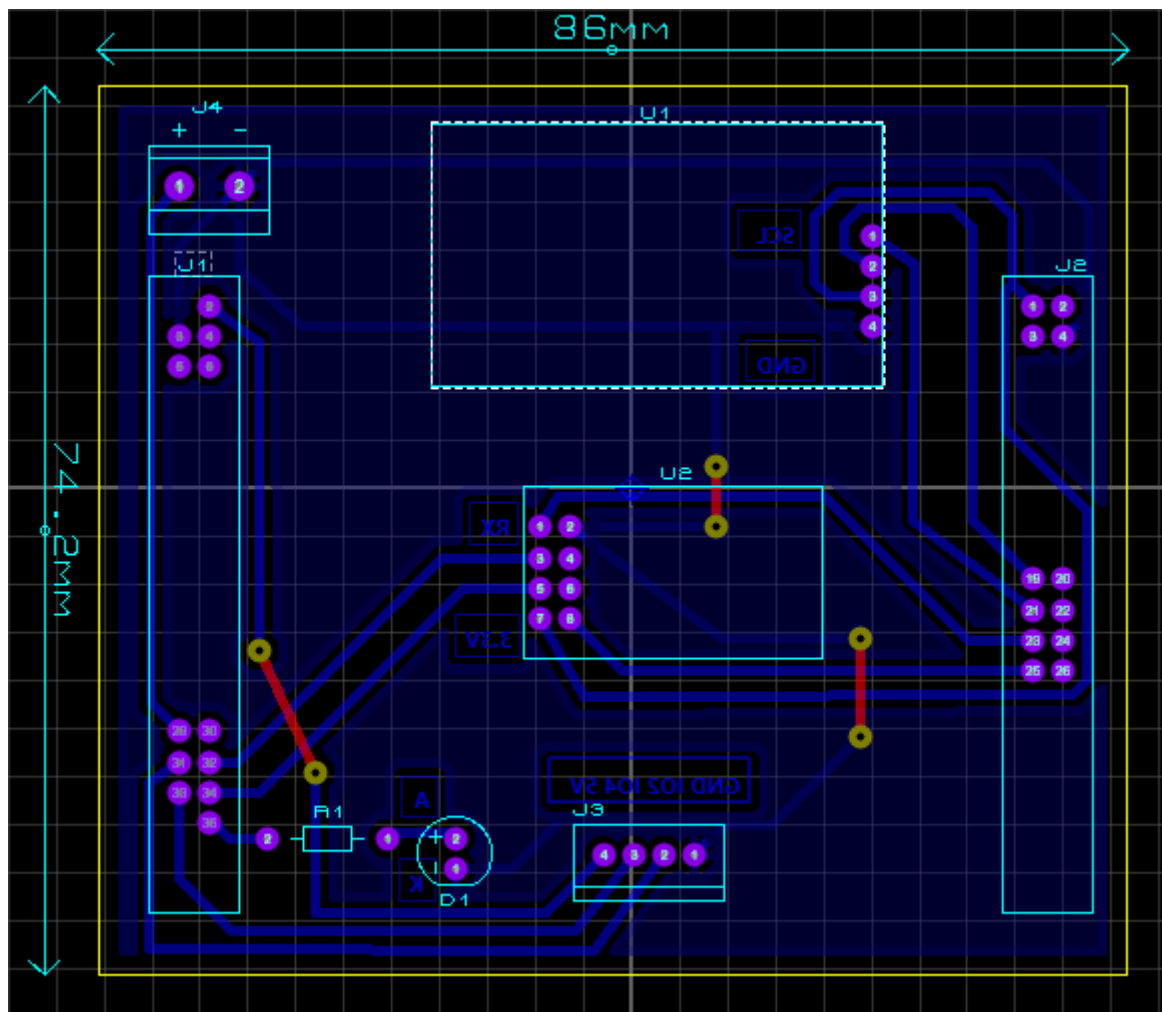


Figura A3. Diagrama del PCB completo en la vista “PCB Layout” con sus medidas aproximadas.



Departamento de electrónica - Taller de proyecto I

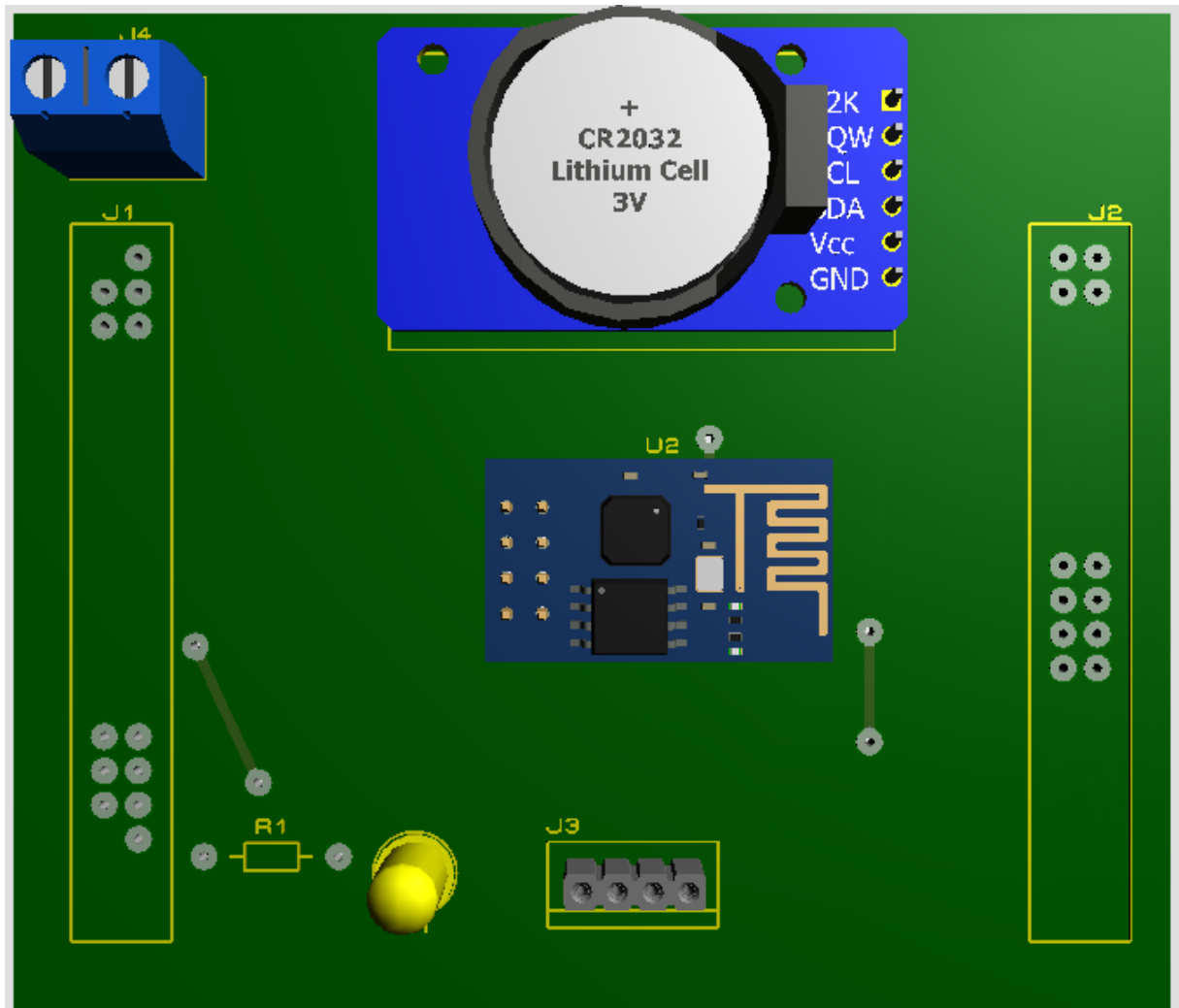


Figura A4. Diagrama del frente del PCB completo en la vista “3D Visualizer”



Departamento de electrónica - Taller de proyecto I

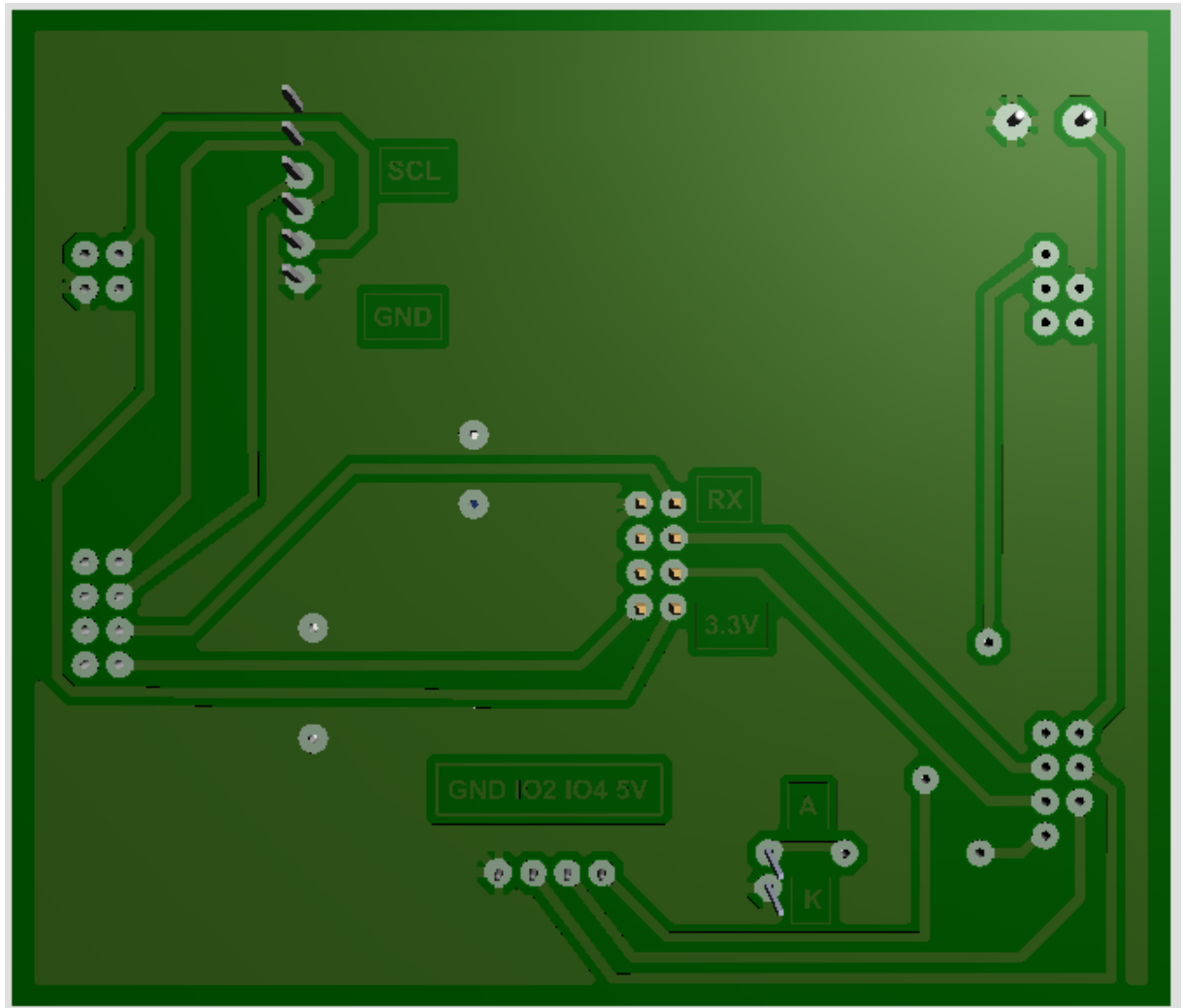


Figura A5. Diagrama del dorso del PCB completo en la vista “3D Visualizer”

Se observa que dos de los pines del RTC atraviesan por las tramas del PCB, pero durante la impresión del PCB se removerán esos dos pines ya que no se usan en el prototipo.



Departamento de electrónica - Taller de proyecto I

Cronograma final

Representa la agenda de trabajo, considerando el lapso desde que se eligió el tema hasta la entrega del informe final.

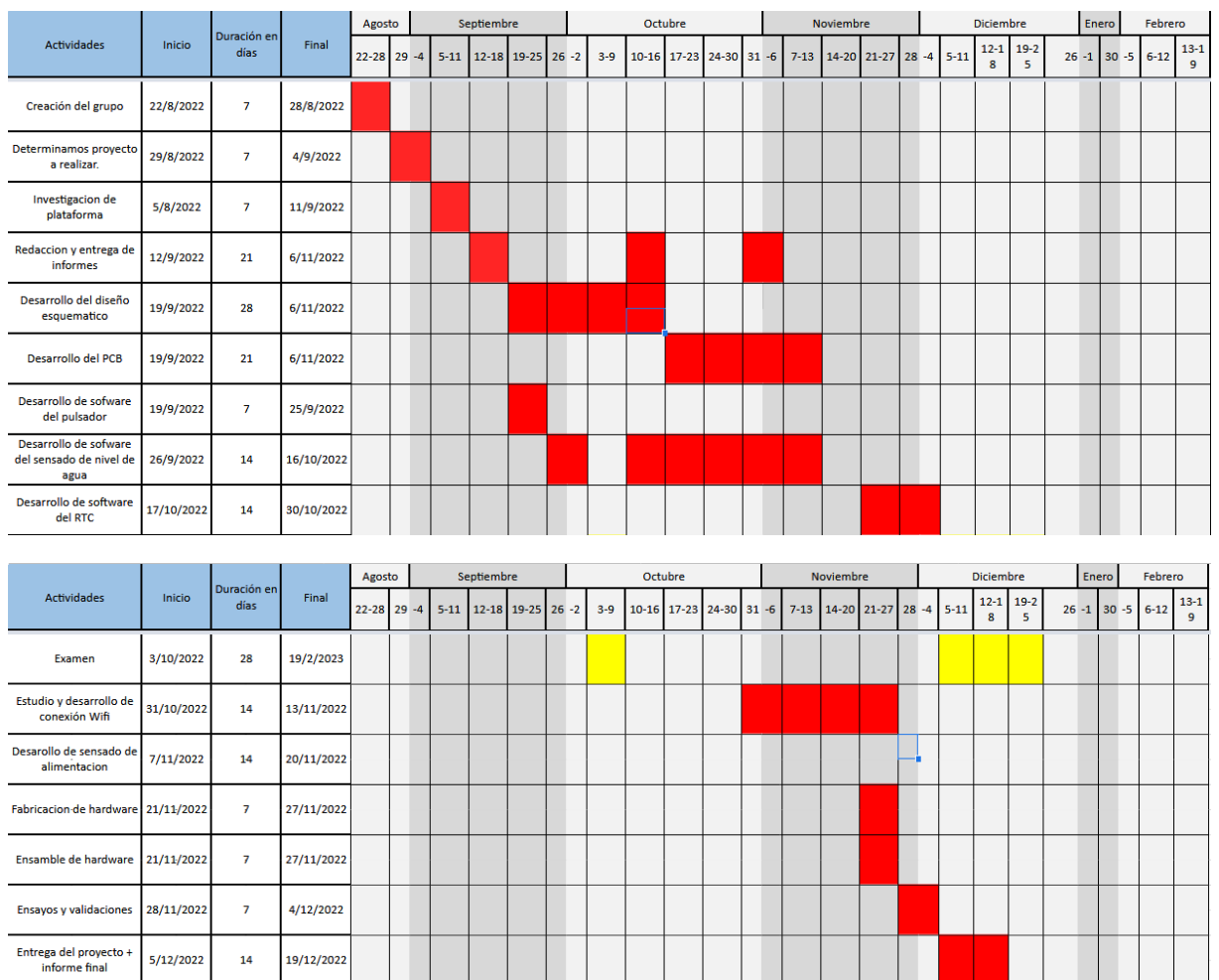


Figura A6. Diagrama de Gantt



Departamento de electrónica - Taller de proyecto I

Lista de Materiales

A continuación se detalla la lista de materiales generados por Proteus para el esquemático.

Bill Of Materials for Esquematico

Design Title Esquematico
Author Dehan Lucas, Victor Duarte
Document Number 1
Revision Final
Design Created jueves, 6 de octubre de 2022
Design Last Modified lunes, 28 de noviembre de 2022
Total Parts In Design 8

0 Modules				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
Sub-totals:				\$0,00
0 Capacitors				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
Sub-totals:				\$0,00
1 Resistors				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
1	R1	R220		\$36,00
Sub-totals:				\$36,00
2 Integrated Circuits				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
1	U1	DS3231		\$1.400,00
1	U2	ESP-01		\$800,00
Sub-totals:				\$2.200,00
0 Transistors				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
Sub-totals:				\$0,00
1 Diodes				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
1	D1	DIODE-LED		\$43,00
Sub-totals:				\$43,00
4 Miscellaneous				
<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
1	J1	PINERA-40		
1	J2	PINERA-40-2		
1	J3	SIL-0, 1-4		
1	J4	BORNERA		
Sub-totals:				\$0,00
Totals:				\$2.279,00

jueves, 16 de febrero de 2023 18:23:34



Departamento de electrónica - Taller de proyecto I

Agradecimientos

Deseamos expresar nuestro agradecimiento a la cátedra Taller de Proyectos I,

Tanto a el Prof: Jose Maria Juarez y a los ayudantes. por la oportunidad de trabajar en este proyecto. Por la dedicación y apoyo que han brindado a este trabajo e ideas aportadas al mismo.