

TECNOLOGICO NACIONAL DE MEXICO
INSTITUTO TECNOLÓGICO DE ENSENADA
INGENIERÍA EN ELECTROMECAÁNICA

RESIDENCIA PROFESIONAL

**ASESORÍAS PARA LA DOCUMENTACIÓN Y REPRODUCCIÓN DE
DISPOSITIVOS CON TECNOLOGÍAS ABIERTAS PARA EL MONITOREO
DE EDIFICACIONES**

ELABORADA POR:
ESCOBAR ESCOBAR JOSÉ WILLIAMS
No. CONTROL: **15090422**

LUGAR DE ADSCRIPCIÓN:
INSTITUTO TECNOLÓGICO DE ENSENADA

ASESOR INTERNO:
M.I. JUAN PABLO VITE FRÍAS

ASESOR EXTERNO:
ING. FRANCISCO RAMOS FLORES

ENSENADA, B.C.
DEL 16/10/20 AL 16/04/21

Agradecimientos

Agradezco a: Mi madre por darme la oportunidad de seguir estudiando algo que me gusta, por enseñarme a ser constante en todo lo que realizo, por enseñarme a ser disciplinado desde pequeño, por inculcarme valores de ética que hasta el día de hoy siguen intactos y me hacen mejor persona, saber que tengo su apoyo incondicional me impulsa a superarme todos los días, mi motivación para nunca rendirme.

Mis abuelos, por enseñarme a trabajar y a ser una persona con compromiso, por mostrarme que independientemente de las circunstancias, siempre hay soluciones para salir adelante.

Mis tías, por compartir tiempo y paciencia en mi formación escolar, por hacerme sentir como el hermano menor de la familia, mi apoyo hacia ellas, estará siempre que lo necesiten.

Mi tío, por ser el claro ejemplo que para llegar a donde quieres, es necesario prepararse bien, ser constante en los estudios, los consejos que me da, siempre me orientan para dar el siguiente paso en mi vida académica y profesional.

A los doctores Guillermo Barrios y Guillermo Ramírez, por darme la oportunidad de apoyar en el desarrollo de este proyecto, por la confianza que me brindaron, pude demostrar habilidades que ni yo mismo sabía que tenía para aportar conocimientos adquiridos durante la carrera.

Resumen

En el Instituto de Energías Renovables (IER), campus de la Universidad Nacional Autónoma de México (UNAM), se desarrollan dispositivos electrónicos para el monitoreo de edificaciones enfocados al internet de las cosas (IOT), por sus siglas en inglés, en donde, el poder controlar las diferentes variables ambientales entre otras, es de suma importancia para lograr un edificio sustentable con un buen confort térmico, para ello, se desarrollan dispositivos electrónicos que puedan medir dichas variables, no sólo se trata de medir y hacer los dispositivos electrónicos, siempre se busca que sean desarrollados con tecnologías abiertas, disponibles para todo público, esta filosofía de trabajo que se maneja dentro del IER, en la etapa actual del proyecto, lo importante a desarrollar es una metodología de documentación enfocada a dispositivos electrónicos o electrónica en general, existen muchas formas de documentar pero ninguna que ayude a documentar los dispositivos etapa por etapa y que se adapte a las actualizaciones que puedan surgir en el desarrollo de los mismos, las existentes se enfocan a la documentación de Software y Hardware pero en etapa final o cuando ya está comprobado el funcionamiento, lo que se busca es poder documentar con un formato simple pero eficiente para las actualizaciones que puedan sufrir los dispositivos durante el desarrollo, dicho lo anterior, lo que se desarrolla en este proyecto, es el buscar un método de documentación que sea fácil de implementar, que se puedan hacer cambios en la documentación, que pueda ser editado fácilmente para otras personas que se sumen al desarrollo de los dispositivos, uno de los principales requisitos para la elección del método, es el poder implementarlo en todos los dispositivos que están avanzados actualmente y no sólo en un dispositivo, también en este proyecto escrito se realizarán actualizaciones de diseño de alguno de los dispositivos, si hay una oportunidad de mejora, es conveniente aplicarlo siempre respaldando esas actualizaciones.

Pareciera que es una tarea fácil el poder implementar un método de documentación, pero la verdad resulta un poco difícil aplicar alguno de los métodos existentes porque los dispositivos, actualmente no están terminados al cien por ciento, entonces se trata de buscar complementos para llegar a un solo método.

Índice

Agradecimientos	i
Resumen.....	ii
Índice	iii
Generalidades del proyecto	1
Introducción	1
Empresa u organización.....	2
Problemas a resolver	3
Objetivos	4
Objetivo general	4
Objetivos específicos	4
Justificación	5
Marco teórico	6
Desarrollo.....	23
Resultados	36
Conclusiones.....	45
Competencias desarrolladas.....	46
Fuentes de información.....	47
Anexos	48
Anexo 1	48

Anexo 2.....	49
Anexo 3.....	50
Anexo 4.....	51

Generalidades del proyecto

Introducción

El futuro de la humanidad y del planeta dependen de la manera en la que produzcamos energía: un sistema energético fiable, asequible y descarbonizado es fundamental.

Las energías renovables son fuentes de energía limpias, inagotables y crecientemente competitivas. Se diferencian de los combustibles fósiles principalmente en su diversidad, abundancia y potencial de aprovechamiento en cualquier parte del planeta, pero sobre todo en que no producen gases de efecto invernadero, causantes del cambio climático, ni emisiones contaminantes.

El crecimiento de las energías renovables es imparable, como queda reflejado en las estadísticas aportadas anualmente por la Agencia Internacional de la Energía (AIE): según las previsiones de la AIE, la participación de las renovables en el suministro eléctrico global pasará del 26% en 2018 al 44% en 2040, y proporcionarán 2/3 del incremento de demanda eléctrica registrado en este período, principalmente a través de las tecnologías eólica y fotovoltaica.

De acuerdo a la AIE, la demanda mundial de electricidad aumentará un 70% hasta 2040, elevando su participación en el uso de energía final del 18% al 24% en el mismo período, espoleada principalmente por regiones emergentes (India, China, África, Oriente Medio y el Sureste asiático).

Con lo anterior y compartiendo la idea de utilizar más las energías limpias, se implementan nuevas formas de controlar los diferentes recursos que se utilizan en el día a día en el Instituto de Energías Renovables, campus de la UNAM, se implementarán dispositivos que puedan medir la cantidad de recursos que se consumen como lo son: el agua, la energía eléctrica, humedad, intensidad luminosa, temperatura, etc., de un edificio del instituto, con la intención de poder medir cada variable y posteriormente observar cuanto se consume para tratar de reducirlos en gran medida, lo anterior se lleva a cabo con dispositivos electrónicos que están integrados principalmente con sensores y haciendo uso de tecnologías abiertas para los programas y envío de datos a la plataforma de Thingsboard para la visualización de gráficos y control.

Empresa u organización

Nombre de la institución:

Instituto de Energías Renovables (IER-UNAM)

Logotipo:



Fig. 1 Logotipo del IER-UNAM

Dirección:

Priv. Xochicalco S/N Temixco, Morelos 62580 México

Giro:

Institución educativa, investigación, innovación, divulgación y formación de especialistas en tecnologías energéticas sustentables.

Área de desempeño del estudiante:

Electrónica enfocada a energías renovables

Problemas a resolver

- Investigar formas de documentar proyectos electrónicos, elegir el adecuado e implementarlo a los dispositivos existentes.
- Revisión de los documentos existentes de los diferentes dispositivos para ver si tienen lo necesario para poder replicar cada uno de manera eficiente.
- Al momento de replicar cada dispositivo, surgen detalles de manufactura, documentar esos detalles para reducir el tiempo y un mejor entendimiento para la práctica.
- Implementar mejoras en caso de ser necesario.

Objetivos

Objetivo general

Replicar dispositivos dedicados a medir parámetros de confort térmico, consumo de energía eléctrica y gasto de agua potable.

Objetivos específicos

- Replicar (manufactura) los dispositivos previamente diseñados por el Grupo de Edificaciones y evaluar la documentación actual de los diferentes dispositivos.
- Determinar si la documentación cumple con los requisitos para replicar de forma rápida y sencilla los dispositivos.
- Utilizar projects de Github para administrar los proyectos.

Justificación

Evaluar el gasto de energía eléctrica del consumo de agua potable representa áreas de oportunidad para reducir gastos innecesarios para el consumidor y sobre todo para reducir la contaminación en general por producción de energía eléctrica, energía empleada para el confort térmico en edificaciones, por otro lado, el consumo desmedido y fugas de agua (si existiese), agotan y desperdician el agua potable, causando que se tenga menos recurso híbrido. Por lo antes mencionado es necesario contar con dispositivos económicos y de bajo consumo energético para evaluar el consumo de energía eléctrica y del gasto de agua potable.

Marco teórico

Revisión bibliográfica

Los siguientes párrafos contienen información de lo más usado en el desarrollo del proyecto, dispositivos, plataformas, etc.

Sensor de caudal

Un sensor de flujo o caudalímetro es un instrumento para la medición de caudal o gasto volumétrico de un fluido, El caudal es la cantidad de líquido o fluido (volumen) que circula a través de una tubería por unidad de tiempo, por lo general se expresa en: litros por minutos (L/min), litros por hora (L/h), metros cúbicos por hora (m^3/h), etc.

Los caudalímetros suelen colocarse directamente en la tubería que transporta el fluido.

YF-S201

El sensor de flujo de agua de 1/2" YF-S201 sirve para medir caudal de agua en tuberías de 1/2" de diámetro. También puede ser empleado con otros líquidos de baja viscosidad, como: bebidas gasificadas, bebidas alcohólicas, combustible, etc. Es un caudalímetro electrónico de tipo turbina. Compatible con sistemas digitales como Arduino, Pic, Raspberry Pi, PLC. El sensor posee tres cables: rojo (VCC: 5VDC), negro (tierra) y amarillo (salida de pulsos del sensor de efecto Hall).

El funcionamiento del sensor es de la siguiente forma: el caudal de agua ingresa al sensor y hace girar una turbina, la turbina está unida a un imán que activa un sensor de efecto Hall, que a su vez emite un pulso eléctrico que puede ser leído por la entrada digital de alguno de los y mencionados. El sensor de efecto Hall está aislado del agua, de manera que siempre se mantiene seco y seguro. Como el volumen de agua por cada pulso es fijo y de un valor conocido (promedio) se puede contar la cantidad de pulsos por unidad de tiempo (segundo o minuto), luego multiplicar el valor de volumen/pulso por la cantidad de pulsos y así determinar el caudal o flujo de agua.

ESP8266-12E

Se trata de un chip integrado con conexión WIFI y compatible con el protocolo TCP/IP.

El objetivo principal es dar a cualquier microcontrolador a una red.

La gran ventaja del ESP8266 es su bajo consumo. Es el producto ideal para wearables y dispositivos del IoT.

Al parecer lleva mucho tiempo en el mercado de los microcontroladores, pero fue hasta agosto del 2014 cuando en realidad dio de que hablar.

Poco a poco, la gran comunidad que lo había usado, empezó a traducir la documentación y a crear firmwares para el ESP8266. Un firmware no es más que el software de bajo nivel, la lógica que controla los circuitos electrónicos.

El uso que este microcontrolador se le puede dar dependerá de si se tiene como chip o como módulo. Dentro de la gran cantidad de usos cabe destacar los siguientes:

- Electrodomésticos conectados
- Automatización del hogar
- Casas inteligentes
- Automatización de la industria
- Monitor de bebés
- Cámaras IP
- Redes de sensores
- Wearables
- IoT (Internet of Things)
- IIoT (Industrial Internet of Things)

Hardware

- Utiliza una CPU Tensilica L106 32-bit
- Voltaje de operación entre 3V y 3,6V
- Corriente de operación 80 mA
- Temperatura de operación -40°C y 125°C

Conectividad

- Soporta IPv4 y los protocolos TCP/UDP/HTTP/FTP

Puertos GPIO (de propósito general)

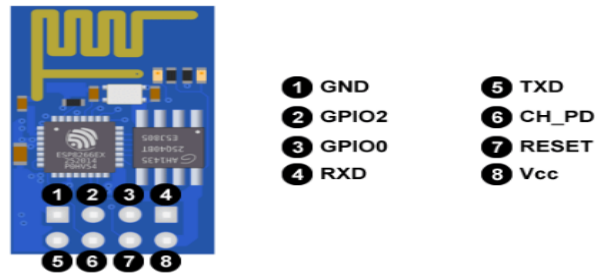


Fig. 2 Chip ESP8266-12E

- Tiene 17 puertos GPIO pero solo se pueden usar 9 o 10. El GPIO16 es especial ya que está conectado al RTC (Real Time Clock).
- Pueden ser configurados con resistencia Pull-up o Pull-down.
- Soporta los principales buses de comunicación (SPI, I2C, UART).

Consumo

Dependerá de diferentes factores, como, por ejemplo, del modo en el que se esté trabajando el ESP8266, de los protocolos que estemos utilizando, de la calidad de la señal Wifi y sobre todo de si enviamos o recibimos información a través de la Wifi. Oscilan entre los 0,5 μ A cuando el dispositivo está apagado y los 170 mA cuando transmitimos a tope de señal.

Formas de operación

Debido a los sectores que va enfocado, wearables, dispositivos del IoT y móviles, el ESP8266 requiere de una gestión de energía eficaz. Dispone de una arquitectura de bajo consumo que trabaja en 3 modos.

Active mode o modo activo: a pleno rendimiento.

Sleep mode o modo dormido: solo el RTC (Real Time Clock) está activo para mantener la sincronización. Se queda en modo alerta de los posibles eventos que le hagan despertar. Mantiene en memoria los datos de conexión y así no hace falta volver a establecer la conexión con la Wifi. Consume entre 0,6 mA y 1mA.

Deep sleep o modo en sueño profundo: el RTC está encendido, pero no operativo. Debe pasar por el modo dormido antes de despertar. Hay que llevar especial cuidado con los

datos ya que en este estado es como si estuviera apagado y todos los datos que no estén almacenados se pierden, consume alrededor de 20 μ A.

NodeMCU

El NodeMCU es el módulo más característico de este tipo. Su precio ronda desde los \$100.00 a los \$150.00. A diferencia de los otros módulos, viene con todo lo necesario para empezar a trabajar de forma autónoma. Incluye un adaptador serie/USB y se alimenta a través del micro-usb. Está basado en el ESP-12 y la última versión oficial es la 2. Lo más interesante de este módulo es que se puede descargar un firmware que permite programar en lenguajes como LUA, Python, Basic o JavaScript. Sin duda alguna este módulo es la mejor opción si queremos adentrarnos en el mundo del ESP8266.



Fig. 3 Módulo ESP8266-12E NodeMCU V3 Lolin

Fritzing



Fig. 4 Logo del software Fritzing

Es un software de desarrollo de circuitos impresos el cual es open source y está disponible para Windows, Mac y Linux.

Conociendo Fritzing

Una vez dentro de Fritzing se verá la pantalla principal la cual tiene algunas partes que se describen a continuación.

- Nuestros Proyectos: En esta parte están los accesos directos a los proyectos más recientes en los que se ha trabajado.
- Vistas disponibles: Al trabajar en Fritzing se podrá visualizar de 3 formas diferentes, las cuales son: Protoboard, esquema y PCB.
- Lista de componentes: Aquí se encuentra una gran biblioteca de componentes más básicos y hasta tarjetas de desarrollo importantes tales como Arduino.
- Inspector de partes: En esta sección se pueden ver detalles de cada componente que seleccionemos.

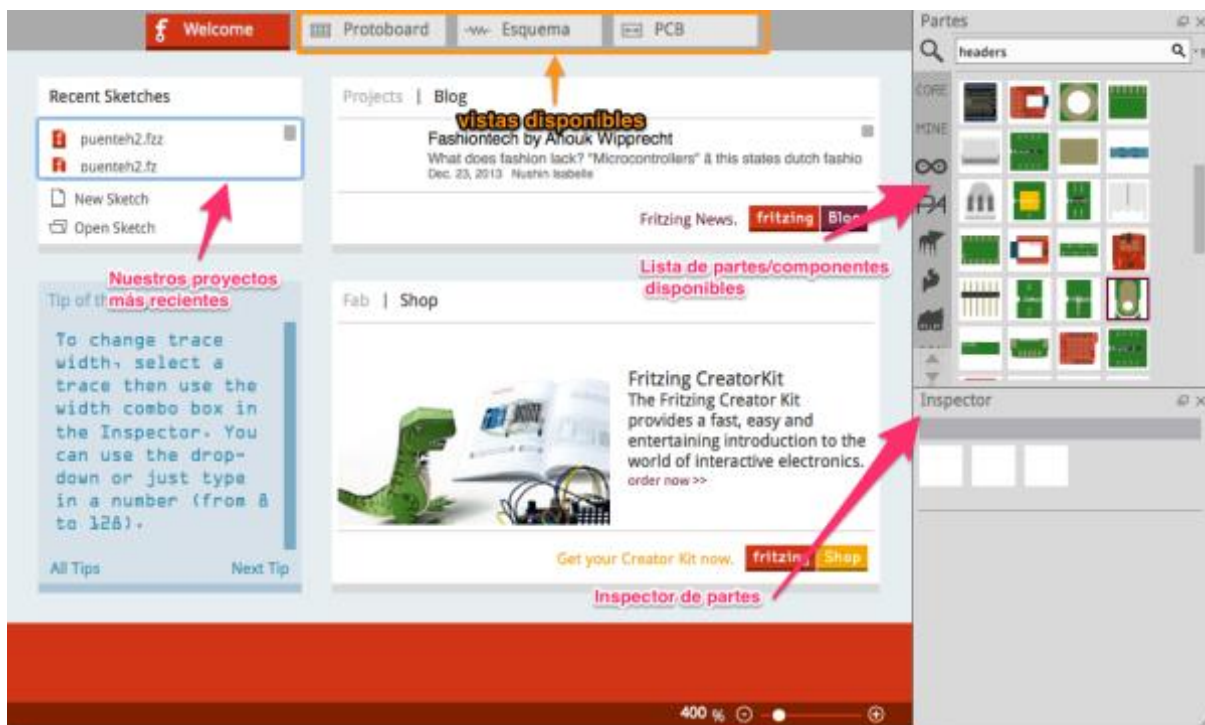


Fig. 5 Identificación de las partes importantes del software Fritzing

Tinkercad

Es un software gratuito online por la empresa Autodesk, una de las empresas punteras en programas de diseño 3D.

El objetivo, al usar Tinkercad, debe ser una primera inmersión en el mundo del diseño 3D de una manera sencilla, ya que la interfaz de trabajo es simple y muy atractiva inicialmente, si bien una vez dominados los conceptos básicos carece de herramientas para llegar a diseños complejos.

Sus ventajas son claras: es sencillo de usar, su aspecto es atractivo y con unas pocas horas de entrenamiento podemos adquirir mucha destreza en su uso.

Como desventaja, se podría señalar que es necesario tener una cuenta de correo para darse de alta como usuario y que sólo posee una versión online, por lo que hace falta conexión a internet.



Fig. 6 Logo del software de diseño Tinkercad

EasyEDA



Fig. 7 Logo del software de diseño PCB

EasyEDA es un paquete de herramientas EDA gratuitas, sin instalación, basadas en la web y la nube, que integra un diseño de esquemas, un simulador de circuitos de señal mixta y un diseño de circuito impreso en un entorno de navegador multiplataforma integrado para ingenieros electrónicos, educadores, estudiantes y aficionados.

Con EasyEDA se ahorra tiempo de diseño usando la masiva cantidad de bibliotecas de componentes, huellas de circuito impreso y encapsulado, modelos de simulación spice y subckts.

- Cool Schematic Capture con EasyEDA.
Permite dibujar esquemas rápidamente en el navegador usando las bibliotecas disponibles. Actualizaciones automáticas transparentes.
- Simulador de circuitos Spice inteligente con EasyEDA.
Se puede verificar circuitos analógicos, digitales y de señal mixta con subcircuitos y modelos spice.
- Diseño de circuitos en línea con EasyEDA.
Con múltiples capas, miles de pads, se puede seguir trabajando con rapidez y diseñar cómodamente.
- Importación de ficheros de esquemas y circuitos impresos con EasyEDA.
Con la ventaja de importar diseños desde Eagle, Altium, Kicad y LTspice.

Diseño de Hardware

- Desarrollo colaborativo.
Existe la opción de que puedan colaborar más personas en el diseño de ingeniería electrónica. Podrán colaborar y comentar diseños fácilmente con opciones de compartir proyectos públicos, privados o de acceso controlado.
- Open-source Hardware.
Se puede aprovechar el poder de la nube, accediendo a módulos Open source desarrollados por miles de ingenieros electrónicos.
- Compartir esquemas y circuitos impresos.
Opción de compartir diseños fácilmente de proyectos públicos, privados y de acceso controlado.
Desarrollo en la nube.
- En cualquier momento y lugar, en Linux, Mac, Windows, Android, PC, Tabletas o Smartphones.

IoT

En los últimos años se ha convertido en una de las tecnologías más importantes del siglo XXI. Ahora que podemos conectar objetos cotidianos como aparatos de cocina, vehículos, termostatos, monitores de bebés a internet mediante dispositivos integrados, es posible la comunicación fluida entre personas, procesos y cosas.

Por medio de la informática de bajo coste, la nube, big data, la analítica y las tecnologías móviles, los objetos físicos pueden compartir y recopilar datos con una intervención humana mínima. En este mundo hiperconectado, los sistemas digitales pueden grabar, supervisar y ajustar cada interacción entre los objetos conectados.

Las tecnologías que hacen posible IoT

La idea que hace referencia al término de IoT existe desde hace mucho tiempo, pero los avances actuales en tecnología la han hecho toda una realidad.

- El acceso a tecnología de sensores de bajo coste y baja potencia.
Los sensores asequibles y fiables hacen que la tecnología de IoT sea posible para más fabricantes.
- Conectividad.
Un conjunto de protocolos de red para internet ha hecho que sea fácil conectar sensores a la nube y a otras “cosas” para conseguir una transmisión de datos eficiente.
- Plataformas de Cloud Computing.
El aumento de la disponibilidad de las plataformas en la nube permite que tanto las empresas como los consumidores acceden a la infraestructura que necesitan para ampliar la capacidad sin tener que gestionarlo todo.
- Machine learning y analítica.
Con los avances logrados en machine learning y en analítica, junto con el acceso a enormes cantidades de datos de una nueva gran variedad almacenados en la nube, las empresas pueden reunir información más rápido y de forma más sencilla. El surgimiento de estas tecnologías relacionadas sigue ampliando los límites de IoT y los datos producidos por IoT también retroalimentan estas tecnologías.

- Inteligencia artificial (IA) conversacional.

Los avances en redes neuronales han llevado el procesamiento de las lenguas naturales (NLP) a los dispositivos de IoT (por ejemplo, los asistentes personales Alexa, Cortana y Siri) y los han convertido en dispositivos atractivos, asequibles y viables para el uso doméstico.

¿Qué es IoT industrial?

IoT industrial (IIoT) hace referencia a la aplicación de la tecnología de IoT en ámbitos industriales, en especial con respecto a la instrumentación y el control de los sensores y dispositivos que participan en las tecnologías en la nube. Recientemente, las industrias han utilizado la comunicación entre máquinas (M2M) para lograr una capa de automatización y crear con ella nuevos ingresos y modelos de negocio. A veces, IIoT se denomina la cuarta ola de la revolución industrial o Industria 4.0.

Las aplicaciones más comunes en la industria son:

- Fabricación inteligente
- Mantenimiento preventivo y predictivo
- Redes eléctricas inteligentes
- Ciudades inteligentes
- Logística conectada e inteligente
- Cadenas de suministro digitales inteligentes

¿Cuáles son las principales aplicaciones de IoT?

La capacidad de IoT para proporcionar información de los sensores, así como para hacer posible la comunicación entre dispositivos da impulso a un amplio conjunto de aplicaciones.

Algunas de las aplicaciones más populares y lo que hacen.

- Crear nuevas eficiencias en fabricación a través de la supervisión de las máquinas y la supervisión de la calidad de los productos.

Las máquinas se pueden supervisar y analizar continuamente para garantizar que funcionan dentro de las tolerancias requeridas. Los productos también se pueden supervisar en tiempo real para identificar y abordar defectos de calidad.

- Mejorar el seguimiento y la “delimitación de activos físicos.

El seguimiento permite a las empresas determinar rápidamente la ubicación de los activos. La delimitación les permite asegurarse de que los activos de alto están protegidos del robo y la extracción.

- Usar wearables para supervisar el análisis de la salud humana y las condiciones ambientales.

Los wearables de IoT hacen que las personas entiendan mejor su propia salud y permiten a los médicos supervisar de forma remota a los pacientes. Esta tecnología también hace posible que las empresas puedan hacer un seguimiento de la salud y la seguridad de sus empleados, lo que es especialmente útil para los empleados que trabajan en condiciones peligrosas.

- Impulsar eficiencias y nuevas posibilidades en los procesos existentes.

Un ejemplo de esto es el uso de IoT para aumentar la eficiencia y la seguridad en la gestión de flotas.

Las empresas pueden usar la supervisión de flotas de IoT para dirigir los camiones en tiempo real con el fin de mejorar la eficiencia.

- Facilitar los cambios de procesos empresariales.

Un ejemplo de ello es el uso de dispositivos de IoT para supervisar el estado de máquinas remotas y activar las llamadas de servicio para el mantenimiento preventivo. La capacidad de supervisar las máquinas de forma remota también hace posible nuevos modelos empresariales de producto como servicio, modelos en los que los clientes ya no tienen la necesidad de comprar un producto, sino que pagan por su uso.

ThingsBoard



Fig. 8 Logo de la plataforma Open Source

ThingsBoard es una plataforma de IoT de código abierto que permite un rápido desarrollo, gestión y escalado de proyectos de IoT. Su objetivo es proporcionar la solución local o en la nube de IoT lista para usar, que habilitará la infraestructura del lado del servidor para las aplicaciones de IoT.

Características

Con ThingsBoard, se logra:

- Provisión de dispositivos, activos y clientes, además de definir relaciones entre ellos.
- Recopilar y visualizar datos de dispositivos activos.
- Analizar la telemetría entrante y activar alarmas con un procesamiento de eventos complejos.
- Controlar dispositivos mediante llamadas a procesamiento remoto (RPC).
- Crear flujos de trabajo basados en el evento del ciclo de vida del dispositivo, el evento de la API REST, la solicitud de RPC, etc.
- Diseñar paneles de control dinámicos y receptivos, presentar información y telemetría de dispositivos activos a sus clientes.
- Habilitar funciones específicas de casos mediante cadenas de reglas personalizables.
- Enviar los datos del dispositivo a otros sistemas.

Arquitectura

ThingsBoard está diseñado para ser:

- Escalable: Plataforma escalable horizontalmente, construida utilizando tecnologías líderes de código abierto.
- Tolerante a fallas: Sin un solo punto de falla, todos los nodos del clúster son idénticos.
- Robusto y eficiente: Un solo nodo de servidor puede manejar decenas o incluso cientos de miles de dispositivos, según el caso de uso. El clúster ThingsBoard puede manejar millones de dispositivos.
- Personalizables: Agregar nuevas funciones es fácil con Widgets personalizables y nodos de motor de reglas.
- Duradero: Nunca pierde los datos.

Mecánica de fluidos

Pérdidas primarias y secundarias en las tuberías.

Las pérdidas de carga en las tuberías son de dos clases: primarias y secundarias.

Las pérdidas primarias son las pérdidas de superficie en el contacto del fluido con la tubería (capa límite), rozamiento de unas capas de fluido con otras (régimen laminar) o de las partículas de fluido entre sí (régimen turbulento). Tienen lugar en flujo uniforme, principalmente en los tramos de tubería de sección constante.

Las pérdidas secundarias son las pérdidas de forma, que tienen lugar en las transiciones (estrechamientos o expansiones de la corriente), codos, válvulas y en toda clase de accesorios de tubería.

Si la conducción es larga (oleoductos, gasoductos...) las pérdidas secundarias tienen poca importancia (de ahí el nombre de pérdidas secundarias), pudiendo a veces desprejarse; o bien se tienen en cuenta al final.

Estos elementos producen una perturbación de la corriente que origina remolinos y desprendimientos, que intensifican las pérdidas.

Estas pérdidas, a pesar de llamarse secundarias, pueden ser más importantes que las primarias, si la conducción es relativamente corta. Se admite generalmente que si la

longitud de la tubería es mayor que 1.000 diámetros el error en que se incurre despreciando las pérdidas secundarias es menor que el error en que se incurre al calcular el valor de λ .

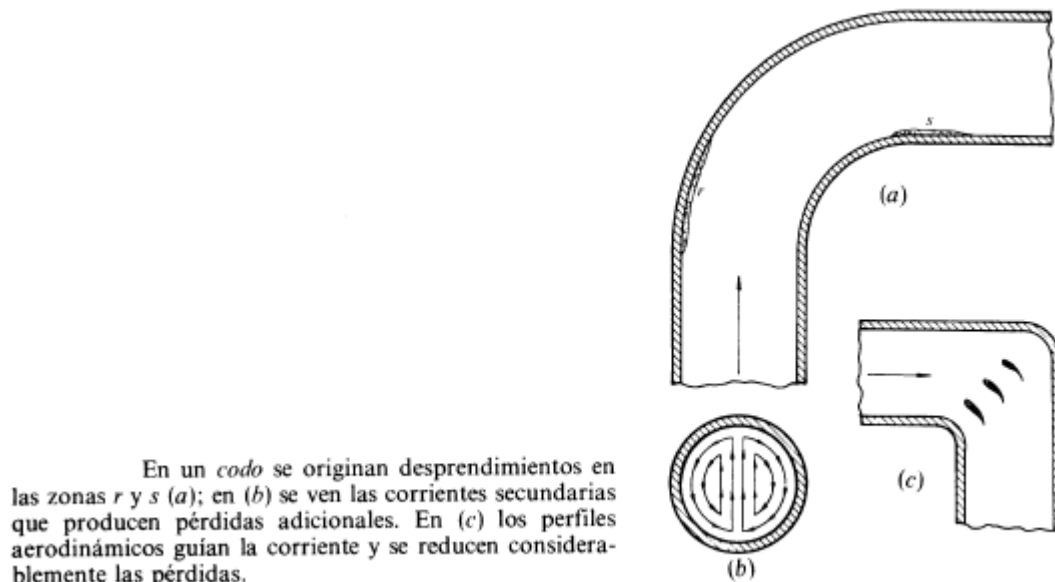


Fig. 9 Extraída del libro de Mecánica de Fluidos Mataix (pérdidas secundarias)

Codos

En el codo que se representa en la Fig. 18 (a) se origina dos tipos de pérdidas:

- Las producidas por la fuerza centrífuga que origina un flujo secundario (Fig. 18 b) que se superpone al flujo principal y que intensifica el rozamiento.
- Las producidas por la separación que se produce en las zonas *r* y *s* (Fig. 18 a). El flujo secundario se evita casi por completo con alabes directrices, cuya forma de perfil aerodinámico se representa en la Fig. 18 c. Esta solución es cara y no se emplea más que en casos especiales.

Lo anterior se menciona por el hecho de considerar las pérdidas de presiones y velocidades en accesorios de las tuberías, para evitar poner los sensores cerca de accesorios que pudiesen afectar el funcionamiento adecuado, se describe de forma muy general, pero si es necesario hacer cálculos de pérdidas en el circuito hidráulico una vez que se hayan montado los sensores, será un tema algo extenso de desarrollar, pero importante y necesario.

Open Know-How



Fig. 10 Logo de la plataforma Open Know-How

Open Know-How es una organización de software abierto que integra un estándar para documentar proyectos técnicos, en dónde se busca que la colaboración entre personas interesadas en un mismo proyecto, sea fácil para expandir el conocimiento, además se busca acelerar la innovación en investigación, diseño y fabricación.

El objetivo del estándar para documentar que usa Open Know-How está enfocado más en la parte técnica, combinando la parte teórica en menor cantidad, los creadores están convencidos que la parte técnica, es decir, la parte práctica resulta más eficiente para aprender más rápido y mejor, en lugar de sólo incluir la parte teórica, los puntos que destacan en el desarrollo son los siguientes:

- Promover el intercambio de conocimientos.
Distribuir diseños de fuentes abiertas, compatibles y de bajo costo para promover la ciencia, inspirar la innovación y salvar vidas.
- Desarrollar estándares abiertos.
Definir estándares y especificaciones abiertos para que el intercambio de conocimientos sea rápido, fácil y accesible para todos.
- Incrementar la adopción.
Inspirar a las personas para que se conviertan en creadores del mundo real y habilitar un entorno próspero de innovación.

Resulta una herramienta útil para personas que buscan documentar proyectos y darles seguimiento con un grupo de colaboradores o simplemente cualquier persona que se interese, el estándar incluye la opción de agregar herramientas, materiales, habilidades de las personas que participan, personas que participan en el proyecto y máquinas o maquinaria.

Lo interesante del estándar de documentación es que ya tiene campos o secciones que sólo hay que llenar con la información necesaria, es portable, tiene algunos formatos en los que se pueden guardar los documentos y fácil de reproducir.

En resumen sobre Open Know-How, fue el elegido y el ideal para documentar los dispositivos electrónicos que se están haciendo en el IER, el único detalle es que los dispositivos están en etapa de desarrollo, pruebas y muchas modificaciones, por lo que sólo se eligieron algunos puntos importantes para empezar a documentar los dispositivos, Open Know-How usa el estándar pensando en dispositivos o proyectos que están terminados y que pueden ser innovados, pero no para los que están en etapa de desarrollo y constantes cambios antes de tener versiones completas. Por lo anterior, sólo se usan puntos muy importantes del estándar.

No sé usa la plataforma de Open Know-How, sólo se usan algunos puntos del estándar y su estructura, la estructura es lo importante a destacar porque toda la documentación sigue una secuencia, desde lo más básico hasta lo último a destacar de los proyectos, la estructura final elegida se almacena en otra plataforma que también se pueden hacer modificaciones durante el desarrollo, se usan dos plataformas, incluso tres, para darle el formato final que se requiere, pero en esta etapa de los dispositivos, es necesario y además pensando en que si alguna persona empieza otro dispositivo desde cero, le resulte amigable y fácil para llevar el orden de la estructura, cuando se domina la estructura elegida y la plataforma final, se pueden reducir esfuerzos y el uso de plataformas hasta terminar con una sola, la plataforma donde se almacenan los dispositivos con la estructura deseada se explica más adelante, se decidió hacerlo así porque es importante que se conozcan los pasos y plataformas usadas antes de llegar al resultado final.

M Editor.md



Fig. 11 Logo del editor de texto plano

M Editor, como su nombre lo indica es un editor de texto, maneja el formato markdown, herramienta de conversión para pasar de textos planos a HTML, es sencillo de usar y muy práctico, fue el elegido para hacer toda la parte escrita, hay diferentes editores de markdown, que todas son iguales pero en diferentes presentaciones, se eligió M Editor porque es muy intuitivo y además para las personas que nunca han documentado en markdown, resulta que la plataforma es muy parecida a Word, en la mayoría de los casos, todos han documentado en Word, por lo que usar M Editor resulta sencillo y en cuestión de horas o minutos, se dominan las herramientas de la plataforma, no se compara con Word, pero es muy parecido en las herramientas básicas y no hace mucho de lo que en Word se puede conseguir.

GitHub



Fig. 12 Logo de la plataforma de alojamiento de los repositorios

GitHub es una plataforma de alojamiento de código para el control de versiones y la colaboración. Le permite al usuario y a otros trabajar juntos en proyectos desde cualquier lugar.

La documentación para un proyecto nuevo puede adaptarse a la estructura de la misma plataforma, es difícil si no se ha trabajado en documentar, estructurar y en general, organizar proyectos, después de trabajarlo por un tiempo, es fácil de usar y resulta interesante lo que se puede hacer en la plataforma.

Entre las cosas a destacar en la plataforma, son:

- Crear y usar un repositorio, desde el lugar que sea e incluso aportar a otros.
- Iniciar y administrar una nueva organización.
- Realizar cambios en un archivo y enviarlos a GitHub como confirmaciones.
- Abrir y fusionar una solicitud de extracción.
- Es seguro, pueden trabajarse repositorios públicos o privados.

Dillinger



Fig. 13 Logo del complemento de editor de texto Dillinger

Dillinger, al igual que M Editor, es un editor de texto plano, la intención de usarlo como complemento, es que se pueden exportar archivos, desde Dillinger hacia otras plataformas e incluso a PDF, eso fue lo más interesante para empezar a usarlo, el procedimiento para usarlo como complemento, es el siguiente, en M Editor se empieza con la escritura en texto plano (Markdown), se optó para que sea la primer plataforma de documentación por su sencillez de entendimiento, es similar a Word, pero muy básico, para que no sea difícil para los compañeros que participaron en los otros dispositivos, después de tener la documentación hecha, se procede a copiar y pegar todo el texto de M Editor a Dillinger, para posteriormente poder guardar el avance en PDF o con extensión .md, que es el que se trabaja en todos los dispositivos.

Desarrollo

Procesos de administración

El proceso de administración de este proyecto fue desarrollado en gran medida por el Dr. Guillermo Ramírez, en cuanto a equipos para la parte práctica y de pruebas en general, suministró los materiales de trabajo, sensores, cables de conexión, microcontroladores y demás materiales requeridos, el Dr. Guillermo Barrios fue de gran ayuda para comprender la lógica de programación de Python, brindó un curso de programación en Micropython que me ayudó a entender mejor gran parte de los dispositivos que se encuentran desarrollando.

Inicio

En el Laboratorio de Tecnologías Abiertas (LaTA) del IER-UNAM se desarrollan diferentes Sistema de Adquisición de Datos (SAD), con la intención de monitorear variables de confort para un edificio sustentable que se está construyendo en el mismo instituto, dichos SAD's son dispositivos, que integran sensores electrónicos, posteriormente los datos recopilados por los sensores son enviados a una plataforma de visualización de datos para analizar el comportamiento de las variables y hasta control de las mismas desde la plataforma, si es necesario o requerido, se pueden hacer muchas cosas más desde la plataforma hacia los dispositivos conectados, desde el lugar en que el usuario se encuentre, esta última característica de la plataforma, es parte de la nueva tecnología que se está desarrollando en los últimos años en todo el mundo, el internet de las cosas o IOT (por sus siglas en inglés), ayudará en gran medida a tomar decisiones de acción sobre los dispositivos conectados en el nuevo edificio del instituto, dicho lo anterior, es necesario documentar todo lo que se va desarrollando para posibles fallas o mejoras en un futuro, será de gran importancia saber dónde y como buscar información de los dispositivos hechos de manera eficiente, es por eso que se busca una forma de documentar todo lo relacionado con los dispositivos y mantenerlos al alcance de los interesados.

Planeación

Descripción de actividades

No.	ACTIVIDAD
1	Curso de programación básica en micropython y uso de Thingsboard (Taller IOT)
2	Revisar que los repositorios estén actualizados y completos
3	Replicar y documentar el primer dispositivo (H2O)
4	Replicar y documentar el segundo dispositivo (Energía eléctrica)
5	Replicar y documentar el tercer dispositivo (THIS)
6	Mantener actualizados los repositorios e integrar una manera de revisar el avance de cada dispositivo (al mismo tiempo que se replican los dispositivos)
7	Utilizar projects de github para administrar los proyectos (al mismo tiempo que se replican los dispositivos)
8	

Tabla 1. Descripción de actividades

Cronograma de actividades (semanal)

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1																								
2																								
3																								
4																								
5																								
6																								
7																								
8																								

Tabla 2. Cronograma de actividades

Cabe mencionar que, durante el desarrollo del proyecto, se fueron elaborando documentos al mismo tiempo que se investigaban, métodos de documentación para proyectos electrónicos, innovación al dispositivo que se planteó al inicio del proyecto, los otros dos dispositivos también eran desarrollados al mismo tiempo por otros compañeros y en la etapa final se aplicaba el formato de documentación elegido en este proyecto, es decir, se desarrolló en su totalidad por otros compañeros y se aplicó lo que se hizo en este proyecto, en la parte final de documentación a los otros dispositivos, cumpliendo con las actividades planteadas en la parte teórica.

Ejecución del proyecto

Antes de proceder con el desarrollo general del proyecto, se revisaron los antecedentes del dispositivo H2O, los documentos desarrollados por otros estudiantes que habían trabajado en el dispositivo, esquemas, diagramas, procesos de operación, entre otras cosas, con la intención de ver lo que se tenía hasta el momento y lo que realmente era necesario complementar para tener al final, un dispositivo funcionando con todo lo necesario para ser replicado de manera eficiente.

Para tener una referencia de lo que se tenía y lo que se debía hacer, se propuso una check-list para llevar un control de lo que se estaba cumpliendo y lo que al final del desarrollo del dispositivo debería entregarse, esto es parte inicial de la propuesta de documentación, se aplicó a los otros dispositivos adaptando lo que era necesario, pero en gran parte debían de cumplir casi todos los dispositivos con lo mismo, se puede observar que al plantear lo que hacía falta o actualizar, era prácticamente todo y cosas que debían empezarse desde cero.

CHECK LIST H2O				Observación 1	Observación 2	Status
El dispositivo cuenta con lista de costos de materiales		Pendientes	1	Falta actualizar		Falta
El dispositivo cuenta con códigos completos		Pendientes	1	Falta para 8 sensores	Para 8 sensores por otro compañero	Falta
Tiene diagrama de conexión (Pictograma)		Pendientes	1	Falta actualizar		Falta
Cuenta con diagrama eléctrico (esquemático)		Pendientes	1	No tiene		Falta
Cuenta con diseño PCB		Pendientes	1	No tiene		Falta
Tiene archivos de PCB (gerber)		Pendientes	1	No tiene		Falta
Cuenta con lista de materiales		Pendientes	1	Falta actualizar		Falta
El dispositivo cuenta con planos CAD de diseño		Pendientes	1	No tiene		Falta
Cuenta con archivos CAD		Pendientes	1	No tiene		Falta
Cuenta con archivos STL para impresión		Pendientes	1	No tiene		Falta
Cuenta con un manual de operación reducido		Pendientes	1	Falta actualizar		Falta
Fichas técnicas de los componentes usados		Pendientes	1	No tiene		Falta
Cuenta con manual de ensamblado		Pendientes	1	No tiene		Falta
Documento de errores y soluciones		Pendientes	1	No tiene		Falta

Tabla 3. Check-list del dispositivo H2O

Con lo que se tenía hasta el momento y después de leer la documentación que se tenía, se realizaron pruebas, para comprender en su totalidad el funcionamiento y poder decidir de que manera se procedería con los archivos faltantes para tener completo el dispositivo como un producto final.

Diseño del diagrama esquemático y de la placa de circuito impreso (PCB)

En primer lugar, se decidió que era necesario tener un PCB general del dispositivo H2O con conexión hasta para 8 sensores, hasta antes de empezar con el diseño, se habían hecho pruebas en la placa de pruebas (protoboard), se investigaron estándares de diseño, se implementaron algunas características de la norma IPC 2221B para diseño de PCB, cabe destacar que no se aplicó toda la norma, se tomaron algunas características importantes como por ejemplo, el ancho de pista de 0.25 para corrientes inferiores a 1A, las pistas sin esquinas de 90° se evitaron lo más que se pudo, es decir, todas las pistas deben tener 30° mínimo en los cambios de dirección, lo anterior, es para facilitar el maquinado del PCB, entre otras cosas, se decidió que no se usarían componentes de superficie, debido a que implican el uso de más normas y cuidados al momento del ensamble, no era necesario para este diseño, por el cruce de pistas en una sola capa, se decidió que sería de doble capa para evitar cualquier detalle de choque entre pistas o el espacio en una sola capa.

El software usado para el diseño en primera instancia fue el de Proteus en versión de estudiante porque ya se había trabajado anteriormente en este software y es muy completo para el diseño de esquemáticos, simulaciones electrónicas y diseño de PCB.

Se eligieron los materiales adecuados y los componentes que el circuito debía tener y se procedió con el diagrama esquemático (Anexo 1), otra cosa importante que es necesario mencionar es que se buscó usar software libre para todo lo que se realizó en el proyecto, con la intención de no tener que pagar licencias y cuando se requiera editar y hacer cualquier modificación, no tengan que solicitar licencias, además es la filosofía del IER-UNAM, no todos los software usados fueron libres, porque en algunos casos no se pueden llevar al nivel que se requiere los diseños usando software libre.

Posteriormente se presentó el diagrama esquemático en el software libre de diseño PCB, EasyEDA, en este último se realizaron los archivos finales del PCB y fueron los que se presentaron.

El diagrama esquemático realizado en Proteus fue de ayuda para simplemente replicarlo en EasyEDA y se guardó en formato PDF (Anexo 2) para mejor visualización y también se entregó el archivo de diseño con extensión (.json), extensión que usa el software.

Las figuras que se presentan a continuación son de simple visualización porque tienen mayor presentación en Proteus que en EasyEDA.

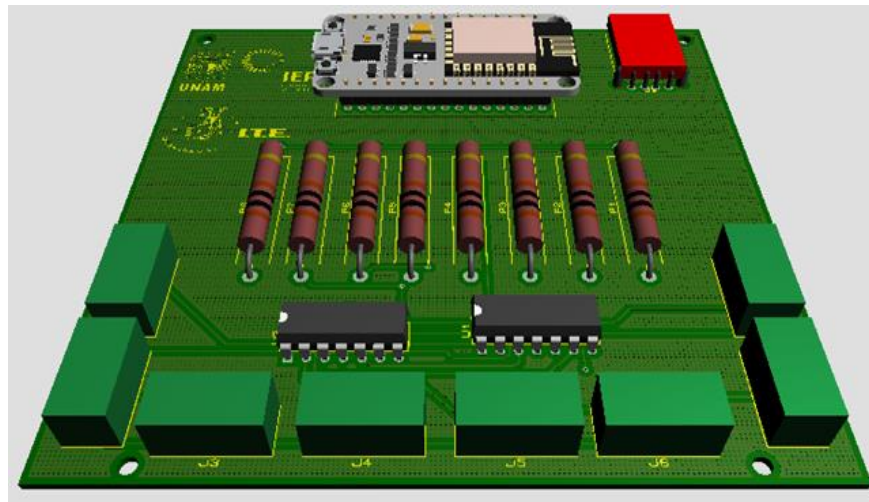


Fig. 14 Presentación 3D del diseño PCB en el software de Proteus

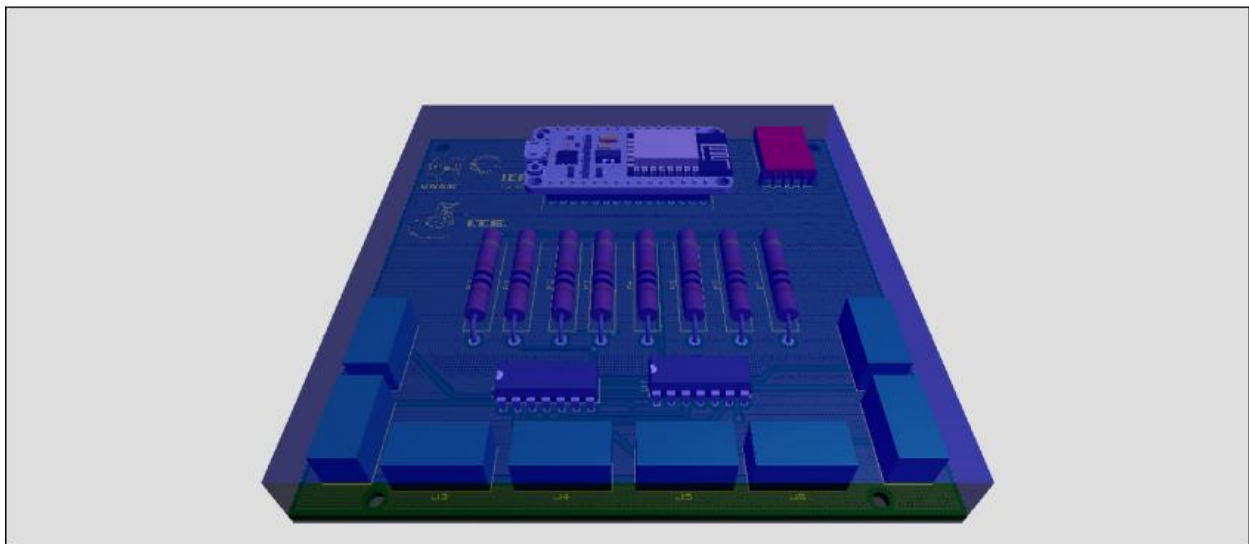


Fig. 15 Presentación 3D con altura del diseño PCB en el software de Proteus

Las dimensiones en el diseño del PCB se presentan en Proteus, porque hay más herramientas en este software para poner medidas, se pueden ver en el diseño del PCB las pistas y los nombres que identifican a cada componente del PCB, el diseño con las medidas se presenta en el Anexo 3, se le realizaron perforaciones en las esquinas para poder asegurarlo en una carcasa.

También se generó el archivo gerber para maquinado del PCB en caso de que sea necesario o también se puede enviar a que lo maquinen desde China, desde el software de EasyEDA se pueden enviar a maquinar con las especificaciones que deseamos y recomendaciones estándares que el software nos brinda.

Otra cosa que se probó antes del diseño final del PCB, fue un divisor de voltaje, los microcontroladores ESP sólo pueden recibir en sus pines 3.3V, por lo que la alimentación externa, si se toma de una computadora, será de 5V, muchos sitios en la web, dicen que se puede alimentar con 5V, pero para no arriesgarse, se diseñó el divisor de voltaje y se probó, cuando funcionó correctamente y se comprobó que realmente reducía el voltaje de 5V a 3V, entonces se pusieron las resistencias adecuadas del divisor de voltaje en el diseño de PCB, para que pueda ser alimentado de forma independiente el microcontrolador sin problemas, se simuló y se realizó en el software TinkerCad.

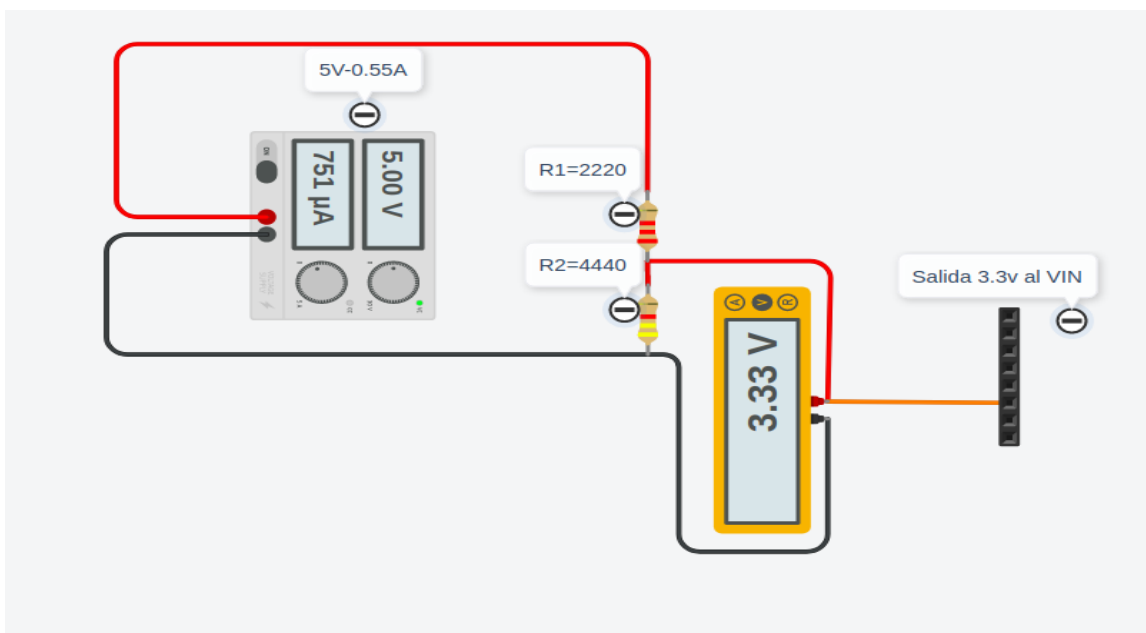


Fig. 16 Diseño del divisor de voltaje de 5V a 3.3V para el PCB

Materiales

Después de haber realizado el diseño del PCB y definir bien los materiales que deben usarse, se recopilaron los nombres comerciales para identificar bien lo que debe llevar el PCB, cabe mencionar que, en las figuras de visualización, por ser de presentación, se exageran un poco el tamaño de los componentes y hasta en colores, no son las que realmente lleva el PCB.

Cantidad	Referencias	Valor	Código almacén
2 Integrados	U1-U2	LM324	
8 Resistencias	R1-R8	1K Ω	W1K
2 Resistencias	R16-R19-R20	2K Ω	W1K
2 Resistencias	R15-R17-R18	220 Ω	W1K
8	J1-J8	TBLOCK-M3	
1	J9	USBCONN	
1	NMCU2	NODEMCU V3	
8	S1-S8	YF-S201	

Tabla 4. Lista de materiales del dispositivo H2O

Diseño CAD

Para el diseño CAD se usó el software SolidWorks por mayor conocimiento del mismo y por las grandes ventajas comparado con los software de domino libre, se decidió que el diseño estaría conformado por tres partes, después de ver algunos ejemplos de carcasas que se han utilizado en placas electrónicas, se decidió que los diseños usados en los kit de Arduino, Raspberry, entre otras, son similares en cuanto a estructura, por lo que el diseño fue desarrollado en dimensiones más grandes que los conocidos y sobre todo con las dimensiones del diseño del PCB, se imprimirá en impresora 3D, usando PLA, que por lo regular es el material que más se usa para impresión 3D.

El diseño de la carcasa está constituido por tres partes:

Base de la carcasa

La base de la carcasa fue diseñado con soportes en las esquinas para que el PCB pueda asegurarse con tornillos, tiene ranuras horizontales para ventilación, aunque no se calienta mucho el microcontrolador, es conveniente que circule el aire, tiene unos nervios pequeños en 3 laterales para que sea ensamblado con la otra parte de la carcasa, también en el diseño , sobre las paredes, en la parte superior tiene cortes, con la intención que formen huecos para las salidas de cables y por último se le hicieron ranuras para asegurarse con tornillos a la pared o cualquier otra superficie.

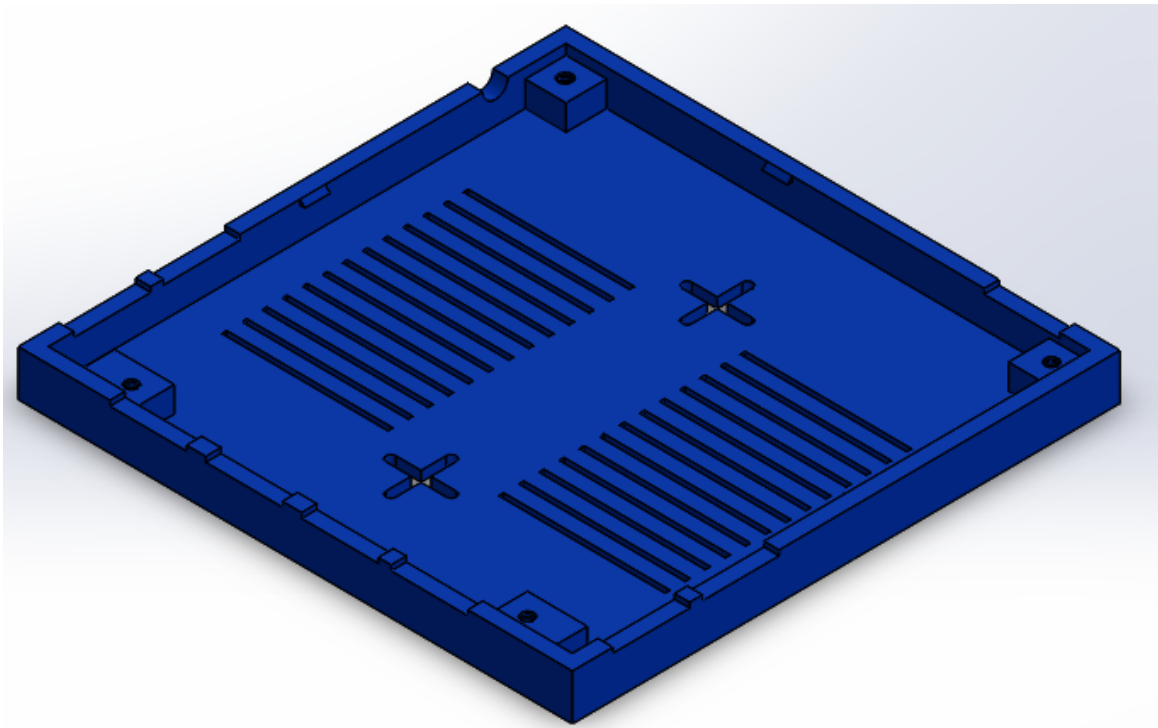


Fig. 17 Base de la carcasa del dispositivo H2O

Base media de la carcasa

La base media de la carcasa tiene los cortes en las paredes laterales, de igual forma es parte de la salida de cables y en las paredes internas tiene unas ranuras que sobresalen para el ensamble con la base de la carcasa, en las paredes laterales en la parte inferior tienen igual, cortes en tres paredes para que sea ensamblado con la tapa de la carcasa.

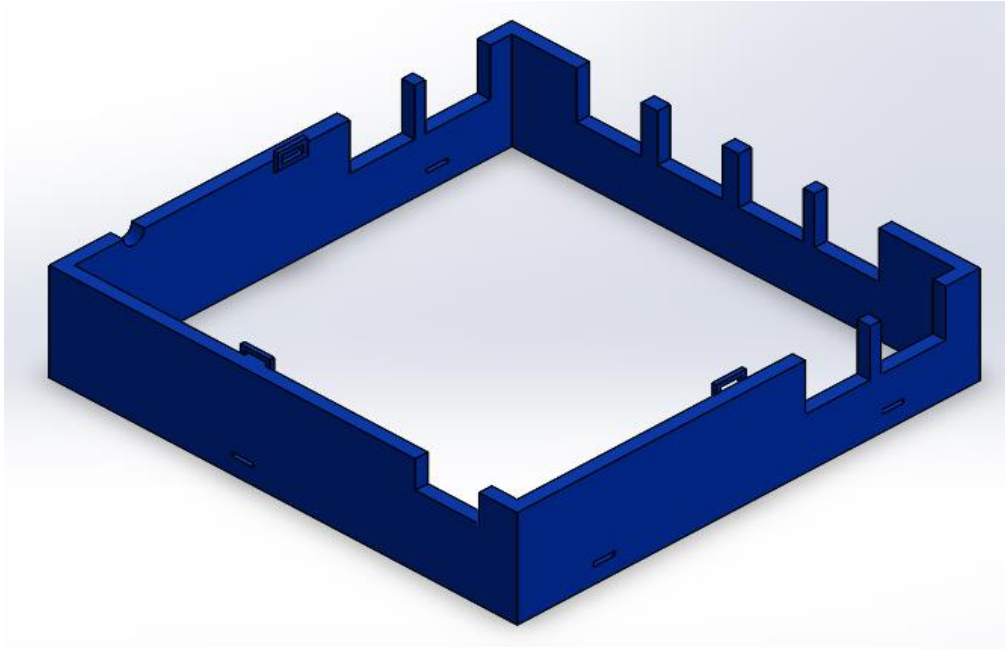


Fig. 18 Base media de la carcasa del dispositivo H2O

Tapa de la carcasa

En la tapa de la carcasa, al igual que en la base, se hicieron ranuras para que circule el aire de un lado a otro, esta parte de la carcasa si es plana, porque no tiene contacto de otro tipo con la base media de la carcasa, lo único que si tiene son los huecos, con los nervios pequeños para que se ensamble a presión con la otra parte.

Los planos del diseño completo de la carcasa están en el Anexo 4.

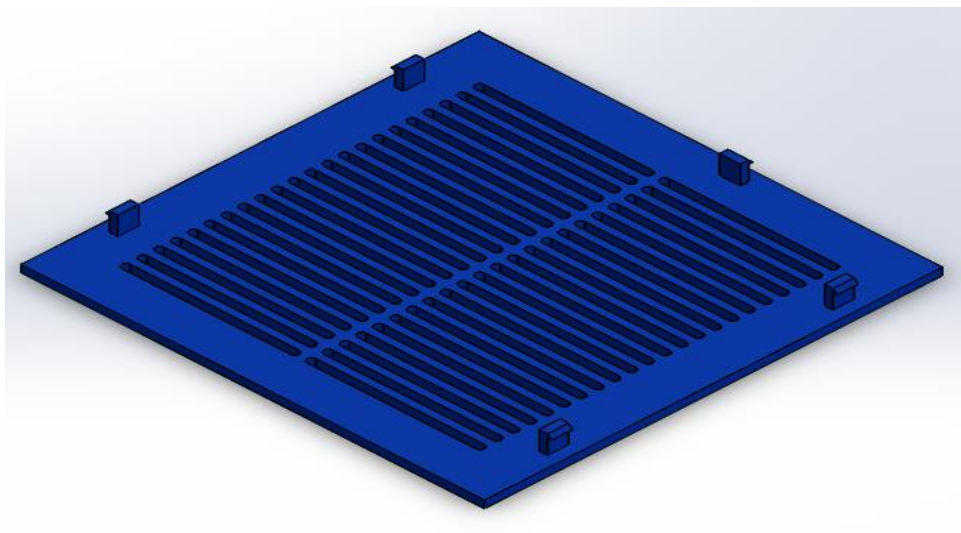


Fig. 19 Tapa de la carcasa del dispositivo H2O

Manual de operación

Después de haber trabajado el diseño de la carcasa, el diseño del PCB y una lista de materiales, se empezaron a realizar pruebas de funcionamiento, ya se tenía un manual de operación, trabajado anteriormente por un compañero, entonces en esta parte del proyecto, lo que se hizo, fue actualizar y comprobar que el manual realmente funcionaba para operar sin problemas el dispositivo H2O.

Para conexión del microcontrolador en las pruebas, se realizó un divisor de voltaje, con la intención de llevar a la práctica lo que se había diseñado en el PCB pero en una placa de pruebas, se tenía que calibrar el sensor de caudal YF-S201, con la intención tener un litro completo, se usó una botella de 1L, la intención era que la cantidad de 1L que se desocupaba en un pequeño banco de pruebas, se reflejara en la misma cantidad en la terminal en Linux, al ejecutar el código, si no se acercaba al litro desocupado, entonces se tenía que ir cambiando la constante de calibración en el código, hasta que en la terminal se imprimiera el litro o lo más cercano al litro. El pequeño banco de pruebas se usó hasta que se logró calibrar el sensor de caudal YF-S201 y hasta que se probaron otras funciones de operación, fue de gran ayuda montarlo y hacer el diagrama antes.

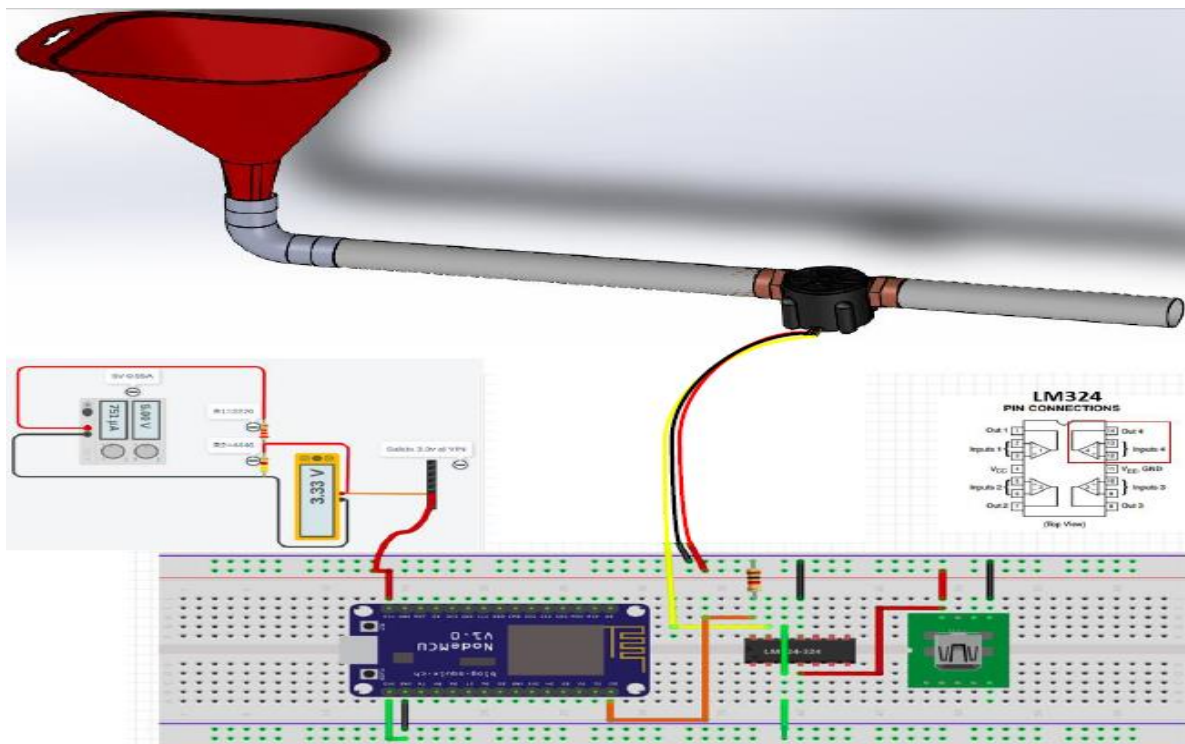


Fig. 20 Banco de pruebas para el dispositivo H2O

Se usó el manual previamente hecho por otros compañeros, para operar el dispositivo y en casos donde se vio la necesidad de actualizar o ver que no funcionaba como estaba planteado, se actualizó y se fueron realizando pruebas para quedar todo documentado, es diferente para cada usuario, porque al momento de operar el dispositivo, cada usuario entiende mejor una parte que otra, pero se busca que en las partes de operación del dispositivo donde parece difícil de entender para la mayoría de los usuarios, sea claro y preciso sobre lo que se desea transmitir al usuario final.

Todo lo necesario para poder operar el dispositivo está en el manual de operación, almacenado en el repositorio del dispositivo H2O.

En cuanto al manual de operación, realmente no se hicieron muchas actualizaciones, sólo se actualizó en la parte de como cargar el nuevo firmware en el dispositivo y otras cosas que eran importantes actualizarlas, los comandos usados en la terminal de Linux también fueron modificados a la versión de Python 3.

Manual de montaje

El manual de ensamble si se trabajó todo, porque no había uno, cabe mencionar que en el manual se usan figuras representativas para el paso a paso del ensamble, porque no se imprimió la carcasa, debido a los tiempos actuales de salud en el que estamos, en el IER-UNAM tienen impresora 3D, pero no se puede entrar, a menos que sea autorizado y necesario ir al instituto, por lo anterior, el manual sólo es ilustrativo y todo con figuras elaboradas, no son fotos ensamblando el dispositivo pero en todo momento se busca que sea lo más parecido a las partes físicas.

En el manual de ensamble no se siguió un estándar, pero se empezó a explicar como se armaría desde cero el dispositivo, la placa sujeta a la base de la carcasa es con tornillos M2.5x0.45 y el posicionamiento adecuado de los componentes en la placa, se describe en el pequeño manual que está en el repositorio del dispositivo.

También se explica algo breve sobre la soldadura de los componentes, corte de pines y finalmente unas recomendaciones.

Documentación en repositorios

Para poner la documentación en Github, fue necesario crear un repositorio para cada dispositivo, en la propuesta final para documentar los dispositivos electrónicos se hizo una recopilación de estándares en la web para proyectos electrónicos, el elegido fue el estándar de documentación de Open Know-How, es un estándar que tiene muchos campos a llenar, fácil de usar y orden adecuado, la plataforma es intuitiva y un poco gráfica, lo suficiente para poder entender que sigue en cada campo a llenar, es importante destacar que el estándar Open Know-How se usa cuando los proyectos están terminados y no cuando están en desarrollo, no quiere decir que no se pueda usar cuando los proyectos están en desarrollo y constantes modificaciones, pero está pensado para cuando están terminados y pueden ser publicados para posibles mejoras por otros usuarios en la misma plataforma.

Se decidió que sería el estándar usado porque está probado y es muy fácil de entender, la decisión fue que se usarían los puntos más importantes del estándar y serían usados en la estructura de documentación de los dispositivos, debido a que los dispositivos están en constante desarrollo y sufren muchas modificaciones.

Cuando se eligió la estructura final basado en el estándar Open Know-How, se procedió a llenar dicha estructura en formato de texto plano, Markdown es fácil de editar y no es complicado de usar, sólo hay que entender unas líneas de comando y serán usados en toda la documentación, no se realizó en Word porque es necesario abrir los archivos desde Windows y en casi todo el proyecto se usó el sistema operativo Linux, por lo que Word era difícil usarlo constantemente, por eso se optó por texto plano (Markdown) y se pueden editar desde plataformas en la online.

Para empezar a escribir texto plano, se usó la plataforma M Editor, porque es muy fácil de entender y resulta intuitivo, es como Word pero demasiado básico, la plataforma elegida no tiene para guardar los archivos en ningún formato, por lo anterior, se buscó otro editor de texto plano, Dillinger es un poco más difícil de entender para personas que nunca han usado Markdown, primero se hace el texto en M Editor y después se copia todo el texto y se pega en la plataforma de Dillinger, en Dillinger si se pueden exportar los archivos hacia otras plataformas de almacenamiento en la nube y hasta en PDF, por eso se eligieron los dos como complementos, cuando se terminan las estructuras de

texto en Markdown, son guardadas desde Dillinger con extensión (.md), posteriormente se pueden subir a las carpetas de los repositorios.

En Github es necesario subir carpetas desde la terminal de Linux o en Windows usando una terminal virtual, existe un repositorio que también se elaboró con la intención de que a otros compañeros les ayude a subir los archivos trabajados de sus respectivos proyectos en Github, para almacenar los repositorios no fue necesario investigar otras opciones porque Github ya es usado desde hace un buen tiempo para almacenar la documentación, se pueden hacer colaboraciones o simplemente trabajarlo en modo privado, todos los repositorios en donde están ubicados los dispositivos están en modo privado, porque es trabajo que está en desarrollo por el IER-UNAM y no se puede correr el riesgo de que sea replicado por otras personas y al no estar completo, lleguen a usar el dispositivo de manera incorrecta y los responsables en gran parte por cualquier accidente con el dispositivo, sería el IER o las personas que se encargaron de liberar el dispositivo como producto final, hasta que se tengan al 100% todos los dispositivos, podrán ponerse los repositorios como “públicos” y con acceso a quien se interese usarlo. Cuando ya se tiene todo estructurado en archivos con extensión md y se tienen los archivos de PCB, CAD, planos en pdf y otros, se procede a subir los archivos correspondientes a cada carpeta en el repositorio del dispositivo H2O y el texto de presentación del dispositivo (README), se deja al final para poder crear los vínculo a cada carpeta y a cada archivo si es necesario.

Resultados

Diagrama esquemático y PCB del dispositivo H2O

El diseño final del diagrama esquemático se realizó en el software EasyEDA, previamente se tenía ya hecho en el software de Proteus, pero al no ser software libre, se tuvo que cambiar a EasyEDA.

El diagrama esquemático está en el Anexo 1, a continuación, se presentan figuras del PCB final, diseñado en EasyEDA con las medidas previamente establecidas en Proteus, ver Anexo 3.

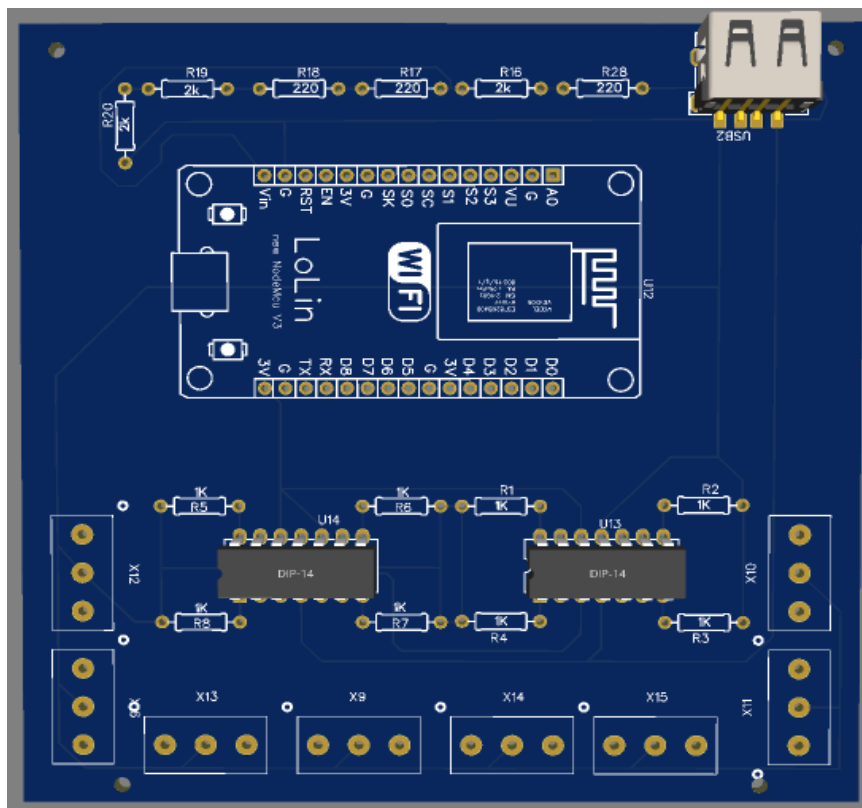


Fig. 21 Diseño de PCB del dispositivo H2O (vista superior)

En la Fig. 21 se puede ver que en el diseño se tienen espacios para 8 conectores TBLOCK-M3, son conectores para 8 sensores de caudal YF-S201, con alimentación general por USB hembra, en la parte superior de la Fig. 21 se pueden ver el arreglo de resistencias, que hacen el divisor de voltaje para energizar el microcontrolador ESP, desde los 5V de la alimentación general USB a los 3V que puede soportar el ESP.

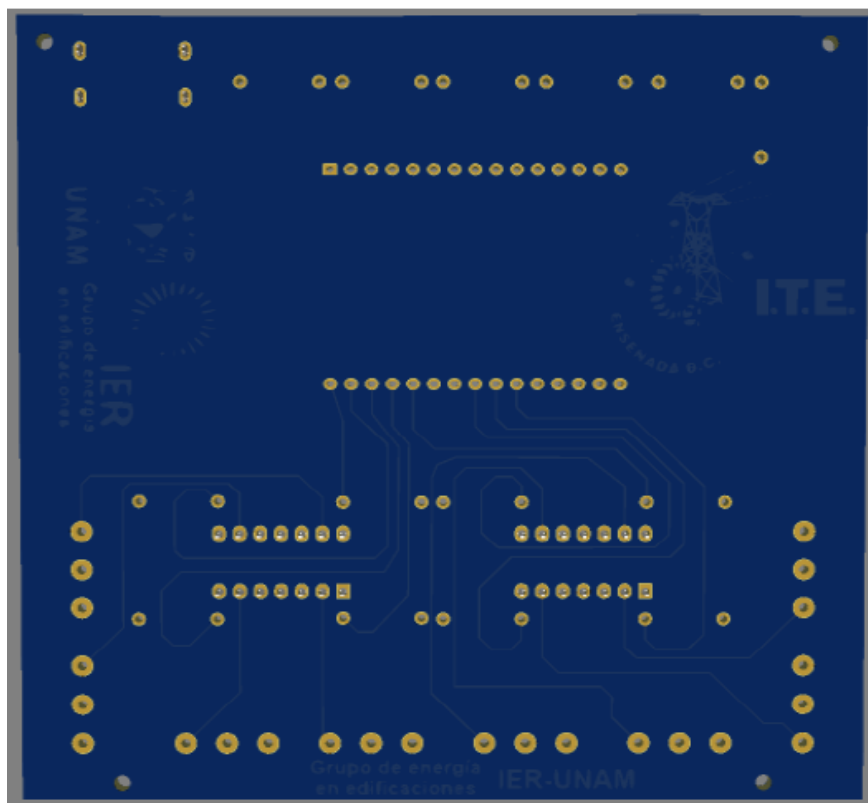


Fig. 22 Diseño de PCB del dispositivo H2O (vista inferior)

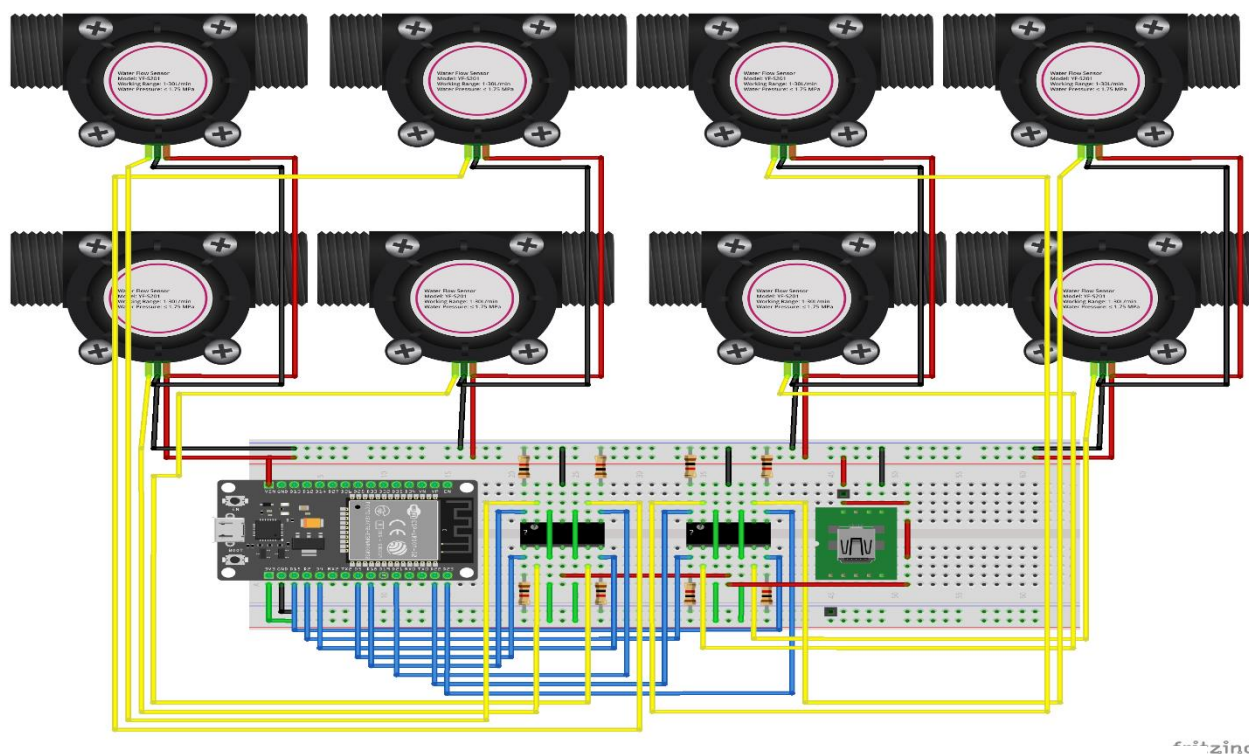


Fig. 23 Diagrama de conexión (pictograma) del dispositivo H2O para 8 sensores

Los archivos de diseño del PCB y el diagrama esquemático se encuentran en el repositorio del dispositivo H2O, en la carpeta de “Diagramas”, el link de esa carpeta en el repositorio es el siguiente:

Carpeta de Diagramas

<https://github.com/jwilliamsee/SensorH2OResidencia/tree/main/Diagramas>

En esa carpeta se encuentran los archivos finales de PCB:

- Dispositivo H2O Esquemático#.json (archivo editable desde EasyEDA)
- Dispositivo_H2O_EsquemáticoA.pdf (el mismo del Anexo 2)
- Gerber_PCB_H2O.zip (archivos para maquinar el PCB)
- SensorH2O-PCB.fzz (diseño de PCB en Fritzing)
- SensorH2O.fzz (archivo de conexiones en Fritzing)

Lista de materiales y costos para un dispositivo

Dispositivo	Material	Cantidad	Precio unitario	Precio total
Sensor H2O	Integrado LM324	1	12	12
	Resistencia de 1K 1/4 W	1	1.5	1.5
	Conector TBLOCK-M3 3 pines	1	4.65	4.65
	Conector USB Hembra	1	15.73	15.73
	ESP8266-NodeMCU	1	115	115
	Sensor de caudal YF-S201	1	130	130
	Gabinete 17.5x13x6	1	120	120
	Glándulas	3	3	9
	Header 15 pines hembra 2.54mm	2	12	24
	PCB 10x17	1	48	48
	Cargador 5v 2A / Micro- USB	1	90	90
			Total	569.88

Tabla 5. Lista de materiales y costos del dispositivo H2O (con 1 sensor)

Diseño CAD final

El diseño CAD de la carcasa del dispositivo H2O final, es el que se muestra a continuación, cabe mencionar que, para presentación final del dispositivo como producto terminado, sí se hará la impresión 3D, para replicar otros dispositivos y los que estarán funcionando, se comprarán gabinetes estándares para ahorrarse tiempo y dinero.

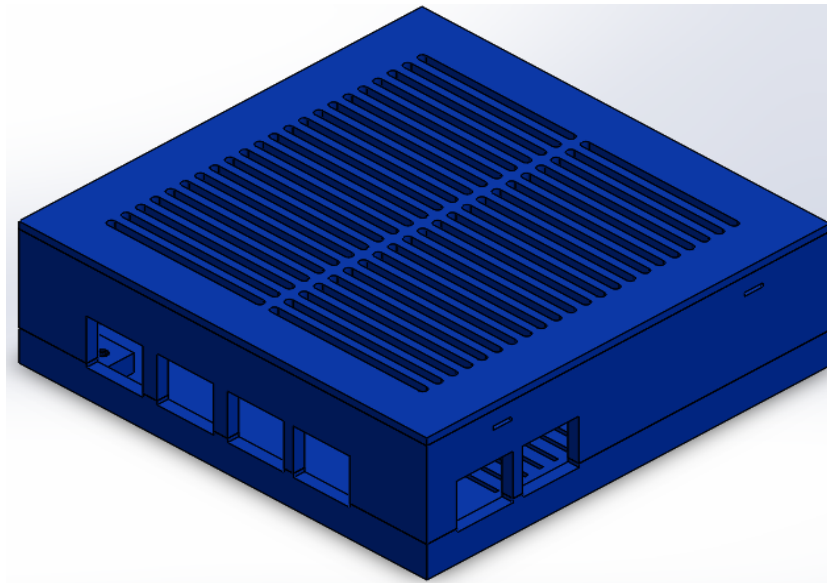


Fig. 24 Diseño CAD de la carcasa para el dispositivo H2O (8 sensores)

Los archivos de diseño CAD, están en el repositorio H2O, en la carpeta “CAD”, a continuación, se encuentra el link de esa carpeta en el repositorio. Ver Anexo 4.

Carpeta CAD

<https://github.com/jwilliamsee/SensorH2OResidencia/tree/main/CAD>

En esa carpeta se encuentran los siguientes archivos:

- BASE_CARCARSA_H2O, BASE_MEDIA_H2O y TAPA_H2O (extensión formato general STL para imprimir en 3D).
- BASE_CARCARSA_H2O_CAD, BASE_MEDIA_H2O_CAD y TAPA_H2O_CAD (extensión STEP para poder visualizarlos en otro software CAD).
- BASE_CARCARSA_H2O_PLANO, BASE_MEDIA_H2O_PLANO y TAPA_H2O_PLANO (formatos PDF para poder imprimir los planos de diseño).

Manual de operación final

En el manual de operación se actualizaron las imágenes de los pasos a seguir, se actualizó el firmware a usar, el más reciente, se explicó lo que es necesario hacer con el microcontrolador ESP, cuando se cargan los códigos por primera vez y justo se muestra la forma correcta de calibrar el sensor de caudal YF-S201 con medidas establecidas de 1L, 600ml y 2L para corroborar la calibración, los resultados de las pruebas de calibración fueran las siguientes:

Con K=0.003125 y según 1L			
Prueba 1.0	1.8L	Promedio 7.12/4=	1.78L
Publicando datos del sensor 1 {"Litros 1": 1.8}			
Prueba 1.1	1.76L		
Publicando datos del sensor 1 {"Litros 1": 1.76875}			
Prueba 1.2	1.80L		
/consumo= 1.80312			
Prueba 1.3	1.76L		
Publicando datos del sensor 1 {"Litros 1": 1.76562}			
Quiere decir que con K=0.003125 mide 1.78L Haciendo regla de 3 Si con K=0.003125 → 1.78L, ¿Cuánto vale K para 1L?			
K= 0.003125		→ 1.78L	
0.00175		← 1L	
Con K= 0.00175 y 1L			
Prueba 2.0	1.030	Promedio 4.028/4=	1.0071L
Publicando datos del sensor 1 {"Litros 1": 1.03075}			
Prueba 2.1	0.974		
Publicando datos del sensor 1 {"Litros 1": 0.97475}			
Prueba 2.2	1.023		
Publicando datos del sensor 1 {"Litros 1": 1.02375}			
Prueba 2.3	0.999		
Publicando datos del sensor 1 {"Litros 1": 0.99925}			
Quiere decir que con K=0.00175 mide 1.007L Haciendo regla de 3 Si con K=0.00175 → 1.007L, ¿Cuánto vale K para 1L?			
K= 0.00175		→ 1.007L	
0.0017376		← 1L	

Tabla 6. Primeras pruebas de calibración del sensor de caudal YF-S201

En las primeras pruebas con la constante $K = 0.003125$ previamente usada en el código principal del dispositivo, se obtuvieron resultados de 1.78L, es decir, la constante K era un valor alto, para corregirlo se hizo una regla de tres y obtener un nuevo valor de K .

En la segunda prueba de calibración, ahora con un valor de $K = 0.00175$, se observó que medía 1.007L, esta prueba es muy exacta, pero se hicieron un par de pruebas más.

Con K=0.0017376 y 1L			
Prueba 3.0	0.985L	Promedio 4.031/4=	1.0078L
Publicando datos del sensor 1 {"Litros 1": 0.985219}			
Prueba 3.1	1.007L		
Publicando datos del sensor 1 {"Litros 1": 1.00781}			
Prueba 3.2	1.018L		
Publicando datos del sensor 1 {"Litros 1": 1.01823}			
Prueba 3.3	1.019L		
Publicando datos del sensor 1 {"Litros 1": 1.01997}			
Quiere decir que con K=0.0017376 mide 1.0078L Haciendo regla de 3 Si con K=0.0017376 → 1.0078L, ¿Cuánto vale K para 1L?			
K= 0.0017376		→ 1.0078L	
0.0017241		← 1L	
Con K= 0.0017241 y 1L			
Prueba 4.0	0.9792	Promedio 4.029/4=	1.0073L
Publicando datos del sensor 1 {"Litros 1": 0.979289}			
Prueba 4.1	1.03963		
Publicando datos del sensor 1 {"Litros 1": 1.03963}			
Prueba 4.2	0.9896		
Publicando datos del sensor 1 {"Litros 1": 0.989633}			
Prueba 4.3	1.0206		
Publicando datos del sensor 1 {"Litros 1": 1.02067}			

Tabla 7. Segunda prueba de calibración del sensor de caudal YF-S201

El resultado final, se muestra en la siguiente tabla, con el valor de $K = 0.001738$ y se probó con 1L, 2L y 600ml, en el manual se explican otros detalles a considerar.

Prueba última con $K = 0.0017380$ y corroboración con 2L y 600ml			
Pruebas finales	Con 1L	Con 2L	Con 600ml
1	Publicando datos del sensor 1 {"Litros 1": 0.99935}	Publicando datos del sensor 1 {"Litros 1": 2.11862}	Publicando datos del sensor 1 {"Litros 1": 0.618728}
2	Publicando datos del sensor 1 {"Litros 1": 1.03759}	Publicando datos del sensor 1 {"Litros 1": 2.03346}	Publicando datos del sensor 1 {"Litros 1": 0.641322}
3	Publicando datos del sensor 1 {"Litros 1": 0.97475}	Publicando datos del sensor 1 {"Litros 1": 2.12731}	
4	1.03759L		

Tabla 8. Prueba final de calibración del sensor de caudal YF-S201

En la carpeta “Manuales” del repositorio H2O, se encuentra este manual de operación.

Link del manual de operación en el repositorio:

<https://github.com/jwilliamsee/SensorH2OResidencia/blob/main/Manuales/ManualDeOperacion.md>

Manual de montaje final

El manual de montaje se detalla gráficamente como deben ensamblarse los componentes en la placa, orientación de los mismos, cortes de pines, ensamble de la carcasa y por último una vista general del dispositivo H2O ensamblado con hasta 8 sensores YF-S201, este último montaje estará en revisión, se comprobó que miden simultáneamente 4 sensores, pero mientras se van agregando sensores, la lectura entre ellos, no es la misma en todos, lo anterior, debido a que son secuenciales, es decir, leen los códigos secuencialmente y si en el proceso de lectura, existen líneas de código que impriman en la terminal un resultado, entonces cada vez, se va sumando milisegundos de retardo entre lecturas, por ejemplo, la lectura de lo que mida el primer sensor no será la misma de lo que mida el octavo sensor conectado, es por eso, que se harán pruebas, el dispositivo puede medir los 8 sensores, pero si se requiere medidas exactas para análisis, entonces hay que considerar ponerlos independientes.

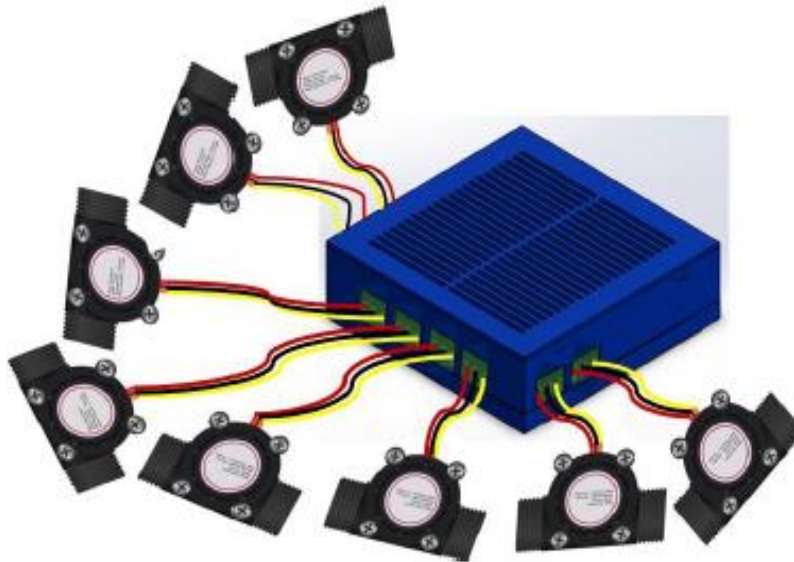


Fig. 25 Ensamble completo del dispositivo H2O (8 sensores)

El manual de montaje se encuentra en el repositorio del dispositivo H2O.

Link del manual de montaje en el repositorio:

<https://github.com/jwilliamsee/SensorH2OResidencia/blob/main/Manuales/Manual%20de%20Montaje.md>

Errores y soluciones

El documento de errores, soluciones y recomendaciones no es muy extenso, los errores y dudas que se presenten, durante la operación del dispositivo, serán diferentes, según el usuario, pero de manera general, se describen algunos, en cada manual, en el de operación y el de montaje, existen soluciones a posibles dudas, por lo que, se espera que no hayan pasado desapercibido las dudas más comunes.

En la misma carpeta de “Manuales”, se encuentra el documento de errores y soluciones.

Link del documento en el repositorio:

<https://github.com/jwilliamsee/SensorH2OResidencia/blob/main/Manuales/ErroresAndSoluciones.md>

Entregables

Los archivos que se entregaron como parte de la documentación del repositorio, fueron los que se presentan en la siguiente check-list, en la Tabla 3 se mostraron los archivos que debían trabajarse o entregarse al iniciar el proyecto.

CHECK LIST H2O				Observación 1	Observación 2	Status
El dispositivo cuenta con lista de costos de materiales		Pendientes	0	Unitario y total		Listo
El dispositivo cuenta con códigos completos		Pendientes	0		Para 8 sensores por otro compañero	Listo
Tiene diagrama de conexión (Pictograma)		Pendientes	0	Tiene para 1,2 y 8 sensores		Listo
Cuenta con diagrama eléctrico (esquemático)		Pendientes	0		De 1 y 8 sensores	Listo
Cuenta con diseño PCB		Pendientes	0		De 8 sensores	Listo
Tiene archivos de PCB (gerber)		Pendientes	0	De 8 sensores	De 8 sensores	Listo
Cuenta con lista de materiales		Pendientes	0	De 1 y 8 sensores		Listo
El dispositivo cuenta con planos CAD de diseño		Pendientes	0	Planos en PDF		Listo
Cuenta con archivos CAD		Pendientes	1	Está en SOLIDWORKS	Falta en FreeCAD	Falta
Cuenta con archivos STL para impresión		Pendientes	0			Listo
Cuenta con un manual de operación reducido		Pendientes	0			Listo
Fichas técnicas de los componentes usados		Pendientes	0			Listo
Cuenta con manual de ensamblado		Pendientes	0			Listo
Documento de errores y soluciones		Pendientes	0			Listo

Tabla 9. Check-list final del dispositivo H2O

En la Tabla 9 se muestra la check-list final, se puede ver que, en los pendientes, hay uno, el diseño de la carcasa se realizó en SolidWorks por lo que se comentó anteriormente, es profesional, es cierto que se requiere una licencia para usarlo, pero el software libre recomendado FreeCAD, aunque si cumple con lo necesario, considero que es más eficiente en cuanto a visualización de gráficos, para el manual fue necesario.

El repositorio terminado, lo que estaba planteado al iniciar el proyecto, se logró terminarlo con el estándar elegido y se aplicaron muchos puntos que se consideraron los más importantes para la etapa de documentación en la que se encuentra el dispositivo, esa estructura extraída de Open Know-How, se adapta a todos los dispositivos que se están desarrollando en el IER, se pretendía que fuera fácil de editar, que se pusiera documentación necesaria y puntos a destacar del dispositivo, con la estructura elegida, se logró tener todos los dispositivos que se estaban trabajando en un repositorio en la plataforma de Github.

Los otros dos dispositivos que se tenían planteados documentar, fueron desarrollados por otros compañeros al mismo tiempo que se trabajaba en la estructura general de documentación, por lo que, sí se implementó la estructura planteada en esos dispositivos y en otros más que también se estaban desarrollando al mismo tiempo.

Se desarrollaron también, videos de como usar la terminal virtual en Windows o la terminal en Linux para subir carpetas a Github y poder hacer los repositorios de los dispositivos en los que estaban trabajando, un video mostrando el proceso de crear un repositorio y un repositorio donde se explica el proceso de documentación con la estructura planteada, usando las plataformas que se usaron en este proyecto, en general, se brindó asesoría a los otros compañeros para poder aplicar la estructura de documentación a sus proyectos y tenerlos en repositorios.

El repositorio del dispositivo H2O, se encuentra en el siguiente link, con el nombre de “SensorH2O”, ahí está el trabajo final, que se desarrolló durante la residencia profesional. Link del repositorio “SensorH2O”:

<https://github.com/jwilliamsee/SensorH2OResidencia>

Conclusiones

El dispositivo H2O representa un avance importante en el desarrollo final del proyecto, porque se planteaba al inicio, que se tenían los proyectos trabajados en reportes de residencia, tesis y otros documentos, que era un poco tedioso poder retomar o incluso conocer información que no se documentaba en los avances de los proyectos, por lo que, resultó muy útil la estructura elegida y así poder darle seguimiento más fácil, editar en los mismos repositorios sin perder información en cada etapa hasta ver desarrollado al 100% cada dispositivo, el repositorio del dispositivo H2O con la estructura de documentación planteada, fue de gran ayuda como referencia para poder documentar los otros dispositivos.

En cuanto a operación del dispositivo H2O, era necesario complementarlo con un manual de ensamble para tener un mejor panorama de lo que había que hacer con el dispositivo terminado.

Las lecturas de más de tres sensores de caudal YF-S201, puede no ser la mejor forma de medir con gran precisión, pero resulta de gran ayuda para tener control en la lectura de flujo, lo recomendable, es ejecutar los cinco códigos del dispositivo, pero para un solo sensor de caudal, es decir, que sea un sensor de caudal por cada microcontrolador ESP y así tener lecturas casi exactas, cabe recordar que calibrado, existe un error del 10%.

En cuanto a lo profesional puedo concluir, que fue un proyecto que representó todo un reto, porque fueron muchas horas de dedicación a entender cuestiones técnicas, uso de la terminal en Linux, entender la programación en Micropython y en menor escala Python, aprender a usar otros software de diseño y simulación electrónica, el reto fue poder llevar a cabo el proyecto que, en su mayoría fueron cuestiones teóricas, también se hicieron pruebas y todo a distancia, por la pandemia que se vive en todo el mundo, obtuve mucho aprendizaje para colaborar a distancia y en equipo, resultó satisfactorio ver como se puede trabajar en equipo y que las cosas avancen bien estando a distancia.

Competencias desarrolladas

El trabajo en equipo a distancia, es decir, sin conocer del todo a las otras personas con las que se colaboró, es una competencia nueva, por todo lo desarrollado y llevar la secuencia idónea del proyecto, resultó satisfactorio.

El fortalecimiento del autoaprendizaje se vio reflejado en todo momento por el trabajo a distancia, eso es una competencia que favoreció mucho para el desarrollo del proyecto.

El uso de las tecnologías de comunicación también fue una competencia que se desarrolló de mejor forma y durante todo el proyecto representó resultados favorables.

El aprendizaje de nuevos y diferentes softwares para desarrollar el proyecto también es una competencia que complementa mi formación profesional.

El fortalecimiento de habilidades, como, por ejemplo, planificación, comunicación, compromiso, organización, trabajo en equipo, decisión, iniciativa para tomar decisiones, autocrítico y adaptación fueron de gran ayuda durante el desarrollo del proyecto.

Sin duda el conocimiento adquirido durante la carrera profesional, fue clave y muy útil para poder desarrollar el proyecto, materias de la formación de la carrera, como, mecánica de fluidos, electrónica analógica, electrónica digital, formulación y evaluación de proyectos, diseño asistido por computadora, entre otras, fueron de gran ayuda y sin mucha complicación, se entendieron conceptos pertenecientes a dichas materias.

Fuentes de información

1. AUTODESK TINKERCAD. (2011). *AUTODESK TINKERCAD*. Obtenido de TINKERCAD:
<https://www.tinkercad.com/>
2. Dillinger. (s.f.). *Dillinger.io*. Obtenido de DILLINGER: <https://dillinger.io/>
3. EasyEDA. (2008). *EasyEDA*. Obtenido de EasyEDA: <https://easyeda.com/>
4. fritzing electronics made easy. (2007). *fritzing.org*. Obtenido de fritzing: <https://fritzing.org/>
5. Geek Factory. (2013). *Geek Factory*. Obtenido de <https://www.geekfactory.mx/tienda/sensores/yf-s201-caudalimetro-sensor-de-flujo-1-2-pulgada/>
5. Github, Inc. (Enero de 2010). *github*. Obtenido de GitHub: <https://github.com/>
6. Mataix, C. (1986). *Mecánica de fluidos y máquinas hidráulicas*. Madrid: Ediciones del castillo S.A.
7. Open Know-How. (2019). *Open Know-How*. Obtenido de Open Know-How:
<https://app.standardsrepo.com/MakerNetAlliance/OpenKnowHow/src/branch/master/1>
8. Oracle. (22 de April de 2016). *oracle*. Obtenido de ORACLE.
9. Pandao. (2015). *Pandao Github*. Obtenido de M Editor.md: <https://pandao.github.io/editor.md/en.html>
10. RANDOM NERD TUTORIALS. (2009). *RANDOM NERD TUTORIALS*. Obtenido de
<https://randomnerdtutorials.com/about>

Anexos

Anexo 1.

Diagrama esquemático realizado en el software de simulaciones electrónicas de Proteus

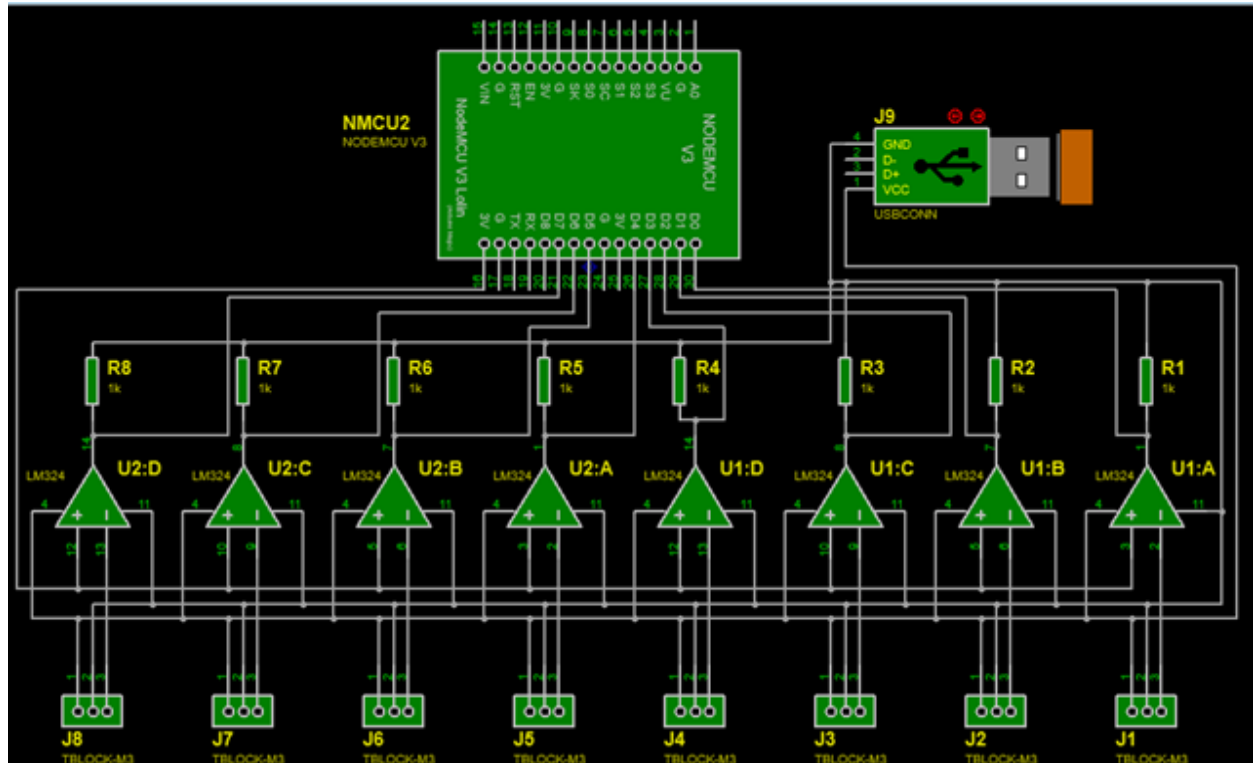
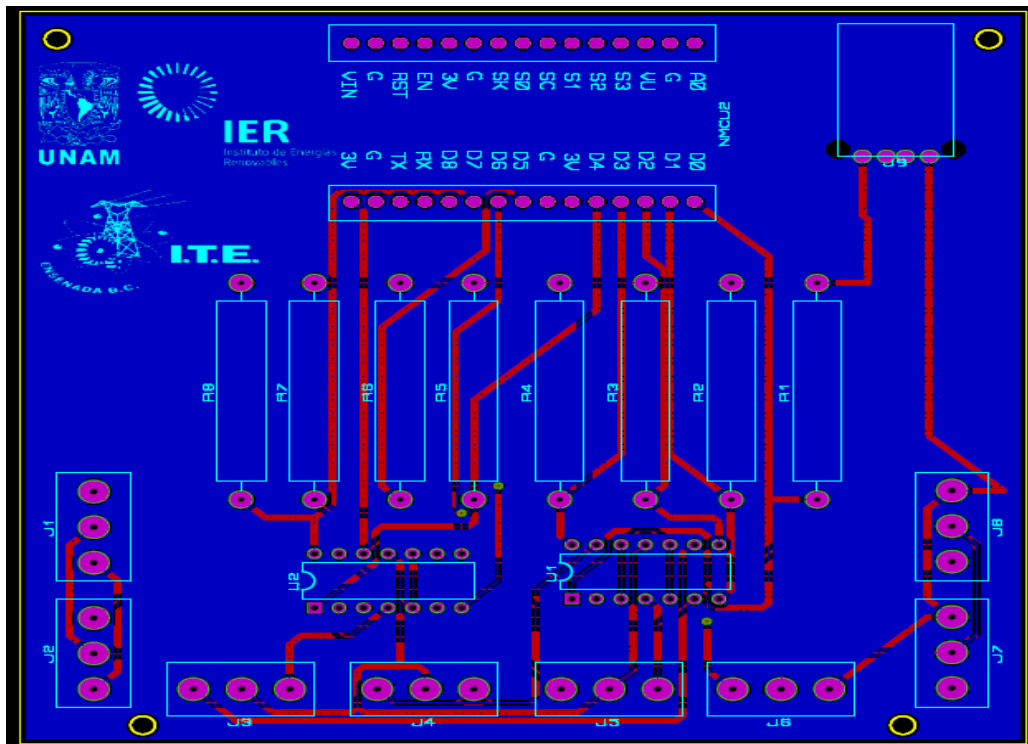
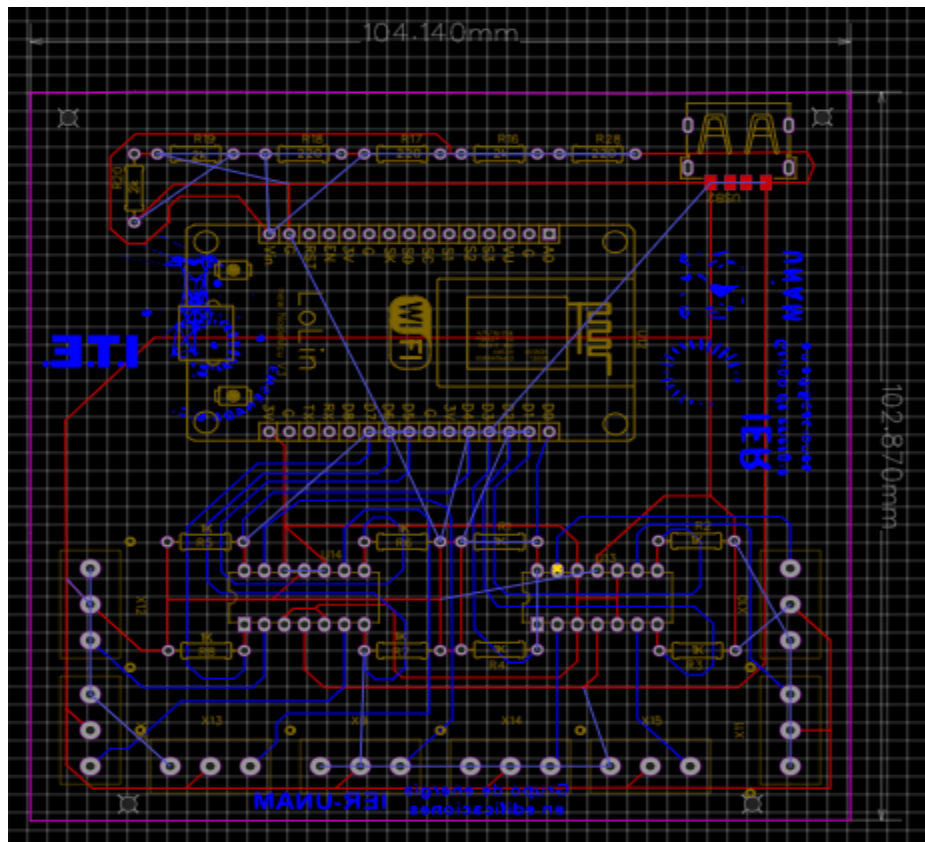


Diagrama esquemático realizado en EasyEDA



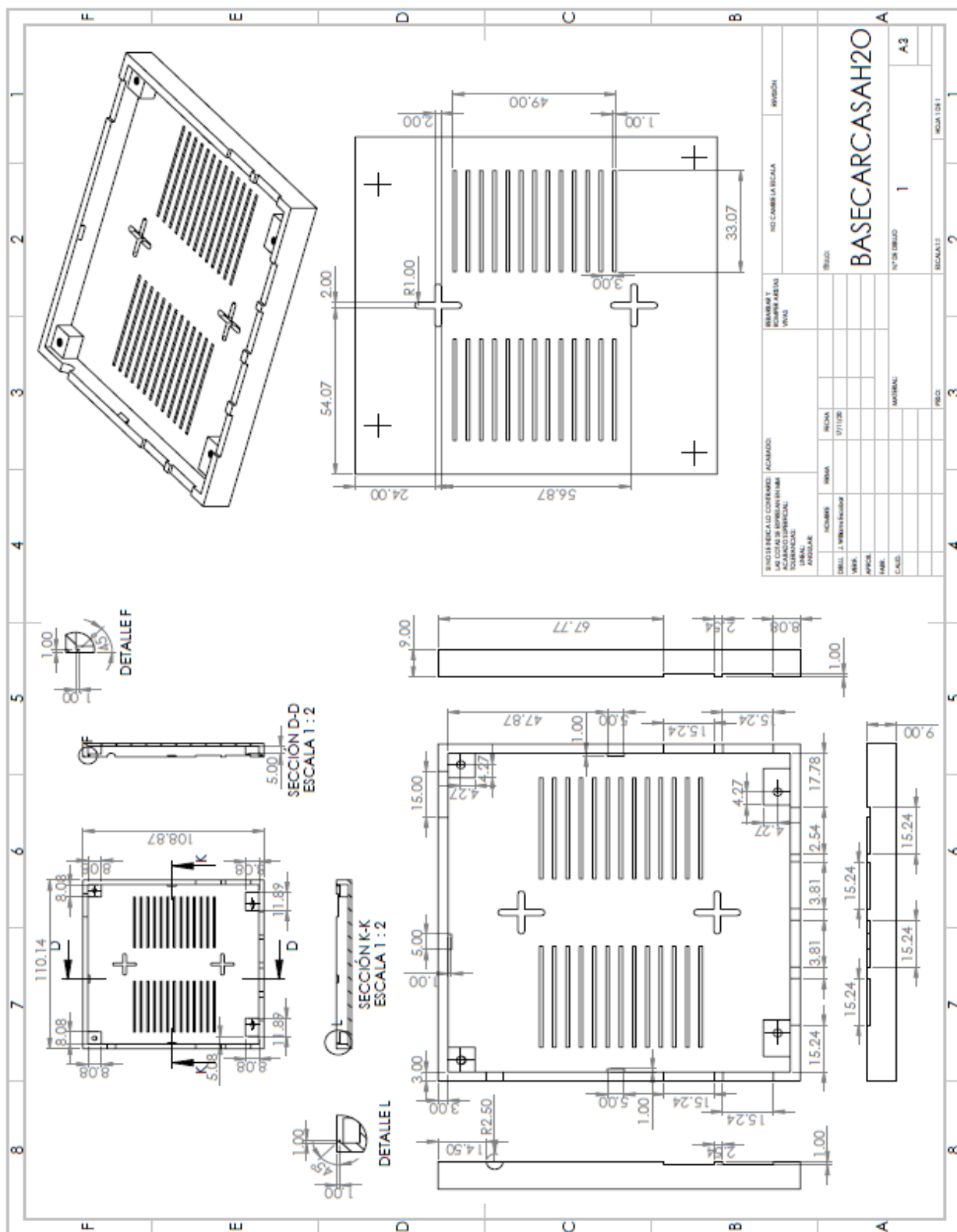
Anexo 3.

Medidas de la placa presentado en el software de simulaciones electrónicas Proteus.

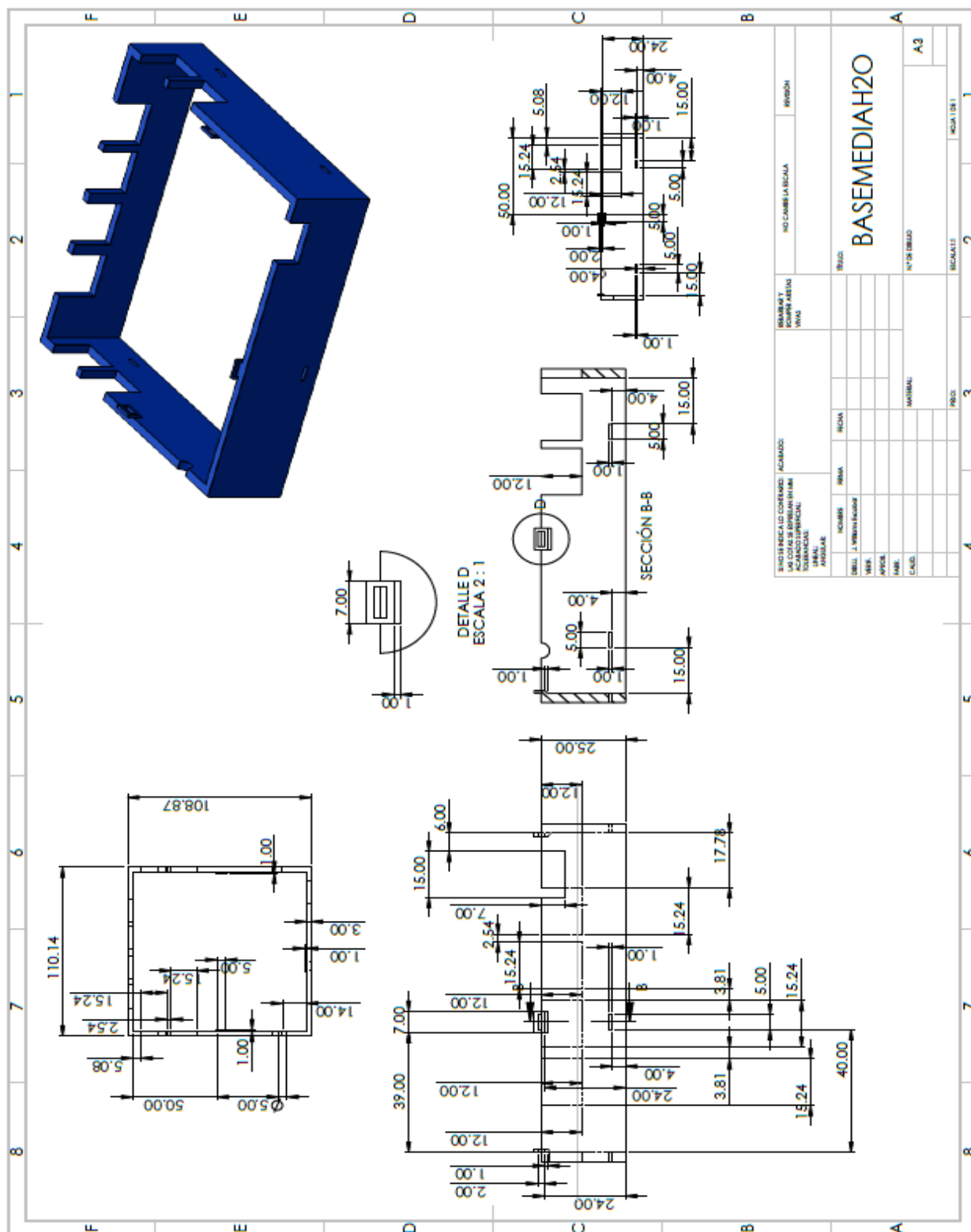


Anexo 4.

Anexo 4a.



Anexo 4B.



Anexo 4c.

