

Informe Tecnico DataLogger

Newton microscopios
Escuela Tecnica N°7

Nombres:	Franco Valentin - Lesme Franco - Sarniguette Valentino
Centro de investigación:	Newton SRL
Título del proyecto:	Prácticas profesionalizantes Newton
Supervisor :	Diego Palmieri

1. Resumen

El presente proyecto, desarrollado en el marco de las Prácticas Profesionalizantes Externas (PPE) en la institución Newton, tiene como finalidad la integración práctica de sensores, módulos electrónicos y programación en Python para el diseño de un sistema de adquisición y registro de datos ambientales. A lo largo de esta experiencia, se ha buscado simular una situación de trabajo real en la industria tecnológica, en donde se exige tanto el dominio técnico como la capacidad de documentar procesos, resolver problemas y trabajar en equipo.

El sistema diseñado consiste en una estación de monitoreo ambiental multipropósito, que permite medir, registrar y analizar parámetros como temperatura, humedad y tiempo exacto, integrando sensores analógicos, digitales y módulos de reloj de tiempo real. La plataforma de desarrollo utilizada es Python, por su facilidad de uso, legibilidad del código y compatibilidad con múltiples dispositivos como Raspberry Pi o microcontroladores con comunicación serial.

Este sistema no solo realiza lecturas aisladas, sino que registra cada valor acompañado de la hora exacta en que fue tomado, lo que habilita la posibilidad de realizar análisis históricos o generación de reportes periódicos.

1.1. Observaciones

Este proyecto pone a prueba habilidades fundamentales:

- Conexión e interpretación de sensores reales.
- Manejo de protocolos de comunicación digital.
- Desarrollo de software para la lectura y gestión de datos.
- Trabajo colaborativo y documentación técnica diaria, respetando un orden y planificación.

A través de la práctica se fortalece el pensamiento lógico, la capacidad de resolución de problemas técnicos y la autonomía en el desarrollo de sistemas reales, competencias indispensables para la futura inserción laboral.

2. Componentes Utilizados

Para llevar a cabo este proyecto, se requiere la utilización de diversos sensores y dispositivos electrónicos que permitirán obtener datos precisos y en tiempo real. Entre los componentes necesarios se encuentran el sensor de temperatura y humedad DHT11, el módulo de reloj de tiempo real MH-Real-Time, una Raspberry Pi como unidad de procesamiento central, y una termocupla, ideal para mediciones de temperatura en entornos más exigentes. La integración de estos elementos es fundamental para garantizar el correcto funcionamiento del sistema y la confiabilidad de los datos recopilados.

2.1. DHT11



2.1.1. Funcionamiento del código

Este código en MicroPython está diseñado para ejecutarse en una Raspberry Pi y leer datos de un sensor DHT11, que mide temperatura y humedad. Primero, importa los módulos necesarios para manejar el sensor y los pines de la placa, así como para controlar el tiempo. Luego, inicializa el sensor DHT11 en el pin 15 de la Raspberry Pi. Dentro de un bucle infinito, el programa solicita al sensor que realice una medición, obtiene la temperatura y la humedad, y muestra estos valores por pantalla. Finalmente, espera dos segundos antes de repetir el proceso, permitiendo así lecturas periódicas y continuas.

2.1.2. Librerías utilizadas

1. **Machine:** Se usa para controlar el hardware, específicamente los pines GPIO en plataformas como ESP32 o Raspberry Pi Pico.
2. **Time:** Permite controlar los tiempos de espera necesarios para leer correctamente el protocolo del DHT11.
3. **dht:** Módulo de MicroPython que permite manejar el sensor DHT11 directamente sin código adicional de decodificación.

2.2. Raspberry

En este proyecto la Raspberry Pi Pico cumple un rol central como unidad de control y procesamiento. A diferencia de modelos más robustos como la Raspberry Pi 4, la Pico está diseñada específicamente para aplicaciones de bajo consumo y control en tiempo real, lo que la hace ideal para Datalogger.

Gracias a su microcontrolador RP2040, que cuenta con un procesador de doble núcleo a 133 MHz y 264 KB de RAM, la Raspberry Pi Pico permite una lectura eficiente de datos de sensores, así como la ejecución de tareas simultáneas mediante programación en MicroPython o C/C++. Su bajo costo, tamaño reducido y gran cantidad de pines GPIO facilitan la conexión directa con dispositivos como el sensor de temperatura y humedad DHT11, el módulo de reloj en tiempo real y las termocupas, incluso permitiendo el uso de protocolos como I2C, SPI o UART según sea necesario.

2.3. MH-Real-Time



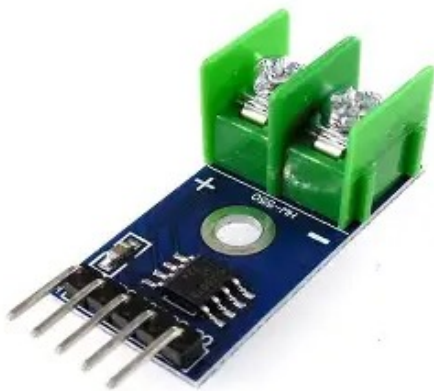
2.3.1. Funcionamiento del código

Este código implementa el control de un reloj de tiempo real (RTC) DS1302 usando MicroPython en una placa compatible, como una Raspberry Pi Pico. Define una clase llamada DS1302 que permite leer y escribir la fecha y hora en el chip a través de pines digitales, utilizando funciones para convertir entre formatos decimal y hexadecimal, y métodos para acceder y modificar cada campo de la fecha y hora (segundos, minutos, horas, día, mes, año, etc.). El programa principal inicializa el RTC con los pines correspondientes y, en un bucle infinito, lee la fecha y hora actual del DS1302, mostrándola por pantalla cada segundo. Además, incluye la opción de establecer la fecha y hora manualmente (comentada por defecto) y maneja la interrupción por teclado para finalizar el programa de forma segura.

2.3.2. Librerías utilizadas

1. **machine:** Se usa para controlar los pines GPIO (para el reloj, datos y habilitación del chip DS1302).
2. **time:** Utilizada para demoras y sincronización.
3. **ds1302:** Es una librería personalizada escrita en Python que implementa la comunicación con el módulo DS1302 usando protocolo serial.

2.4. Termocupla



2.4.1. Funcionamiento del código

El HW-550 es un sensor de humedad del suelo de tipo digital. Funciona detectando la presencia de humedad mediante conductividad eléctrica. Al insertar las dos sondas metálicas en la tierra, el sensor mide si hay paso de corriente entre ellas.

Cuando el suelo está húmedo, el sensor activa su salida digital y el valor leído desde el microcontrolador es 0. En cambio, si el suelo está seco, no hay conductividad y la salida es 1.

2.4.2. Librerías utilizadas

1. **machine:** Se usa para configurar el pin de entrada digital conectado al HW-550
2. **time:** Se utiliza para pausar entre lecturas y evitar lecturas demasiado seguidas.

3. Objetivos alcanzados y mejoras posibles

A lo largo de las practicas estuvimos realizando la programación de los componentes mostrados anteriormente, donde aqui detallaremos todo lo realizado y los detalles que podrian tener una optimización.

3.1. Objetivos cumplidos

1. Funcionamiento ideal de los sensores de manera independiente.
2. Descripcion del funcionamiento de cada componente en [Github](#)
3. Realizacion de las respectivas conexiones

3.2. Posibles mejoras

1. Posible optimización de código
2. Congeniar la totalidad de los códigos de todos los sensores
3. Funcionamiento en Conjunto

4. Conclusión final

El desarrollo del proyecto DataLogger representó una experiencia enriquecedora tanto a nivel técnico como formativo. A lo largo de las Prácticas Profesionalizantes Externas, se logró integrar con éxito distintos sensores, módulos electrónicos y microcontroladores, aplicando conocimientos de electrónica, programación en MicroPython y protocolos de comunicación digital. Se diseñó un sistema funcional capaz de medir, registrar y visualizar en tiempo real parámetros ambientales como la temperatura, la humedad y la fecha/hora exacta, demostrando la utilidad y versatilidad de plataformas como la Raspberry Pi Pico en aplicaciones de monitoreo.

El trabajo permitió consolidar habilidades clave como la interpretación de sensores reales, el desarrollo de software orientado al control de hardware, y la documentación técnica diaria, fortaleciendo también el pensamiento lógico, la autonomía y el trabajo en equipo. Además, se alcanzaron todos los objetivos propuestos, dejando en evidencia la viabilidad del sistema desarrollado.

Si bien el proyecto cumplió sus metas fundamentales, se identificaron posibles mejoras a futuro, como la optimización del código y la integración total del funcionamiento de los sensores en un único sistema coordinado. Estas oportunidades de mejora abren la puerta a futuras iteraciones del proyecto, acercándolo aún más a una solución profesional aplicable en entornos reales.

En conclusión, el proyecto DataLogger no solo evidenció la capacidad técnica adquirida durante la formación, sino que también sirvió como puente entre el ámbito académico y el mundo laboral, simulando con fidelidad una experiencia profesional dentro del sector tecnológico.