

UNIVERSIDAD DE MÁLAGA
ESCUELA TÉCNICA SUPERIOR DE
INGENIERÍA DE TELECOMUNICACIÓN

INGENIERÍA DEL SOFTWARE PARA INTERNET DE LAS
COSAS

SISTEMA DE MONITOREO AMBIENTAL CON ARDUINO Y ESP32

MÁSTER EN TELEMÁTICA Y
REDES DE TELECOMUNICACIÓN

DANIEL BAZO CORREA Y LAURA CAMACHO LARA
MÁLAGA, 2024

Índice

1	Desarrollo del sistema	1
1.1	Introducción	1
1.2	Desarrollo	1
1.2.1	Arduino Mega	2
1.2.2	ESP32	3
1.2.3	Streamlit	4

Capítulo 1

Desarrollo del sistema

1.1. Introducción

El sistema de monitorización propuesto se fundamenta en la utilización de un microcontrolador ESP32 y una Arduino Mega, equipada con un módulo HC-06. Su principal objetivo es la recopilación de datos de diversos sensores para obtener información sobre temperatura, humedad y luminosidad, transmitir estos datos a un ordenador y procesarlos para su posterior visualización y análisis, con el propósito de simular un sistema de monitorización para un invernadero. La Figura 1.1 ilustra un diagrama del sistema de monitorización propuesto.

En la etapa inicial, los sensores se conectan a la ESP32, que se encarga de recoger las mediciones. Posteriormente, los datos son enviados al módulo Bluetooth HC-06 de la Arduino Mega. La Arduino Mega, conectada a un ordenador, recibe estos datos y los transfiere al ordenador mediante una conexión por cable. Finalmente, se emplea el lenguaje de programación Python para recoger y procesar los datos.

Los datos recogidos se almacenan en una estructura de datos bidimensional, similar a una hoja de cálculo de Excel, facilitando así una manipulación de datos eficiente y flexible. En resumen, el sistema de monitorización ofrece una solución eficaz para la recopilación, transmisión, almacenamiento y análisis de datos de sensores.

1.2. Desarrollo

En las siguientes secciones se detalla el desarrollo del sistema.

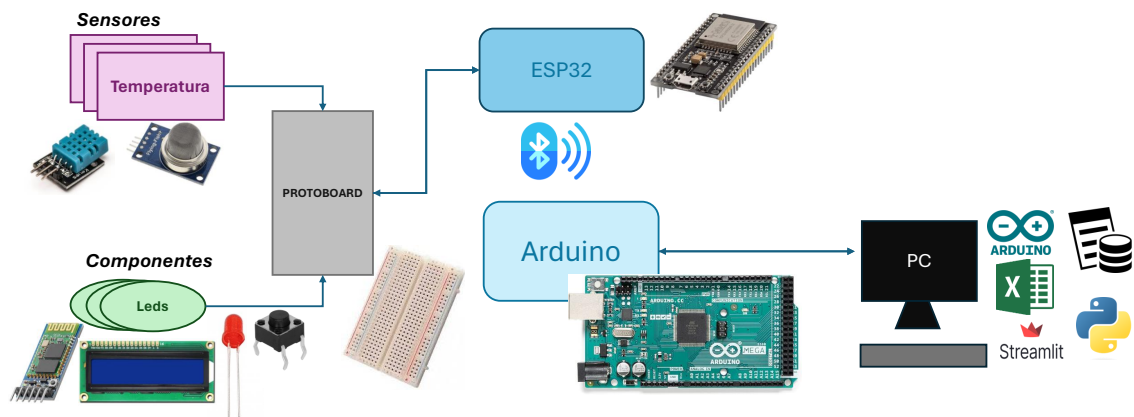


Figura 1.1: Diagrama del sistema de monitorización propuesto.

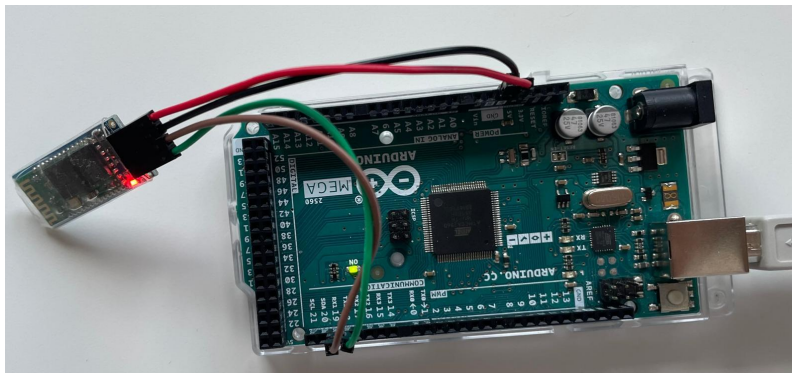


Figura 1.2: Conexión del módulo HC-06 a la Arduino Mega.

1.2.1. Arduino Mega

El código implementado en la Arduino Mega facilita la recepción de datos a través de una comunicación Bluetooth desde un ESP32, utilizando un módulo Bluetooth HC-06. En caso de interrupción en la comunicación entre el ESP32 y la Arduino Mega, los datos recogidos por los sensores se almacenan en una base de datos. Para gestionar esta interrupción, la Arduino Mega emite el comando *Comunicación cortada* a través del puerto serie, que se captura posteriormente mediante Python en el ordenador vinculado a la Arduino Mega. Los datos almacenados se visualizan en gráficos mediante la herramienta Streamlit.

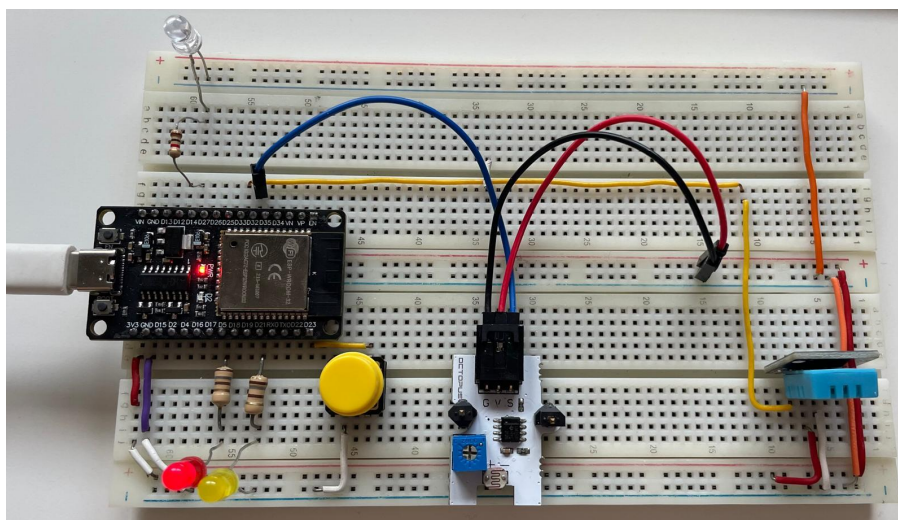


Figura 1.3: Conexión de sensores a la ESP32 en una protoboard.

1.2.2. ESP32

Se configura el módulo Bluetooth del ESP32 como dispositivo maestro y se establece la conexión con un dispositivo esclavo Bluetooth, en este caso, un HC-06. Esta configuración requiere la definición previa de los baudios a utilizar, así como el nombre de la conexión y el pin de la red Bluetooth del módulo HC-06 mediante comandos AT.

El código supervisa un botón de color amarillo, ver Figura 1.3, para modificar el estado de la transmisión de datos. Cuando se activa la transmisión, se leen y transmiten al dispositivo esclavo los valores capturados por los sensores. Además, se incluyen funcionalidades como la reconexión automática en caso de pérdida de la conexión Bluetooth. Los estados, tanto de transmisión como de interrupción de la comunicación, se reflejan mediante dos indicadores LED: un LED rojo que se ilumina en caso de interrupción de la comunicación y un LED amarillo que indica la correcta transmisión de los datos.

Para establecer una comunicación bidireccional, el sistema entra en un estado de espera para recibir datos del dispositivo esclavo vía Bluetooth cuando la transmisión de datos está inactiva. El estado de la recepción de datos se indica mediante un LED azul en el ESP32, que se enciende o apaga dependiendo del mensaje ('ON' o 'OFF') recibido del dispositivo esclavo. Así, el encendido del LED azul indica la detención de la recepción de datos, proporcionando una retroalimentación visual instantánea del estado de la comunicación.



Inicio



Introducción

Bienvenidos a nuestro proyecto, que se centra en la creación de un sistema IoT basado en las tecnologías Arduino y ESP32, diseñado para adquirir, manipular y visualizar datos. Nuestro objetivo principal es simular el funcionamiento de un invernadero.

Para lograr una simulación precisa, recopilamos datos sobre varios factores ambientales críticos. Estos incluyen la temperatura, la luminosidad y la humedad. Al monitorear y analizar estos factores, podemos optimizar las condiciones dentro del invernadero y mejorar la eficiencia de la producción.



Aplicación

Introduce los baudios:

9600,00

- +

Introduce el nombre del fichero:

invernadero

Comenzar la adquisición de datos

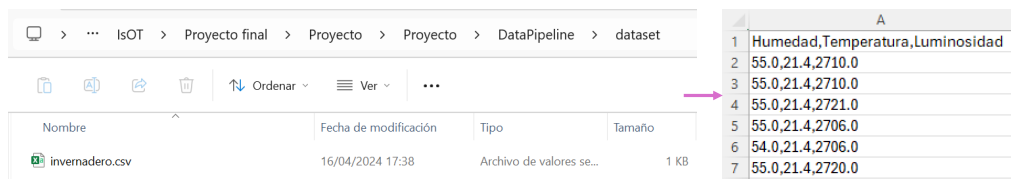
Figura 1.4: Interfaz web de Streamlit.

1.2.3. Streamlit

Una vez capturados y almacenados los datos de los sensores, se procede a su procesamiento. Para este propósito, se ha desarrollado una aplicación web utilizando Streamlit, una biblioteca de Python especializada en la creación de aplicaciones interactivas.

La Figura 1.4 muestra la interfaz de la aplicación web, que incluye una breve descripción del proyecto. Esta interfaz permite la selección de parámetros de entrada, como la velocidad de transmisión en baudios y el nombre del archivo CSV donde se guardarán los datos recopilados.

La ejecución de la aplicación web es coordinada por el archivo principal, denominado *streamlit_web.py*. Este archivo hace uso de dos módulos adicionales para llevar a cabo tareas específicas. Dichos módulos son detallados a continuación.



The image shows a file explorer window with the path 'IsOT > Proyecto final > Proyecto > Proyecto > DataPipeline > dataset'. It lists a file 'invernadero.csv' with a modification date of '16/04/2024 17:38', type 'Archivo de valores se...', and size '1 KB'. To the right, an Excel spreadsheet is open, showing a table with 7 rows and 4 columns. The first column is labeled 'A' and the first row is labeled '1'. The data in the spreadsheet is as follows:

	Humedad	Temperatura	Luminosidad
1	55.0	21.4	2710.0
2	55.0	21.4	2710.0
3	55.0	21.4	2721.0
4	55.0	21.4	2706.0
5	54.0	21.4	2706.0
6	55.0	21.4	2720.0
7			

Figura 1.5: Almacenamiento de datos en Excel.

Adquisición de datos

Una vez se introducen todos los parámetros requeridos en la aplicación web, el código principal invoca al módulo *data_acquisition.py*. Este módulo se encarga de capturar los datos desde un puerto serie y almacenarlos en un archivo CSV, ver Figura 1.5, utilizando para ello la biblioteca Pandas.

La función toma tres parámetros de entrada:

1. **Puerto:** Corresponde al puerto serie al que está conectado el dispositivo.
2. **Baudios:** Representa la velocidad de transmisión en baudios del puerto serie.
3. **Nombre del archivo:** Especifica el nombre del archivo CSV con los datos adquiridos.

Si hay datos disponibles, la función los lee y procesa, generando un Data-Frame de Pandas a partir de la lista de datos recibidos, que posteriormente se guarda en un archivo CSV.

Análisis de datos

El script *plot_data.py* ha sido diseñado para visualizar datos a partir de un archivo CSV. En primer lugar, este script lee el archivo CSV que almacena los datos de los sensores previamente recopilados. Posteriormente, procesa estos datos y genera diferentes gráficas para ilustrar la evolución de la temperatura a lo largo del tiempo, reflejando así la información ambiental recogida por nuestro sistema, la variación de la humedad, y los cambios producidos en la luminosidad. Estas gráficas se muestran en la interfaz de la aplicación web mediante Streamlit. La Figura 1.6 ilustra un ejemplo de cómo se visualizan las gráficas en Streamlit.

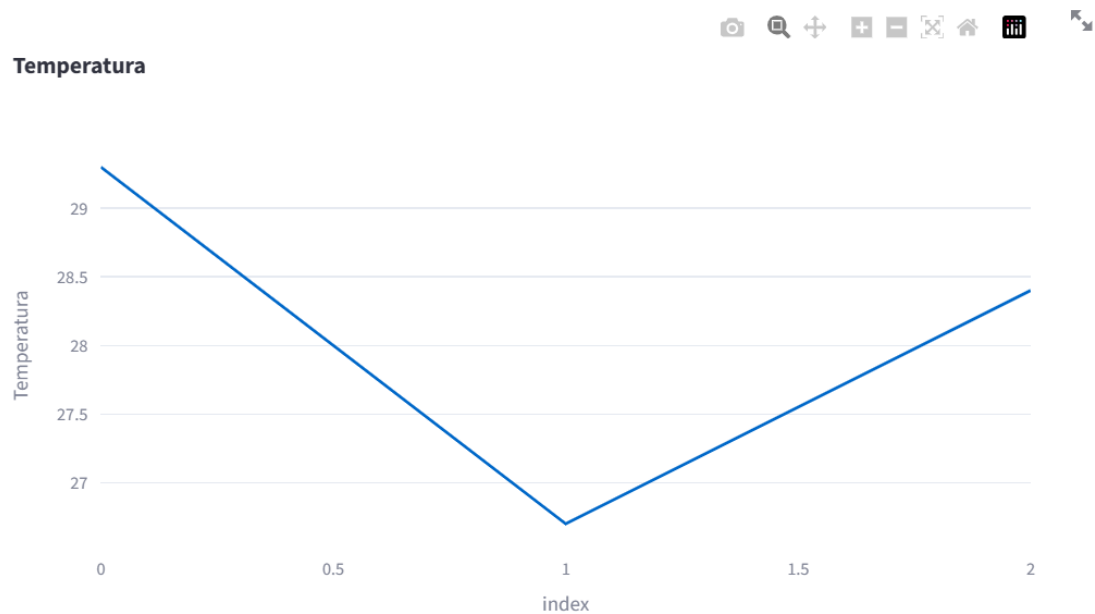


Figura 1.6: Almacenamiento de datos en Excel.

También se incluyen otras funcionalidades, como el cálculo de la media de los parámetros recopilados, así como alertas y notificaciones en caso de que la humedad supere un umbral determinado. Además, se ofrecen recomendaciones para el cultivo en el invernadero según las condiciones ambientales específicas del entorno, ver Figuras 1.7 y 1.8.

Media de los parámetros ambientales

Media de Temperatura: 28.133333333333336

Media de Humedad: 95.0

Media de Luminosidad: 2730.6666666666665

Figura 1.7: Media de los parámetros ambientales.

Alertas y Notificaciones

Humedad alta detectada

Recomendaciones de Cultivo

Dadas las condiciones actuales del invernadero, se recomienda cultivar tomates.

Figura 1.8: Alertas y recomendaciones.