	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	03
		Fecha	18-07-2023

1. IDENTIFICACIÓN DE LA GUÍA

Nombre de la guía:	Proyecto Microcontroladores MicroPython y CircuitPython
Código de la guía (No.):	001
Taller(es) o Laboratorio(s) aplicable(s):	
Tiempo de trabajo práctico estimado:	10 horas
Asignatura(s) aplicable(s):	Programación Avanzada
Programa(s) Académico(s) / Facultad(es):	Ingeniería Electrónica e ingeniería en Telecomunicaciones

COMPETENCIAS	CONTENIDO TEMÁTICO	INDICADOR DE LOGRO
Desarrollar habilidades para resolver problemas prácticos en microcontroladores empleando herramientas graficas de simulación. Aplicar conceptos de programación para proponer una solución al problema planteado. Evaluar el desempeño del código programado para la solución del problema.	Simulación de microcontroladores programados con CircuitPython y MicroPython. Uso de condicionales y ciclos. Uso de listas y diccionarios.	Programa el microcontrolador usando Python para solucionar un problema de automatización y control.

2. FUNDAMENTO TEÓRICO

La plataforma WokWi permite simular la conexión de microcontroladores con sensores y actuadores. Aún más, permite evaluar la programación del microcontrolador interactuando con sensores y actuadores. Adicionalmente, la plataforma cuenta con la opción de realizar simulaciones de WiFi lo que implica que puedes conectar la simulación a plataformas de Internet de las cosas. La documentación oficial de la plataforma esta disponible en <https://docs.wokwi.com/>.

En la figura 1 se presenta la plantilla básica de la plataforma WokWi donde contamos con unas pestañas para modificar los archivos del proyecto (main.py y diagram.json). Al lado izquierdo debajo de las pestañas podemos modificar el contenido de cada archivo. Mientras que, al lado derecho contamos con la ventana de simulación, donde se encuentran conectados los sensores y actuadores con el microcontrolador.

	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	03
		Fecha	18-07-2023

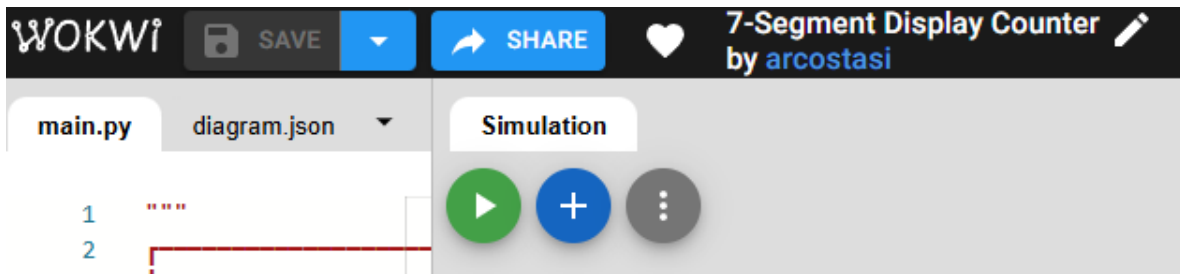


Figura 1. Elementos básicos de WokWi.

La plataforma cuenta con gran variedad de microcontroladores dispositivos análogos y digitales de entrada y salida. También se encuentra soportada por una documentación oficial <https://docs.wokwi.com/>. Esta plataforma permite simular la programación y la interconexión de los microcontroladores Arduino, ESP32, STM32 y Raspberry Pi Pico, como se muestra en la figura 2.

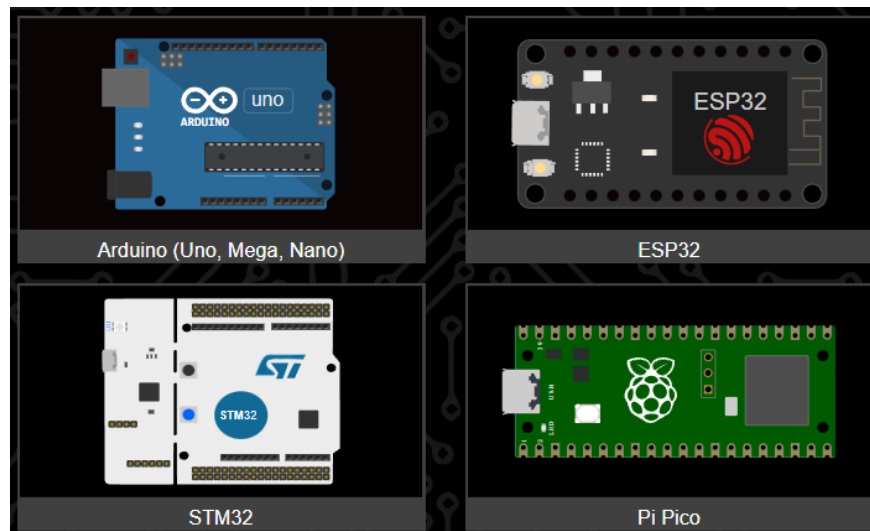


Figura 2. Elementos básicos de WokWi.

La mayoría de los microcontroladores tienen pines que pueden ser configurados como entrada o salida digital (GPIO). Algunos microcontroladores también pueden tener entradas (ADC) y salidas analógicas (DAC) en los mismos pines que se configuran como digitales, ver figura 3. El hardware que se puede emplear en la plataforma esta descrito en <https://docs.wokwi.com/getting-started/supported-hardware>.

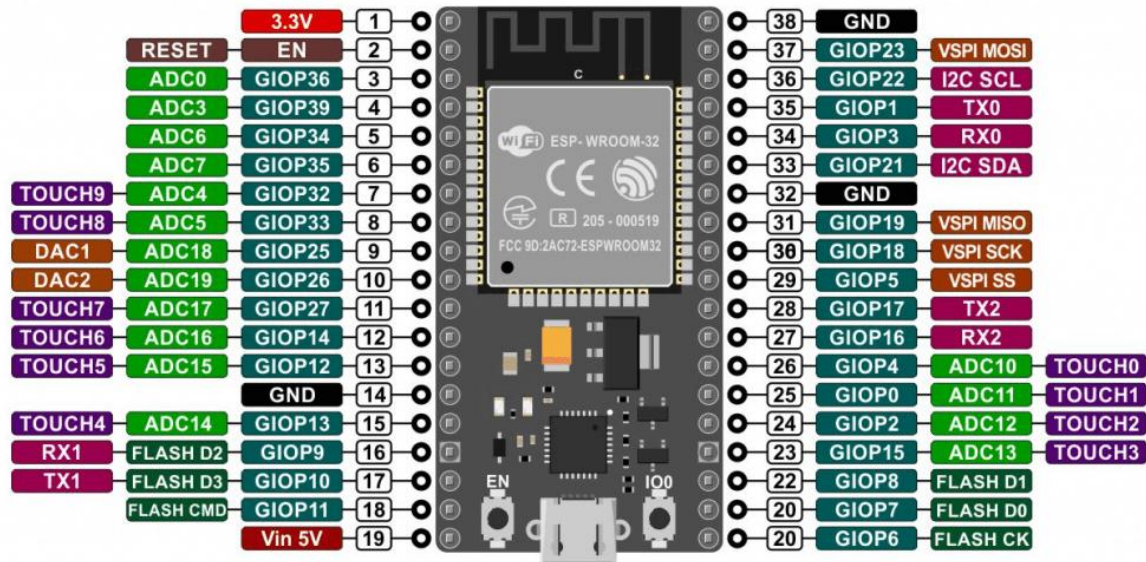


Figura 3. Ejemplo de asignación de pines para el microcontrolador ESP-WROOM-32.

Ejemplos:

Raspberry Pi Pico conectado a un Display de 7 segmentos programado con MicroPython. Realiza la cuenta de 0 a 15 en hexadecimal (0 a F), y por medio de un switch se puede invertir el sentido de la cuenta de ascendente a descendente, y viceversa. <https://wokwi.com/projects/300210834979684872>

En muchas ocasiones es necesario controlar la periodicidad en la cual se ejecuta el algoritmo de control implementado en el microcontrolador. En el siguiente ejemplo, se realiza la acción dentro del condicional del "while" cada 100 mili segundos. Probar dicho código en la plataforma WokWi con un microcontrolador ESP32.

```
import time
```

```
sampling_interval_ms = 100 # Desired sampling interval in milliseconds
last_sample_time = time.ticks_ms()
```


```
while True:
```

```
    current_time = time.ticks_ms()
    elapsed_time = time.ticks_diff(current_time, last_sample_time)
```

```
    if elapsed_time >= sampling_interval_ms:
        # Perform sampling actions here
        print(f"Sample taken after {elapsed_time} ms")
        last_sample_time = current_time
```

Después de definir la interconexión entre el microcontrolador y los elementos de salida y entrada, se procede a programar el microcontrolador. La estructura típica de un programa o código para el microcontrolador es:

- Importa librerías a utilizar.
- Configurar pines, constantes (periodo de muestreo, entre otros), conexiones.
- Verificar actuadores o sensores.

	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	03
		Fecha	18-07-2023

- d. Código principal de control dentro de un “while True”.

Para los proyectos propuestos los estudiantes deben desarrollar el código de cada paso.

3. OBJETIVO(S)

- 3.1. Desarrollar el programa que permita solucionar un problema de automatización electrónica.
- 3.2. Aplicar los conocimientos de la asignatura Programación Avanzada en casos prácticos.

4. RECURSOS REQUERIDOS

Un computador con acceso a internet.

5. PROCEDIMIENTO O METODOLOGÍA PARA EL DESARROLLO

5.1. Estación de Clima básica

Componentes: Sensor DHT22, LED RGB, y un Display.

Descripción: El ESP32 lee temperatura y humedad con un periodo de muestreo establecido. Si la temperatura supera un umbral alto, enciende el LED rojo; si esta por debajo de un umbral bajo se activa la luz azul; si esta entre ambos umbrales se activa la luz verde. Los valores de temperatura se almacenan en una lista, y se calcula el promedio para una cantidad N de muestras. Cuando se calcula el promedio de la temperatura, se muestra en el Display el valor calculado.

Consideraciones: los umbrales alto y bajo, el periodo de muestreo, y la cantidad N, son constantes que deben ser programadas en la sección de configuración del microcontrolador.

Ejemplos:

LED RGB (MicroPython) <https://wokwi.com/projects/338502152832942675>.


LCD Display (MicroPython) <https://wokwi.com/projects/359400194112248833>.

OLED Display (MicroPython) <https://wokwi.com/projects/305568836183130690>.

5.2. Sistema de riego automático

Componentes: Sensor DHT22, LED indicador, y un Relé.

Descripción: El sistema monitorea la humedad del suelo para un tiempo de muestreo establecido. Si la humedad está BAJA, activa el relé que enciende una bomba de agua. La tierra se humedece hasta un nivel considerado como ALTO. Mientras se encuentre activa la bomba, se debe activar el LED indicador. Cada vez que la humedad alcance el nivel alto se debe enviar por MQTT una notificación indicando que la humedad alcanzo el nivel deseado.

	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	03
		Fecha	18-07-2023

Consideraciones: los niveles BAJO y ALTO los define el estudiante de acuerdo con los valores recibidos desde el sensor DHT22. También se debe configurar el periodo de muestreo para tomar datos.

Ejemplos:

DHT22 (MicroPython) <https://wokwi.com/projects/322577683855704658>.

5.3. Control de temperatura en habitación

Componentes: Sensor DHT22, Buzzer, Display OLED.

Descripción: El microcontrolador mide temperatura y humedad constantemente (periodo de muestreo). Usa una lista para registrar las últimas 10 mediciones, en el display se muestra el promedio de estas 10 mediciones. Si la temperatura promedio supera el umbral ALTO, activa una alerta.

Consideraciones: el nivel ALTO lo define el estudiante de acuerdo con los valores recibidos desde el sensor DHT22. También se debe configurar el periodo de muestreo para tomar datos y analizarlos.

Ejemplos:

DHT22 (MicroPython) <https://wokwi.com/projects/322577683855704658>.

5.4. Sistema de detección de presencia y control de energía

Componentes: Sensor PIR (movimiento), relé, LED indicador, sensor de luz LDR (para evitar encendido diurno).

Descripción: Sí se detecta algún movimiento se activa el relé o el LED. Si no hay movimiento durante "tiempo>umbral", apagar la salida. Registrar cada evento guardando la fecha y hora en la lista. Mostrar para cada instante de muestreo el estado (ON o OFF) por consola o display.

Consideraciones: El valor de umbral es una constante definida en el código de configuración del microcontrolador.

Ejemplos:


LCD con fecha y hora <https://wokwi.com/projects/406173305722190849>.

6. PARÁMETROS PARA ELABORACIÓN DEL INFORME

Los grupos de estudiantes debe realizar una presentación en el cual incluya:

- 6.1. Nombres de los estudiantes y programa al que pertenecen.
- 6.2. Resumen de las actividades realizadas.
- 6.3. Desarrollo de las actividades propuestas en el numeral 5.
- 6.4. Conclusiones.

7. RUBRICA EVALUACIÓN

 Institución Universitaria <small>Reacreditada en Alta Calidad</small>	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	03
		Fecha	18-07-2023

Los estudiantes deben presentar el desarrollo del proyecto, comprender completamente el código entregado. Se les solicitara hacer un cambio, si no muestran capacidad de modificar el código ante un cambio de condición la nota se calificara sobre 4.0.

8. BIBLIOGRAFÍA

Didactronica por Pedro D. Domingo y Jorge Herrera (2024). Comunicación Wifi Simulada online en Wokwi. Programada de forma gráfica en microLSB para Esp32. <https://www.youtube.com/watch?v=3MuG8UpcfFw>
<https://didactronica.microlsb.es/docs>

FreeRTOS multi task project on Wokwi. <https://wokwi.com/projects/322609470223942226>

Elaborado por:	<i>Cristian Guarnizo Lemus</i>
Revisado por:	
Versión:	1.0
Fecha:	10/07/2025

8.1. Sistema de monitoreo de ruido ambiental

Componentes: Ruido (ADC), Display OLED.

Descripción: .

Consideraciones: .

Ejemplos:

Entrada Analoga <https://wokwi.com/projects/385612179110169601>
