

ไมโครคอนโทรลเลอร์และการใช้งาน  
APPLICATIONS OF MICROCONTROLLERS

03376126

# เนื้อหาตามหลักสูตร

ศึกษาและปฏิบัติในหัวข้อ สถาปัตยกรรมไมโครคอนโทรลเลอร์  
การเชื่อมต่อแอนะล็อกและดิจิทัลเข้ากับไมโครคอนโทรลเลอร์  
การเขียนโปรแกรมไมโครคอนโทรลเลอร์ การนำเฟิร์มแวร์เข้าสู่  
ไมโครคอนโทรลเลอร์ การแก้ปัญหา การประยุกต์ใช้งานผ่าน  
การทำโครงงาน

Study and practice in Microcontroller architecture,  
Microcontroller interfacing, Programming,  
Development process, Problem-solving, Project-based  
applications

แผนการเรียนรู้รายสัปดาห์

# 1 แนะนำไมโครคอนโทรลเลอร์และเครื่องมือเบื้องต้น

## ทฤษฎี (2 ชม.)

- ความรู้เบื้องต้นเกี่ยวกับไมโครคอนโทรลเลอร์, ความสำคัญ, การทำงานพื้นฐาน
- สถาปัตยกรรมพื้นฐานของไมโครคอนโทรลเลอร์

## ปฏิบัติ (2 ชม.)

- แนะนำเว็บไซต์ WOKWI การใช้งานเบื้องต้น, การสร้างวงจรจำลองง่ายๆ (ESP32)
- ทดลองเขียนโปรแกรม "Hello World" บน WOKWI (จำลอง Serial Monitor)

## 2 Programming Model และสถาปัตยกรรม

### ทฤษฎี (2 ชม.)

- ตัวแบบการโปรแกรม (Programming Model)
- สถาปัตยกรรมไมโครโพรเซสเซอร์และไมโครคอนโทรลเลอร์

### ปฏิบัติ (2 ชม.)

- การสร้างโปรเจกต์ใน WOKWI (ESP32 Dev Module)
- การสำรวจข้อมูลในหน่วยความจำเบื้องต้น

# 3 พื้นฐานการเขียนโปรแกรมไมโครคอนโทรลเลอร์

## ทฤษฎี (2 ชม.)

- โครงสร้างพื้นฐานของโปรแกรม (setup(), loop())
- ประเภทข้อมูล (int, float, char, Boolean), ตัวแปร, ค่าคงที่
- ตัวดำเนินการทางคณิตศาสตร์, ตรรกะ, เปรียบเทียบ
- คำสั่งควบคุมการทำงาน: if-else, switch-case

## ปฏิบัติ (2 ชม.)

- ฝึกการเชื่อมต่อวงจร (บน WOKWI)
- ฝึกการใช้งานภาษาโปรแกรมในหัวข้อที่กล่าวถึงในภาคทฤษฎี

## 4 คำสั่งควบคุมการวนซ้ำและการใช้งาน Digital I/O

### ทฤษฎี (2 ชม.)

- คำสั่งควบคุมการวนซ้ำ: for loop, while loop, do-while loop
- ฟังก์ชัน pinMode(), digitalWrite(), digitalRead()
- แนวคิด Pull-up/Pull-down resistor สำหรับปุ่มกด

### ปฏิบัติ (2 ชม.)

- ทดลองเขียนโปรแกรมควบคุม LED แบบจับซ้อนขึ้น (เช่น วนกระพริบหลายดวง เป็นรูปแบบต่าง ๆ ) (บน WOKWI)
- จำลองการต่อปุ่มกดและอ่านสถานะปุ่มกด (บน WOKWI)
- เขียนโปรแกรมควบคุม LED ด้วยปุ่มกด (เปิด/ปิด, สลับสถานะ)

# 5 การใช้งาน Analog I/O และการจัดการเวลา

## ทฤษฎี (2 ชม.)

- การแปลงสัญญาณ Analog เป็น Digital (ADC)
- ฟังก์ชัน `analogRead()`, `analogWrite()` และการใช้งาน PWM
- การจัดการเวลาเช่น `delay()`, `millis()`, `micros()`

## ปฏิบัติ (2 ชม.)

- จำลองการอ่านค่าจาก Potentiometer (ตัวต้านทานปรับค่าได้) บน WOKWI
- เขียนโปรแกรมควบคุมความสว่าง LED โดยใช้ค่าจาก Potentiometer บน WOKWI
- ทดลองสร้าง LED Fading (ค่อยๆ สว่างขึ้น/หรือลง ด้วย PWM)



## 6 เริ่มต้นใช้งาน ESP32 จริง, Digital I/O

### ทฤษฎี (2 ชม.)

- โครงสร้างขาของ ESP32 (GPIO pins)
- ข้อควรระวังในการต่อวงจรกับ ESP32 (แรงดันไฟฟ้า, กระแส)
- การใช้ VSCODE เพื่อสร้างโปรแกรมสำหรับ ESP32
- การอัปโหลดโปรแกรมจาก VSCode ไปยัง ESP32

### ปฏิบัติ (2 ชม.)

- ทดลองต่อ Digital I/O เข้ากับบอร์ด ESP32
- การอัปโหลดโปรแกรมและทดสอบบน ESP32 จริง

## 7 ESP32 Analog I/O และ PWM

### ทฤษฎี (2 ชม.)

- ADC ของ ESP32 ความละเอียด, ช่วงการวัด
- PWM ของ ESP32 การกำหนดช่องสัญญาณ, ความถี่, Duty Cycle

### ปฏิบัติ (2 ชม.)

- ทดลองต่อ Digital I/O เข้ากับบอร์ด ESP32
- การอัปโหลดโปรแกรมและทดสอบบน ESP32 จริง

## 8 การสื่อสารแบบอนุกรม (Serial Communication)

### ทฤษฎี (2 ชม.)

- หลักการสื่อสารแบบ Serial (UART, Baud Rate)
- การใช้งาน Serial Monitor เพื่อสื่อสารระหว่าง ESP32 กับ คอมพิวเตอร์

### ปฏิบัติ (2 ชม.)

- เขียนโปรแกรมส่งค่าจาก ESP32 ไปยัง Serial Monitor
- เขียนโปรแกรมรับค่าจาก Serial Monitor เพื่อควบคุม LED หรือ ส่งคำสั่งอื่นๆ

# 9 การขัดจังหวะ (Interrupt)

## ทฤษฎี (2 ชม.)

- แนวคิดการขัดจังหวะ: ทำไมต้องใช้, ประเภทของ Interrupt (External, Timer)
- ข้อดีของการใช้ Interrupt เมื่อเทียบกับการ Polling
- ข้อควรระวังในการเขียน ISR

## ปฏิบัติ (2 ชม.)

- ทดลองสร้าง External Interrupt โดยใช้ปุ่มกดเพื่อเปลี่ยนสถานะ LED หรือนับจำนวนการกด
- การเขียน ISR อย่างง่าย

# 10 การเชื่อมต่อหน่วยความจำ (Memory Interfacing)

## ทฤษฎี (2 ชม.)

- ประเภทของหน่วยความจำในไมโครคอนโทรลเลอร์ (Flash, RAM, EEPROM)
- การใช้งานหน่วยความจำแบบ Non-volatile (เช่น Preferences ใน ESP32) สำหรับเก็บค่าที่ไม่ต้องการหายเมื่อปิดเครื่อง

## ปฏิบัติ (2 ชม.)

- ทดลองเก็บค่าสถานะ LED หรือค่าตัวเลขง่ายๆ ลงในหน่วยความจำของ ESP32
- อ่านค่าที่เก็บไว้เมื่อเปิดเครื่องขึ้นมาใหม่

# 11 การเชื่อมต่ออุปกรณ์ I/O อื่นๆ (I2C)

## ทฤษฎี (2 ชม.)

- หลักการสื่อสารแบบ I2C (Inter-Integrated Circuit)
- Master-Slave concept, Address
- การใช้งาน Library สำหรับ I2C

## ปฏิบัติ (2 ชม.)

- ทดลองเชื่อมต่อและอ่านค่าจากเซ็นเซอร์ I2C
  - เช่น DHT11/DHT22 (อุณหภูมิ/ความชื้น) หรือ Gyro/Accelerometer

# 12 การเชื่อมต่ออุปกรณ์ I/O อื่นๆ (SPI)

## ทฤษฎี (2 ชม.)

- หลักการสื่อสารแบบ SPI (Serial Peripheral Interface)
- Master-Slave concept, CS (Chip Select)
- การใช้งาน SPI Library

## ปฏิบัติ (2 ชม.)

- ทดลองเชื่อมต่อและใช้งานโมดูล SD Card หรือหน้าจอ LCD/OLED แบบ SPI
- การเขียนข้อมูลลง SD Card หรือแสดงข้อความบนหน้าจอ

# 13 การหาข้อผิดพลาดและการแก้ไข

## ทฤษฎี (2 ชม.)

- เทคนิคการ Debugging เบื้องต้น (Serial Print, LED indicators)
- ข้อผิดพลาดที่พบบ่อยในการเขียนโปรแกรมไมโครคอนโทรลเลอร์
- การแก้ปัญหาจาก Error message จาก Compiler/IDE

## ปฏิบัติ (2 ชม.)

- การ Debug โปรแกรมที่มีข้อผิดพลาดที่เตรียมไว้
- ฝึกวิเคราะห์ปัญหาจากข้อความ Error และพฤติกรรมของฮาร์ดแวร์



# 14 การประยุกต์ใช้งานเบื้องต้น

## ทฤษฎี (2 ชม.)

- ภาพรวมของการประยุกต์ใช้ไมโครคอนโทรลเลอร์ในชีวิตประจำวัน (IoT, ระบบควบคุมอัตโนมัติ, หุ่นยนต์)
- แนะนำแนวคิดของการออกแบบโปรเจกต์
- ตัวอย่างโปรเจกต์ง่ายๆ ที่สามารถทำได้ด้วย ESP32

## ปฏิบัติ (2 ชม.)

- โครงการย่อย 1 (Mini Project 1)
  - เช่น ระบบวัดอุณหภูมิและแสดงผลบน Serial Monitor/LCD
- นักศึกษาเริ่มวางแผนและออกแบบโปรแกรม

# 15 การประยุกต์ใช้งานและโครงงานย่อย

## ทฤษฎี (2 ชม.)

- เทคนิคการออกแบบวงจรสำหรับโปรเจกต์
- การเลือกส่วนประกอบที่เหมาะสม
- การเขียนผังงาน (Flowchart) หรือ Pseudocode สำหรับโปรเจกต์ที่ซับซ้อนขึ้น

## ปฏิบัติ (2 ชม.)

- ดำเนินการโครงงานย่อย 1 ต่อ

# Course Learning Outcomes (CLOs)

- CLO-1 มีวินัย ตรงต่อเวลา และความรับผิดชอบต่อตนเองและสังคม
- CLO-2 มีภาวะความเป็นผู้นำและผู้ตาม สามารถทำงานเป็นทีมและสามารถแก้ไขข้อขัดแย้งและลำดับความสำคัญ
- CLO-3 มีความรู้และความเข้าใจเกี่ยวกับหลักการและทฤษฎีที่สำคัญในเนื้อหา
- CLO-4 สามารถประยุกต์ความรู้และทักษะกับการแก้ไขปัญหาได้อย่างเหมาะสม
- CLO-5 สามารถให้ความช่วยเหลือและอำนวยความสะดวกแก่การแก้ปัญหาต่างๆ ในกลุ่มทั้งในบทบาทของผู้นำผู้ร่วมทีม
- CLO-6 มีทักษะในการใช้เครื่องมือที่จำเป็นที่มีอยู่ในการทำงานที่เกี่ยวข้องกับไมโครคอนโทรลเลอร์

# Grading policy

ช่วงคะแนน	80-100	ได้ผลการเรียน	A
ช่วงคะแนน	75 - 79	ได้ผลการเรียน	B+
ช่วงคะแนน	70 - 74	ได้ผลการเรียน	B
ช่วงคะแนน	65 - 69	ได้ผลการเรียน	C+
ช่วงคะแนน	60 - 64	ได้ผลการเรียน	C
ช่วงคะแนน	55 - 59	ได้ผลการเรียน	D+
ช่วงคะแนน	50 - 54	ได้ผลการเรียน	D
ช่วงคะแนน	0 - 49	ได้ผลการเรียน	F

# หมายเหตุ

- เนื้อหาในแต่ละสัปดาห์สามารถปรับเปลี่ยนได้ตามความเหมาะสมและความเข้าใจของนักศึกษา
- มีการผลิตบอร์ด PCB ขึ้นมาใช้ในการเรียนการสอน เพื่อลดความเสียหายหรือสูญหายของอุปกรณ์ต่าง ๆ ซึ่งต้องใช้ระยะเวลาประมาณ 1 เดือน
- ทำให้ต้องเรียนบน WOKWI ไปพลางก่อน
- อาจใช้เวลาเรียนบน เรียนบน WOKWI นานขึ้น ถ้าบอร์ดยังไม่พร้อมใช้