

## **Introduction to Device Programming for IoT: C Edition**

**Modality:** Self-Paced Learning

**Duration:** 12 Hours

**SATV Value:**

**CLC:**

**NATU:**

**SUBSCRIPTION:** Learn, Master

### **About this course:**

This course is part of the Microsoft Professional Program Certificate in IoT.

We all know that smart devices play a big role in our day-to-day lives, and that more devices pop up every day. Have you considered that there needs to be a developer behind the scenes for every one of those devices? If you're ready to change the world by converting the envisioned gadgets of tomorrow into digital reality, join this course and take your first steps toward programming for the Internet of Things.

Learn the basics of embedded device programming by implementing various projects on Raspberry Pi and MXChip AZ3166 devices (the AZ3166 is Arduino-software compatible). Learn how to write the software that controls the hardware (temperature sensors, photo cells, and more), and get started making a difference with procedural programming.

Begin with an intro to embedded programming, exploring embedded device types and the inputs and outputs for devices and sensors. Then work your way through a series of hands-on lab projects that teach you how to develop the hardware/software interface, help you to understand C programming for embedded devices, and explore basic data management for a resource-constrained device. From there, dive into embedded solutions that use your own circuit designs to solve real-world problems.

After completing this course, students will be able to develop embedded device systems that capture and process data from analog and digital sensors commonly found in IoT solutions.

**NOTE:** The lab project work in this course is based on the hardware found in the Microsoft IoT Pack for Raspberry Pi and the MXChip IoT DevKit. You may want to purchase in advance of starting the course.

### **Course Objective:**

- Describe the characteristics of an embedded device
- Configure the development environment for an embedded device
- Create a simple program that accesses GPIO pins
- Implement features of the C programming language in a constrained resource environment
- Configure I/O libraries and take pinout readings
- Develop a simple embedded device application

- Explain how to construct simple circuits using common electrical components
- Describe the sensor and device resources available in the marketplace
- Implement sensors within an application using SPI and I2C
- Explain how to develop a solution that uses multiple sensors
- Build the circuit for a solution
- Develop a software/hardware solution for a common scenario

## **Audience:**

- IoT Engineers
- IoT developers

## **Prerequisite:**

- Some basic programming experience is required.
- Some knowledge of the C programming language will be beneficial but is not required.

## **Course Outline:**

### **Introduction Embedded Device Programming**

- Preparing Your Dev Environment
- Developing Your First Application
- Module 1 Assessment

### **Data and Device Inputs**

- Building a Binary Counter
- Getting Started with the MXChip AZ3166 Device
- Module 2 Assessment

### **Circuit Components and Sensor Hardware**

- Building Simple Circuits
- Building a Temperature Gauge
- Module 3 Assessment

### **Real-World Solutions**

- Building a Pho Cell Device
- Comparing Digital and Analog Inputs

### **Final Evaluation**

- Final Assessments