

ADVANCE IOT

INTRODUCTION

- What is IoT and Working with IoT
- How does IoT work?
- Building Blocks of IoT
- What are the words I need to know?
- Why IOT (Opportunities & Application of IOT)?
- Challenges in IoT and Understand IoT design solution

LINUX

- Linux Concepts
- Linux Architecture
- Introduction to components of I/O architecture
- Objectives of Linux I/O model
- Virtual file system
- File system services
- Accessing the command line (terminal and desktop)
- Accessing and using manual pages
- Working with the command line and the shell

- Piping and redirection
- Linux OS
- Different Linux commands like cp , mv mount
- Introduction to VI editor. VI editor settings
- Creating script
- Shell variables conditions (if else)
- Shell control structures
- Shell programs to read command line parameters
- Linux lab for shell programming
- Autostart Scripts

PYTHON

- Python Basics and programming
- using python on Raspberry Pi
- Data types in python
- using variables in python
- Casting, string operations
- use of Python Tuples
- use of operators in python
- python looping in linux
- functions used in python
- Classes and objects

- File handling in python

ARDUINO

- Getting Started with Arduino
We get you up and running with the Arduino platform. You will install the tools, write and upload your first piece of code to run on a microcontroller.
- Arduino Code Language (C Refresher)
In case your C code skills are rusty, this video provides a quick refresher. We also look briefly at the differences between plain C and the Arduino Code Language.
- Inputs and App Control
We build a slightly more complex program for a microcontroller. This one feature user inputs via button and a way to control the operation of the microcontroller from an app running on your smartphone.
- Arduino Simulation Environment
Is there a way to run microcontroller code without having one at your disposal? In this video, we explore simulation technologies that allow you to build rapid prototypes, giving you greater insight into the working of your code.

Sensors, Signals & Electronics:

- Introduction to Open Source platform
- An Overview of Open Hardware
- Arduino Board Description
- Getting started with the Arduino IDE writing your first program on Arduino
- **Analog Sensor**
We begin our exploration of sensors by learning how to read analog signals from them in our microcontroller code. We also look briefly at some issues while using analog sensors.

- Basic electronics refresher
A quick overview of all the components you will need as you build your Arduino projects, their usage and functions.
- **Digital Sensors**
Learn to read ambient temperature and humidity from an integrated sensor by talking to it using digital signals. Interfacing of Analog Devices with Digital World
- Digital Signals
We delve into the representation of digital signals on the wire and how signals are represented and transmitted between sensors and controller.
- Sensors overview
We covered a few sensors in detail in the past videos, but there a wide variety of sensors out there that measure all manner of real world phenomena. This video provides a broad overview of commonly available sensors and sensor modules that play well with the Arduino hardware.
- Difference between Parallel and Serial Communication
- Communication Protocols
- USART / UART Protocol
- I2C Protocol
- SPI Protocol
- RS232 Standard
- TTL Converter
- UART Programming
- GSM Interfacing

Microcontroller Concepts

- User Interface
Buttons and displays: IoT devices often require a "human-machine interface" and this is often hard to build due to microcontroller

limitations. In this video, we look at buttons and display modules and explain how these are controlled from microcontroller code.

- **Microcontroller Concepts – Memory Map, IO Peripherals**
We have been doing a lot of microcontroller coding. Now it is time to look under the hood and understand the internal architectures of these machines. We also look at how code and data are organized.
- **Arduino Hardware Overview**
We have been using one microcontroller platform – the Arduino Nano – but there are several others out there. Here, we take a walk through the world of popular Arduino hardware – both official and community-supported.

Basic Networking with nodemcu

Intro to nodemcu

This is the first look at the second of our two microcontroller platforms – the nodemcu. This is a true IoT enabled microcontroller, since it has built in Wi-Fi. We jump right in and write our very first program for it and control an LED from the cloud!

- **Basic Wireless Networking**
Time to take a step back and look deeper into how Wi-Fi works. This video teaches you the basics of networking, Wi-Fi architecture and security.
- **Sensor Log**
Here are more advanced exercise using the node mcu. We connect a few sensors to it and build a sort of mini weather station.

HTTP and REST API

Networks allow us to transport data from one point to another. But how is one device controlled by another? This video looks at the client-server architecture and in particular at HTTP and RESTful interfaces.

Advanced Micro controller Concepts

- **LED blink using MQTT**
In this video, we introduce MQTT, a lightweight yet powerful protocol for messaging. We use it to build an example, which, while similar in functionality to an example we built before, is transparent in terms of its underlying architecture.
- **Messaging and MQTT**
Here, we take a deeper look at messaging and the “pub-sub” pattern. We discuss how it is different from REST and why it is often preferred in the case of IoT devices.
- **Microcontrollers Concepts – Interrupts, Timers, Callbacks, Re-entrancy, Watchdogs**
In this video on micro controller internals, we look at common design patterns used in building firmware. We also go over basic concepts like re-entrancy, mutexes and error handling. We also look at performance aspects – latencies, throughput and other parameters.
- **Device Configurations, Discovery and OTA**
Time to take a leap from building a demo to building a product. What should you worry about as you put your IoT device into the end-user’s hand? How is the product deployed, discovered, secured and maintained?

Raspberry pi :

Getting Started with Raspberry Pi

- **Introducing the Raspberry Pi**
Get up and running with the Raspberry Pi 3. Learn to make the connections, power it up and take a tour of its desktop environment.
- **Course Objectives and Course Map**
Understand the overall plan for this module. We provide a quick overview of what we’ll learn and of the major projects we’ll be doing.
- **Booting the Raspberry Pi 3**
A more fundamental look at booting the Raspberry Pi 3. Understand how to download an operating system, format an SD card and boot the OS. Also, learn the rudiments of the file system.
- **OS Options Overview**
While we’ll use the Raspbian image as the reference for the rest of this course, we spend some time in this video looking at what other OS

options are available for the Raspberry Pi, and why you might want to check them out.

- Camera Interfacing : Application Imaging, Video Recording,

Interfacing Hardware with the Raspberry Pi

- Raspberry Pi Remote Access
Learn to operate the Raspberry Pi in “headless mode” by logging into it remotely. We’ll also discuss how to move data (or files) from your PC to the Raspberry Pi and back.
- Bash Command-line
A short primer on Linux terminal commands, or more specifically, a quick tour through the basic command set for the bash shell. Get comfortable operating your Raspberry Pi without needing a GUI interface.
- Raspberry Pi LED Blink Example
Time to bring out the electronic components. The Raspberry Pi’s connector allows you to extend its functionality using external hardware and in this video, we begin to explore how this is done.
- Linux Overview and Device Drivers
High-level operating systems, such as Linux, require the use of a device driver for user programs to access hardware. We look at why this is, what a device driver does, and how to build one.

Google firebase

- Connecting device to Google Firebase
- Creating real time database in google firebase
- Creating Web server using Firebase
- HTTP and MQTT On firebase
- Appliance control through google assistant
- use of Amazon Alexa on Raspberry Pi

Machine-to-Machine Communication

- **Sensors Integration**
We continue the journey we started in video and interface yet more hardware to the Raspberry Pi. We learn how to deal with inputs and digital protocols from within user programs.
- **Local Blynk Server**
Blynk provides a simple means to tie together smartphones and IoT gadgets. In this video, we control a NodeMCU from a smartphone via a Blynk server running on Raspberry Pi.
- **Node-red: M2M and Gateway**
We expand on the previous example and use it as a basis to discuss machine-to-machine communication, an IoT node vs gateway and related concepts.

Multimedia Concepts

- **Raspberry Pi Media Server**
This video introduces a high-performance application on the Raspberry Pi – a complete content streaming server.
- **Running Services**
Learn about the Linux boot up sequence, run levels, and how to run programs automatically on boot up.

Speech Processing Concepts

- **Voice Control Application**
Explore speech processing on Raspberry Pi. Speech recognition is an important emerging interface for IoT devices and this example introduces the topic.
- **Voice Control System Architecture**
A more detailed look at how the last example was put together. Once again, we look at performance, introduce metrics, and compare with video processing.
- **Amazon Alexa tool & Google voice assistant:**
- **Raspberry Security Cam**
We have covered video playback and speech processing. Now, its time to learn video processing. In this video, we set up our example and demonstrate how it works.

Amazon AWS IOT:

- We move to IaaS with Amazon's cloud offering – AWS.
- This means creating your own server instance in the cloud, choosing and configuring its OS, and running your service.
- AWS IoT core services, AWS Message Brokers
- AWS Elastic Compute Cloud ec2 instance, Server deployment in AWS
- Case Study on Home Automation
In this first of our case studies, we examine what an ideal home automation system will look like and how it will be organized. We'll talk about the nodes, the gateway and the cloud services that such a system would need.

Connectivity beyond Wi-Fi

- Case Study on Smart City
IoT technology is relevant for more than just consumer goods. This video looks at how we might go about building smart city solutions.
- BLE
Connectivity at ultra-short ranges can be just as important as those at longer distance when it comes to certain IoT applications. We look at bluetooth and other radio frequency alternatives.

Post completion of the course, you will be able to:

- Develop and implement IoT system, solutions, and applications.
- Good understanding of reference architecture, like- Network, Device, Application & Client server vs Public Subscribe.
- Design IoT Device – sensors, embedded development boards, Interfacing peripherals & programming GPIOs & Design Considerations.
- Work with IoT Communication Protocols like- wired, wireless, Networking & Application
- Write applications using programming Languages