

# Embedded Systems Development

## Task 1.1P: Switching ON Lights of the Assisted Living - Modular Programming Approach.

As outlined in Week 1, modular programming is an approach used to separate the functionalities of a complex program into independent and manageable chunks of code. In this task, we will follow a modular programming approach to switching ON lights of the smart elderly care home.

### Hardware Required

- Arduino Nano 33 IoT Device all set up and ready to go
- Micro-USB to USB-A cable
- A mobile phone with Arduino Cloud app installed.
- WiFi enabled device like a laptop or a mobile phone
- LEDs
- Push button, slider switch
- Breadboard and jumper wires
- Resistors

### Notes:

- If you are going to use your device on campus, your device will not be able to connect to eduroam by default. Instead, please use Wi-Fi at home, or use your mobile/laptop hotspot when connecting the Arduino device to the Internet.
- you might be asked to update the firmware on the software the very first time you connect it to the Internet, in case you are using your mobile phone as the hotspot.

### Software Required

Web browser IDE/ Arduino IDE 2

### Pre-requisites

- Follow the instructions provided in “**Setting up the Arduino**” to complete the setting up of the Arduino Nano 33 IoT.
- Create GitHub repository and add your tutors

# Embedded Systems Development

## Task Objective

### Problem Statement:

"Linda usually attends the evening art and craft, music, and bingo sessions at the retirement village. During winter however, by the time she gets home, it's very dark. She has mentioned it would be lovely to have the lights of the porch and the entryway of the house light up when she gets back home so that she doesn't have to fumble through her purse to find the keys and get enter the house safely as well."

### Proposed Solution:

Design a system to switch ON the porch light and the hallway light when Linda presses the switch/push button. The porch light should stay ON for 30 seconds and the hallway one for 60 seconds.

- Arduino group has provided a nice example on how to write a program to make a Nano 3 IoT device blink its LED on pin D13.
  - Read the tutorial available here: [LED Blink](#)
  - Note that the LED pin could be different in other Arduino devices.
- Modify the code to light up the porch and hallway lights of Linda's house.

## Task Submission Details

1. Create a folder, Task1.1, in your SIT210/730 GitHub repository. The folder should contain the code file, **Task1.1Lights\_ON** and readme file. The Readme file should include a short description of the system you are building and explanations on what each part of your code is doing. Please note we do not require line by line descriptions, but rather general overview of what different functions of the code is doing. code Include the link to your repository here.
2. Include a schematic of your circuit. You can use a tool such as Tinkercad for this purpose.
3. Take a short video of your Arduino board with the system working. Upload it to YouTube or Deakin Panopto and Include the link here.
4. Describe how you have used modular programming to improve the usability of your program. Please note, generic explanations of modular programming taken from GenAI will not be accepted as an answer. You will need to relate to your task requirements and explain how modular programming was useful.
5. From a design perspective, do you see any problems with the solution? Discuss.
6. Considering the smart assisted living you are planning to implement in this trimester, what other modular programs would you consider writing. Give three examples and explain your reasons in using them.
7. **For SIT730 Students Only**
  - Investigate a day-to-day embedded system that has ARM processor and explain how it is used in the given application. You can include details such as what additional components are connected to this processor in this embedded system and

# Embedded Systems Development

- Outlines some of the recent methods/frameworks used to design embedded systems.

## Important:

- Please note explanations, diagrams, or code directly taken from GenAI is NOT accepted in this unit. The tasks submitted should be your own work in accordance with the academic integrity policies of Deakin. Task submissions that do not adhere to [Deakin Academic Integrity Guidelines](#) will be reported to the Academic Integrity Team for further action. Please refer to [Ethical and Evaluative Use of GenAI](#) for more information. If you have used GenAI in certain parts of your task and not sure if that is allowed, please talk to the unit chair.
- If you need any clarifications on the task requirements or require help to complete the tasks, please contact your teaching team.
- Anytime you submit a task to OnTrack, it is a good practice to check the status of any existing tasks, and the future tasks you are expected to complete. If you have got feedback on previous tasks, you may need to fix and resubmit some of your work. You want to check out why, so that you can learn from this and make it faster and easier to accomplish later work to the required standard.