# Embedded Systems Development

## Task 3.3D Particle - Buddy system

Publish subscribe is a method of communication between two or more devices. In this task, you will be using the basics of publish subscribe to communicate between two Particle Devices.

**Please note** that this is a Distinction task. You should expect that the complexity of tasks and amount of research and time you have to put in each would increase the higher the level of each task gets. As such, the instructions on credit and above tasks will be less like step by step instructions, and more on setting the requirements and the submission.

## Hardware Required

Particle device (Photon/Argon/etc.)

## Software Required

Particle web IDE

## Pre-requisites: You must do the following before this task

1) Previous Particle tasks (1.2P, 2.1P, 3.1P)
2) Read the tutorial about Publish and Subscribe: https://docs.particle.io/guide/getting-started/examples/photon/#the-buddy-system-publish-and-subscribe

## Task Objective

In this task, you will build an embedded system that allows you to receive virtual 'waves' from a buddy, and react to it using Particle and the publish subscribe structure.

### Steps: Argon/Photon

1) Build an embedded system using your Particle device and 1 LED light.
2) We have setup a Photon at RIoT lab that sends "wave" signal to any registered Particle device. It is intended to simulate a Photon buddy that supposedly detects a waving motion using the motion sensor; i.e. when someone moves their hand left and right in front of the sensor.
   a) This photon publishes using following command every minute or so:
      Particle.publish("Deakin_RIOT_SIT210_Photon_Buddy", "wave");

3) Create a handler method that whenever your system received a "wave", the system will flash the LED 3 times.
4) Test out the system.

## Task Submission Details

Q1: Submit a video that shows the outcome of the task. Include the link here.

Q2: Create a repository named SIT210-Task3.3D-CloudFunction on your Github account. Upload your code to the repository. Include the link to your repository here.

Q3: The buddy Photon sends a "pat" signal as well as "wave". Modify your code for the "pat" handler to flash the LED in a different way to a "wave".

*Remember, 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.*