



OSC PIHUB conceptual network overview

The OSC PIHUB implementation adds a sequencer, cue list and complex behavior functionality as well as a framework for live coding multi-robot interactions across varying control platforms.

Each robot will require a working PIBOT control and then in addition a central controller, the PIHUB, is required to act as a sequencer and sync source. Rather than sending messages directly to the individual Pi on a bot, the control messages are sent to the HUB Pi. The HUB can maintain a cue list of detailed actions that can be edited live via Clojure, and triggered as a group via an OSC to the address /pihub/set-scene

The PIHUB is built on a modular basis to allow for each individual bot to have its own control architecture - e.g. ROS2, iRobot, TurtleBot, and other architectures. IRobot Python support is complete and works reliably, ROS2 functionality is included but suffers from significant latency on the iROBOT platform, and contributions to extend the functionality are actively sought.

Basic Requirements:

- 1) At least one OSC PIBOT setup consisting of a Bot and Pi;
- 2) At least one OSC control source that can send an OSC message like Max-MSP, Touch-Designer, Isadora, Qlab, Supercollider, Touch-OSC, LEMUR, or any software that can format and send an OSC message. Multiple controls are possible, however current development status assumes a first in -first out command execution and does not provide any method to prioritise one control source over another. Examples are available in Max-MSP, Isadora, and QLab (pending).
- 3) A raspberry pi capable of running ubuntu 22.04 or later, desktop suggested for ease of configuration, with required libraries installed. Tested on a 4gb Pi 4, and Pi 5.
- 3) A robot that will communicate using the iROBOT Python SDK (ICreate 3, Root) or with with additional configuration any robot using ROS2 humble

