Diagram boxes:
- Control computer or tablet (OSC source - max-msp for example)
- OSC Message exchange via wifi
- PI running OSC PIBOT framework
- Robot Control via BLE SDK or ROS2
- I2C and GPIO PWM control of lights, motors etc
- ICreate 3 or Root (bluetooth) or ROS2*
- status messages

OSC PIBOT basic implementation

The basic OSC PIBOT implementation requires the following for a basic implementation:

1) An OSC 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.

Examples are available in Max-MSP, Isadora, and QLab (pending).

2) A raspberry pi capable of running ubuntu 22.04 or later, desktop suggested for ease of bluetooth / wifi configuration, with Python 3 and the required libraries installed. Tested on a 4gb Pi 4, and Pi 5. If comfortable using SSH or xfce desktop install, a Pi Zero 2 W is sufficient to run the PIBOT framework.

3) A robot that will communicate using the iROBOT Python SDK (ICreate 3, Root) or with with additional configuration any robot using ROS2 Humble. Note that ROS2 support is still in testing stages and suffers from multi-second lag for action servers on the iCreate 3. Other ROS2 implementations have not been tested.

4) Optionally additional I2C, I2S, and GPIO connected devices, such as for PWM of lighting, motor control, and sensor input. The PiBOT assembly board uses 16 channels of PWM, plus an I2S audio amp, I2S audio input and GPIO sensor breakouts