Interconnected Robot Network User Guide

Brandon O, Caleb Wheeler, Clay Scarboro, Tanner Boylan

Introduction

The interconnected robot network, or IRN for short, is a collaborative project between NC State's Electrical and Computer Engineering senior design program and Texas Instruments. The goal of the IRN is to utilize multiple TI-RSLK to demonstrate swarm functionality. By utilizing an expanded TI-RSLK localization apparatus and Wi-Fi connection, the IRN consists of multiple TI-RSLK moving to form an equilateral triangle, and then simultaneously travel around the triangle's vertices in a circle.

Setup

The following materials are required for IRN Functionality:

- 3 TI-RSLK
- 5 ESP32UWB
- 3 custom PCBs
- 18 AA batteries
- 2 5V AC power adapters with micro usb connectors (power other 2 dwm1000)
- Personal computer
- Tape measure

To begin IRN setup, we will begin with the Ti-RSLK and Anchors

1. Attach the PCB onto the TI-RSLK's MSP432 with the 2 2x10 Pin headers
2. Attach the ESP32UWB Module onto the Custom PCB with the 1x12 and 1x20 Pin headers
3. Place 6 AA Batteries in the TI-RSLK on the underside of it's chassis
4. Install caster wheels on the TI-RSLK chassis (optional)
5. Repeat steps 1-3 (and optionally 4) for each robot
6. Set 2 ESP32UWB 81 inches apart from one another. These 2 modules will serve as the northern bound for our operating theater.

Next, we will program our microcontrollers, the MSP432 and the ESP32UWB

MSP432:

1. Download and Install Code Composer Studio: https://www.ti.com/tool/CCSTUDIO#downloads
2. Open the following Github repository in Code Composer Studio: https://github.com/cbwheele/Interconnected_Robot_Network
3. Navigate to Interconnected_Robot_Network/MSP432_Code/IRN_TI_RSLK/Final_Robot_Code.c

4. Download the code onto the microcontroller via a USB connection from computer to MSP432

ESP32UWB (TI-RSLK):

1. Open Arduino IDE
2. Navigate to https://github.com/cbwheele/Interconnected_Robot_Network/blob/main/ESP32_Code/IRN_Robot_ESP32/IRN_Robot_ESP32.ino
3. Navigate to Tool tag
4. Choose board type as ESP32 Dev
5. Connect ESP32UWB to PC
6. Choose Port from Tool tag
7. Press the arrow button to flash the code onto the board (might need to press the flash button on the board to successfully flash the code)

ESP32UWB (Anchor):

1. Open Arduino IDE
2. Navigate to https://github.com/cbwheele/Interconnected_Robot_Network/blob/main/ESP32_Code/anchor/anchor.ino
3. Navigate to Tool tag
4. Choose board type as ESP32 Dev
5. Connect ESP32UWB to PC
6. Choose Port from Tool tag
7. Press the arrow button to flash the code onto the board (might need to press the flash button on the board to successfully flash the code)

Finally, the python script must be configured.

1. Download the script via github: https://github.com/cbwheele/Interconnected_Robot_Network/blob/main/Python_Controllers/robot_control.py
2. **Edit line 8 of the script with your computer's ip address**.
3. Install python3
4. Open a terminal (type "cmd" into windows search)
5. Change directories with:
   "cd …/Interconnected_Robot_Network/blob/main/Python_Controllers/robot_control.py"
6. Run the script with the command:
   Windows: "python3 robot_control.py"
   Mac: "sudo python3 robot_control.py"

This completes the setup process, and the IRN is ready to be executed.

Test and Debug

Now that the IRN is set up, it can be tested and debugged. We'll begin by powering up all of the modules.

1. Place all three TI-RSLK in the operating theater
   By placing robots within the two anchors, location information will be more accurate.
2. Power both anchor ESP32UWBs
3. Run the python script
4. Turn on each TI-RSLK via the buttons on the MSP432
   A red LED will flash while the Robot connects to wifi, and will glow green when it's connected. If the red LED flashes for over 15 seconds, restart the Robot with another button press
5. Verify that the python script accepted the connection
   Continue to restart robots until the script verifies they are all connected
6. Insert the coordinates (prompted from python script) for each vertex of the desired triangle
   DWM1000's inaccuracy and inconsistency can interfere with the robot's ability to reach its destination coordinates. In this event, manual driving mode will be triggered (indicated by a green and red LED on the custom PCB). Use WASD controls to move the robot to desired location
7. Repeat step 6 for each robot
8. Run the robots in a circle. They will stop when they return to their original location.

Maintenance

As you continue to use the IRN, keep in mind that upkeep and regular maintenance is important. Always make sure that your TI-RSLKs are powered with charged batteries. Additionally, the peripherals of the TI-RSLK (bumper switches, encoders, etc…) should be inspected regularly to confirm they're functioning.