TL101 - Drone System User Manual

Table of Contents

1. Introduction
2. Safety Guidelines
3. Included Components
4. Battery and Power Setup
5. Flight Controller and Receiver Setup
6. Manual and Autonomous Flight Operation
7. Maintenance and Inspection
8. Receiver switch summary table
9. PC and Drone Software setup

---

1. Introduction

This manual provides guidance for operating the ICON Lab indoor drone system. It outlines power-up, flight operation, safety procedures, and maintenance responsibilities.

---

2. Safety Guidelines

- Always operate in open indoor spaces
- Never touch spinning propellers.
- Maintain a line of sight during flight.
- Land immediately if unexpected behaviour occurs.
- Do not fly with damaged propellers or loose hardware.

---

3. Included Components

- Fully assembled drone
- Radiolink AT9S Pro transmitter
- Battery and charger

---

4. Battery and Power Setup

Slide the 4S 2200 mAh battery into the top mount's cavity for a secure friction fit. Connect the battery to the XT60 connector on the power module and wait approximately 30 seconds for system initialization. Two beeps will sound when ESC initialization is complete.

Battery voltage is displayed in real time on the connected host computer. The user must manually monitor this voltage during flight. At 14.0 volts, the operator should begin preparing to land. At 13.5 volts, the battery level is critically low and the drone must be landed immediately. Maximum flight time is approximately 7.5 minutes.

---

5. Flight Controller and Receiver Setup

The Holybro Kakute H7 Mini is preconfigured with PX4 firmware. The R12DSM receiver communicates using SBUS. All ESCs are configured with DShot600 protocol, allowing motor direction to be set in software.

Figure 1: Radiolink AT9S Pro transmitter layout

Switch assignments are as follows:

- **SwD** controls arming and disarming. Flip up to arm the drone, down to disarm.
- **SwE** toggles autonomous hover mode. When flipped toward the user, auto-hover is enabled; when in the middle or away, the drone is in manual mode.
- **SwG** starts and stops autonomous mapping via the Auto-MUS command mode. Flip toward the user to begin mapping, away to return to hover.
- **SwF** functions as the kill switch. When flipped toward the user, it immediately stops all motors.
- **Sticks (Channels 1–4)** control throttle, roll, pitch, and yaw.

---

6. Manual and Autonomous Flight Operation

6.1 Pre-Flight Checklist

Ensure the battery is fully charged and seated securely in the mount. Confirm that all propellers are secure and free of obstruction. Power on the transmitter before connecting the battery. Place the drone on a level surface before arming.

Switch Pre-Flight Positions:

- **SwD** (Arm/Disarm): Up (Disarmed)
- **SwE** (Auto-Hover): Away from user (Manual mode)
- **SwG** (Command Control): Away from user (Idle)
- **SwF** (Kill Switch): Away from user (Normal operation)

6.2 Powering On and Arming

After powering the drone as described in Section 4, wait for the startup beeps. Flip **SwD** up to arm the drone. The propellers will begin spinning at low speed.

6.3 Manual Flight

Use the transmitter joysticks to fly manually once the drone is armed. Raise the drone to approximately one meter in height before switching to autonomous modes.

6.4 Autonomous Mode

Flip **SwE** toward the user to activate autonomous hover. Once the drone stabilizes, flip **SwG** toward the user to begin mapping using Auto-MUS command mode. Before the transition, ensure that both server and client scripts are running on both the remote client laptop and the Raspberry Pi 4. See section 9.

6.5 Ending Flight

To stop mapping, flip **SwG** away from the user. Flip **SwE** away to return to manual mode. Descend the drone manually. Flip **SwD** down to disarm. Disconnect the battery to power down.

6.6 Emergency Controls

SwF serves as the kill switch. When flipped toward the user, all motors shut off immediately. In the event of a lost signal, the drone automatically enters Land Mode and descends safely.

---

7. Maintenance and Inspection

Before each flight, inspect the propellers for damage. Tighten all screws and fasteners. Verify that ESC wiring is secure and that each motor spins in the correct direction. If the drone behaves unpredictably, recalibrate sensors using QGroundControl.

---

8. Receiver switch summary table

Switch	Function	Towards user	Away from the user
SwD	Arm / Disarm	Arm (props spin)	Disarm (props stop)
SwE	Autonomous Hover Mode	Enable Auto-Hover	Manual Mode
SwG	Command Control	Begin Mapping	Return to Auto-Hover
SwF	Kill Switch	Immediately stop all motors	Normal Operation

Table 1: Receiver switch summary table

9. PC and Drone Software Setup

IMPORTANT

Read the section below very carefully. Any attempt to set the drone into autonomous flight without the software being properly loaded will result in failure to update flight controller commands and will crash the drone. Only make the switch once the software setup has been verified.

9.1 Running the Source files before flying

The drone is designed to independently load its software at boot. Once the drone is connected to the power, wait around 1 minute before running the client script on your PC.

After powering the Drone and waiting for one minute, head over to the PC and open the terminal using

- Ctrl + Alt + Terminal

Once the terminal opens, type in the terminal and head to the directory that contains all the source files using

- cd ELEC491_TL101/icon_drone/shfiles

The terminal will lead you inside the shfiles directory, now to run the client.sh script, type

- source client.sh

This will run the client.sh and connect the Drone to your PC. A premature load up of the client may cause errors. After careful inspection of the client-side terminal for any errors (in red font), verify that the IMU calibration is completed. Failure in calibration will result in loss of autonomous control. Finally, verify that the client setup is complete–this will take another 30 seconds; then, the screen will display the FSM state of the flight controller node and wait for RC signal.

9.2 Setting up WiFi Communication

Before flying the drone, it is IMPORTANT to make sure your PC and Drone are able to communicate with each other.

To make sure the drone and PC communicate, connect both the PC and the Raspberry Pi (on the drone) to the same WiFi system.

- For the PC:
    - Head over to the settings and select the suitable WiFi network, make sure it is a 5G connection
    - Once connected, check the settings of your connection and note down the IPv4 Address (This is the PC's IP address)
- For the Raspberry Pi:
    - Connect your Pi to power using a C-type power cable.
    - Carefully remove the USB connections of the IMU and the mono camera, and connect your own keyboard and mouse.

    - Head over to the settings and select the same WiFi network that was used for the PC, make sure it is a 5G connection.
    - Once connected, check the settings of your connection and note down the IPv4 Address (This is the Raspberry Pi's IP address)

Once the PC and Raspberry Pi are on the same network, the Raspberry Pi will also be accessible via SSH, if the user would like to access the files on the device.

- ssh <username>:@<Raspberry Pi's IP>
- Use the scp command to copy files from the Pi

Once you have the IP addresses of both the PC and the RPi, you will have to update the IP address fields of the shfiles to allow communication between both the RPi and PC.

- For the PC:
    - Head to the shfiles directory by opening a terminal using Ctrl + Alt + T, then in the terminal execute the two commands below in order
        - cd ELEC491_TL101/icon_drone/shfiles (then press enter)
        - nano client.sh (then press enter)
    - Once this is done, the client.sh file will open, and you will see a series of codes within it. Carefully scroll down to the section and enter the IP addresses
        export ROS_MASTER_URI=http://<Raspberry Pi's IP>:11311
        export ROS_HOSTNAME=<PC's IP>
    - To make the changes and exit the script, press Ctrl + X. The terminal will then prompt you to save the changes, press Y, and then enter

- For the Raspberry Pi:
    - Head to the shfiles directory by opening a terminal using Ctrl + Alt + T, then in the terminal execute the two commands below in order
        - cd ELEC491_TL101/icon_drone/shfiles (then press enter)
        - nano server.sh (then press enter)
    - Once this is done, the client.sh file will open, and you will see a series of codes within it. Carefully scroll down to the section and enter the IP addresses
        export ROS_MASTER_URI=http://<Raspberry Pi's IP>:11311
        export ROS_HOSTNAME=<Raspberry Pi's IP>
    - To make the changes and exit the script, press Ctrl + X. The terminal will then prompt you to save the changes, press Y, and then enter

NOTE: Use the correct IP addresses, or else the system will fail to connect, leading to no autonomous flight. Upon connection failure, the client-side terminal will display a warning denoting that it failed to connect to MASTER. Be aware that this error can also occur if the client attempts a connection before the server-side setup (through boot-up, followed by the automatic execution of the server.sh script) is complete. Again, please allow up to one minute for the Raspberry Pi to boot up and finish the execution of the server script.

After all the changes have been made, disconnect the Raspberry Pi's power and reconnect the IMU and Mono-camera. Once that is done, connect the cable that powers the Pi and connect the Drone battery, and

follow steps **9.1**. After updating the IP addresses, there is no need to repeat the steps in **9.2** unless the IP addresses are to change again. To facilitate the process, use static IP settings on the router so that the IPs are not reassigned to new values.