# CONNECTING THE DRONES

1. Install fresh batteries if necessary. Make sure the battery clicks into place. If not, make sure the locking latch on the back is down.
2. Power on the appropriate drones by pressing the power button on the back. The light will begin flashing while the drone boots.
3. Once the light stops flashing, press the button rapidly THREE times. The drone will beep once.
4. About 20 40 seconds late, it will beep again to indicate it has reconfigured the Wifi.
5. Run the max\_clocks.sh script in the /home/hackathon folder. The TX2 fan will turn on and give maximum processing power.
6. Connect the TX2 to the appropriate Wifi network for the testing area. The networks are called “Hack2019A” and “Hack2019B”. The use the same password: hack1337
7. It might take up to another minute for the drones to be fully connect. Check if they are ready by using the appropriate ping\_alpha.sh, ping\_bravo.sh, ping\_charlie.sh scripts. If connected you will see a time in milliseconds for the ping response.

# FLYING THE DRONES WITH THE PROVIDED DEMO.

1. Make sure the contestant has commented out the #define NO\_FLIGHT in the demo source file and re-run make. If this is not done, the drone will connect video but not fly the demo mission.
2. The demo requires TWO drones, alpha and bravo must be connected to the network.
3. Place alpha and bravo near the edge of the testing area.
4. Run source setupEnv.sh in the terminal before running the demo.
5. Launch the demo with ./bebop2Swarm\_arm64 alpha bravo
6. If both drones do not take off, press SPACEBAR to land the drones. Try running the programming again. If SPACEBAR does not land the drones, check to make sure one of the openCV windows is the active window before pressing SPACEBAR.

# FLYING THE DRONES FOR HACKATHON MISSIONS

1. Make sure the contestant has commented out the #define NO\_FLIGHT if they have not removed it entirely.
2. Confirm (ask) the contestant has added to the keyboard controls to takeoff the drones with the ‘t’ key and start their mission with the ‘g’ key.
3. Make sure only the necessary drones are powered on and connected for the mission.
4. Run source setupEnv.sh in the terminal before running the application.
5. Have the contestant run their application and pass the required drone callsigns. If the mission only requires alpha:  
   ./bebop2Demo\_MyApp\_arm64 alpha

For all three drones  
./bebop2Demo\_MyApp\_arm64 alpha bravo charlie.

AFTER THE MISSION DELTE ALL PHOTOS AND VIDEOS FROM THE DRONES. Have the contestant install Filezilla in order to FTP in to <drone IP address>/internal\_000/Bebop\_2/media/

Typos in the name may cause the program to crash. If a program crash occurs the drones will continue to hover. Restart their app and use the SPACEBAR to land them. If drones are stuck hovering and cannot be landed, throw a towel onto the drone to stop the rotors and catch it when it falls.

# REPLACING A DAMAGED PROP

Use the triangular prop tool to grip the studs while twisting to disengage the prop. The studs ride along a slot, where once end the studhead can fit through.

Use the correct prop to replace the broken one. Each motor has a FLUSH pin in the center or a RAISED PIN in the center.

RAISED PIN – use a prop that has a hole in the center for the pin to fit through.

FLUSH PIN – use a prop that has no hole in the CENTER.