Inter IIT Tech Meet 11.0

PLUTO DRONE SWARM CHALLENGE

Task 1 code documentation

Submitted by - : Team 27

FEBRUARY 2023

Contents

1 Package Installation

The following packages are required to be installed for the codes to work (make sure python version is 3.6 or higher):-

- Python library manager

```
$ sudo apt-get update
$ sudo apt-get -y install python3-pip
```

- opencv-contrib version 4.7

```
$ pip install opencv-contrib-python==4.7.0.68
```

- keyboard

```
$ pip install keyboard
```

2 Class: *plutodrone*

It interacts with a drone over a TCP/IP network. The class has several attributes and methods that allow the user to send commands and receive data from the drone.

2.1 Attributes

- TCP_IP : Specifies the IP address of the drone
- TCP_PORT : Specifies the port of the drone
- mySocket: Socket object that allows the class to connect to the drone.
- Roll : Specifies the roll value
- Pitch : Specifies the pitch value
- Yaw : Specifies the yaw value
- Throttle : Specifies the throttle Value
- rcvalues : Bytearray that holds the data for MSP_SET_RAW_RC Packet
- attvalues : Bytearray that holds the data for MSP_ ATTITUDE Packet
- altvalues : Bytearray that holds the data for MSP_ALTITUDE Packet
- cmdvalues : Bytearray that holds the data for MSP_SET_COMMAND Packet
- imuvalues : Bytearray that holds the data for MSP_IMU Packet

2.2 Methods

- __init__
  - Description: This is the constructor method that is called when an instance of the class is created. It sets the default values for TCP_IP and TCP_PORT and initializes a socket connection to the drone. The socket connects to the specified IP and port of the drone.
  - Input: None
  - Output: None
- send
  - Description: This method is used to send the commands to the drone. The commands argument specifies the commands to be sent to the drone.

  - Input:
    * commands - a list of bytes that represent the commands to be sent to the drone
  - Output: None
- receive
  - Description: This method is used to receive the data from the drone.
  - Input: None
  - Output: None
- calcCRC
  - Description: This method calculates the cyclic redundancy check value. It is calculated by XOR operation of 1byte at a time. This operation includes bytes from “Msg Length” ,”Command” and all the bytes of “Payload”
  - Input: None
  - Output:
    * val: the computed crc value
- takeoff
  - Description: This method initially set the throttle value to 1000 and then continuously send throttle value of 1700 to the drone for a duration of 3.5 seconds. After this, the drone hovers at this altitude for 0.5 seconds.
  - Input: None
  - Output: None
- land
  - Description:This method lands the drone at its position. Firstly, it resets pitch,roll and yaw values. Sets the throttle value for hovering and gradually decreases the throttle.
  - Input: None
  - Output: None
- arm
  - Description: This method arms the drone.
  - Input: None
  - Output: None
- disarm
  - Description: This method disarms the drone.
  - Input: None
  - Output: None
- disconnect
  - Description:This method disconnects the drone.
  - Input: None
  - Output: None
- converttobytes
  - Description: This method converts the given value into two bytes.
  - Input:
    * val – the value that needs to be converted into bytes
  - Output:
    * temp – returns a bytearray of size 2, the first index has the LSB byte and the second index has the MSB byte.
- setcmd
  - Description: This method sets the throttle, roll, pitch and yaw and sends it to the drone
  - Input:

    * : period – time duration for which the packet needs to be sent
  - Output: None
- getattitude
  - Description: This method is used to get the attitude of the drone.
  - Input: None
  - Output: None
- getaltitude
  - Description: This method is used to get the altitude of the drone.
  - Input: None
  - Output: None
- getimu
  - Description: This method is used to get the imu data of the drone.
  - Input: None
  - Output: None

3 Class : *pluto_control*

This class controls the drone to demonstrate the roll, pitch, yaw, takeoff and land capabilities of the drone.

3.1 Methods

- __init__
  - Description: This is the constructor of the class, which establishes a connection with the drone using the class plutodrone. It also arms the drone and sets throttle, roll, pitch and yaw to default values.
  - Input: None
  - Output: None
- sequence
  - Description: This methods performs different actions of the drone in a sequence for a fixed time period. The entire sequence is put inside a try block, so that in case of emergency, the drone can be landed immediately by interrupting using keyboard. The sequence is as follows:
    1. Takeoff – Takeoff the drone
    2. Pitch Front – Pitches the drone front.
    3. Counter Pitch - Pitches the drone backward to counter the velocity obtained during front pitch.
    4. Roll right – Rolls the drone to the right.
    5. Counter Roll – Rolls the drone to the left to counter the velocity obtained during the previous action.
    6. Pitch Backward – Pitches the drone backward.
    7. Counter Pitch - Pitches the drone forward to counter the velocity obtained during the previous action.
    8. Roll left – Rolls the drone to the left.
    9. Counter Roll – Rolls the drone to the right to counter the velocity obtained during the previous action.
    10. Yaw Clockwise – Performs yaw in clockwise direction
    11. Yaw Anticlockwise – Performs yaw in anticlockwise direction.
    12. Land – Lands the drone
    13. Disarm – Disarms the drone
    14. Disconnect – Disconnects the drone
  - Input: None

- Output: None

4 Class: *pluto_keyboard*

This class is used to control the drone using keyboard.

4.1 Methods

- __init__
  - Description: This is the constructor of the class that establishes connection with the drone and sets the pitch, roll, yaw and throttle to default.
  - Input: None
  - Output: None
- key_cmd
  - Description: This method sends commands at a frequency of 50 hertz to the drone based on the keyboard input from the user, allowing the user to fly the drone. The key mapping are as follows:
    1. v – Arm
    2. b – Disarm
    3. w – Increase height of the drone
    4. s – Decrease height
    5. l – Roll right
    6. j – Roll left
    7. i – Pitch front
    8. k – Pitch Back
    9. a – Yaw anticlockwise
    10. d – Yaw clockwise
    11. t- Takeoff
    12. y – Land
  - Input: None
  - Output: None