

01_03_circle_maneuvers_teacher

May 2, 2025

Part of the InnovatED STEM and DroneBlocks Land, Air, and Sea Robotics Curriculum
Licensed for educational use in schools only.
Redistribution, commercial use, or resale is strictly prohibited.
© 2025 InnovatED STEM & DroneBlocks. All rights reserved.

1 Circle Maneuvers with CrazyflySimulator

In this notebook, you'll learn how to fly your simulated or real Crazyflyie in circular paths using `circle_left()` and `circle_right()`.

```
[1]: # Imports and Setup
import time
from crazyfly_sim import CrazyflyieSimulator

# Use simulation mode unless you're connected to a real drone
drone = CrazyflyieSimulator(real=False)
```

1.1 Takeoff

Always take off before running any motion command.

`takeoff(height, velocity)` - height in meters - velocity in meters/second

```
[2]: drone.takeoff(2.0, 0.4)
```

Taking off to 2.0m at 0.4m/s!
Executing command: takeoff 2.0 0.4

1.2 circle_left(radius, velocity, angle_degrees)

Makes the drone fly counterclockwise in a circle or arc.

Parameters: - radius: circle radius in meters (ex: 0.3) - velocity: speed in meters/sec (ex: 0.3)
- angle_degrees: portion of the circle to complete (default 360 for full)

Keep radius < 0.5m indoors for safety and accuracy.

```
[3]: drone.circle_right(0.3, 0.3, 360)
```

Executing circle right (optimized)...

1.3 circle_right(radius, velocity, angle_degrees)

Same as circle_left, but goes clockwise.

Try a half-circle or quarter circle by adjusting angle_degrees.

```
[ ]: drone.circle_left(0.3, 0.3, 180)
```

Executing circle right (optimized)...

1.4 Using time.sleep() After Circle

Circle commands already include a sleep inside their implementation, based on the circle duration. But you can still pause after to let the drone stabilize.

```
[ ]: time.sleep(2)
```

1.5 Exercise 1: Try a small half-circle to the left

Goal: Make the drone fly a 180° arc with a radius of 0.2m and velocity of 0.2 m/s.

```
[ ]: # drone.circle_left( , , )
```

1.6 Exercise 2: Try a full circle to the right

Goal: Make the drone fly a complete 360° turn clockwise.

```
[ ]: # drone.circle_right( , , )
```

1.7 Exercise 3: Try a quarter-circle left, then right

Hint: Use 90° turns and a radius under 0.4m.

```
[ ]: # drone.circle_left( , , )
# drone.circle_right( , , )
```

1.8 Exercise 4: Create a Funnel Flight Pattern

Goal: Use a combination of circle_left and circle_right commands with varying radii to create a funnel-shaped flight pattern.

Hint: Gradually decrease or increase the radius for each circle to form the funnel shape.

```
[3]: drone.takeoff(2.0, 0.4)
time.sleep(2)

# Level 1 - 2.0m
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 2 - 1.8m
drone.down(0.2, 0.2)
```

```

drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 3 - 1.6m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 4 - 1.4m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 5 - 1.2m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 6 - 1.0m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 7 - 0.8m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Level 8 - 0.6m
drone.down(0.2, 0.2)
drone.circle_left(0.3, 0.3, 360)
time.sleep(1)

# Land and close
drone.land(0.3)
drone.close()

```

Already in the air!

Executing circle left (optimized)...

Moving: vx=0, vy=0, vz=-0.2, yaw=0

Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

Moving: vx=0, vy=0, vz=-0.2, yaw=0

Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

```
Moving: vx=0, vy=0, vz=-0.2, yaw=0
Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

Moving: vx=0, vy=0, vz=-0.2, yaw=0
Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

Moving: vx=0, vy=0, vz=-0.2, yaw=0
Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

Moving: vx=0, vy=0, vz=-0.2, yaw=0
Executing command: move 0 0 -0.2 0 1.0

Executing circle left (optimized)...

Landing at 0.3m/s...
Executing command: land 0.3
```

1.9 Land and Close

Always land and close the drone at the end of your session.

```
[ ]: drone.land(0.3)
     drone.close()
```