



블록•코딩 프로그램

제이씨블록

-컵드론 스팀 워크북-



Chapter.1

기본편

자율비행 드론은 무선 조종기없이 SW코딩으로 제어한다. 본 장에서는 제이씨블록을 이용하여 기본적인 자율 비행 방법을 이해하고 실내에서 간단하게 코딩 비행하도록 한다.

제이씨블록의 기능 설명은 “제이씨블록 설명서”를 참고하시기 바랍니다.
본 워크북은 컵드론 스팀 자율 비행을 위한 워크북입니다.

블루투스 연결


2

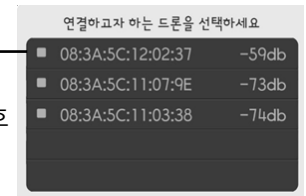
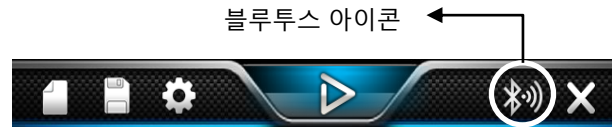
JBC 앱 설치하기

- ① Play Store 또는 App Store에서 "JCBlock"을 검색한다.
- ② 순서에 따라 앱을 설치한다.
- ③ 바탕화면에 JCBlock 아이콘이 생성된다.
- ④ 아이콘을 실행하여 프로그램 정상 동작을 확인한다.



JBC 블루투스 연결하기

- ① 블루투스 장치 전원을 공급하고 켜도록 한다.
- ② 블루투스 아이콘을 클릭하여 연결 시도한다.
- ③ 연결하고자 하는 장치를 선택한다.
- ④ 연결 성공시  아이콘으로 변경된다.



한번 연결된 블루투스 정보는 저장되므로 두번째 연결 부터는 자동으로 검색후 연결된다.

만약, 새로운 블루투스 장치와 연결할 경우에는 기존 장치정보를 삭제 후 다시 연결 시도한다.



JBC 블루투스 해제하기

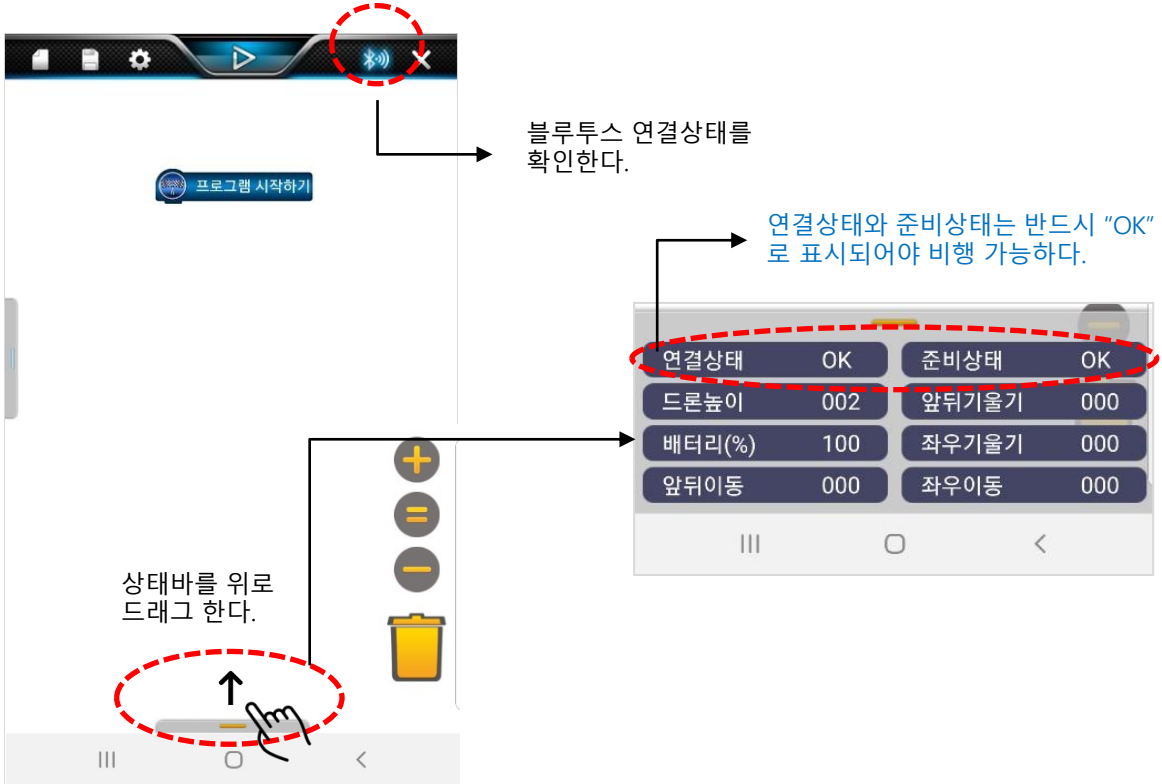
- ① 블루투스 장치 전원을 차단하면 자동으로 연결 해제 된다.

드론 상태 확인

3



컵드론 스팀 자율 비행 준비를 위해 제이씨블록에서 드론 상태정보를 확인하도록 한다.



연결 상태	드론과 제이씨블록의 페어링 여부를 표시한다. OK : 정상 연결 FAIL : 블루투스 연결 안됨
준비상태	드론의 수평, 블루투스 연결정상 준비 여부를 표시한다. 연결상태가 "OK"이고 준비상태가 "0"일 경우에는 "센서보정"한다. OK : 비행준비 완료 FAIL : 비행준비 안됨
드론높이	드론의 높이를 표시한다. 메인보드 기준의 지면에서부터의 드론 높이이다.
배터리(%)	배터리 잔량을 표시한다. 잔량이 20%미만일 경우에는 방전률이 약해 이륙시 흔들림이 나타날 수 있다.
앞뒤기울기 /좌우 기울기	드론이 수평일 때는 앞뒤좌우 "0"으로 표시된다. 만약, 평평한 지면에 두었을 때 "0"으로 표시되지 않을 때 "센서 보정"하도록 한다.
앞뒤이동 /좌우이동	드론의 위치 기준으로 이동 거리만큼 합해져서 값을 표시하난. 단, 드론 앞(전면)기준이다.

준비상태 점검

4

J B C 컵드론 스팀의 자율비행 준비 상태 점검을 LED로 구현한다.



패드에 LED를 배치하여 하기 조건으로 코딩한다.

-준비상태가 참(OK = "1")이면 초록 LED 켜짐

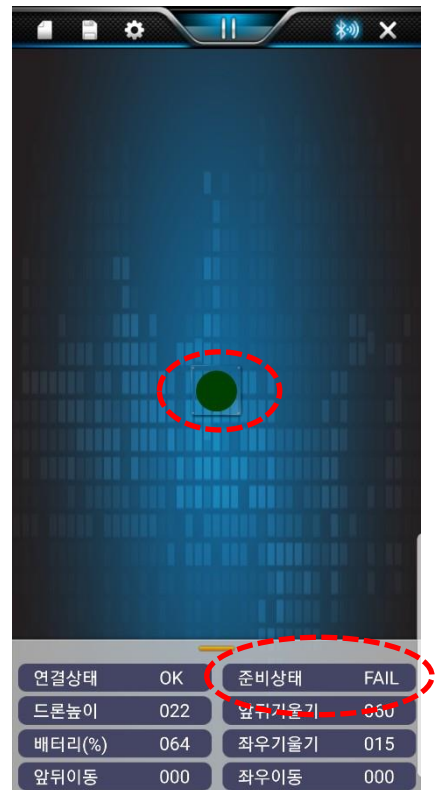
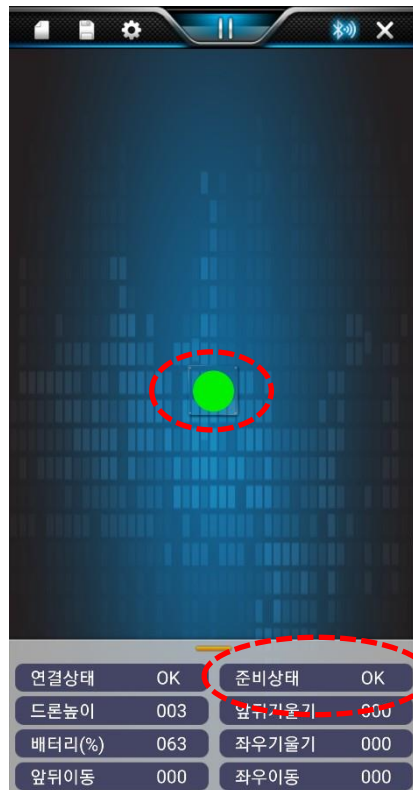
-준비상태가 거짓(FAIL = "0")이면 초록 LED 꺼짐



활용하기

예제 : 1.준비하기

컵드론 스팀을 앞뒤로 기울여서 준비 상태 값과 LED 변화를 확인하도록 한다. 자율비행을 위해서는 반드시 준비상태가 "OK"가 되어야 한다.

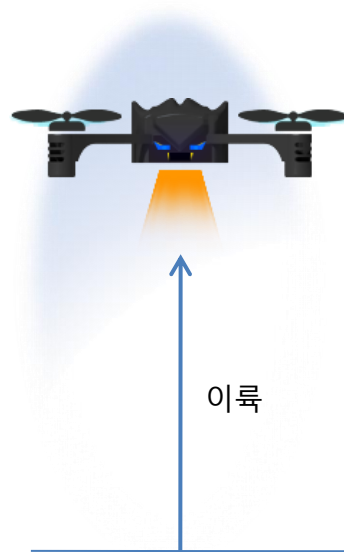


드론 이착륙(1)



JBC 드론 이륙하기

- 주요블록 : 드론 이륙하기
- 용 도 : 제이디코드 착륙하기



코딩실습



프로그램 시작하기

드론 이륙하기

연결상태	OK	준비상태	OK
드론높이	075	앞뒤기울기	000
배터리(%)	089	좌우기울기	001
앞뒤이동	009	좌우이동	004

드론이 이륙하기 시작하면 "드론 높이" 데이터값이 빠르게 변화되는 모습을 확인할 수 있다. 드론은 이륙 후 70cm전후까지의 높이까지 올라간 후 제자리에 멈춘다. 이 때 드론은 외항에 따른 변화에 높이를 고정 하기위해 조금씩 움직이면서 자율적으로 움직임을 조정한다. 변화 값은 상태 메시지로 확인가능하다.

드론 이착륙(2)

6

JBC 드론 착륙하기

- 주요블록 : 드론 착륙하기
- 용 도 : 제이디코드 착륙하기



코딩실습

프로그램 시작하기
드론 이륙하기
드론 착륙하기

착륙하기는 먼저 이륙을 한 후 착륙할 수 있도록 한다. 위의 같이 코딩하면 드론이 비행하지 않고 멈춘 상태로 있을 것이다. 이는 "드론 이륙하기"와 "드론 착륙하기" 블록이 밀리세컨단위로 실행되기 때문에 드론 이륙 명령 코드를 실행 후 드론 착륙하기가 실행되어 비행하지 못하는 것 처럼 보이지만 코딩은 정상적으로 실행 되는 것을 인지하고 코딩하도록 한다.

프로그램 시작하기
드론 이륙하기
10 초 기다리기
드론 착륙하기

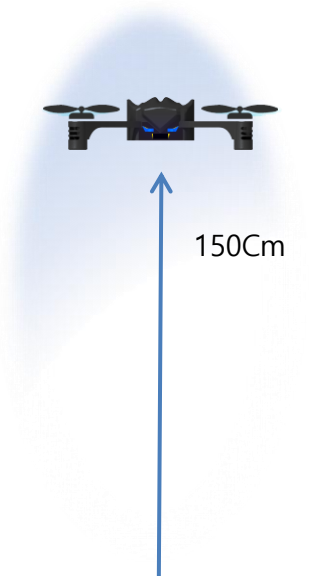
"드론 이륙하기"를 실행 후 드론이 이륙할 수 있도록 잠깐의 시간 동안 기다리게 할 수 있도록한다. 정상적으로 비행 확인 한 후 "드론 착륙하기"를 삽입하여 드론을 착륙하도록 한다.

높이 제어(1)



JBC 높이 제어하기

- 주요블록 : 1 cm 높이로 비행
- 높이제어 구간: 50~150cm
- 용 도 : 드론을 원하는 높이로 지정한다.



코딩실습

예제 : 2, 높이제어



컵드론 스템의 기본 이륙하기 값은 70cm로 정의된다.

사용자가 임의로 고도 제어를 위해 사용되는 블록이다.

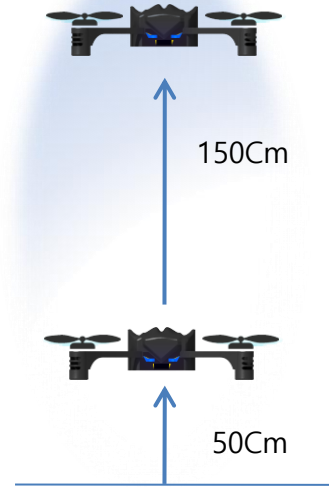
예제와 같이 "드론 비행 준비 상태" 블록을 삽입하여 드론 상태 점검 후 이륙 → 높이지정 → 착륙으로 코딩 하여 실행 해 보자.

높이 제어(2)

8

J B C 고도 반복 제어하기

제어 블록을 이용하여 컵드론 스템의 높이를 반복하여 위아래 움직여 보도록 한다.



코딩실습

예제 : 3.고도반복제어



컵드론 스템은 높이를 50~150cm까지 제어 가능하다. 예제 코드와 같이 최소 최대 값을 넣어 높이의 변화를 코딩을 통해 구현 하고 높이 구간을 확인 해보도록 하자. 50Cm와 150cm를 위아래로 2번 반복하기를 실행 후 착륙한다.

만약, 드론 착륙하기를 코딩을 넣지 않을 경우에는 마지막 명령어를 수행하므로 150cm 높이로 기다리기가 계속 실행되어 착륙하지 않는다. 코딩 할때는 "드론 착륙하기"를 반드시 삽입하여 안정적으로 드론을 착륙 하도록 한다.

속도 제어(1)

9

JBC 직진 후진하기

- 주요블록 : 앞 ▾ (으)로 1 속도(cm/s)로 비행
- 속도제어 구간: 0~200cm/s
- 용 도 : 드론을 원하는 속도로 직진 또는 후진한다.



코딩실습

예제 : 4. 직진후진속도



위의 2개의 코딩의 차이점은 무엇일까? 결론은 "10초 기다리기"이다.

왼쪽의 10초 기다리기를 하지 않을 경우에는 마지막 코딩(앞으로 70속도로 비행) 한 줄을 실행 후 바로 "드론 착륙하기"를 실행 하게 된다. 따라서 "○초 기다리기" 넣어 직진하고자 하는 거리만큼을 속도와 기다리는 시간을 가늠하여 코드값을 입력한다.

$70\text{cm/s} \times 10\text{s(초)} = 700\text{cm}$ 1초당 70cm 이동하므로 10초 기다리기하면 700cm거리 만큼 이동 한다.

속도 제어(2)

10

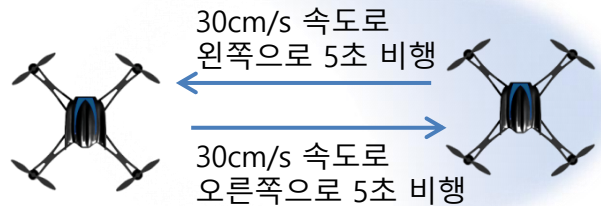
J B C 오른쪽/왼쪽으로 비행하기

- 주요블록 :



- 속도제어 구간: 0~200cm/s

- 용도 : 드론을 원하는 속도로
오른쪽/왼쪽으로 비행한다.



코딩실습

예제 : 5.좌우속도



왼쪽/오른쪽 비행은 비행 회전 없이 꽃게와 같이 옆으로 비행하는 모습이다.

왼쪽 코딩과 같이 하면 왼쪽으로 30cm/s를 5초 동안 비행하고 오른쪽으로 같은 속도로 같은 시간만큼 이동 하므로 제자리 위치로 컴백하게 된다. 하지만 외항과 오차범위를 내에서 착륙한다.

$$30\text{cm/s} \times 5\text{s(초)} = 150\text{cm}$$

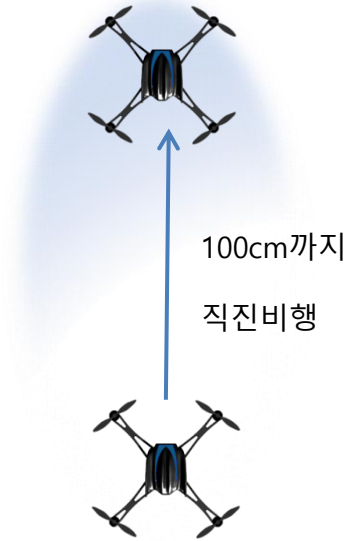
1초당 30cm 이동하므로 5초 기다리기하면 150cm거리만큼 이동 한다.

거리 제어(1)

11

J B C 거리 제어하기

- 주요블록 : 앞 ▾ (으)로 1 cm 거리를 비행
- 거리제어 구간 : 0~1,000cm
- 용 도 : 드론을 지정한 거리만큼 100cm/s속도로 비행한다.



코딩실습

예제 : 6.거리제어



프로그램 시작하기

준비상태 = 1 까지 기다리기

드론 이륙하기

5 초 기다리기

100 cm 높이로 비행

5 초 기다리기

앞 ▾ (으)로 100 cm 거리를 비행

10 초 기다리기

드론 착륙하기

거리 제어는 지정한 거리 만큼 100cm/s 속도로 비행한다. 속도 100cm/s는 내부적으로 정의되어 있으며 사용자가 변경할 수 없다.

예제 코드를 살펴보면 100cm/s 속도로 100cm 이동하므로 1초만에 지정 거리에 도달하고 9초는 제자리 비행한 후 착륙하게 된다.

① $100\text{cm/s} \times 1\text{s(초)} = 100\text{cm}$

1초당 100cm 이동하므로 1초 기다리기하면 100cm거리만큼 이동 한다.

② $100\text{cm/s} \times 3\text{s(초)} = 300\text{cm}$

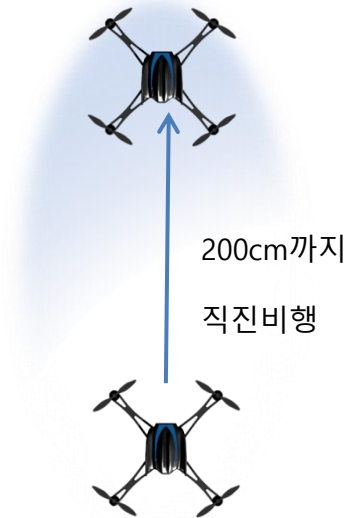
1초당 100cm 이동하므로 3초 기다리기하면 100cm거리만큼 이동 한다. 단, 3초보다 작은 값으로 기다리기 할 경우에 지정 거리만큼 비행하지 못한다.

거리 제어(2)

12

JBC 거리 제어 vs 속도 제어

거리제어와 속도 제어 블록의 미세한 차이점을 비교하고 학습하여 자율 비행 시 보다 스마트하게 코딩 해보도록 한다.



코딩실습

예제 : 7.거리vs속도

```
프로그램 시작하기
준비상태 = 1 까지 기다리기
드론 이륙하기
5 초 기다리기
100 cm 높이로 비행
5 초 기다리기
앞 (으)로 50 속도(cm/s)로 비행
4 초 기다리기
드론 착륙하기
```

```
프로그램 시작하기
준비상태 = 1 까지 기다리기
드론 이륙하기
5 초 기다리기
100 cm 높이로 비행
5 초 기다리기
앞 (으)로 200 cm 거리를 비행
4 초 기다리기
드론 착륙하기
```

위 코드는 거리와 속도를 기준으로 데이터를 계산하면 동일한 결과를 예상 할 수 있다. 하지만 코딩 처리 과정은 서로 다르다.

[속도 기준] 50cm/s 로 4초동안 직진, 즉 $50\text{cm/s} * 4\text{s} = 200\text{cm}$ 까지 비행

[거리 기준] 200cm까지 100cm/s 속도로 비행 즉 $100\text{cm/s} * 2\text{s} = 200\text{cm}$ 까지 비행

거리 기준 코딩은 약200cm 까지 도달 하기 위해 100cm/s 로 비행하지만 가속과 거리와의 관계를 고려해 자동으로 속도를 줄여서 지정 거리에서 멈추게 되고, 속도 기준은 50cm/s 으로 4초동안 비행을 하고 “드론착륙하기” 가 실행되면 서서히 속도를 줄여가기 때문에 200cm 넘어 착륙하게 된다.

회전 제어(1)

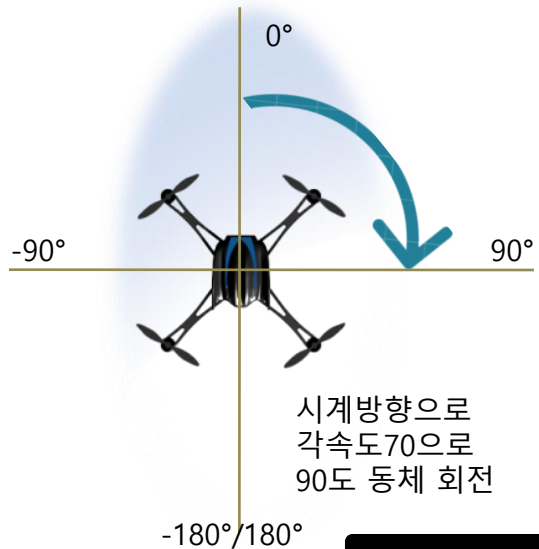
13

JBC 90도 회전 제어

- 주요블록 : 시계방향 ▾ (으)로 1 도 회전

- 회전 각도 제어 구간 : -180~180

- 용도 : 드론 앞면 기준으로 지정한 각도 만큼 회전한다.



코딩실습

예제 : 8.90도회전



회전 방향과 회전 각도를 설정 하면 지정한 값으로 동체가 회전하다.

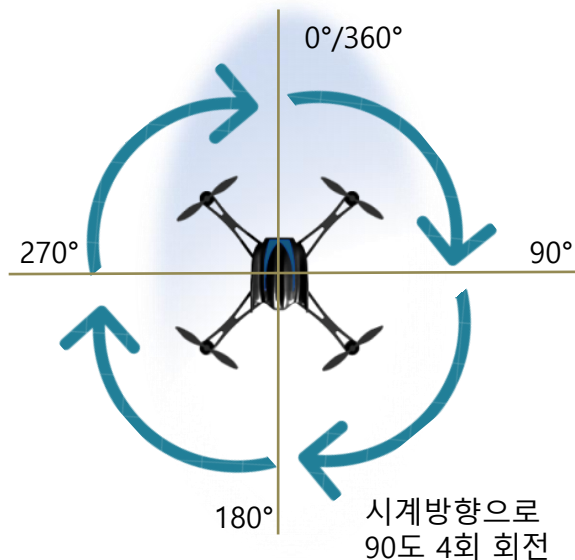
방향은 전면 기준으로 오른쪽으로 0~180도 왼쪽으로 0도~-180도로 지정하도록 한다.

회전 제어(2)

JBC 360도 반복 회전

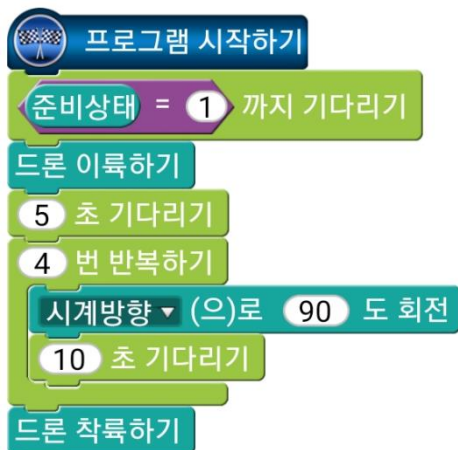
90도씩 4번 반복 회전하여 360도 회전해보도록 한다.

가속력과 반작용에 의해 오차가 발생하며 제자리 회전한다.



코딩실습

예제 : 9.반복회전



위의 두 예제를 비교하여 코딩 시연해 보도록 하자.

결과는 동일하나 과정은 서로 상이하다. 왼쪽 예제는 90도씩 4번 회전하여 360도를 회전하고, 오른쪽 예제는 회전 없이 360도 지점에 고정 호버링 하고 있다. 회전 제어 블록에서 -180~180도로 구간을 지정한 이유도 270도 시계방향 회전은 반시계방향 -90도랑 동일한 결과를 나타내기 때문이다.



Chapter.2

응용편

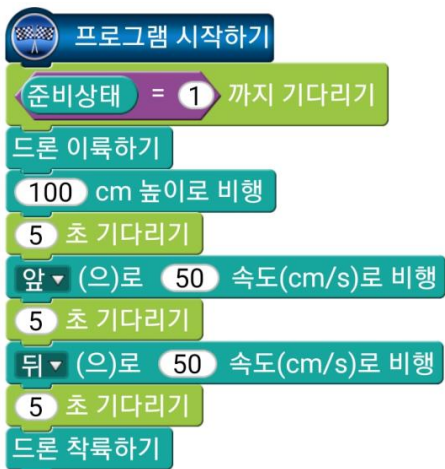
본 장에서는 기본편에서 학습한 코딩을 바탕으로 드론을 보다 다이내믹 제어 할 수 있도록 다양한 비행 방법을 구현 하도록 한다.

제이씨블록의 기능 설명은 “제이씨블록 설명서”를 참고하시기 바랍니다.
본 워크북은 컵드론 스팀 자율 비행을 위한 워크북입니다.

회전 컴백

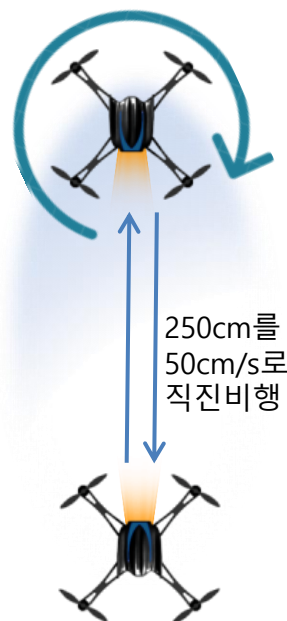
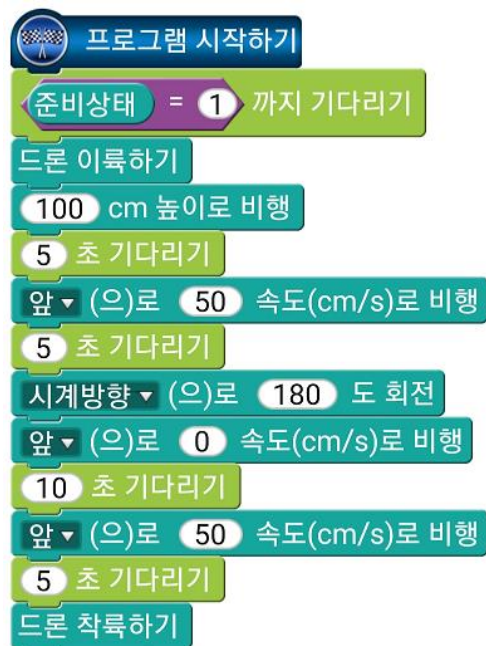
16

J B C 아래의 리턴홈 방법을 비교하고 또다른 다양한 코딩 방법으로 리턴 홈을 구현해 보자.



코딩실습

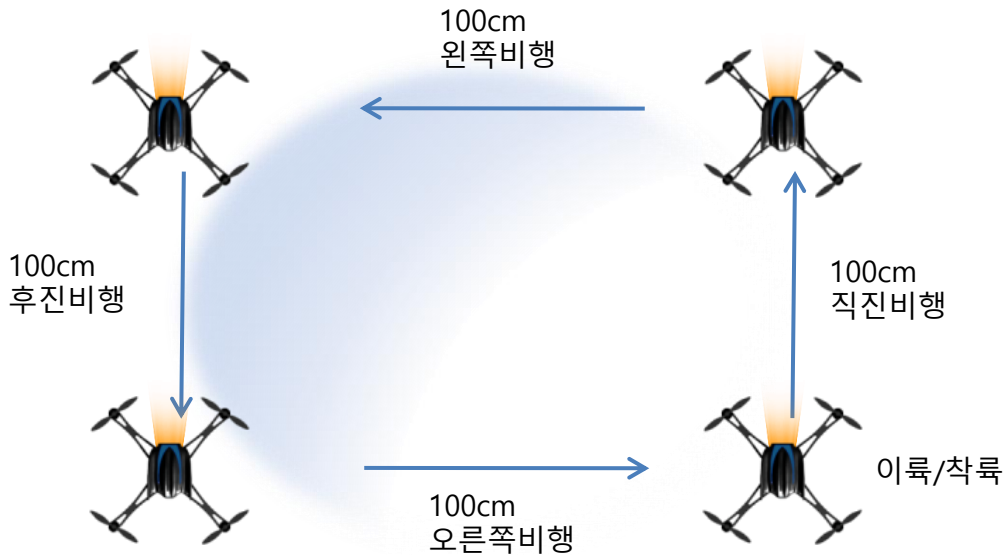
예제 : 회전컴백



스퀘어 비행

17

JBC 속도와 방향을 사용자가 정의하여 스퀘어 비행을 구현한다.



코딩실습

예제 : 11. 스퀘어비행



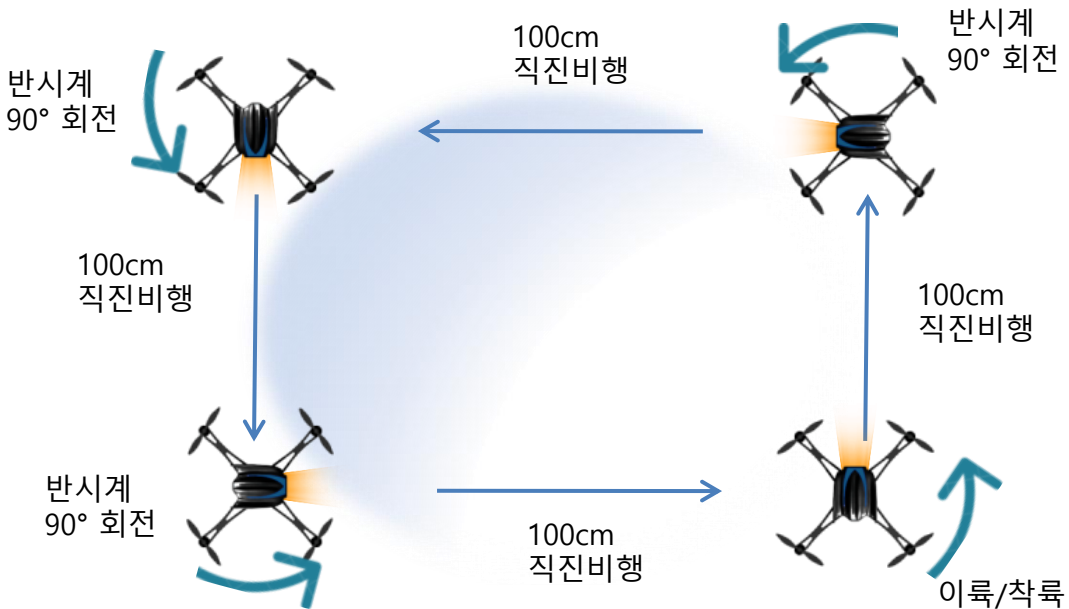
비행 코딩 시 유의 사항이 "○초 기다리기" 블록을 적절히 활용하는 것이다. 비행은 자세 제어와 가속에 의해 오차가 크게 발생할 수 있다.

오차를 줄이기 위한 방법으로 "○초 기다리기"를 삽입하여 드론이 포지셔닝 할 수 있도록 기다려주고 오차를 줄이도록 한다.

스퀘어 회전 비행

18

JBC 속도와 방향을 사용자 정의하여 스퀘어 비행을 구현해 본다.



코딩실습

예제 : 12.스퀘어회전



프로그램 시작하기

준비상태 = 1 까지 기다리기

드론 이륙하기

5 초 기다리기

4 번 반복하기

앞 ▾ (으)로 100 cm 거리를 비행

10 초 기다리기

반시계방향 ▾ (으)로 90 도 회전

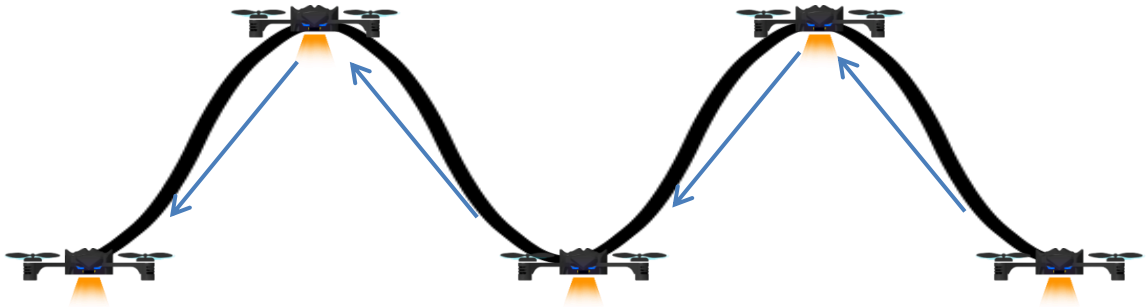
10 초 기다리기

드론 착륙하기

스퀘어 비행에 드론 회전기능과 조건문을 이용하여 구현해 보자.

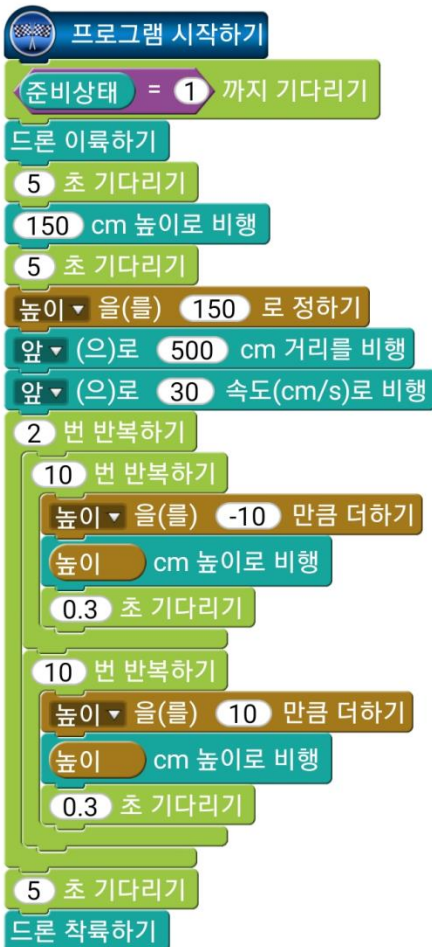
스퀘어 비행 보다 코딩은 간단해 지지만, 회전이란 블록을 사용하게 되므로 드론 비행의 오차가 조금 더 클 수 있다.

J B C 높이를 계단식으로 낮추면서 자연스러운 웨이브 비행하도록 구현한다.



코딩실습

예제 : 13.웨이브비행



웨이브 비행은 계단식으로 높이를 줄이고 늘려서 자연스럽게 고도를 제어해서 웨이브 모습으로 비행하고자 한다.

웨이브 비행 할때의 코딩 핵심사항은 지정한 거리까지 직진하면서 높이를 반복적으로 제어 하면서 전체 증폭을 만들어 가는 방법이다.

높이의 폭을 조정하면서 사용자가 원하는 증폭을 만들어 보도록 한다.



Chapter.3

기술편

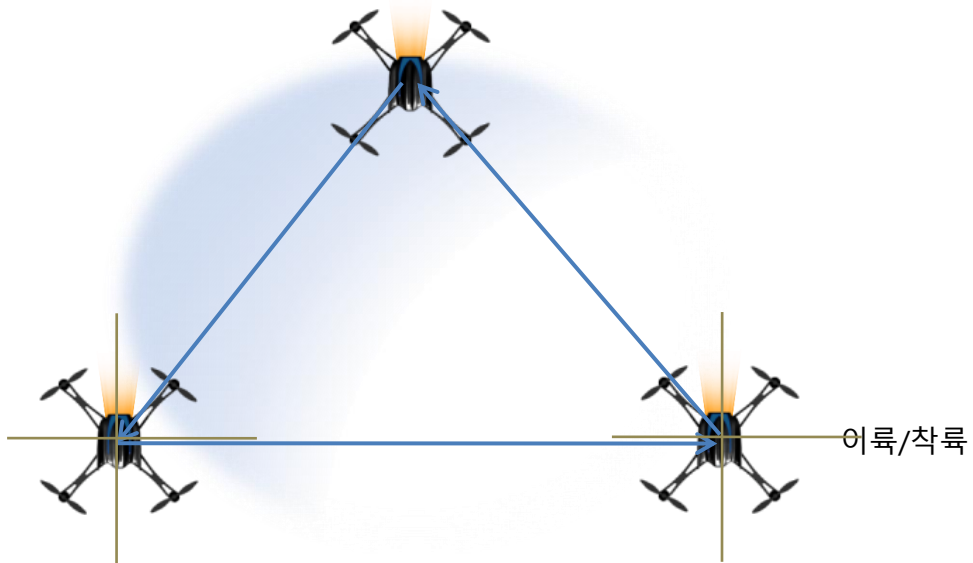
본 장에서는 기본편에서 학습한 코딩을 바탕으로 드론을 보다 다이내믹 제어 할 수 있도록 비행 기술 방법을 구현 하도록 한다.

제이씨블록의 기능 설명은 “제이씨블록 설명서”를 참고하시기 바랍니다.
본 워크북은 컵드론 스팀 자율 비행을 위한 워크북입니다.

트라이앵글 비행

21

JBC 방향과 속도, 거리를 사용자 정의하여 트라이앵글 비행을 구현해 본다.



코딩실습

예제 : 1.트라이앵글(1)



프로그램 시작하기

준비상태 = 1 까지 기다리기

드론 이륙하기

10 초 기다리기

왼쪽 ▾ (으)로 100 cm 거리를 비행

앞 ▾ (으)로 100 cm 거리를 비행

10 초 기다리기

왼쪽 ▾ (으)로 100 cm 거리를 비행

뒤 ▾ (으)로 100 cm 거리를 비행

10 초 기다리기

오른쪽 ▾ (으)로 200 cm 거리를 비행

5 초 기다리기

드론 착륙하기

코딩은 스쿼어 비행처럼 다양한 방법으로 동일한 결과를 표현할 수 있다.

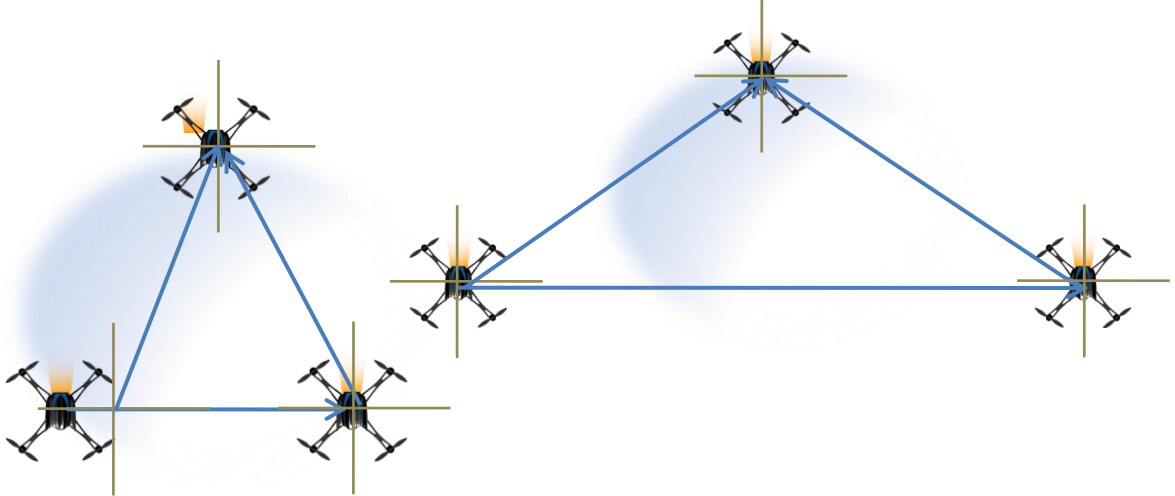
본 트라이앵글 비행은 방향과 속도를 이용하여 드론의 쏠림을 유도하여 원하는 동선으로 비행 하도록 한다.

예제와 같이 동일 거리, 속도로 왼쪽과 앞쪽으로 비행을 동시에 명령하면 힘의 균형에 의해 대각선으로 비행 한다.

트라이앵글 비행(2)

22

J B C 방향과 속도, 거리를 사용자 정의하여 트라이앵글 비행을 구현해 본다.



코딩실습

예제 : 15.트라이앵글(2)

제이디코드에서 트라이앵글 비행을 하고자 할 때 방향과 속도를 이용하여 삼각형 각도의 크기를 조절하여 원하는 모양으로 비행한다.

위의 그림과 같이 각도에 따라 쓸림의 방향이 달라지고 삼각형 모양이 변하게 된다.

위와 같이 코딩하고자 할 경우 트라이앵글(1)을 45도 기준으로 생각하고 가고자 하는 방향의 거리와 속도 비율을 조정해 보도록 한다.

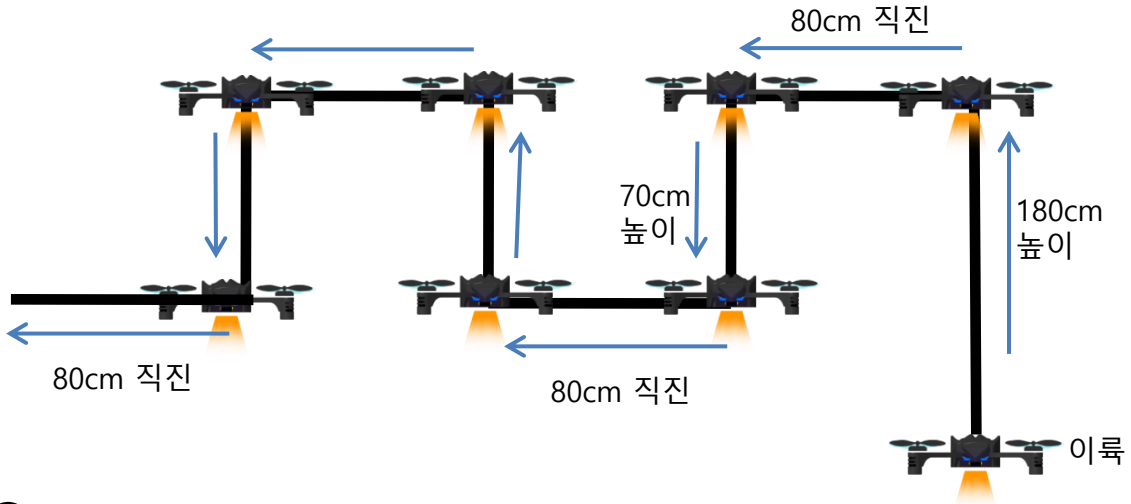
```
프로그램 시작하기
준비상태 = 1 까지 기다리기
드론 이륙하기
10 초 기다리기
왼쪽 ▾ (으)로 50 cm 거리를 비행
앞 ▾ (으)로 100 cm 거리를 비행
10 초 기다리기
왼쪽 ▾ (으)로 50 cm 거리를 비행
뒤 ▾ (으)로 100 cm 거리를 비행
10 초 기다리기
오른쪽 ▾ (으)로 100 cm 거리를 비행
5 초 기다리기
드론 착륙하기
```

```
프로그램 시작하기
준비상태 = 1 까지 기다리기
드론 이륙하기
10 초 기다리기
왼쪽 ▾ (으)로 200 cm 거리를 비행
앞 ▾ (으)로 100 cm 거리를 비행
10 초 기다리기
왼쪽 ▾ (으)로 200 cm 거리를 비행
뒤 ▾ (으)로 100 cm 거리를 비행
10 초 기다리기
오른쪽 ▾ (으)로 400 cm 거리를 비행
5 초 기다리기
드론 착륙하기
```

요철(凹) 비행

23

J B C 고도와 속도를 이용하여 웨이브와 같은 형상으로 비행하도록 한다.



코딩실습

예제 : 16.요철비행



프로그램 시작하기

준비상태 = 1 까지 기다리기

드론 이륙하기

5 초 기다리기

4 번 반복하기

180 cm 높이로 비행

5 초 기다리기

앞 ▾ (으)로 80 cm 거리를 비행

5 초 기다리기

70 cm 높이로 비행

5 초 기다리기

앞 ▾ (으)로 80 cm 거리를 비행

5 초 기다리기

드론 착륙하기

본 코딩은 고도를 급격하게 제어를 하면서 요철을 만들어 본다.

동체가 급격히 낙하 할 때 가속에 의해 지면에 부딪히면 비정상적인 움직임으로 간주하고 자동을 멈추게 된다. 또한 거리와 속도를 계산해서 비행거리만큼 도달 할 수 있도록 "○초 기다리기"를 삽입하도록 주의한다.

다양한 형태로 드론 비행을 해보면서 자율비행코딩의 주의점을 익히도록 한다.



Chapter.3

무선조정기

본 장에서는 제이씨블록을 이용하여 무선조종기를 제작하여 다양한 방법으로 자율비행 방법을 시도하도록 한다.

제이씨블록의 기능 설명은 “제이씨블록 설명서”를 참고하시기 바랍니다.
본 워크북은 컵드론 스팀 자율 비행을 위한 워크북입니다.

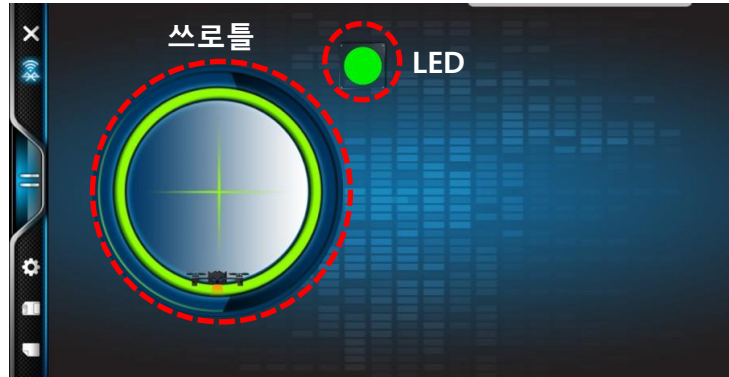
쓰로틀 높이 제어

25

J B C 패드 디자인을 이용하여 무선 조정기를 제작 하여 드론을 원격 제어 하도록 한다.

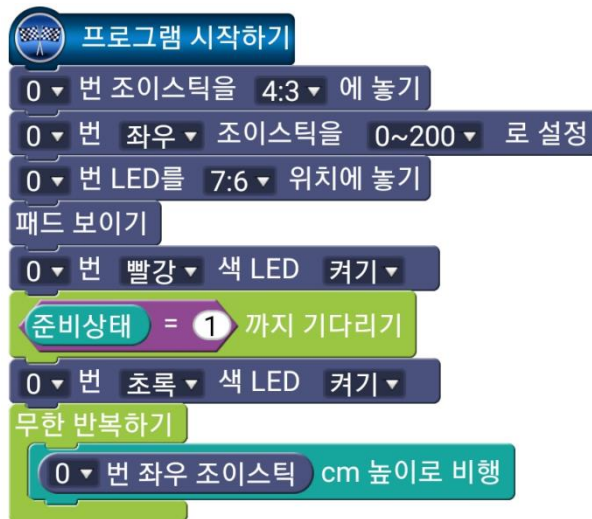
쓰로틀 레버의 값을
드론 높이 값으로 정의하여
쓰로틀을 구현한다.

LED로 연결 상태를 표시한다.



코딩실습

예제 : 17.쓰로틀 높이



1. 조이스틱 패드를 위치에 놓고 값을
0~200 (최저값~최고값)으로 정의 하여
드론의 높이 값으로 이용한다.

2. LED를 패드 위치에 놓는다.
드론의 준비 상태 정보를 LED를 이용
하여 확인하도록 한다.

- 빨강 LED → "준비 상태 =0"
- 초록 LED → "준비 상태 =1"

3. 무한 반복하기 블록 내에 코딩한다.

1 cm 높이로 비행

4. 실행하기 버튼으로 패드 실행한다.



응급 정지 버튼

26

J B C 패드 디자인을 이용하여 무선 조정기를 제작 하여 드론을 원격 제어 하도록 한다.

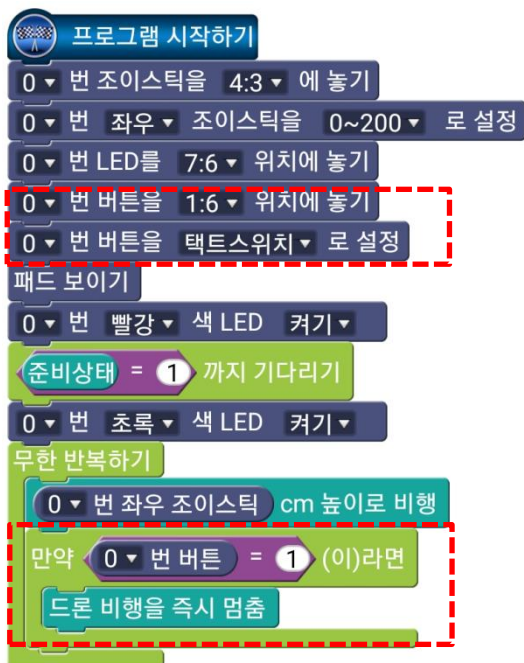
드론의 응급정지 버튼을 구현한다.

응급 상황 시 버튼을 눌렀을 때 드론을 즉시 멈추는 기능이다.



코딩실습

예제 : 18.응급정지버튼



1. 스로틀 높이 제어 예제에 응급 정지 버튼을 추가 코딩하도록 한다.
2. 버튼을 위치에 놓고, 해당 버튼을 "텍스트 스위치"로 설정한다.
3. 만약 조건절을 이용하여 버튼값에 따라 이벤트를 실행한다.
 - 버튼을 눌렀을 때 → "0번버튼 = 1" "드론 비행을 즉시 멈춤" 실행한다.
 - 버튼을 안눌렀을 때 → "0번 버튼 = 0" 실행 이벤트 없다.
4. 실행하기 버튼으로 패드 실행한다.



J B C 패드 디자인을 이용하여 무선 조정기를 제작 하여 드론을 원격 제어 하도록 한다.

조이스틱을 이용하여 드론의 방향(롤/피치)를 제어하는 프로그래밍을 한다.

조이스틱 값의 좌우/상하 값에 따라 드론 방향 제어한다.



코딩실습

예제 : 롤피치방향

1. 응급 정지 버튼 예제에 롤/피칭 기능을 추가 코딩하도록 한다.
2. 오른쪽에 조이스틱을 위치하고 값을 설정한다.
3. "동작"에서 속도블록에 조이스틱 값을 삽입하여 코딩 구현한다.

앞 (으)로 1 속도(cm/s)로 비행

3. 실행하기 버튼으로 패드 실행한다.



요(YAW)

28

J B C 패드 디자인을 이용하여 무선 조정기를 제작 하여 드론을 원격 제어 하도록 한다.

왼쪽 조이스틱을 이용하여 드론의 회전(요, YAW)를 제어하는 프로그래밍을 한다.

요(YAW)변수를 생성하여 변수 값에 비례하여 비행회전이 제어되도록 한다.



코딩실습

예제 : 20.요(YAW)



1. 롤/피치 예제에 요(YAW)기능을 추가 코딩하도록 한다.
2. 0번에 조이스틱 값을 -100~100까지 설정을 추가한다.
3. 조건문 좌회전(-50), 우회전(50)기준으로 요변수 값을 증가 시켜 회전하도록 한다.
 - 좌회전 → 요를 "-1"만큼 더하기
 - 우회전 → 요를 "1"만큼 더하기
4. 실행하기 버튼으로 패드 실행한다.



무선 조종기

29

J B C 변수를 사용하여 코딩을 논리적으로 기능을 표현해보도록 한다.



코딩실습

예제 : 21.무선조종기

*코딩 편집창에서 상태 정보 활성화후 실행하면 패드에서도 상태정보를 확인할 수 있다.

프로그램 시작하기

0번 조이스틱을 4:3에 놓기
0번 좌우 조이스틱을 0~200로 설정
0번 상하 조이스틱을 -100~100로 설정
1번 조이스틱을 4:9에 놓기
1번 상하 조이스틱을 -100~100로 설정
0번 LED를 7:6 위치에 놓기
0번 버튼을 1:6 위치에 놓기
0번 버튼을 텍스트위치로 설정

패드 디자인

패드 보이기

0번 빨강 색 LED 켜기

준비상태 = 1까지 기다리기

0번 초록 색 LED 켜기

드론 연결 상태

요 날(음) 0로 정하기

무한 반복하기

쓰로틀을 음(음) 0번 좌우 조이스틱로 정하기

롤을 음(음) 1번 상하 조이스틱 * 2로 정하기

피치를 음(음) 1번 좌우 조이스틱 * 2로 정하기

변수 정의



쓰로틀 cm 높이로 비행

앞 (음)로 피치 속도(cm/s)로 비행

왼쪽 (음)로 롤 속도(cm/s)로 비행

만약 0번 상하 조이스틱 < -50 (이)라면

요 음(음) -1 만큼 더하기

만약 0번 상하 조이스틱 > 50 (이)라면

요 음(음) 1 만큼 더하기

시계방향 (음)로 요 도 회전

만약 0번 버튼 = 1 (이)라면

드론 비행을 즉시 멈춤

쓰로틀/롤/피치/요 기능

응급정지기능





Chapter. 4

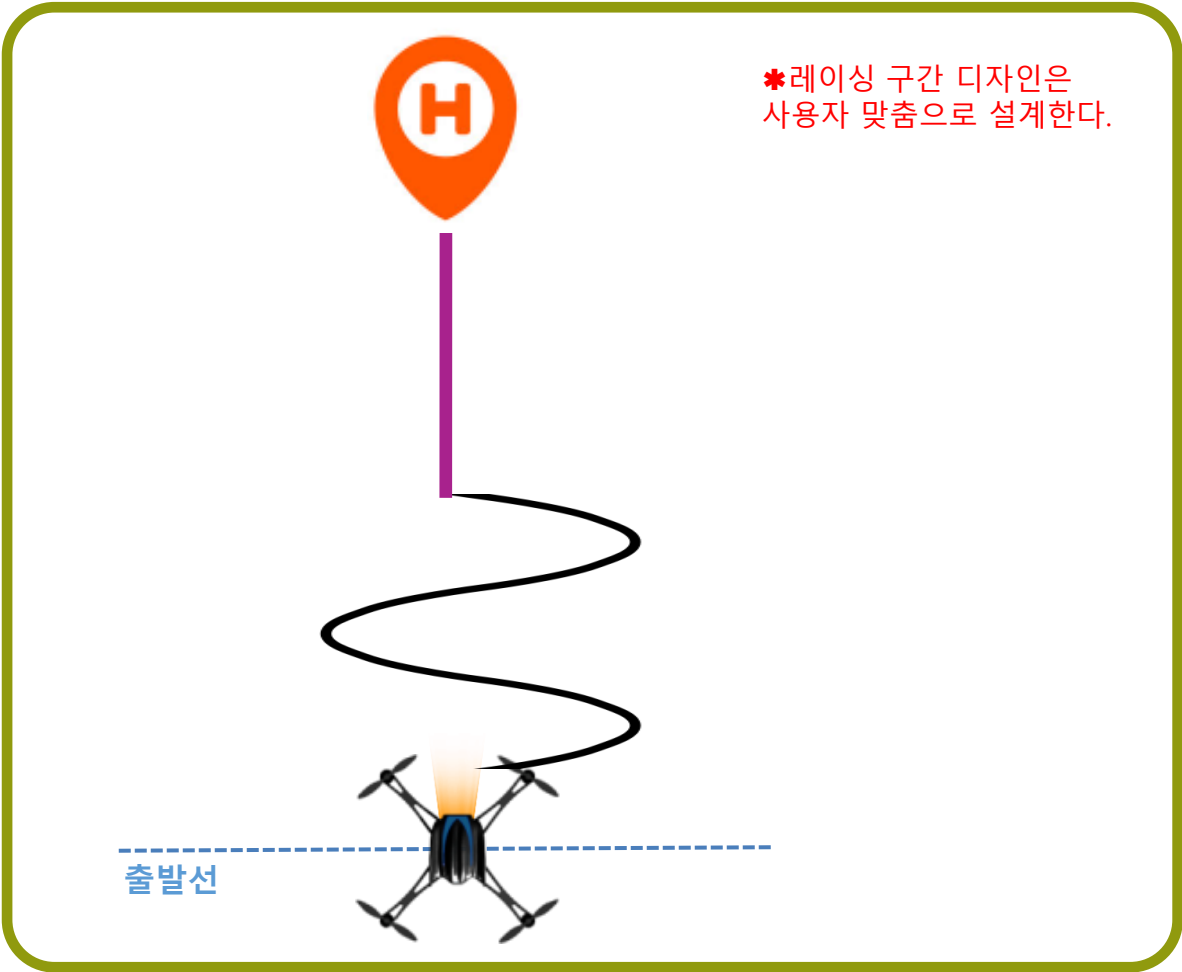
자율비행 레이싱

제이디코드 자율비행기본과 응용편을 통해서 코딩을 통한 비행 방법을 익혔다. 학습한 코딩 방법을 기반으로 자율 비행 미니레이싱을 펼쳐본다.

제이씨블록의 기능 설명은 “제이씨블록 설명서”를 참고하시기 바랍니다.
본 워크북은 컵드론 스팀 자율 비행을 위한 워크북입니다.

목표점 드론 착지

JBC 웨이브 비행으로 목적지에 착지한다.
웨이브 횟수와 도착 시간 기준으로 점수 산정하여 순위를 결정한다.

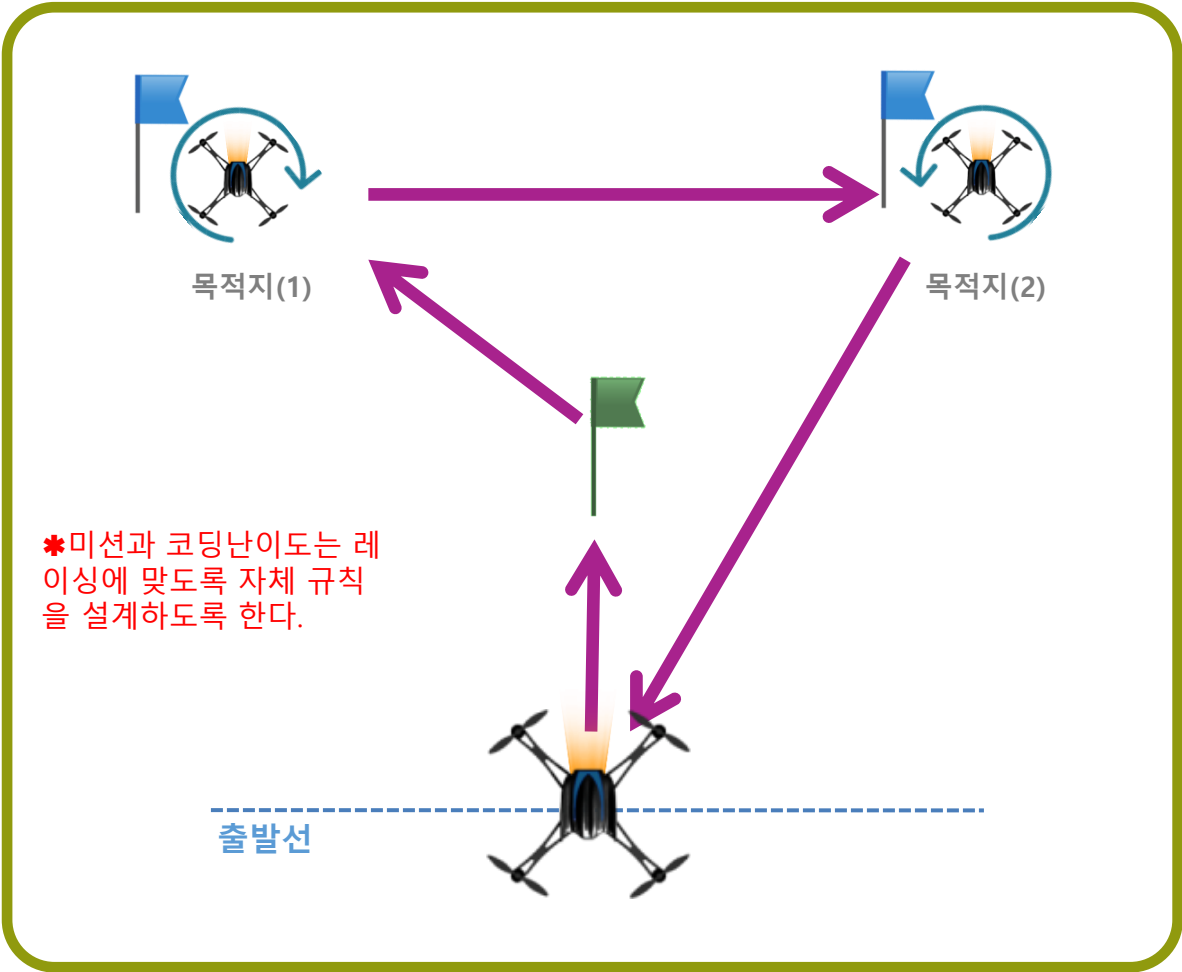


선수명						
비행시간(초)						

선수명						
비행시간(초)						

회전 미션 수행

JBC 드론 회전을 이용하여 레이싱 할 수 있는 구간을 설정한다.
도착 시간 기준으로 점수 산정하여 순위를 결정한다.



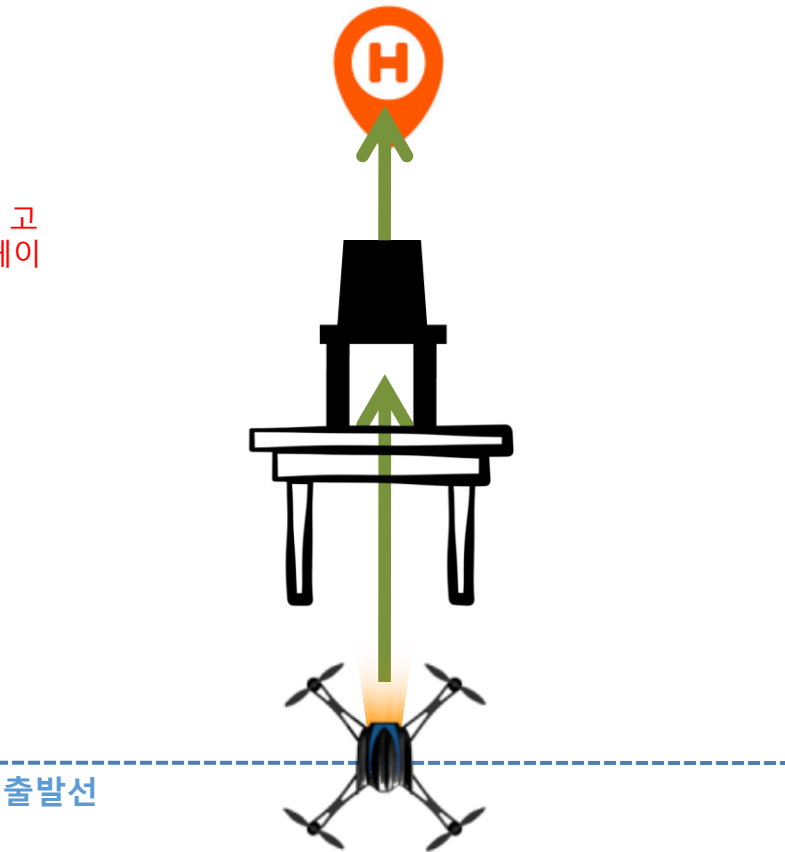
선수명						
비행시간(초)						

선수명						
비행시간(초)						

장애물 통과

JBC 장애물을 통과하여 목적지에 착지한다.
도착 시간 기준으로 점수 산정하여 순위를 결정한다.

*장애물 통과를 위한 고도 제어를 사용하여 레이싱한다.

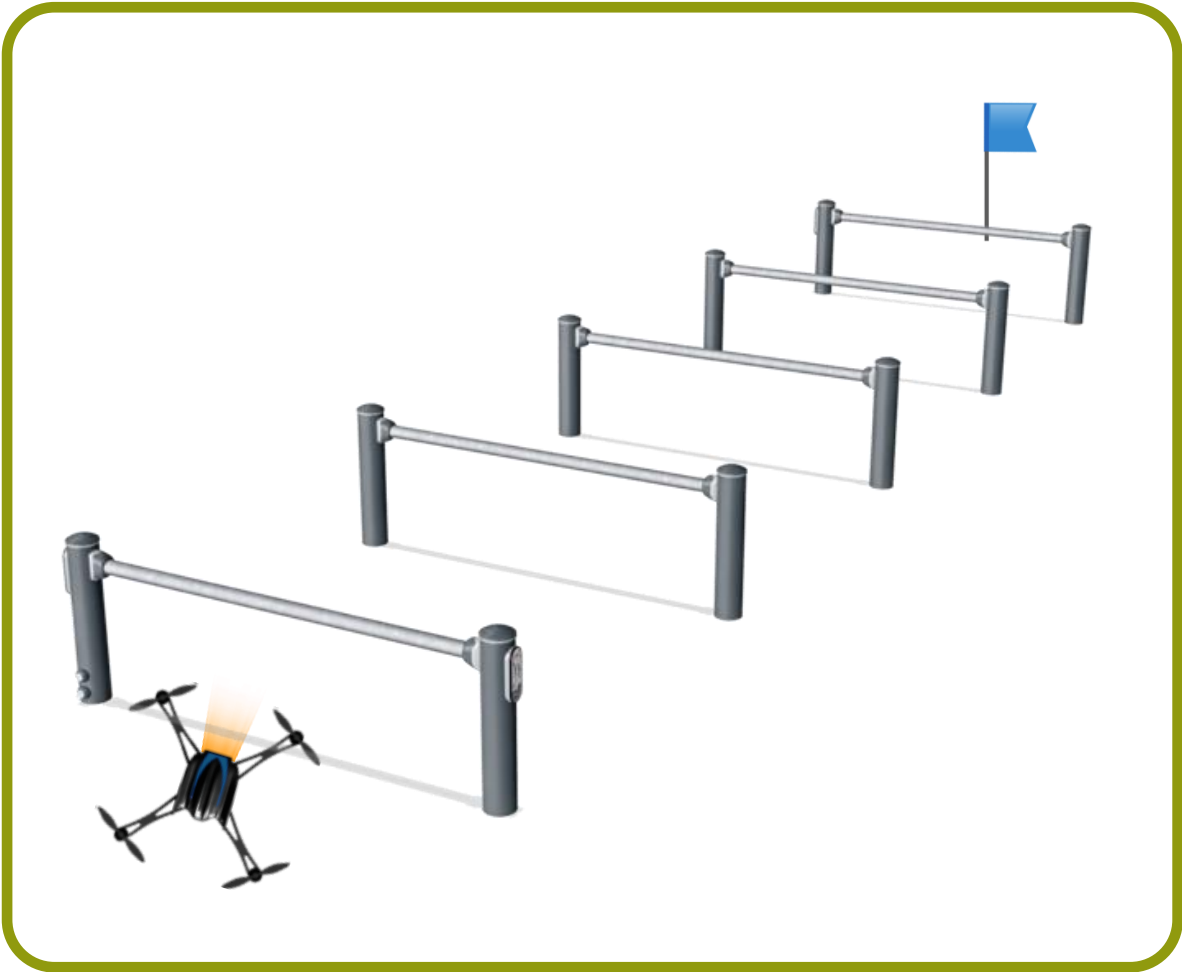


선수명						
비행시간(초)						

선수명						
비행시간(초)						

허들 비행

JBC 가장 빠른시간에 4개의 허들을 넘고 가장 빨리 목적지에 도달하는 순서대로 순위를 결정한다.

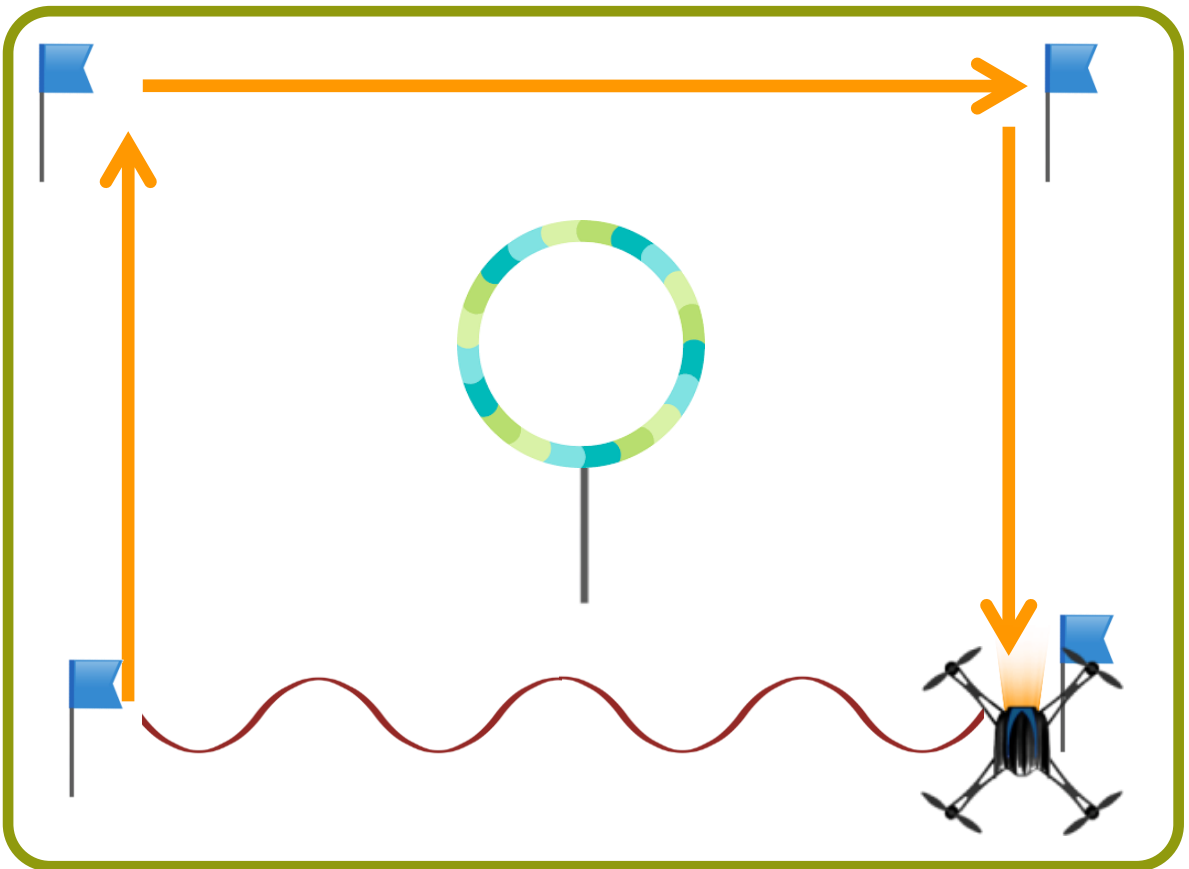


선수명						
비행시간(초)						

선수명						
비행시간(초)						

JBC 사각형 경기장을 만들어 앞서 배운 비행 기술을 모두 적용하여 가장 빠른 시간에 미션을 수행하고 돌아오는 순서로 순위를 결정한다.

1. 꼭지점 깃발 4개 지점에서 5초 동안 제자리 비행하기
2. 후프 1회 통과, 웨이브 비행 1회, 제자리 회전 비행 2회 할 것.
3. 깃발간 이동할 때 고도 제어 2회 할 것.



선수명						
비행시간(초)						

선수명						
비행시간(초)						



스팀메이커

주소: 경기도 성남시 수정구 대왕판교로815, 355호 고객센터 : 070-8225-5897
홈페이지 : www.steammaker.co.kr 이메일 : ceo@steammaker.co.kr