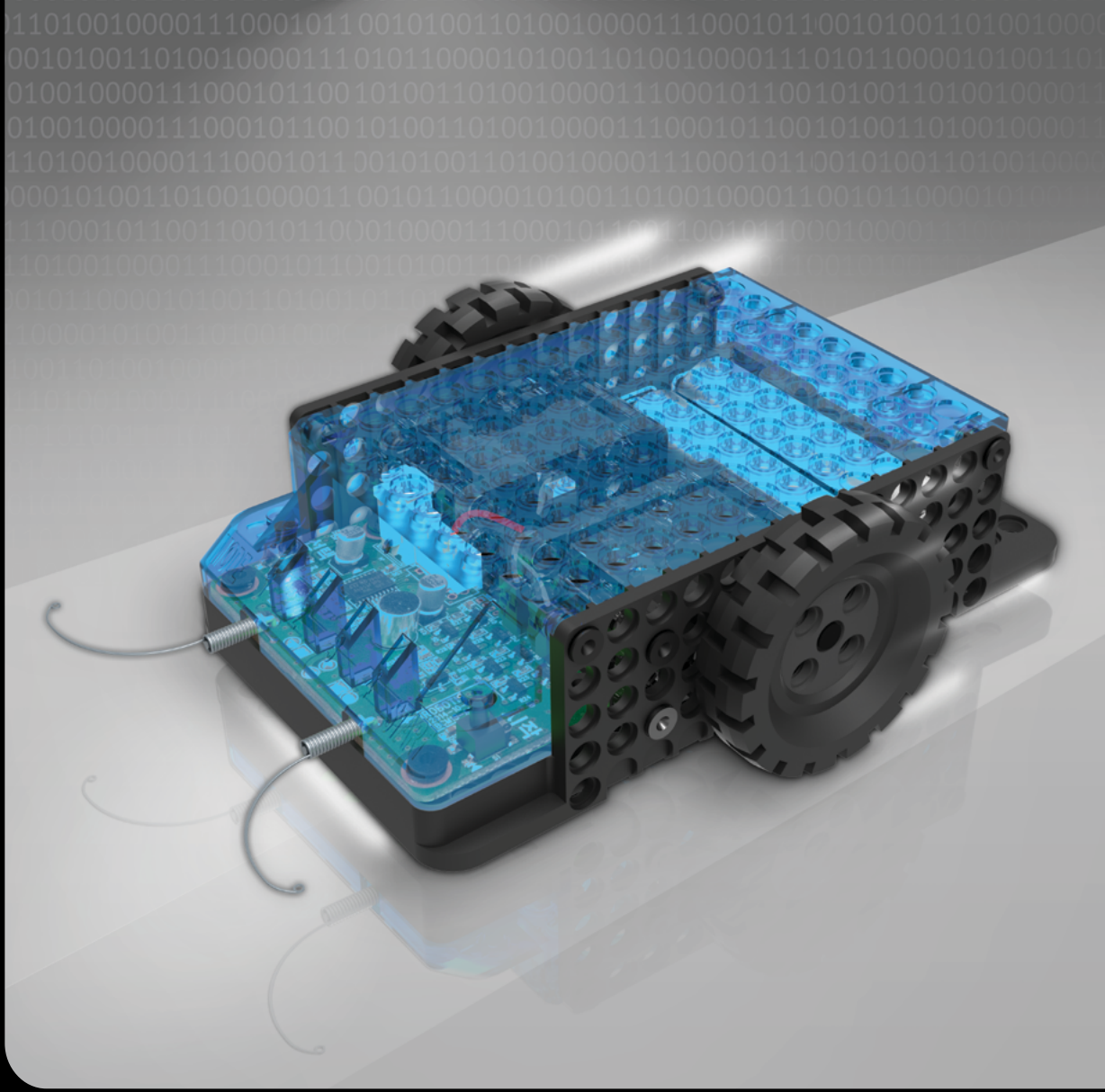




ROBOTIS 로보티즈 자동차로봇



ROBOTIS 로보티즈 자동차로봇



[엔트리편]

수업준비 3

프로그램 설치하기	4
자동차로봇과 컴퓨터 연결하기	5
엔트리 파일 열기	5
자동차로봇을 제어하는 블록들	8
자동차로봇의 장치들	9

1부 운전면허시험 12

1. 전진	13
2. 곡선회전	18
3. 굴절코스	22
4. 변속코스	26
5. 돌발사고 시 급정지 및 출발	29
6. 철길건널목 통과	32
7. 횡단보도 일시정지	35
8. 후진주차코스	38
9. 평행주차코스	42
10. 기타(키보드 조종)	46

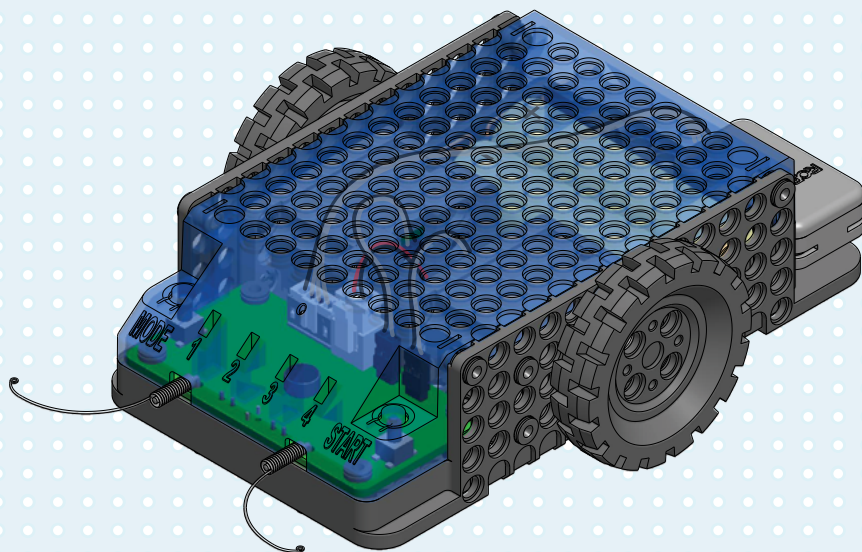
2부 기본 동작 49

11. 소리센서 모드	51
12. 접촉센서 모드	54
13. 적외선센서 모드(라인트레이서)	57
14. 같은 입력으로 두 가지 동작 명령하기	60
15. 같은 입력으로 여러 동작 명령하기	63

예시답안 67

ROBOTIS 로보티즈
자동차로봇 [엔트리판]

수업준비



프로그램 설치하기

자동차로봇을 엔트리로 제어하려면 컴퓨터에 하드웨어 연결 프로그램을 설치해야 합니다. 하드웨어 연결 프로그램을 이용하여 자동차로봇과 컴퓨터와 연결이 되면 엔트리 온라인 사이트 또는 엔트리 오프라인 에디터를 이용하여 프로그램을 실행할 수 있습니다.

(1) 하드웨어 연결 프로그램 설치하기

엔트리에 자동차로봇을 연결하여 제어하려면 하드웨어 연결 프로그램이 필요합니다.

- ① 엔트리 홈페이지에 접속합니다. (<http://play-entry.com>)
- ② [블록꾸러미]에서  - **연결 프로그램 다운로드** 를 클릭하여 하드웨어 연결 프로그램을 다운로드 하여 설치를 완료합니다.

(2) entry 오프라인 설치하기

엔트리 프로그램은 온라인에서 바로 프로그램을 실행 할 수 있지만 인터넷이 연결 안될 경우를 대비하여 오프라인 버전을 설치합니다.

- ① 엔트리 홈페이지에 접속합니다. (<http://play-entry.com>)
- ② 위쪽의  를 클릭하고 **다운로드** 를 클릭합니다.



- ③ 화면 중간의 운영체제에 맞게



- 클릭하여 엔트리 프로그램을 다운받아 설치를 완료합니다.

자동차로봇과 컴퓨터 연결하기

(1) 컴퓨터에 블루투스장치 등록하기

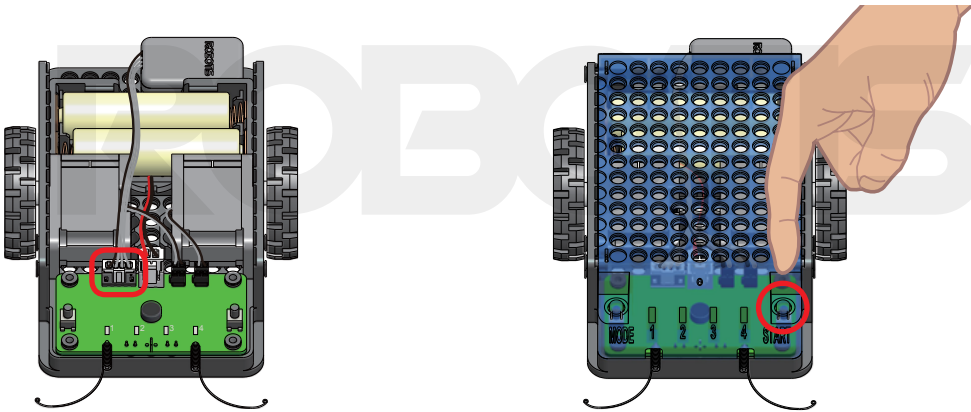
블루투스장치 기종을 확인하고, E-MANUAL에서 등록 방법을 검색합니다.



(2) 엔트리 하드웨어로 자동차로봇과 연결하기

① 제어기에 블루투스 장치를 꽂습니다.
(이미 꽂혀 있다면 다음으로 넘어갑니다.)

② START 버튼을 눌러 전원을 켭니다.



③ 엔트리 하드웨어를 실행시킵니다.



④ 로보티즈 자동차로봇을 선택합니다.



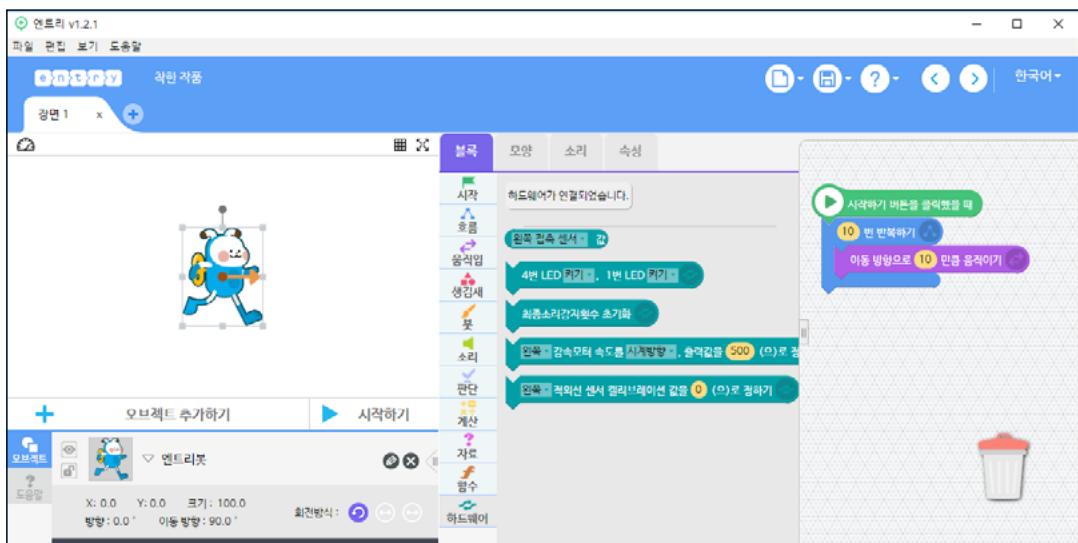
⑤ 블루투스가 연결된 COM PORT를 선택합니다.



- ⑥ 로봇 자동차가 하드웨어와 연결되면 “연결 성공”이라 표시됩니다. 하드웨어가 연결된 상태로 프로그램을 실행해야 하기에 현재 연결 창을 끄면 안됩니다.

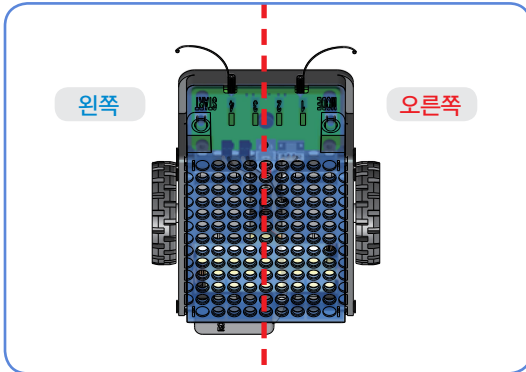


- ⑦ 엔트리 프로그램을 실행하고 블록카테고리에서 하드웨어 - 하드웨어에 연결하기 를 클릭하여 엔트리 프로그램에서 사용할 수 있는 블록들을 확인할 수 있습니다.



자동차로봇을 제어하는 블록들

다음 블록들은 자동차로봇의 입출력장치를 제어합니다. 왼쪽, 오른쪽은 다음과 같이 로봇을 기준으로 정합니다.



왼쪽(4번)이나 오른쪽(1번) LED를 끄거나 켭니다.

4번 LED ☐ 켜기 , 1번 LED ☐ 켜기

✓ 켜기
끄기

✓ 켜기
끄기

최종소리감지횟수를 설정합니다.

최종소리감지횟수 초기화

적외선센서의 캘리브레이션 값을 설정합니다.

※캘리브레이션하는 방법 : <https://www.youtube.com/watch?v=tmtqYxU4IqE>

왼쪽 ☐ 적외선 센서 캘리브레이션 값을 0 (으)로 정하기

✓ 왼쪽
오른쪽

감속모터의 회전 방향과 출력값을 설정합니다.

왼쪽 ☐ 감속모터 속도를 시계방향 ☐ , 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향

제어기에 내장된 센서들의
입력값을 변수 값으로
사용할 수 있습니다.

왼쪽 접촉 센서 ☐ 값

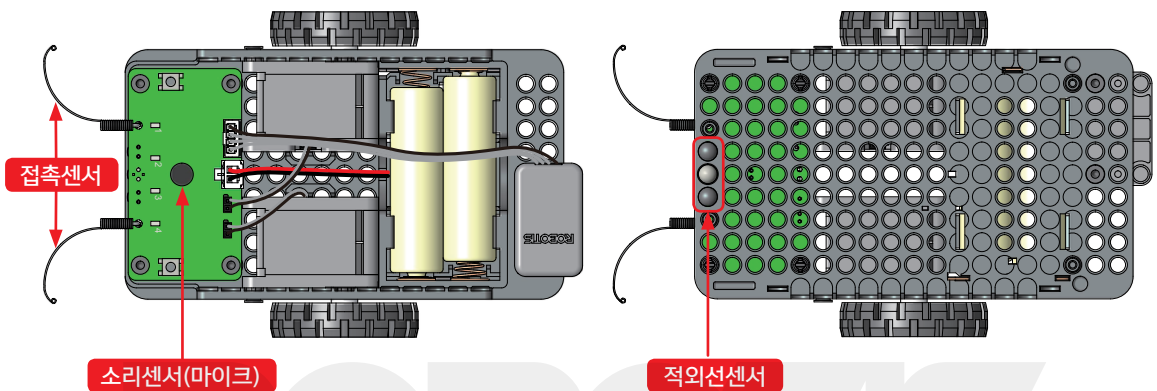
- ✓ 왼쪽 접촉 센서
- 오른쪽 접촉 센서
- 선택 버튼 상태
- 최종 소리 감지 횟수
- 실시간 소리 감지 횟수
- 왼쪽 적외선 센서
- 오른쪽 적외선 센서
- 왼쪽 적외선 센서 캘리브레이션 값
- 오른쪽 적외선 센서 캘리브레이션 값

자동차로봇의 장치들

로봇은 하드웨어와 소프트웨어로 구성됩니다. 하드웨어란 우리가 보고 만지는 장치들입니다. 로봇의 하드웨어는 크게 센서, 제어장치, 구동장치, 전원장치로 구분할 수 있습니다. 자동차로봇이 어떻게 구성되어 있는지 살펴봅시다.

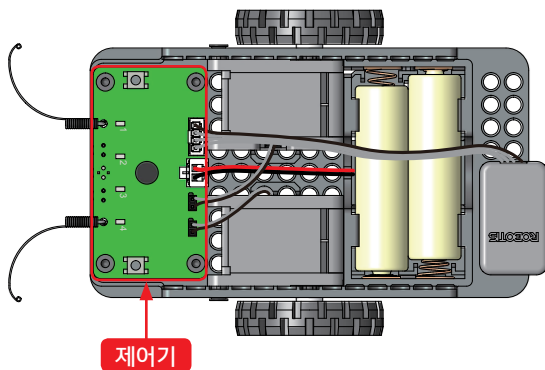
(1) 센서

접촉센서, 소리센서, 적외선센서가 있습니다.



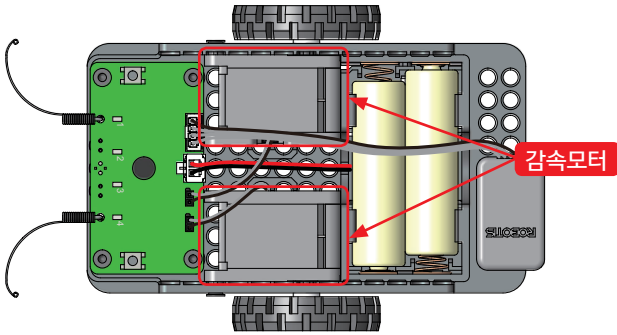
(2) 제어장치

각 장치들을 제어하는 부분입니다.



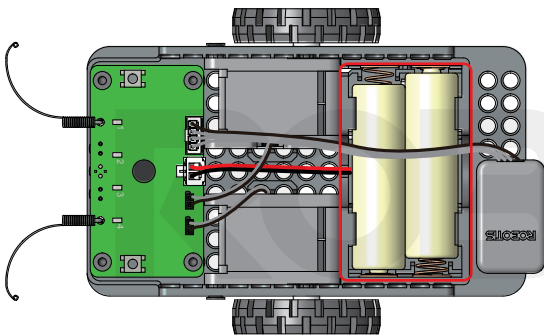
(3) 구동장치

로봇에서 실제로 움직이는 부분으로, 자동차로봇에는 2개 감속모터가 있습니다.



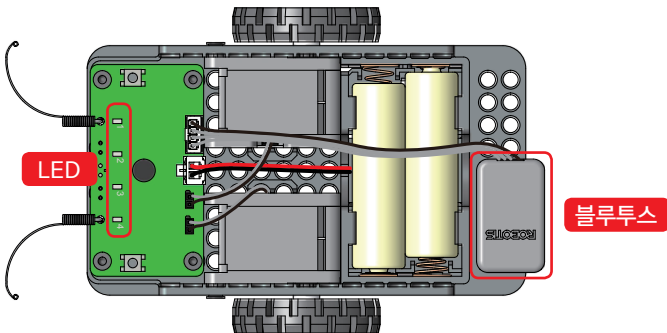
(4) 전원장치

로봇에 전기를 공급하는 부분으로, AA건전지 2개를 사용합니다.



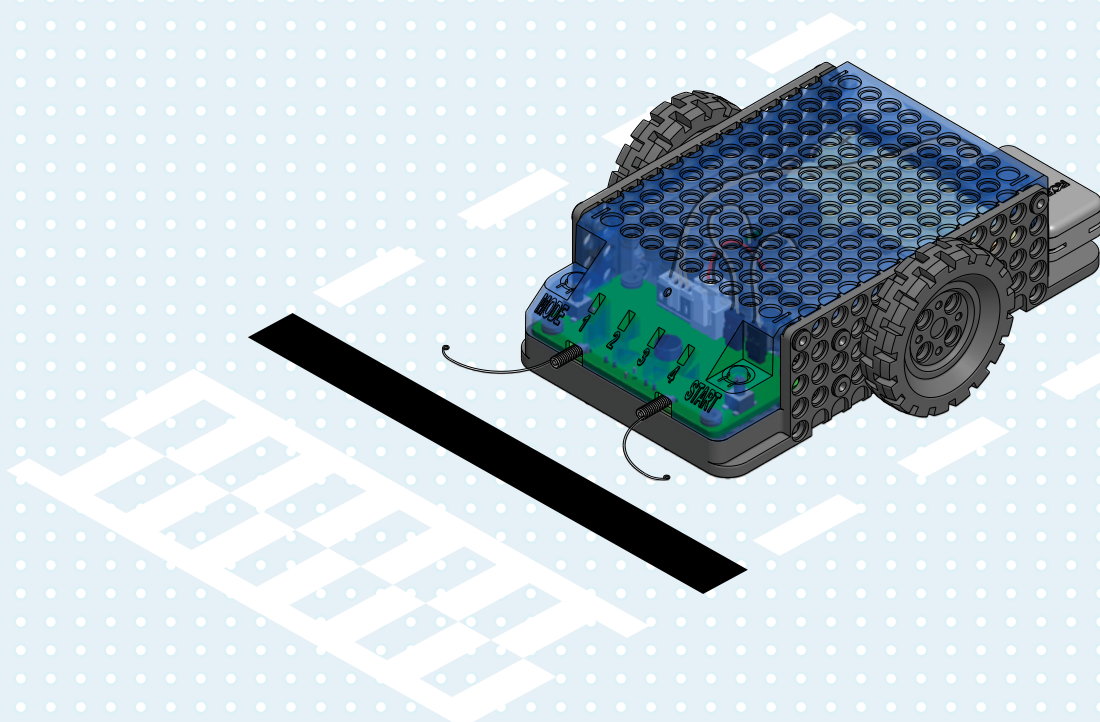
(5) 기타

그 밖에 4개의 LED와 블루투스 등이 있습니다.



ROBOTIS 로보티즈
자동차로봇 [엔트리판]

1부
운전면허시험

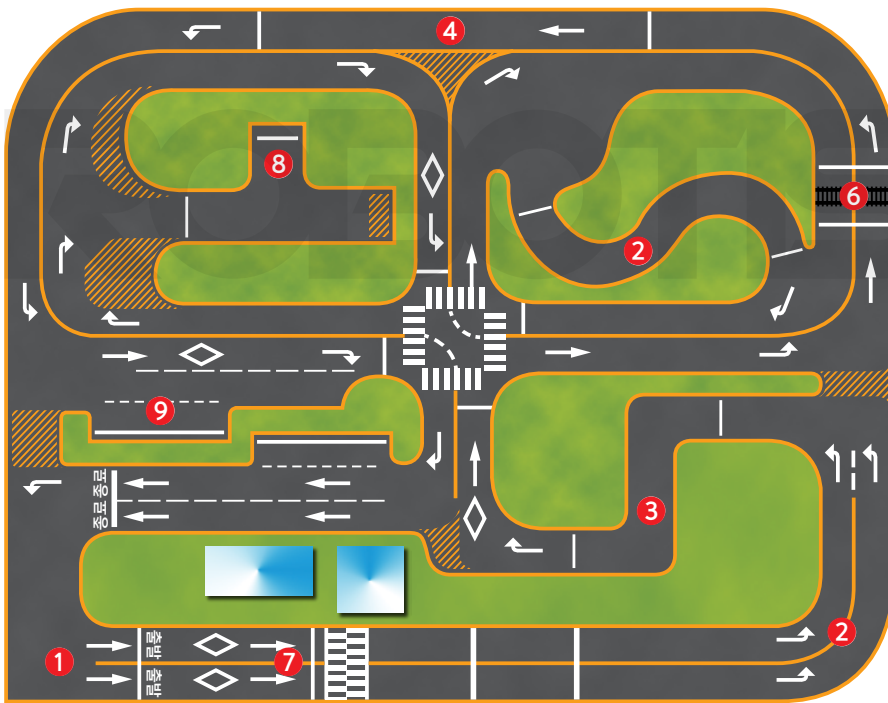


1부. 운전면허시험

사람이 생각에 의해 행동하는 것처럼 로봇은 프로그램에 의해 동작합니다. 지금부터 우리는 엔트리로 프로그램을 작성하여 자동차로봇에게 생명을 불어넣을 것입니다.

자동차로봇으로 운전면허시험의 코스들을 마스터해 봅시다. 이때 우리의 목적은 '코딩'이 아니라 '생각'입니다. 자동차로봇을 어떻게 움직여야 할지 생각하고, 그것을 프로그램으로 어떻게 표현할지 생각하는 것, 프로그램을 완성할 때까지 오류를 해결할 방법을 생각하는 것이 우리의 목적이자 사고력, 창의력을 키울 수 있는 방법입니다.

다음은 실제 대형면허시험 코스입니다. 어떤 코스를 마스터하게 될지 살펴 봅시다.



항목	코스명	항목	코스명
1	전진	6	철길건널목 통과
2	곡선코스	7	횡단보도 일시정지
3	굴절코스	8	후진주차코스
4	변속코스	9	평행주차코스
5	돌발사고 시 급정지 및 출발	10	기타(키보드 조종)

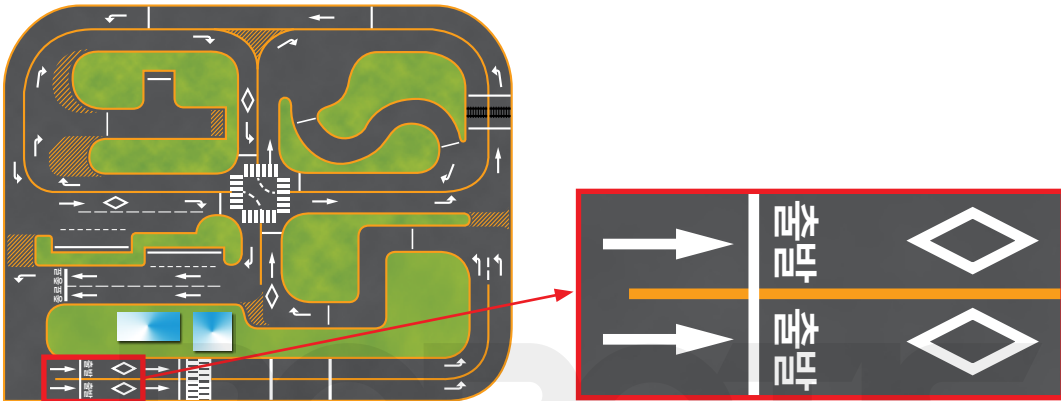
1. 전진

< 학습목표 >

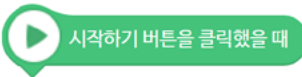
- 자동차로봇이 전진하거나, 후진하게 할 수 있다.
- 프로그램 시작을 명령할 수 있다.

(1) 전진

우선 출발을 해야겠지요? 자동차로봇이 앞으로 나아가게 해 봅시다.



사용할 블록



▶ **시작하기** 를 클릭하면 명령을 실행합니다.

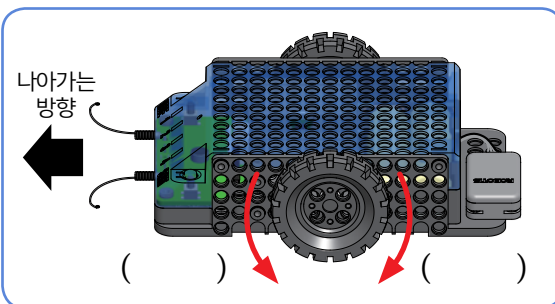


감속모터의 회전 방향과 출력(회전속력)을 설정합니다.
출력은 0~1023 사이의 값으로 클수록 바퀴가 빠르게 회전합니다.

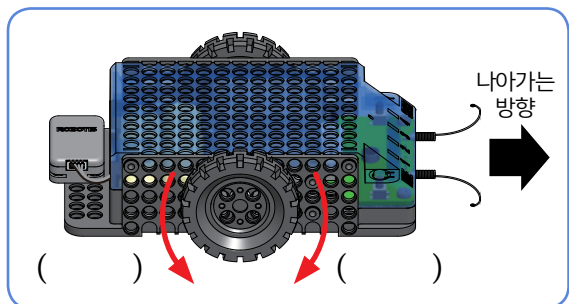
생각하기

자동차로봇이 앞으로 나아가게 해 봅시다. 코딩에 앞서 자동차로봇이 앞으로 나아가는 모습을 상상해 봅시다. 양쪽 바퀴가 각각 어느 방향으로 회전해야 할까요? 생각한 뒤, 아래 그림에 ○표해 봅시다.

왼쪽 바퀴

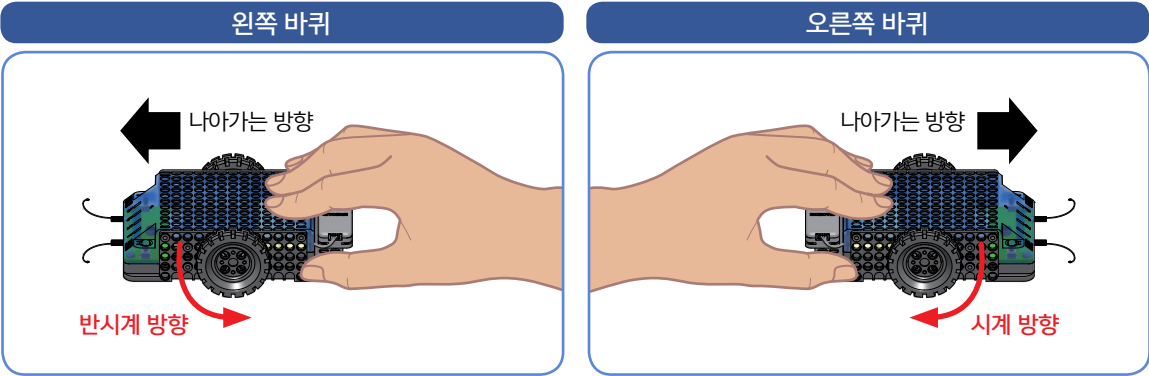


오른쪽 바퀴

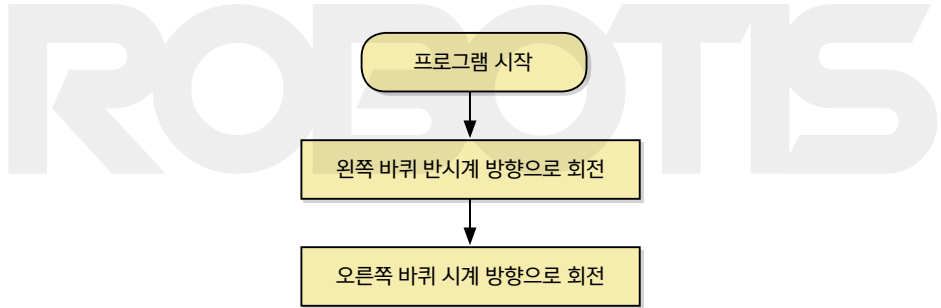


💡 생각 확인하기

자동차로봇을 앞으로 밀며 바퀴가 어느 방향으로 회전하는지 살펴 봅시다. 여러분이 생각했던 대로 바퀴가 회전하였나요? 오른쪽 바퀴는 시계바늘이 도는 방향, 왼쪽 바퀴는 그 반대 방향으로 회전하는 것을 확인할 수 있습니다. 사람들은 시계바늘이 도는 방향을 ‘시계방향’이라고 부르고, 그 반대 방향을 ‘반시계방향’이라고 부릅니다.



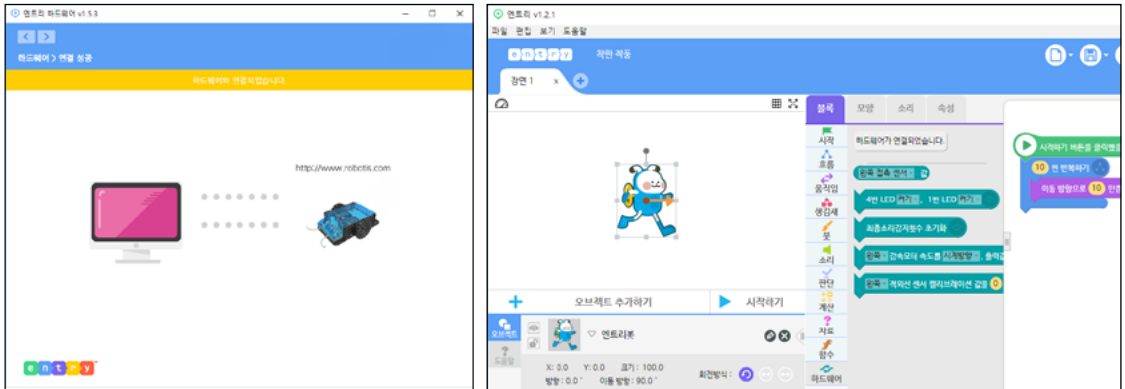
자동차로봇이 앞으로 나아가는 동작을 알아보기 쉽게 정리하면 다음과 같습니다. 이처럼 동작 규칙을 기호와 글로 나타낸 것을 ‘순서도’라고 합니다. 다음부터는 순서도를 직접 채워 넣어 봅시다.



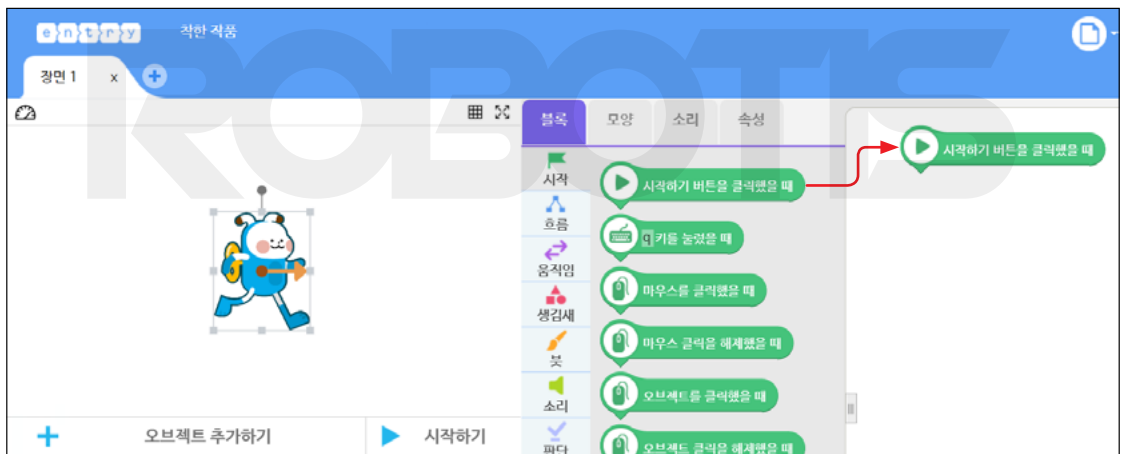
☐ 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

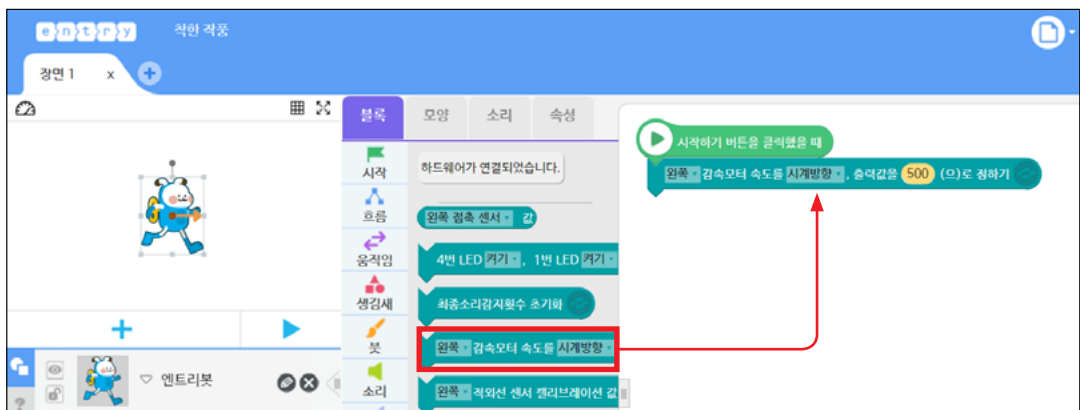
① 자동차로봇을 하드웨어와 연결하고 엔트리 프로그램을 실행합니다.



② 블록 꾸러미에서 **시작** 블록을 선택하고 **시작하기** 버튼을 클릭했을 때 블록을 오른쪽 블록 조립소에 끌어다 놓습니다.



③ 하드웨어 블록 꾸러미에 있는 **왼쪽 감속모터 속도를 시계방향**, **출력값을 500 (으)로 정하기** 블록을 끌어다 붙입니다.



- ④ 왼쪽 바퀴가 반시계 방향, 출력 300으로 회전하도록 블록을 수정합니다.

시작하기 버튼을 클릭했을 때

왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

- ⑤ 왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기 블록을 끌어다 붙이고 오른쪽 감속모터를 시계방향, 출력 300으로 회전하도록 블록을 수정합니다.

시작하기 버튼을 클릭했을 때

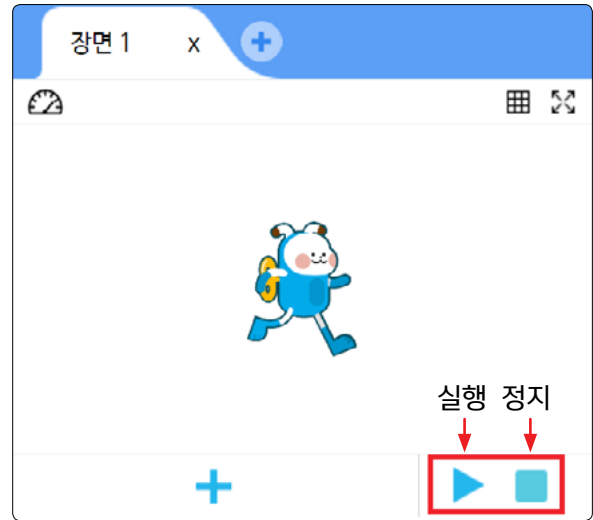
왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

ROBOTIS

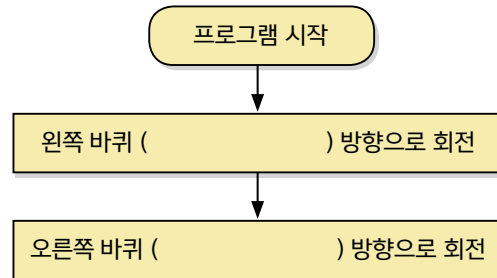
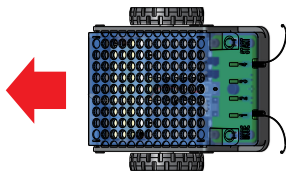
▶ 동작 확인하기

- ① ▶ **시작하기**를 클릭하여 프로그램을 실행시킵니다.
- ② 자동차로봇이 앞으로 나아가나요? 자동차로봇이 정확히 직선으로 이동하지 않고, 약간 비스듬히 이동할 수도 있습니다. 이것은 양쪽 모터의 성능차이에 의한 것이므로 정상입니다.
- ③ 프로그램을 정지시키려면 ■를 클릭합니다.
- ④ 만약 앞으로 나아가지 않고, 뒤로 가거나 좌회전, 우회전한다면 다시 생각하여 프로그램을 수정합니다.



(2) 후진

앞에서 사용했던 블록들로 자동차로봇이 후진하게 해 봅시다. 먼저 자동차로봇이 후진하려면 양쪽 바퀴에 각각 어떤 명령을 내려야 할지 생각합니다. 그리고 자동차로봇을 손으로 잡고, 뒤로 잡아당기며 양쪽 바퀴를 관찰한 뒤, 예상한 바와 같으면 이를 순서도에 나타냅니다. 이를 바탕으로 프로그램을 작성합니다.



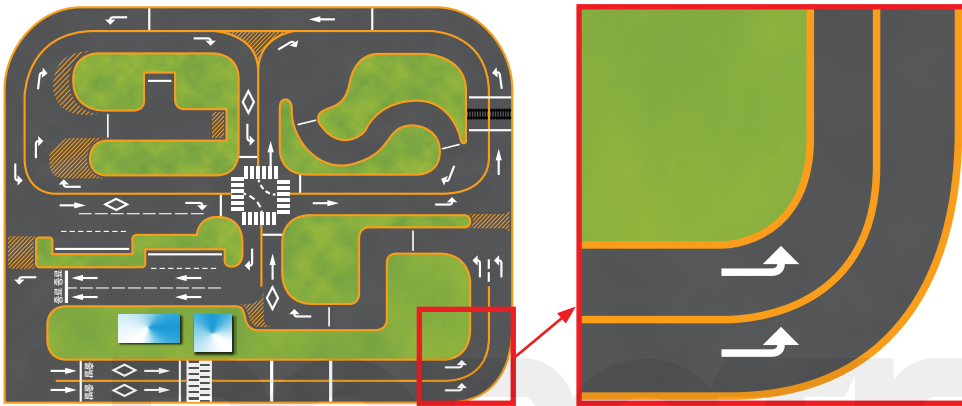
2. 곡선코스

< 학습목표 >

- 자동차로봇이 좌회전, 우회전하여 곡선코스를 통과하게 할 수 있다.
- 동작을 순차적으로 실행시킬 수 있다.

(1) 좌회전

자동차가 길을 따라 다니려면 진행 방향을 바꿀 수 있어야 합니다. 우선 좌회전해 봅시다.



사용할 블록



시작하기 버튼을 클릭했을 때

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향



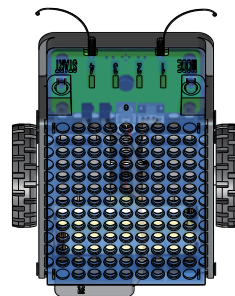
생각하기

한쪽 바퀴는 천천히 돌고, 다른 쪽 바퀴는 빨리 돌면 자동차로봇이 휘어서 나아갈 것입니다. 다음은 좌회전할 때 같은 시간 동안 양쪽 바퀴가 이동한 거리를 표시한 것입니다. 이를 참고하여 좌회전할 때 더 천천히 회전해야 하는 바퀴에 ○표해 봅시다.



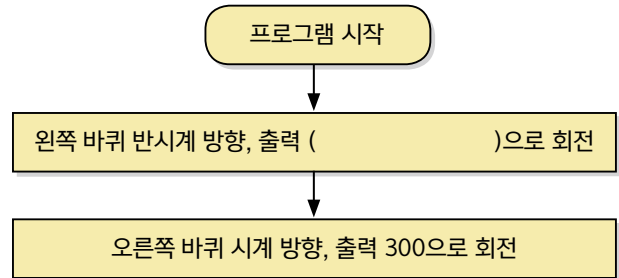
[양쪽 바퀴가 이동한 거리]

() ()



💡 생각 확인하기

손으로 자동차로봇을 제자리 좌회전시키며 양쪽 바퀴가 어느 방향으로 회전하는지 살펴 봅시다. 여러분 생각과 같나요? 코딩하기 전에 다음 순서도를 완성해 봅시다.



📺 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

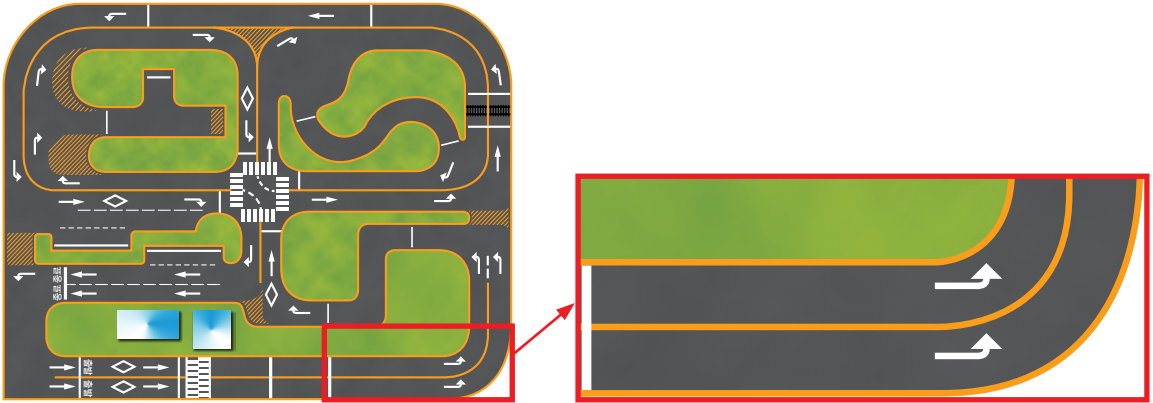


🎯 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각해 프로그램을 수정합니다.

(2) 직진 후 좌회전

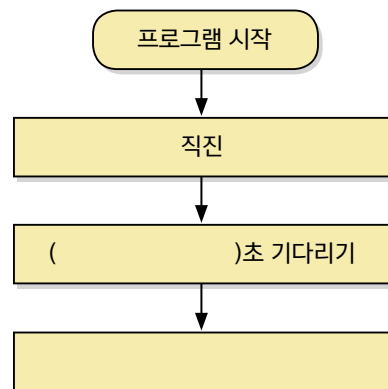
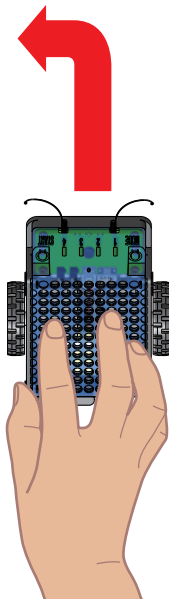
앞에서 사용했던 블록에 다음 블록을 추가하여 자동차로봇이 직진하다가 좌회전하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.



추가할 블록

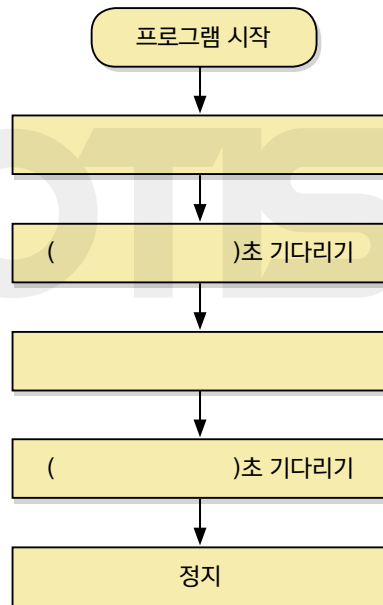
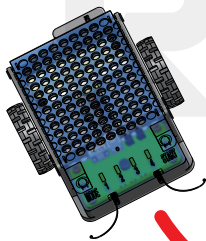
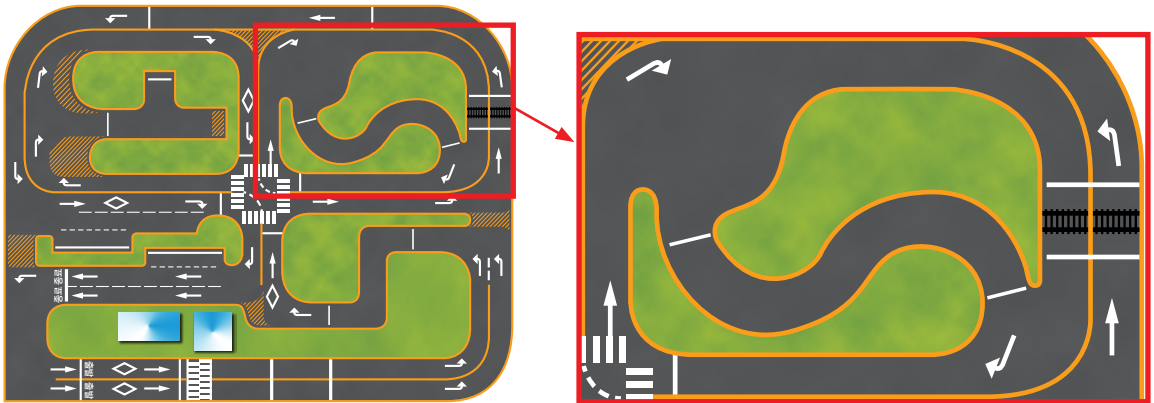
처음 사용하는 블록

2 초 기다리기 블록 꾸러미에서 **호름** - **2 초 기다리기** 블록을 이용하여 프로그램 실행을 정해진 시간 동안 멈춥니다. 동작을 일정 시간 동안 유지할 때 사용할 수 있습니다.



(3) 곡선코스

앞에서 사용한 블록들로 운전면허시험의 곡선코스처럼 S자로 이동하게 해 봅시다.
동작 순서를 생각한 뒤, 프로그램을 작성합니다.



※ 이 밖에도 좌측으로 차선변경, 우측으로 차선변경, U턴 등의 동작을 만들어 봅시다.

3. 굴절코스

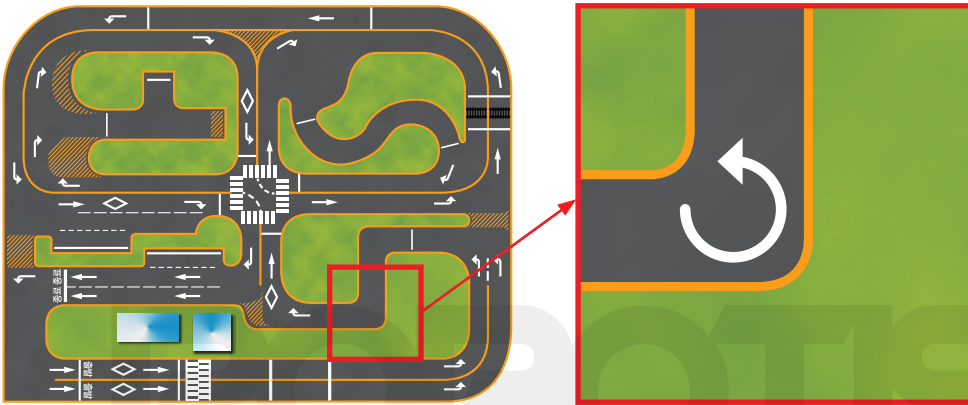
< 학습목표 >

- 자동차로봇이 제자리에서 회전하게 할 수 있다.
- 동작을 원하는 횟수만큼 반복 실행시킬 수 있다.

(1) 제자리 좌회전

양쪽 바퀴의 출력을 다르게 하면 자동차로봇이 앞으로 나아가며 회전하였습니다.

이번에는 제자리에서 회전하게 해 봅시다.



사용할 블록



시작하기 버튼을 클릭했을 때

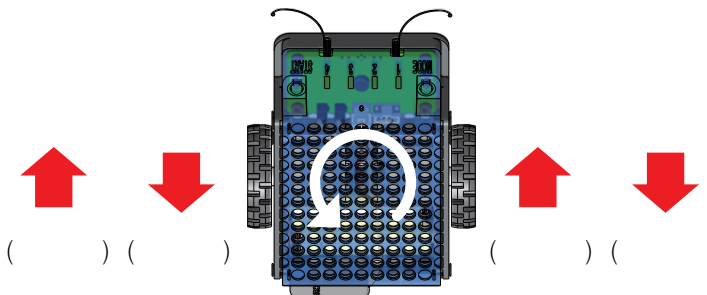
왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향

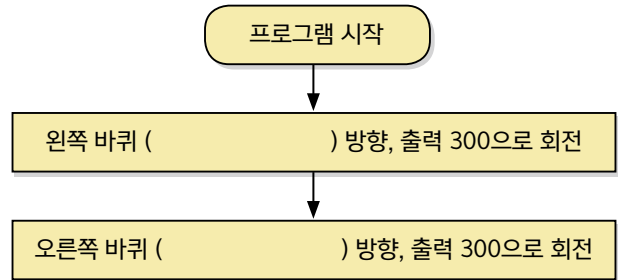
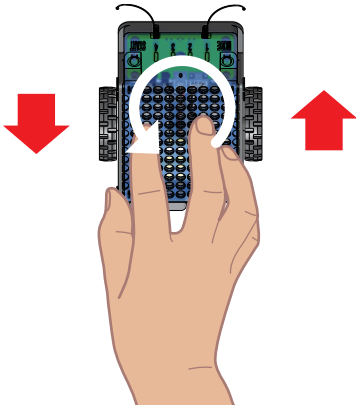
생각하기

자동차로봇이 제자리에서 좌회전하게 하려면 양쪽 바퀴를 각각 어느 방향으로 회전시켜야 할까요? 생각한 뒤, 양쪽 바퀴 옆에 ○표 해 봅시다.



💡 생각 확인하기

손으로 자동차로봇을 밀어 좌회전시키며 어느 쪽 바퀴가 더 천천히 도는지 살펴 봅시다.
여러분 생각과 같나요? 코딩하기 전에 다음 순서도를 완성해 봅시다.



💻 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

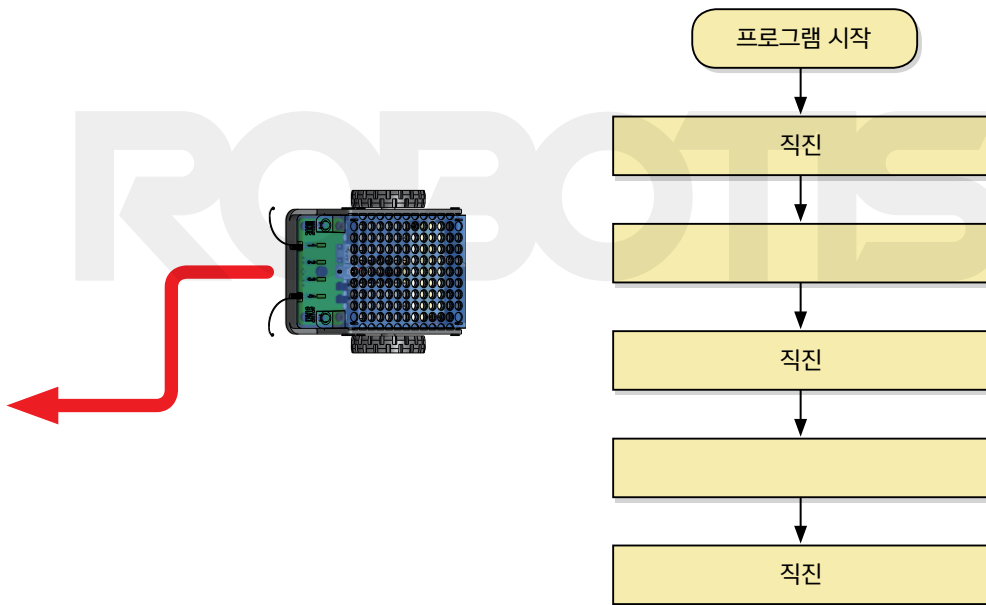
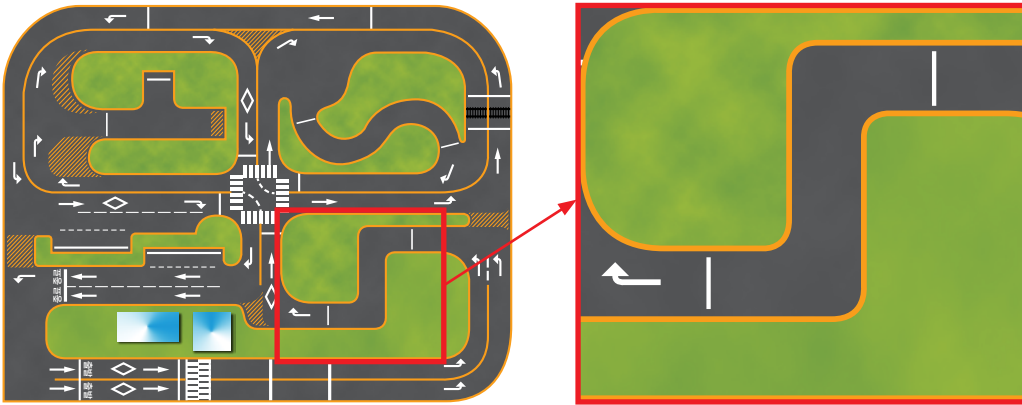


▶ 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

(2) 굴절코스

앞에서 사용했던 블록들과 다음 블록으로 자동차로봇이 굴절코스 모양으로 이동하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.



(3) P턴

좌회전이 금지된 곳에서는 P자 모양으로 돌아야 합니다. 앞에서 사용했던 블록과 다음 블록으로 자동차로봇이 P자 모양으로 이동하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

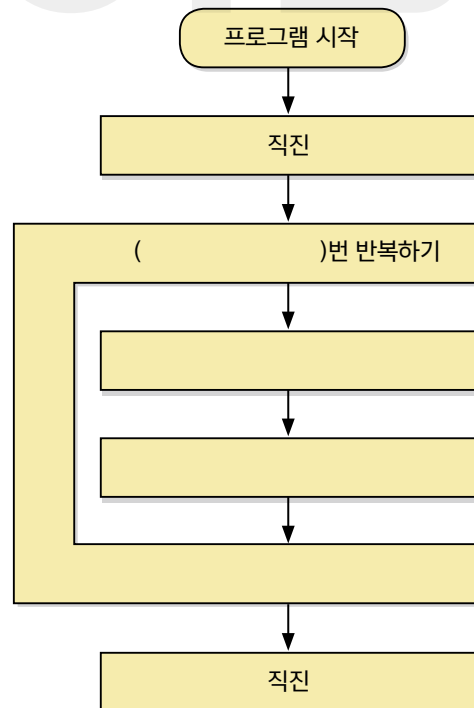
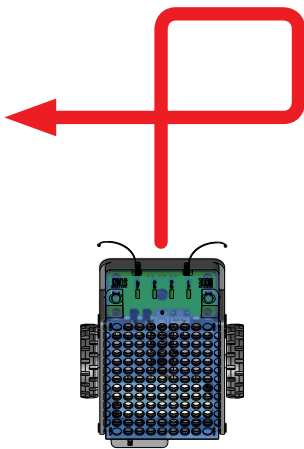


추가할 블록

처음 사용하는 블록

10 번 반복하기

정해진 횟수만큼 명령을 반복합니다.



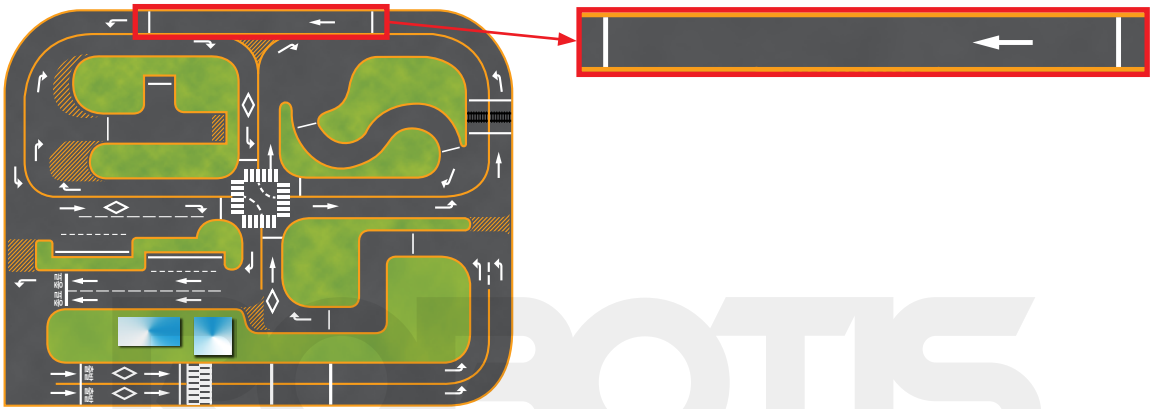
4. 변속코스

< 학습목표 >

- 자동차로봇이 점점 빠르게 또는 점점 느리게 이동하게 할 수 있다.
- 변수를 활용할 수 있다.

(1) 가속

지금까지는 자동차로봇이 일정한 빠르기로 이동했습니다. 이번에는 자동차로봇이 점점 빠르게 이동하게 해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

10 번 반복하기

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽

오른쪽

✓ 시계방향

반시계방향

처음 사용하는 블록

속력 값을 10 로 정하기 ?

변수의 값을 정합니다.

속력 에 10 만큼 더하기 ?

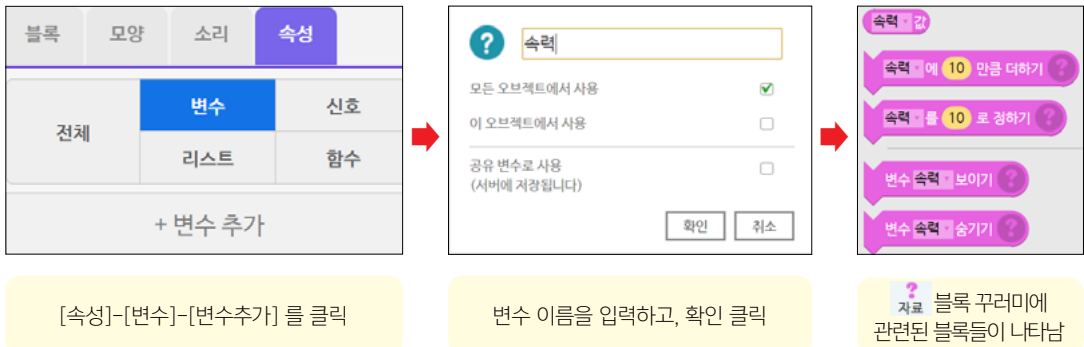
변수의 현재 값에 입력한 값 만큼 더합니다.

속력 값

변수값을 사용합니다.

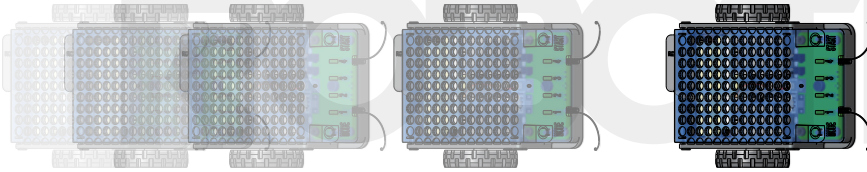
변수

변수란 값을 담아두는 상자와 같습니다. 변수를 이용하면 값을 일괄적으로 입력, 수정할 수 있어 편리합니다. 엔트리에서 변수를 새로 만드는 방법은 다음과 같습니다.



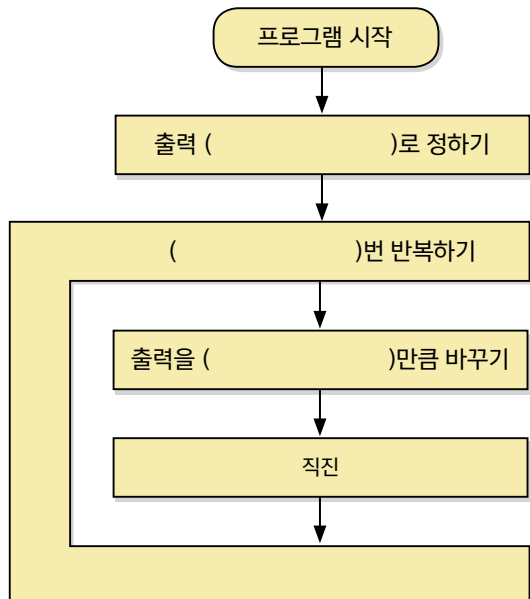
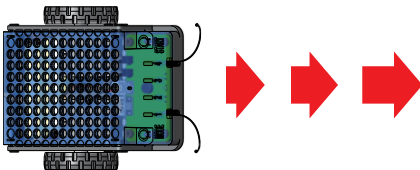
생각하기

감속모터 출력을 어떻게 변화시켜야 자동차로봇이 점점 빠르게 나아갈지 생각해 봅시다.



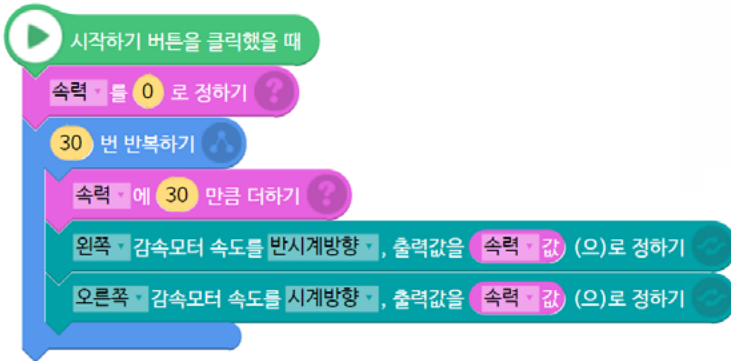
생각 확인하기

자동차로봇을 손으로 점점 빠르게 밀며 양쪽 바퀴가 도는 빠르기를 살펴 봅시다. 여러분이 생각했던 것과 같나요? 코딩하기 전에 순서도를 완성해 봅시다.



☞ 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

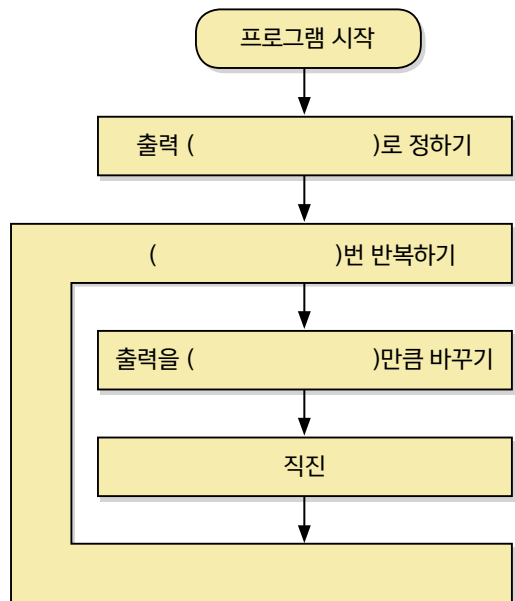
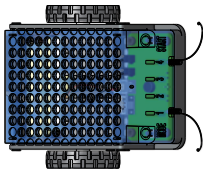


🕒 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

(2) 감속

앞에서 사용했던 블록들로 자동차로봇이 점점 느리게 나아가게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.



5. 돌발사고 시 급정지 및 출발

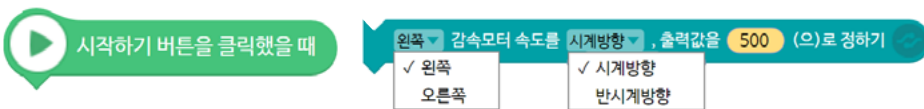
< 학습목표 >

- 돌발상황을 감지하면 자동차로봇이 정지하게 할 수 있다.
- 조건에 따라 동작을 다르게 할 수 있다.

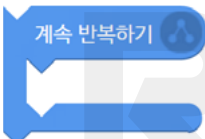
(1) 돌발사고 시 급정지

돌발사고가 감지되면 추가 사고를 방지하기 위하여 급하게 정지해야 합니다. 운전면허시험에서는 운행 중 경고음이 나면 급정지하여 비상등을 켜는 시험항목이 있습니다. 우리도 운행 중 소리가 감지되면 정지하게 해 봅시다.

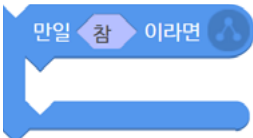
사용할 블록



처음 사용하는 블록



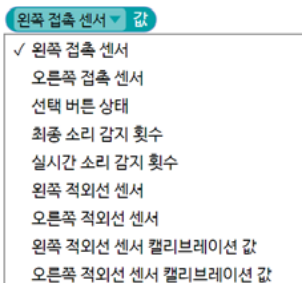
블록이 감싸고 있는 명령들을 계속 반복합니다.



조건에 만족하면 블록이 감싸고 있는 명령을 실행합니다.



양쪽의 값을 비교합니다.

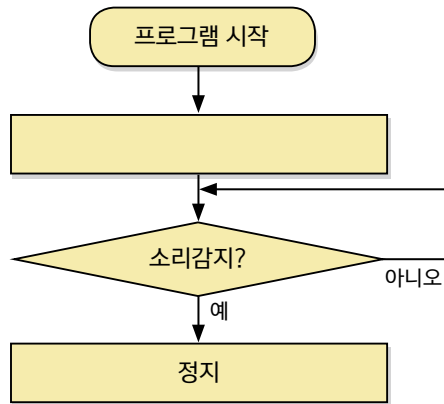


제어기의 센서에서 입력된 값을 변수로 사용할 수 있습니다.



생각하기

앞으로 나아가던 자동차로봇이 소리를 감지하면 정지하게 하려고 합니다. 동작 순서를 생각하여 순서도를 완성하시오.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.



동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

(2) 돌발사고 시 급정지 후 비상등 켜기

앞에서 사용했던 블록과 다음 블록들로 소리를 감지하면 급정지하여 비상등(양쪽 LED)을 5번 깜빡이게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

추가할 블록

10 번 반복하기

2 초 기다리기

처음 사용하는 블록

4번 LED 켜기, 1번 LED 켜기

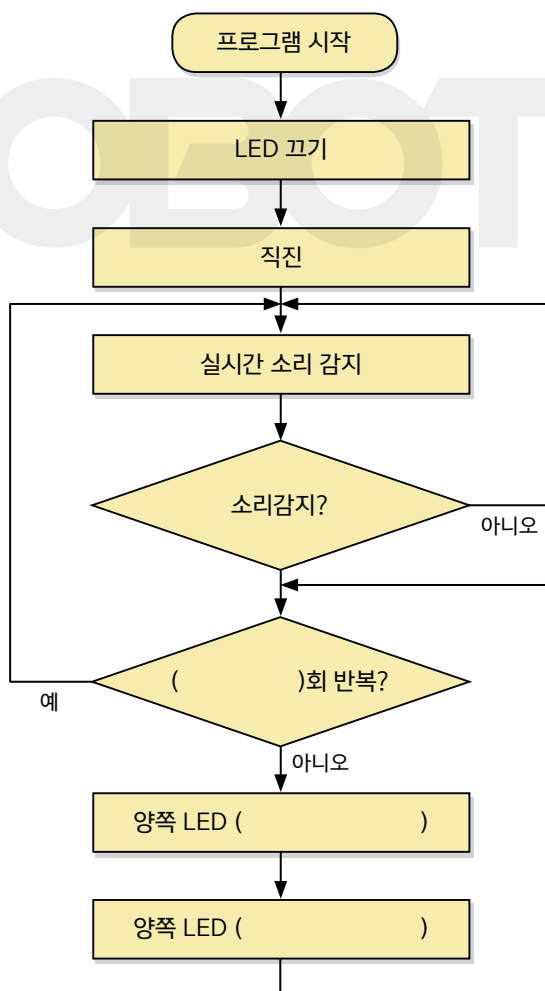
✓ 켜기

끄기

✓ 켜기

끄기

왼쪽(4번)과 오른쪽(1번) LED를 각각 켜거나 끕니다.



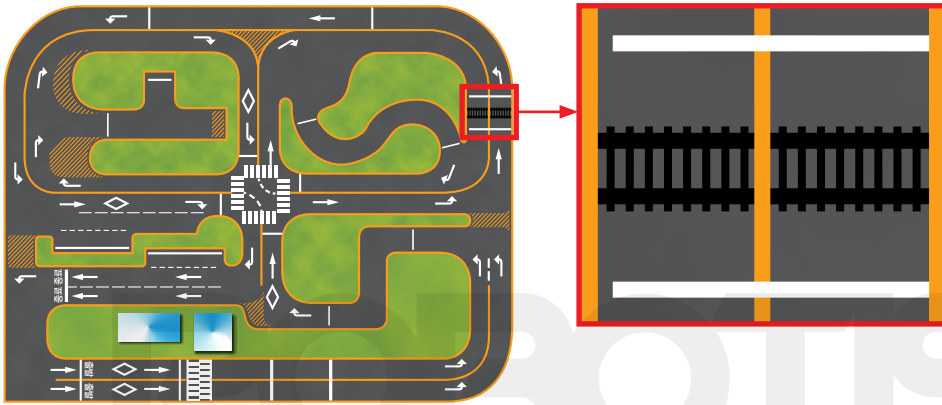
6. 철길건널목 통과

< 학습목표 >

- 안전바를 감지하면 정지하였다가 출발하게 할 수 있다.
- 조건 여러 개를 동시에 검사할 수 있다.

(1) 철길건널목에서 정지

철길건널목 사고는 자칫 대형 사고로 이어질 수 있어 주의가 필요합니다. 자동차로봇의 접촉센서에 안전바가 감지되면 정지하게 해 봅시다. 안전바를 대신해 손으로 자동차로봇을 가로 막습니다.



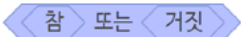
사용할 블록



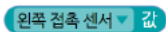
처음 사용하는 블록



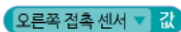
양쪽의 값이 같은지 비교합니다.



양쪽의 조건 중 하나라도 만족하는지 검사합니다.



왼쪽 접촉센서에 물체가 감지되면 1, 감지되지 않으면 0을 변수에 저장합니다.

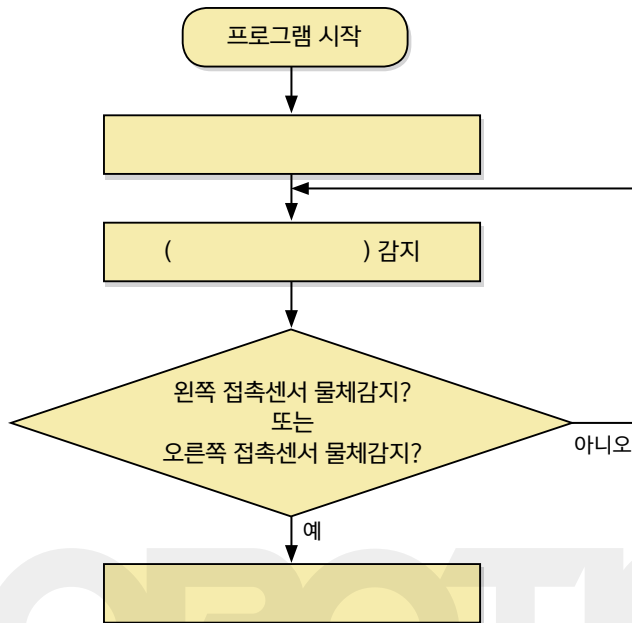


오른쪽 접촉센서에 물체가 감지되면 1, 감지되지 않으면 0을 변수에 저장합니다.



생각하기

앞으로 나아가던 자동차로봇의 접촉센서에 물체가 감지되면 정지하게 하려고 합니다.
동작 순서를 생각하여 순서도를 완성하십시오.



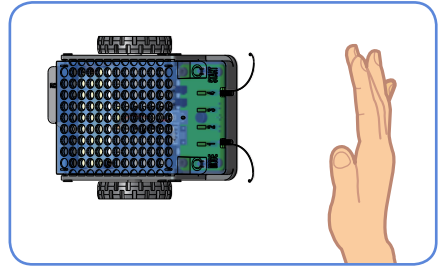
생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.



▶ 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 직진하는 자동차로봇 앞을 손으로 가로막았을 때 정지하는지 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

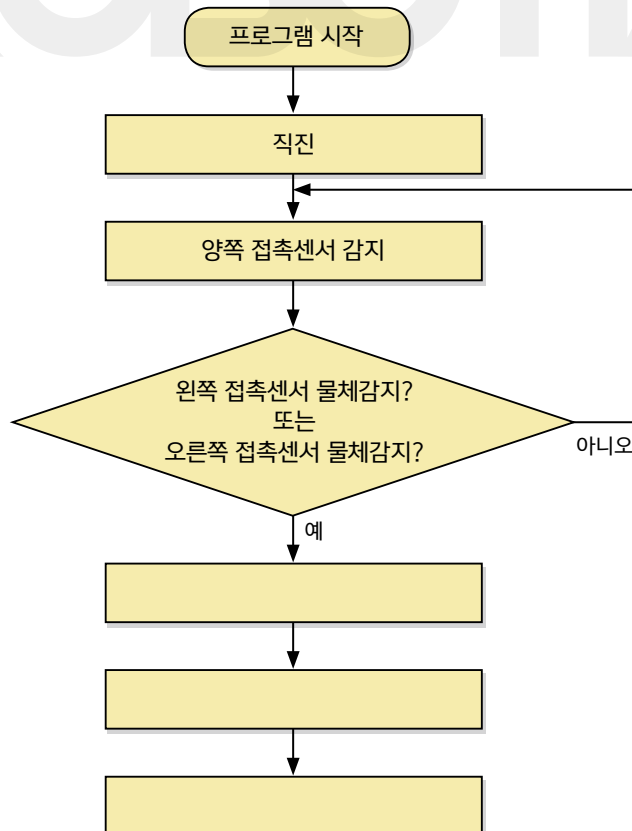


(2) 철길건널목에서 후진 후 정지

철길건널목 안전바에 너무 가까이 정지해 있으면 위험합니다. 앞에서 사용한 블록과 다음 블록으로 접촉센서에 물체가 감지되면 약간 뒤로 물러나 정지하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

추가할 블록

2 초 기다리기



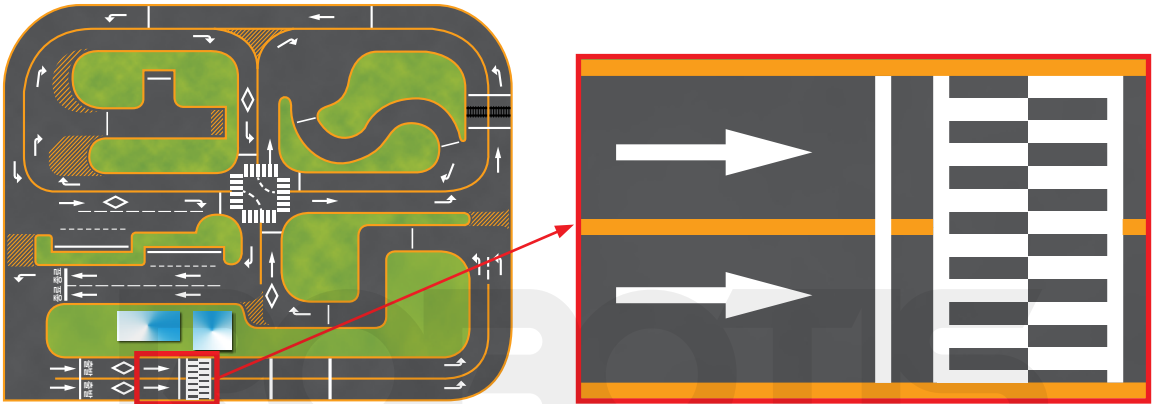
7. 횡단보도 일시정지

< 학습목표 >

- 정지선을 감지하면 일시정지할 수 있다.
- 조건 여러 개가 동시에 만족하는지 검사할 수 있다.

(1) 정지선에서 정지

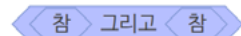
횡단보도는 사람이 건너는 길이므로 언제나 주의해야 합니다. 따라서 횡단보도 앞에서는 일시적으로 정지하였다가 출발해야 합니다. 바닥의 정지선을 감지하면 정지하게 해 봅시다.



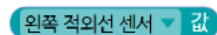
사용할 블록



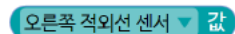
처음 사용하는 블록



양쪽 조건에 모두 만족하는지 검사합니다.



바닥의 왼쪽 적외선센서의 값을 저장합니다.

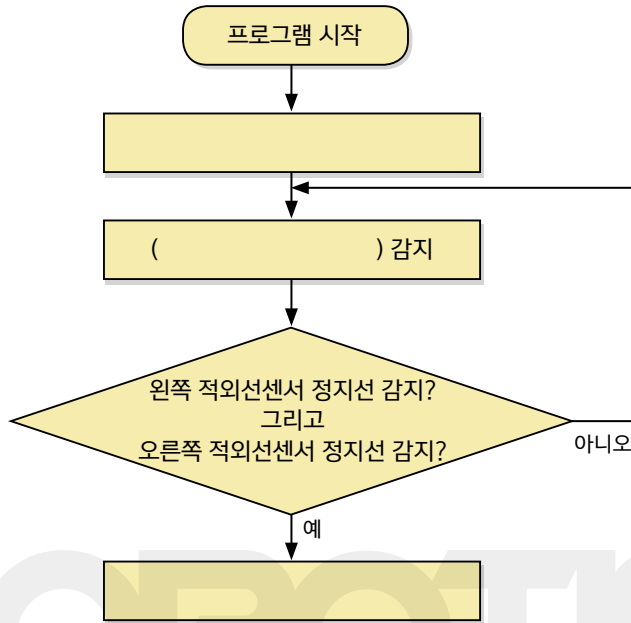


바닥의 오른쪽 적외선센서의 값을 저장합니다.



생각하기

앞으로 나아가던 자동차로봇이 정지선을 감지하면 잠시 정지하게 하려고 합니다.
동작 순서를 생각하여 순서도를 완성하시오.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.



시작하기 버튼을 클릭했을 때

왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

계속 반복하기

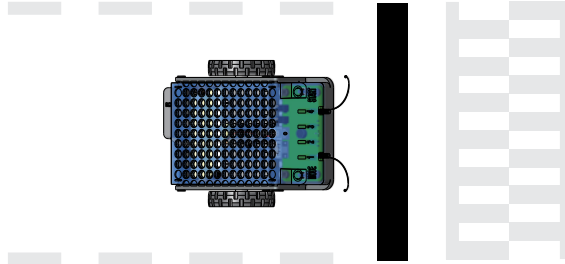
만일 왼쪽 적외선 센서 값 < 300 또는 오른쪽 적외선 센서 값 < 300 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

▶ 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.



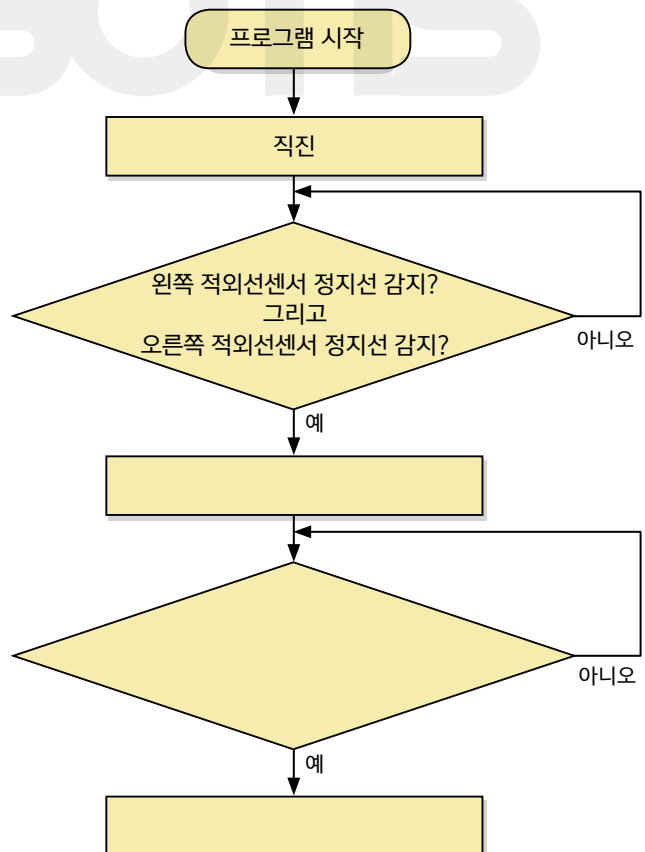
(2) 출발선에서 대기하다가 출발음이 감지되면 직진

앞에서 사용한 블록과 다음 블록으로 정지선에서 대기하던 자동차로봇이 출발음을 감지하면 다시 직진하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

추가할 블록

실시간 소리 감지 횟수 ▼ 값

- 왼쪽 접촉 센서
- 오른쪽 접촉 센서
- 선택 버튼 상태
- 최종 소리 감지 횟수
- ✓ 실시간 소리 감지 횟수
- 왼쪽 적외선 센서
- 오른쪽 적외선 센서
- 왼쪽 적외선 센서 캘리브레이션 값
- 오른쪽 적외선 센서 캘리브레이션 값



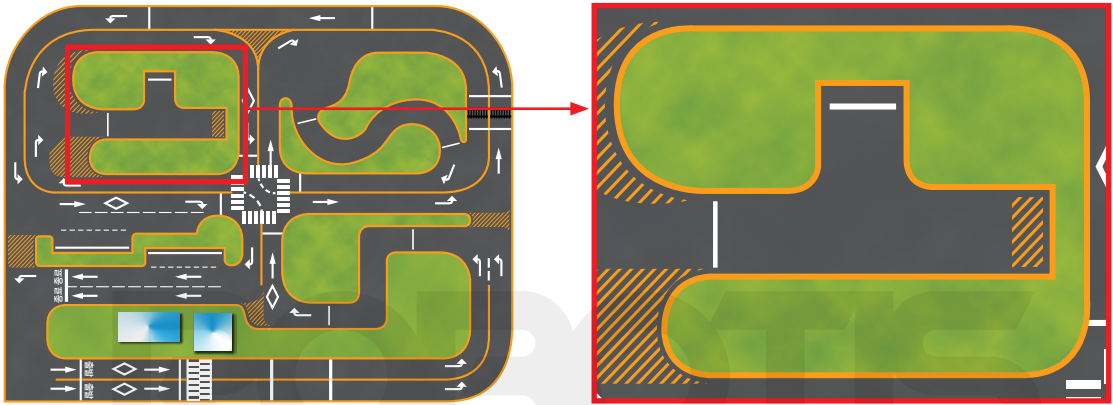
8. 후진주차코스

< 학습목표 >

- 적외선센서를 활용하여 정해진 위치에 후진으로 주차할 수 있다.
- 조건에 만족할 때까지 명령을 반복할 수 있다.

(1) 후진주차

막다른 길에서 차를 돌리거나 주차장에 주차하려면 후진으로 정해진 위치까지 이동할 수 있어야 합니다. 적외선센서를 활용하여 자동차로봇이 정해진 곳까지 후진으로 이동하게 해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

계속 반복하기

왼쪽 적외선 센서 > 값

오른쪽 적외선 센서 > 값

10 < 10

10 > 10

참 그리고 참

왼쪽 > 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽

오른쪽

✓ 시계방향

반시계방향

처음 사용하는 블록

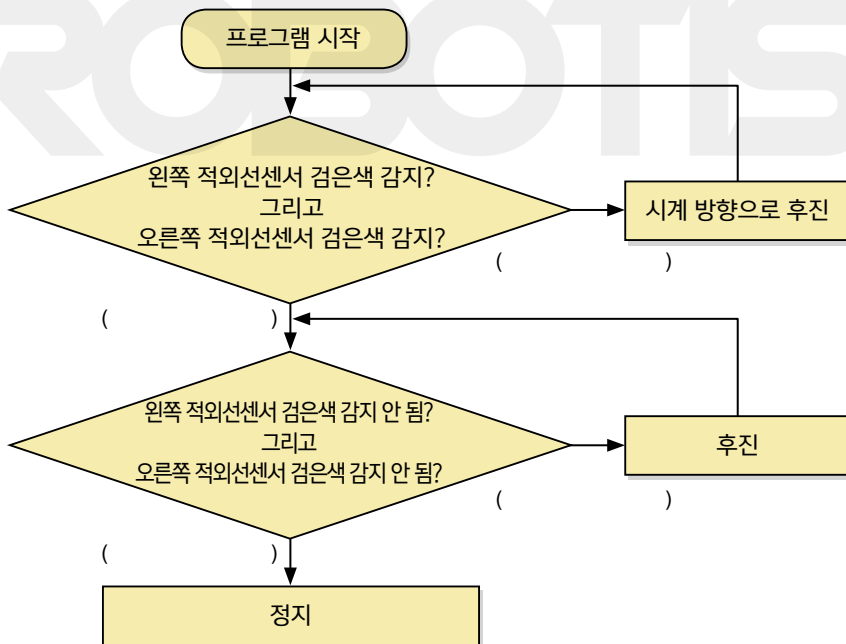
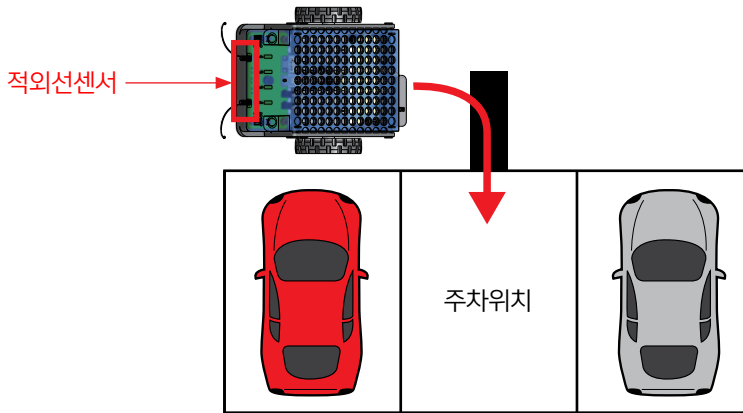
참 이 될 때까지 > 반복하기

조건에 만족할 때까지 명령을 반복합니다.



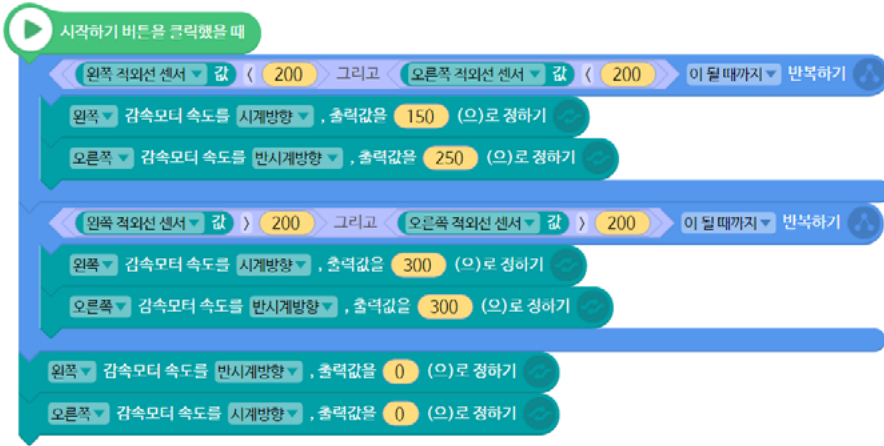
생각하기

적외선센서를 활용하여 자동차로봇을 후진주차하려고 합니다. **쪽 활동자료집 위에서 어떤 순서로 동작해야 할지 생각하고, 생각한대로 자동차로봇을 움직여 봅시다. 예상한대로 주차위치에 들어간다면 순서도를 완성해 봅시다.



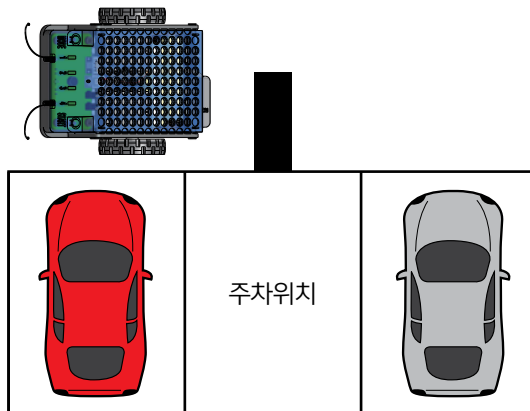
☞ 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다. 이때 모터 출력 값이나 센서 기준값은 건전지, 주변 밝기와 같은 환경에 따라 달라질 수 있으므로 각자 조절해야 합니다.



☞ 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 8~10쪽 활동자료를 활용합니다. 이때 모터 출력 값이나 센서 기준값은 건전지, 주변 밝기와 같은 환경에 따라 달라질 수 있으므로 각자 조절해야 합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.



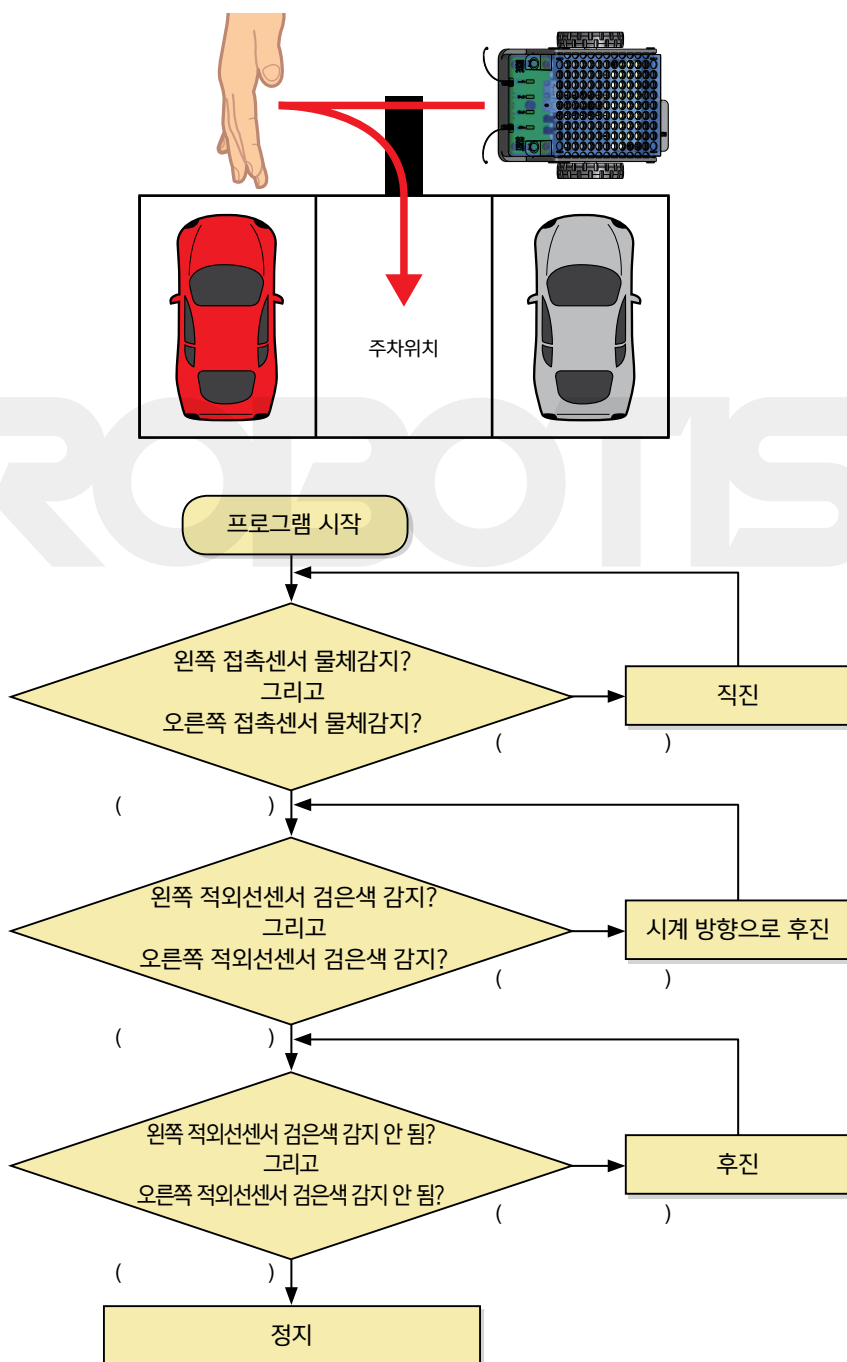
(2) 막다른 골목에서 방향 전환

앞에서 사용한 블록과 다음 블록으로 자동차로봇이 벽(손)을 감지하면 지정된 위치까지 후진으로 이동하게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

추가할 블록

왼쪽 적외선 센서 ▼ 감

오른쪽 적외선 센서 ▼ 감



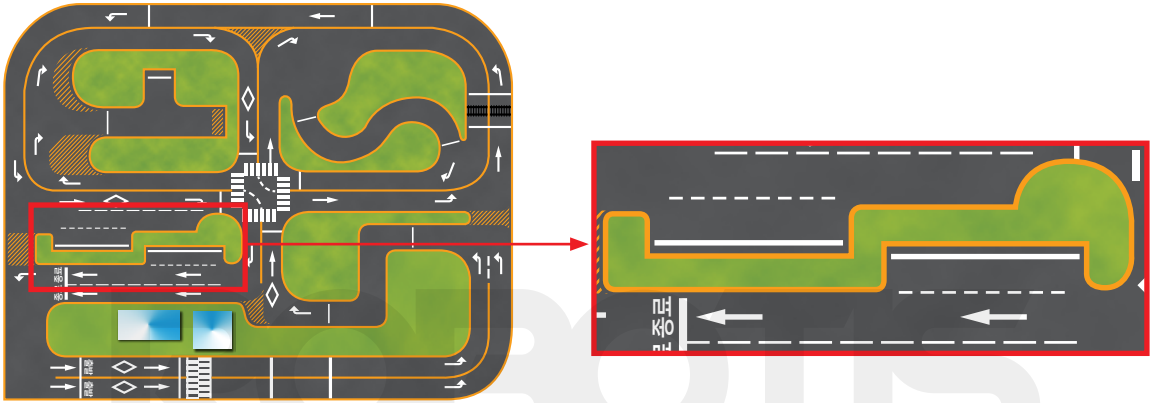
9. 평행주차코스

< 학습목표 >

- 적외선센서를 활용하여 평행 주차할 수 있다.
- 조건에 만족할 때까지 프로그램 진행을 멈추고 대기할 수 있다.

(1) 평행 주차

자동차가 나아가는 방향과 평행하게 주차하는 것을 평행주차라고 합니다. 적외선센서를 활용하여 자동차로봇을 정해진 곳에 평행주차해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 참 이라면

10 < 10 10 > 10 참 그리고 참

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

왼쪽 적외선 센서 값 오른쪽 적외선 센서 값

✓ 왼쪽 ✓ 시계방향 반시계방향

처음 사용하는 블록

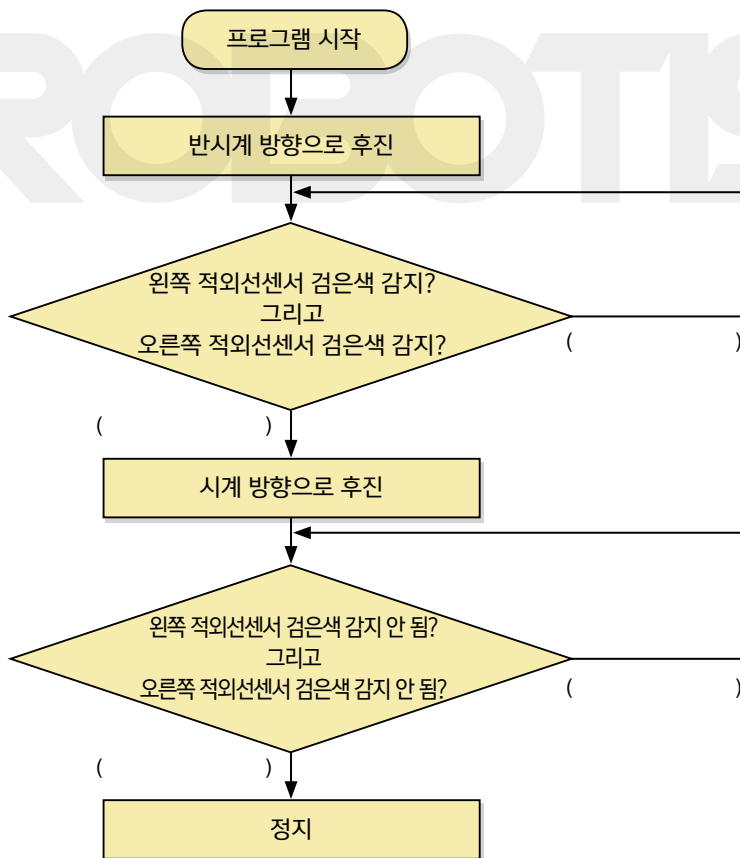
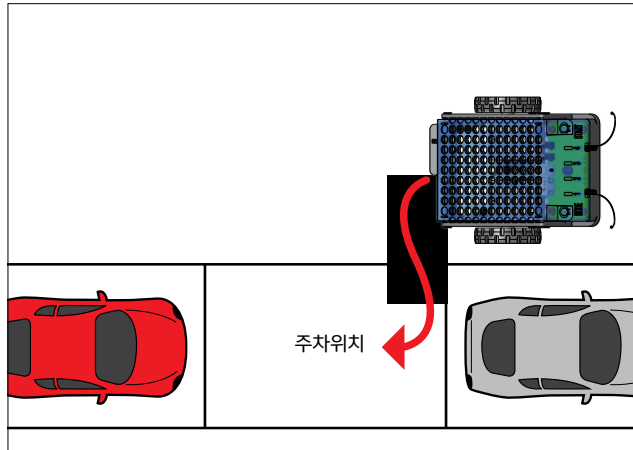
참 이 (가) 될 때까지 기다리기

조건에 만족할 때까지 기다립니다.



생각하기

적외선센서를 활용하여 자동차로봇을 평행 주차하려고 합니다. **쪽 활동자료집 위에서 자동차로봇을 어떻게 동작시켜야 할지 생각하고, 생각한대로 움직여 봅시다. 생각대로 주차위치에 들어간다면 순서도를 완성해 봅시다.



☞ 생각 적용하기

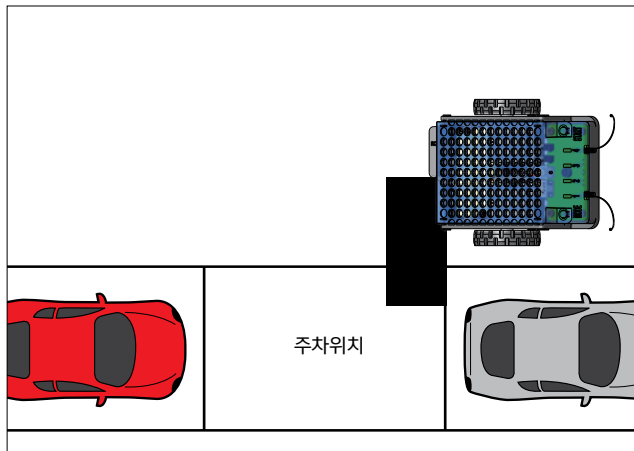
생각한대로 자동차로봇이 동작하도록 코딩해 봅시다. 이때 모터 출력 값이나 센서 기준값은 건전지, 주변 밝기와 같은 환경에 따라 달라질 수 있으므로 각자 조절해야 합니다.



ROBOTIS

🎯 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 8~10쪽 활동자료를 활용합니다. 이때 모터 출력 값이나 센서 기준값은 건전지, 주변 밝기와 같은 환경에 따라 달라질 수 있으므로 각자 조절해야 합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

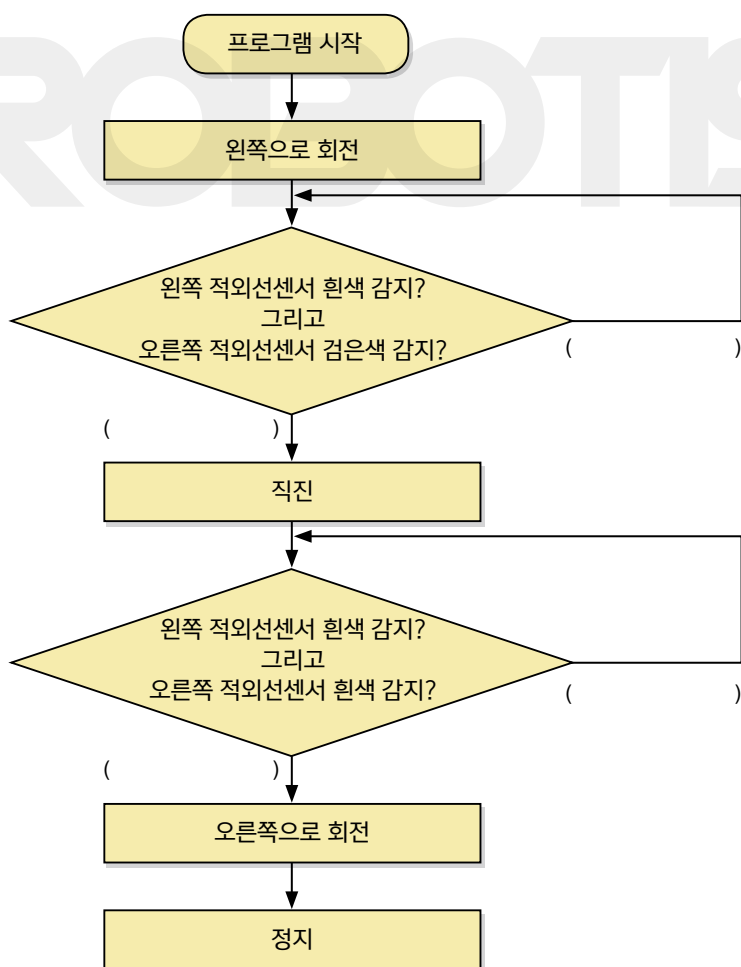
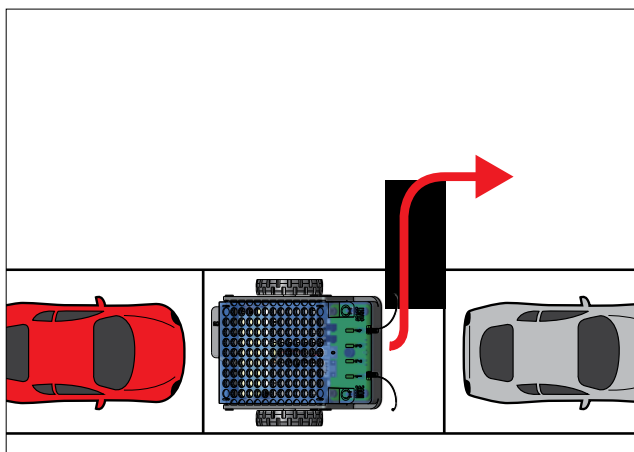


(2) 평행주차구역에서 빠져나오기

앞에서 사용한 블록과 다음 블록으로 주차되어 있는 자동차로봇이 적외선센서를 활용하여 주차구역을 빠져나오게 해 봅시다. 어떤 순서로 명령을 내려야 할지 생각한 뒤, 순서도를 완성합니다. 그리고 프로그램을 작성해 봅시다.

추가할 블록

2 초 기다리기



10. 키보드 조종

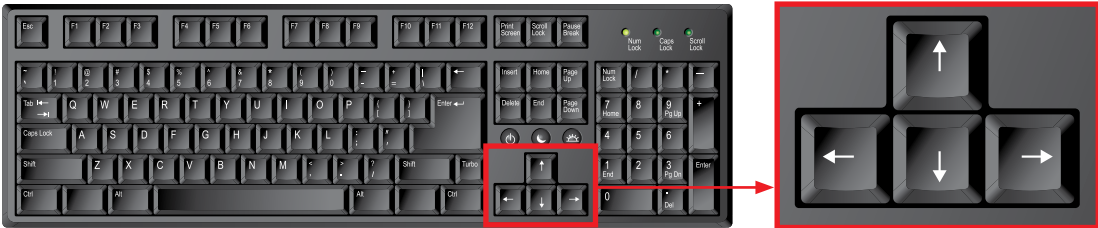
< 학습목표 >

- 키보드로 자동차로봇을 조종하는 프로그램을 만들 수 있다.
- 키보드로 자동차로봇을 조종하여 운전면허시험코스를 통과할 수 있다.

(1) 화살표키로 전후좌우로 이동

키보드에는 네 방향의 화살표가 있습니다.

화살표키를 눌렀을 때 각 방향으로 자동차로봇이 이동하게 해 봅시다.



사용할 블록

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽

오른쪽

✓ 시계방향

반시계방향

처음 사용하는 블록



키를 눌렀을 때



시작 블록 꾸러미에서



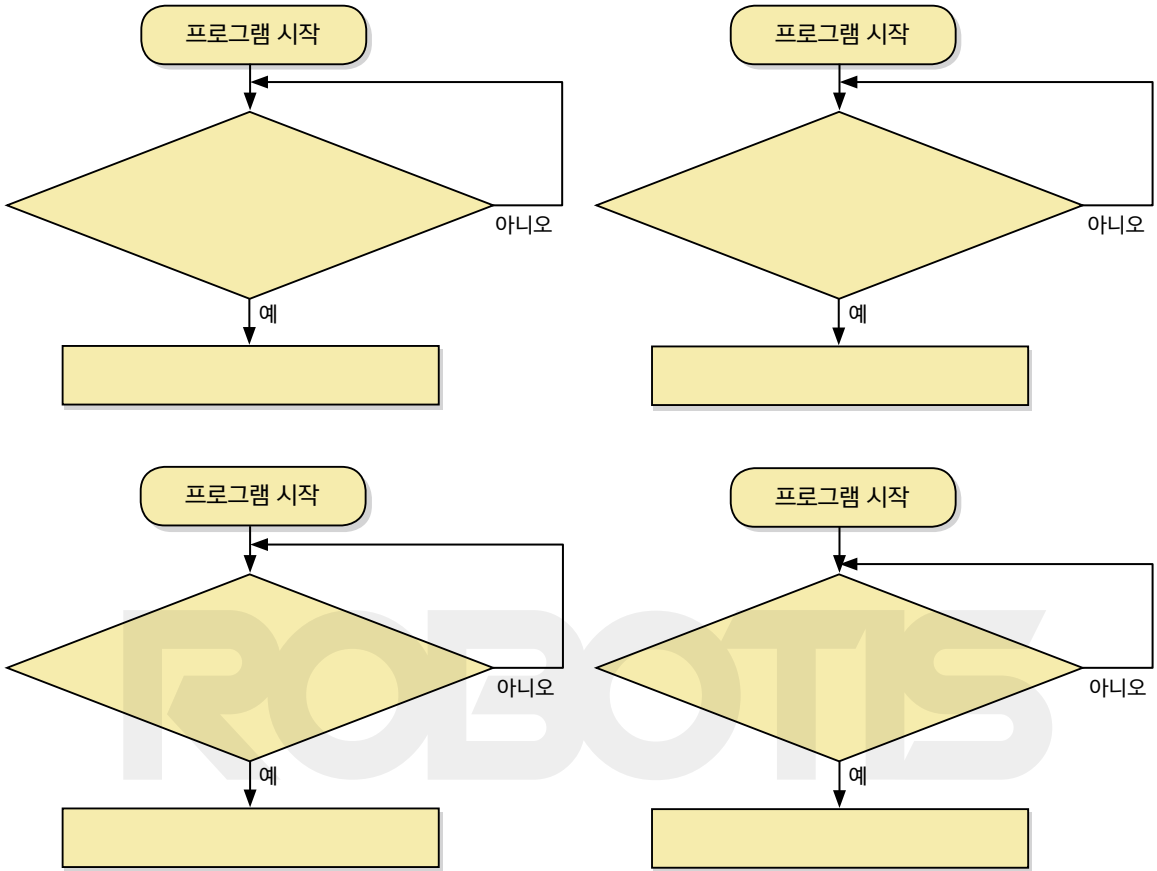
키를 눌렀을 때

블록을 이용하여 키보드의 어떤 키를 누르는지 감지합니다.



생각하기

각 화살표를 누를 때 어느 방향으로 나아가게 할지 생각하고, 순서대로 나타내어 봅시다.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

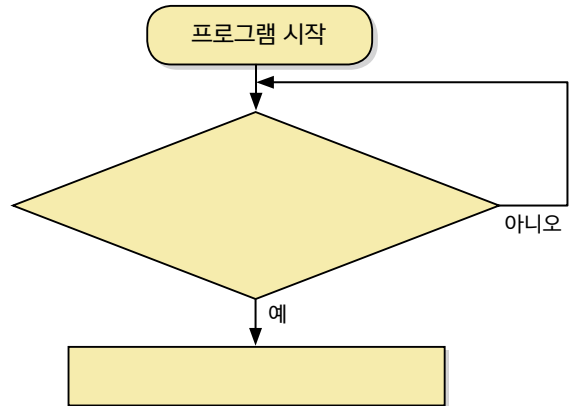


동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 지금까지 사용했던 활동자료집들을 활용하여 각 코스대로 동작시켜 봅시다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

(2) 스페이스키로 정지

같은 방식으로 키보드의 스페이스키를 눌렀을 때 자동차로봇이 정지하게 해 봅시다.



(3) 다른 블록을 사용하여 같은 명령하기

키보드의 키를 눌러 명령을 내릴 때 키를 눌렀을 때 블록을 사용했습니다. 다음 블록들을 사용하여 화살표와 스페이스키를 눌렀을 때 전진, 후진, 좌회전, 우회전, 정지 명령을 내려 봅시다.

사용할 블록



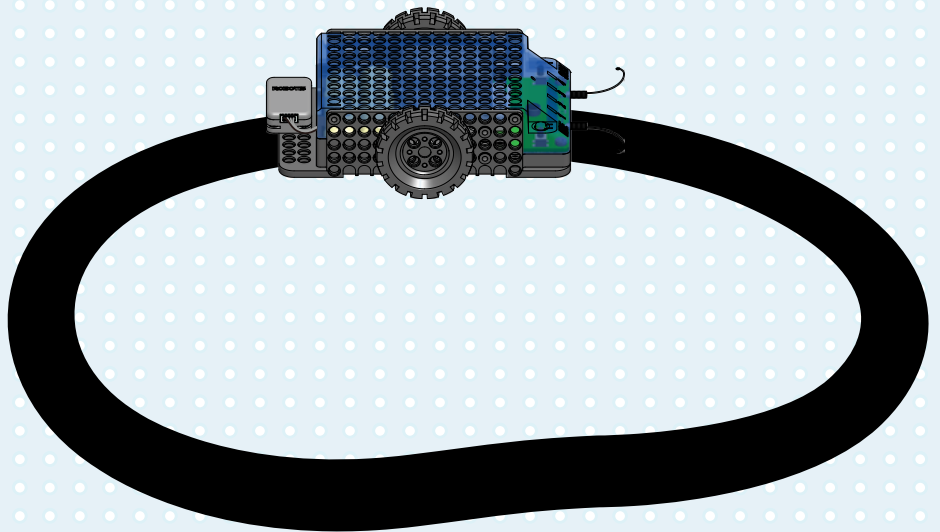
처음 사용하는 블록



판단 블록 꾸러미의 **q 키가 눌러져 있는가?** 블록을 이용하여 키보드의 어떤 키가 눌리는지 검사합니다.

ROBOTIS 로보티즈
자동차로봇 [엔트리판]

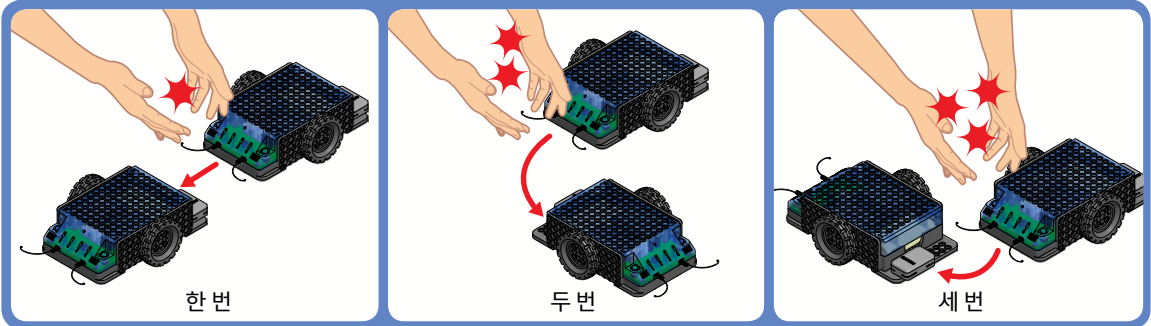
2부
기본 동작



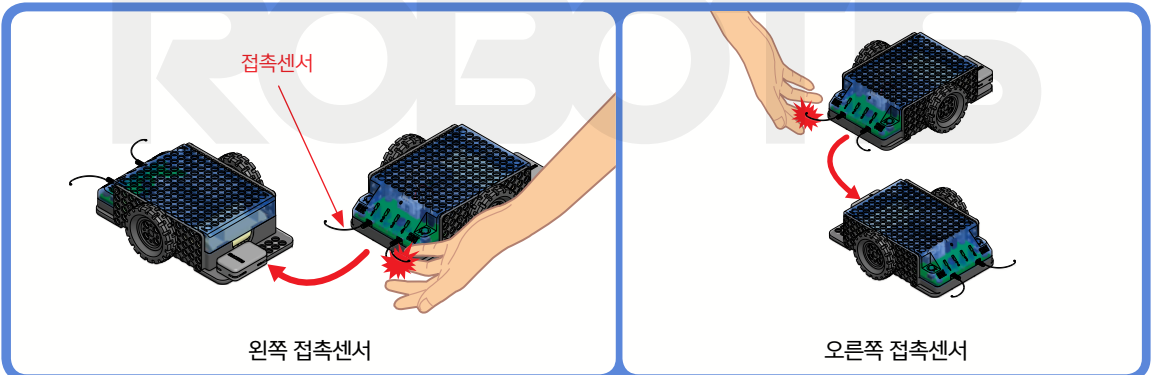
2부. 기본 동작

자동차로봇에 기본으로 들어 있는 동작들을 엔트리로 구현해 봅시다. 그리고 좀 더 다양한 동작을 하기 위해 변수를 활용하는 방법을 알아 봅시다.

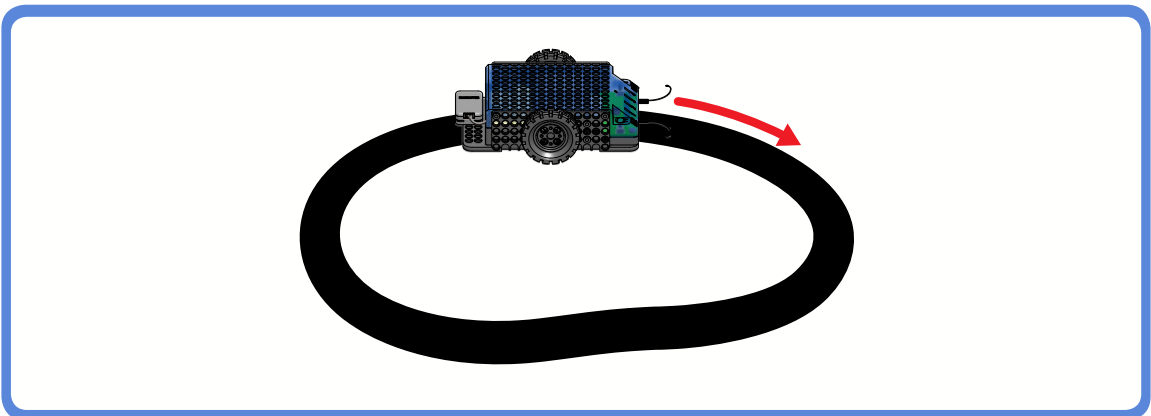
소리센서 모드



접촉센서 모드



적외선센서 모드



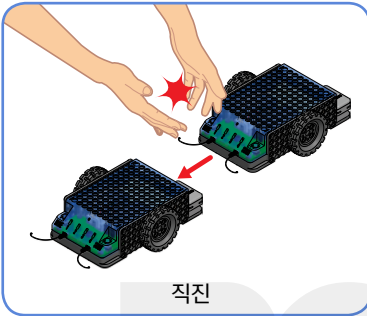
11. 소리센서 모드

< 학습목표 >

- 최종소리횟수에 따라 동작하는 프로그램을 만들 수 있다.
- 일정 시간 동안만 동작하고 정지하게 할 수 있다.

우선 자동차로봇을 소리센서 모드로 동작시켜 봅시다. 다음과 같이 동작하나요? 박수소리가 나면 정해진 방향으로 일정 시간(약 0.5초) 동안 동작합니다. 다음의 블록들로 소리센서 모드 프로그램을 완성해 봅시다.

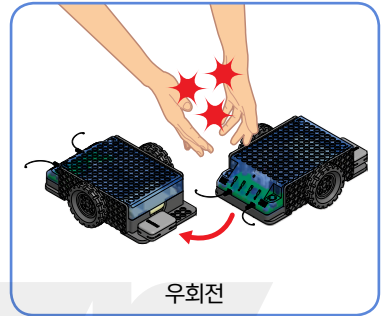
박수 한 번



박수 두 번



박수 세 번



사용할 블록



처음 사용하는 블록

최종 소리 감지 횟수 ▼ 값

- 왼쪽 접촉 센서
- 오른쪽 접촉 센서
- 선택 버튼 상태
- ✓ **최종 소리 감지 횟수**
- 실시간 소리 감지 횟수
- 왼쪽 적외선 센서
- 오른쪽 적외선 센서
- 왼쪽 적외선 센서 캘리브레이션 값
- 오른쪽 적외선 센서 캘리브레이션 값

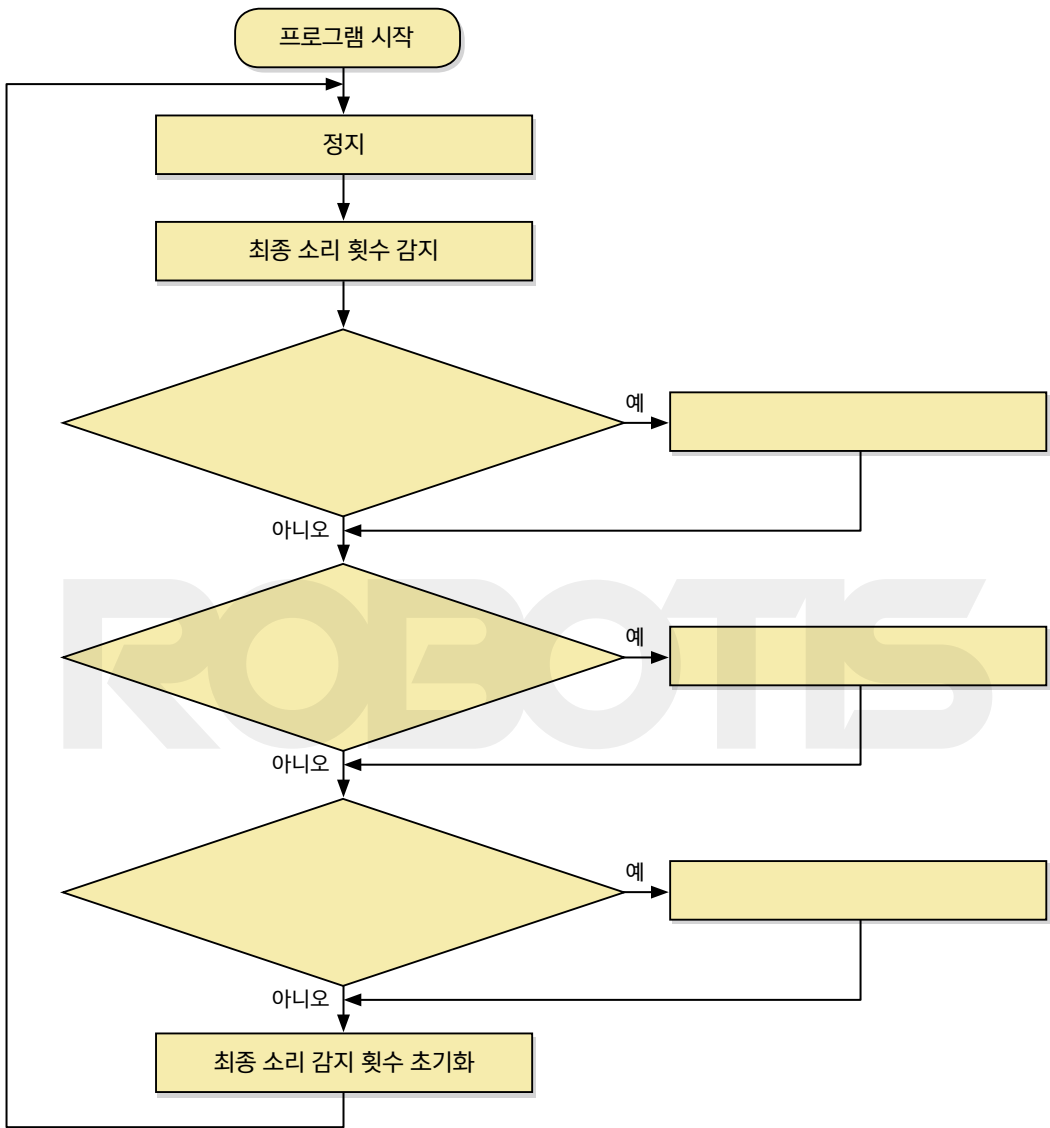
제어기에 들어 있는 마이크에는 소리횟수를 감지하는 기능이 있습니다. 연속적인 소리 입력이 멈추면 '최종 소리 감지 횟수' 값이 입력됩니다.

최종소리감지횟수 초기화

최종 소리감지 횟수는 자동으로 초기화되지 않기 때문에 직접 0으로 초기화해 주어야 합니다.

생각하기

어떻게 하면 최종 소리 감지 횟수에 따라 일정 시간 동안 동작시킬 수 있을까요? 동작순서를 생각하고, 순서도를 완성해 봅시다.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각해 프로그램을 수정합니다.

완성 코드

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 최종 소리 감지 횟수 값 = 1 이라면

- 왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기
- 0.5 초 기다리기

만일 최종 소리 감지 횟수 값 = 2 이라면

- 왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기
- 0.5 초 기다리기

만일 최종 소리 감지 횟수 값 = 3 이라면

- 왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기
- 0.5 초 기다리기

최종소리감지횟수 초기화

- 왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

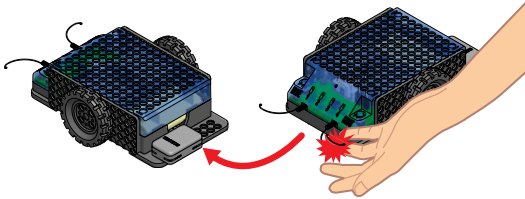
12. 접촉센서 모드

< 학습목표 >

- 접촉센서를 이용해 물체를 피해 이동하는 프로그램을 만들 수 있다.

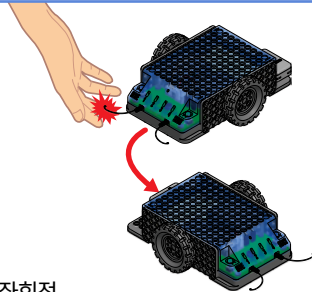
자동차로봇을 접촉센서 모드로 동작시켜 봅시다. 다음과 같이 동작하나요? 접촉센서에 물체가 감지되면 물체를 피해 나아갑니다. 다음의 블록들로 접촉센서 모드 프로그램을 완성해 봅시다.

왼쪽 접촉센서에 물체가 닿을 때



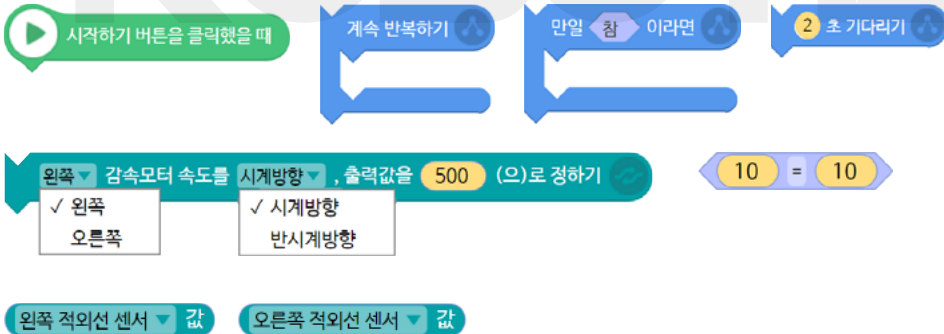
후진 후 우회전

오른쪽 접촉센서에 물체가 닿을 때



후진 후 좌회전

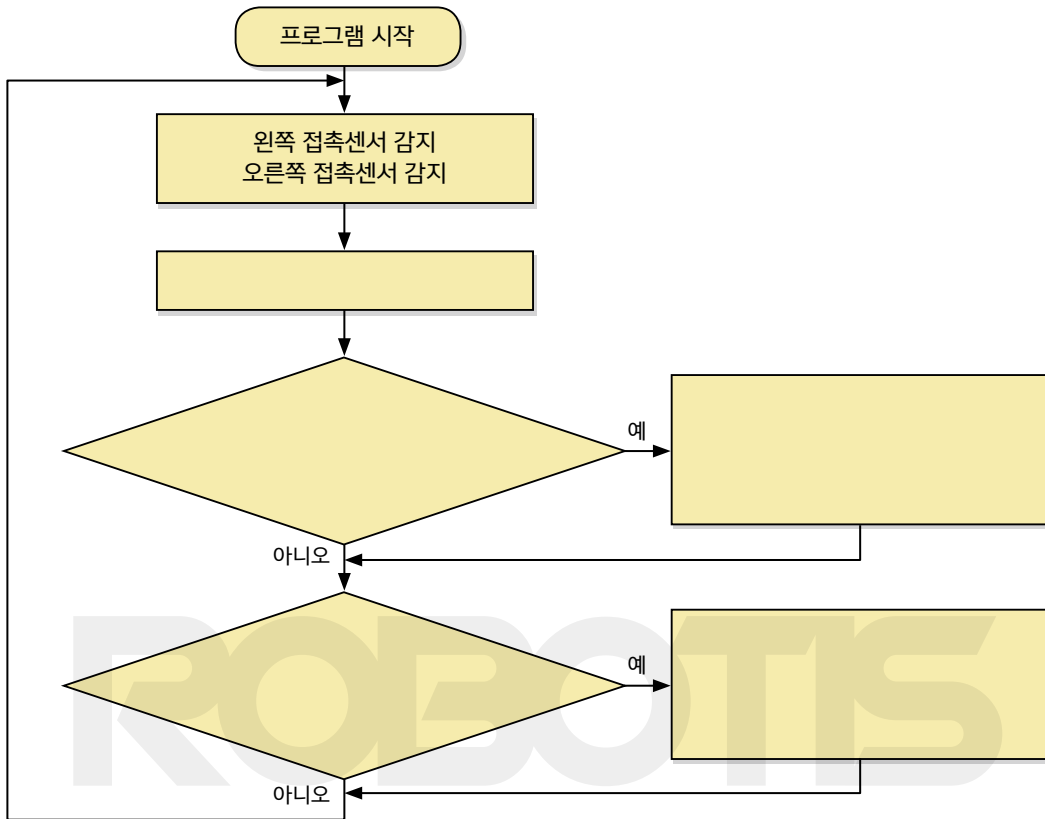
사용할 블록





생각하기

어떻게 하면 물체를 피할 수 있을까요? 동작순서를 생각하고, 순서도를 완성해 봅시다.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.



동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

완성 코드

시작하기 버튼을 클릭했을 때

계속 반복하기

왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기

만일 왼쪽 접촉 센서 값 = 1 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기

0.5 초 기다리기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

0.5 초 기다리기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

만일 오른쪽 접촉 센서 값 = 1 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 400 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기

0.5 초 기다리기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 400 (으)로 정하기

0.5 초 기다리기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

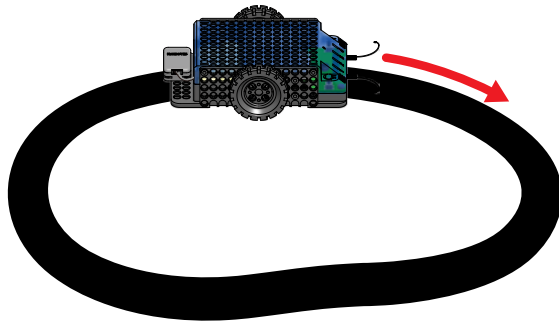
오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

13. 적외선센서 모드(라인트레이서)

< 학습목표 >

- 자동차로봇이 검은 선을 따라 이동하게 해 봅시다.

적외선센서 모드로 설정하고, 라인트레이싱 맵 위에서 동작시켜 봅시다. 자동차로봇이 검은 선을 따라 이동합니다. 다음의 블록들로 동작센서 모드 프로그램을 완성해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 참 이라면

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향

왼쪽 적외선 센서 값

오른쪽 적외선 센서 값

10 < 10

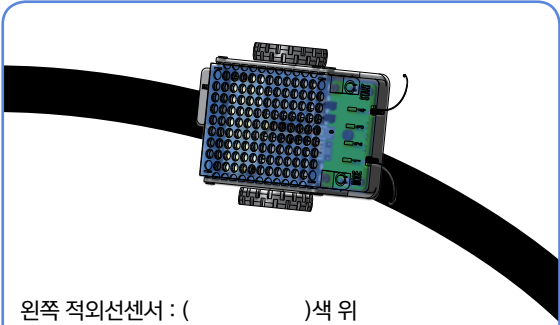
10 > 10

참 그리고 참

생각하기

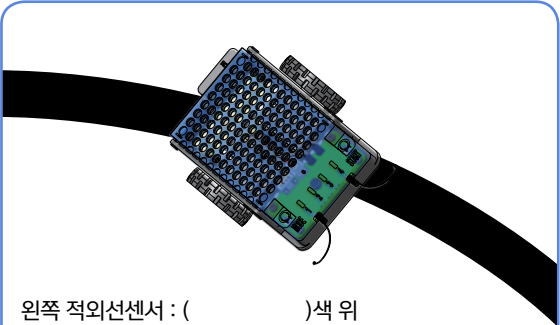
어떻게 하면 검은 선을 따라 이동하게 할 수 있을까요? 우선 검은 선을 벗어날지 어떻게 확인할 것인가 먼저 생각해 봅시다. 왼쪽이나 오른쪽 적외선센서가 검은 선에서 벗어날 경우 양쪽 적외선센서가 각각 어느 색에 위치하는지, 이때 적외선센서값이 어떻게 될지 생각해 봅시다.

왼쪽 적외선센서가 벗어난 경우



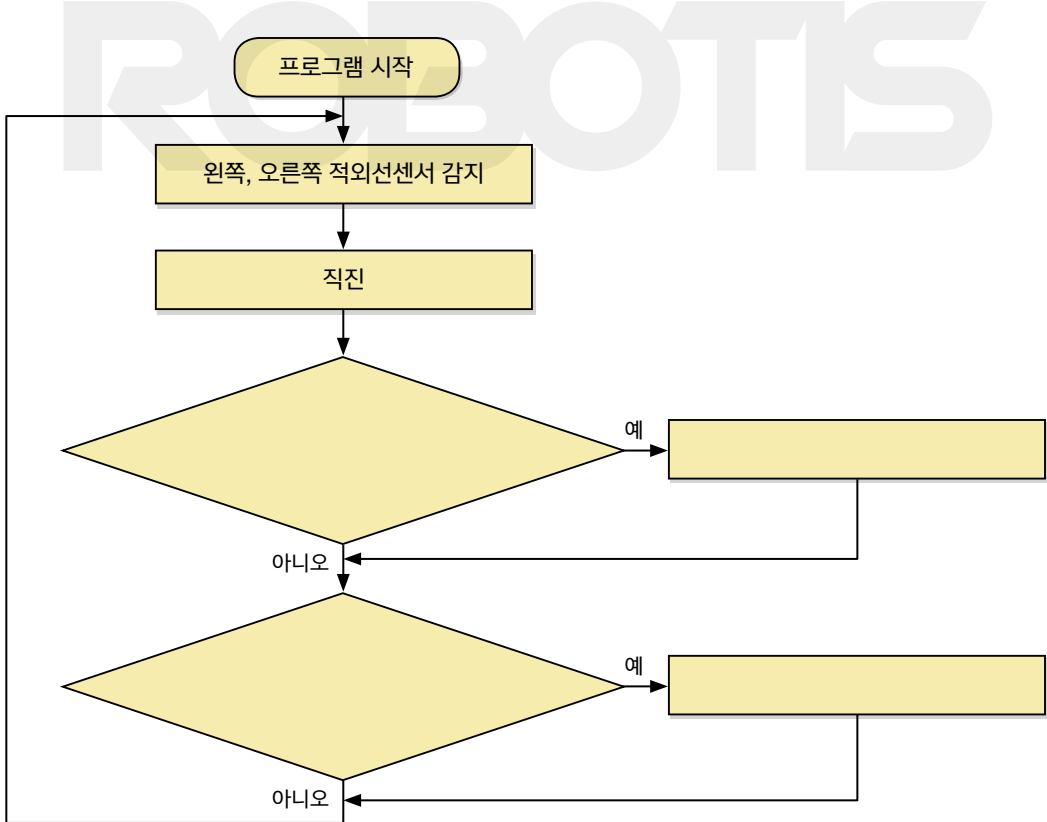
왼쪽 적외선센서 : ()색 위
 오른쪽 적외선센서 : ()색 위

오른쪽 적외선센서가 벗어난 경우



왼쪽 적외선센서 : ()색 위
 오른쪽 적외선센서 : ()색 위

검은 선을 벗어났을 때 어떻게 동작해야 할지 생각하여 순서도를 완성해 봅시다.



🗨️ 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

🎮 동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다. (검은색 절연테이프를 바닥에 붙여 원하는 대로 트랙을 만들 수 있습니다.)

🔧 완성 코드

The code starts with a 'When the green flag is clicked' event block, followed by a 'Say Hello for 2 seconds' speech bubble block. The main logic is contained within a 'Repeat (4) times' loop. Inside the loop, there are two 'If the green flag is clicked' conditional blocks. The first conditional block has two 'If the green flag is clicked' sub-conditions. The first sub-condition is 'If the green flag is clicked' (true), which leads to a 'Set motor speed to 200' block, a 'Set motor direction to Reverse' block, and a 'Set motor power to 200' block. The second sub-condition is 'If the green flag is clicked' (false), which leads to a 'Set motor speed to 200' block, a 'Set motor direction to Forward' block, and a 'Set motor power to 200' block. The second conditional block has two 'If the green flag is clicked' sub-conditions. The first sub-condition is 'If the green flag is clicked' (true), which leads to a 'Set motor speed to 270' block, a 'Set motor direction to Reverse' block, and a 'Set motor power to 270' block. The second sub-condition is 'If the green flag is clicked' (false), which leads to a 'Set motor speed to 270' block, a 'Set motor direction to Forward' block, and a 'Set motor power to 270' block.

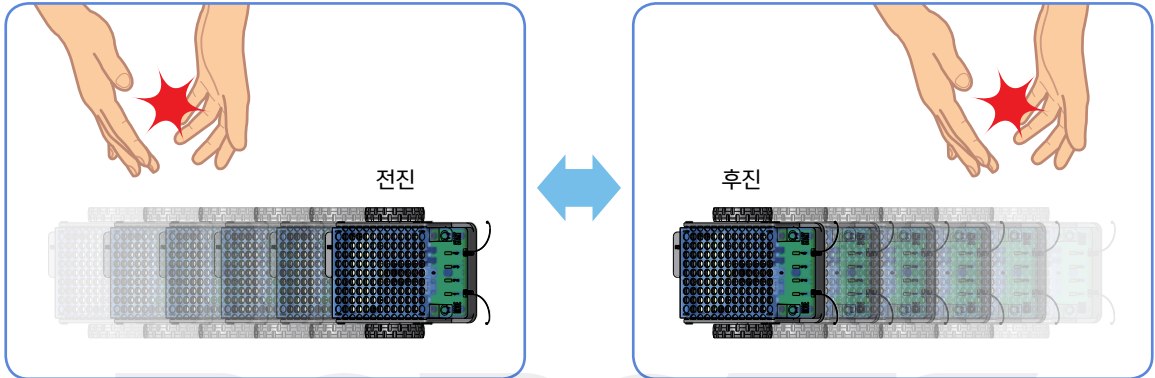
```
when green flag clicked
  say Hello for 2 seconds
  repeat (4) times
    if the green flag is clicked
      if the green flag is clicked
        set motor speed to 200
        set motor direction to Reverse
        set motor power to 200
      if the green flag is clicked
        set motor speed to 200
        set motor direction to Forward
        set motor power to 200
    if the green flag is clicked
      if the green flag is clicked
        set motor speed to 270
        set motor direction to Reverse
        set motor power to 270
      if the green flag is clicked
        set motor speed to 270
        set motor direction to Forward
        set motor power to 270
```

14. 같은 입력으로 두 가지 동작 명령하기

< 학습목표 >

- 같은 입력을 받을 때마다 두 가지 동작을 번갈아 하게 할 수 있다.

지금까지는 자동차로봇이 각 조건에 따라 다른 동작을 했습니다. 이번에는 같은 입력을 받을 때, 즉 소리가 날 때마다 두 가지 동작을 번갈아 하게 해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 참 이라면

참 이(가) 될 때까지 기다리기

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향

10 > 10

10 = 10

참 그리고 참

상태 를 0 로 정하기 ?

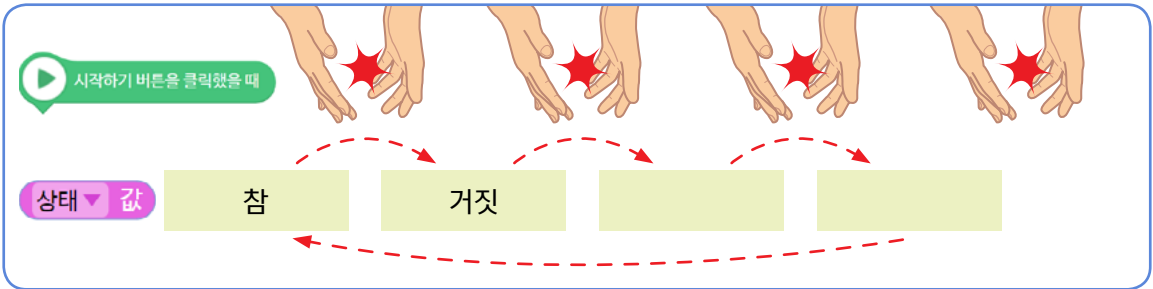
실시간 소리 감지 횟수 값

- 왼쪽 접촉 센서
- 오른쪽 접촉 센서
- 선택 버튼 상태
- 최종 소리 감지 횟수
- ✓ 실시간 소리 감지 횟수
- 왼쪽 적외선 센서
- 오른쪽 적외선 센서
- 왼쪽 적외선 센서 캘리브레이션 값
- 오른쪽 적외선 센서 캘리브레이션 값

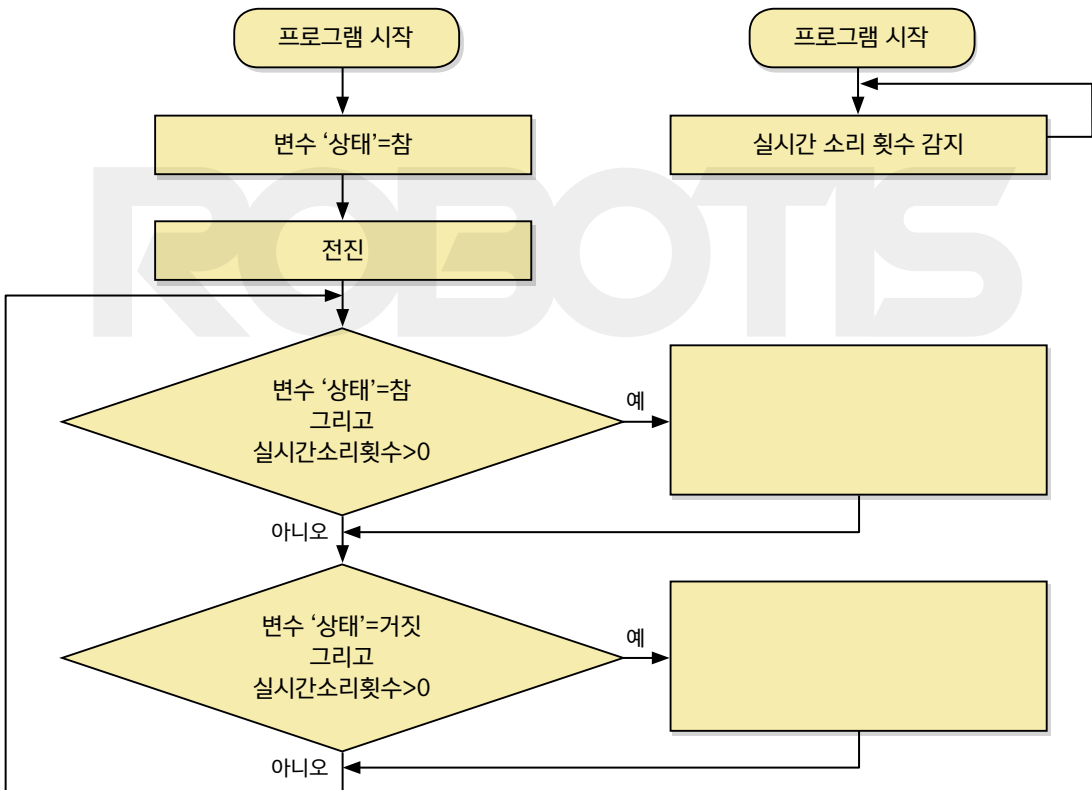


생각하기

같은 입력(박수)을 두 가지로 구분하려면 변수를 만들어 박수 소리가 감지될 때마다 두 값으로 번갈아 바꾸면 될 것입니다. 박수 소리가 감지될 때마다 변수를 어떻게 바꾸면 좋을지 생각하여 빈칸을 채워 봅시다.



박수소리가 감지될 때마다 전진, 후진을 번갈아 하려면 어떤 순서로 동작해야 할지 생각하고, 순서도를 완성해 봅시다.



생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.



동작 확인하기

하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

완성 코드

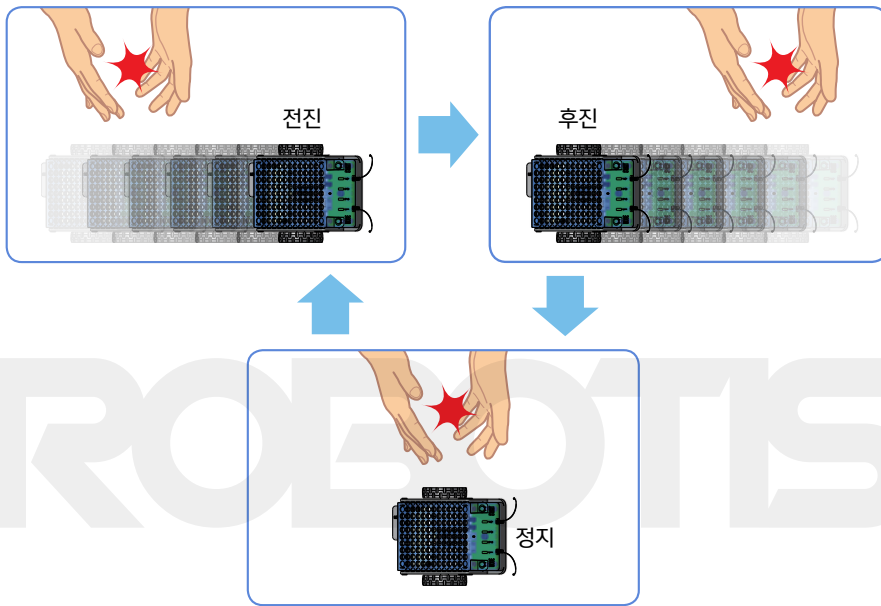


15. 같은 입력으로 여러 동작 명령하기

< 학습목표 >

- 같은 입력을 받을 때마다 여러 명령이 순차적으로 실행되게 할 수 있다.

지난 시간에는 박수를 칠 때마다 두 동작을 번갈아 명령해 보았습니다. 그렇다면 같은 입력으로 여러 동작을 명령할 수도 있을까요? 이번 시간에는 다음과 같이 박수를 칠 때마다 미리 정해진 명령이 순차적으로 실행되게 해 봅시다.



사용할 블록

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 참 이라면

참 이(가) 될 때까지 기다리기

왼쪽 감속모터 속도를 시계방향, 출력값을 500 (으)로 정하기

✓ 왼쪽
오른쪽

✓ 시계방향
반시계방향

왼쪽 적외선 센서 값

오른쪽 적외선 센서 값

10 > 10

10 = 10

참 그리고 참

상태 를 0 로 정하기 ?

실시간 소리 감지 횟수 값

- 왼쪽 접촉 센서
- 오른쪽 접촉 센서
- 선택 버튼 상태
- 최종 소리 감지 횟수
- ✓ 실시간 소리 감지 횟수
- 왼쪽 적외선 센서
- 오른쪽 적외선 센서
- 왼쪽 적외선 센서 캘리브레이션 값
- 오른쪽 적외선 센서 캘리브레이션 값

생각하기

이번에도 변수를 활용합니다. 박수 소리가 감지될 때마다 변수를 순차적으로 바꾸면 여러 동작을 순차적으로 할 수 있습니다. 세 가지 동작을 순차적으로 하게 하려면 변수를 어떻게 바꾸어야 할지 생각하여 빈칸을 채워 봅시다.

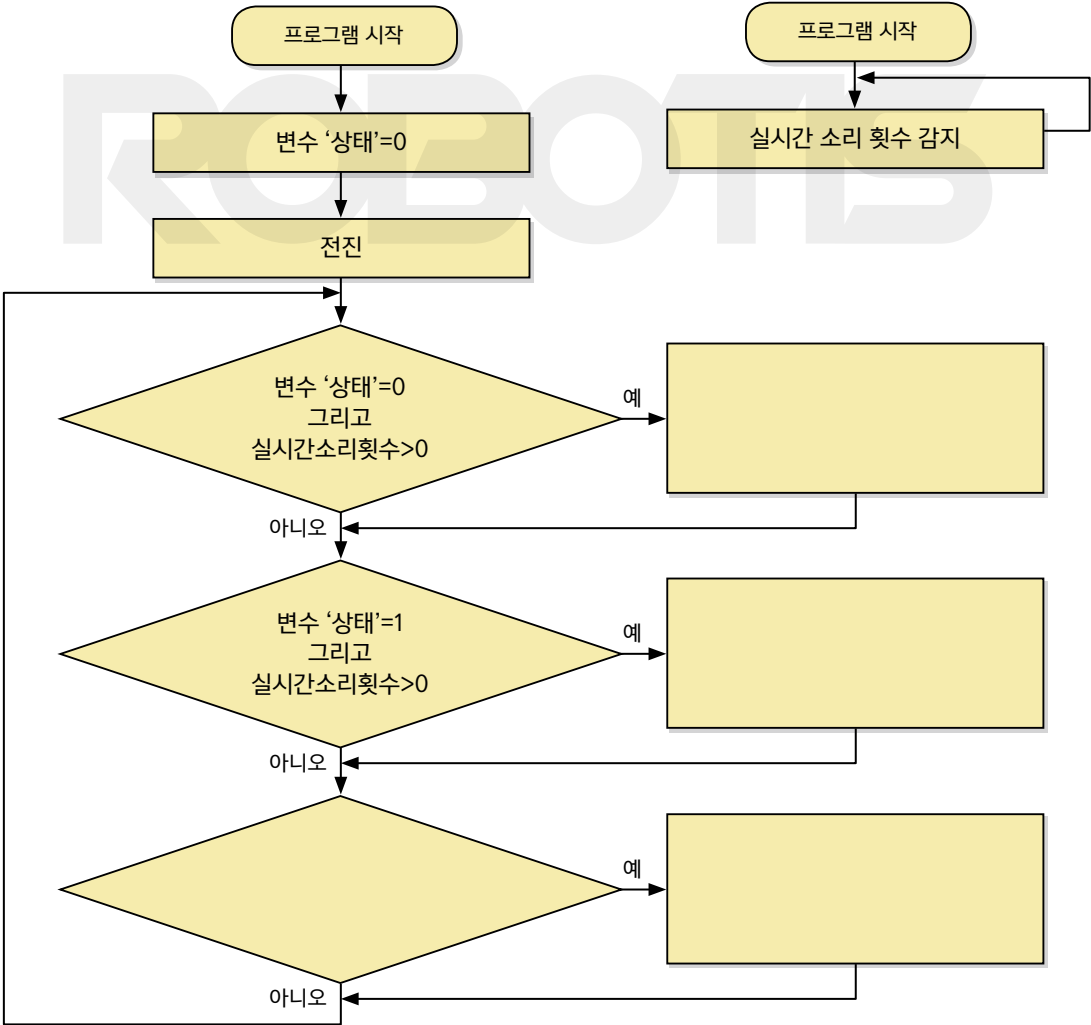
시작하기 버튼을 클릭했을 때

상태 ▼ 값

0

1

프로그램이 시작되면 전진하다가 박수소리가 감지될 때마다 후진, 정지, 전진을 순차적으로 실행하려면 어떤 순서로 동작해야 할지 생각하고, 순서도를 완성해 봅시다.



🗨️ 생각 적용하기

생각한대로 자동차로봇이 동작하도록 코딩해 봅시다.

▶ 동작 확인하기

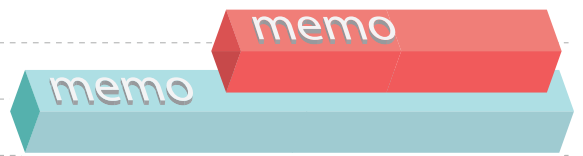
하드웨어와 연결된 상태에서 프로그램을 실행시켜 동작을 확인합니다. 예상대로 동작하지 않는다면 다시 생각하여 프로그램을 수정합니다.

📄 완성 코드

```

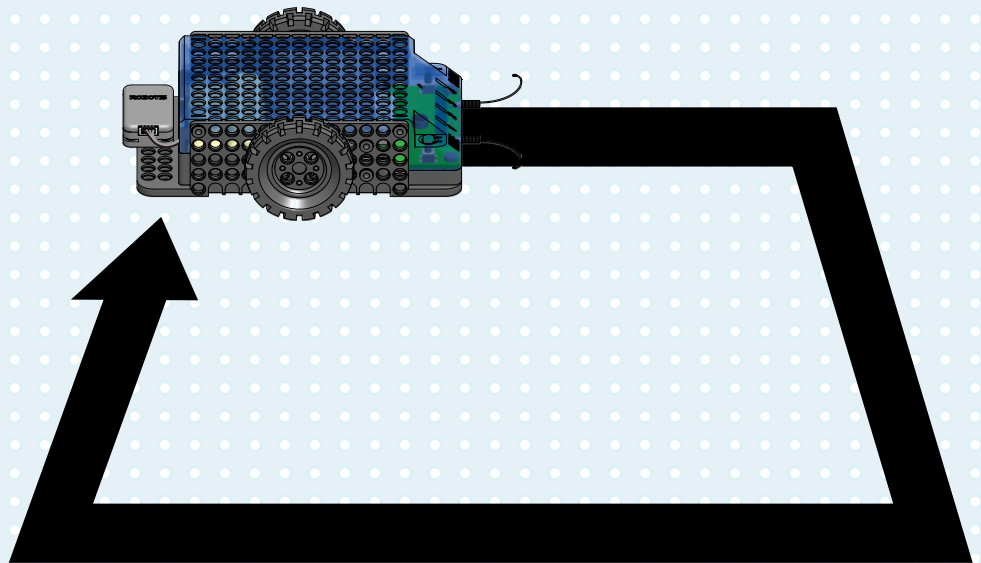
[시작하기 버튼을 클릭했을 때]
  상태 를 0 로 정하기
  왼쪽 감속모터 속도를 반시계방향 , 출력값을 300 (으)로 정하기
  오른쪽 감속모터 속도를 시계방향 , 출력값을 300 (으)로 정하기
  계속 반복하기
    만일 상태 값 = 0 그리고 실시간 소리 감지 횟수 값 > 0 이라면
      상태 를 1 로 정하기
      왼쪽 감속모터 속도를 시계방향 , 출력값을 300 (으)로 정하기
      오른쪽 감속모터 속도를 반시계방향 , 출력값을 300 (으)로 정하기
      실시간 소리 감지 횟수 값 = 0 이(가) 될 때까지 기다리기
    만일 상태 값 = 1 그리고 실시간 소리 감지 횟수 값 > 0 이라면
      상태 를 2 로 정하기
      왼쪽 감속모터 속도를 시계방향 , 출력값을 0 (으)로 정하기
      오른쪽 감속모터 속도를 반시계방향 , 출력값을 0 (으)로 정하기
      실시간 소리 감지 횟수 값 = 0 이(가) 될 때까지 기다리기
    만일 상태 값 = 거짓 그리고 실시간 소리 감지 횟수 값 > 0 이라면
      상태 를 참 로 정하기
      왼쪽 감속모터 속도를 반시계방향 , 출력값을 300 (으)로 정하기
      오른쪽 감속모터 속도를 시계방향 , 출력값을 300 (으)로 정하기
      실시간 소리 감지 횟수 값 = 0 이(가) 될 때까지 기다리기

```



ROBOTIS 로보티즈
자동차로봇 [엔트리판]

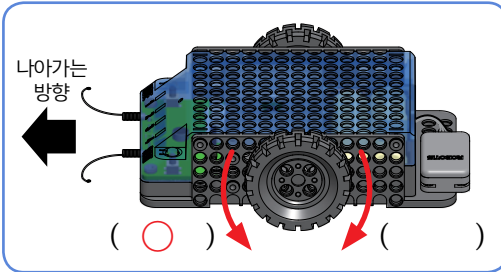
예시 답안



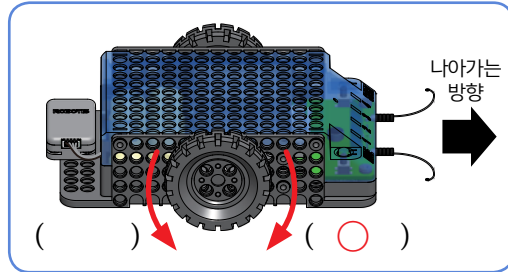
1. 전진

p13 (1) 전진

왼쪽 바퀴



오른쪽 바퀴



p17 (2) 후진

프로그램 시작

왼쪽 바퀴 (**시계**) 방향으로 회전

오른쪽 바퀴 (**반시계**) 방향으로 회전

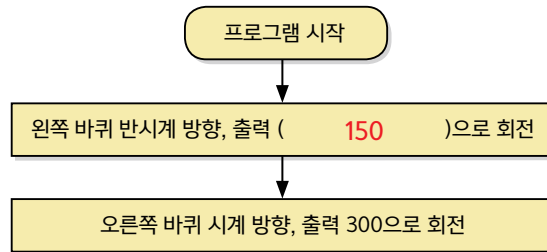
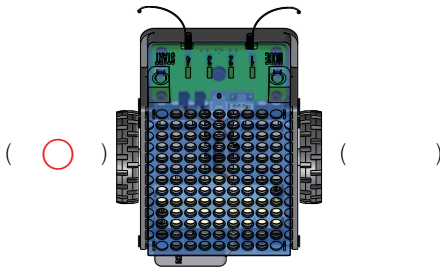
▶ 시작하기 버튼을 클릭했을 때

왼쪽 - 감속모터 속도를 **시계방향**, 출력값을 **300** (으)로 정하기

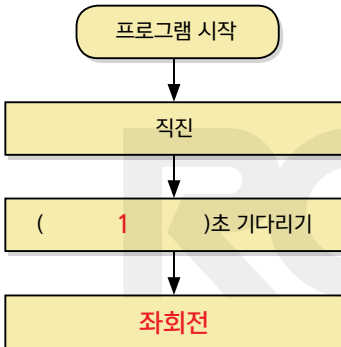
오른쪽 - 감속모터 속도를 **반시계방향**, 출력값을 **300** (으)로 정하기

2. 곡선코스

p19 (1) 좌회전



p20 (2) 직진 후 좌회전

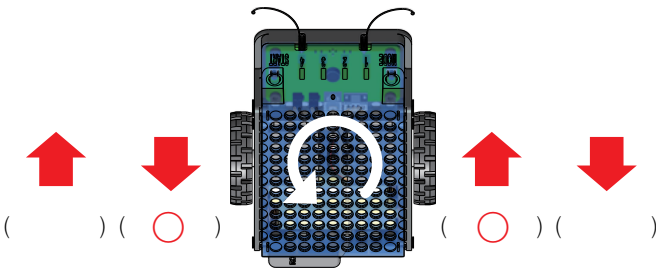


p21 (3) 곡선 코스



3. 굴절코스

p22 (1) 제자리 좌회전



프로그램 시작

왼쪽 바퀴 (**시계**) 방향, 출력 300으로 회전

오른쪽 바퀴 (**반시계**) 방향, 출력 300으로 회전

p24 (2) 굴절코스

프로그램 시작

직진

제자리 좌회전

직진

제자리 우회전

직진

▶ 시작하기 버튼을 클릭했을 때

왼쪽 ▶ 감속모터 속도를 반시계방향 ▶, 출력값을 300 (으)로 정하기

오른쪽 ▶ 감속모터 속도를 시계방향 ▶, 출력값을 300 (으)로 정하기

2 초 기다리기

왼쪽 ▶ 감속모터 속도를 시계방향 ▶, 출력값을 300 (으)로 정하기

오른쪽 ▶ 감속모터 속도를 시계방향 ▶, 출력값을 300 (으)로 정하기

0.5 초 기다리기

왼쪽 ▶ 감속모터 속도를 반시계방향 ▶, 출력값을 300 (으)로 정하기

오른쪽 ▶ 감속모터 속도를 시계방향 ▶, 출력값을 300 (으)로 정하기

1 초 기다리기

왼쪽 ▶ 감속모터 속도를 반시계방향 ▶, 출력값을 300 (으)로 정하기

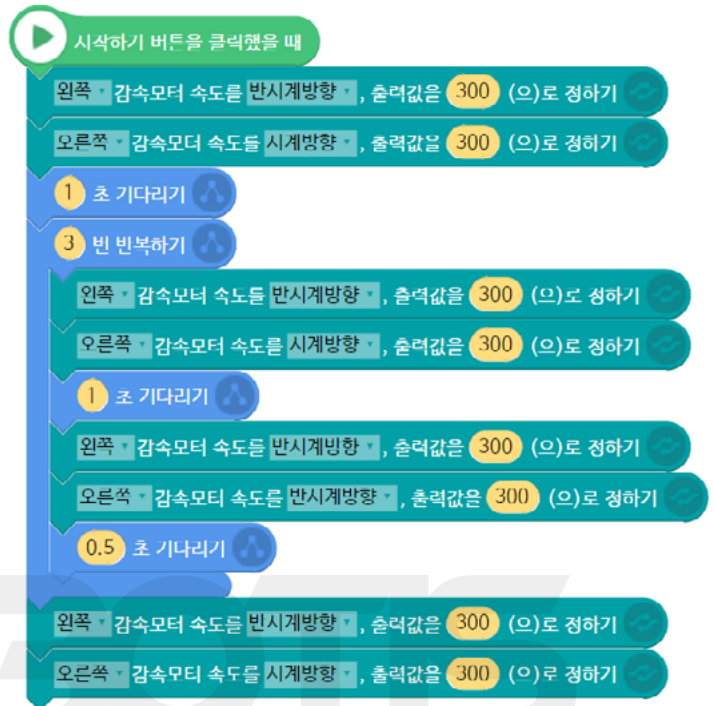
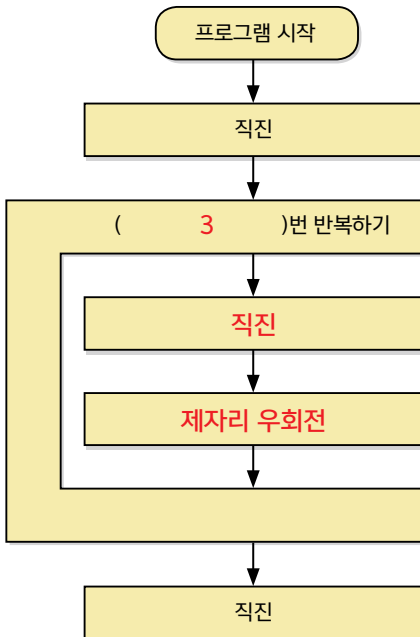
오른쪽 ▶ 감속모터 속도를 반시계방향 ▶, 출력값을 300 (으)로 정하기

0.5 초 기다리기

왼쪽 ▶ 감속모터 속도를 반시계방향 ▶, 출력값을 300 (으)로 정하기

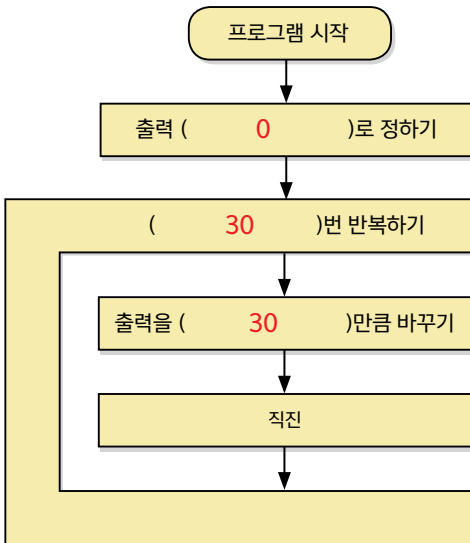
오른쪽 ▶ 감속모터 속도를 시계방향 ▶, 출력값을 300 (으)로 정하기

p25 (3) P턴

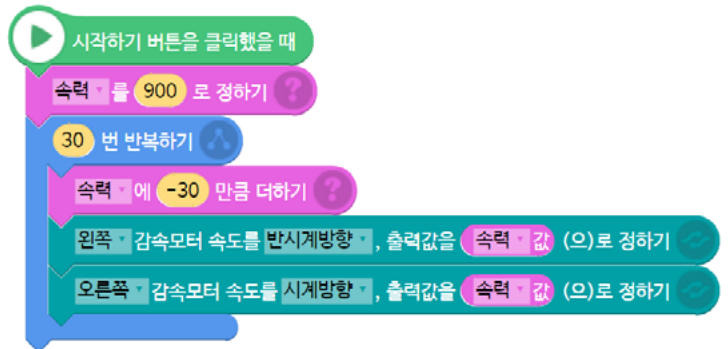
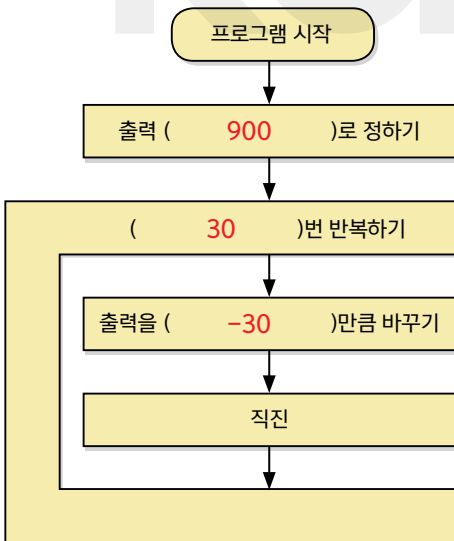


4. 변속코스

p27 (1) 가속

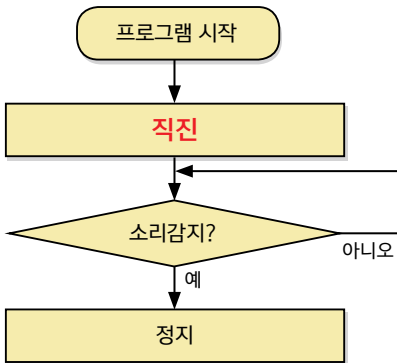


p28 (2) 감속

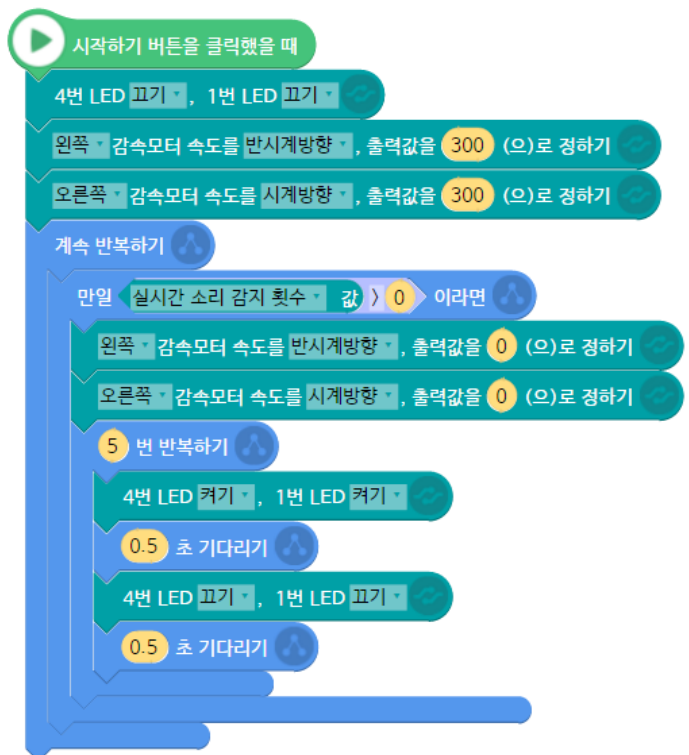
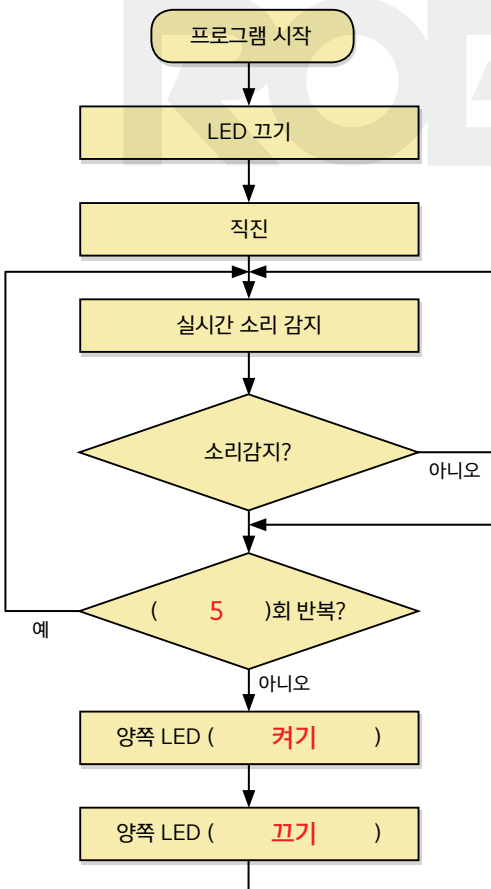


5. 돌발사고 시 급정지 및 출발

p30 (1) 돌발사고 시 급정지

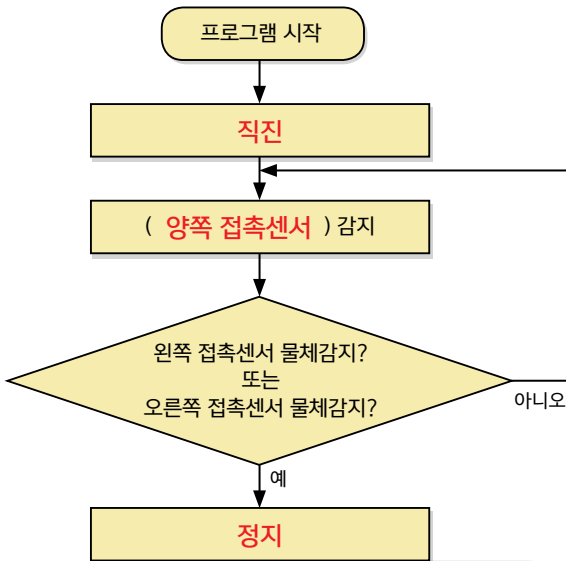


p31 (2) 돌발사고 시 급정지 후 비상등 켜기

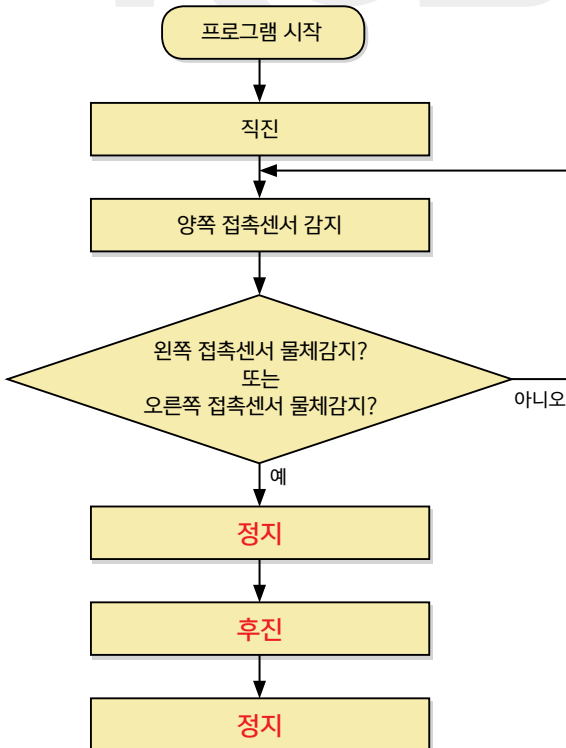


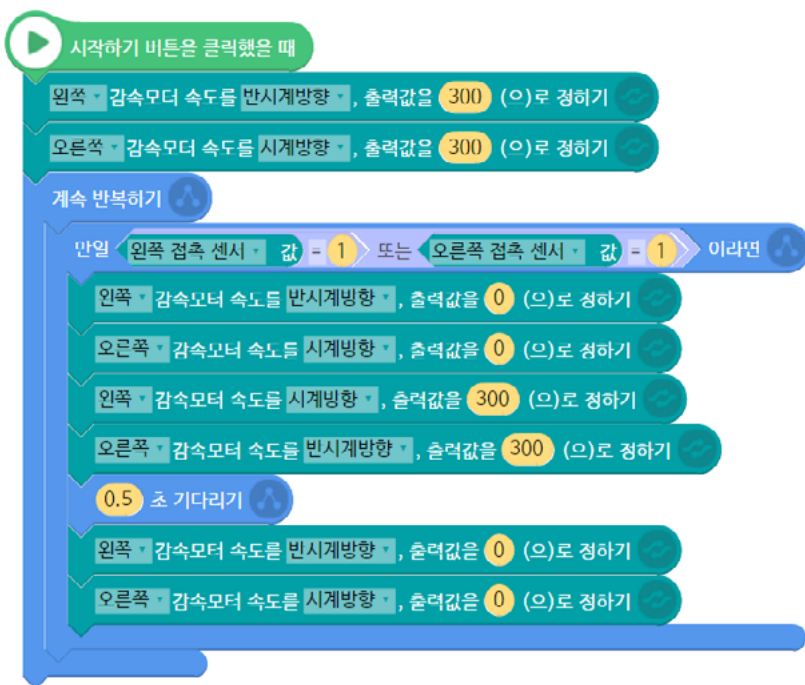
6. 철길건널목 통과

p33 (1) 철길건널목에서 정지



p34 (2) 철길건널목에서 후진 후 정지

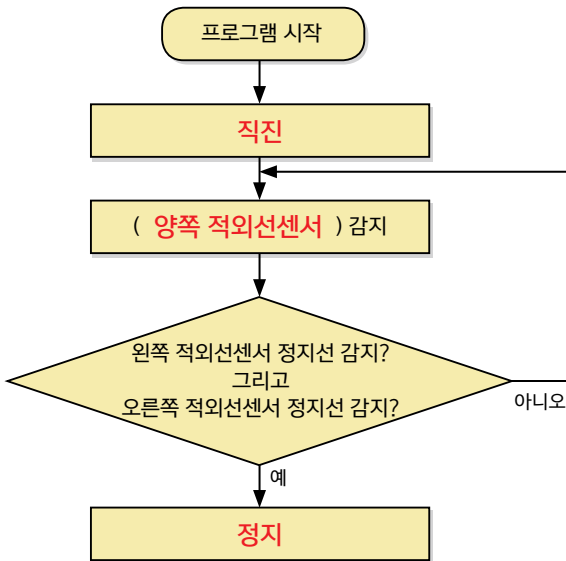




ROBOTS

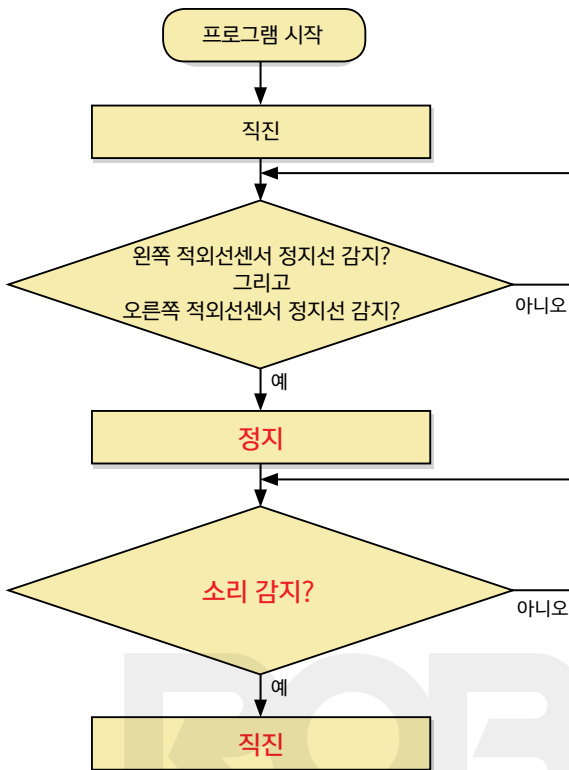
7. 횡단보도 일시정지

p36 (1) 정지선에서 정지



ROBOTIS

p37 (2) 출발선에서 대기하다가 출발음이 감지되면 직진



▶ 시작하기 버튼을 클릭했을 때

왼쪽 - 감속모터 속도를 반시계방향 , 출력값을 300 (으)로 정하기

오른쪽 - 감속모터 속도를 시계방향 , 출력값을 300 (으)로 정하기

계속 반복하기

만일 왼쪽 적외선 센서 값 < 300 그리고 오른쪽 적외선 센서 값 < 300 이라면

왼쪽 - 감속모터 속도를 반시계방향 , 출력값을 0 (으)로 정하기

오른쪽 - 감속모터 속도를 시계방향 , 출력값을 0 (으)로 정하기

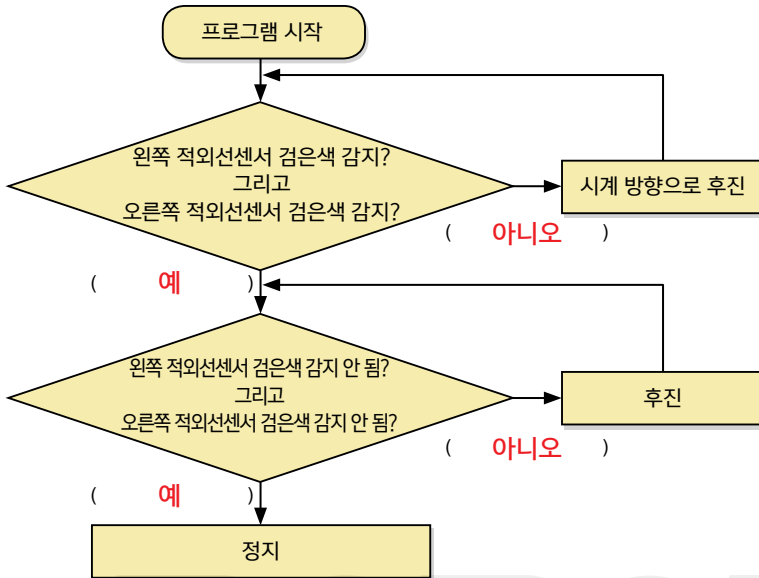
만일 실시간 소리 감지 횟수 값 > 0 이라면

왼쪽 - 감속모터 속도를 반시계방향 , 출력값을 300 (으)로 정하기

오른쪽 - 감속모터 속도를 시계방향 , 출력값을 300 (으)로 정하기

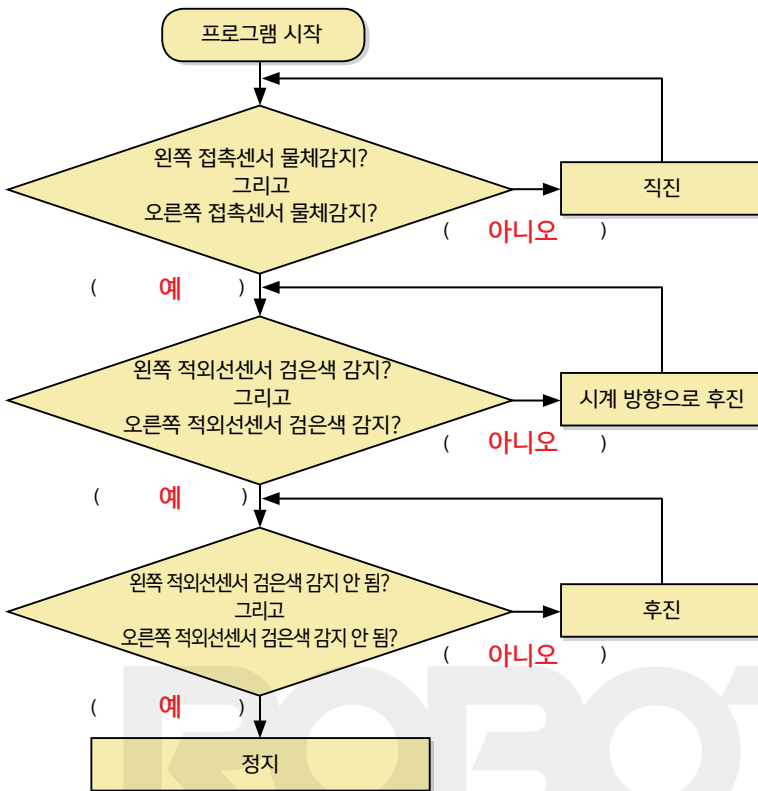
8. 후진주차코스

p39 (1) 후진주차



ROBOTIS

p41 (2) 막다른 골목에서 방향 전환



시작하기 버튼을 클릭했을 때

왼쪽 접촉 센서 값 = 1 그리고 오른쪽 접촉 센서 값 = 1 이 될 때까지 반복하기

왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

왼쪽 적외선 센서 값 < 200 그리고 오른쪽 적외선 센서 값 < 200 이 될 때까지 반복하기

왼쪽 감속모터 속도를 시계방향, 출력값을 150 (으)로 정하기

오른쪽 감속모터 속도를 반시계방향, 출력값을 250 (으)로 정하기

왼쪽 적외선 센서 값 > 200 그리고 오른쪽 적외선 센서 값 > 200 이 될 때까지 반복하기

왼쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

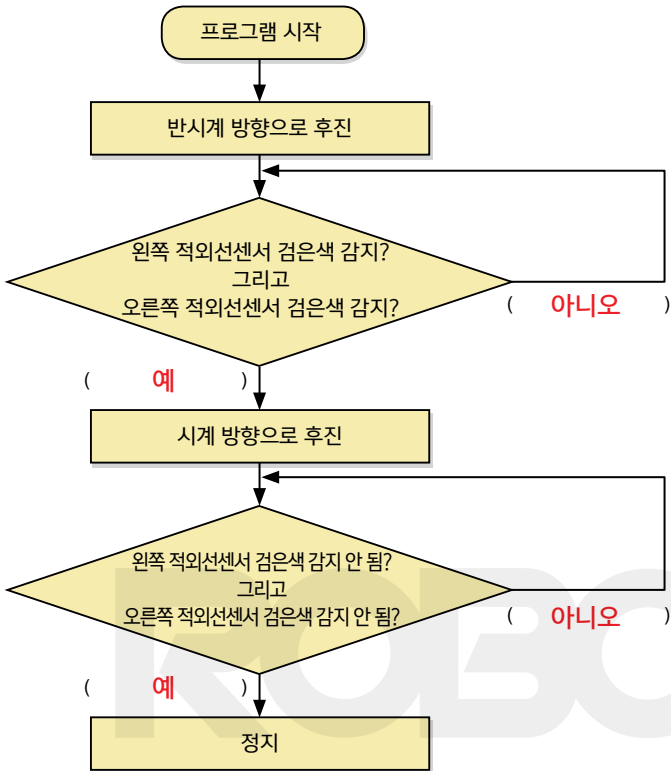
오른쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

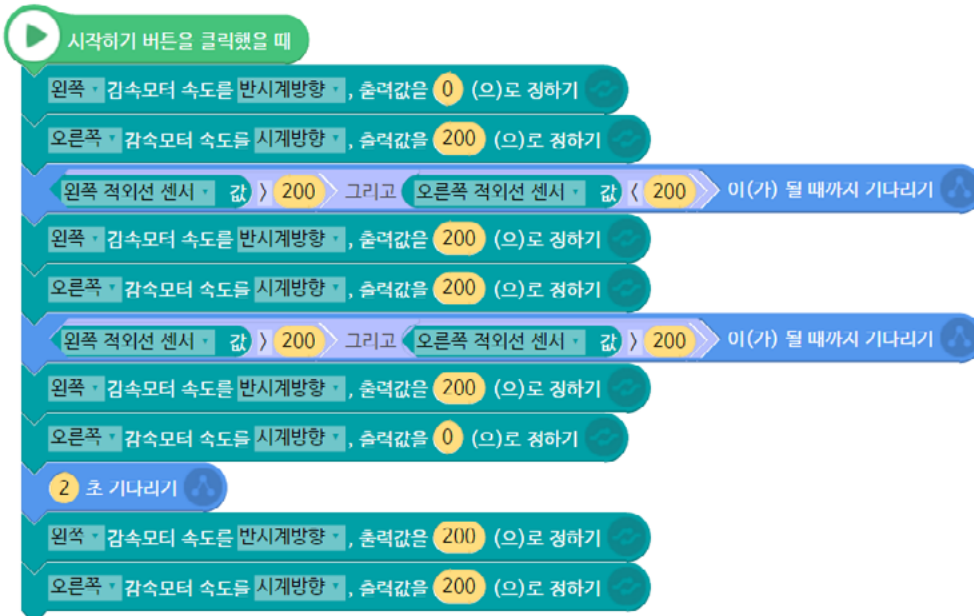
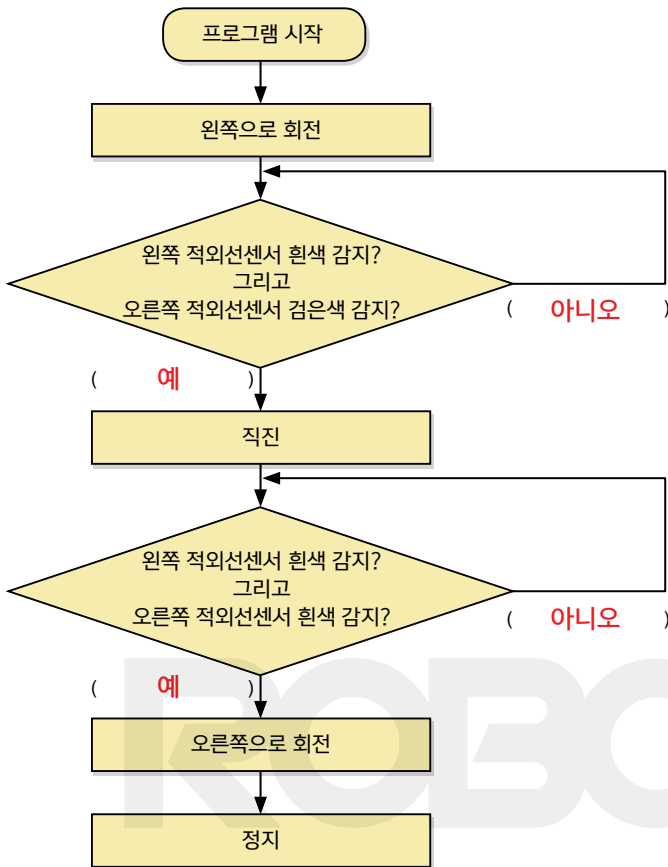
오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

9. 평행주차코스

p43 (1) 평행 주차

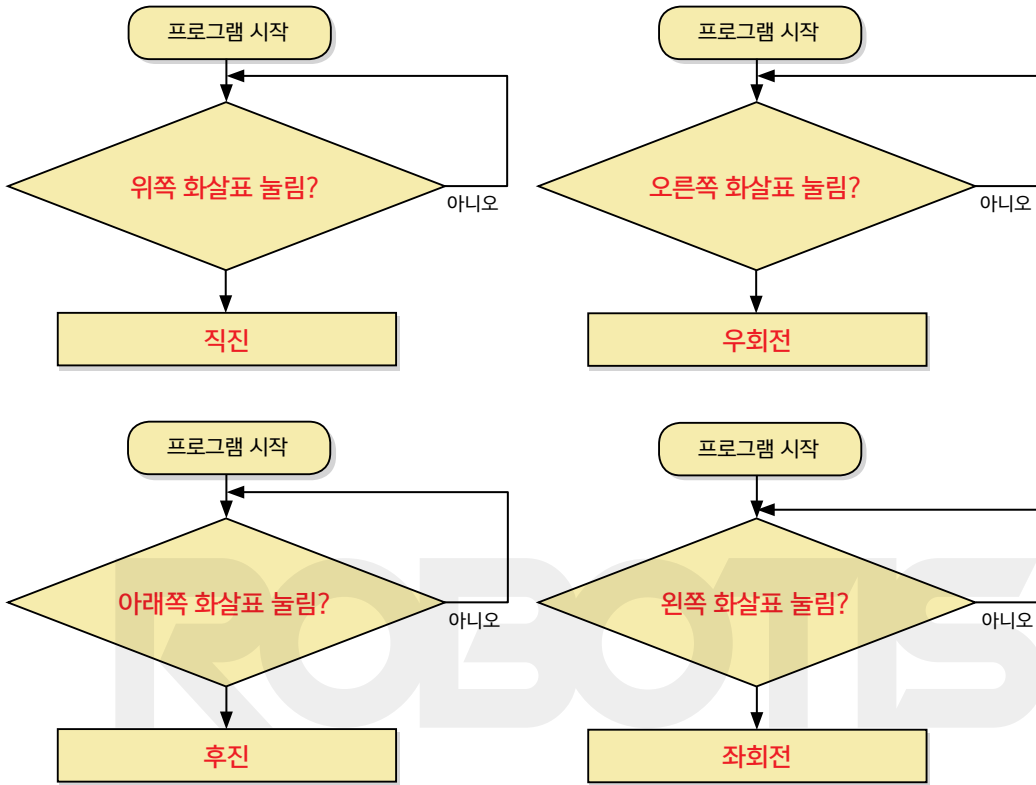


p45 (2) 평행주차구역에서 빠져나오기

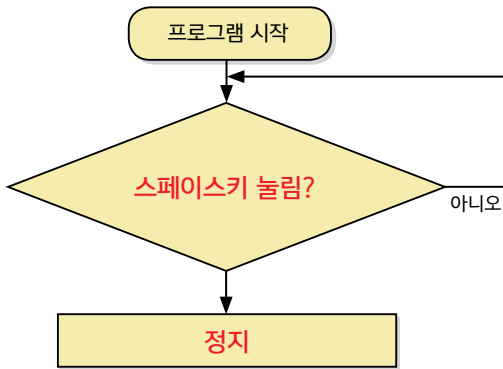


10. 키보드 조종

p47 (1) 화살표키로 전후좌우로 이동



p48 (2) 스페이스키로 정지



스페이스 키를 눌렀을 때

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

p48 (3) 다른 블록을 사용하여 같은 명령하기

시작하기 버튼을 클릭했을 때

계속 반복하기

만일 위쪽 화살표 키가 눌러져 있는가? 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

만일 아래쪽 화살표 키가 눌러져 있는가? 이라면

왼쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

만일 왼쪽 화살표 키가 눌러져 있는가? 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 150 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기

만일 오른쪽 화살표 키가 눌러져 있는가? 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 150 (으)로 정하기

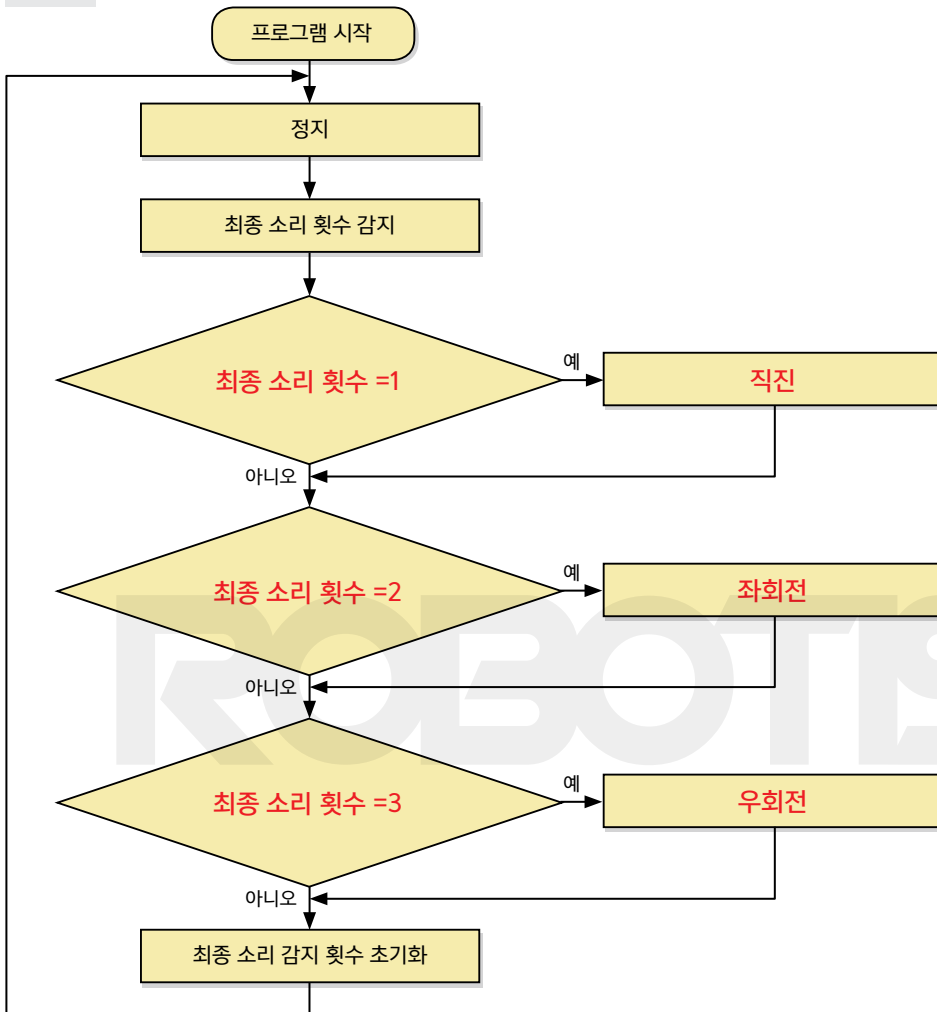
만일 스페이스 키가 눌러져 있는가? 이라면

왼쪽 감속모터 속도를 반시계방향, 출력값을 0 (으)로 정하기

오른쪽 감속모터 속도를 시계방향, 출력값을 0 (으)로 정하기

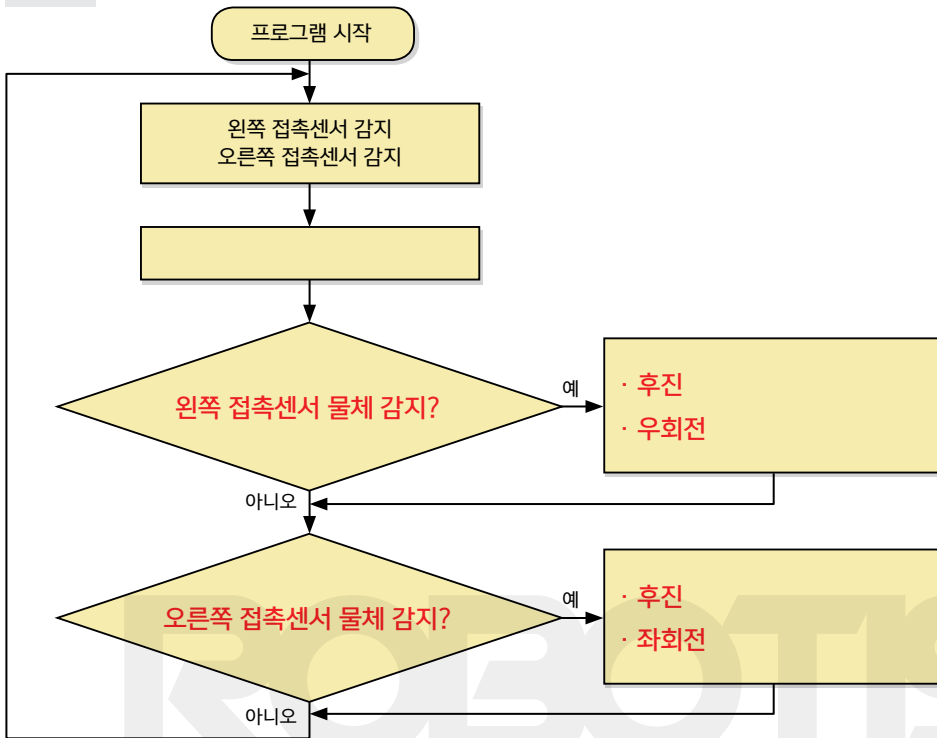
11. 소리센서 모드

p52



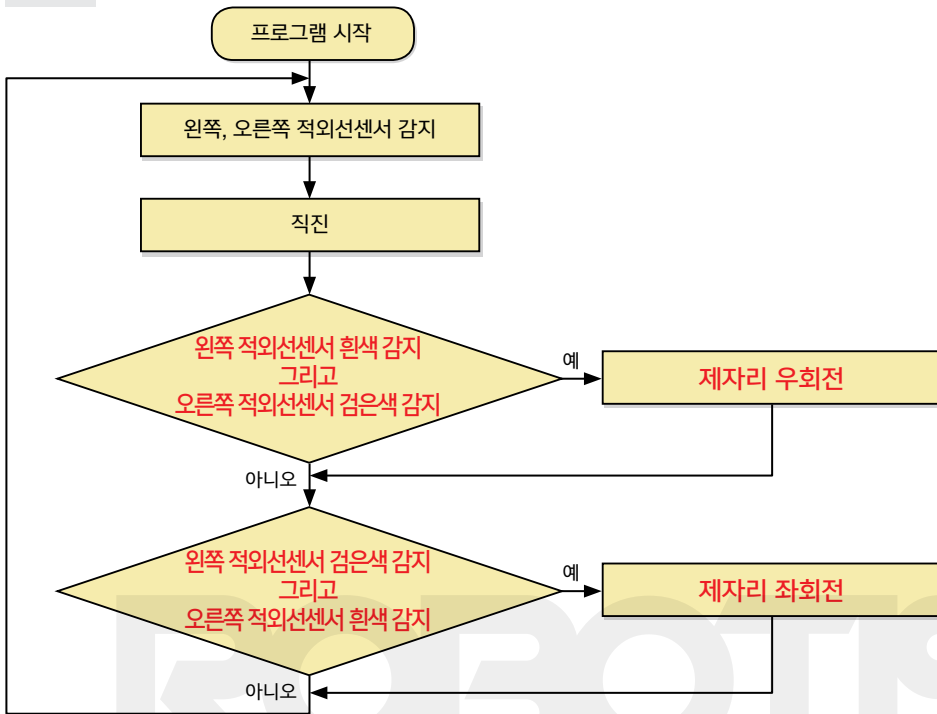
12. 접촉센서 모드

p55



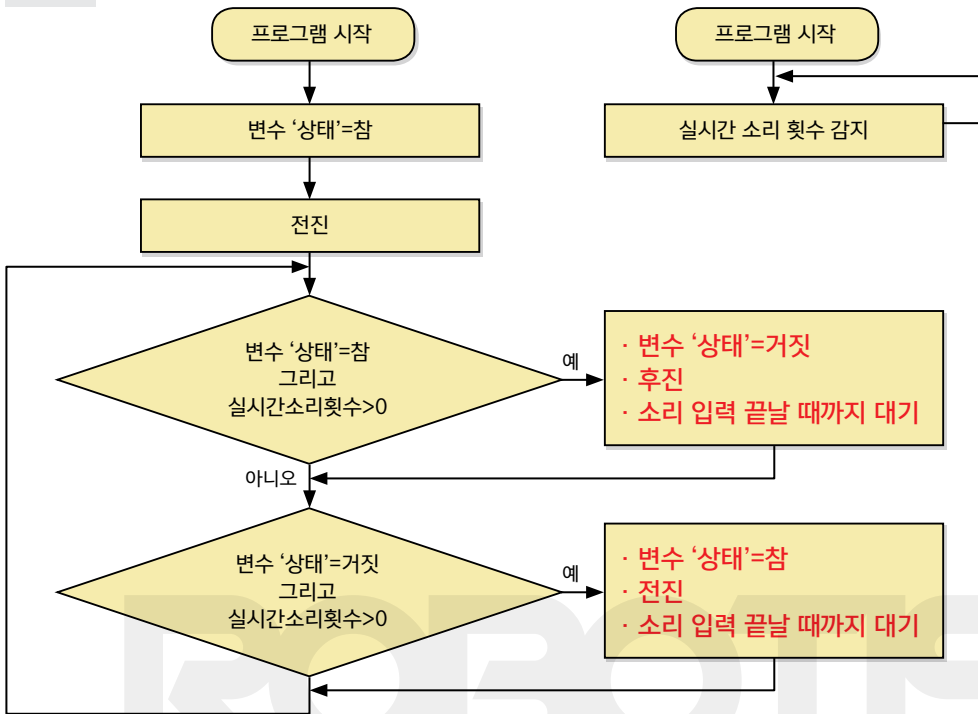
13. 적외선센서 모드(라인트레이서)

p58



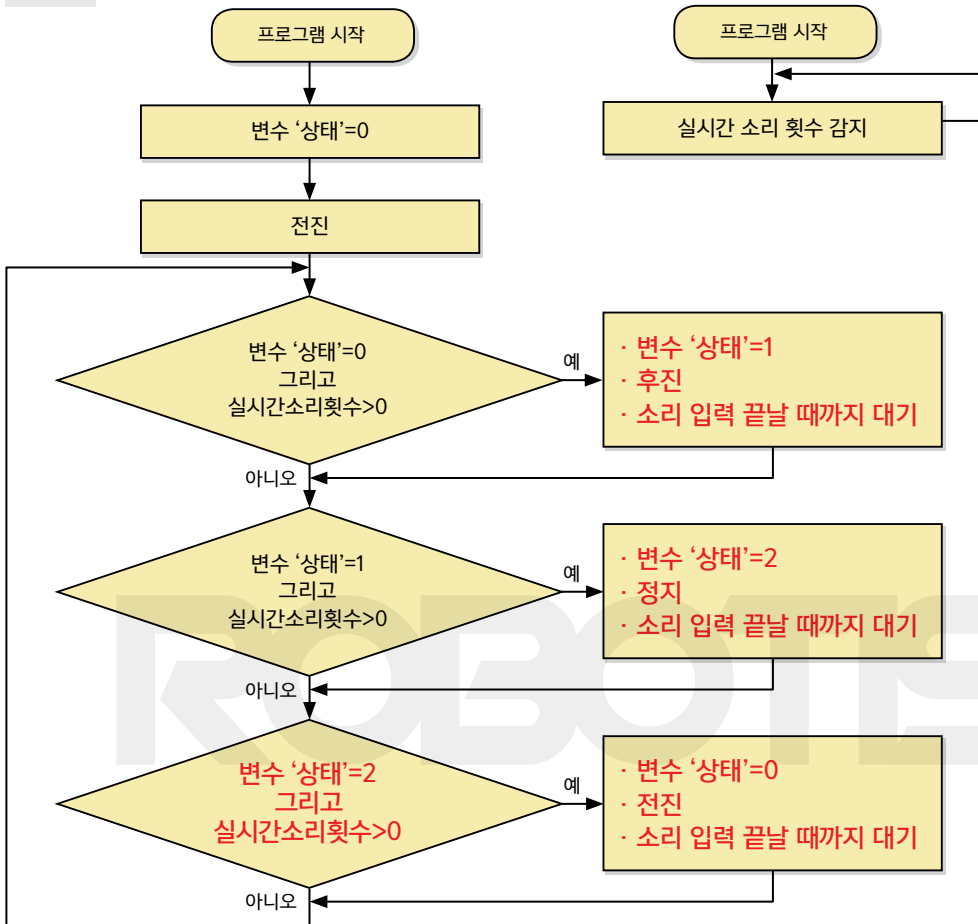
14. 같은 입력으로 두 가지 동작 명령하기

p61



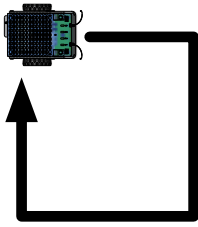
15. 같은 입력으로 여러 동작 명령하기

p64



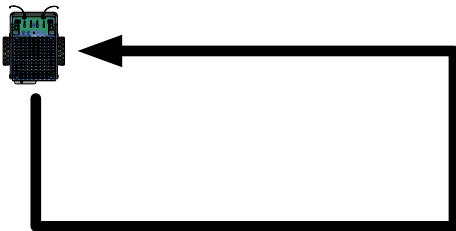
16. 자동차로봇으로 도형 그리기

p72



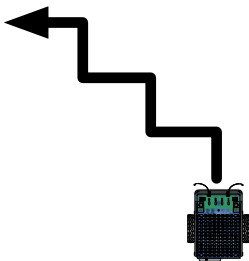
시작하기 버튼을 클릭했을 때

- 4 번 반복하기
 - 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
 - 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 1 초 기다리기
- 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
- 오른쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기



시작하기 버튼을 클릭했을 때

- 2 번 반복하기
 - 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
 - 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 1 초 기다리기
- 왼쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 0.5 초 기다리기
- 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 2 초 기다리기



시작하기 버튼을 클릭했을 때

- 3 번 반복하기
 - 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
 - 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 1 초 기다리기
- 왼쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 0.5 초 기다리기
- 왼쪽 감속모터 속도를 반시계방향, 출력값을 300 (으)로 정하기
- 오른쪽 감속모터 속도를 시계방향, 출력값을 300 (으)로 정하기
- 1 초 기다리기

ROBOTIS 로보티즈

자동차로봇



[엔트리편]

AS문의 및 제품문의 안내번호

Tel. +82-70-8671-2600 E-mail. korea@robotis.com

ALL Rights Reserved. Copyright © by ROBOTIS Co., Ltd.

본 교재의 내용에 대한 저작권은 (주)로보티즈에 있으며 허락없이 복사하여 사용할 수 없습니다.