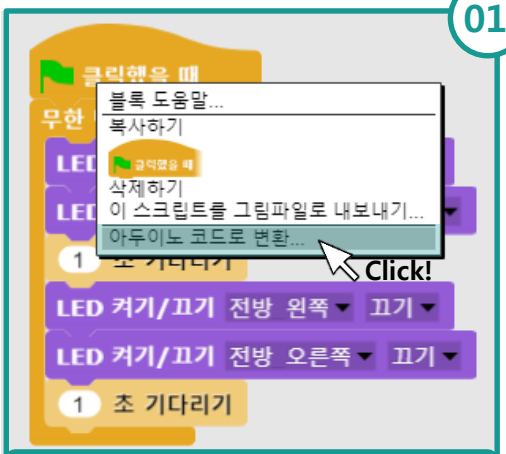
SG 모드 매뉴얼

SG MODE MANUAL

ROKIT BRICK for Zumi

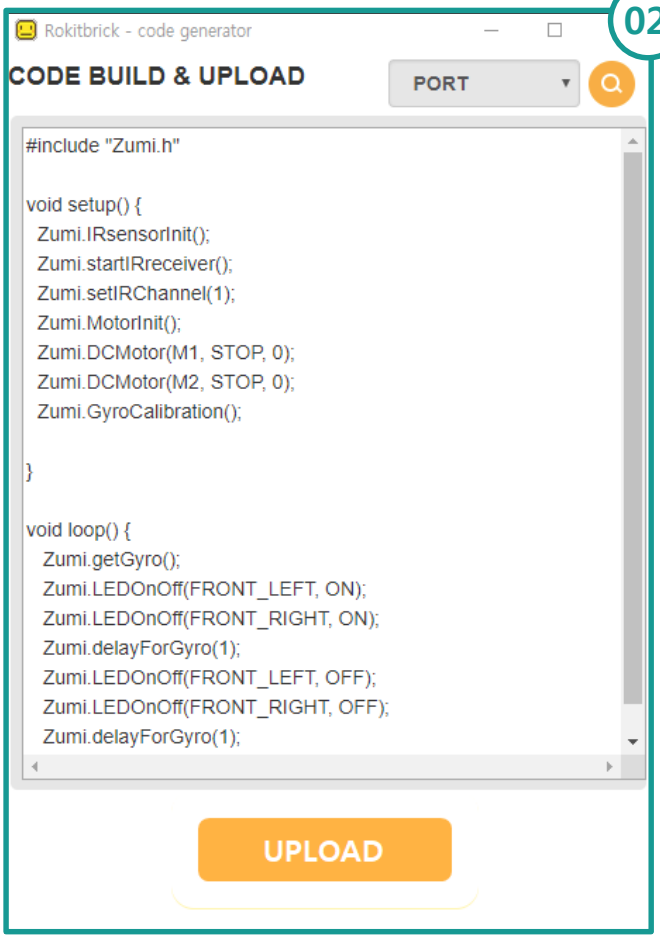
01 코드 제너레이터 창 띄우기 및 업로드

01

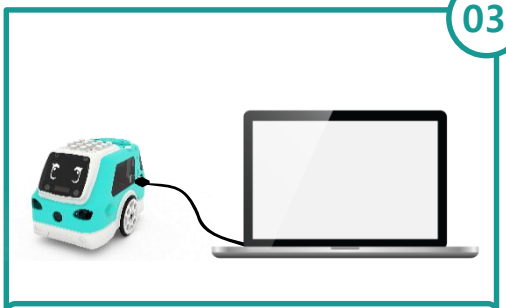


마우스 오른쪽 버튼 클릭 !

02

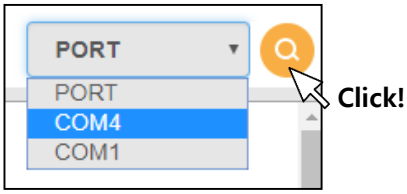


03



주미 연결

04

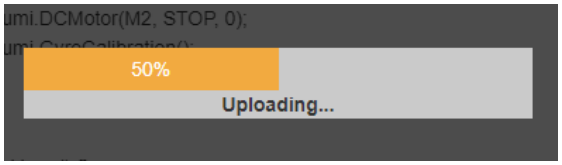


포트 검색 버튼 클릭 > 포트 선택

05

클릭하면 컴파일 / 업로드 실행

06



코드 제너레이션 모드 사용 팁

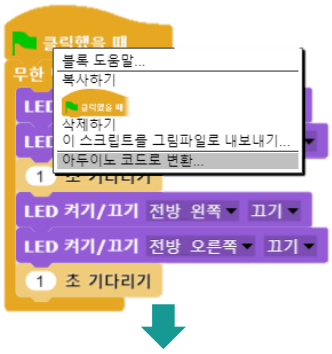
A

화면(screen) 카테고리의 블럭들을 사용하면 컴파일 시간이 느려집니다. 만약 화면 블록들이 굳이 필요하지 않는 경우에는 스크립트에서 블럭들을 삭제해 주시는 것이 좋습니다.

B

일부 블록에 표시된 로봇 아이콘은 이 블록을 사용하면 자이로 센서가 작용한다는 것을 의미합니다.
센서 값을 기반으로 로봇의 동작들을 자동으로 제어하면서 주미가 좀 더 정확하게 움직일 수 있도록 해줍니다.

C



변경 사항 반영

코드 제너레이터 창이 띄워져 있다고 해도, 스크립트의 변경 사항이 자동으로 반영되지는 않습니다. 변경 사항을 코드 제너레이션 창의 아두이노 코드에 반영하기 위해서는 창을 처음에 띄울 때처럼 '아두이노 코드로 변환'을 클릭해 주셔야 합니다.

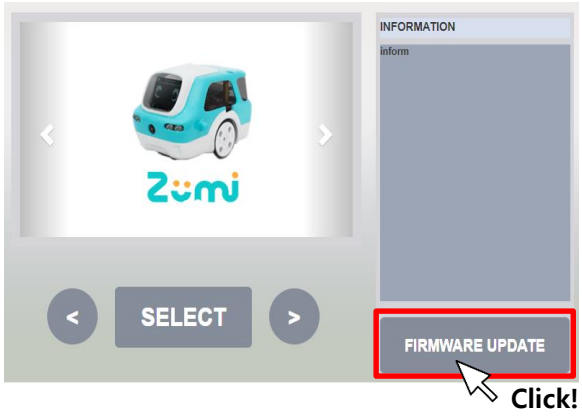
02 기본 펌웨어 복구 방법

코드 제너레이터 모드에서 아두이노 코드를 업로드하면 사용하던 주미의 펌웨어는 지워집니다.
따라서 다시 주미의 본래 기능을 사용하시려면 **펌웨어를 복구해 주세요.**

01

주미와 PC를 연결해 주세요

02



시작화면에서 펌웨어 업데이트 버튼 누르기

03

주미와 연결된 포트 검색 버튼 클릭 > 포트 선택 > 펌웨어 업로드 버튼 클릭!!

03 조종 카테코리 블록

A

DC모터 블록 :
주미의 DC모터를 독립해서 제어할 수 있습니다.
자이로 센서를 사용하지 않고 동작합니다.

DC모터 ▾ 방향 ▾ 속도 ▾

M1 M2	정회전 역회전 멈춤 풀림	0 20 40 60 80 / 100 130 160 180 210 250

예제 1 주미를 2초간 전진하다가 멈추기

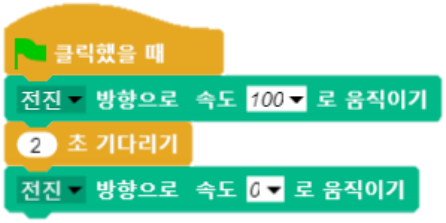
B

방향 제어 블록 :
DC모터를 따로 따로 제어하는 것이 아니라
2개의 DC모터를 동시에 제어해서 사용자가
원하는 방향으로 주미를 움직이게 합니다.
자이로 센서를 사용하지 않고 동작합니다. .

▾ 방향으로 속도 ▾ 로 움직이기

정회전 역회전 멈춤 풀림	0 20 40 60 80 / 100 130 160 180 210 250

예제 2 주미를 2초간 전진하다가 멈추기

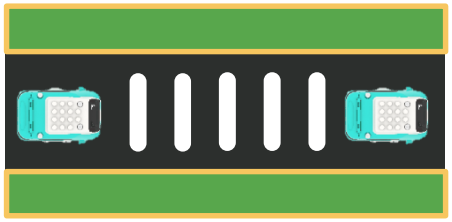


C

블록 지나가기 블록 :
검은색 바탕위에 2 ~ 3cm 이내 폭을 가진 하얀
블록들이 있을 때(사진참조) 이 블록을 세어가면서
주미를 직진할 수 있게 해주는 블록입니다.
블록들을 세면서 진행하기 때문에 블록을
단위로 원하는 거리만큼만 갈 수 있습니다.
바닥의 블록들을 셀 때에는 주미의 밑부분에
있는 2개의 센서로 감지합니다.

: 블록의 지나갈 개수 입력(직접 입력)

예제 3 주미가 바닥에 있는 하얀 블록 5개를 넘어간 후 정지하기



예제 4 주미가 바닥에 있는 하얀 블록 5개를 넘은 후 적당한 속도로 1초간 좌회전 하다가 멈추기

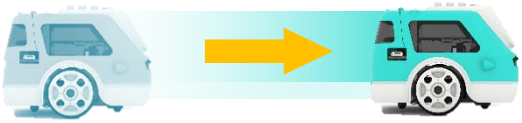
D

전진 블록 :
정해진 시간(초) 동안 정해진 속도로 전진하며 자이로 센서를 이용하여 최대한 기울어짐 없이 직진 합니다.

: 전진할 시간(초) 입력(직접 입력)

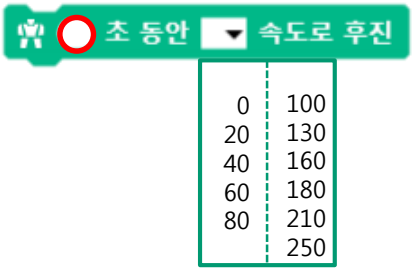
0	100
20	130
40	160
60	180
80	210
	250

예제 5 2초 동안 60의 속도로 전진 후 60속도로 1초간 우회전 한 후 정지하기



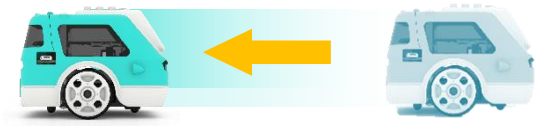
E

후진 블록 :
정해진 시간(초) 동안 정해진 속도로 후진하며 자이로 센서를 이용하여 최대한 기울어짐 없이 진행합니다.



: 후진할 시간(초) 입력(직접 입력)

예제 6 **2초 동안 60의 속도로 후진 후 잠시 후 다시 2초 동안 60의 속도로 전진하기**



F

회전 블록 :
정해진 방향으로 정해진 각도만큼 회전합니다. 자이로센서를 이용하여 최대한 정확한 각도로 회전합니다. 속도는 내장된 기본 값을 사용합니다.

: 회전할 각도(degree) 입력 (직접 입력)

예제 7 **주미가 10개의 블록을 지난 후 우회전 방향으로 90도 돌기**

알아두기

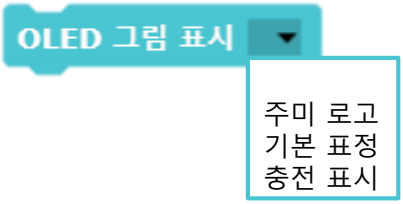
아래처럼 팝업 메뉴에서 원하는 값이 없을 때에는 팝업창에 직접 원하는 값을 입력할 수 있습니다.

※ 다만 하얀색의 팝업메뉴만 직접 입력이 가능하며 하얀색이 아닌 경우에는 직접 입력이 되지 않습니다.

04 화면(Screen) 카테고리 블록

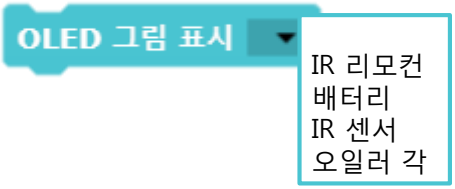
A

그림표시 블록 :
주미의 OLED 화면에 주미 펌웨어에 미리 저장되어 있는 이미지를 띄웁니다.
(메모리 문제로 3개의 이미지만 준비 되어 있습니다)



B

OLED 출력 블록 :
배터리 잔량과 같은 주미의 상태나 적외선 센서, 그리고 오일러 값 (자이로 센서 출력값)을 OLED 화면에 출력해 줍니다.
주미의 센서값등을 모니터링하여 값을 적용할 때 유용한 블록입니다.



예제 8 **주미의 IR 센서 값을 OLED로 출력합니다.**

무한 반복 블록 사용해서 출력값을 계속 읽어야 현재의 센서 값을 알아낼 수 있습니다.

예제 9

주미의 전원을 켜면 OLED 에 로고 화면이 나타나고 3초 후에 배터리 잔량이 표시 될 수 있도록 해 봅시다.

05 모양 / 소리 / 빛 카테고리 블록

A

원 모양으로 움직이기 (원 그리기) 블록 :
정해진 방향으로 정해진 각도만큼 원을 그리며 주미가 움직입니다.
속도는 지정하지 않으며 기본 속도로 주행 합니다.

우회전

좌회전

B

연주하기 블록 :
주미 안에 있는 버저를 사용하여 멜로디를 연주할 수 있는 블록입니다.

- **각 멜로디는 알파벳 표기로 아래 음계를 표시합니다.**

  C (도) **D** (레) **E** (미) **F** (파) **G**(솔) **A**(라) **B**(시)

- **S 는 반음을 올린 높은 소리로 으로서 각 음계의 샵(#) 을 의미합니다.**

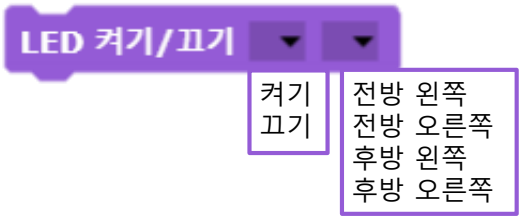
  - CS = 도#
  - DS – 레#

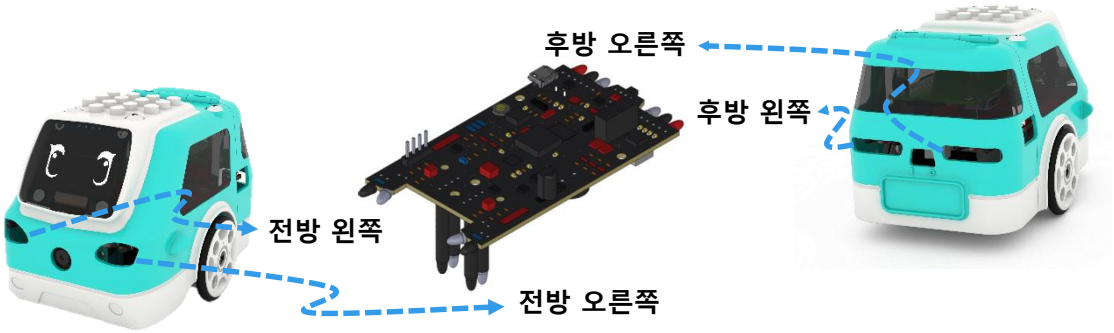
예제 10 음계를 연주할 수 있는 연주하기 블록을 사용하여 간단한 멜로디를 연주해 봅니다.

※ 음계 사이마다 ~초 기다리기 블록을 넣어서 음 사이의 간격을 구현합니다

C

LED 블록 :
주미의 LED를 켜고 끌수 있게 해주는 블록입니다.





06 센서 카테고리 블록

A

타이머 블록:
전원이 켜지고 주미의 프로그램이 시작된 이후부터 밀리세컨드(1/1000초) 단위로 흘러간 시간을 측정합니다.
전원을 껐다 켜거나 리셋이 되면 다시 0부터 시작합니다.

타이머

예제 11 **'~초 기다리기' 블록을 사용하지 않고 타이머를 시간을 계산하여 1초마다 주미의 버저에서 소리가 나게 합시다.**

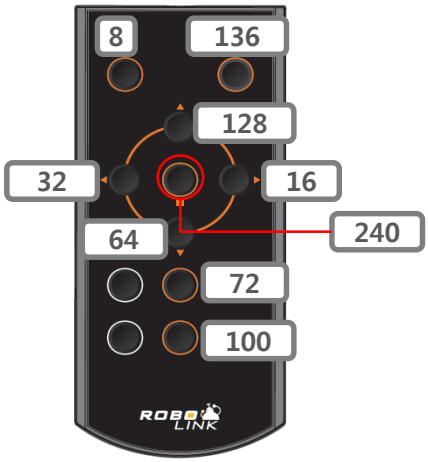
B

IR 리모컨 블록:
로보링크 IR리모컨 값을 수신받는 역할을 합니다. IR리모컨의 키가 눌려질 때 그 키에 해당하는 값을 받게 됩니다

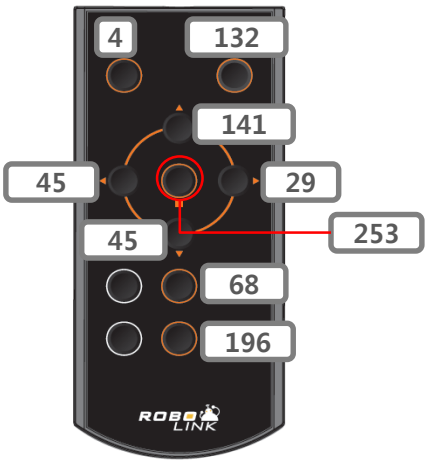
IR 리모컨

채널에 따른 IR 리모컨 키패드 값

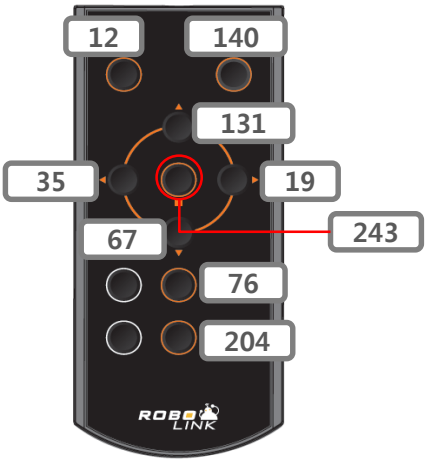
CH1



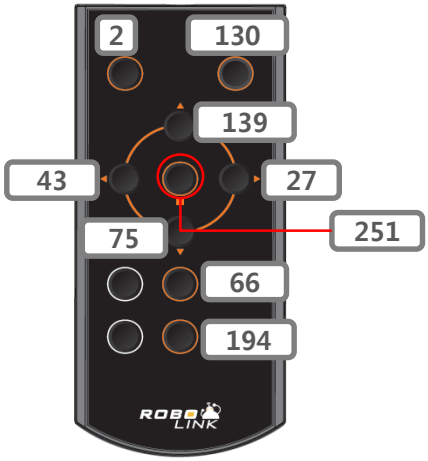
CH2



CH3



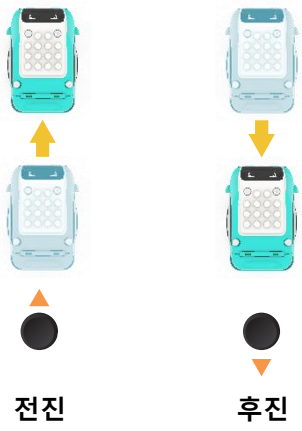
CH4



※ 리모컨의 채널은 리모컨의 채널 버튼을 리모컨 상단의 붉은색 LED가 깜빡일 때 까지 약 4초간 꾹 눌러주시면 변경 됩니다.

※ 주미 패키지에 IR 리모컨은 포함 되어 있지 않습니다.

예제 12 로보링크 IR 리모컨의 채널을 1로 하고, 전진 버튼을 누르면 주미가 전진하고 후진 버튼을 누르면 주미가 후진하게 해 봅시다.



C

배터리 체크 블록 :
배터리의 상태를 전압으로 알려줍니다. 배터리 값을 체크하여 전압이 주미가 안정적으로 동작할 수 있는 범위인지를 살펴 볼 수 있습니다.
(주미가 안정적으로 운영되는 배터리의 전압 값은 3.45V ~ 4.14V 사이입니다)

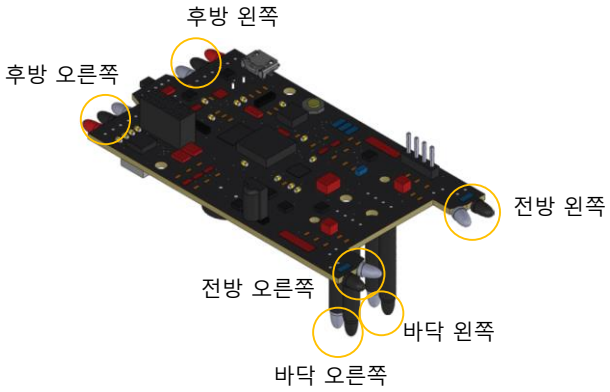
배터리 체크

D

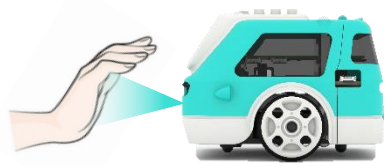
센서값 읽기 블록 :
주미의 IR센서값들을 출력해 줍니다.
주미는 6개의 센서가 있으며 배치는 옆 그림과 같습니다.

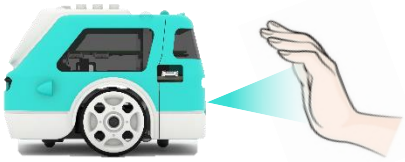
센서값 읽기

- 전방 왼쪽
- 전방 오른쪽
- 후방 왼쪽
- 후방 오른쪽
- 바닥 왼쪽
- 바닥 오른쪽



예제 13 **주미가 멈추어 있다가 전방 센서에 물체나 손을 감지하면 뒤로 후진하고 후방 센서에 감지하면 전진하도록 해 봅시다.**

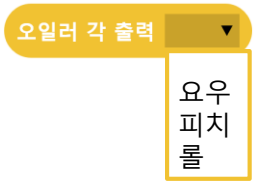




E

오일러 값 출력 블록 :
주미 동체의 기울어진 정도와 방향을 출력해 줍니다.

- **롤 (Roll)** : 좌우 기울어짐
- **피치(Pitch)** : 앞뒤 기울어짐
- **요우(yaw)** : 주미의 방향



예제 14

주미의 전원을 켜면 가만히 있다가
주미의 뒷부분을 손가락으로 눌러서
앞부분을 조금 들어올린 후 놓으면
1초간 전진하도록 해 봅시다.

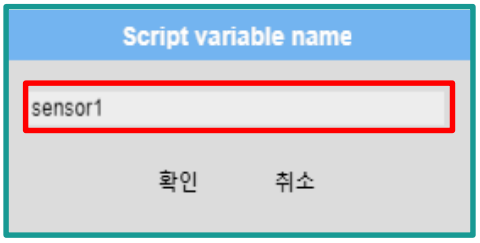
(들어올려졌을 때 버저가 소리가 나게 합니다)

07 스크립트 변수 사용법

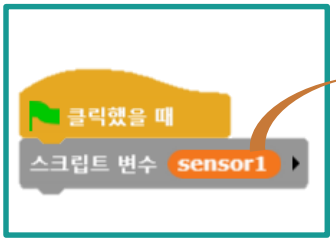
실시간 제어 모드와는 달리 변수 사용을 위해서는
먼저 스크립트 변수 블록이 필요합니다.

스크립트 블록 준비

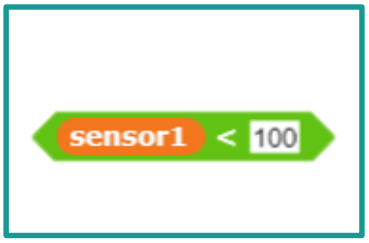
주황색 블록 클릭



이름 입력 후 확인



스크립트 블록 준비



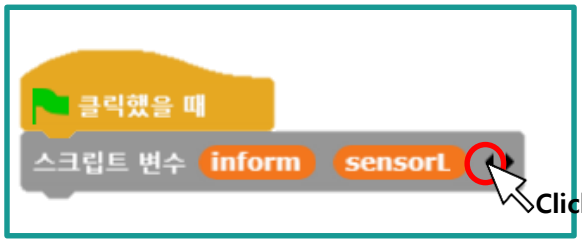
블록에서 직접 드래그 해서 사용

새로운 변수를 만들고자 할 때에는 스크립트 변수 블록의 화살표를 클릭 합니다.

스크립트 블록 준비

클릭하면 변수 블록이 늘어남

만들어 놓은 변수 블록을 삭제하고자 할 때는 반대편 화살표를 클릭합니다.



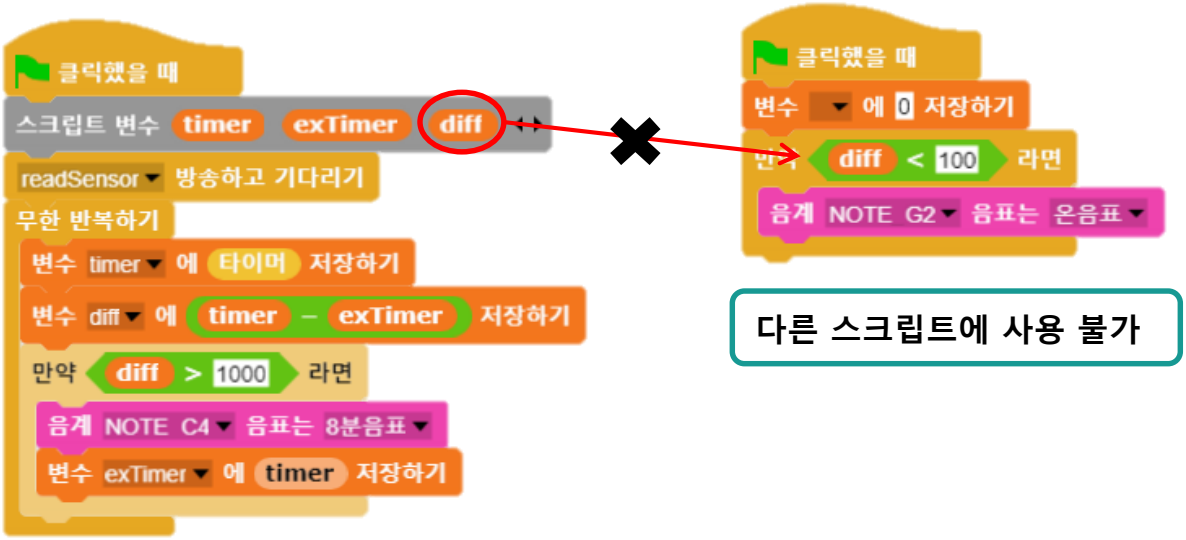
스크립트 블록 준비

블록이 삭제 됩니다.

알아두기

스크립트 변수는 ‘클릭했을 때’ 블록과 연결된 스크립트에만 영향을 미치고 다른 조건 블록등으로 시작하는 스크립트에는 사용할 수 없습니다.

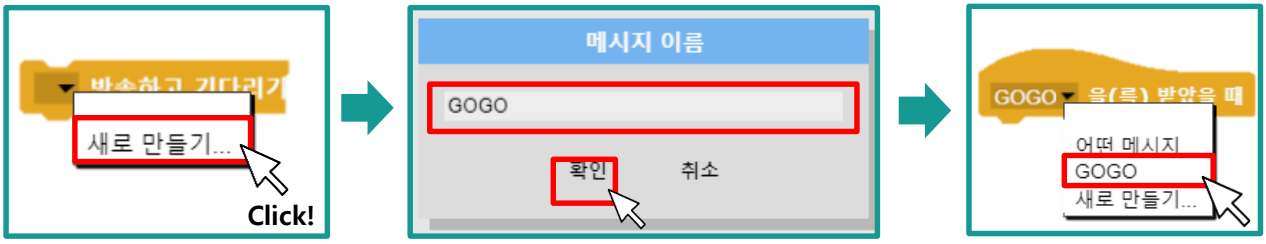
다른 조건 블록으로 시작하는 스크립트는 독립된 다른 프로그램으로 보셔야 합니다.



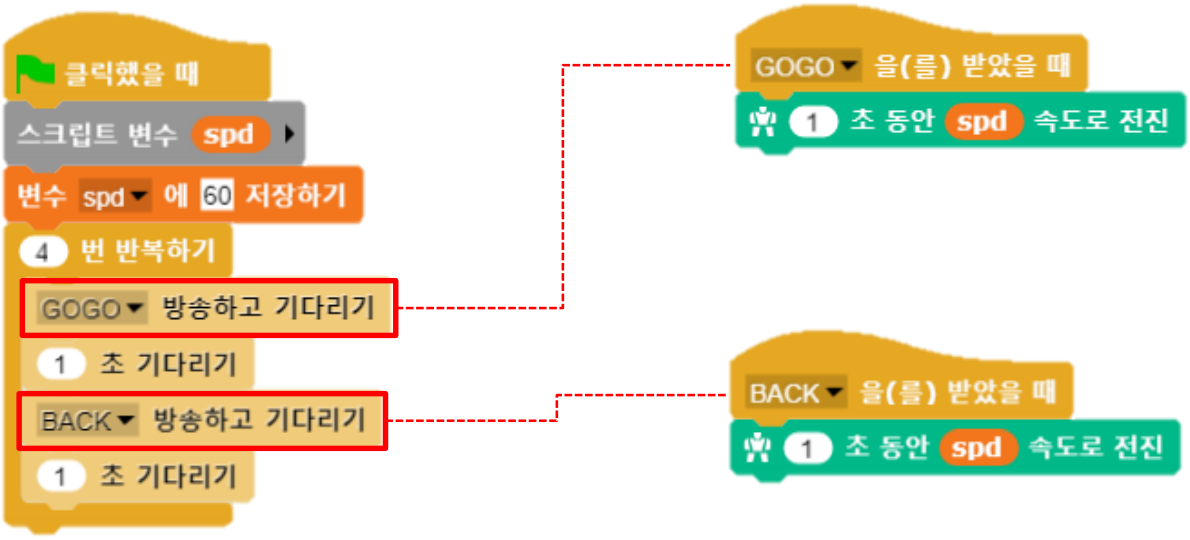
08 방송하고 기다리기 블록 사용법

스크립트가 길어지거나 반복해서 실행해야 하는 스크립트들이 있는 경우 따로 스크립트들을 분할하여 필요할 때 분할된 스크립트를 호출하는 형태로 스크립트를 만들 수 있습니다.

(텍스트 언어로 작성하는 프로그램에서 함수 만들기와 함수 호출에 해당합니다)



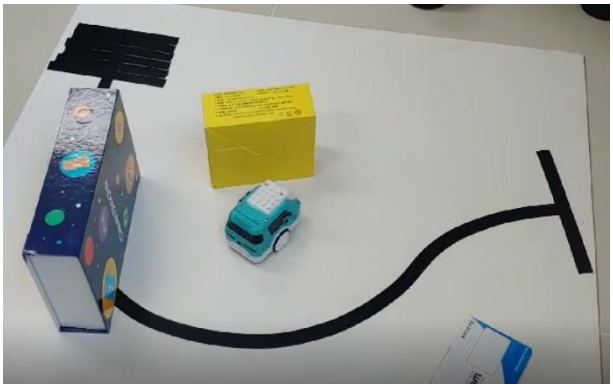
새로 만들기 클릭 | 이름 입력 후 확인 | 받았을 때 설정



09 응용 예제

예제 1 **주미가 전방에 있는 장애물을 피하면서 계속 주행할 수 있는 어보이더 기능을 구현해 봅시다.**

※ 센서는 전방에 있는 왼쪽과 오른쪽 적외센서를 사용합니다.



```
when [flag] clicked
script variables sensor_L sensor_R sp
set sp to 70
wait 1 secs
forever
  OLED Output IR SENSOR
  set sensor L to sensor reading THE FRONT LEFT
  set sensor R to sensor reading THE FRONT RIGHT
  if sensor_L < 100 or sensor_R < 100
    if sensor_L < 100
      moving to BACKWARD in speed sp
      wait 0.3 secs
      moving to RIGHT in speed sp
      wait 0.5 secs
    if sensor_R < 100
      moving to BACKWARD in speed sp
      wait 0.3 secs
      moving to LEFT in speed sp
      wait 0.5 secs
  else
    moving to FORWARD in speed sp
```

왼쪽 센서 /오른쪽 센서 / 모터 속도를 저장할 변수 생성

두 개의 센서값 읽어서 변수에 저장

두 센서 중 하나라도 감지되었는가?
(센서 값 100 이하면 감지된 것으로 봄)

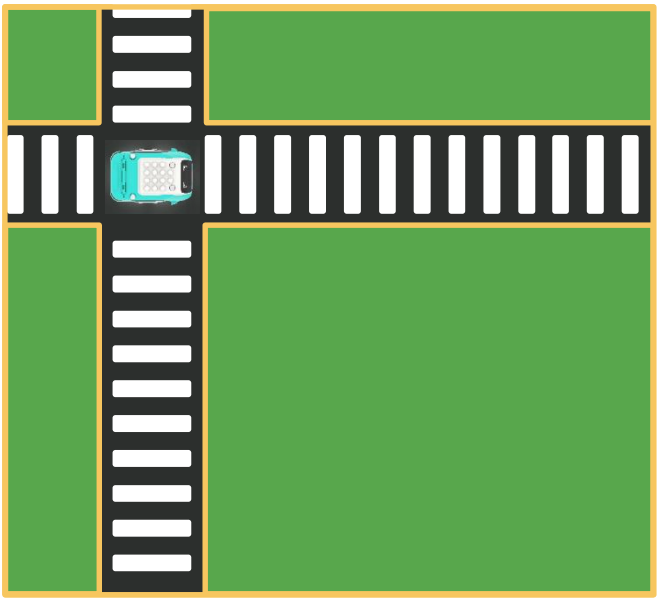
왼쪽 센서 감지시 후진후 오른쪽으로 회전

오른쪽 센서 감지시 후진후 왼쪽으로 회전

감지 안된 경우엔 전진

예제 2

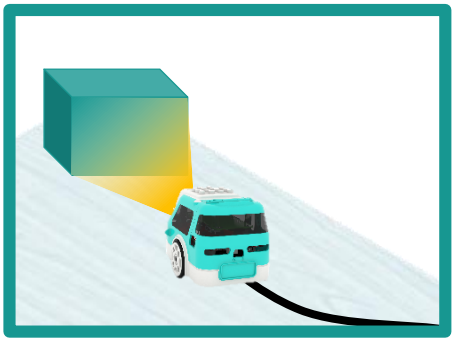
아래의 사진과 같은 블록 레일을 따라 10칸 전진 후
오른쪽으로 꺾어서 다시 10칸 전진하여
'ㄱ' 자 형태로 이동할 수 있도록 스크립트를 작성하도록 해 봅시다.



START

예제 3 주미의 바닥 센서 2개로 폭 2cm 이내의 선을 따라갑니다. 선을 따라가는 도중 벽을 만나면 감지하여 멈추도록 해 봅니다.

```
when [flag] clicked
script variables sensorBL sensorBR
script variables baseL baseR spd status frontL frontR
forever
  set baseL to 70
  set baseR to 70
  set spd to 50
  set status to 0
  set sensorBL to sensor reading THE BOTTOM LEFT
  set sensorBR to sensor reading THE BOTTOM RIGHT
  set frontL to sensor reading THE FRONT LEFT
  set frontR to sensor reading THE FRONT RIGHT
  if sensorBL > baseL and sensorBR > baseR
    set status to 1
    DCmotor M1 speed CCW with spd
    DCmotor M2 speed CW with spd
  if sensorBL < baseL and sensorBR > baseR
    set status to 1
    DCmotor M1 speed CCW with spd
    DCmotor M2 speed CCW with spd
  if sensorBL > baseL and sensorBR < baseR
    set status to 1
    DCmotor M1 speed CW with spd
    DCmotor M2 speed CW with spd
  if status = 0
    if sensorBL > baseL
      DCmotor M1 speed CW with spd
      DCmotor M2 speed CW with spd
    else
      if sensorBR > baseR
        DCmotor M1 speed CCW with spd
        DCmotor M2 speed CCW with spd
  if frontL < 100 and frontR < 100
    DCmotor M1 speed STOP with spd
    DCmotor M2 speed STOP with spd
    wait 3 secs
```



전방 센서와 밑바닥 센서 읽기

두 센서가 검은색인 경우 전진

오른쪽만 검은색은 경우 우회전

왼쪽만 검은색인 경우 좌회전

위의 세가지에 해당되지 않은 경우에도 센서값에 따라 모터를 제어해 줌

전방 센서가 감지되면 3초간 멈춤

www.RobolinkSW.com

- 온라인 사이트에서 매뉴얼과 프로그램 다운로드, 교육 자료 등을 확인해주세요..

- 로보링크 SW / Arduino 등이 오픈 소스로 제공 됩니다.