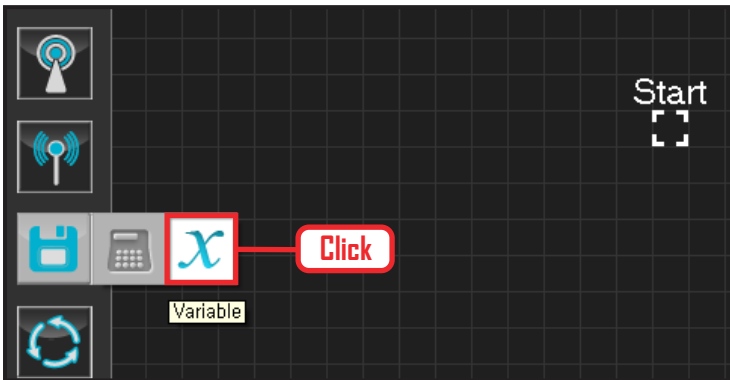


Digital Distance Sensor 예제 따라하기

예제설명

아날로그 거리센서는 거리별로 모두 측정이 가능하지만, 디지털 거리센서는 특정 거리를 기준으로 가깝냐 머냐 만을 판단합니다. 바퀴가 있는 로봇은 벽감지보다는 낭떠러지 감지용으로 많이 활용됩니다. 휴머노이드에서는 발 자체가 움직이기 때문에 낭떠러지 감지보다는 벽감지 프로그래밍을 해봅니다. 아날로그 거리센서와 비교해보기 바랍니다.

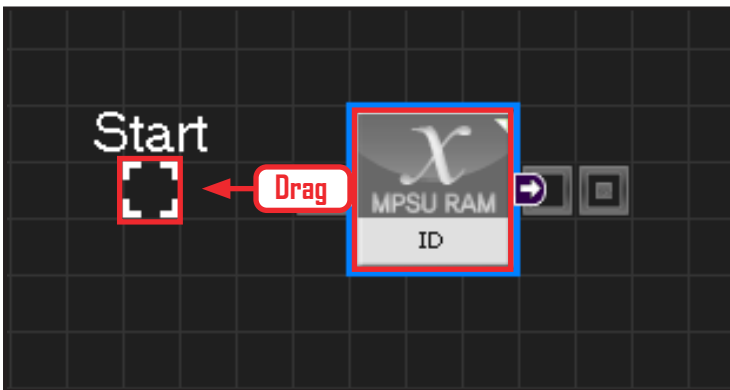
벽이 일정거리 가까워지면 뒷걸음질 하다가 방향을 틀고 전진하는 프로그래밍입니다. 이 예제를 실행하려면 ADC포트 1번 (좌측)에 디지털 거리 센서를 장착한 상태여야 합니다.



01 변수 지정

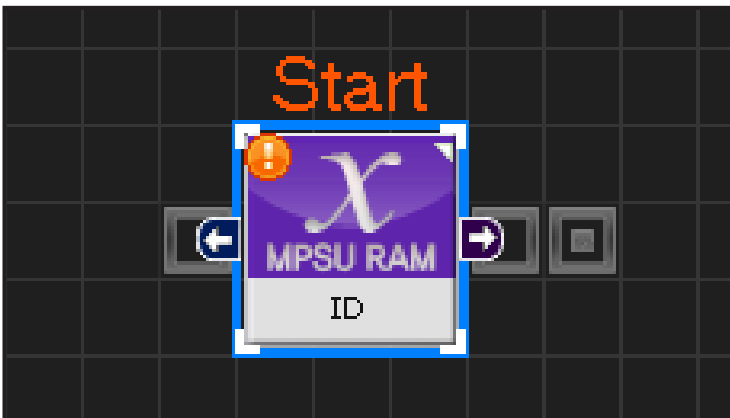
로봇을 동작시킨다는 것은 로봇의 서보 모터를 동작시킨다는 의미입니다. 서보가 스스로 움직일 수 있는 상태로 값을 지정해주어야 합니다.

Data > Variable 모듈을 클릭합니다.



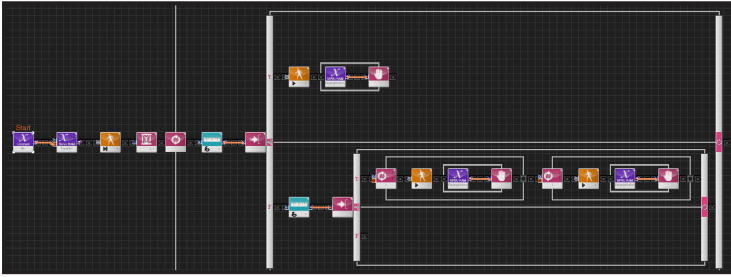
02 시작

모듈의 왼쪽 연결선을 Start Point 에 드래그하여 정확히 도킹을 시킵니다.



03 프로그래밍 시작

모듈과 Start Point 가 정확히 도킹하면 왼쪽과 같이 활성화된 칼라 이미지 모듈로 변합니다. 그럼 프로그래밍이 시작되었다는 의미입니다.

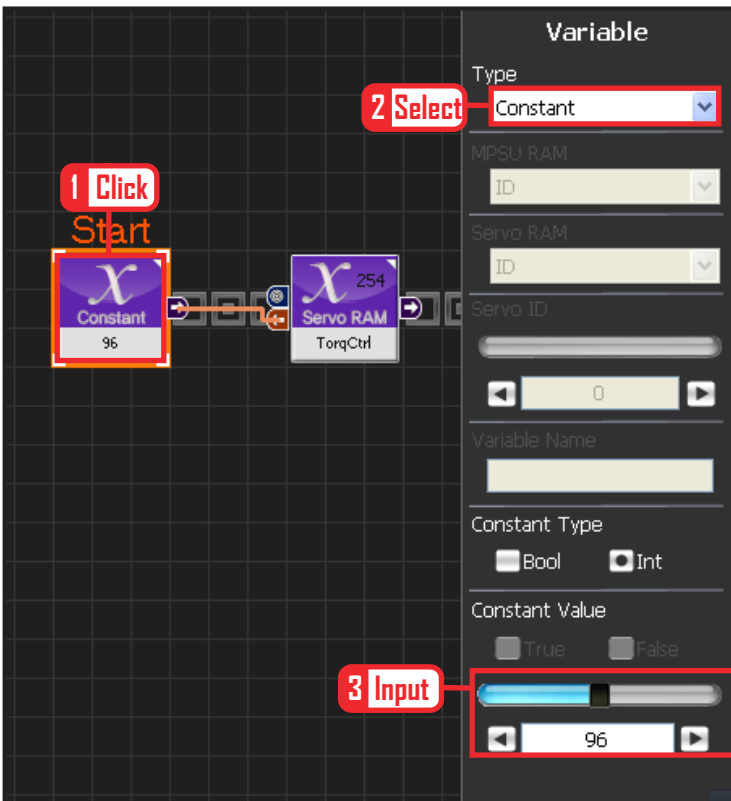


C-like Graphic

```

1 void main()
2 {
3     SERVO_TorqCtrl[254]
4     motionready( 0 )
5     delay( 1500 )
6     while( true )
7     {
8         if( ( MPSU_ADCType1 == 2 && MPSU_ADCVal1 == 1 ) )
9         {
10            motion( 0 )
11            waitwhile( MPSU_PlayingMotion )
12        }
13        else
14        {
15            if( ( MPSU_ADCType1 == 2 && MPSU_ADCVal1 == 0 ) )
16            {
17                for( i = 1 ~ 2 )
18                {
19                    motion( 1 )

```



04 전체 프로그래밍

디지털 거리센서를 이용한 전체 프로그래밍입니다.

05 C-Like 보기

오른쪽 상단의 Graphic 탭에서 C-like 탭을 클릭하면 왼쪽과 같은 Task 프로그래밍 화면이 나옵니다. 디지털 거리센서를 이용한 전체 프로그래밍 화면입니다. C와 유사한 문법 구조를 가지고 있으므로 C 문법 선행학습 효과도 있습니다. 각 모듈별로 클릭하면 커서가 따라서 움직이므로 모듈 별로 Text로 어떻게 변환하는지 확인할 수 있습니다.

06 상수 설정

서보 모터를 스스로 움직일 수 있는 상태로 만드는 과정입니다. Variable Type 을 Constant 로 선택합니다. 속성중에 Constant Value 값을 96 으로 설정합니다. 서보의 TorqControl 레지스터에 96(0x60) 이라는 값이 들어가면 서보가 움직일 수 있는 상태가 됩니다. 그 값은 Output 커넥터를 통하여 뒤 모듈의 토크값에 전달합니다.



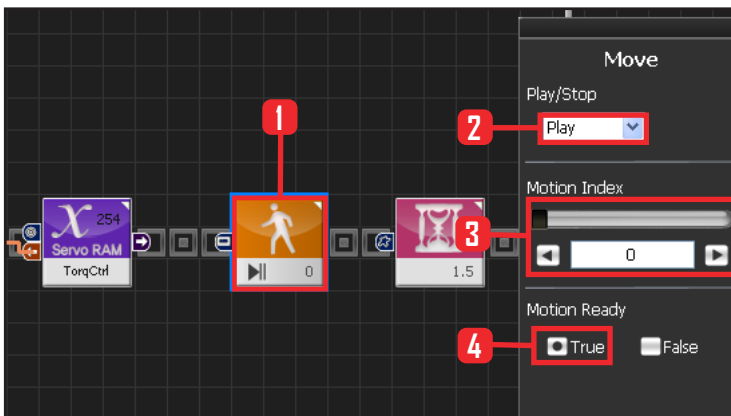
07 모든 서보에 적용

앞에서 받은 96 상수값을 모든 서보에 적용하는 과정입니다.

Variable > Type : Servo RAM을 선택합니다.

Servo RAM : TorqCtrl 을 선택합니다.

Servo ID : 254 를 선택합니다. 254는 연결되어있는 모든 서보에 적용하겠다는 의미입니다.



08 모션 동작준비

모션이 동작하기 위해서는 준비과정을 거칩니다. 이전 로봇 모션에서 갑작스럽게 변동하면 로봇에 무리가 가해질 수 있습니다. 따라서 현재 실행하고자 하는 로봇의 첫 모션으로 천천히 이동시키는 과정입니다. Motion Ready 가 True 이면 모션 첫 장면을 준비하는 것이고, False 이면 모션이 동작합니다.

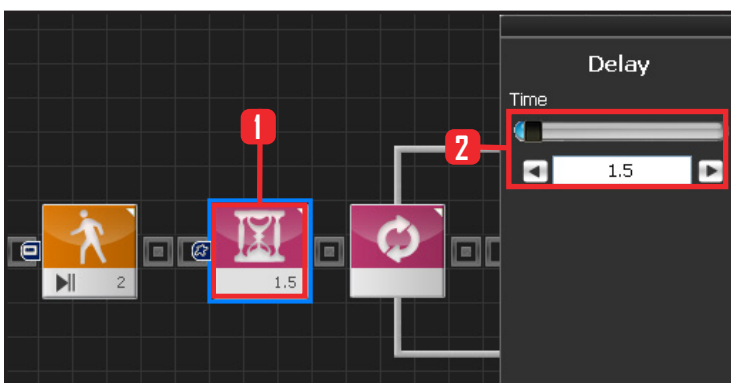
Motion > Move 모듈을 선택합니다.

Play/Stop : Play 를 선택합니다.

Motion Index : 0 번을 선택합니다. 0번은 전진하면서 걷기 모션입니다.

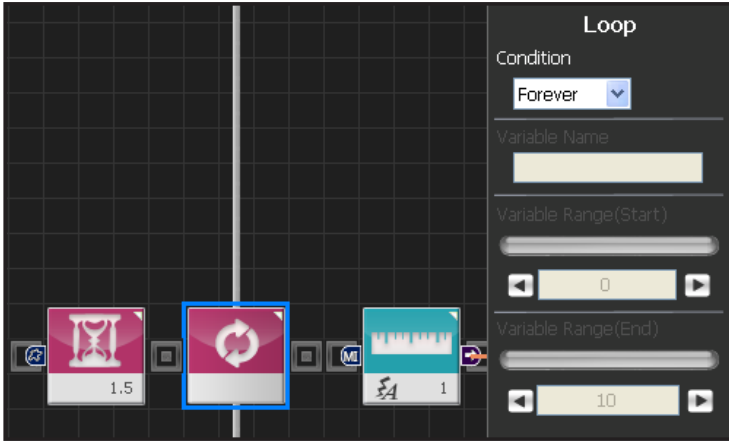
Motion Ready : True 를 선택합니다.

모션 동작 준비 과정입니다.



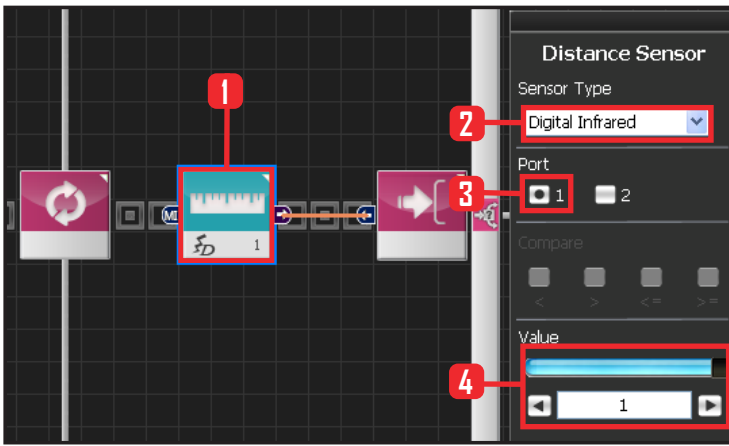
09 Delay

Motion Ready 동작이 끝나기 전에 진행하는 것을 방지하기위해 Delay 값을 1.5 초로 설정합니다.



10 반복

Forever 무한 반복을 설정합니다.



11 디지털 거리센서 설정

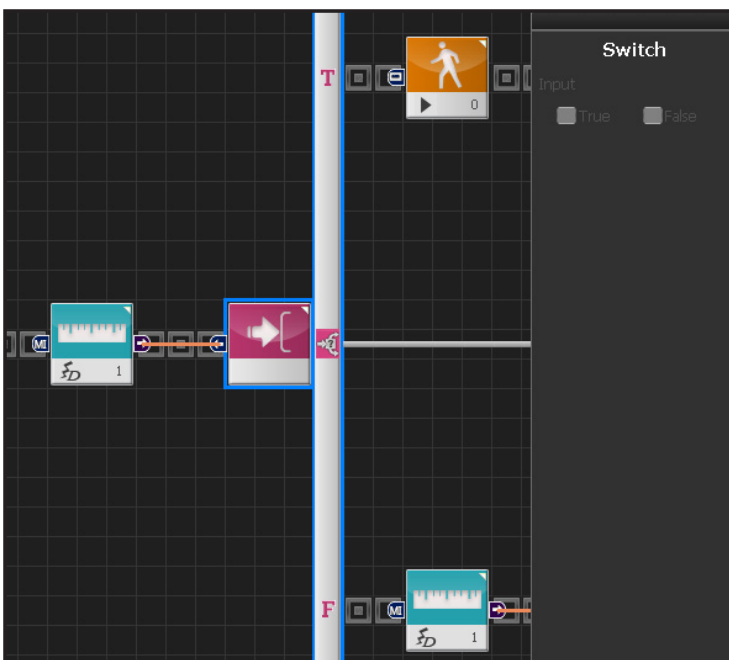
디지털 거리센서는 제품마다 측정 거리가 다릅니다. 여기서는 20cm 기준으로 설정됩니다.

Sensor>Distance Sensor 모듈을 선택합니다.

Sensor Type : Digital Infrared 를 선택합니다.

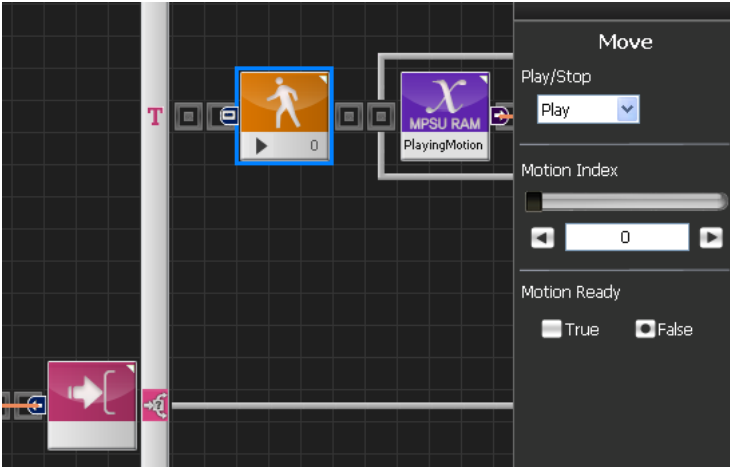
Port : 1번을 선택합니다.

Value : 1 을 선택합니다. 10cm 보다 멀리 있다는 것을 의미합니다.



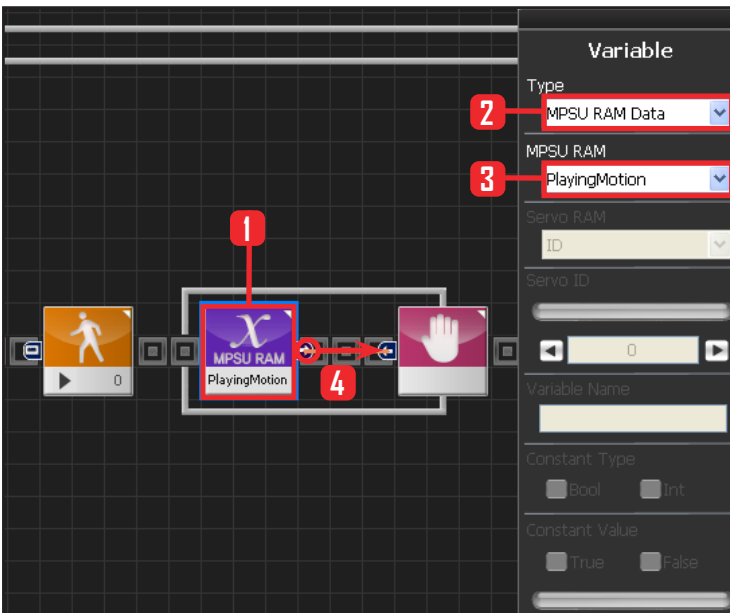
12 If 분기문

True 일때는 전진하고, False 일때는 다음 조건문으로 들어갑니다.



13 전진하기

10cm 보다 먼거리이기 때문에 로봇은 전진합니다. Motion Ready 값을 False 로 선택하면 로봇은 전진 모션을 실행합니다.



14 모션 동작 확인

Loop 는 지속적인 반복을 의미합니다. Move 명령을 내리고 나서 실제 모션이 실행되어 완료되기 까지는 시간이 걸리므로 loop안에 Move모듈 하나만을 넣고 실행하면 모션을 이미 실행중임에도 loop를 계속 돌면서 모션실행 명령을 반복하게 됩니다. 이렇게 되면 Move모듈을 만난 횟수와 실제모션을 실행한 횟수가 달라집니다. 따라서 실행한 모션이 끝날때까지 기다렸다가 다시 loop의 처음으로 돌아가게 하는 편이 더 정확합니다. Variable > MPSU RAM Data 에 들어가면 Playing Motion 이 있습니다. Playing Motion 은 로봇이 모션을 실행중인지 확인하는 변수입니다. 그 Playing Motion에 Wait 를 걸어주면 로봇의 동작이 끝날 때까지 Loop는 기다려줍니다.

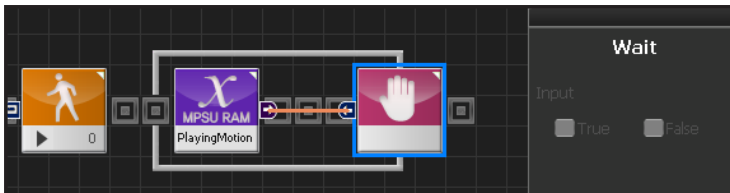
Data > Variable 모듈을 선택합니다.

Type : MPSU RAM Data 를 선택합니다.

MPSU RAM : Playing Motion 을 선택합니다.

Output 커넥터 값을 뒤에 Wait 모듈에 연결합니다.

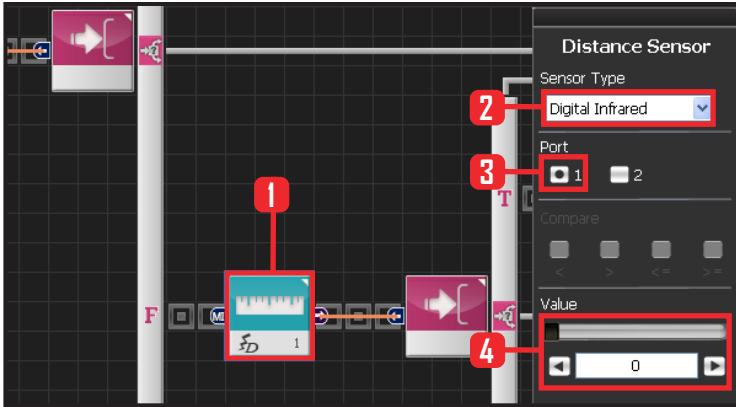
모션이 끝날 때까지 기다리겠다는 의미입니다



15 Wait

모션이 끝날때까지 기다립니다.

모션이 끝나면 다시 처음으로 돌아가 모션을 반복합니다.



16 벽 근접시 동작

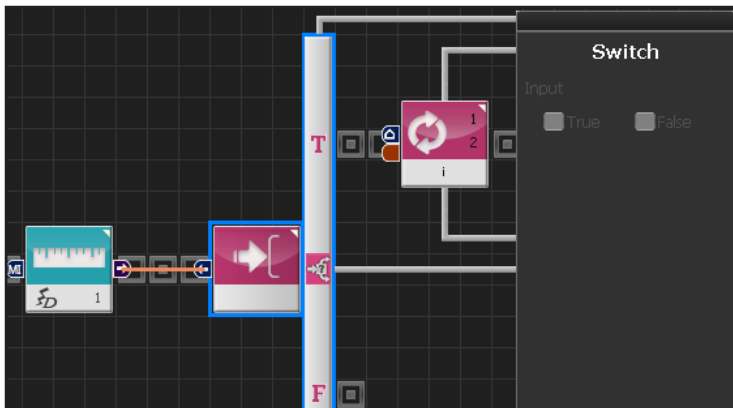
로봇이 벽과 10cm 이내로 근접시에는 로봇을 후진보행 시켰다가 우회전 시키는 프로그래밍입니다.

Sensor > Distance Sensor 모듈을 선택합니다.

Sensor Type : Digital Infrared 를 선택합니다.

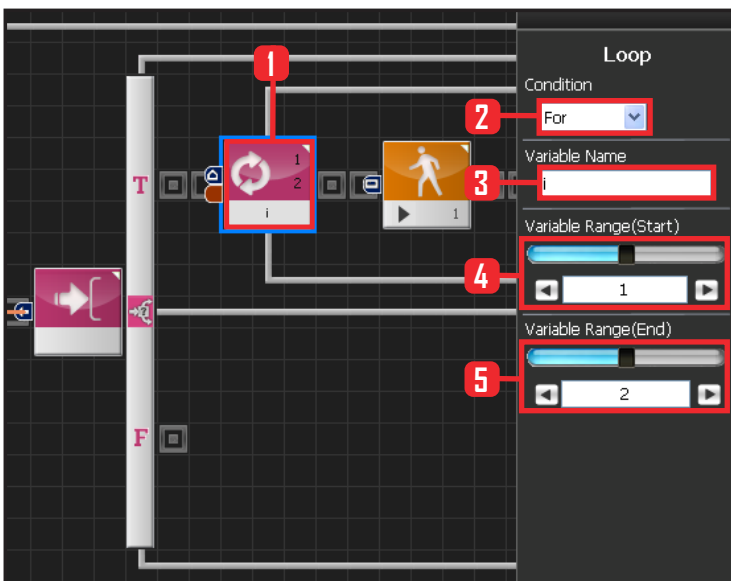
Port : 1번을 선택합니다.

Value : 0 을 선택합니다. 10cm 안에 들어왔다는 의미입니다.



17 If 분기문

10cm 보다 더 가까울 때 True 안의 문을 실행시킵니다.



18 For 반복

로봇의 모션은 같은 동작을 일정하게 반복하여 원하는 모션까지의 동작을 취하게 할 수 있습니다.

모션 1번은 로봇이 후진하는 모션입니다. 후진은 왼발 오른발 한번씩만 뒤로 이동하게 되어있습니다. 그 모션에 원하는 수만큼 반복을 줄 수 있는 For 문을 써서, 원하는 위치만큼 이동시킬 수 있습니다.

Flow > Loop 모듈을 선택합니다.

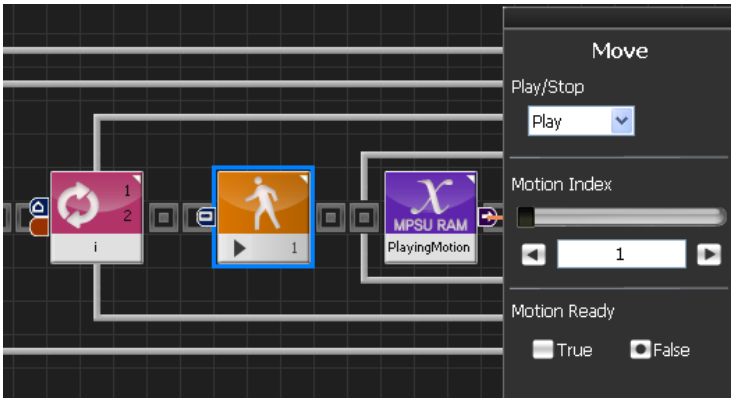
Condition : For 를 선택합니다.

Variable Name 은 i 로 입력합니다.

Variable Range(Start) 1 로 입력합니다.

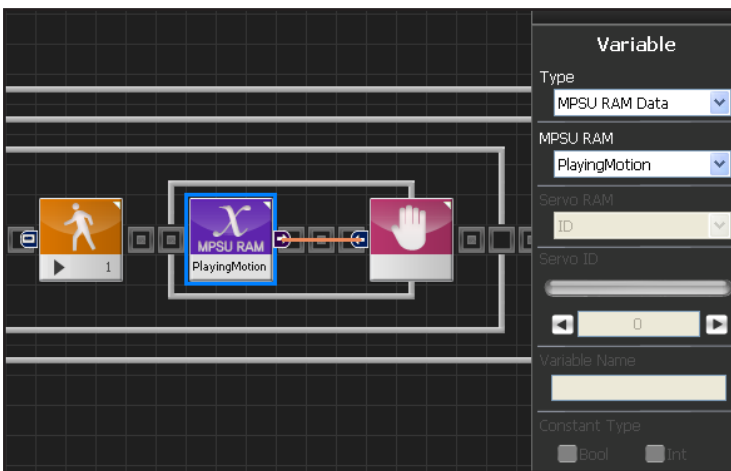
Variable Range(End) 2 로 입력합니다.

모션을 두번을 반복하여 실행시킨다는 의미입니다.



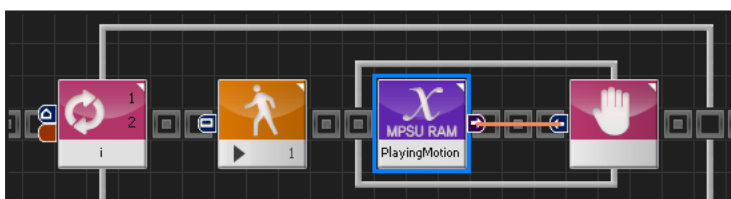
19 뒤로 물러서기

1번은 뒤로 물러서는 모션입니다.
False 를 선택하면 로봇이 후진모션을 실행합니다.



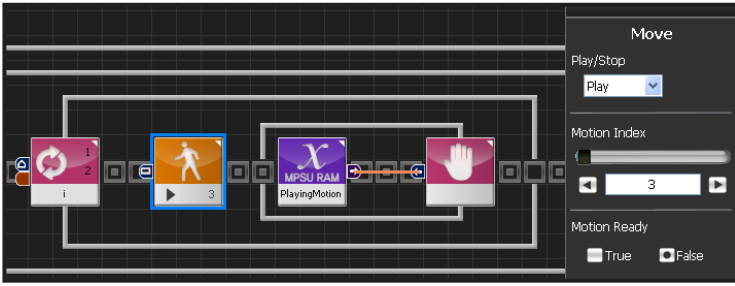
20 동작 감지

Playing Motion 으로 로봇의 동작을 감지하고, 끝나면 다시 for 문의 처음으로 돌아갑니다.



21 두번 반복 후진

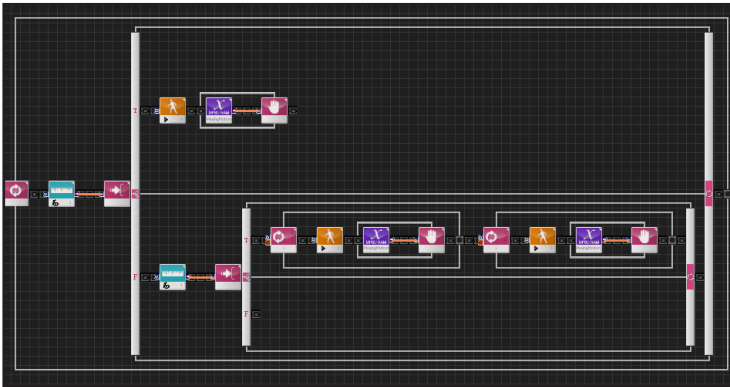
로봇을 두번 반복해서 후진 시키는 프로그래밍이었습니다.



22 우회전 시키기

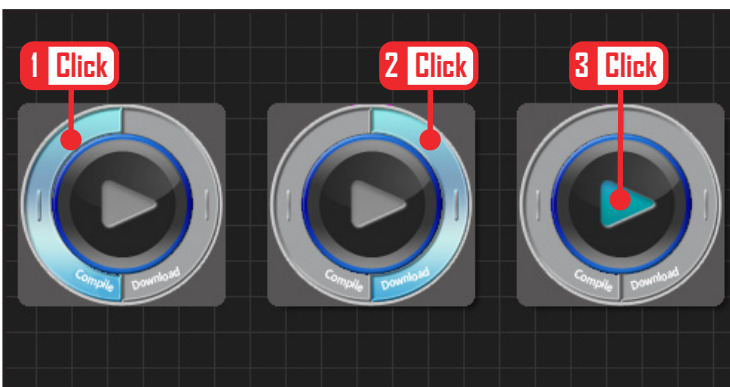
로봇 모션 3 번은 제자리에서 좌회전 하는 동작입니다. 우회전 동작도 마찬가지로 For 문을 써서 원하는 만큼 위치를 조절할 수 있습니다.

Motion 은 3번을 선택하고 For문은 1~3 번까지 설정하여 위와 같이 프로그래밍 합니다.



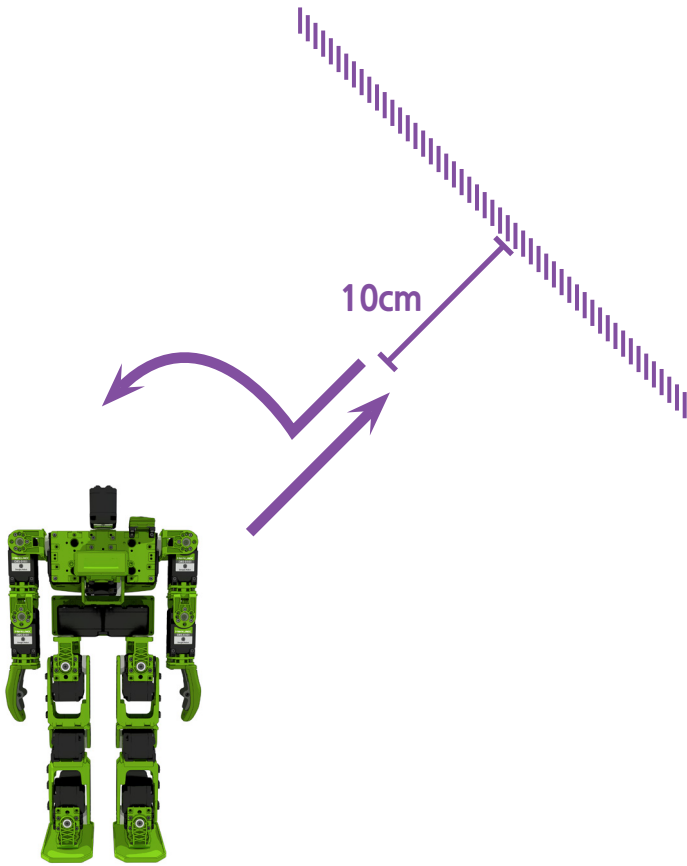
23 전체 살펴보기

10cm 보다 멀때는 로봇을 그대로 전진시키고, 10cm 보다 가까울 때 에는 For 문으로 후진과 우회전 모션을 일정하게 반복시켜 회피하는 프로그래밍입니다.



24 컴파일, 다운로드, 실행

왼쪽 클릭하여 컴파일 시킵니다. 에러가 없으면 오른쪽 클릭하여 로봇에 다운로드 시킵니다. 다운로드 완료되면 가운데 화살표 실행버튼을 눌러 로봇에서 실행시킵니다.



25 로봇동작

로봇이 10cm 앞의 벽을 감지하면 후진하였다가 우회 전 한 후 전진합니다.