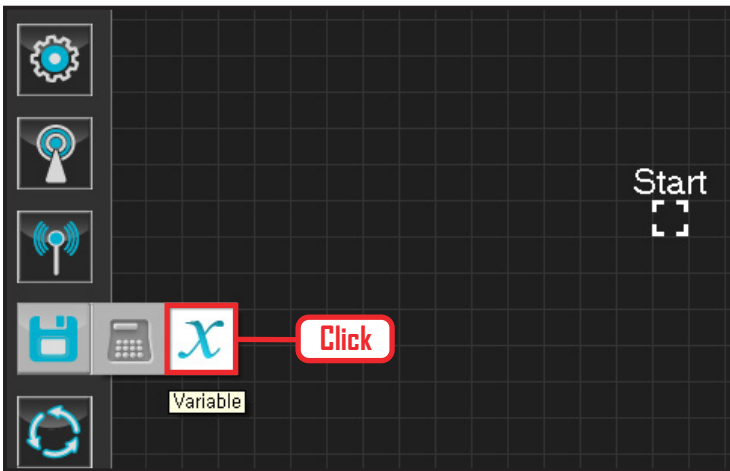


Move 예제 따라하기

예제설명

Move 는 제어기 DRC에 저장된 로봇의 모션을 가져와 프로그램에 적용하는 것입니다. 로봇 모션은 번호로 가져올 수 있으며, 번호별 모션 이름은 DR-SIM 에서 확인할 수 있습니다. DR-SIM 에서 편집된 특정 모션을 로봇에서 무한반복적으로 돌리는 프로그래밍을 해봅니다. 로봇 신뢰성 테스트나 전시등에서 활용도가 높고, 난이도가 있는 프로그램이므로 천천히 따라해보기 바랍니다.

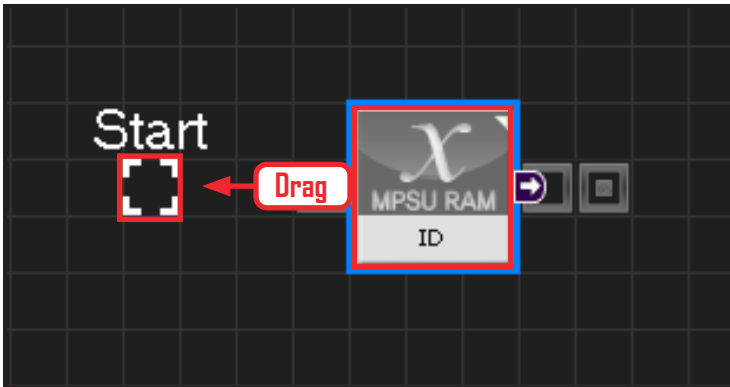
* 이 예제에서 사용하는 모션과 모션번호는 기본제공 모션과 다릅니다. DR-SIM 을 통해 편집한 모션을 DRC에 다운로드 한 경우를 가정하고 작성된 예제입니다.



01 변수 지정

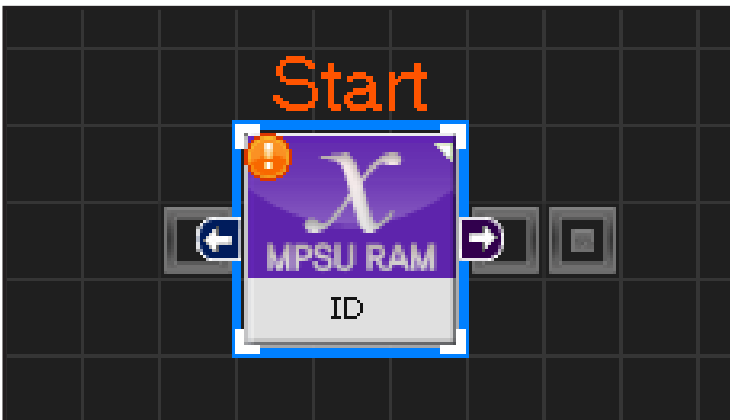
로봇을 동작시킨다는 것은 로봇의 서보 모터를 동작 시킨다는 의미입니다. 서보가 스스로 움직일 수 있는 상태로 값을 지정해주어야 합니다.

Data > Variable 모듈을 클릭합니다.



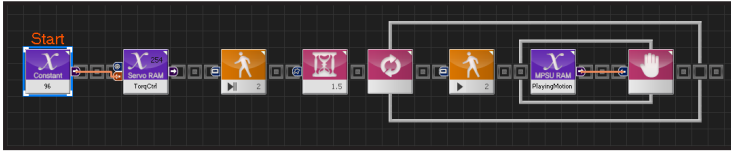
02 시작

모듈의 왼쪽 연결선을 Start Point 에 드래그하여 정확히 도킹을 시킵니다.



03 프로그래밍 시작

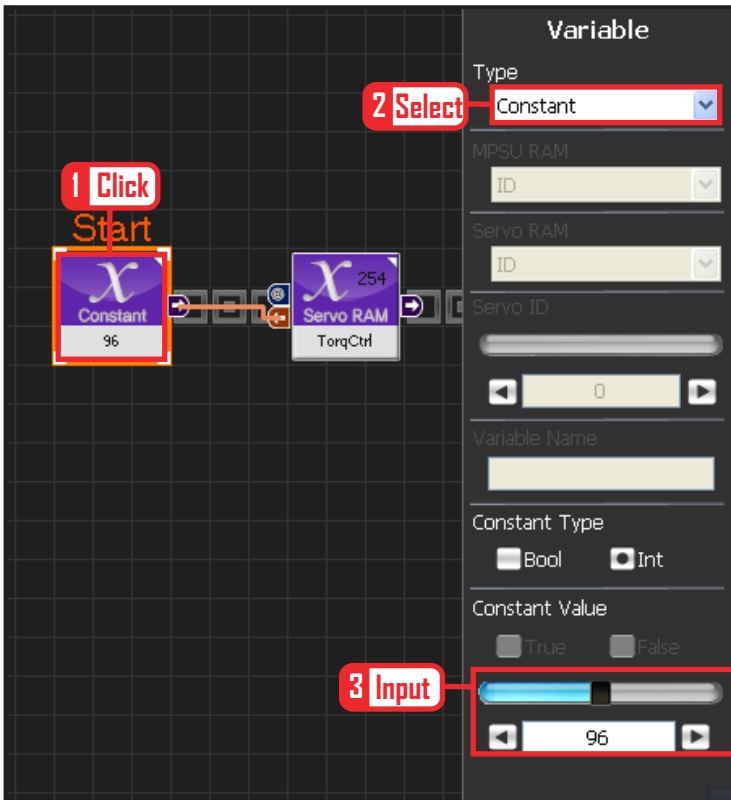
모듈과 Start Point 가 정확히 도킹하면 왼쪽과 같이 활성화된 칼라 이미지 모듈로 변합니다. 그럼 프로그래밍이 시작되었다는 의미입니다.



C-like Graphic

```

motion_move... x
1 void main()
2 {
3     SERVO_TorqCtrl [254] =96
4     motionready( 2 )
5     delay( 1500 )
6     while( true )
7     {
8         motion( 2 )
9         waitwhile( MPSU_PlayingMotion )
10    }
11 }
    
```



04 전체 프로그래밍

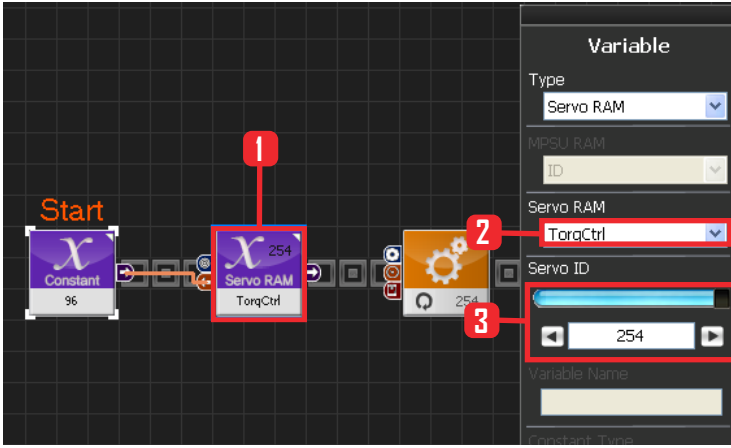
저장된 모션을 가져와 일정하게 반복 시키는 프로그래밍입니다. Motion ready 값에 주의합니다.

05 C-Like 보기

오른쪽 상단의 Graphic 탭에서 C-like 탭을 클릭하면 왼쪽과 같은 Task 프로그래밍 화면이 나옵니다. 저장모션을 이용한 전체 프로그래밍 화면입니다. C와 유사한 문법 구조를 가지고 있으므로 C 문법 선행학습 효과도 있습니다. 각 모듈별로 클릭하면 커서가 따라서 움직이므로 모듈별로 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.



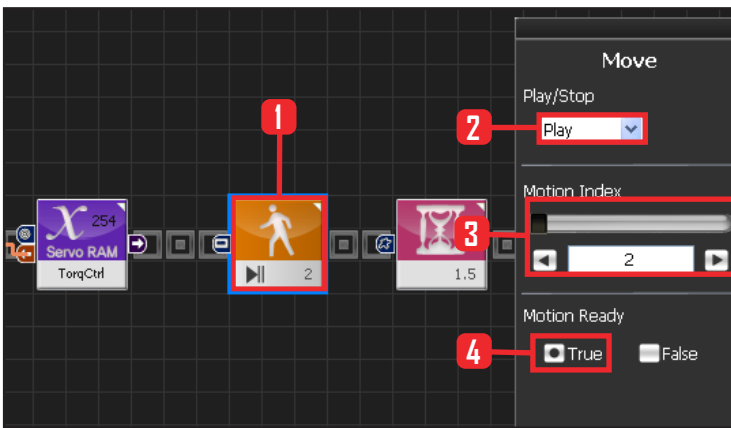
07 모든 서보에 적용

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 Motion Ready

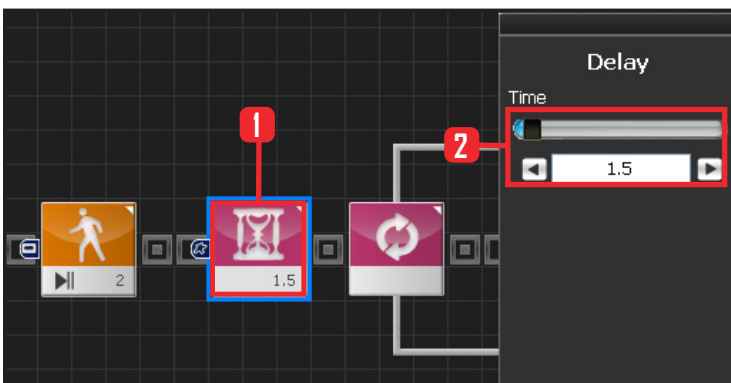
저장된 모션을 가져올 때 로봇의 현재 상태에서 갑작스럽게 모션이 변동하고 움직일 수가 있습니다. 현재 상태와 모션 시작상태가 너무 다르면 모터에 무리가 가거나 사용자에게 위험할 수도 있습니다. 그래서 Motion Ready 즉 모션이 동작할 준비시간을 줘야합니다.

Motion > Move 를 선택합니다.

Play/Stop : Play 를 선택합니다.

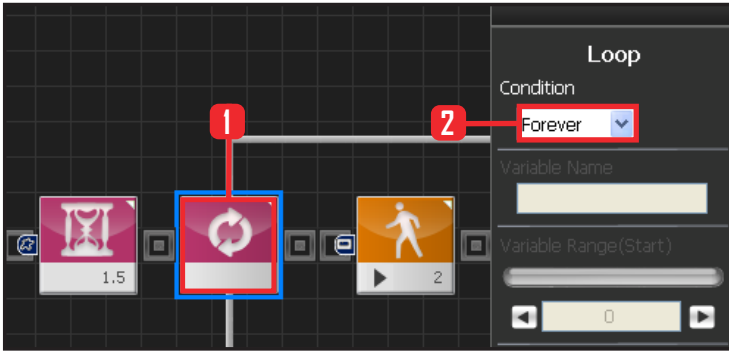
Motion Index : 2를 선택합니다. 2번 모션을 가져오겠다는 뜻입니다. 참고로 이 프로그래밍의 2번 모션은 없었다 일어나는 것입니다. 꼭 2번일 필요는 없으며, 사용자가 실행하고 싶은 모션 번호를 입력하면 됩니다.

Motion Ready : True 를 선택합니다. True 를 선택하면 동작하고자 하는 모션의 첫 번째 상태로 서서히 이동합니다.



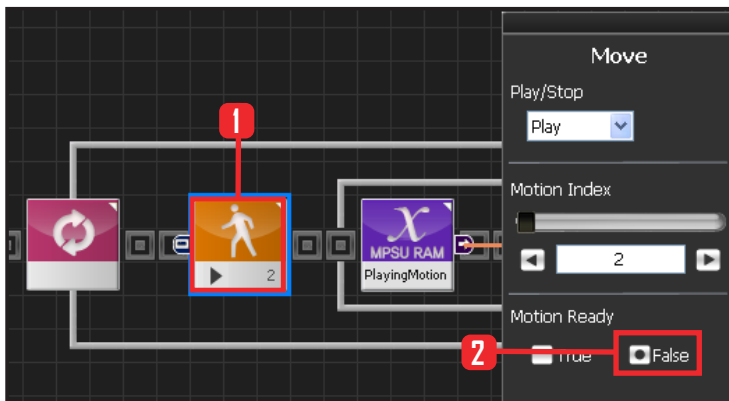
09 Delay

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



10 반복

Forever 무한 반복을 설정합니다.



11 모션 동작

Motion Ready 값을 False 로 설정하면 모션을 처음 부터 끝까지 동작시킨다는 의미입니다.

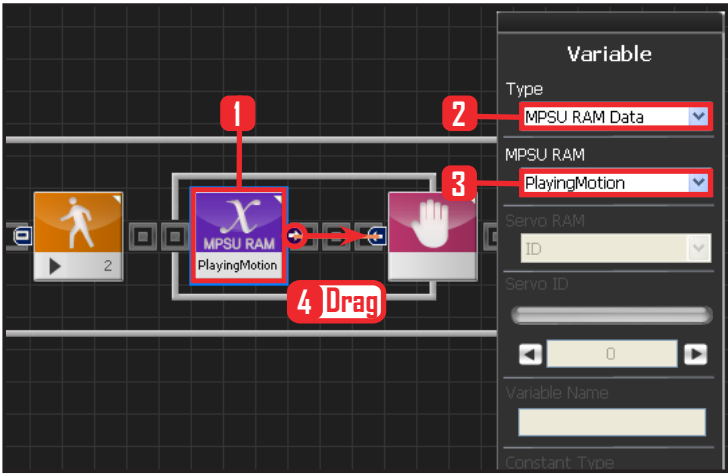
Motion Ready : False 로 선택합니다.



참조: 모션 보기

DR-SIM 프로그램에서 로봇과 연결하여 로봇설정을 클릭하면, 현재 제어기에 있는 로봇 모션을 확인할 수 있습니다.

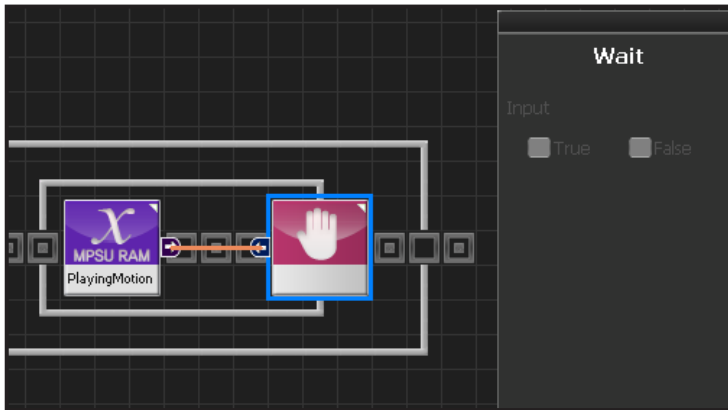
현재 2번 모션은 양팔을 벌려 앉았다 일어나는 모션입니다.



12 모션 동작 확인

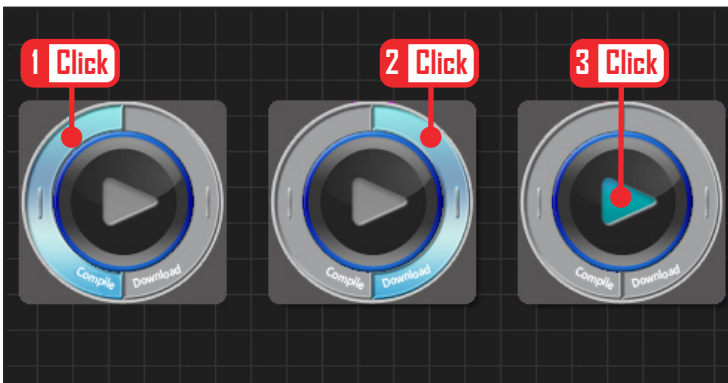
Loop 는 지속적인 반복을 의미합니다. Move 명령을 내리고 나서 실제 모션이 실행되어 완료되기 까지는 시간이 걸리므로 loop안에 Move모듈 하나만을 넣고 실행하면 모션을 이미 실행중임에도 loop를 계속 돌면서 모션실행 명령을 반복하게 됩니다. 이렇게 되면 Move모듈을 만난 횟수와 실제모션을 실행한 횟수가 달라집니다. 따라서 실행한 모션이 끝날때까지 기다렸다가 다시 loop의 처음으로 돌아가게 하는 편이 더 정확합니다. Variable > MPSU RAM Data 에 들어가면 Playing Motion 이 있습니다. Playing 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.



13 Wait

모션이 끝날때까지 기다립니다.
 모션이 끝나면 다시 처음으로 돌아가 모션을 반복합니다.



14 컴파일, 다운로드, 실행

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

1



2



15 로봇동작

로봇이 앉았다 일어나는 모션을 반복적으로 수행합니다.