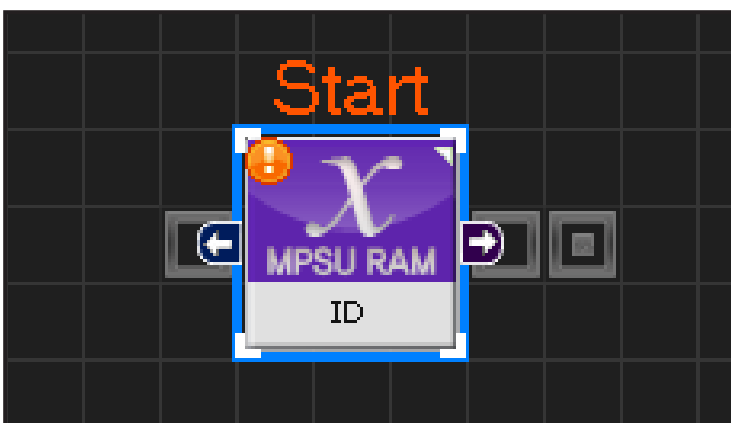
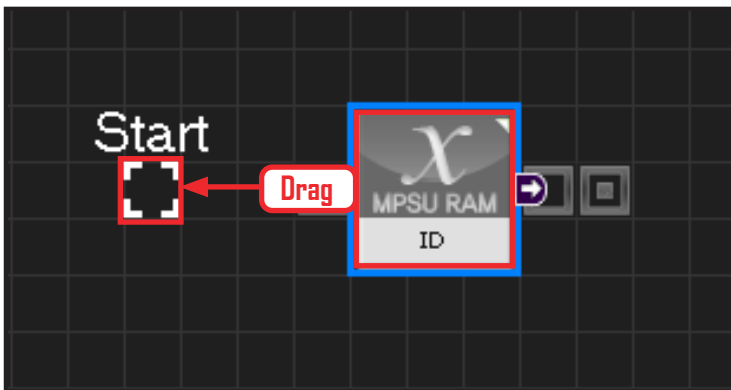
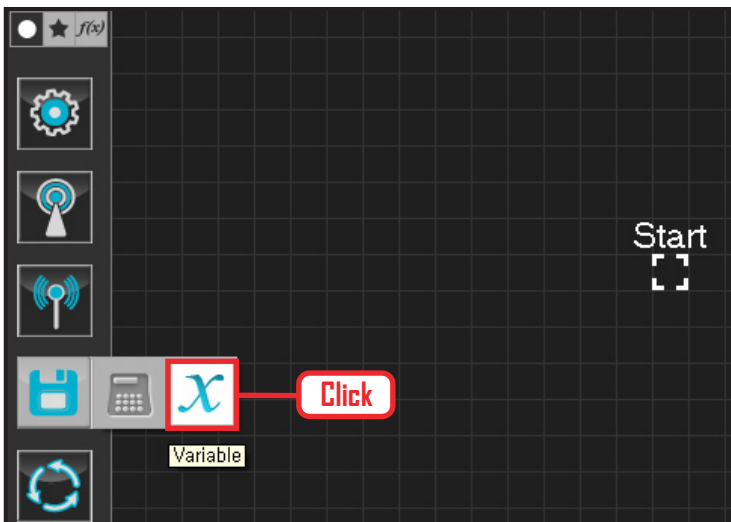


### Light 예제 따라하기

#### 예제설명

빛의 세기에 따라 로봇의 모터를 동작하는 예제입니다.

외부의 빛이 어두워지면 로봇이 왼팔을 올립니다. (제어기 뒤쪽의 CDS 센서를 손바닥으로 가리면 빛의 들어오는 양이 없어 어두워지므로, 로봇이 왼팔을 올리도록 프로그래밍합니다.)



### 01 변수 지정

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

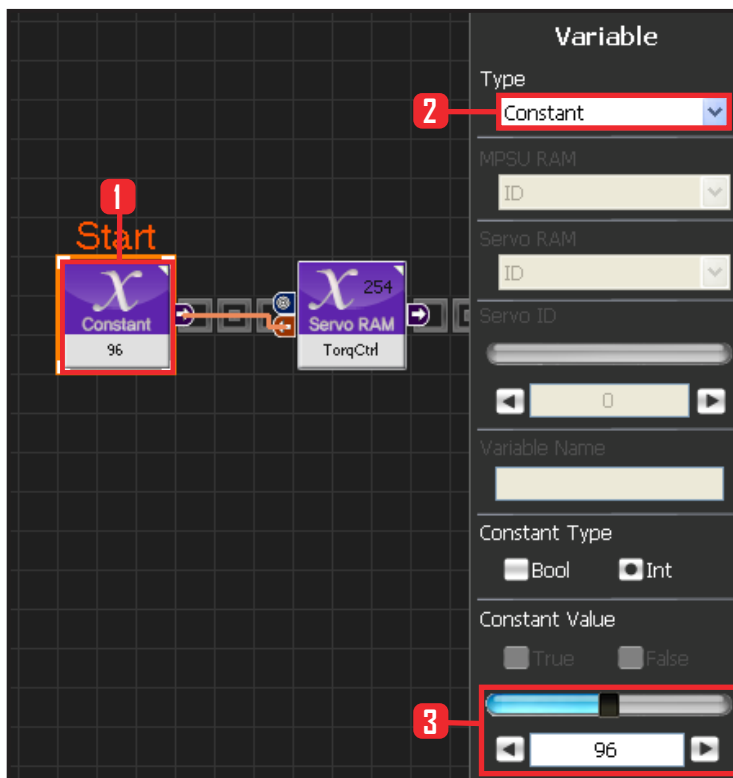
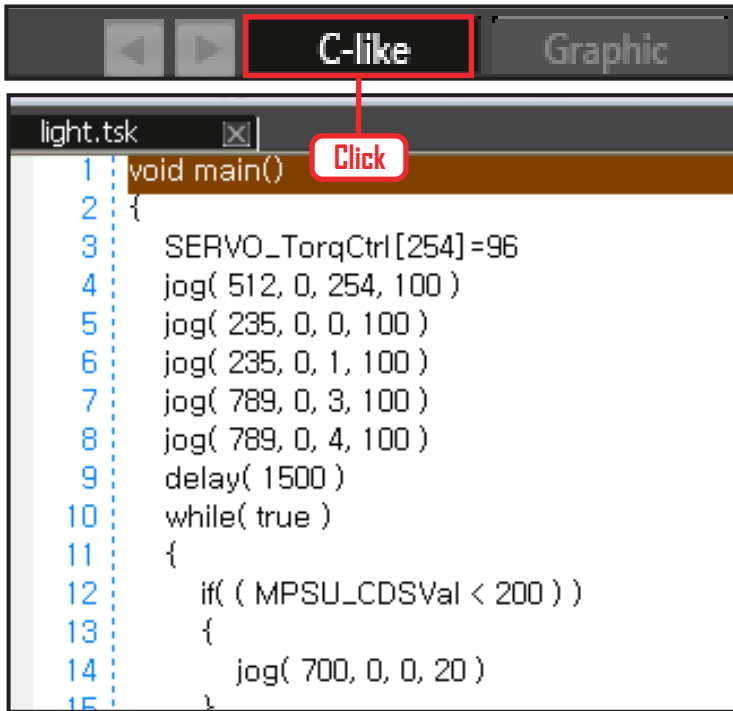
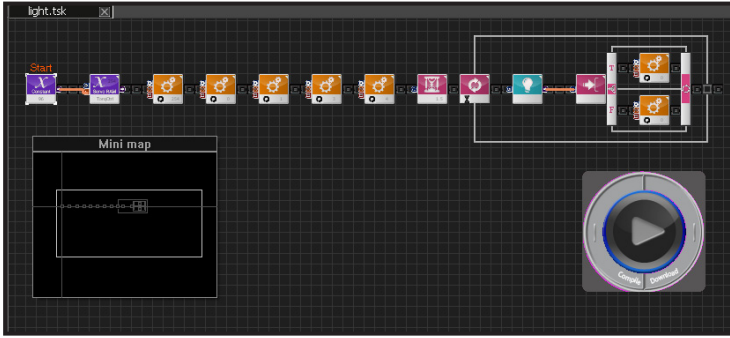
Data > Variable 모듈을 클릭합니다.

### 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.



## 04 전체 프로그래밍

빛 센서를 이용하여 로봇의 모터를 움직이는 전체 프로그래밍 전개 화면입니다.

## 05 C-Like 보기

오른쪽 상단의 Graphic 탭에서 C-like 탭을 클릭하면 왼쪽과 같은 Task 프로그래밍 화면이 나옵니다. 빛센서를 이용한 전체 프로그래밍 화면입니다.

C와 유사한 문법 구조를 가지고 있으므로 C 문법 선행학습 효과도 있습니다.

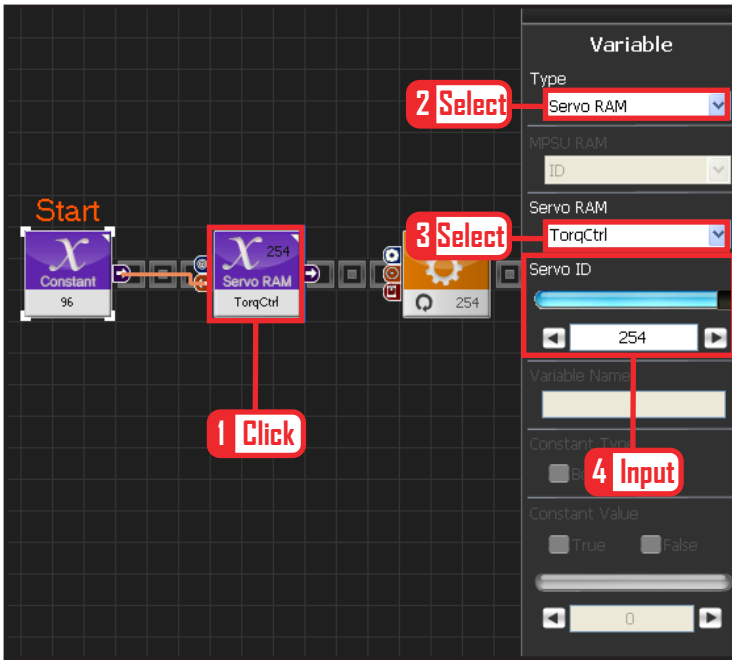
각 모듈별로 클릭하면 커서가 따라서 움직이므로 모듈별로 Text로 어떻게 변환하는지 확인할 수 있습니다.

## 06 상수 설정

서보 모터를 스스로 움직일 수 있는 상태로 만드는 과정입니다.

Variable Type 을 Constant 로 선택합니다.

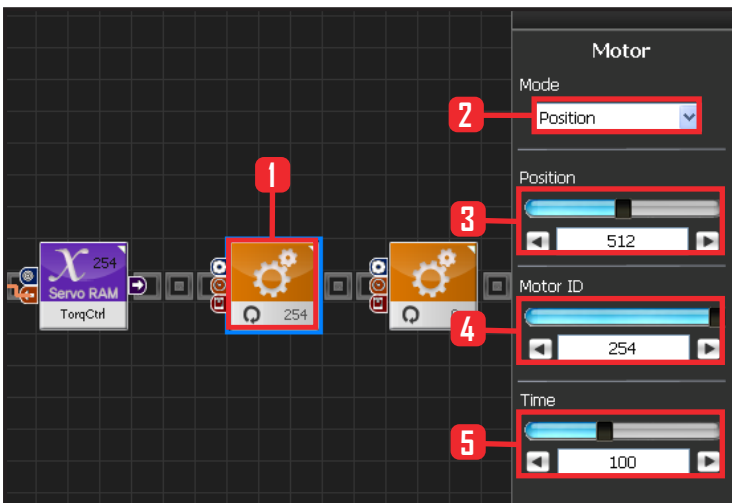
속성중에 Constant Value 값을 96 으로 설정합니다. 서보의 TorqControl 레지스터에 96(0x60) 이라는 값이 들어가면 서보가 움직일 수 있는 상태가 됩니다. 그 값은 Output 커넥터를 통하여 뒤 모듈의 토크값에 전달합니다.



### 07 모든 서보에 적용

앞에서 받은 96 상수값을 모든 서보에 적용하는 과정입니다.

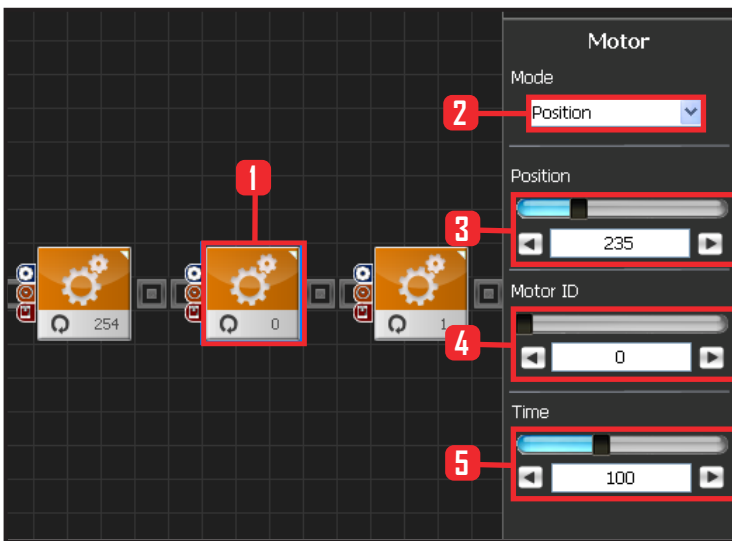
Variable > Type : Servo RAM을 선택합니다.  
 Servo RAM : TorqCtrl 을 선택합니다.  
 Servo ID : 254 로 설정합니다. 254는 연결 되어있는 모든 서보에 적용하겠다는 의미입니다.



### 08 모든 서보모터 각도 설정

모든 서보모터의 각도를 중앙에 보내는 과정입니다.

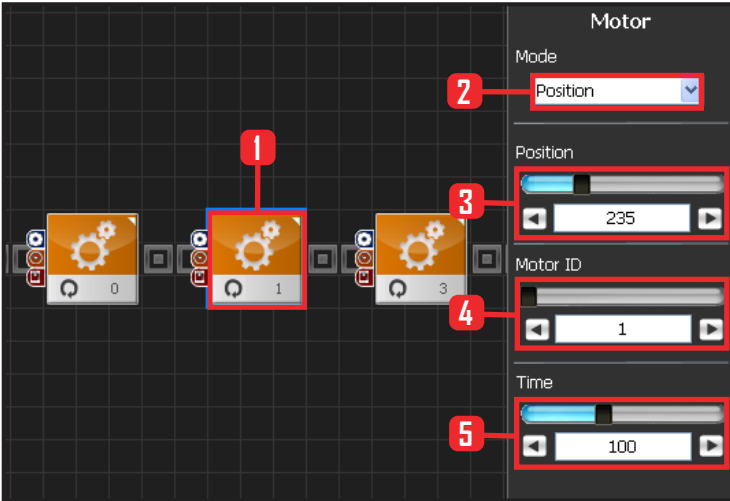
Motion > Moter 를 선택합니다.  
 Mode : Positon 으로 선택합니다. 각도를 조절합니다.  
 Position : 512 로 설정합니다. 512 번은 모터를 중앙으로 보낸다는 의미입니다.  
 Motor ID : 254 로 설정합니다. 254 는 모든 모터에 적용하겠다는 의미입니다.  
 Time : 100 으로 설정합니다. 단위는 1당 11.2ms로, 100은 약 1.12초를 의미합니다.  
 1.12초동안 원하는 각도로 이동시킨다는 의미입니다.



### 09 모터 0번 (오른쪽 어깨) 설정

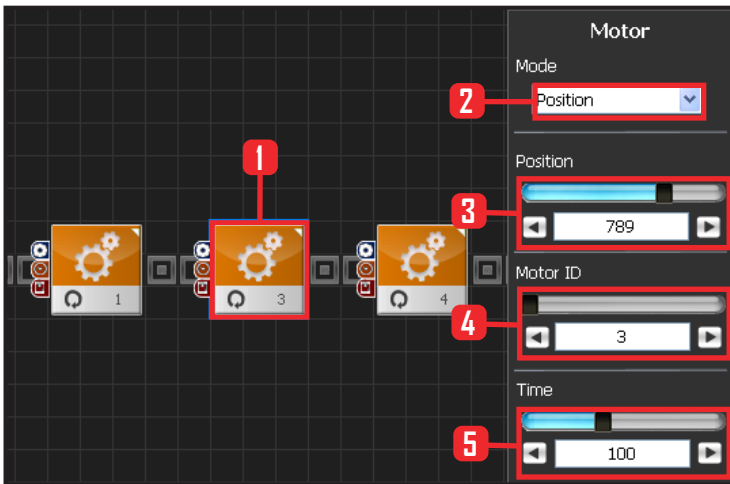
차려자세(기본자세) 만들기  
 모든 로봇의 모터의 각도를 중앙으로 정렬하면 휴머노이드에서는 팔을 좌우로 뻗게됩니다. 이것을 차려 자세로 되돌려 놓아야만 기본 자세를 유지하여 동작 시키기가 용이해집니다.

Motion > Moter 를 선택합니다.  
 Mode : Position 으로 선택합니다.  
 Position : 235 로 설정합니다. 235 는 수평으로 들고 있던 오른손을 수직으로 내려 갈 수 있게 모터를 돌리게 됩니다.  
 Motor ID : 0 으로 설정합니다. 오른쪽 어깨 모터 ID 가 0번입니다.  
 Time : 100 으로 설정합니다. 약 1.12초동안 원하는 각도로 이동합니다.



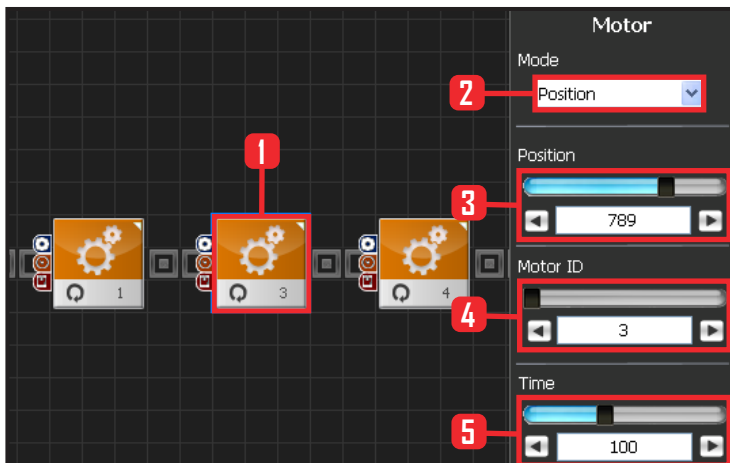
### 10 모터 1번(오른쪽 팔) 설정

Mode : Position 으로 선택합니다.  
 Position : 235 로 선택합니다. 235는 수평으로 되어 있던 팔을 수직으로 내리게 됩니다.  
 Motor ID : 1 로 설정합니다. 오른쪽 위쪽 팔 어깨와 닿는 모터가 1번 모터입니다.  
 Time : 100 으로 설정합니다. 약 1.12초동안 원하는 각도로 이동합니다.



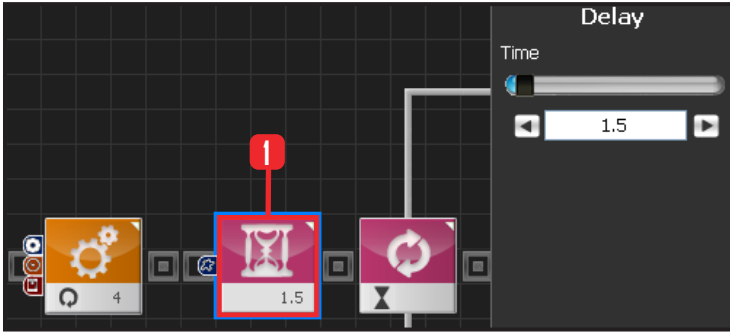
### 11 모터 3번(왼쪽 어깨) 설정

Mode : Position 으로 선택합니다.  
 Position : 789 로 설정합니다. 789 는 수평으로 들고 있던 왼손을 수직으로 내려 갈 수 있게 모터를 돌리게 됩니다.  
 Motor ID : 3 으로 설정합니다. 왼쪽 어깨 모터 ID가 3번입니다.  
 Time : 100 으로 설정합니다. 약 1.12초동안 원하는 각도로 이동합니다.



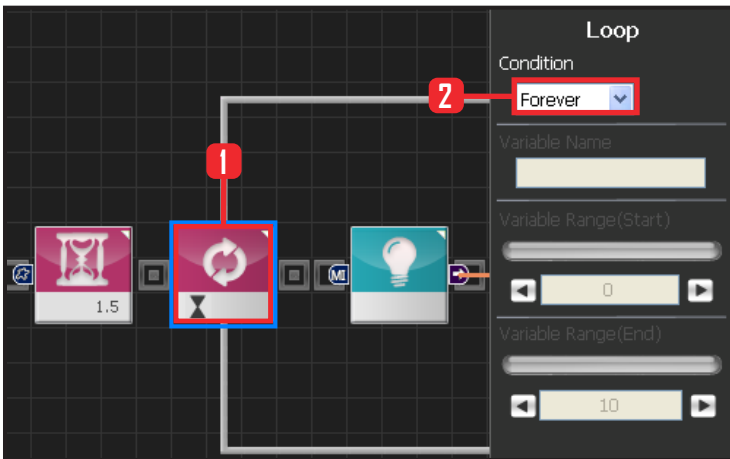
### 12 모터 4번(왼쪽 팔) 설정

Mode : Position 으로 선택합니다.  
 Position : 789 로 선택합니다. 789 는 수평으로 되어 있던 팔을 수직으로 내리게 됩니다.  
 Motor ID : 4 로 설정합니다. 왼쪽 위쪽팔 어깨와 닿는 모터가 4번 모터입니다.  
 Time : 100 으로 설정합니다. 약 1.12초동안 원하는 각도로 이동합니다.



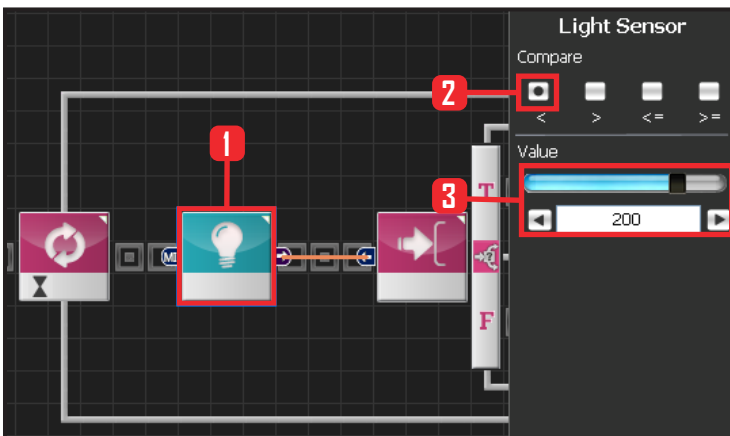
### 13 Delay 지연시키기

로봇이 차려자세가 되기까지 기다린 후에 다음 모듈을 실행시키기 위해 잠시 지연하기 위한 과정입니다. Flow > Delay 모듈을 선택합니다. Time : 1.5 로 설정합니다. 여기에서 단위는 초입니다. 약 1.5초동안 Delay 시킵니다.



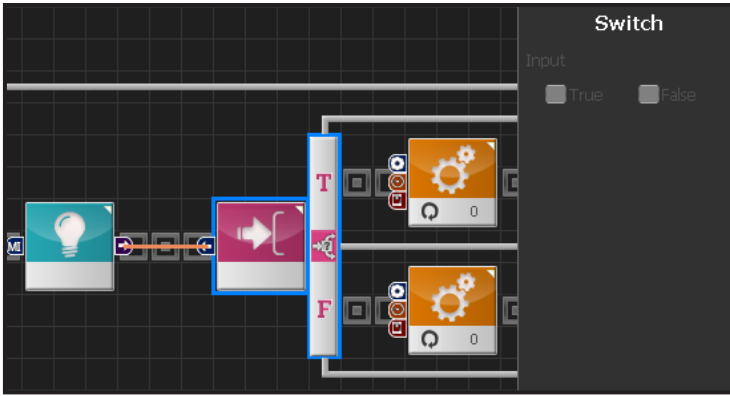
### 14 Loop 반복문

Flow > Loop 모듈을 선택합니다. Condition 은 Forever 를 선택합니다. 무한루프로 돌립니다.



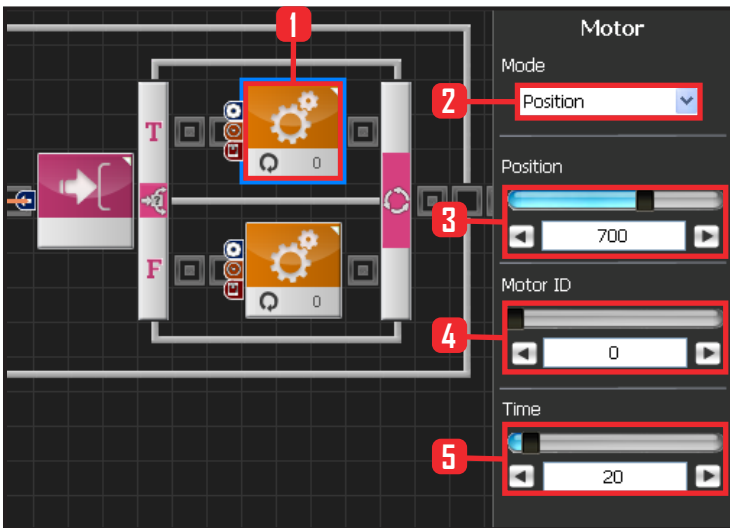
### 15 Light Sensor 빛 센서

Sensor > Light 모듈을 선택합니다. Compare : < 를 선택합니다. 어떤 값보다 작았을 때를 의미합니다. Value : 200 으로 설정합니다. 광도값이 200 임을 의미합니다. 광도값이 200 보다 작으면 모듈의 output이 True 가 되고, 크거나 같으면 False가 됩니다.



## 16 Switch IF 분기문

앞의 값이 True 이거나 False 일때 각각에 해당하는 결과를 실행합니다.



## 17 모터 0번(어깨) 설정

빛광도가 200보다 낮으면, 즉 어두우면(True) 오른쪽 팔을 올리게 되고, 환해지면(False) 오른쪽 팔이 내려진 상태로 그대로 유지됩니다.

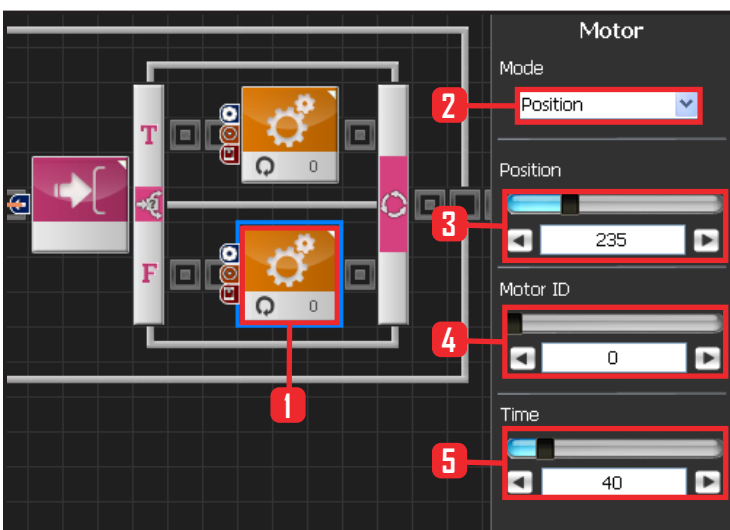
Motion > Moter 를 선택합니다.

Mode : Position 으로 선택합니다.

Position : 700 으로 설정합니다. 700 은 수직 차려자 세로 놓여있는 팔을 앞으로 올리게 됩니다.

Motor ID : 0 으로 설정합니다. 오른쪽 어깨 모터 ID 가 0번입니다.

Time : 20 으로 설정합니다.



## 18 모터 0번(어깨) 설정

제어기 cds 센서에서 손가락을 떼면, 다시 환해지므로, 모터는 차려자세로 돌아갑니다.

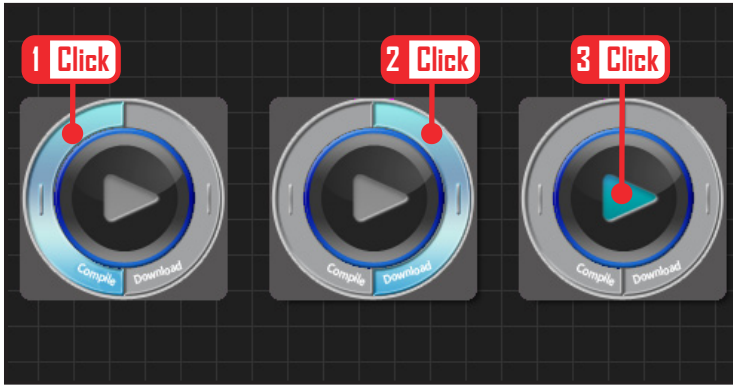
Motion > Moter 를 선택합니다.

Mode : Position 으로 선택합니다.

Position : 235 으로 설정합니다. 235 는 수직 차려자 세로 돌아가게 합니다.

Motor ID : 0 으로 설정합니다. 오른쪽 어깨 모터 ID 가 0번입니다.

Time : 40 으로 설정합니다. 올라간 속도보다, 좀 더 느리고 내려옵니다.



## 18 다운로드

프로그래밍 후 컴파일 - 로봇에 다운로드 - 실행 하는 과정을 거칩니다.

Compile을 클릭합니다. 에러가 없으면, 우측 Download 를 클릭합니다. 로봇에 다운로드 합니다. 다운로드 완료된 후, 실행버튼(중간 화살표)을 클릭 합니다.

로봇에서 외부 입력을 기다리게 됩니다.

## 19 로봇동작

현재는 빛이 밝은 상태이므로 차려자세를 유지합니다. 손가락으로 cds 를 가리면, 로봇이 오른팔을 듭니다. 손가락을 떼면 로봇이 오른팔을 내립니다. 정확히 실행되는 것을 확인합니다.

