|  |
| --- |
|  |
| 서보 모터 실습 |
|  |
| 지능형 인터페이스 |

**교수 : 김상연**

3조

조장 : 김영규

팀원 : 이재영

팀원 : 여창민

서보 모터 실습

지능형 인터페이스

1. 과제 설명
2. 스크립트 코드(주석 포함 및 소스 코드 설명)
3. 고찰
4. 구동 동영상 별첨
5. 과제 설명

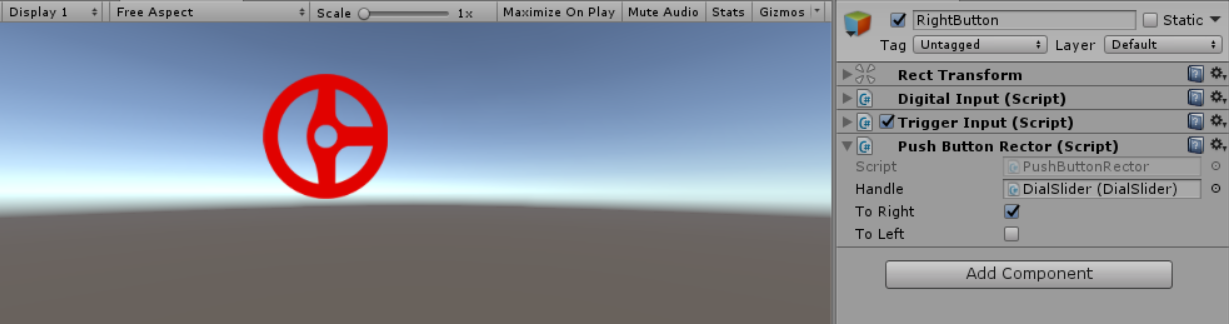
유니티에서 핸들 오브젝트를 움직이면 서보모터가 회전하는 실습을 바탕으로 푸쉬버튼으로 핸들과 서보모터를 회전하는 실습이다. 왼쪽 푸쉬버튼을 누르면 서보모터와 핸들 오브젝트가 왼쪽으로 회전하고, 오른쪽 푸쉬버튼을 누르면 핸들 오브젝트와 서보모터가 오른쪽으로 회전한다..

스크립트 파일 PushButtonReactor를 수정하였다. gameObject객체인handle을 DialSlider로 바꾸어 푸쉬버튼을 누르면 handle의 angle 값이 수정되도록 구현하였다.

1. 스크립트 코드

(PushButtonRector 클래스)

|  |
| --- |
| public class PushButtonRector : ArdunityReactor {  private IWireInput<Trigger> \_triggerInput;  public DialSlider handle; //이동시킬 오브젝트  …  //오브젝트 이동 함수  private void MoveObject(Trigger value)  {  if (value.value)  {  // ToLeft 또는 ToGight 값에 따라 왼쪽 또는 오른쪽 회전  if (ToLeft)  handle.angle += 20;  else if (ToRight)  handle.angle -= 20;  else  return;  }  }  … |



DialSlider클래스로 변경한 Handle에 다시 DialSlider을 연결하여 준다. 푸쉬버튼이 눌리면 handle의 angle 값을 수정한다.

1. 고찰

세 번째 과제를 마쳤다. 이번 과제는 현실에서 스위치로 입력을 하면 컴퓨터 화면에서도 값이 변화하고, 아우이노에 연결된 모터에도 값이 변화한다. 단순 두 기기끼리 값을 주고 받는 수준을 넘어서 그에 연결된 기기까지 값을 변화시켜야 했기에 난이도가 높은 과제였다.

하나하나 찾아가고 배워가며 발전을 이루고 있다. 모르는 것을 하나하나 배울 때 마다 진전의 즐거움이 느껴진다.