

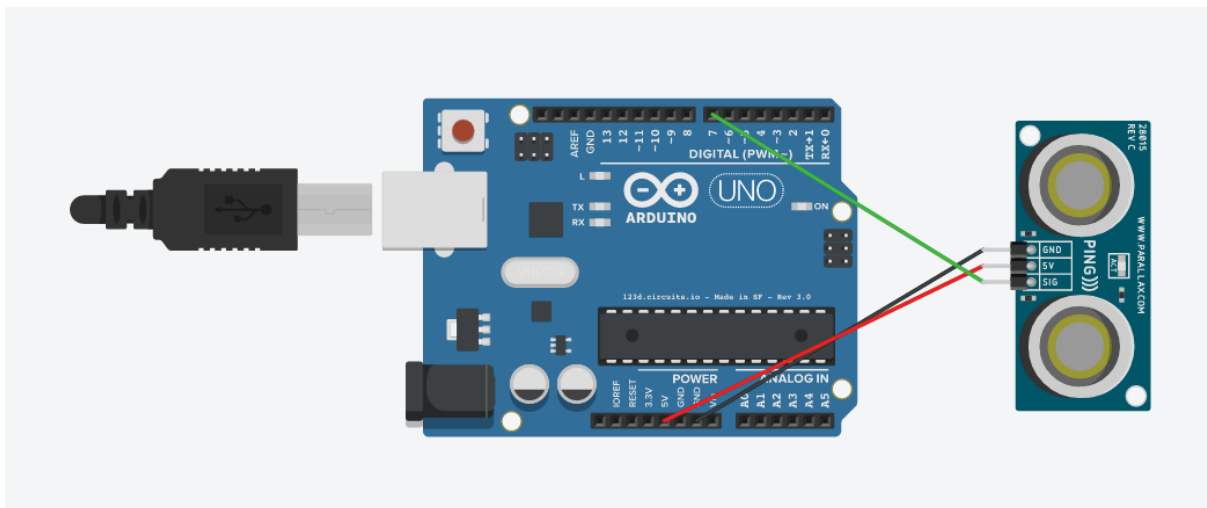
# Design Document

For the AR/VR project I have decided to design a Virtual Reality interface that uses the HTC Vive and the Vives hand controllers. I was also planning on integrating an Arduino into this interface, however due to various reasons this wasn't possible. However in this document I will outline how I was planning on setting the Arduino and Vive up to work with the game.

The game that I was planning on integrating this interface into did not have a playable build that I could work with. However I did make my own test environment to test out the Vive and Arduino in.

The idea was for the player to be holding two objects in their hands, which are tracked by the vive controllers, then when the player brings their hands closer together the weapons that the player is holding switch.

Below is how I would have planned to wire up the Arduino with the distance sensor.



This sensor would be attached to the side of a glove, along with the Arduino. Furthermore in the final version of this product, the Arduino would be powered by a small battery back also attached to the glove and the Arduino would connect to the PC via Bluetooth.

In the game the player controls two items that are attached to each of the vive controllers, in future iterations these items would vary in functionality.

To get the Vive controllers working within Unity, I downloaded the “Steam VR” plugin for Unity and dragged the “CameraRig” Prefab into my scene, then created some cubes to act as the weapons.

Here is the code for the weapons:

```
using UnityEngine;
using System.Collections;

public class Weapon : MonoBehaviour {

    public GameObject cube;
    public Rigidbody rb;

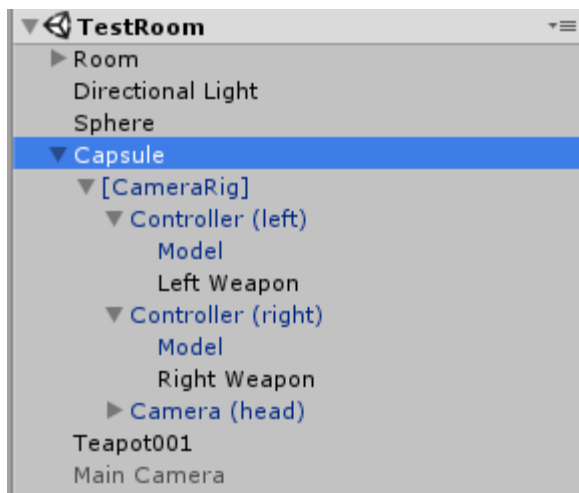
    // Use this for initialization
    void Start()
    {
        rb = this.cube.GetComponent<Rigidbody>();
    }

    void OnTriggerEnter(Collider other)
    {
        if(other.tag == "Sphere")
            Destroy(other.gameObject);
    }

    // Update is called once per frame
    void Update()
    {
        //rb.AddForce(0, 5, 0);
    }
}
```

This code makes the sphere in the scene be deleted if either of the weapons collides with it.

Here is the hierarchy of the scene.



Here is an image of the Vive controllers working in the game engine:

