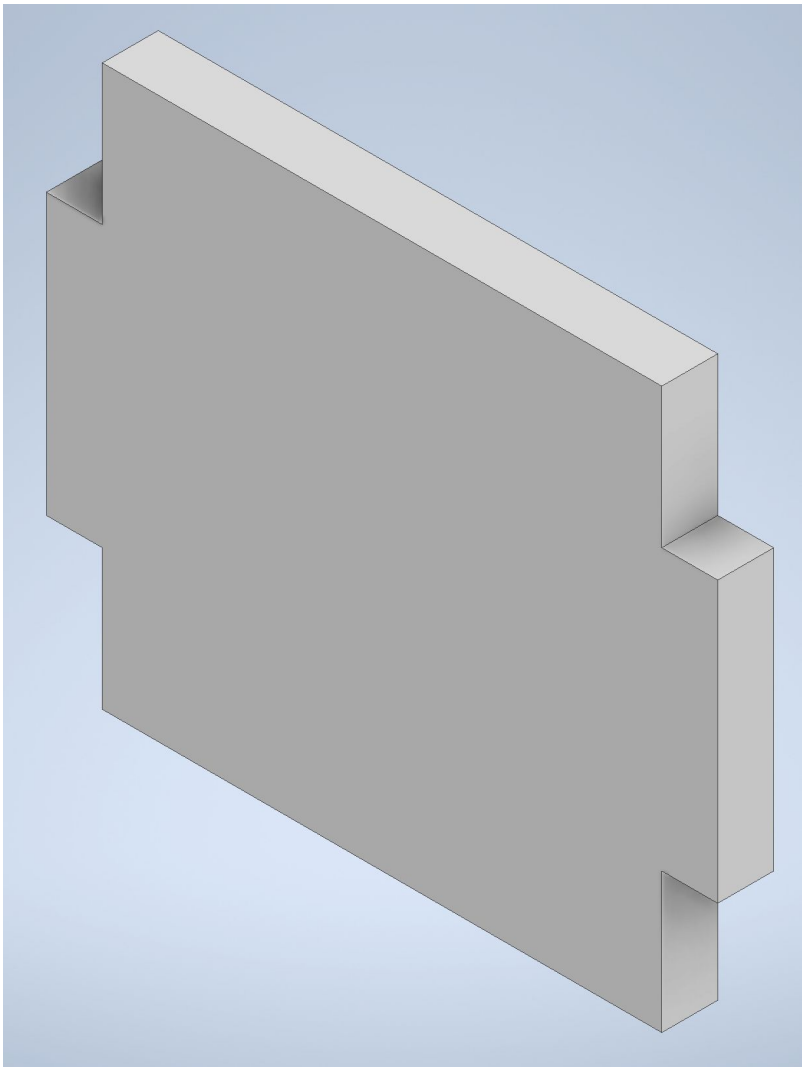
House Side Wall

Here is the CAD file for the side wall of the houses.



House Side Wall Part File

Cody Jones

1/19/24

There will be two of this part in the house assembly, one for each side wall.

The walls connect to each other via finger joints.

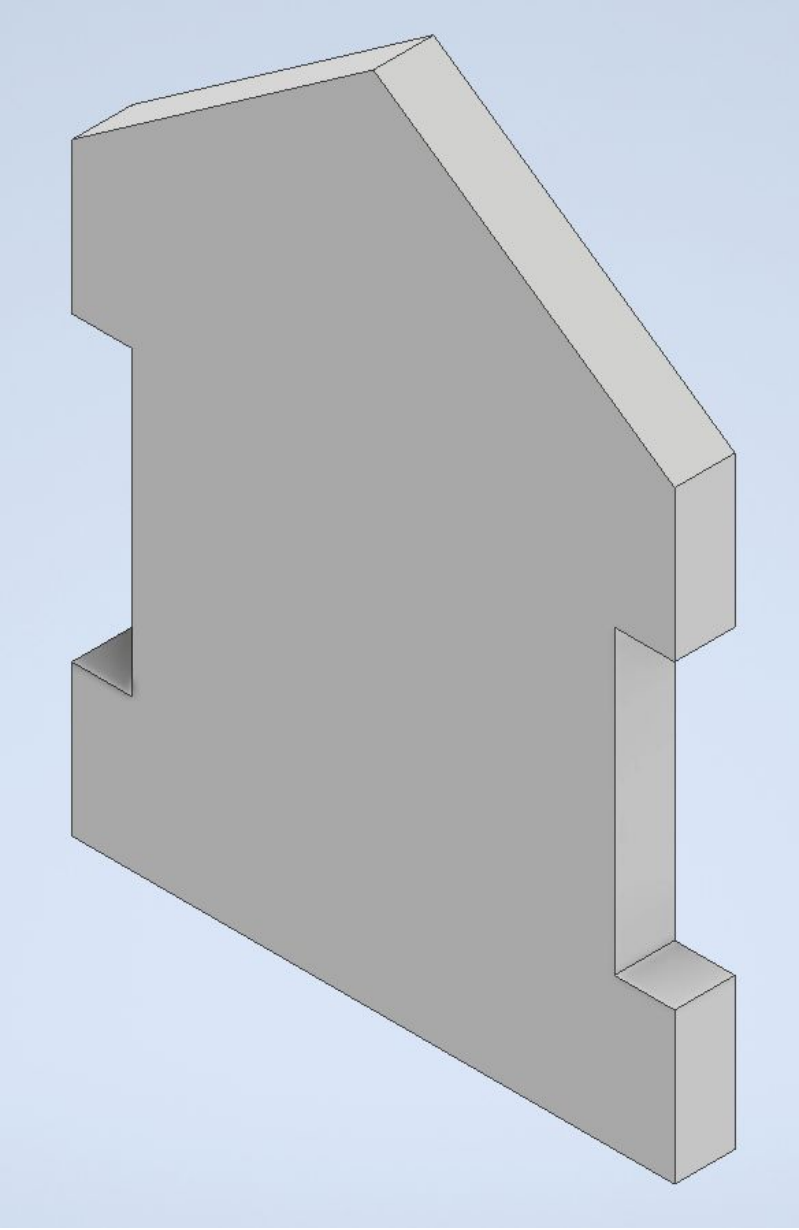
We plan on 3D printing the final house assembly.

Signature: Cody Jones Date: 1/19/24

Witness: Ayden Loven Date: 1/19/24

House Front and Back Wall

Here is the CAD file for the front and back walls of the houses.



House Front and Back Wall Part File

Cody Jones

1/19/24

There will be two of this part in the house assembly, one for the front, and one for the back.

The walks connect to each other via finger joints.

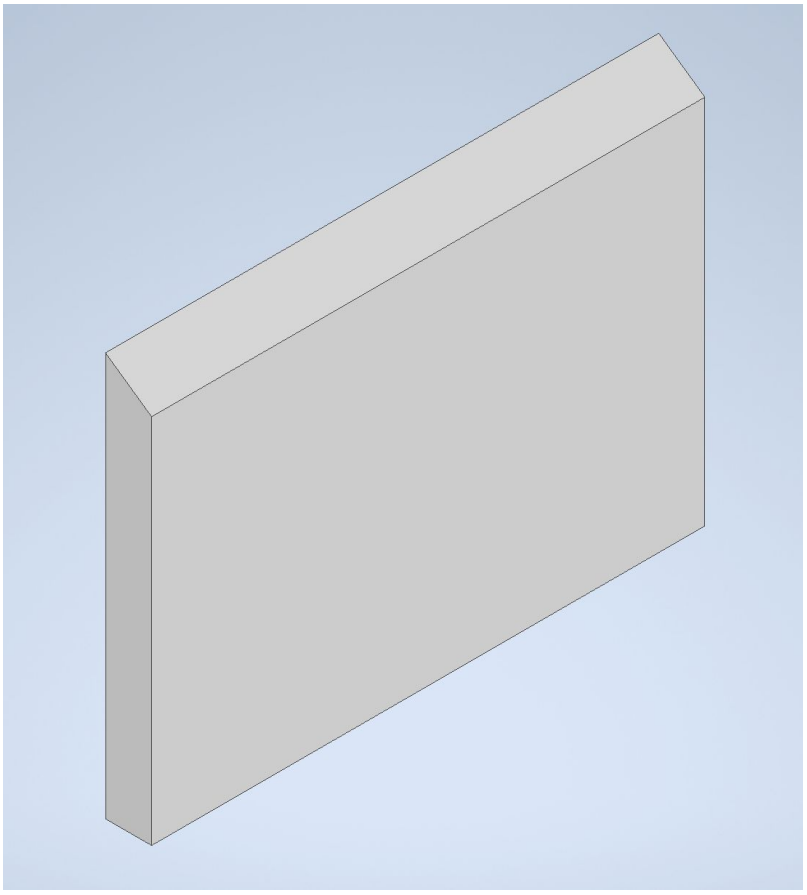
We plan on 3D printing the final house assembly.

Signature: Cody Jones Date: 1/19/24

Witness: Maria Vanderputten Date: 1/19/24

House Roof

Here is the CAD file for the roof of the houses.



House Roof Part File

Cody Jones

1/19/24

There will be two of this part in the house assembly, one for each half of the roof.

The angled edges will connect via miter joint to each other, and they will connect to the tops of the walls via butt joint.

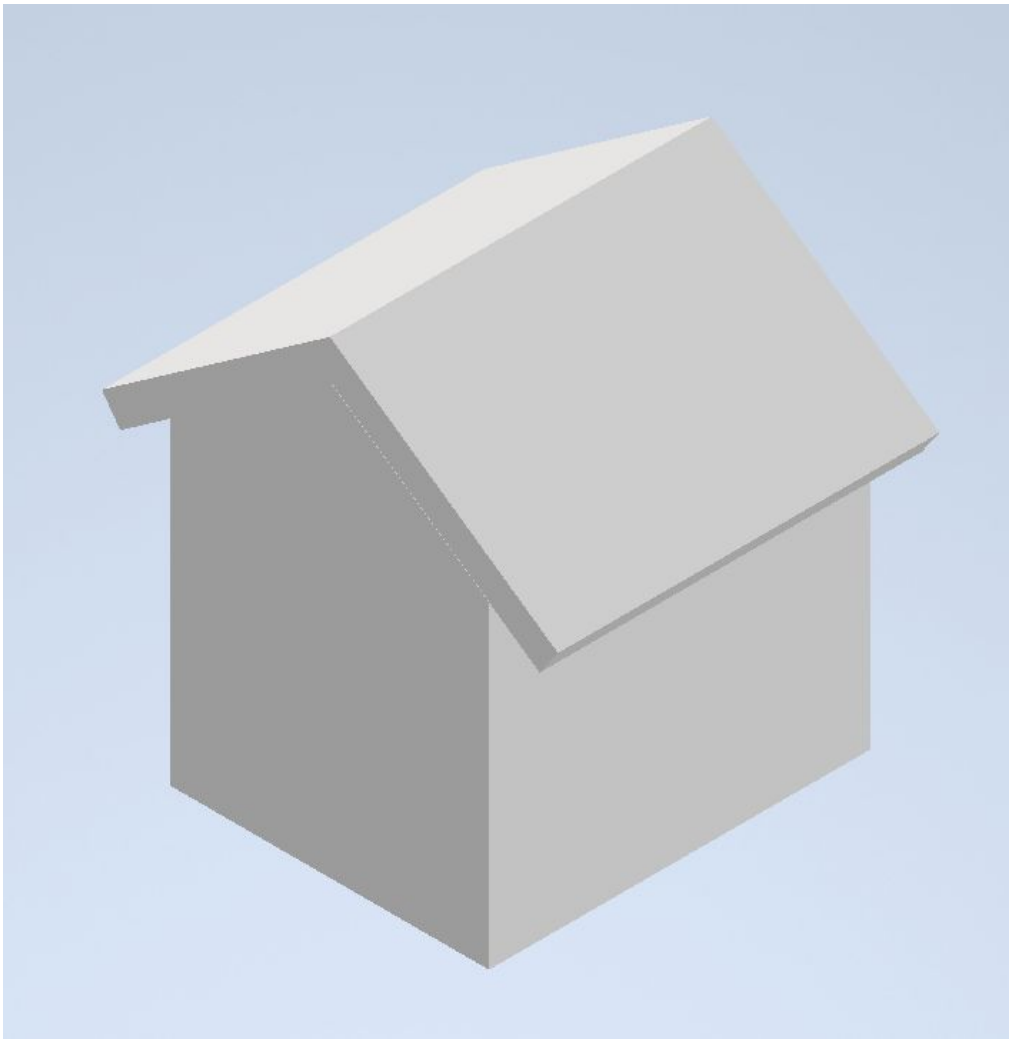
We plan on 3D printing the final house assembly.

Signature: Cody Jones Date: 1/19/24

Witness: Khylee Marshall Date: 1/19/24

Completed House Assembly

Here is the completed house CAD assembly file for the SEE display project.



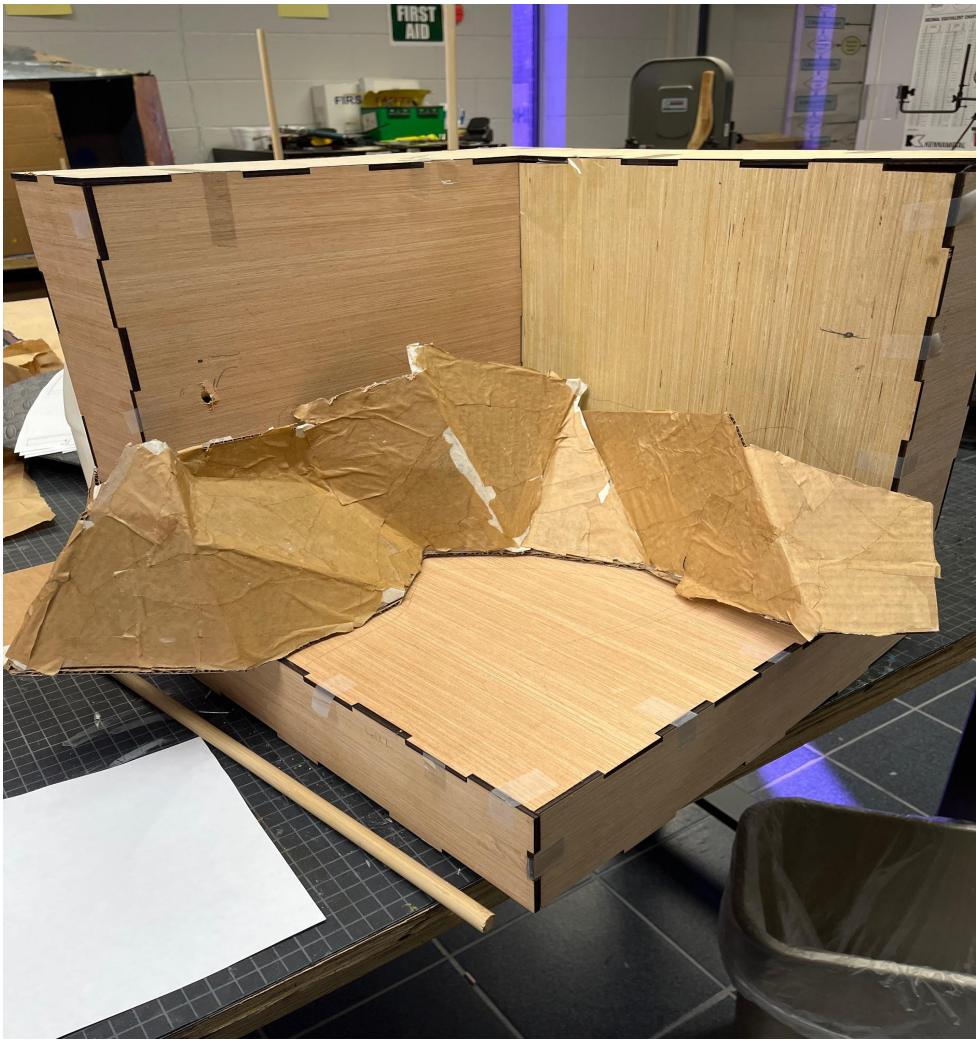
SEE House Assembly - Cody Jones - 1/19/24

Signature: Cody Jones Date: 1/19/24

Witness: Khylee Marshall Date: 1/19/24

Image of Display and Mountain

After finishing building our display and mountain, we put them together. THis is what they look like together.



SEE Display Image - Cody Jones - 1/22/24

Signature: Cody Jones Date: 1/22/24

Witness: Khylee Marshall Date: 1/22/24

Mini Fair at Gordon Cooper

These are some of the things that Mrs. Crouch and the seniors recommended for our SEE project.

Mrs. Crouch said that we should work on the ‘look’ of our display, such as the mountain and background.

Therron recommended that we simplify the coding that we have because it looked repetitive. Maybe we could use some functions to replace the repetitive sections. He also suggested that we add comments throughout our code so it is easily understandable

We also need to develop our coding so we have some working components!

We also need to make sure that we get our currently working components (lights, motor) connected to the display.

Image of our display of the Mini Fair Day - Cody Jones - 1/30/24

Signature: Cody Jones Date: 1/30/24

Witness: Khylee Marshall Date: 1/30/24

Update on Display

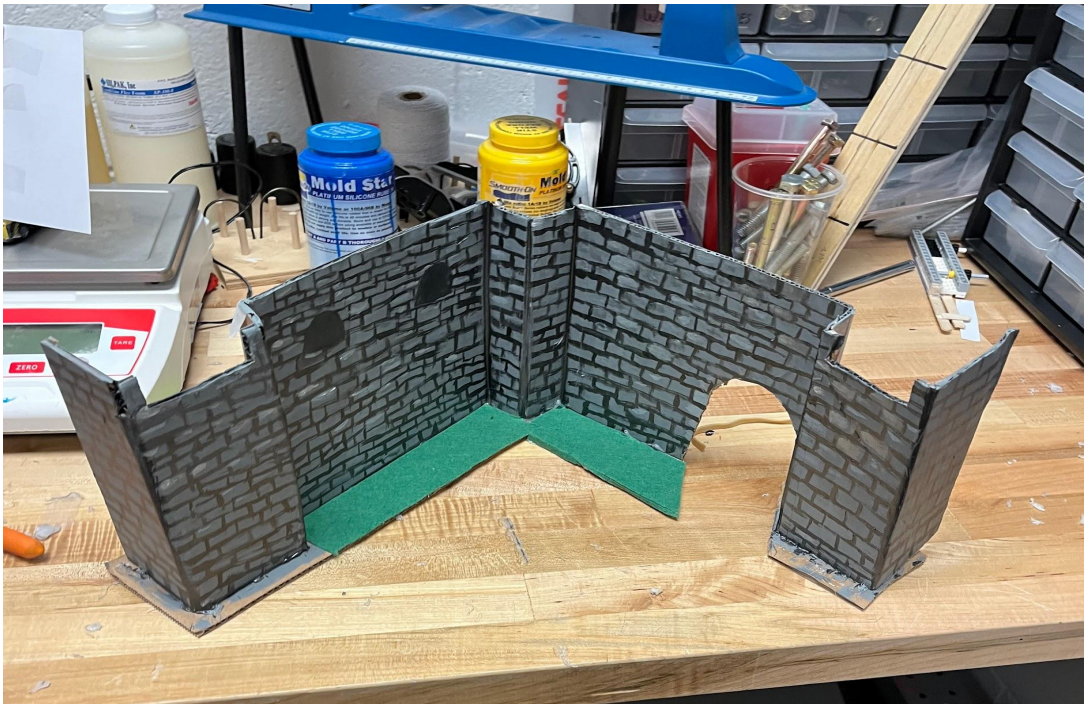
Here is an update on the background of our display.

Since the mountain took up so much unnecessary space, we decided to replace it with castle walls. Since we will have a catapult following out from the background, it makes more sense to have it rolling out from castle walls than from a mountain.

Cody assembled the castle walls, and Khylee painted and decorated them.

Image of Castle Walls - Khylee Marshall

2/13/24



Signature: Khylee Marshall Date: 2/13/24

Witness: Cody Jones Date: 2/13/24

Motor to LEGO Technic Converter

We want to create a moving piece using LEGO Technic parts. Here are the inspiration photos and planned measurements for our converter.

Width of Entire Axle Slot - 4.72 mm

- Converts to 0.1858

Width of Axle Spline - 1.83 mm

- Converts to 0.720

With all of these measurements, we will round up a little bit to accommodate for any inaccuracies with printing.

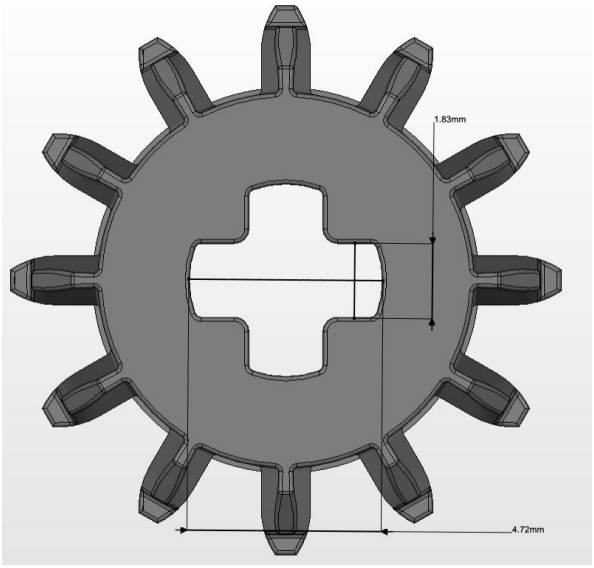
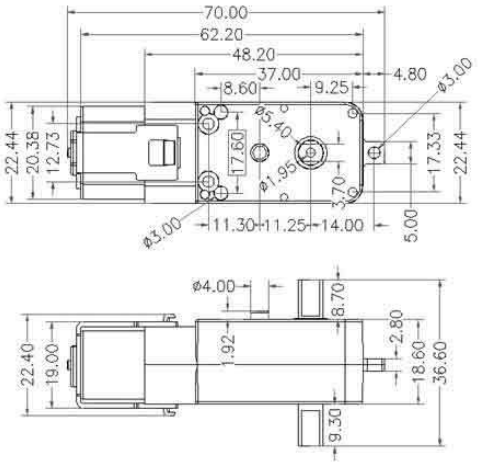


Diagram of LEGO Axle Hole in Gear

Cody Jones - 1/30/24

Diagram of DC Motor - Cody Jones - 1/30/24



Diameter of Rotor Bar - 5.40 mm

- Converts to 0.2125 in

Distance between Rotor Bar Flat Edges - 3.70 mm

- Converts to 0.1457 in

Signature: Cody Jones Date: 1/30/24

Witness: Khylee Marshall Date: 1/30/24

Motor to LEGO Technic Converter (continued)

We want to create a moving piece using LEGO Technic parts. Here is the CAD file and real image for our converter.

This adapter will connect to an axle, which will turn a gear for a LEGO rack, which will move a platform for the catapult to sit on.

Dimensions

Axle Slot Width - 0.200 in

Spline Slot Height - 0.055 in

Spline Slot Width - 0.090 in

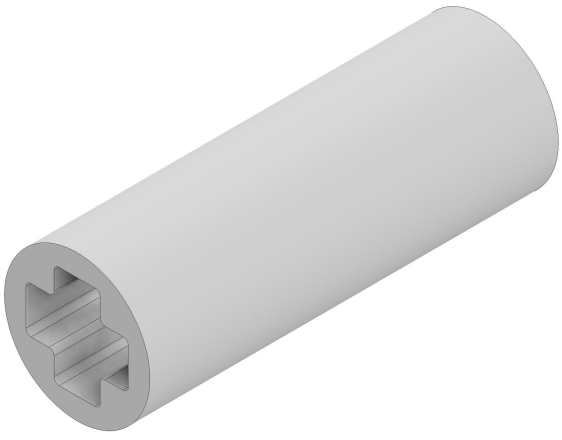
Adapter Length - 0.800 in

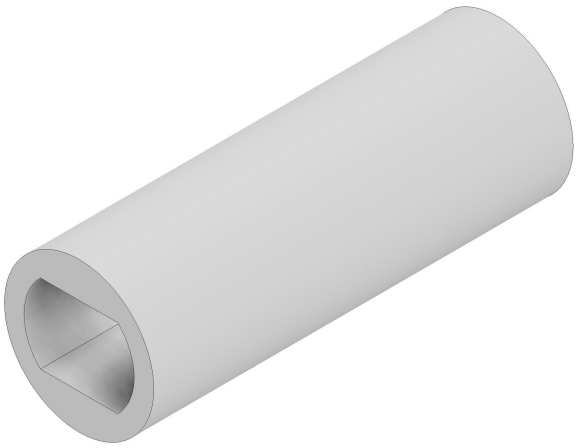
DC Slot Inner Diameter - 0.105 in

DC Slot Width - 0.150 in

Outer Diameter - 0.285

Adapter Part File - Cody Jones - 1/30/24





Signature: Cody Jones Date: 1/30/24

Witness: Khylee Marshall Date: 1/30/24

Coding

S.O.L.I.D. Principle

For our coding, we had to learn about S.O.L.I.D. principles. What are they?

S: Single Responsibility Principle

- This means that each class should be responsible for a single part or functionality of the system.

O: Open-Closed Principle

- Software components should be open for extension, but not for modification.

L: Liskov Substitution Principle

- Objects of a superclass should be replaceable with objects of its subclass without breaking the system.

I: Interface Segregation Principle

- No client should be forced to depend on methods that it does not use.

D: Dependency Inversion Principle

- High-level modules should not depend on low-level modules. Both should depend on abstractions.

S.O.L.I.D. helps save time, effort, make code more modular, easy to maintain, understand, debug, and refactor. It also helps build long lasting programs and software to maintain hard work

Signature: Khylee Marshall Date: 12/4/23

Witness: Cody Jones Date: 12/4/23

Notes (Raspberry Pi Pico Lessons)

These are the materials required for the Raspberry Pi Pico Lessons.

What is needed for the Lessons:

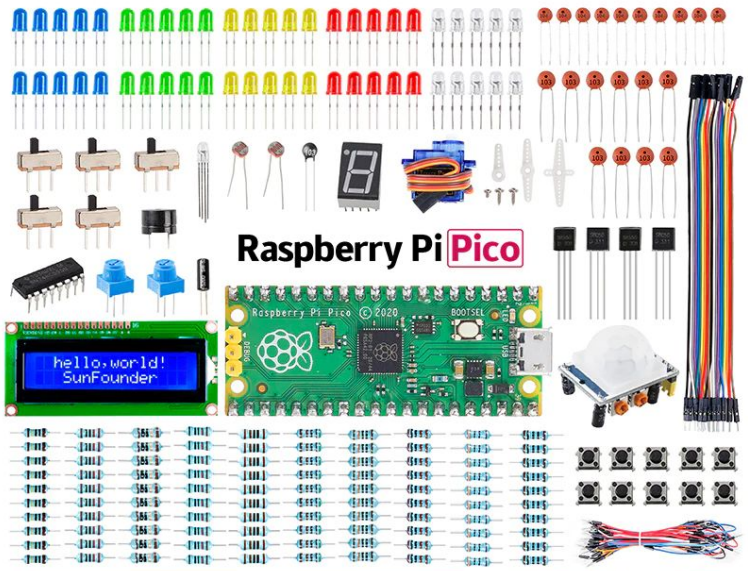
- Micro USB cables
  - USB connected to computer

❖ Micropython
- Thonny (Python for Beginners)
  - Will possibly throw warning, scroll down to app rec setting & select from anywhere.
  - Blue icon click on and menus will pop up

❖ Use this website for motor control directions!
- https://www.donskytech.com/raspberry-pi-pico-motor-control-using-the-drv8833/

Online Picture of Pico Kit Parts

Khylee Marshall - 12/4/23



Signature: Khylee Marshall Date: 12/4/23

Witness: Cody Jones Date: 12/4/23

Raspberry Pi Pico Lesson 1 Notes

Here are Khylee's notes from the first episode on the Raspberry Pi Pico.

Lesson Summary

- Configure Pico
- Set up comp
- Write first 4 programs

- Using USB, you can connect your PC to the Pico
- You have to make the Pico recognizable by holding down the white button
- (Select Pico W)
- It won't show ups as a USB once micropython is downloaded.

- GPIO (General-Purpose Input-Output) Pins

0 = OFF

1 = ON

Write First Line of Code!

- print("Hello World")

Signature: Khylee Marshall Date: 12/4/23

Witness: Cody Jones Date: 12/4/23

Raspberry Pi Pico Lesson 2 Notes

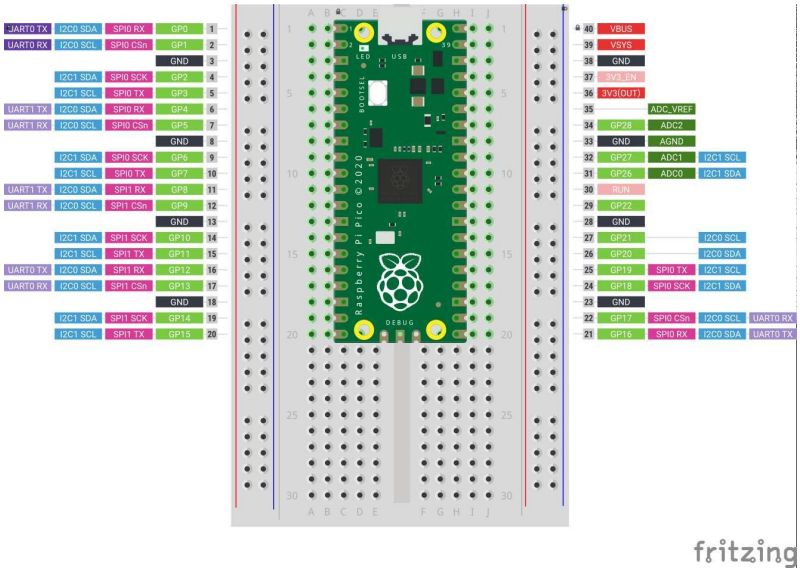
Here are Khylee's notes from the second episode on the Raspberry Pi Pico.

Understanding and Using Breadboards

- The long leg of an LED is the positive part of the LED.
- In order to not fry you need to add a current limiting resistor in front of it so you can use it as many times as you want and it shouldn't die.
- Pins are 1 through 40
- Short leg goes to the negative part of the LED

- Picture of breadboard
- https://scruss.com/wordpress/wp-content/uploads/2021/02/Pico-breadboard_bb.jpg

- Reference the link to understand next things, which means columns will be horizontal and rows vertical
- Holes along continuous columns are connected
- Holes across rows and along non-continuous columns (separated by the trench) are not connected.



Signature: Khylee Marshall Date: 12/5/23

Witness: Cody Jones Date: 12/5/23

Raspberry Pi Pico Lesson 3 Notes

Here are Khylee's notes from the third episode on the Raspberry Pi Pico.

Understanding and Using Binary Numbers

- Everything is based on zeroes and ones

0 0 0 0 : 0

0 0 0 1 : 1

0 0 1 0 : 2

0 0 1 1 : 3

0 1 0 0 : 4

0 1 0 1 : 5

0 1 1 0 : 6

0 1 1 1 : 7

1 0 0 0 : 8

1 0 0 1 : 9

1 0 1 0 : 10

1 0 1 1 : 11

1 1 0 0 : 12

1 1 0 1 : 13

1 1 1 0 : 14

1 1 1 1 : 15

Signature: Khylee Marshall Date: 12/5/23

Witness: Cody Jones Date: 12/5/23

Dragon Brainstorming

Here are our ideas for how our dragon will look and work.

- We should have the dragon on a wire that moves across the top of the box and stop at a knot.
- Since the dragon will have to have large wings, we need to clear out a large space for its path through the mountain.
- This will be the number 1 priority and probably the hardest thing to program!

Signature: Khylee Marshall	Date: 1/12/24
Witness: Cody Jones	Date: 1/12/24

Notes on Wires

This page is about the types of wires we learned about and ordered, as well as our plans.

Wire 1: Male Wire

- This wire end connects into the breadboard and goes into the female wire.

Wire 2: Female Wire

- Some wires have two male ends, two female ends, or one of each.
- For our SEE project, we will use both types of wire.

- Our breadboard needs to go on the back of the display.
- We need to wire the speaker, and it will go through hole in the bottom of the display.
- Wires will need to go through holes in all interior sides of the display.
- We need to order extender wires (wires with female ends) because some lights will be far away from the breadboard.

Signature: Khylee Marshall Date: 1/12/24

Witness: Cody Jones Date: 1/12/24

Wiring Placements

Here are our ideas for placing our electrical components on our SEE Project.

Wire 1: Male Wire

- This wire end connects into the breadboard and goes into the female wire.

Wire 2: Female Wire

- Some wires have two male ends, two female ends, or one of each.
- For our SEE project, we will use both types of wire.

- Our breadboard needs to go on the back of the display.
- We need to wire the speaker, and it will go through hole in the bottom of the display.
- Wires will need to go through holes in all interior sides of the display.
- We need to order extender wires (wires with female ends) because some lights will be far away from the breadboard.

- The motors will have to go on either ends of the box to work both things they need to run: the dragon and catapult mover.
- The house fires will be LED strips over and around the houses
- The speaker will be wired under the box hopefully.
- The catapult LEDs will sit in the castle walls.

Signature: Khylee Marshall Date: 1/19/24

Witness: Cody Jones Date: 1/19/24

Wiring Placements

Here are our ideas for placing our electrical components on our SEE Project.

- First coded the catapult LEDs using adjusted code resources from Paul McWhorter's Youtube series.
- Instead of having the motor run on a continuous loop, we decided to time it with our audio, which should be completed in the weeks before competition.
- Khylee and Gage still need to figure out how to hook up some kind of speaker system as well as some kind of way for our buttons to start, stop, and pause the show.

Signature: Khylee Marshall	Date: 2/10/24
Witness: Cody Jones	Date: 2/10/24

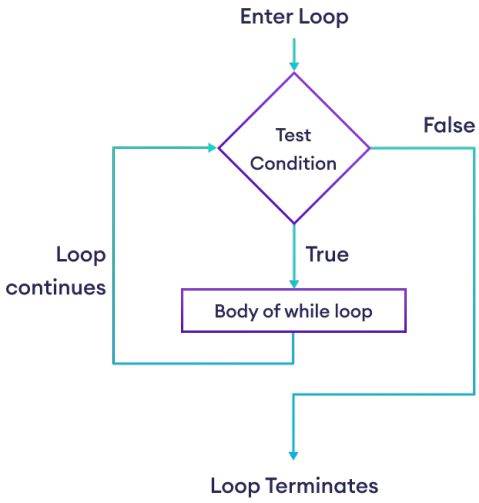
How to Make a Pause Button

Here is Cody's research on how to make a button that will pause and unpause the show.

- For our pause/unpause button, we have a few possibilities for what we can do.
- For any situation, we would need a variable called pausePlay
- We could use an if statement within a whileTrue function to test the value of the pausePlay button (TRUE/FALSE) to run a sleep function.
- I am not quite sure if micropython has an event handler, but if it does, we could have code that runs specifically when the pausePlay button is pressed.

Example of how a while loop works

Cody Jones - 2/19/24



Signature: Cody Jones Date: 2/19/24

Witness: Khylee Marshall Date: 2/19/24

How to Make a Pause Button

Here is Gage's research on how to make our sound work through speakers and amplifiers.

VCC– Power 5-12 V

IN– input

GND– ground

GND– ground

Image of how speaker and amplifier wires to breadboard - Gage Danielwicz

2/19/24

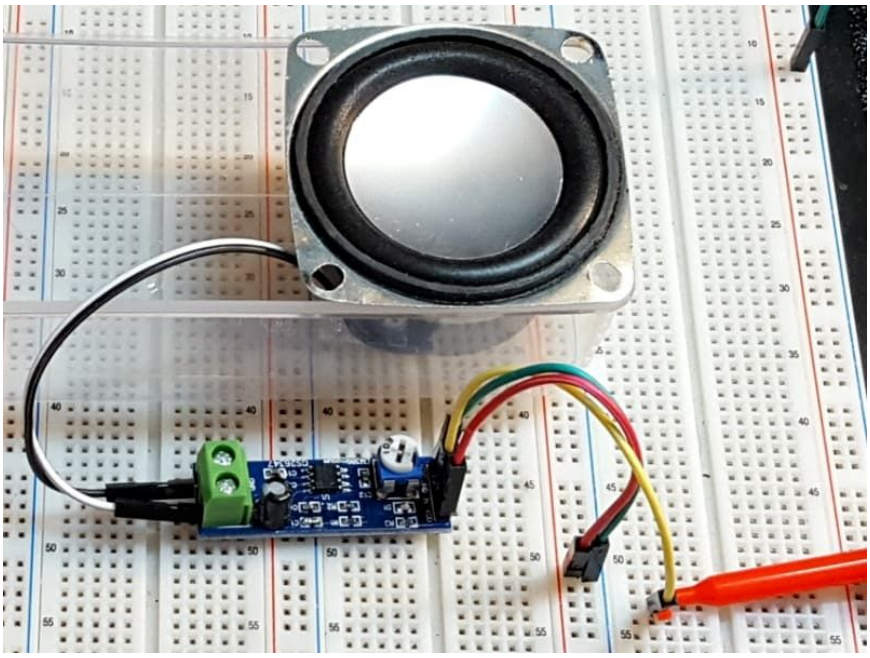


Image of Speaker - Gage Danielwicz
2/19/24

Signature: Gage Danielwicz Date: 2/19/24

Witness: Khylee Marshall Date: 2/19/24

Example Code

Here are some examples from our code to show how our project actually functions.

```
from machine import Pin
import time

# Define pin for Bakery LED
b_led_pin = [Pin(15, Pin.OUT)]

m_led_pin = [Pin(28, Pin.OUT)]

# Define pins for LEDs
led_pins = [Pin(18, Pin.OUT), Pin(16, Pin.OUT), Pin(19, Pin.OUT), Pin(20, Pin.OUT)]

button_pin = Pin(22, Pin.IN, Pin.PULL_UP)

pausebut_pin = Pin(26, Pin.IN, Pin.PULL_UP)

# Define pins for Fire
f_led_pin1 = [Pin(9, Pin.OUT)]
f_led_pin2 = [Pin(8, Pin.OUT)]
f_led_pin3 = [Pin(7, Pin.OUT)]
f_led_pin4 = [Pin(6, Pin.OUT)]
f_led_pin5 = [Pin(5, Pin.OUT)]

# Derfine pins for Motors
motor1a=Pin(12,Pin.OUT)
motor1b=Pin(13,Pin.OUT)
motor1c=Pin(10,Pin.OUT)

# Function to turn on a specific LED
def turn_on_led(led_pin):
    led_pin.value(1)

#Function to turn off a specific LED
def turn_off_led(led_pin):
    led_pin.value(0)

# Function to turn on Bakery LED
def run_b_sequence():
    for led_pin in b_led_pin:
        turn_on_led(led_pin)

# Function to turn on Musican LED
def run_m_sequence():
    for led_pin in m_led_pin:
        turn_on_led(led_pin)
```

```
1  from machine import Pin
2  import time
3
4  # Define pin for Bakery LED
5  b_led_pin = [Pin(15, Pin.OUT)]
6
7  m_led_pin = [Pin(28, Pin.OUT)]
8
9  # Define pins for LEDs
10 led_pins = [Pin(18, Pin.OUT), Pin(16, Pin.OUT), Pin(19, Pin.OUT), Pin(20, Pin.OUT)]
11
12 button_pin = Pin(22, Pin.IN, Pin.PULL_UP)
13
14 pausebut_pin = Pin(26, Pin.IN, Pin.PULL_UP)
15
16 # Define pins for Fire
17 f_led_pin1 = [Pin(9, Pin.OUT)]
18 f_led_pin2 = [Pin(8, Pin.OUT)]
19 f_led_pin3 = [Pin(7, Pin.OUT)]
20 f_led_pin4 = [Pin(6, Pin.OUT)]
21 f_led_pin5 = [Pin(5, Pin.OUT)]
22
23 # Derfine pins for Motors
24 motor1a=Pin(12,Pin.OUT)
25 motor1b=Pin(13,Pin.OUT)
26 motor1c=Pin(10,Pin.OUT)
27
28 def pause_button
29
30
31 # Function to turn on a specific LED
32 def turn_on_led(led_pin):
33     led_pin.value(1)
34
35 #Function to turn off a specific LED
36 def turn_off_led(led_pin):
37     led_pin.value(0)
38
39 # Function to turn on Bakery LED
40 def run_b_sequence():
41     for led_pin in b_led_pin:
42         turn_on_led(led_pin)
43
```

Signature: Khylee Marshall Date: 2/19/24

Witness: Cody Jones Date: 2/19/24

Example Code

Here are some examples from our code to show how our project actually functions.

```
from machine import Pin#
from time import sleep#
LEDPin= 21#
myLED=Pin(LEDPin,Pin.OUT)#
butPin= 22#
myButton= Pin(butPin,Pin.IN,Pin.PULL_UP)#
butStateNow=1#
butStateOld=1#
LEDState=False#
while True:#
    butStateNow=myButton.value()#
    if butStateNow==1 and butStateOld==0:#
        LEDState= not LEDState#
        myLED.value(LEDState)#
    print(LEDState,butStateNow)#
    butStateOld=butStateNow#
    sleep(.1)#
```

```
# Function to turn on Musican LED
def run_m_sequence():
    for led_pin in m_led_pin:
        turn_on_led(led_pin)


def run_f_sequence():
    for led_pin in f_led_pin1:
        turn_on_led(led_pin)
    for led_pin in f_led_pin2:
        turn_on_led(led_pin)
    for led_pin in f_led_pin3:
       turn_on_led(led_pin)
    for led_pin in f_led_pin4:
        turn_on_led(led_pin)
    for led_pin in f_led_pin5:
        turn_on_led(led_pin)

def run_of_sequence():
    for led_pin in f_led_pin1:
        turn_off_led(led_pin)
    for led_pin in f_led_pin2:
        turn_off_led(led_pin)
    for led_pin in f_led_pin3:
        turn_off_led(led_pin)
    for led_pin in f_led_pin4:
        turn_off_led(led_pin)
    for led_pin in f_led_pin5:
        turn_off_led(led_pin)

#Function to turn off Bakery LED
def run_ob_sequence():
    for led_pin in b_led_pin:
        turn_off_led(led_pin)

#Function to turn off Bakery LED
def run_om_sequence():
    for led_pin in m_led_pin:
        turn_off_led(led_pin)

# Function to run the sequence
def run_sequence():
    for led_pin in led_pins:
        turn_on_led(led_pin)
        time.sleep(0.5) # Adjust delay time as needed
        turn_off_led(led_pin)
```

Signature: Khylee Marshall Date: 2/19/24

Witness: Cody Jones Date: 2/19/24

Receipts

Here are our expenses for this SEE project.

HOBBY LOBBY
Super Savings, Super Selection!
4838 Marketplace Blvd
Shawnee, OK 74804
Hobby Lobby Store #635 (405) 273-1930

S-635 R-1 T-9434 CASSIDY G SALE

105000000 Crafts 5.97
3 @ 1.99 ea
102500000 Toys&Hobbies 9.99
102500000 Toys&Hobbies 10.99

SUBTOTAL 25.95
TAX TOTAL 2.69
TOTAL 29.64

DEBIT 29.64
ACCOUNT #: **********1485
ACCT : Debit INSERTED

US DEBIT
CARD #**********1485 INSERTED
REF # AUTH # RESP 00
163101101042 ISO 00
AID: A0000000980840
TSI: 6800 ARC: CUR:0840
TVR: 8000048000
APP: US DEBIT
IAD: 06011203A0A000

Verified by PIN

HOBBY LOBBY
Super Savings, Super Selection!
4838 Marketplace Blvd
Shawnee, OK 74804
Hobby Lobby Store #635 (405) 273-1930

S-635 R-2 T-0306 LORA M SALE

102500000 Toys&Hobbies 3.98
2 @ 1.99 ea

SUBTOTAL 3.98
TAX TOTAL 0.40
TOTAL 4.38

CASH 20.00
CHANGE DUE 15.62

Number of Items Purchased: 2

Thank You. Please come again.
Become a fan on Facebook.
Sign up today for our Hobby Lobby App

Visit our website at www.hobbylobby.com

0635002003060106240

1/6/24 01:33 PM

Signature: Khylee Marshall Date: 2/19/24

Witness: Cody Jones Date: 2/19/24