Synthetic Relationships

Documentation:

1. Write down your responses to the questions posted in the brief, and how your project addresses them.

1) **Q:** Are both creatures going to be in the same environment or in different ones?
A: In our system, there are two input sensors (ultrasonic distance and potentiometer) and three output sensors (LED light, buzzer and servo motor). These sensors are in the same environment.

input sensors	ultrasonic distance	potentiometer	
output sensors	LED light	buzzer	servo motor

2) **Q:** Will their "worldviews" (the sensory data they receive) be the same, opposite, or unrelated?
A: The buzzer and LED light receive the same data source from the potentiometer. The servo motor separately receives the data stemmed from the ultrasonic distance.

same data source	buzzer	LED light
separate data source	servo motor	

3) **Q:** Will they be stationary or mobile - and if mobile: self-propelled, or moving with the help of another being. How will their sensory perceptions be processed internally - and is there difference in the way one processes data from the other?
A: The ultrasonic distance generates and mobiles data by sensing the distance from the hand, and then the servo motor rotates accordingly.

The potentiometer produces and mobiles data by rotating the button, and meanwhile the LED light will turn on or off responding to the data range set according to the original data stemmed from the potentiometer, and the buzzer will ring or not on the basis of the data range set according to the same data source with the LED light.

The ultrasonic distance has digital signals, which are transformed to float format in programming.

The potentiometer has analog signals, and the data with integer format it generated upload and transmitted directly.

4) **Q:** How will the result of this process be encoded for transmission?

A: The data with float format from the ultrasonic distance are transmitted to control the servo motor by limiting the data range from 0 to 150, based on the feathers of the servo motor.

The data with integer format sourcing from the potentiometer are set to the data range in order to control the LED light and buzzer. (If data received are larger than 2000, the light turns on and the buzzer rings.

Sensor		Data format
ultrasonic distance	digital signals	float format
potentiometer	analog signals	integer format
LED light		data range (0-2000, 2000-4095)
buzzer		data range (0-2000, 2000-4095)
servo motor		data range (0-150)

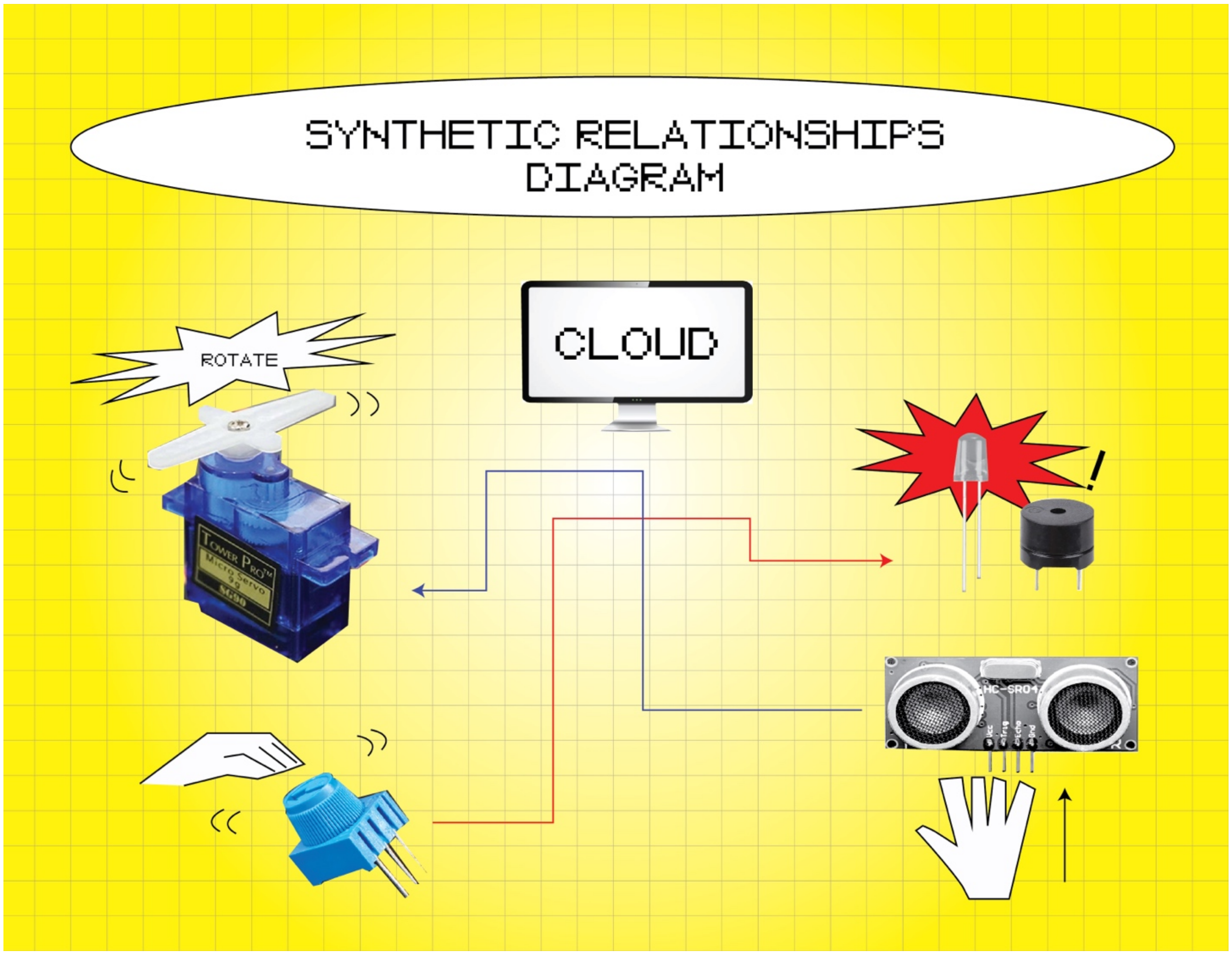
5) **Q:** Will the transmission be continuous or intermittent?

A: the transmissions are continuous.

6) **Q:** Will they be able to transmit and receive information at the same time or will they take turns?
   A: the LED light and the buzzer are sensitive to the data received from the input sensor. The sensitivity of the servo motor depends on the parameter set by delay().

7) **Q:** Is their communication going to be perceptible by the outsiders?
   A: Yes, when rotating the potentiometer, the light and buzzer will show the responding changes. When moving hand close to or being away from
   The ultrasonic distance, the rotation of the servo meter changes accordingly.

2. Create a diagram outlining the communication schema and depicting all your decisions.


SYNTHETIC RELATIONSHIPS DIAGRAM
CLOUD
ROTATE
Tower Pro Micro Servo 9g SG90
HC-SR04

3. *Create a video explaining the inner workings of your system and showing it in action.*

Video: potentiometer& LED light & buzzer

Video: ultrasonic distance & servo

Video: servo working

4. *Write about the process of making from iteration to iteration. You can reference this documentLinks to an external site. to see examples of how the techniques of collage, illustration, photography and sketching*

can be used to explain your vision for the possibilities and implications of your project.

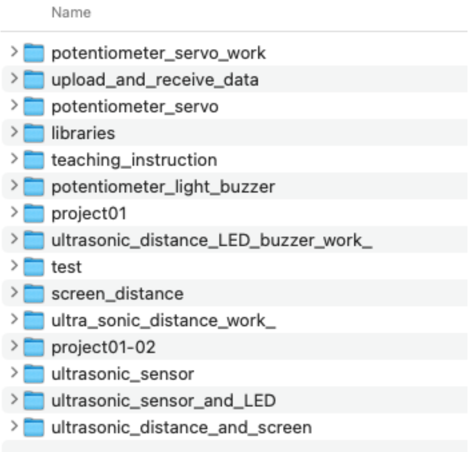
https://github.com/yuliniris/CreativeTech3/tree/main/Synthetic%20Relationships%20Brief

potentiometer & light & buzzer
At the beginning, the potentiometer was set 0, and the buzzer did not ring, and the light was turn off.
When the data of potentiometer increased to 500, the buzzer ringed and the light was turn off.
When the data of potentiometer increased to 1800, the buzzer ringed and the light was turn on.
Partner's program can receive the data generated and uploaded from the potentiometer continuously.
When turning the potentiometer back to 0, the buzzer stopped ring and light turned off.
The data uploaded and changed accordingly, and sent to the partner's program simultaneously.

ultrasonic distance & servo
When the hand was close to the ultrasonic distance, the servo worked.
When the hand was far away from the ultrasonic distance, the servo stopped work.

In order to make it work, we found different versions of codes, revised, tested, in order to figure out whether they were suitable for our boards and sensors. Based on errors, we debugged and did iterations.

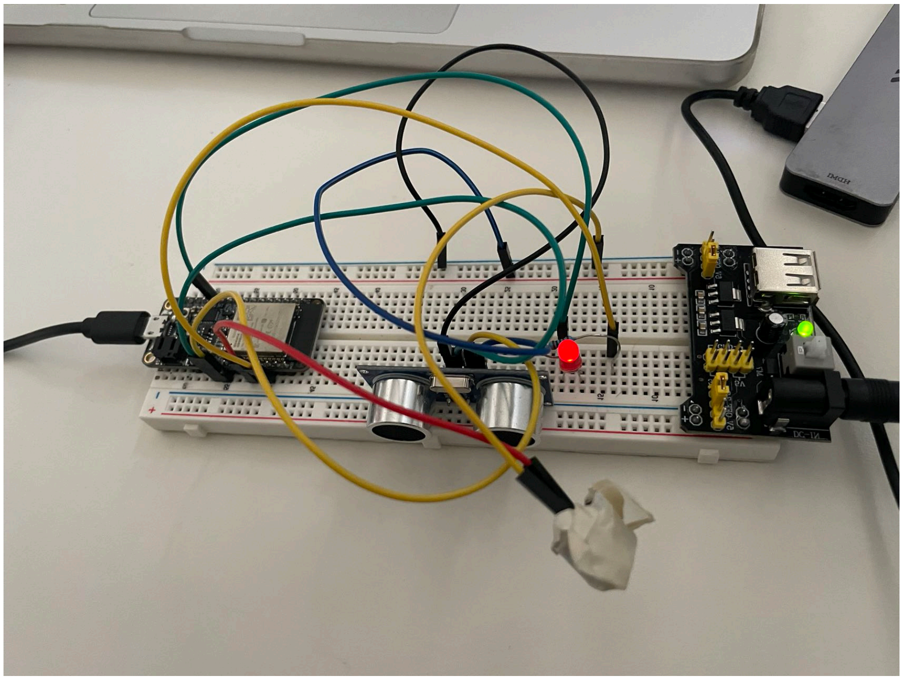
As for boards and sensors, we tested the line connections in different ways. Also, we added the charger bridge to provide matched voltage.



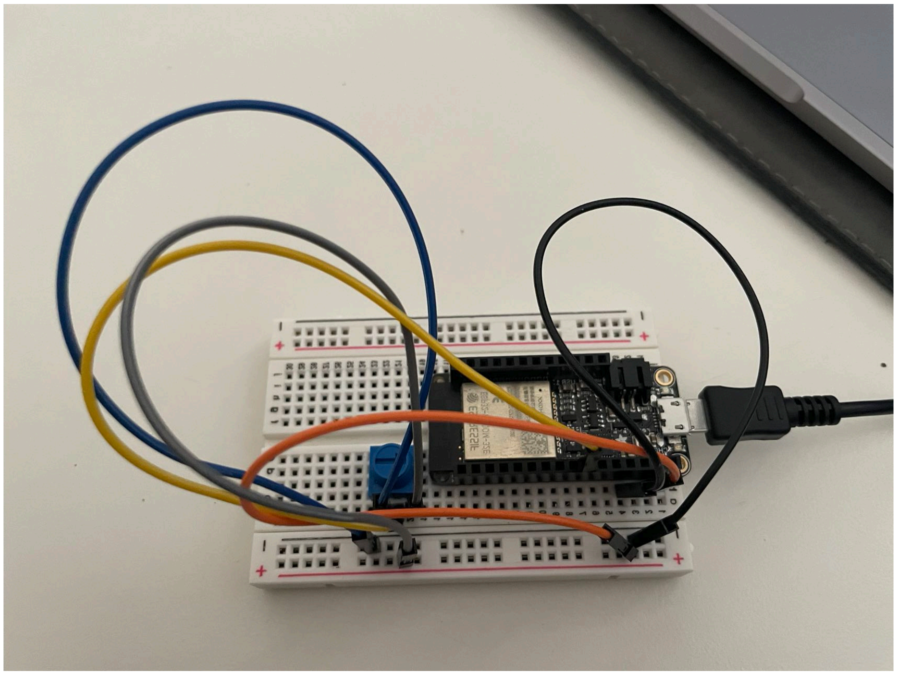
Name

potentiometer_servo_work
upload_and_receive_data
potentiometer_servo
libraries
teaching_instruction
potentiometer_light_buzzer
project01
ultrasonic_distance_LED_buzzer_work_
test
screen_distance
ultra_sonic_distance_work_
project01-02
ultrasonic_sensor
ultrasonic_sensor_and_LED
ultrasonic_distance_and_screen

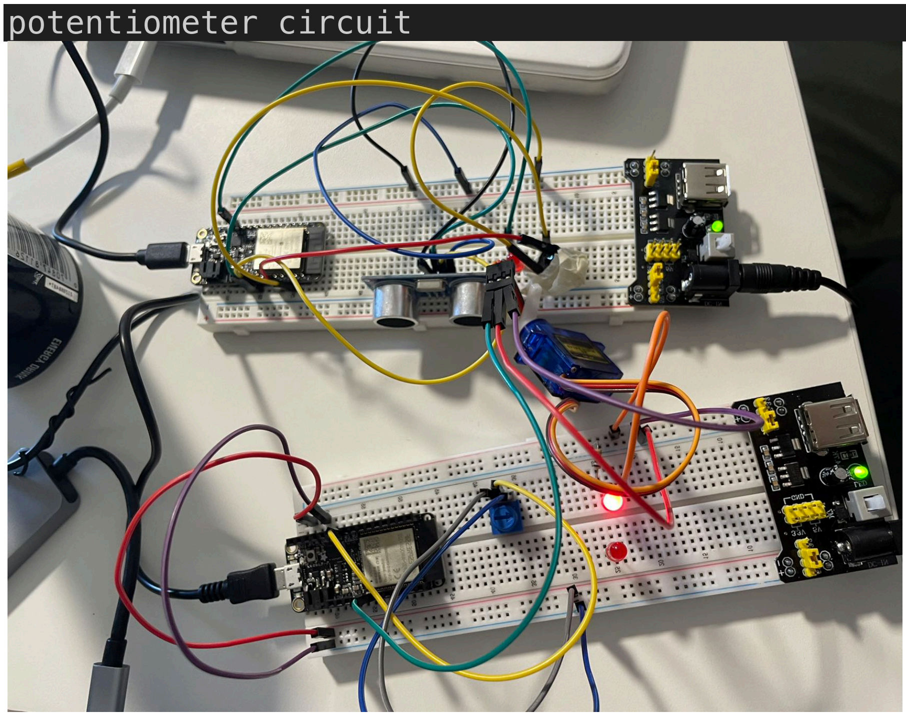
Codes files



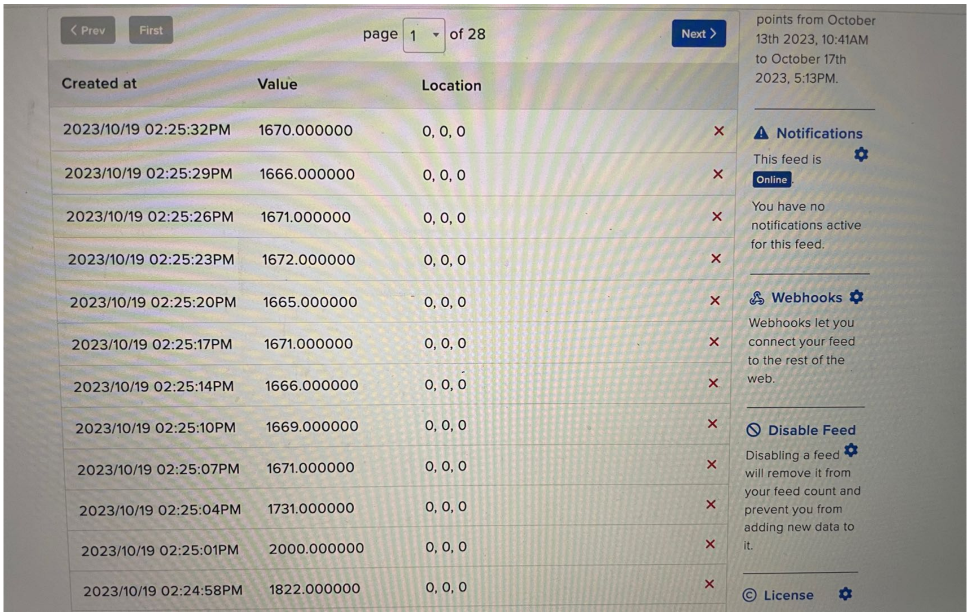
ultrasonic distance circuit



potentiometer circuit



servo circuit



Created at	Value	Location
2023/10/19 02:25:32PM	1670.000000	0, 0, 0
2023/10/19 02:25:29PM	1666.000000	0, 0, 0
2023/10/19 02:25:26PM	1671.000000	0, 0, 0
2023/10/19 02:25:23PM	1672.000000	0, 0, 0
2023/10/19 02:25:20PM	1665.000000	0, 0, 0
2023/10/19 02:25:17PM	1671.000000	0, 0, 0
2023/10/19 02:25:14PM	1666.000000	0, 0, 0
2023/10/19 02:25:10PM	1669.000000	0, 0, 0
2023/10/19 02:25:07PM	1671.000000	0, 0, 0
2023/10/19 02:25:04PM	1731.000000	0, 0, 0
2023/10/19 02:25:01PM	2000.000000	0, 0, 0
2023/10/19 02:24:58PM	1822.000000	0, 0, 0

points from October 13th 2023, 10:41AM to October 17th 2023, 5:13PM.

Notifications

This feed is Online.

You have no notifications active for this feed.

Webhooks

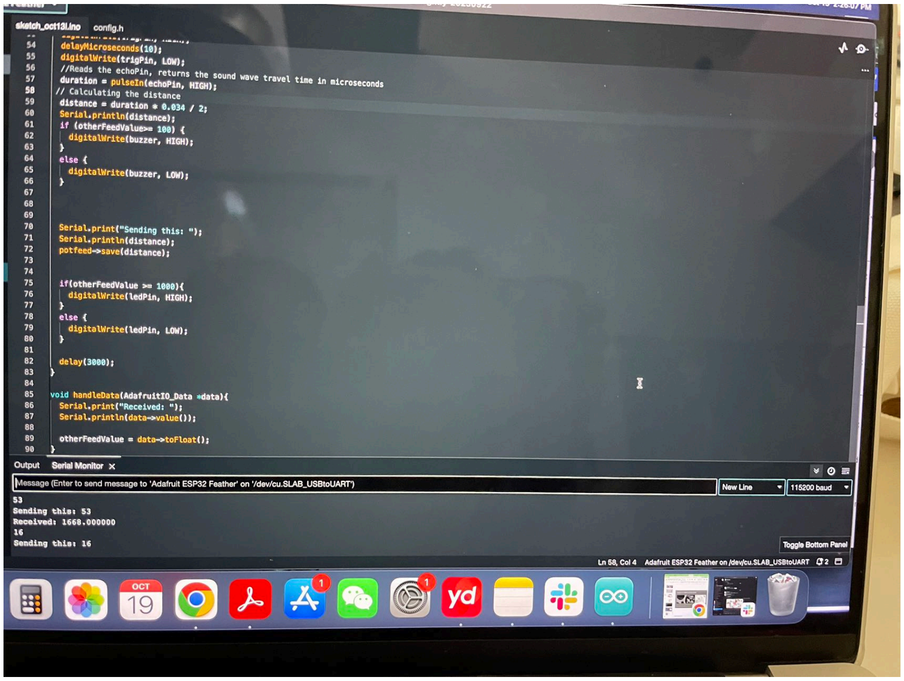
Webhooks let you connect your feed to the rest of the web.

Disable Feed

Disabling a feed will remove it from your feed count and prevent you from adding new data to it.

License

data uploading to IO online



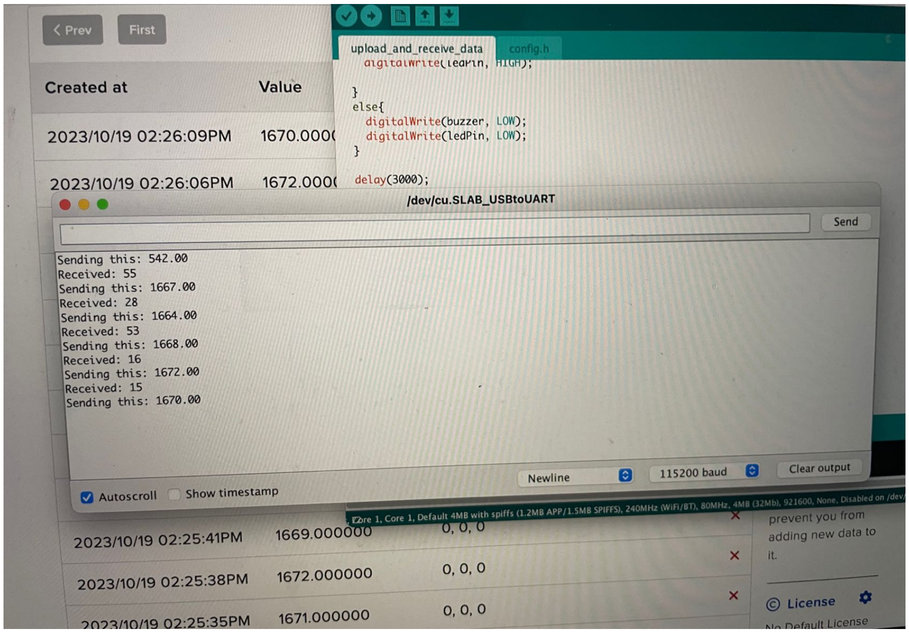
```
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  //Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
 // Calculating the distance
  distance = duration * 0.034 / 2;
  Serial.println(distance);
  if (otherFeedValue>= 100) {
    digitalWrite(buzzer, HIGH);
  }
  else {
    digitalWrite(buzzer, LOW);
  }



  Serial.print("Sending this: ");
  Serial.println(distance);
  potfeed->save(distance);


  if(otherFeedValue >= 1000){
    digitalWrite(ledPin, HIGH);
  }
  else {
    digitalWrite(ledPin, LOW);
  }

  delay(3000);
}

void handleData(AdafruitIO_Data *data){
  Serial.print("Received: ");
  Serial.println(data->value());

  otherFeedValue = data->toFloat();
}
```

Message (Enter to send message to 'Adafruit ESP32 Feather' on '/dev/cu.SLAB_USBtoUART')

New Line | 115200 baud

```
53
Sending this: 53
Received: 1668.000000
16
Sending this: 16
```

partner's data sending and receiving



Created at	Value	
2023/10/19 02:26:09PM	1670.0000	
2023/10/19 02:26:06PM	1672.0000	
2023/10/19 02:25:41PM	1669.000000	0, 0, 0
2023/10/19 02:25:38PM	1672.000000	0, 0, 0
2023/10/19 02:25:35PM	1671.000000	0, 0, 0

```
  digitalWrite(ledPin, HIGH);

}
else{
  digitalWrite(buzzer, LOW);
  digitalWrite(ledPin, LOW);
}

delay(3000);
```

/dev/cu.SLAB_USBtoUART

```
Sending this: 542.00
Received: 55
Sending this: 1667.00
Received: 28
Sending this: 1664.00
Received: 53
Sending this: 1668.00
Received: 16
Sending this: 1672.00
Received: 15
Sending this: 1670.00
```

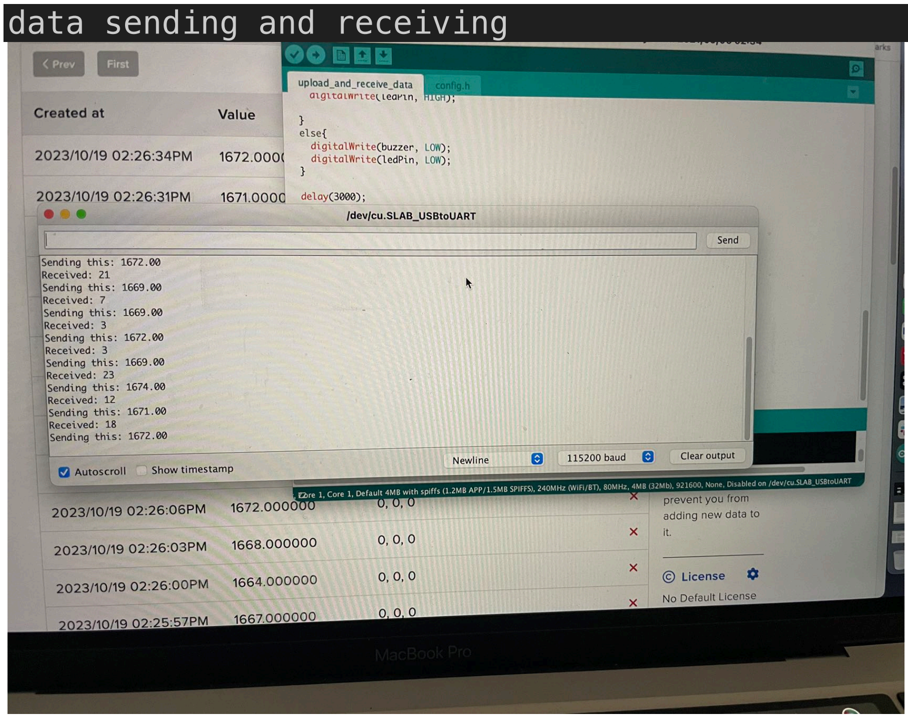
☑ Autoscroll □ Show timestamp — Newline — 115200 baud — Clear output

Core 1, Core 1, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), 80MHz, 4MB (32Mb), 921600, None, Disabled on /dev/

prevent you from adding new data to it.

License
No Default License

data sending and receiving



Created at	Value	
2023/10/19 02:26:34PM	1672.0000	
2023/10/19 02:26:31PM	1671.0000	
2023/10/19 02:26:06PM	1672.000000	0, 0, 0
2023/10/19 02:26:03PM	1668.000000	0, 0, 0
2023/10/19 02:26:00PM	1664.000000	0, 0, 0
2023/10/19 02:25:57PM	1667.000000	0, 0, 0

```
  digitalWrite(ledPin, HIGH);

}
else{
  digitalWrite(buzzer, LOW);
  digitalWrite(ledPin, LOW);
}

delay(3000);
```

/dev/cu.SLAB_USBtoUART

```
Sending this: 1672.00
Received: 21
Sending this: 1669.00
Received: 7
Sending this: 1669.00
Received: 3
Sending this: 1672.00
Received: 3
Sending this: 1669.00
Received: 23
Sending this: 1674.00
Received: 12
Sending this: 1671.00
Received: 18
Sending this: 1672.00
```

☑ Autoscroll □ Show timestamp — Newline — 115200 baud — Clear output

Core 1, Core 1, Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), 80MHz, 4MB (32Mb), 921600, None, Disabled on /dev/cu.SLAB_USBtoUART

prevent you from adding new data to it.

License
No Default License

data sending and receiving