Group Member Names: Cooper Ayers, Garret Wagner, and Grace Jenkins Welsh

Course and Quarter: ENGR114 Winter 2020

Date: 03/13/2020

Revision number: 4

**People Counter**

**Problem Statement:**

It can be useful to know how many people are in a room when one cannot watch a door constantly. For example, in a daycare, a teacher MUST know how many people are in a room but obviously has a lot to pay attention to. Perhaps one must know how many people remain in an area, so they may ensure there is no one left in that room: For example, how many are left in an office at the end of the day? We will record and display a count of the people in a room at any given time by adding and subtracting as people enter and leave a room.

**Hardware Setup:**

Bill of Materials:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Part Name** | **Purpose** | **Item Name** | **URL** | **Price** |
| RedBoard | Logic Board | SparkFun RedBoard Qwiic DEV-15123 | <https://www.sparkfun.com/products/15123> | $19.95 |
| Vernier Interface Shield | Interfaces with Vernier Photogates | SparkFun Vernier Interface Shield DEV-12858 | <https://www.sparkfun.com/products/12858> | $24.95 |
| 2 x Photogates | Relays information to be recorded | Vernier Photogate | <https://www.vernier.com/product/photogate/> | 2 x $49.00 |
| 2 x Laser Pointers | Used with photogate | Vernier Laser Pointer | <https://www.vernier.com/product/laser-pointer/> | 2 x $19.00 |
| 2 x Laser Stands | Holds Laser Pointers | Vernier Laser Pointer Stand | <https://www.vernier.com/product/laser-pointer-stand/> | 2 x $14.00 |
| USB Micro-B Cable | Power RedBoard | CAB-10215 | <https://www.sparkfun.com/products/10215> | $4.95 |

Hardware Schematic:

![A circuit board

Description automatically generated]()

Hookup Guide:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Part** | **Pin** | **Connector** | **Pin** | **Part** |
| RedBoard |  | Micro USB |  | Laptop |
| Vernier Interface Shield |  | board |  | RedBoard |
| Photogate 1 | 2 |  |  | Vernier Interface Shield |
| Photogate 2 | 6 |  |  | Vernier Interface Shield |

Images:

Video Documentation Attached

**Code:**

Python Code:

import serial

import time

serial. \_\_version\_\_

import numpy as np

sensor1=[]

sensor2=[]

ser = serial.Serial('COM3', 9600) #Open the port to recive data from the aurdinuo board

q=0 #variable to count how many people are in the room.

for i in range(0,30): #the amount of time that the code will run.

b = ser.readline() # variable to save the data coming in from the board

data\_str=bytes.decode(b) #Turning the bytes into strings

x=data\_str.strip()# getting rid unwated characters

data=np.array(x) #saving data to an array to orginize that data.

y=x[0] #indexing all the data from senor 1

z=x[1] #indexing all the dtat from sensor 2

sensor1.insert(0,y) #putting the newest data from sensor 1 at the start

sensor2.insert(0,z) #puting the newest data from sensore 2 at the top of the list

if sensor1[0]==sensor2[0]: #if statement if both lazers are in the same state

time.sleep(.1) #we done want anything to happen if they are both in the same state.

if sensor1[0]>sensor2[0]:#if statement if sensor 1 is triped and sensor 2 is not

q=q+1 #adding a person to the count in the room

print(f'there are {q} people in the room right now.') #printing the current total in the room

while not sensor1[0]==sensor2[0]: # building the conditions for the dealy

time.sleep(.5) # building a delay

break #break out of the loop

if sensor1[0]<sensor2[0]: # if statement for when someone is leaving

q=q-1 #removing a person from the room

print(f'there are {q} people in the room right now.') #printing how many people are in the room

while not sensor1[0]==sensor2[0]:#building conditions to let someone walk trough

time.sleep(.5) # break

break # get out of the loop

time.sleep(.2) # sleep

ser.close() # close the loop to the board.

>>>there are -1 people in the room right now.

there are 0 people in the room right now.

there are 1 people in the room right now.

there are 2 people in the room right now.

there are 3 people in the room right now.

there are 4 people in the room right now.

there are 5 people in the room right now.



ser.close()

Arduino Code:

digitalWrite(LEDpin, LOW);// turn off LED

gate1 = "1";

delay(500);

}

else

{

digitalWrite(LEDpin, HIGH);// turn on LED

gate1 = "0";

}

photogateStatus = digitalRead(photogatePin2);//low when blocked

if (photogateStatus == HIGH)

{

digitalWrite(LEDpin, LOW);// turn off LED

gate2 = "1";

delay(1000);

}

else

{

digitalWrite(LEDpin, HIGH);// turn on LED

gate2 = "0";

}

String gates = gate1 + gate2;

Serial.println(gates);

}

// end of loop

**Results:**

The People Counter did succeed in adding and subtracting people based on the laser disruption. There were errors with timing which made initial testing inconsistent. Many of these issues were addressed with altering wait times.

**Future Work:**

Improvements can be made to this project by keeping the number accurate when there are multiple entrances and exits and there are multiple sensors to keep track of. Being able to display this number to an application or multiple devices would be helpful to make this process more useful to groups of people rather than people directly in the room. Multiple lasers could solve the problem of multiple people entering together, but it is an issue which would need to be addressed if there is no policy in place for entering a room one at a time.

**License:**

Copyright ©2020 Cooper Ayers, Grace Jenkins Welsh, Garret Wagner

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.