Práctica 3: Simulación comunicación entre Arduinos

# Ejercicio 1:

# Ejercicio2:

Código:

-Librería para el sensor de ultrasonidos:  
//Librerias necesarias

#include <arduino.h>

class ultrasonidos{

public:

ultrasonidos(){

trigger = 5;

echo = 4;

}

ultrasonidos(int trig, int Echo){

trigger = trig;

echo = Echo;

}

void iniciar()

{

pinMode(trigger, OUTPUT);

pinMode(echo, INPUT);

}

long medicion(){

long tiempo;

long distancia;

digitalWrite(trigger, LOW);

delayMicroseconds(5);

digitalWrite(trigger, HIGH);

delayMicroseconds(15);

digitalWrite(trigger, LOW);

tiempo = pulseIn(echo, HIGH);

distancia = tiempo \* 0.01657;

return (distancia);

}

private:

int trigger, echo;

};

-Código para el Arduino A:

//Importamos librerias necesarias

#include "ultrasonidos.cpp"

//Declaramos las variables y sensores necesarios

ultrasonidos sensor1(6, 7);

#define transmisor 3

#define receptor A0

#define led 13

//Funciones para la transmision y la recepcion

float recibir\_datos\_pwm(int pin)

{

  unsigned long duracion = pulseIn(pin, HIGH);

  double dutycicle = (duracion / 2145.0) \* 255;

  float distancia\_trans = round((dutycicle / 255) \* 400);

  return distancia\_trans;

}

void enviar\_datos\_pwm(int pin, long datos)

{

  float distancia\_trans = float(datos) / 400 \* 255;

  int pwm = int(distancia\_trans);

  analogWrite(transmisor, pwm);

}

//Programa principal

void setup() {

  // put your setup code here, to run once:

  Serial.begin(9600);

  pinMode(led, OUTPUT);

  pinMode(receptor, INPUT);

  sensor1.iniciar();

}

void loop() {

  // put your main code here, to run repeatedly:

  enviar\_datos\_pwm(transmisor, sensor1.medicion());

  if (recibir\_datos\_pwm(receptor) <= (30 \* 255 / 100)) digitalWrite(led, HIGH);

  else digitalWrite(led, LOW);

}

-Código del Arduino B:

//Importamos librerias necesarias

#include "ultrasonidos.cpp"

//Declaramos las variables y sensores necesarios

ultrasonidos sensor1(6, 7);

#define transmisor 10

#define receptor A0

//Funciones para la transmision y la recepcion

float recibir\_datos\_pwm(int pin)

{

  unsigned long duracion = pulseIn(pin, HIGH);

  double dutycicle = (duracion / 2145.0) \* 255;

  float distancia\_trans = round((dutycicle / 255) \* 400);

  return distancia\_trans;

}

void enviar\_datos\_pwm(int pin, long datos)

{

  float distancia\_trans = float(datos) / 400 \* 255;

  int pwm = int(distancia\_trans);

  analogWrite(transmisor, pwm);

}

//Programa principal

void setup() {

  // put your setup code here, to run once:

  sensor1.iniciar();

  pinMode(transmisor, OUTPUT);

}

void loop() {

  // put your main code here, to run repeatedly:

  float datos\_arduino\_a = recibir\_datos\_pwm(receptor);

  if (datos\_arduino\_a <= float(sensor1.medicion())) analogWrite(transmisor, 50);

  else analogWrite(transmisor, 255);

  delay(1000);

}

Enlaces de github y de tinkercad para el ejercicio:  
<https://www.tinkercad.com/things/0pZKfoDCb1O-proxextos1practica3ej2>

<https://github.com/mans199876/proxectos1_practica3.git>