**Assignment -4**

|  |  |
| --- | --- |
| Assignment Date | 19 September 2022 |
| Student Name | S.Dharani |
| Student Roll Number | 911919104701 |
| Maximum Marks | 2 Marks |

Reference link: <https://wokwi.com/projects/312346565007114818>

#include "Ultrasonic.h"

Ultrasonic::Ultrasonic(uint8\_t trigPin, uint8\_t echoPin, unsigned long timeOut) { trig = trigPin; echo = echoPin;

threePins = trig == echo ? true : false;

pinMode(trig, OUTPUT); pinMode(echo, INPUT); timeout = timeOut;

}

unsigned int Ultrasonic::timing() {

if (threePins)

pinMode(trig, OUTPUT);

digitalWrite(trig, LOW); delayMicroseconds(2); digitalWrite(trig, HIGH); delayMicroseconds(10); digitalWrite(trig, LOW);

if (threePins)

pinMode(trig, INPUT);

previousMicros = micros();

while(!digitalRead(echo) && (micros() - previousMicros) <= timeout); // wait for the echo pin HIGH or timeout

previousMicros = micros();

while(digitalRead(echo) && (micros() - previousMicros) <= timeout); // wait for the echo pin LOW or timeout

return micros() - previousMicros; // duration

}

/\*

* If the unit of measure is not passed as a parameter,
* sby default, it will return the distance in centimeters.
* To change the default, replace CM by INC.

\*/

unsigned int Ultrasonic::read(uint8\_t und) { return timing() / und / 2; //distance by divisor

}

/\*

* This method is too verbal, so, it's deprecated.
* Use read() instead.

\*/

unsigned int Ultrasonic::distanceRead(uint8\_t und) { return read(und);

}

class Ultrasonic { public:

Ultrasonic(uint8\_t sigPin) : Ultrasonic(sigPin, sigPin) {};

Ultrasonic(uint8\_t trigPin, uint8\_t echoPin, unsigned long timeOut = 20000UL); unsigned int read(uint8\_t und = CM);

unsigned int distanceRead(uint8\_t und = CM) \_\_attribute\_\_ ((deprecated ("This method is deprecated, use read() instead.")));

void setTimeout(unsigned long timeOut) {timeout = timeOut;} void setMaxDistance(unsigned long dist) {timeout = dist\*CM\*2;}

private:

uint8\_t trig; uint8\_t echo; boolean threePins = false; unsigned long previousMicros; unsigned long timeout; unsigned int timing();

};

