International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

Exploring the Potential of Arduino and Ultrasonic Sensor Integration in Radar Systems

Neetu Settia[1], Krishan Mohan Mandal[2]
[1]Assistant Professor, Electronics and communication Engineering, Guru Tegh Bahadur Institute of Technology, GGSIPU, New Delhi, India.
[2]UG – Artificial Intelligence and Data Science, Guru Tegh Bahadur Institute of Technology, GGSIPU, New Delhi, India.
Emails: *neetu_aug3@yahoo.co.in[1], krishan97mohan@gmail.com[2]*

Abstract
Radar, short for Radio Detection and Ranging, is a system that utilizes radio waves to determine various characteristics of objects in its vicinity. This powerful technology enables us to measure the distance, direction, and radial velocity of objects relative to its location. Radar has found numerous applications in various domains, including aviation, maritime, space exploration, defence, meteorology, and mapping. In this paper, we explore a novel approach to radar systems by integrating Arduino technology and ultrasonic sensors. Unlike traditional radar systems that solely rely on waves, this setup introduces the use of ultrasonic signals, resulting in a significant change. The setup includes an Arduino board that is connected to a servo motor responsible for moving a sensor. This sensor detects and measures objects within a specific range. To simplify the programming process for the Arduino board, we utilize the Arduino IDE, while the Processing IDE enhances visualization through a user interface. By combining both hardware and software in this innovative manner, the radar system successfully achieves object detection and visualization, opening up possibilities for various applications such as security, navigation, and more.
Keywords: *Arduino Board; Arduino IDE; Object Detection; Processing IDE; Traditional Radar System.*

1. Introduction

Radar systems have always been a part of detection technology using radio waves to identify the presence, distance and position of objects in their operating area. Yet the emergence of sensors introduces an alternative by utilizing sound waves to achieve similar goals. This article delves into how Arduino based technology and ultrasonic sensors combined to create a radar system [1]. Ultrasonic sensors are well known for their capacity to determine distance and orientation offering benefits compared to radar systems. By converting energy into signals these sensors allow for contactless measurements enabling precise detection, across various scenarios such as collision prevention and surveillance. Additionally, the speed at which sound travels, affected by factors, like temperature enhances the flexibility and precision of ultrasonic driven detection. In this project we've taken an approach to radar detection by using technology. The core of the system is an Arduino board that manages how the ultrasonic sensor communicates, with a servo motor for controlling direction. Programmers can easily integrate code using the Arduino IDE while the Processing IDE helps with visualizing and analyzing data to enhance user experience. This journal dives into the hardware and software aspects involved in Arduino based radar systems shedding light on their mechanisms and complexities [2]. By exploring real world implementation and performance measures readers can grasp the applications and advancements in detection technology. Whether its monitoring pathways or enhancing border security, the versatility and effectiveness of radar systems highlight their importance, in contemporary surveillance and navigation contexts [3].

1.1. Components (12 pt)

Hardware

- Arduino UNO
- Servo Motor (SG90)
- Ultrasonic Sensor (HC-SR04)
- Jumping Wires

Software

- Arduino IDE
- Processing IDE

1.1.1. Hardware

The hardware components of the radar system encompass the tangible elements responsible for sensing real-world data [4].

Arduino UNO: The Arduino Uno stands out as a microcontroller board known for its versatility, accessibility and user-friendly nature, in electronics projects and prototyping. Created by Arduino LLC the Uno is powered by the ATmega328P microcontroller offering a variety of analog input/output pins for connecting with sensors, actuators and other electronic devices. Featuring 14 pins with 6 PWM outputs and 6 analog input pins the Uno offers connectivity options for various applications in Figure 1.

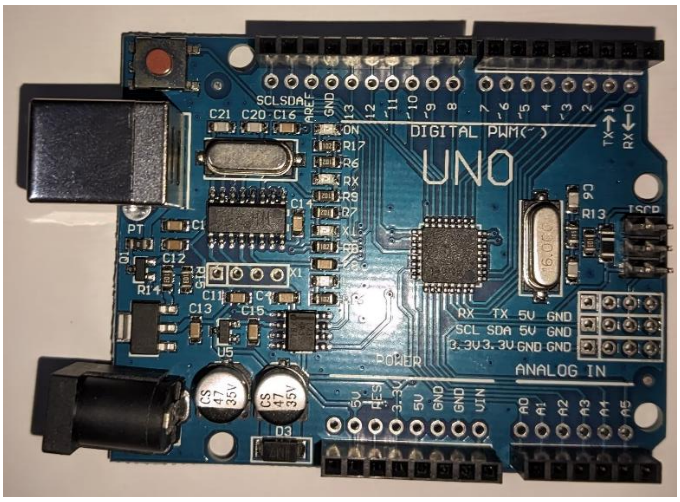

Figure 1 **Arduino UNO**

A significant benefit of the Arduino Uno is its to use development environment called the Arduino Integrated Development Environment (IDE). This open-source software simplifies programming tasks, allowing users to write and upload code in the Arduino programming language based on Wiring and C/C++. The IDE provides a range of written functions and examples that help even beginners to start coding quickly [5]. Moreover, the Arduino Uno includes a USB interface for connection to computers to facilitate code uploading and serial communication. Its compact size, affordability and compatibility with shields and modules make it a preferred choice, for hobbyists, educators and professionals looking to bring their projects to fruition. The Arduino Uno represents a blend of creativity and accessibility, in technology enabling people to delve into the realm of electronics and devise answers to problems [6].

Servo Motor (SG90): The SG90 servo motor is a choice, for hobbyist projects, robotics, and automation due to its size, light weight and affordability. It is manufactured by companies. Known for its reliability and user-friendly nature. When referred to as "SG90 " it often indicates the model number or type of the motor in Figure 2.

Figure 2 **Servo Motor (Sg90)**

This servo motor functions based on width modulation (PWM) adjusting the motor shafts

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

rotation by changing the width of pulses it receives. It can rotate between 0 to 180 degrees making it versatile for motion control applications [7]. Typically equipped with power) brown or black (ground) and orange or yellow (signal) wires, the SG90 servo motor can be easily connected to microcontrollers like Arduino and Raspberry Pi. Despite its size it provides torque and speed for tasks like managing robotic arms, steering systems in remote controlled vehicles as well as handling automation duties such, as door operations or object manipulation. The SG90 servo motor is highly valued for its size, cost effectiveness and flexibility making it a top pick, for hobbyists, teachers and experts looking to meet mechanical control requirements.

Ultrasonic Sensor (HC-SR04): The HC SR04 ultrasonic sensor is a popular and budget friendly distance measuring tool commonly used in electronics projects and robotics. It functions by utilizing echolocation like how bats navigate, emitting waves and calculating the distance by timing their reflection off objects accurately in Figure 3.

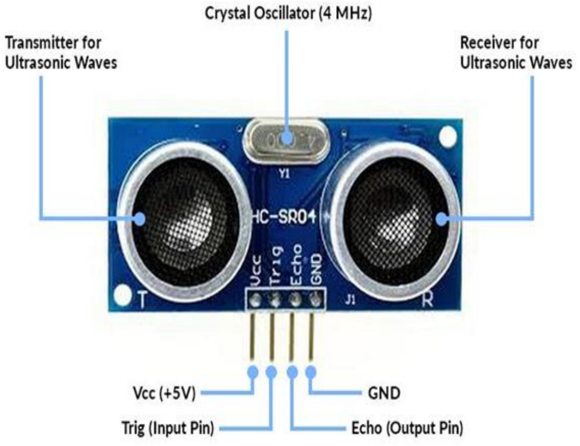

Figure 3 **Ultrasonic Sensor (Hc-Sr04)**

Comprising a transmitter and receiver the HC SR04 sensor operates by transmitting waves at 40 kHz with the receiver detecting the returning signals. By measuring the time, between transmission and reception of these signals the sensor can determine object distance with accuracy. This user-friendly sensor can easily connect to microcontrollers like Arduino, Raspberry Pi or other embedded systems, with four connections: VCC (power) GND (ground) TRIG (trigger) and ECHO (echo signal). To utilize the sensor effectively one must send a pulse to the pin to initiate the ultrasonic wave emission and measure how long it takes for the ECHO pin to receive the reflected signal. The HC SR04 ultrasonic sensor is famous, for its dependability, precision and adaptability making it ideal for a variety of uses like spotting obstacles measuring distances tracking objects and sensing proximity. Its affordability, user nature and ability to work with platforms have cemented its status as a key element in numerous do it yourself endeavours, educational initiatives and business products, in various sectors [8].

1.1.2. Software

The software part of the radar system is crucial, for showing data and creating a to use interface, for understanding the information. Essentially two software platforms work together with the hardware: the Arduino Integrated Development Environment (IDE). The Processing IDE.

Arduino IDE: The Arduino Integrated Development Environment (IDE) is a software tool created to simplify the programming of Arduino microcontroller boards. It offers a user platform, for writing compiling and uploading code to Arduino boards making it simple for users to bring their projects to fruition. A standout feature of the Arduino IDE is its ease-of-use catering to both beginners and seasoned programmers. It utilizes a version of the C/C++ programming language. Provides an array of libraries and examples to assist users in coding for their projects. The IDE includes a built-in text editor with features like syntax highlighting, auto completion and error checking streamlining the code development process. Supporting Arduino boards such as models like the Uno, Nano, Mega among others the Arduino IDE simplifies board selection and port configuration for easy code

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

uploading to the target device. Moreover, boasting debugging capabilities and time serial monitoring features the Arduino IDE enables users to troubleshoot their code effectively and communicate with their Arduino boards on the fly. It offers a serial monitor interface, for monitoring data transmitted from the board and debugging output messages.

Processing IDE: The Processing IDE is a software platform crafted for programming, interactive artistry and visual design. Created with the aim of making coding accessible and enjoyable, for artists, designers, educators and enthusiasts alike Processing offers an environment for crafting visual experiences. At its essence Processing is rooted in the Java programming language. Presents simplified syntax and an array of built functions and libraries tailored for visual and interactive applications. This abstraction layer allows users to concentrate on expression than grappling with the intricacies of programming making it a valuable tool for individuals with varying backgrounds and levels of expertise. The Processing IDE boasts a user interface comprising a text editor, a canvas for showcasing graphics and an integrated development environment for coding writing, testing and debugging. It includes features like syntax highlighting, auto completion and error checking to aid in code development and issue resolution. A notable aspect of Processing is its capabilities that empower users to craft captivating visual effects, animations, and interactive encounters. The incorporated graphics library offers an array of functions, for creating shapes, images, text manipulation colors well as gradients textures in Figure 4.

1.2. Flow Chart

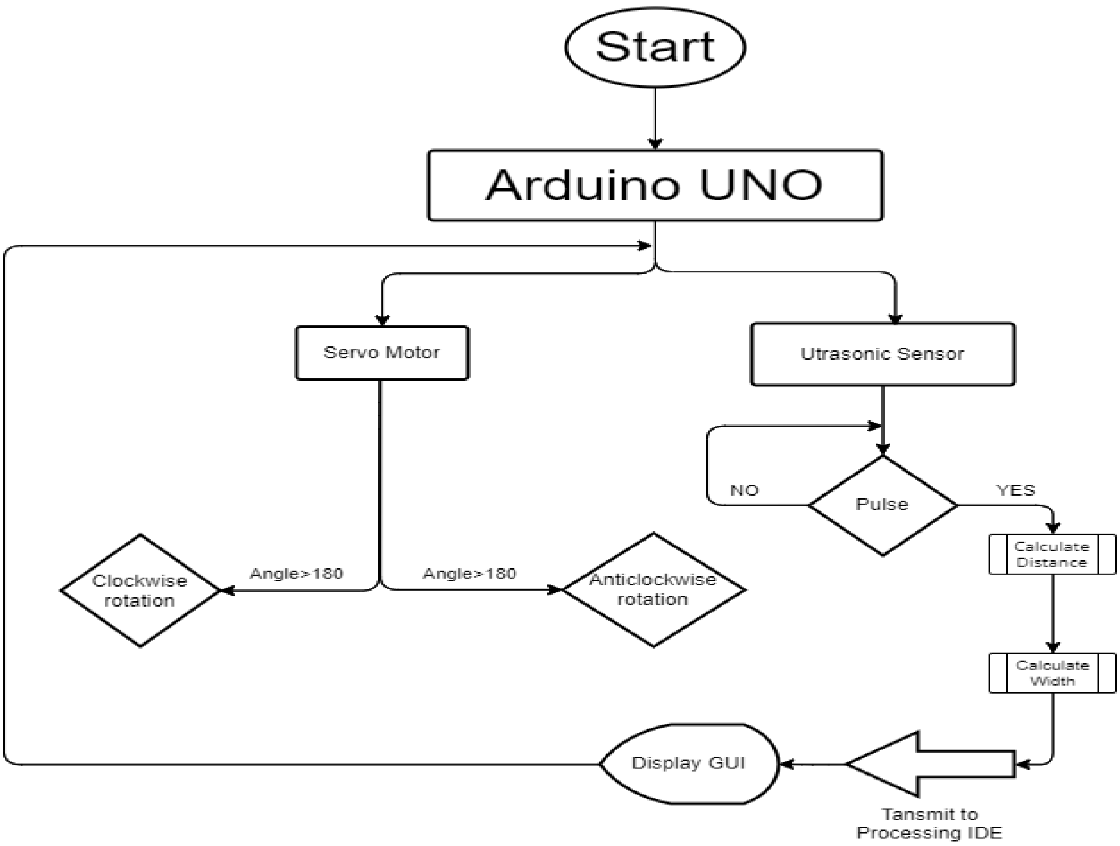

Figure 4 **Flow Chart of Program**

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

2. Pseudocode

2.1. For Arduino IDE

Include Servo library.
Declare constants:
trigPin = 10
echoPin = 11
Declare variables:
duration
distance
Create a servo object named myServo.
Setup:
Set trigPin as OUTPUT.
Set echoPin as INPUT.
Begin serial communication at 9600 baud rates.
Attach servo motor to pin 12.
Loop:
Iterate from 0 to 180 degrees:
Set servo position to current iteration value.
Delay for 30 milliseconds.
Call calculate Distance function. Print current iteration value and distance to Serial Port.
Iterate from 180 to 0 degrees:
Set servo position to current iteration value.
Delay for 30 milliseconds.
Call calculateDistance function.
Print current iteration value and distance to Serial Port.
CalculateDistance function:
Set trigPin to LOW.
Delay for 2 microseconds.
Set trigPin to HIGH for 10 microseconds.
Set trigPin to LOW.
Measure pulse duration from echoPin.
Calculate distance using the formula: distance = (duration * 0.034) / 2.
Return distance.

2.2. For Processing IDE

Import serial communication library.
Import KeyEvent library for reading data from serial port.
Declare variables:
myPort (Serial object)
angle (String)
distance (String)
data (String)
noObject (String)
pixsDistance (float)
iAngle (int)
iDistance (int)
index1 (int)
index2 (int)
orcFont (PFont object)
Setup:
Set canvas size.
Enable anti-aliasing.
Initialize serial communication on COM5 at 9600 baud rates.
Set buffer to read data until '.' character is received.
Load font for text display.
Draw:
Set fill color to green.
Draw motion blur and slow fade effect.
Call functions to draw radar, line, object, and text.
SerialEvent:
Read data from serial port until '.' character.
Extract angle and distance values from the received data.
Convert angle and distance to integers.
DrawRadar:
Translate coordinates to center.
Draw concentric arcs.
Draw angle lines.
DrawObject:
Translate coordinates to center.
Convert distance from sensor to pixels.
If distance is less than 40cm, draw object line based on angle.
DrawLine:
Translate coordinates to center.
Draw line based on angle.
DrawText:
Draw text on screen:
- Range indicators
- Object status (in range/out of range)
- Angle
- Distance
- Angle labels

Translate and rotate text labels for angle indicators.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

3. Circuit Diagram

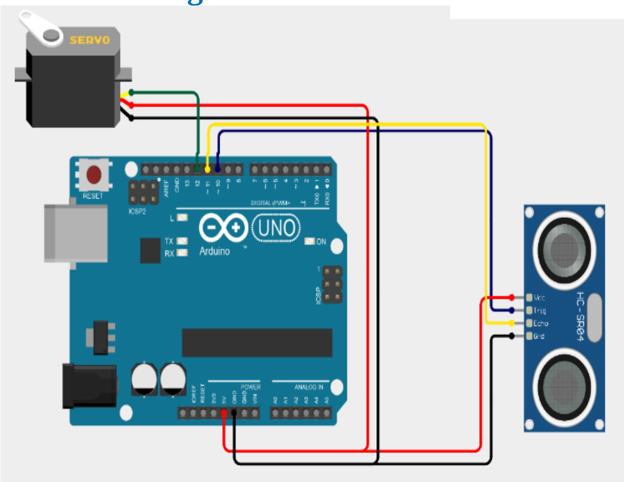

Figure 5 Circuit Diagram with Software [3]

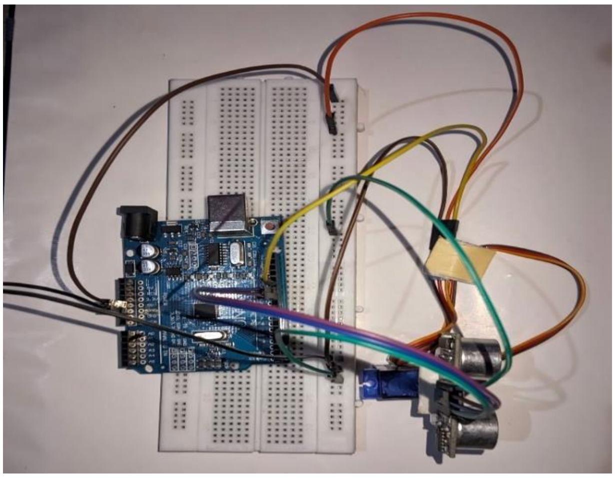

Figure 6 Image of the Project [3]

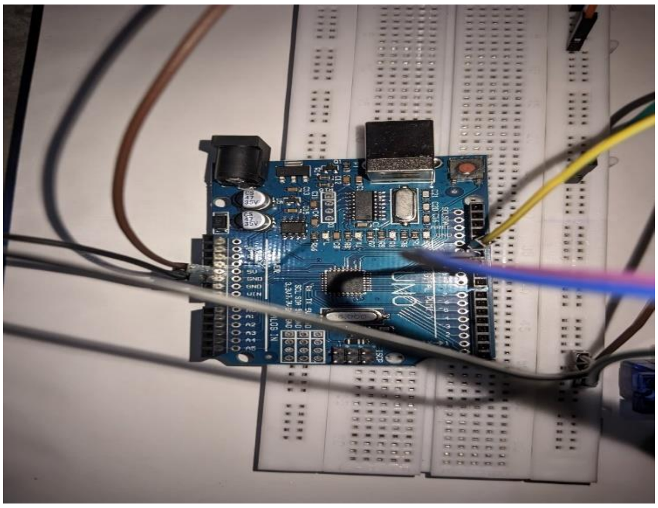

Figure 7 Connection Image of the Project [3]

4. Results and Future Scope

4.1. Results

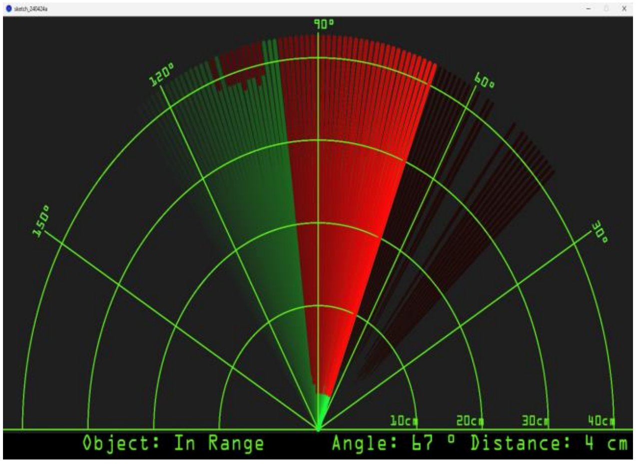

Figure 8 Output of the Project [4.1]

4.2. Future Scope

Our idea, for the radar project was sparked by observing the cutting-edge technologies used in defence and military sectors. How they are now being utilized in civilian areas like the automotive industry. The features found in automated parking systems and safety measures in vehicles have demonstrated how radar-based systems can improve safety and effectiveness. Recent advancements in self-parking technology from car manufacturers such as Audi and Ford along with the progress made in driverless cars by companies like Google and Tesla highlight the increasing importance of radar tech in today's transportation landscape Figure [5-8]. Our project is versatile, capable of being incorporated into systems like cars, bikes or other relevant contexts. By utilizing Arduino technology, we have the flexibility to customize the radar module to meet needs and functions. Looking forward potential improvements to our project could include incorporating a buzzer to assist impaired individuals with travel and navigation. Through exploration of ideas and refining our design we hope to contribute to the evolution of radar-based solutions with practical applications, across different scenarios.

e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1530- 1536
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0210

Conclusion

In conclusion the projects showcased in the presentations illustrate how radar systems, those utilizing sensors possess a wide range of applications, from security systems to robotics. While some implementations have limitations like restricted detection range due to servo motor constraints there is a recognition of the potential for improvement and expansion in the future. By incorporating control methods and integrating technologies like 360-degree rotating servo motors and longer-range ultrasonic sensors, radar systems efficiency and effectiveness can be greatly enhanced. These upgrades do not expand the usability of the systems. Also open up possibilities for features such as mobility, alarm systems and obstacle avoidance. Furthermore, the successful execution of project goals highlights the practicality and importance of radar-based solutions in tackling real world issues. Whether its security monitoring or identifying objects while avoiding obstacles these systems demonstrate directions for exploration and advancement. Ultimately the research findings and hands on work showcased in these projects set a groundwork for progress in radar technology. By welcoming innovation and adapting to changing needs radar systems have the potential to play a role, in shaping security measures, automation processes and more.

References

[1]. Hatem, H.R., Abdalla, A.I., Al-Rawi, Z.N.A. (2018). Design and implementation of ultrasonic radar system for distance measurements using Arduino. International Journal of Engineering, (February).
[2]. Mehta, S., Tiwari, S. (2018). Radar System Using Arduino and Ultrasonic Sensor. International Journal of Novel Research in Engineering and Science, 3(4).
[3]. Radar. (n.d.). In Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Radar
[4]. Radar system arduino - Google Scholar Search. (n.d.). Retrieved from https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=radar+system+arduino&btnG=
[5]. Tejashwini, M., Rohith, Ch., Amrutha, G. Sai, & SMRUTHI, Y. (2023). A Short Range Radar Using Arduino. Journal of Engineering Sciences, 14(06), ISSN: 0377-9254.
[6]. Gupta, P., Singh, A., Kumar, A., & Mehtre, V. V. (2022). Radar System Using Arduino and Ultrasonic Sensor. IRE Journals, 6(1), ISSN: 2456-8880.
[7]. Bochare, A., & Saini, M. (2017). Short Range Radar System Using Arduino Uno. International Research Journal of Engineering and Technology (IRJET), 04(10), ISSN: 2395-0056.
[8]. Risi, I., Amakiri, R. C., & Tasie, N. N. (2020). Arduino Uno Controlled Robotic Radar System Object Detector Using Ultrasonic Sensor. American Journal of Engineering Research (AJER), 9(12), pp. 01-08.