

Ultrasonic Sensor

1.0	Learning Outputs.....	1
2.0	Hardware Set Up.....	1
3.0	Simulink Set Up and Distance Measurement Testing.....	2
4.0	Last Good Value (LGV) Logic Filter and Testing.....	3
5.0	Additional Exercises.....	4
6.0	Concluding Remarks.....	5

1.0 Learning Outcomes

After completing this Exercise, you will be able to:

1. Explain and demonstrate how to acquire distance measurements using a digital ultrasonic sensor.
2. Design and implement a Last Good Value (LGV) logic filter to manage erroneous or intermittent readings from a digital sensor.

After completing this Exercise, it is recommended that you refer to the Learning Outcomes.

2.0 Hardware Set Up

The exercise involves connecting an HC-SR04 ultrasonic sensor to an Arduino Uno to measure distance, see Figure 1.

Note that while the ACE-Box (Base + Sense) can be used for all the exercises, it is not required and only the individual components are needed.

Required hardware for this exercise:

- Arduino Uno board (supported by Simulink)
- USB Cable Type A to B
- Breadboard
- Ultrasonic HC-SR04 sensor
- 4 x male-to-male wires

HC-SR04 Ultrasonic Specifications

- Distance measurement range: 0.02 to 4.00 meters
- Accuracy: ± 0.003 meters

Set-up the hardware as shown and following these steps:

1. Connect the ultrasonic sensor directly to the breadboard (as shown in the Figure 1)
2. Connect the 'Trig' signal to Pin 5 and the 'Echo' signal to Pin 4
3. Connect the VCC terminal of the sensor to the 5V supply pin on the Arduino Uno.
4. Connect Gnd terminal of the sensor to a GND pin on the Arduino Uno.

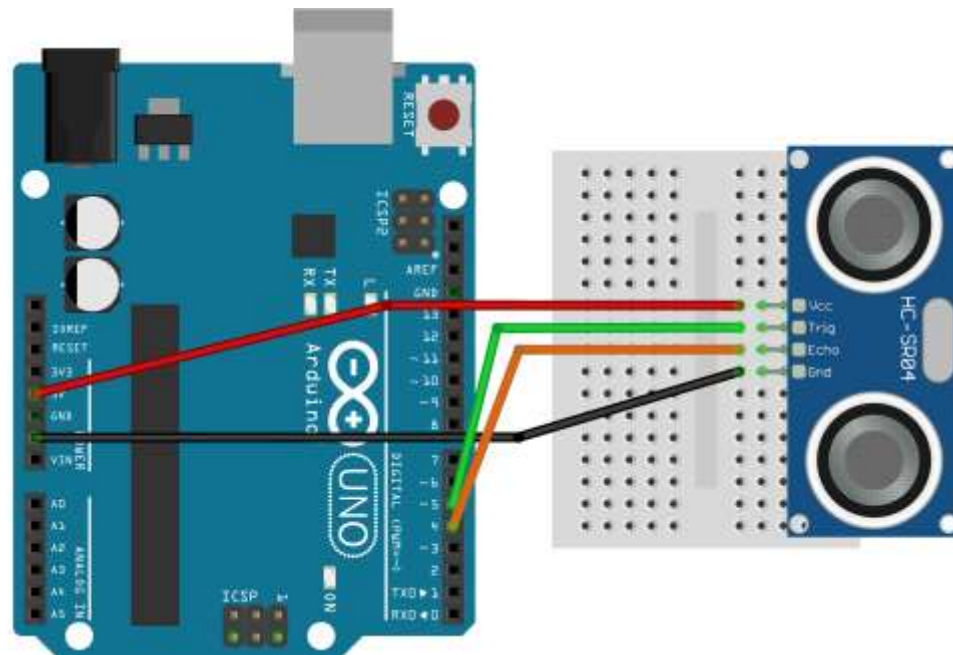


Figure 1: Hardware Set-Up for Ultrasonic Sensor Exercise

3.0 Simulink Set Up and Distance Measurement Testing

The complete Simulink diagram (i.e., algorithm design) for the hardware described above is given here. The "Ultrasonic Sensor" block can be found under the Sensors tab in the Simulink Support Package for Arduino Hardware. Both the Display and Scope blocks are located under the Sinks tab in Simulink.

The steps to set up the model are as follows:

1. Double-click the Ultrasonic Sensor block, then set the number of signal pins to 2.
2. Define the trigger pin as 4 and the echo pin as 5.
3. Select a sampling time/interval (e.g., 0.01 seconds). You should experiment with different values to determine the optimal setting.
4. Run the Simulink model and observe the measured distance to the object ahead on the Scope (see the scope output illustrated in the Simulink illustration).

Recall how to perform code-generation by visiting [HERE](#).

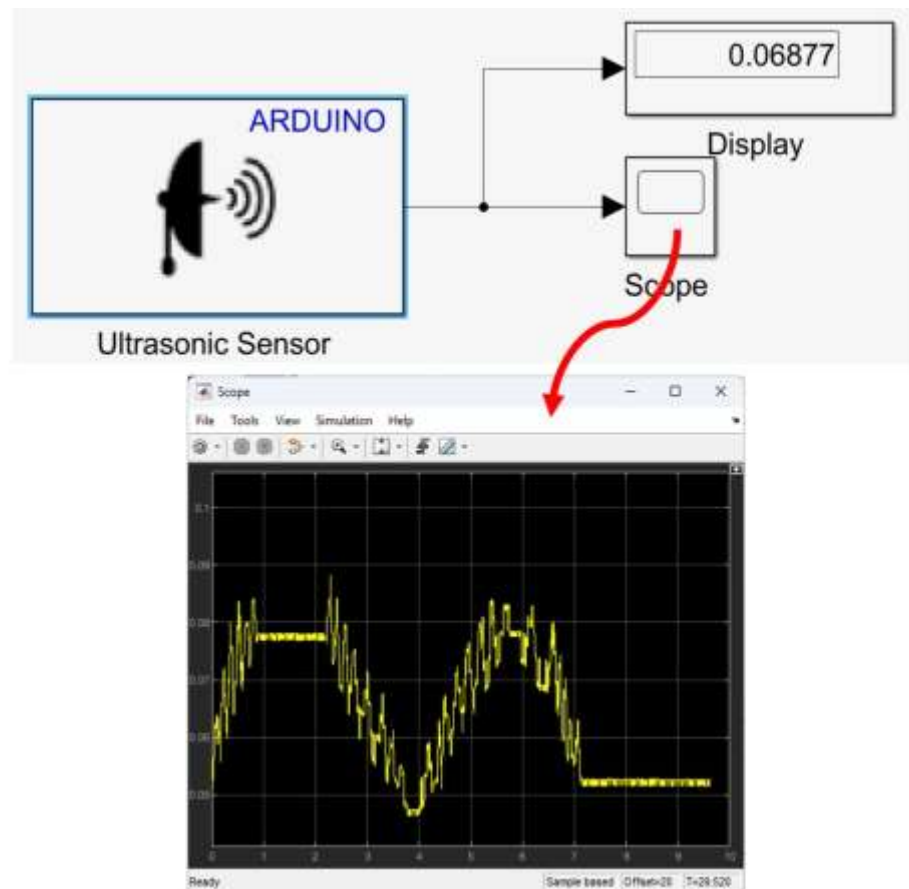


Figure 2: Simulink Set-Up for Ultrasonic Sensor Exercise

4.0 Last Good Value (LGV) Logic Filter and Testing

A last good value (LGV) logic filter is implemented. The LGV filter is used to maintain a valid output when the input signal becomes unreliable or invalid. It is particularly useful in situations where a sensor or input source might occasionally fail or produce noisy, erroneous data. The filter essentially "remembers" the last valid input value and continues to output this value until a new, valid reading is received. For example, in vehicles, if a sensor like an ultrasonic or radar sensor used for parking assistance fails or goes out of range temporarily, the LGV filter ensures the system still provides useful information based on the last good reading.

This setup can be configured to meet the operational requirements of the ultrasonic sensor, which has a measurement range of 0.02 to 4.00 meters. To implement this in Simulink use is made of the following blocks:

- The Switch block can be found under the Signal Routing section in Simulink. This block will be used to control the flow of data based on the sensor's output.

- The Memory block, located under the 'Discrete' section, will store the last valid reading when the sensor output is below a certain threshold.

The blocks are configured, with the set-up of the LGV filter logic as follows:

1. Set up the flow of signals as displayed in the Simulink diagram
2. Double-click the Switch block and set the Threshold to being greater than 0.1.
3. Add a mux and scopes to allow for the signals to be viewed in real-time.
4. Deploy the model and test the operation by operating the sensor within 0.1 meters and then beyond this.

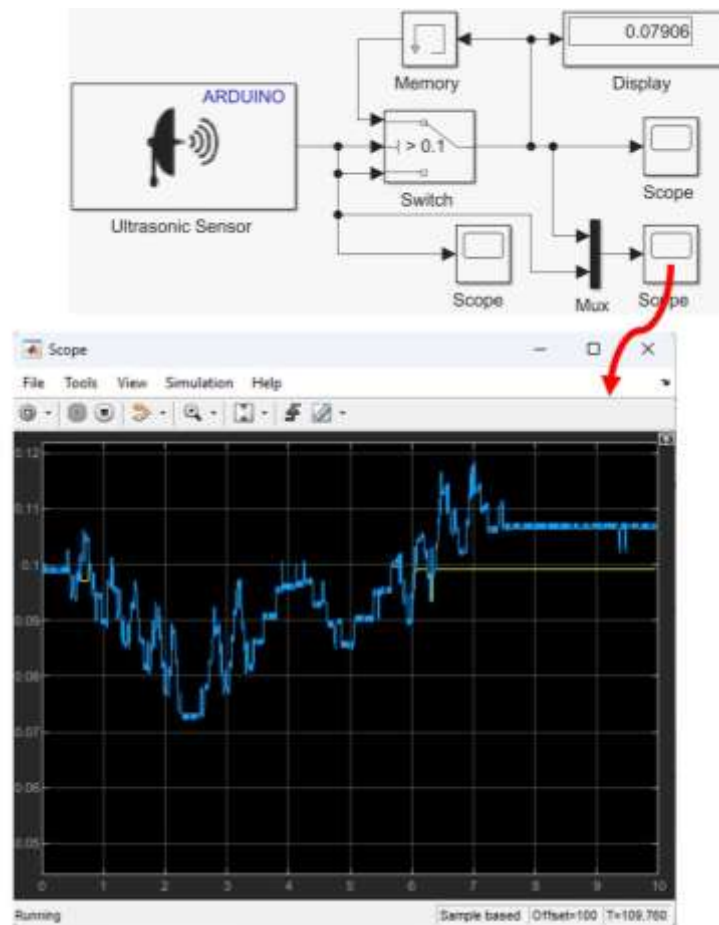


Figure 3: Last Good Value (LGV) Logic Filter Simulink Diagram and Testing

5.0 Additional Exercises

1. **Sampling Time Investigation:** Test at least three different sampling intervals and comment on the performance of the measurements.
2. **Static Accuracy Check:** Measure three known distances and compare against reference values.

3. **Improved Validity Logic:** Replace the $0.1m$ threshold with a full range check ($0.02m - 4.00m$) and compare behaviour.
4. **Moving Target Test:** Move an object towards and away from the sensor and observe signal smoothness and delay.

6.0 Concluding Remarks

The exercises have demonstrated practical distance measurement using an ultrasonic sensor integrated with Simulink and Arduino hardware. The exercise reinforces correct hardware configuration, real-time data acquisition, and validation of sensor signals.

The implementation of the last good value (LGV) filter highlights the importance of basic signal conditioning to improve measurement reliability. These principles form the foundation for closed-loop control systems, e.g., autonomous sensing applications for parking aids.

Recommended citation:

Pickering, J.E. (2026). Ultrasonic Sensor. ACE-Lab Teaching Notes. Available at: www.ace-lab.co.uk.