LESSON 09

Building a Scorpion Robot

Sensors, Autonomous Behavior & Iterative Design

LESSON OVERVIEW

Subject Area	STEAM, Robotics, Computer Science
Grade Level	Grades 6–8 (Adaptable)
Duration	3 Class Periods × 45–60 Minutes = 135–180 Minutes Total
Framework	5E Instructional Model (Engage → Explore → Explain → Elaborate → Evaluate)
Key Themes	Ultrasonic Sensor • Soil Moisture Sensor • Autonomous Behavior • Sensor Fusion • MakeCode
Materials	NEZHA Inventor's Kit V2 • MakeCode • Projector • Whiteboard • Optional: cardboard for obstacle course

LEARNING OBJECTIVES

01	Design and build a scorpion-themed robot using the NEZHA Inventor's Kit V2 with a functional tail/pincer mechanism.
02	Explain how ultrasonic sensors measure distance and how soil moisture sensors detect environmental conditions.
03	Program the robot to move autonomously and react to sensor input with conditional logic (if/else).
04	Troubleshoot hardware and software problems using systematic debugging strategies.
05	Communicate the design process, sensor logic, and findings clearly in a presentation or written report.

KEY VOCABULARY

Term	Definition
Ultrasonic Sensor	Sensor that measures distance using sound waves; detects obstacles at set thresholds.
Soil Moisture Sensor	Sensor that detects the moisture level of a surface or substance.

Conditional Logic	If-then-else programming structure that changes behavior based on sensor readings.
Threshold	A set value; when sensor reading crosses this value, the robot changes its behavior.
Autonomous	Capable of operating independently without direct human control, guided by sensors.
Feedback Loop	System where sensor output is used to continuously adjust the robot's behavior.
Actuator	Component that moves or controls a mechanism (motor, servo) in response to signals.
Debugging	Systematic process of finding and fixing errors in code or hardware connections.

⚡ 5E MODEL — LESSON PLAN

ENGAGE

Step	Activity	Key Focus
1	Show students a 60-sec video of a real scorpion moving and using its tail, then a robotic scorpion.	Spark curiosity and connect animal behavior to robotic design inspiration.
2	Ask students to brainstorm: 'What sensors does a scorpion use? How could we give a robot the same abilities?'	Build background knowledge and connect biology to sensor technology.
3	Introduce the NEZHA Inventor's Kit V2 components, highlighting the ultrasonic and moisture sensors.	Familiarize students with the tools and sensors they will use.
4	Issue the challenge: 'Build a scorpion robot that moves away from obstacles and reacts to touch!'	Provide a clear and biologically inspired engineering goal.

EXPLORE

Step	Activity	Key Focus
1	Divide students into groups; each group explores kit components and sensor wiring options.	Foster teamwork, collaborative exploration, and component identification.
2	Allow students to explore the ultrasonic sensor in MakeCode: read distance values and display them.	Encourage hands-on sensor testing and data interpretation.
3	Guide students through building a basic scorpion frame with a functional tail or pincer.	Teach students how to translate a biological form into an engineered structure.
4	Encourage students to experiment with sensor placement positions and observe reading accuracy.	Promote creativity, systematic testing, and engineering curiosity.

□ EXPLAIN

Step	Activity	Key Focus
1	Lead discussion about student sensor experiments: what readings triggered what behaviors?	Share and analyze student sensor data to build collective understanding.
2	Formal lesson: how ultrasonic sensors emit and receive sound pulses to calculate distance.	Build precise understanding of sensor operation and data interpretation.
3	Demonstrate conditional MakeCode: 'If distance < 20 cm → stop and reverse; else → move forward.'	Integrate conditional logic coding with physical sensor hardware.
4	Have students write their own conditional sensor code and test it on their scorpion robot.	Apply coding skills in a hands-on, immediately testable way.

□ ELABORATE

Step	Activity	Key Focus
1	Challenge students to add obstacle avoidance AND a second behavior triggered by the moisture sensor.	Encourage advanced multi-sensor programming and creative behavior design.
2	Have students test their robots in an obstacle course made from books or cardboard walls.	Apply problem-solving skills in a practical, unpredictable environment.
3	Allow students to share their code and designs with another group; give and receive peer feedback.	Develop communication, collaboration, and code-reading skills.

□ EVALUATE

Step	Activity	Key Focus
1	Observe students throughout to assess sensor understanding, debugging approach, and teamwork.	Conduct continuous formative assessment of process and reasoning.
2	Students complete a written reflection: describe a bug you found and how you fixed it.	Assess systematic debugging thinking and reflective learning.
3	Groups present robots: demonstrate obstacle avoidance live, explain sensor logic and threshold values.	Assess communication, technical accuracy, and programming understanding.
4	Rubric evaluation: sensor functionality, code quality, creativity, and presentation clarity.	Conduct summative assessment against explicit criteria.

☐ PERIOD-BY-PERIOD TEACHER & STUDENT SCRIPTS

PERIOD 1 — ENGAGE (40 min)

Time	Teacher Actions	Student Actions	Key Questions
0–8 min	Play scorpion video. Ask: 'How does the scorpion know something is near its tail before seeing it?'	List 2 ways a scorpion senses its environment in notebook.	What senses does a scorpion have that a basic robot does not?
8–20 min	Ultrasonic sensor demo: point at wall, move closer. Display distance values on micro:bit.	Read and record 5 distance measurements from different distances in a data table.	How does the sensor know how far away an object is without a ruler?
20–35 min	Groups begin building scorpion frame. Coach: 'Where will you mount the ultrasonic sensor?'	Assemble scorpion body and attach sensor in front-facing position.	How does the sensor need to be oriented for accurate distance readings?
35–40 min	Groups share sensor placement decision and reasoning with class.	Record predicted sensor range and planned obstacle threshold in notebook.	What threshold (cm) will make your robot stop before hitting an obstacle?

PERIOD 2 — EXPLORE (45 min)

Time	Teacher Actions	Student Actions	Key Questions
0–15 min	Guide conditional logic in MakeCode: introduce if/else blocks, sensor input, motor output.	Code basic behavior: 'if ultrasonic < 15 cm → reverse; else → forward.' Test.	What happens if your threshold is too high? Too low?
15–30 min	Introduce moisture sensor. Demo: touch the sensor pad; observe reading change.	Add moisture sensor behavior: 'if moisture > 50 → raise tail (motor B forward).'	How would you use a moisture sensor to protect the robot from water?
30–38 min	Obstacle course challenge: place 3 books as obstacles. Run robot and observe avoidance.	Run robot through obstacle course; count successful avoidances out of 5 attempts.	What triggered the most false stops? How would you fix the threshold?
38–45 min	Peer code review: swap code sheets with another group, find one improvement.	Read another group's code; write one suggestion for improving the threshold logic.	Is using a higher or lower threshold safer for the robot? Why?

PERIOD 3 — EXPLAIN → ELABORATE → EVALUATE (45 min)

Time	Teacher Actions	Student Actions	Key Questions
0–15 min	Mini-lecture: sensor fusion	Sketch a flowchart: sensor	How would combining

	(combining 2 sensors for richer behavior), feedback loops.	input → decision → motor output for their robot.	ultrasonic AND moisture readings create smarter behavior?
15–30 min	Final test: run robot through full obstacle course with both sensors active. 3 attempts.	Record: # obstacles avoided, # moisture triggers, # false stops per attempt.	Which sensor caused the most problems? How would you recalibrate it?
30–42 min	Group presentations: live demo of robot behavior, explain sensor thresholds and code logic.	Present 90-sec summary: sensor setup → threshold chosen → behavior observed → fix made.	What real-world robot uses a similar sensor system to the one you built?
42–45 min	Exit ticket: draw sensor flowchart for a new scenario; write if/else logic for it.	Complete individually and submit before leaving.	How would your scorpion robot behave differently if you added a light sensor?

□ DIFFERENTIATION & SCAFFOLDING

SUPPORT — Struggling Students	EXTENSION — Advanced Students
• Provide a pre-built scorpion frame so students focus only on sensor wiring and coding. • Offer a completed conditional code template students only modify threshold values. • Pair struggling students with a peer for the obstacle course testing phase. • Reduce to one sensor (ultrasonic only) for the core task.	• Challenge students to combine 3 sensors and program a priority hierarchy for conflicting signals. • Have advanced students research PID control and explain how it relates to their sensor feedback loop. • Design a scoring system for the obstacle course and compete for the highest avoidance score. • Write a formal engineering report: hypothesis, sensor data table, conclusion, real-world application.

□ ASSESSMENT BREAKDOWN

Assessment Component	Weight	Description
Exit Ticket	**20%**	Sensor flowchart + if/else logic for a new scenario (individual)
Obstacle Course	**25%**	Robot avoids obstacles in 3+ of 5 attempts with both sensors active
Code Quality	**20%**	Conditional logic is correct, threshold is justified, code is readable
Group Presentation	**20%**	Live demo with clear explanation of sensor logic and threshold reasoning
Engineering Notebook	**15%**	Sensor data table, flowchart, debugging notes, and

		reflection entry

TEACHER NOTES

- Adapt lesson complexity based on student abilities and available time.
- Encourage collaboration and peer support throughout the design process.
- Emphasize the value of iterative design: every 'failure' is data for the next attempt.
- Celebrate creativity, originality, and engineering effort regardless of final outcome.
- Provide regular formative check-ins using the engineering notebook as a formative tool.

REAL-WORLD CONNECTIONS

The engineering principles explored in this lesson appear throughout real-world industries:

- Robotics Engineering: Professional robots use the same motor control and sensor concepts in manufacturing, medicine, and exploration.
- Biomimicry Design: Engineers worldwide study animals to design better robots, vehicles, and structures.
- STEAM Integration: The connection between art, science, math, and engineering is a cornerstone of 21st-century innovation.
- Career Connections: Robotics engineers, software developers, mechanical engineers, and product designers all apply these skills daily.