

## LESSON 08

## The Walking Robot

## Center of Gravity, Gait Design &amp; Bipedal/Quadrupedal Locomotion

## □ LESSON OVERVIEW

<b>Subject Area</b>	STEM, Robotics, Computer Science, Math
<b>Grade Level</b>	Grades 6–8
<b>Duration</b>	3 Class Periods × 45 Minutes = 135 Minutes Total
<b>Framework</b>	5E Instructional Model (Engage → Explore → Explain → Elaborate → Evaluate)
<b>Key Themes</b>	Center of Gravity • Gait Patterns • Biomechanics • Stability • MakeCode Navigation
<b>Materials</b>	NEZHA Inventor's Kit V2 • MakeCode • Rulers • Tape (for maze)

## □ LEARNING OBJECTIVES

- 01 Design and build a walking robot using the NEZHA Inventor's Kit V2 that achieves stable forward locomotion.
- 02 Understand how the center of gravity position affects a robot's stability and walking performance.
- 03 Adjust the center of gravity systematically to maintain balance across different walking speeds.
- 04 Program the robot to walk forward and stop using a MakeCode sequence tied to motor timing.

## □ KEY VOCABULARY

Term	Definition
<b>Center of Gravity</b>	The single point where all of an object's weight acts; determines balance and tipping.
<b>Stability</b>	A robot's ability to maintain its upright position without falling during motion.
<b>Gait</b>	A specific pattern of leg movement used for walking (e.g., bipedal, quadrupedal).
<b>Bipedal</b>	Walking on two legs; requires dynamic balance adjustments during

	each step.
<b>Quadrupedal</b>	Walking on four legs; inherently more stable than bipedal locomotion.
<b>Balance</b>	Equilibrium maintained when the CG is above the base of support polygon.
<b>Biomechanics</b>	Application of mechanical principles to the study of living organism movement.
<b>Algorithm</b>	Ordered set of programming instructions the robot executes to perform a task.

⚡ 5E MODEL — LESSON PLAN

□ ENGAGE

Step	Activity	Key Focus
1	Show students a 90-sec video of a walking robot (bipedal like ASIMO or quadrupedal like Spot).	Spark curiosity and provide visual inspiration for the build challenge.
2	Ask students to brainstorm at least 3 different ways a robot could be designed to walk.	Encourage critical thinking about locomotion mechanics and creativity.
3	Discuss different animal gaits: bipedal (human, bird), quadrupedal (dog, horse). Connect to robotics.	Link biology and biomechanics to engineering design principles.

□ EXPLORE

Step	Activity	Key Focus
1	Divide students into groups of 2–3; each group selects a gait type (bipedal or quadrupedal).	Foster collaboration and intentional design decision-making.
2	Have each group build a walking robot using kit materials, following their design sketch.	Encourage hands-on exploration, assembly, and creative problem-solving.
3	Have students experiment with different leg lengths and attachment positions to affect walking.	Analyze how structural choices impact stability and step length.
4	Encourage students to adjust CG by moving components; observe effect on walking stability.	Develop problem-solving and engineering optimization skills.

□ EXPLAIN

Step	Activity	Key Focus
1	Have students share their designs and observations; identify the most common stability challenge.	Promote reflection, peer learning, and collaborative analysis.

2	Formal discussion: CG position relative to support polygon determines stability during each gait phase.	Build foundational knowledge in robotics and biomechanics.
3	Explain how to program the robot to walk forward (motor on) and stop (pause) using MakeCode.	Integrate coding into the robotics locomotion design process.

**ELABORATE**

Step	Activity	Key Focus
1	Have students modify their robots to improve walking consistency: adjust CG, leg timing, or step width.	Encourage iterative design and systematic problem-solving.
2	Challenge students to create a simple tape maze for their robot to navigate in a straight path.	Foster creativity, spatial planning, and programming precision.
3	Guide students in programming their robots to follow a specific 3-turn path through the maze.	Develop advanced programming, planning, and debugging skills.

**EVALUATE**

Step	Activity	Key Focus
1	Groups present their robots: explain design decisions, CG adjustments, and gait type chosen.	Assess creativity, problem-solving, and technical understanding.
2	Live demonstration: robot walks through the maze path; class evaluates stability and success.	Evaluate functionality and programming effectiveness objectively.
3	Students complete a written reflection: what was the biggest CG challenge? How did you solve it?	Assess understanding of concepts and the iterative design process.

**PERIOD-BY-PERIOD TEACHER & STUDENT SCRIPTS**

**PERIOD 1 — ENGAGE (40 min)**

Time	Teacher Actions	Student Actions	Key Questions
0–8 min	Show walking robot video. Ask: 'What is the hardest moment in the walking cycle for balance?'	Identify the moment of maximum instability and sketch it in the notebook.	When is a walking robot most likely to fall? Why?
8–20 min	Balance experiment: hold a ruler horizontally, add a weight, shift it. Where is the tipping point?	Record 3 CG positions and stability outcome (stable / unstable) in a table.	How does moving weight change where the robot wants to fall?
20–35 min	Groups choose gait type, sketch design, and identify	Produce labeled design sketch: gait type, leg	How will you ensure your CG stays above the support

	CG placement on sketch.	positions, CG location marked.	polygon while walking?
35–40 min	Groups share one design choice and reasoning. Identify potential stability risk.	Record predicted stability challenge and proposed solution in notebook.	What is the first thing you will test when you run the robot for the first time?

**PERIOD 2 — EXPLORE (45 min)**

Time	Teacher Actions	Student Actions	Key Questions
0–15 min	Coach build phase: check leg alignment, motor mount, and CG position relative to support base.	Assemble walking frame; verify legs reach the ground symmetrically.	Does your robot stand up without falling when you let go of it?
15–30 min	Guide MakeCode: motor A on (1s) → pause (0.5s) → motor B on (1s) → pause (0.5s) → loop.	Program and download walk sequence; run 3 walking tests. Record success/fail.	Does the robot walk in a straight line? What causes it to veer?
30–38 min	CG adjustment experiment: move one heavy component 1 cm; re-run walking test.	Record before/after CG position and stability score for each adjustment.	Did moving the CG improve or worsen stability? By how much?
38–45 min	Set up tape maze. Groups plan 3-turn route on paper before programming.	Sketch the maze path and write out the turn sequence before coding it.	How will your code know when to turn? What timing will you use?

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

Time	Teacher Actions	Student Actions	Key Questions
0–15 min	Mini-lecture: support polygon, dynamic vs. static stability, gait efficiency in nature.	Label support polygon on their robot sketch; classify their gait as static or dynamic.	Why is a 4-legged gait statically stable when a 2-legged gait usually is not?
15–30 min	Maze navigation challenge: each group runs robot through tape maze. 3 attempts allowed.	Program turn sequence; run robot through maze. Record which attempt succeeded.	How did you decide the duration of each turn command?
30–42 min	Group presentations: show robot walking, explain CG data, describe biggest design change.	Present 90-sec summary: original design → CG problem → fix → maze result.	How does your walking robot compare to an animal of similar size and gait?
42–45 min	Exit ticket: sketch CG position on 2 robots — one stable, one unstable. Explain why.	Complete individually and submit before leaving.	Could you program your robot to walk backwards? What would you change?

□ DIFFERENTIATION & SCAFFOLDING

SUPPORT — Struggling Students	EXTENSION — Advanced Students
<ul style="list-style-type: none"> <li>• Provide step-by-step illustrated build instructions for the quadrupedal gait design.</li> <li>• Pre-program a basic walk-forward template students only need to adjust timing for.</li> <li>• Allow partner support during the maze programming phase.</li> <li>• Simplify maze to a straight 1-meter path instead of a 3-turn route.</li> </ul>	<ul style="list-style-type: none"> <li>• Challenge students to design a robot that can walk on an uneven surface (textured board).</li> <li>• Have advanced students calculate the support polygon area for their robot at each gait phase.</li> <li>• Program the robot to navigate an autonomous maze using an ultrasonic obstacle sensor.</li> <li>• Compete in a classroom walking robot race across a measured track.</li> </ul>

□ ASSESSMENT BREAKDOWN

Assessment Component	Weight	Description
Exit Ticket	20%	CG diagram: stable vs. unstable positions labeled with explanation (individual)
Maze Navigation	25%	Robot successfully navigates 3-turn maze within 3 attempts
CG Adjustment Data	20%	Before/after CG positions with stability scores and justified change
Group Presentation	20%	Clear explanation of gait choice, CG data, and design iteration
Engineering Notebook	15%	Sketches, CG marks, maze route plan, 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.