Central York School District

Robotics

Overview

Grade Level: 9 -12

Course Description:
This course introduces students to the exciting world of robotics, combining principles of technology education, science, engineering, mathematics, and computer science. There is a heavy emphasis on problem-solving as students design, build, and program robots using sensors, motors, and controllers to complete real-world challenges. Through hands-on projects, students learn about mechanical design, electronics, and coding while developing teamwork, creativity, and critical thinking skills.

Course Big Ideas:

- Robots combine mechanical structures, electronics, sensors, and programming that must work together as a unified system.
- Effective robotic solutions are developed through cycles of designing, building, testing, analyzing, and improving.
- Robots rely on algorithms and code to make decisions, respond to sensor input, and perform tasks without direct human control.
- Motors, gears, and drivetrains translate electrical energy into controlled mechanical motion with tradeoffs in speed, torque, and efficiency.
- Real-world challenges can be addressed by defining constraints, brainstorming solutions, and building functional prototypes.
- Successful robots must perform consistently, requiring careful calibration, testing, and debugging.
- Robotics projects depend on teamwork, communication, and shared problem-solving, similar to real engineering environments.

Course Essential Questions:

- How do mechanical systems, electronics, and programming integrate to create functional and autonomous robots?
- How does the engineering design process guide the development, testing, and improvement of robotic systems?
- How do robots use sensors and programming to perceive and respond to the real world?
- What trade-offs must engineers consider when designing robots to meet specific goals and constraints?
- How do collaboration, communication, and iteration improve the effectiveness of engineering solutions?

Course Competencies:

- Investigation
- Analysis
- Compare/Contrast
- Contextualization
- Synthesis
- Error Analysis

Course Assessments:

- Unit written assessments
- Unit performance assessments

Central York School District

Robotics

Grade Level: 9-12

Unit 1: Introduction to Robotics

Time: 12 days

Standards:
PA Science, Technology & Engineering, Environmental Literacy and Sustainability (STEELS) Standards: • 3.5.9-12.Q: Implement and critique principles, elements, and factors of design. • 3.5.9-12.U: Evaluate and define the purpose of a design. • 3.5.9-12.X: Implement the best possible solution to a design using an explicit process. • 3.5.9-12.Y: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. • 3.5.9-12.AA: Safely apply an appropriate range of making skills to a design thinking process.
Know:
• Know the proper and safe use of tools. • Know the use and function of 6 simple machines. • Know that motors run in continuous mode and can move large loads. • Know that servos are for precise motion and move small loads. • Know how to connect a computer to the control hub for the purpose of programming/coding.
Understanding/Key Learning:
• Safely use tools, materials, and technology while designing, building, and testing products or systems.Apply the engineering design process to identify problems, develop solutions, build prototypes, and improve designs through testing and feedback. • Analyze and solve complex real-world problems by breaking them into smaller parts and evaluating multiple possible solutions. • Use technology effectively to learn, communicate, and gather feedback that enhances engineering and design solutions.
Do:
• Demonstrate the proper and safe use of tools. • Identify and explain the use and function of the six simple machines. • Describe how motors operate in continuous mode and use them to move large loads. • Describe how servos provide precise motion and use them to move small loads. • Connect a computer to the control hub in order to program and upload code. • Program and test the control hub using a connected computer.

Unit Essential Questions:

- What is a robot?
- How do robots interact with their environment?
- How can a robot complete a task?

Lesson Essential Questions:

- What are the parts of our robotics kit and their usage and functions?
- How can we assemble the parts in the robotics kit to perform a task?
- How do simple machines play a role in robotics?
- How do we control actuators?
- How do we code a program to control actuators independently?
- How do we code a program to control actuators using a gamepad/external input?

Materials/Resources:
- Rev robotics hardware kits
- iPad or laptop computer

Vocabulary:

15 mm T-slotted Extrusion	Structural brackets	Nyloc nuts	Motion brackets
M3	Hardware	Actuator brackets	Spacers
Hex shaft	Pre-loading brackets	Shaft collars	Pillow blocks
Bearings	Pitch	OpMode	Control hub
Driver hub	Simple machine	Lever	Wheel & axle
Pulley	Screw	Wedge	Inclined plane
Actuators	Pinion	PWM	DC motor
Servo	Wiring diagram	Encoder	Testbed
Through-bore	Continuous mode	Angular mode	

Assessments:
- Formative assessments
- Unit 1 Key Terms summative assessment
- Unit 1 performative assessment

Central York School District

Robotics

Grade Level: 9-12

Unit 2: Mechanical Motion - Gearing and Gear Trains

Time: 18 days

Standards:
PA Science, Technology & Engineering, Environmental Literacy and Sustainability (STEELS) Standards: • 3.5.9-12.J: Synthesize data and analyze trends to make decisions about technological products, systems, or processes. • 3.5.9-12.Q: Implement and critique principles, elements, and factors of design. • 3.5.9-12.U: Evaluate and define the purpose of a design. • 3.5.9-12.W: Optimize a design by addressing desired qualities within criteria and constraints while considering trade-offs. • 3.5.9-12.X: Implement the best possible solution to a design using an explicit process. • 3.5.9-12.Y: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. • 3.5.9-12.AA: Safely apply an appropriate range of making skills to a design thinking process. • 3.5.9-12PP: Demonstrate the use of conceptual, graphical, virtual, mathematical, and physical modeling to identify conflicting considerations before the entire system is developed and to aid in design decision making.
Know:
• Know the steps of the engineering design process, including how to identify criteria, constraints, and generate solutions through ideation. • Know how gear ratios affect mechanical systems, including changes in speed, torque, force, and mechanical advantage. • Know the function and purpose of different gear systems, including spur gears, idler gears, gear trains, compound gearing, and gearboxes. • Know how to interpret and use gear measurements such as pitch diameter (PD) and outside diameter (OD) in design situations. • Know how motors behave in mechanical systems, including free speed, stall torque, and the effects of motor stalling. • Know how to evaluate system performance using efficiency and apply an appropriate safety factor in design decisions. • Know how planetary gear systems work, including the roles of the sun gear, planet gears, outer ring gear, and carrier. • Know how compound reduction and multi-stage gear trains are used to increase torque or reduce speed in mechanical designs.
Understanding/Key Learning:
• Understand how the engineering design process guides problem-solving by using criteria, constraints, and ideation to develop effective solutions. • Understand how gear systems (including gear ratios, gear trains, compound gearing, and planetary gears) change mechanical performance by affecting speed, torque, force, and mechanical advantage. • Understand how to evaluate and design mechanical systems using key performance factors such as motor

characteristics (free speed, stall torque, stalling), efficiency, safety factor, and gear geometry (pitch diameter and outside diameter).
Do:
• Explain and apply the steps of the engineering design process, including identifying criteria, constraints, and generating ideas (ideation). • Calculate and analyze gear ratios, gear trains, and compound gearing to determine changes in speed, torque, force, and mechanical advantage. • Identify and describe the function of common gear system components, including spur gears, idler gears, gearboxes, and planetary gear systems. • Measure and apply gear specifications such as pitch diameter (PD) and outside diameter (OD) when designing mechanical systems. • Evaluate motor performance using concepts such as free speed, stall torque, motor stalling, efficiency, and safety factor. • Explain how planetary gearing works and identify the roles of the sun gear, planet gears, outer gear ring, and carrier. • Design and test gear-driven mechanisms to achieve a desired speed reduction, torque increase, or motion transfer. • Compare different gearing solutions and justify design decisions based on performance requirements and engineering constraints.

Unit Essential Questions:
- How does the engineering design process help engineers solve problems using criteria, constraints, and iterative ideation?
- How do different gear systems (spur, compound, planetary, and gear trains) change speed, torque, force, and mechanical advantage in a mechanical system?
- How do motor characteristics such as free speed, stall torque, and stalling behavior influence the performance and design of gear-driven systems?
- How do engineers choose and evaluate mechanical designs using efficiency, safety factors, and gear geometry to meet performance requirements?

Lesson Essential Questions:
- How do gears interact?
- How do gears multiply force and speed and what are the tradeoffs?
- How can gears be used to span larger distances than the outer diameter of the two gears?
- What is a compound gear system?
- How efficient are gears?
- How does the stall torque of a motor affect the design considerations we make when using gears?
- What is the safety factor and how is it incorporated into your design?

Materials/Resources:
- Rev robotics hardware kits
- iPad or laptop computer

Vocabulary:

Engineering design process	Criteria	Constraints	Ideate
Gear Ratio	Pitch diameter (PD)	Outside diameter (OD)	Spur gear
Gear train	Speed	Torque	Idler gear
Compound gearing	Reduction	Mechanical advantage	Compound reduction

Motor stalling	Free speed	Stall torque	Force
Efficiency	Safety factor	Gearbox	Planetary gearing
Sun gear	Planet gears	Outer gear ring	Carrier

Assessments:

- Formative assessments
- Unit 2 Key Terms summative assessment
- Unit 2 Performative assessment

Central York School District

Robotics

Grade Level: 9-12

Unit 3: Robots in Motion

Time: 15 days

Standards:

PA Science, Technology & Engineering, Environmental Literacy and Sustainability (STEELS) Standards:

- 3.5.9-12.J: Synthesize data and analyze trends to make decisions about technological products, systems, or processes.
- 3.5.9-12.Q: Implement and critique principles, elements, and factors of design.
- 3.5.9-12.U: Evaluate and define the purpose of a design.
- 3.5.9-12.W: Optimize a design by addressing desired qualities within criteria and constraints while considering trade-offs.
- 3.5.9-12.X: Implement the best possible solution to a design using an explicit process.
- 3.5.9-12.Y: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering.
- 3.5.9-12.AA: Safely apply an appropriate range of making skills to a design thinking process.
- 3.5.9-12PP: Demonstrate the use of conceptual, graphical, virtual, mathematical, and physical modeling to identify conflicting considerations before the entire system is developed and to aid in design decision making.

Know:

- Know that a Boolean is a data type and is used for true/false decisions in programming.
- Know that a Float is a data type with decimal values for more precise measurements and when to use it in programming.
- Know that a drivetrain is how robots transfer power from motors to produce robot movement.
- The difference between a differential drivetrain and other drive systems, including how independent left/right control allows turning.
- What an omnidirectional drivetrain is and how it allows movement in any direction using specialized wheels.
- Know the difference between a passive mechanism and an active mechanism, including whether or not motors are used.
- Know that a variable is used in programming to store data that can change during a program.
- Know that a constant is used in programming and it represents a fixed value that does not change during program execution.
- Know that telemetry is real-time data provided about how a robot is performing and is used to monitor and adjust behavior.

Understanding/Key Learning:

- Robot movement depends on drivetrain design and mechanism selection. Different drivetrains and mechanisms provide different capabilities, advantages, and limitations that affect how a robot performs tasks.
- Programming uses data types and variables to control robot behavior. Booleans, floats, variables, and constants allow programmers to store, process, and use information to make decisions and control actions.
- Mechanisms can be active or passive depending on how they operate. Engineers select and design mechanisms based on the task requirements, available power, and desired performance.
- Telemetry provides valuable feedback for evaluating and improving robot performance. Real-time data helps

operators and programmers monitor robot systems, troubleshoot issues, and make informed decisions.
Do:
• Define and identify a Boolean and use it to make true/false decisions in a program. • Select and apply float values when working with measurements that require decimals and precision. • Explain and describe how a drivetrain works to transfer power from motors to robot movement. • Compare and distinguish a differential drivetrain from other drive systems and explain how left/right motor control produces turning. • Describe and demonstrate how an omnidirectional drivetrain moves in multiple directions using specialized wheels. • Differentiate between passive and active mechanisms and identify whether a system uses motors or external force. • Create and use variables in a program to store and update changing data. • Recognize and apply constants in programming to represent values that remain fixed. • Collect and interpret telemetry data to monitor robot performance in real time.

Unit Essential Questions:

- How do different drivetrain designs and mechanisms affect the movement and performance of a robot?
- How do programmers use data types, variables, and constants to control robot behavior and solve problems?
- What are the differences between active and passive mechanisms, and how do engineers choose the best solution for a specific task?
- How can telemetry and other feedback data be used to monitor, evaluate, and improve robot performance?

Lesson Essential Questions:

- What a Boolean is and how it is used to make true/false decisions in programming?
- What a Float is and when to use decimal values for more precise measurements?
- What a drivetrain is and how it transfers power from motors to robot movement?
- The difference between a differential drivetrain and other drive systems, including how independent left/right control allows turning?
- What an omnidirectional drivetrain is and how it allows movement in any direction using specialized wheels?
- The difference between a passive mechanism and an active mechanism, including whether or not motors are used?
- What a variable is in programming and how it stores data that can change during a program?
- What a constant is and how it represents a fixed value that does not change during program execution?
- What telemetry is and how robots use it to send real-time data to a computer or driver station?

Materials/Resources:

- Rev robotics hardware kits
- iPad or laptop computer

Vocabulary:

Boolean	Float	Drivetrain	Differential drivetrain
Omnidirectional drivetrain	Passive mechanism	Active mechanism	Variable
Constant variable	Telemetry		

Assessments:

- Formative assessments
- Unit 3 Key Terms summative assessment
- Unit 3 Performative assessment

Central York School District

Robotics

Grade Level: 9-12

Unit 4: Thinking Like a Robot

Time: 18 days

Standards:
PA Science, Technology & Engineering, Environmental Literacy and Sustainability (STEELS) Standards: • 3.5.9-12.J: Synthesize data and analyze trends to make decisions about technological products, systems, or processes. • 3.5.9-12.Q: Implement and critique principles, elements, and factors of design. • 3.5.9-12.U: Evaluate and define the purpose of a design. • 3.5.9-12.W: Optimize a design by addressing desired qualities within criteria and constraints while considering trade-offs. • 3.5.9-12.X: Implement the best possible solution to a design using an explicit process. • 3.5.9-12.Y: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. • 3.5.9-12.AA: Safely apply an appropriate range of making skills to a design thinking process. • 3.5.9-12PP: Demonstrate the use of conceptual, graphical, virtual, mathematical, and physical modeling to identify conflicting considerations before the entire system is developed and to aid in design decision making.
Know:
• Know that pseudocode is used to plan and organize a program before writing code. • Know that an autonomous robot can perform tasks without direct human control. • Know that sleep commands control the timing and sequence of actions within a program. • Know that loops allow a set of instructions to repeat automatically. • Know that digital sensors provide information in binary states, such as on/off or pressed/not pressed. • Know that analog sensors provide a range of values that represent changing conditions. • Know that I2C sensors communicate with the robot controller using a shared data protocol. • Know that the SCL (Serial Clock) line synchronizes communication between devices on an I2C bus. • Know that the SDA (Serial Data) line carries data between devices on an I2C bus. • Know that a touch sensor detects physical contact and can be used as an input for decision-making. • Know that active-low behavior means a signal is considered active when its voltage is low rather than high. • Know that sensor inputs can be used to trigger actions and influence robot behavior. • Know that exit conditions determine when a loop or autonomous action should stop. • Know that robots use sensor data to make decisions and respond to their environment. • Know that combining timing, loops, and sensor feedback allows robots to perform complex autonomous tasks. • Know that testing and troubleshooting are necessary to ensure sensors and programs operate correctly.
Understanding/Key Learning:
• Autonomous robots rely on programming logic, timing, and decision-making structures to perform tasks independently. • Sensors collect data that allows robots to perceive and respond to their environment. • Communication systems such as I^2C allow controllers and sensors to exchange information effectively.

• Well-designed programs use loops, conditions, and exit criteria to create efficient and reliable robot behavior.
Do:
• Develop and interpret pseudocode to plan and communicate the logic of a robot program before coding. • Design, test, and refine an autonomous robot that can complete tasks without driver control. • Use sleep commands to control timing and sequencing within a robot program. • Create and apply loops to repeat actions efficiently and reduce unnecessary code. • Identify and use digital sensors and analog sensors to collect information from the robot's environment. • Connect, configure, and troubleshoot I^2C sensors using the SCL (Serial Clock) and SDA (Serial Data) communication lines. • Program and test a touch sensor to detect user input or physical contact. • Explain and implement active-low behavior when programming digital inputs and outputs. • Create programs that respond appropriately to sensor data and changing environmental conditions. • Define and use an exit condition to stop loops or end autonomous actions when specific criteria are met.

Unit Essential Questions:

- How do programmers use pseudocode and programming structures to create autonomous robot behaviors?
- How do sensors provide information that allows robots to make decisions and interact with their environment?
- How do communication protocols such as I^2C enable sensors and controllers to exchange data?
- How can loops, timing commands, and exit conditions be used to create reliable and efficient robot programs?

Lesson Essential Questions:

- How can pseudocode help plan a robot program before writing code?
- What is an autonomous robot, and how does it differ from a driver-controlled robot?
- How does a sleep command affect the execution of a robot program?
- When should timing delays be used in an autonomous routine?
- Why are loops useful in robot programming?
- What are the differences between finite and infinite loops?
- What information can a digital sensor provide to a robot?
- How does a robot interpret digital sensor inputs?
- How can digital sensor data be used to trigger robot actions?
- How does an analog sensor differ from a digital sensor?
- How can analog sensor values be used in decision-making?
- What is the purpose of the I2C communication protocol?
- How do SCL and SDA work together to transfer data?
- Why is I2C commonly used for robotics sensors?
- How does a touch sensor detect physical contact?
- How can a touch sensor be used to control robot behavior?
- What does active-low behavior mean in a robotic system?
- How does active-low logic differ from active-high logic?
- What is an exit condition, and why is it important in programming?
- How can exit conditions prevent unintended robot behavior?
- How do sensors and exit conditions work together to control loops and autonomous actions?
- What programming structures allow robots to react to changing conditions?
- How can multiple sensors be combined to improve robot performance?
- How can programmers determine whether a sensor is functioning correctly?
- What strategies can be used to debug autonomous programs?

Materials/Resources:

- Rev robotics hardware kits
- iPad or laptop computer

Vocabulary:

Pseudocode	Autonomous robot	Sleep	Loops
Digital sensor	Analog sensor	12C sensor	Active-Low behavior
SCL (serial clock)	SDA (serial data)	Touch sensor	Exit condition
Bus			

Assessments:

- Formative assessments
- Unit 4 Key Terms summative assessment
- Unit 4 Performative assessment

Central York School District

Robotics

Grade Level: 9-12

Unit 5: Sensor Exploration

Time: 25 days

Standards:
PA Science, Technology & Engineering, Environmental Literacy and Sustainability (STEELS) Standards: • 3.5.9-12.J: Synthesize data and analyze trends to make decisions about technological products, systems, or processes. • 3.5.9-12.Q: Implement and critique principles, elements, and factors of design. • 3.5.9-12.U: Evaluate and define the purpose of a design. • 3.5.9-12.W: Optimize a design by addressing desired qualities within criteria and constraints while considering trade-offs. • 3.5.9-12.X: Implement the best possible solution to a design using an explicit process. • 3.5.9-12.Y: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. • 3.5.9-12.AA: Safely apply an appropriate range of making skills to a design thinking process. • 3.5.9-12PP: Demonstrate the use of conceptual, graphical, virtual, mathematical, and physical modeling to identify conflicting considerations before the entire system is developed and to aid in design decision making.
Know:
• Know that different sensor functions allow robots to detect contact, motion, position, and orientation to control behavior. • Know that a limit switch is a digital sensor that detects when a mechanism reaches a physical stopping point or boundary and can be used in programming to stop a robot. • Know that an incremental encoder measures changes in position over time relative to a starting point and can be used in programming to elicit desired behaviors. • Know that an absolute encoder provides a specific position value that remains accurate even after power loss. • Know that an IMU (Inertial Measurement Unit) measures a robot's movement and orientation using multiple built-in sensors. • Know that an accelerometer measures changes in linear motion (acceleration) in one or more directions and can be used in programming to elicit desired behaviors. • Know that a gyroscope measures rotational motion and angular velocity and can be used in programming to elicit desired behaviors. • Know that yaw describes rotation left or right around a vertical axis and can be used in programming to elicit desired behaviors. • Know that pitch describes forward and backward tilt around a horizontal axis and can be used in programming to elicit desired behaviors. • Know that roll describes side-to-side tilt around a front-to-back axis and can be used in programming to elicit desired behaviors.
Understanding/Key Learning:
• Robots rely on a combination of sensors to detect position, motion, and orientation, allowing them to interact

accurately with their environment.

- Different sensors (limit switches, encoders, IMUs) provide different types of data, and each has specific strengths depending on the task.
- Understanding how to interpret sensor data is essential for controlling robot movement, navigation, and decision-making.
- Combining multiple sensors improves accuracy and reliability in robotic systems by providing a more complete picture of movement and orientation.

Do:

- Identify and describe how a limit switch functions as a boundary-detecting digital sensor in a robotic system.
- Classify different sensor functions and determine how each is used to support robot decision-making and control.
- Compare and explain the differences between an incremental encoder and an absolute encoder in terms of how position is measured and maintained.
- Interpret encoder data to determine movement, position change, or exact shaft position.
- Explain how an IMU (Inertial Measurement Unit) combines multiple sensors to measure motion and orientation.
- Analyze data from an accelerometer to determine changes in linear motion.
- Analyze data from a gyroscope to determine rotational movement and angular velocity.
- Describe and differentiate yaw, pitch, and roll as the three types of rotational orientation in a robot.
- Apply sensor data (IMU and encoders) to help a robot maintain orientation or follow a controlled path.
- Evaluate which sensor type is most appropriate for a given robotics task involving motion, position, or orientation.

Unit Essential Questions:

- How do different sensors help a robot understand its position, movement, and orientation in real time?
- When is it better to use relative sensing (incremental encoders) versus absolute sensing (absolute encoders)?
- How does combining IMU data (accelerometer + gyroscope) improve robot navigation and control accuracy?

Lesson Essential Questions:

- How does a limit switch help a robot detect physical boundaries and prevent damage or over-travel?
- How do different sensor types (encoders, IMUs, limit switches) collect and communicate information about a robot's motion and position?
- What is the difference between incremental and absolute encoders, and how does each affect how a robot tracks position?
- How can encoder data be interpreted to measure distance, rotation, or position changes in a robot system?
- How does an IMU combine accelerometer and gyroscope data to determine a robot's movement and orientation?
- How do accelerometers and gyroscopes measure different aspects of motion, and why are both important?
- How do yaw, pitch, and roll describe a robot's orientation in 3D space?
- How can understanding yaw, pitch, and roll improve a robot's ability to navigate and stay balanced?
- How do multiple sensors work together to improve accuracy and reliability in robotic control systems?
- How do engineers decide which sensor or combination of sensors is best for a specific robotic task?

Materials/Resources:

- Rev robotics hardware kits
- iPad or laptop computer

Vocabulary:

Limit switch	Functions	Incremental encoder	Absolute encoder
IMU	Accelerometer	Gyroscope	Yaw
Pitch	Roll		

Assessments:

- Formative assessments
- Unit 5 Key Terms summative assessment
- Unit 5 Performative assessment