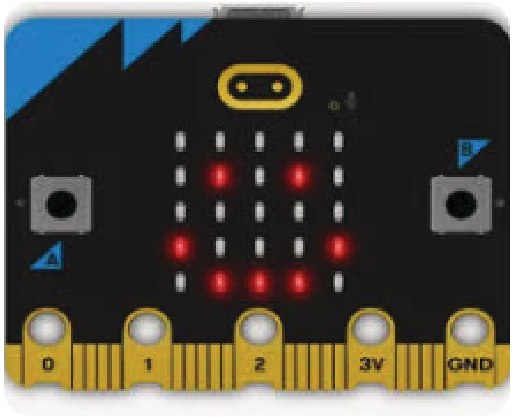
USING GAMES TO INSPIRE LEARNING

MICRO:BIT

MICROCONTROLLER



A pocket-sized powerhouse for coding, creativity, and real-world problem-solving—bringing hands-on learning to the classroom!

Overview:

The micro:bit is a small, programmable microcontroller that introduces students to coding and computational thinking through hands-on, interactive projects. Designed for ease of use, the micro:bit allows students to experiment with digital creativity, physical computing, and problem-solving. With its built-in sensors, LED display, and wireless communication capabilities, the micro:bit provides an accessible entry point into coding and electronics.

Time to Play:

30–45 minutes per session

Can be used as a one-time lesson or integrated into a multi-week coding unit

Objective:

Students use the micro:bit to design and program interactive projects, reinforcing coding concepts and computational thinking in an engaging, hands-on way.

Learning Goals:

Computational Thinking & Coding – Develop problem-solving skills by writing and debugging programs.

Physical Computing – Learn how hardware and software interact through micro:bit's sensors and inputs.

Creative Expression – Encourage innovation through coding projects that integrate real-world applications.

Data & Sensors – Explore data collection using built-in sensors like temperature, light, and motion.

Teaching Ideas

- **Interactive Name Badge** – Program the LED matrix to display scrolling text for personalized messages.
- **Step Counter Project** – Use the accelerometer to track movement and count steps.
- **Temperature Sensor** – Measure and display ambient temperature changes using built-in sensors.
- **Wireless Messaging** – Explore radio communication by sending and receiving messages between micro:bits.

Suggested Classroom Adaptations

- **Individual Play**– Students program micro:bits independently to explore basic functions.
- **Pair Programming**– Encourage teamwork by having students troubleshoot and refine each other's code.
- **STEM Challenge**– Assign open-ended projects where students solve real-world problems using the micro:bit.

Next Steps

- Consider making micro:bits a permanent feature in a Makerspace or Brain Break Corner, allowing students to engage with coding in an informal, exploratory setting.
- Integrate the micro:bit into a broader STEM unit, incorporating discussions on hardware, software, and real-world applications.
- Introduce block-based programming (MakeCode) before transitioning to Python for more advanced coding projects.

CSTA Standards:

Algorithms & Programming

- **2-AP-11**: Create programs that use variables to store and modify data.
  - *Students write code that interacts with sensors, storing and manipulating real-time data.*
- **2-AP-12**: Design and iteratively develop programs that combine control structures, including loops and conditionals.
  - *Encourages students to build interactive, decision-based projects using event-driven programming.*

Computing Systems

- **2-CS-02**: Design projects that combine hardware and software components to collect and exchange data.
  - *Introduces physical computing through micro:bit's sensor integration.*

Impacts of Computing

- **3A-IC-28**: Discuss the impact of computing innovations on equity, access, and culture.
  - *Encourages discussions on how microcontrollers are used in assistive technologies and STEM fields.*

****Connection to Real-World Computing**

The micro:bit introduces students to physical computing concepts used in robotics, automation, and IoT (Internet of Things). By engaging in hands-on coding experiences, students see how computational thinking applies to real-world problem-solving, from wearable technology to environmental monitoring. Understanding how hardware and software interact prepares students for future STEM careers in engineering, technology, and digital design.