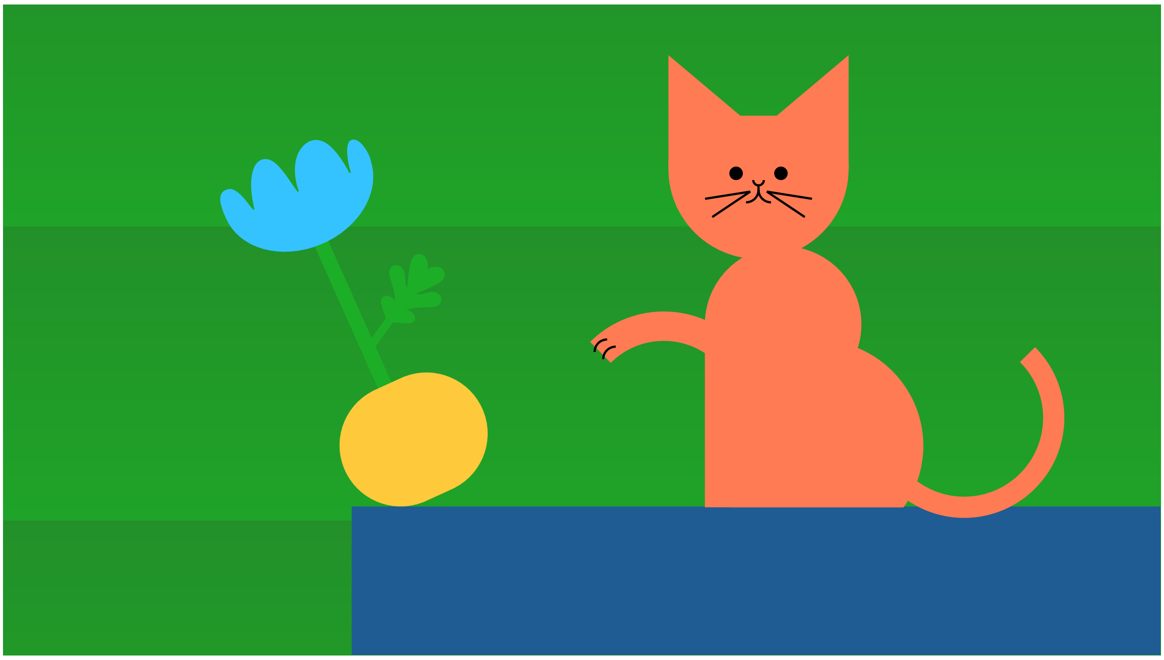

Build a Fall Detector

Learn how to measure total acceleration with your phone's motion sensor to create a system that can detect a sudden fall.

Courses

- Grades 3-12

Materials

- Cell phone, tablet, or computer
- Internet connection

Educational Objectives

- Understand the concepts of “if-else” and the “Cartesian plane.”
- Create a technological object (prototype) using a device.
- Identify relationships between technology and the surrounding world.
- Evaluate personal and others’ work.
- Engage in dialogue and reflection on improvement ideas.

Start (10 minutes) - The Physics of a Fall

1. Welcome students and introduce the day’s activity: **"Today we will learn to prototype a ‘fall detector’ with our phones."**
2. Start by asking the class: **"How do smartwatches or phones sometimes know if a person has had a fall?"** Guide the discussion toward the idea of detecting sudden motion.
3. Explain that the phone's **accelerometer** is the key. Introduce the core challenge: a fall can happen in any direction, so how can our program detect the overall *magnitude* of the motion? This leads perfectly into the concept of combining the X, Y, and Z axes to get a total motion value.

How does the phone detect a fall?

Your phone's **accelerometer** is a powerful motion sensor. It constantly measures acceleration in three directions or "axes": left-right (X-axis), forward-backward (Y-axis), and up-down (Z-axis). When a phone is dropped, it experiences a sharp, sudden change in acceleration. To detect a fall, we don't care about the *direction* of the fall, only that it was a powerful jolt. To measure this total jolt, we can take the **absolute value** (the positive version of the number) of the motion on each axis and add them together. This gives us a single number representing the total magnitude of the motion.

Let's get to work!

Protobject simplifies this process for us with the `generalMotion` variable. This special variable automatically does the math for us: it sums the absolute values of the X, Y, and Z axes in real-time. All we need to do is set a threshold. We'll tell our program: **IF** the `generalMotion` value exceeds a certain number (like 150), it means a sharp jolt has occurred, and we should trigger an alarm sound.

Development (20-30 minutes) - Building the Detector

1. Now that the students understand the logic of measuring total motion, it's time to build the fall detector.
2. Lead them through **the instructions for creating the prototype and programming the detection logic**, as detailed in the hands-on section below. Emphasize how the 'IF' statement is used to check if the total motion exceeds a certain threshold.
3. Have students test their prototypes by carefully but quickly shaking the device, or by dropping it a very short distance onto a soft surface like a book or pillow.

Closure (5-10 minutes) - Real-World Reliability

1. Once the fall detectors are working, it's time to think about the real-world implications and challenges of such a device.
2. Use the final section to guide a critical discussion about the reliability of their fall detector. This encourages them to think like engineers about false positives and improving their designs.

Reflect

You've built a device with real-world applications. Now, let's think like engineers.

- How reliable do you think this fall detector is? Could a false alarm be triggered just by running, jumping, or putting the phone down too hard?
- Would you trust this to be used in a real-life situation for an elderly person? What improvements would you need to make first?