Explaining Arduino's Voltage and Current Limits

An Arduino board has limitations on how much current and voltage it can handle safely. Let’s break down the key aspects and answer your question.

1. Voltage Limits

- The **voltage regulator** on the Arduino is designed to step down an external voltage (e.g., from a 9V or 12V adapter) to 5V or 3.3V, which powers the microcontroller and connected devices.
- The **voltage remains constant** as long as the regulator is not overloaded or damaged.
- Typical external voltage input range: **7-12V**.
- Exceeding the voltage limit (e.g., providing 20V) can damage the regulator or the microcontroller.

2. Current Limits

- **Per Pin Current Limit:**
  - Each I/O pin on the Arduino can provide a maximum of **20 mA** safely (absolute max is 40 mA, but that can damage the chip if sustained).
  - Exceeding this limit on a single pin may cause it to fail or malfunction.
- **Total Current Limit:**
  - The microcontroller has an overall limit on how much current can pass through all its pins combined. For most Arduino boards, this is **200-400 mA**.
  - Exceeding this total current limit can damage the voltage regulator or the microcontroller itself.

3. Impact of Reaching the Current Limit

- If the total current drawn by sensors, actuators, and other connected devices exceeds the Arduino’s capacity:
  1. **Voltage remains constant initially**, but the regulator may overheat or shut down to prevent damage.
  2. **Sensors or devices may stop working** due to insufficient current.
  3. The Arduino may behave unpredictably or reset due to insufficient power.

4. How Many Sensors Can You Connect?

The maximum number of sensors you can connect depends on:

1. **Current Consumption per Sensor:**
   - For example, if each sensor draws 10 mA, you can connect up to 400 mA10 mA=40\frac{400 \, \text{mA}}{10 \, \text{mA}} = 4010mA400mA=40 sensors (assuming no other devices are connected).
   - If each sensor draws 50 mA, the total current exceeds the Arduino's limit, and the system will not work reliably.
2. **Powering Sensors Externally:**
   - If your sensors draw too much current, you can use an external power source for the sensors and use the Arduino only for data communication (e.g., using transistors or MOSFETs to control external power).

Example: Connecting Sensors

- Assume you connect the following to an Arduino:
  - 10 LEDs (each drawing 10 mA).
  - 5 sensors (each drawing 20 mA).
- **Total Current = (10 × 10 mA) + (5 × 20 mA) = 100 mA + 100 mA = 200 mA**
  This is within the Arduino's limit, but if you add more devices, the total current may exceed the safe range.

Key Takeaways

1. **Voltage remains constant** at the regulator output (5V or 3.3V) as long as the current draw stays within the limit.
2. The maximum number of sensors or devices depends on their individual current consumption and the total current limit of the Arduino.
3. Use external power supplies for high-current devices to prevent overloading the Arduino.
4. Monitor total power consumption carefully to ensure reliable operation.