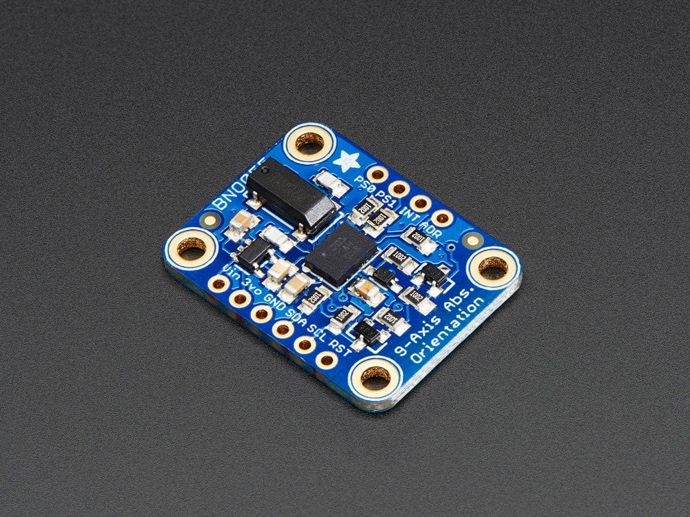
**BNO055 (IMU - Gyroscope, Accelerometer, Magnetometer)**

<https://thepihut.com/products/adafruit-9-dof-absolute-orientation-imu-fusion-breakout-bno055>



* Basic idea - turning the sensor data from an accelerometer, gyroscope and magnetometer into actual "3D space orientation”
* This spit out data which we can use in quaternions, Euler angles or vectors.

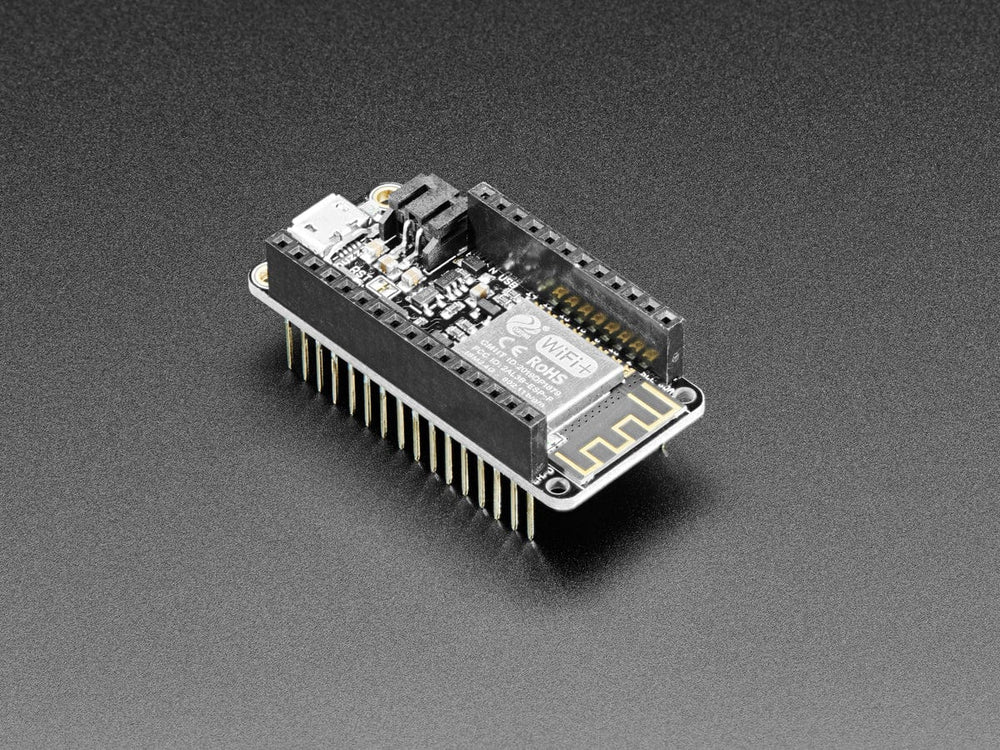
The BNO055 can output the following sensor data:

* **Absolute Orientation** (Euler Vector, 100Hz) Three axis orientation data based on a 360° sphere
* **Absolute Orientation** (Quatenrion, 100Hz) Four point quaternion output for more accurate data manipulation
* **Angular Velocity Vector** (100Hz) Three axis of 'rotation speed' in rad/s
* **Acceleration Vector** (100Hz) Three axis of acceleration (gravity + linear motion) in m/s^2
* **Magnetic Field Strength Vector** (20Hz) Three axis of magnetic field sensing in micro Tesla (uT)
* **Linear Acceleration Vector** (100Hz) Three axis of linear acceleration data (acceleration minus gravity) in m/s^2
* **Gravity Vector** (100Hz) Three axis of gravitational acceleration (minus any movement) in m/s^2
* **Temperature** (1Hz) Ambient temperature in degrees Celsius

<https://github.com/adafruit/Adafruit_BNO055>

[Adafruit BNO055 Absolute Orientation Sensor](https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview)

**ESP8266 - (alternative of raspberry pi)**



<https://thepihut.com/products/adafruit-assembled-feather-huzzah-w-esp8266-wifi-with-stacking-headers>

**1. What it is**

* **ESP8266** is a microcontroller with Wi-Fi capabilities
* It can **read sensor data** and send it over Wi-Fi to another device (like my laptop)

**2. What I Need to Do:**

* **Connect the BNO055 sensor to the ESP8266** via the **I2C** protocol to read posture data (e.g., Euler angles or quaternions).
* **Send data wirelessly** from the ESP8266 to my laptop, where I can display it on a **terminal window**.

**3. Hardware Connections:**

* Use a **breadboard** to connect the **BNO055 sensor** to the **ESP8266**.
* **ESP8266 Pinouts**:
  + **VIN** or **3.3V** (poour supply) – Connect to the **3.3V pin** of the BNO055.
  + **GND** (ground) – Connect to the **GND pin** of the BNO055.
  + **SCL** (Serial Clock Line for I2C) – Connect to the **SCL pin** of the BNO055.
  + **SDA** (Serial Data Line for I2C) – Connect to the **SDA pin** of the BNO055

**4. Software Setup:**

* **Code for ESP8266**: We will write a simple program to:
  + Initialize the **BNO055 sensor**.
  + Read posture data using I2C.
  + Send that data over Wi-Fi to our laptop.
* Use **Arduino IDE** to program the ESP8266 (since it’s widely used with this microcontroller and has a library for the BNO055).
* **Steps**:
  + **Install ESP8266 Board in Arduino IDE**.
  + **Add the BNO055 Library**: We can find the library in Arduino's library manager.
  + Write the code to **read sensor data** and send it via **Serial** or **Wi-Fi**.

**5. Displaying Data on Our Laptop:**

* We can display the sensor data in two ways:
  + **Via Serial Monitor**: If We connect the ESP8266 to Our laptop via USB, We can use the Arduino IDE's **Serial Monitor** to see the data.
  + **Via Wi-Fi**: The ESP8266 can send the data to Our laptop over Wi-Fi (We’ll need to write code that allows We to create a simple **web server** on the ESP8266, which streams the sensor data to Our laptop’s browser).

**6. Powering the ESP8266:**

* If We are powering the **ESP8266 through a USB cable**, it can communicate with Our laptop via serial (just like Raspberry Pi).
* If powering it externally (e.g., a battery), We’ll rely on Wi-Fi to send the data to Our laptop.

**7. Wi-Fi Streaming (Optional Step):**

* If We want to send data via Wi-Fi instead of USB serial, We can:
  1. Set up a **web server** on the ESP8266.
  2. Access the server from Our laptop via the local network.

**Parts Connected Together:**

!<https://cdn-learn.adafruit.com/assets/assets/000/024/674/original/sensors_2472-04.gif.pagespeed.ce.jcQT-E7GOs.gif?1448317739>