MPU-6050

The MPU-6050 is a popular and widely used Motion Processing Unit (MPU), which is essentially a 6-axis sensor on a single chip. It combines a 3-axis gyroscope and a 3-axis accelerometer.

Here's a small note on its key features and applications:

6-Axis Sensing: It measures motion in three dimensions. The accelerometer measures linear acceleration (changes in velocity) along the X, Y, and Z axes, while the gyroscope measures angular velocity (rate of rotation) around the same axes.

DMP (Digital Motion Processor): A key feature of the MPU-6050 is its built-in Digital Motion Processor. This dedicated processor offloads complex calculations from the main microcontroller. It can process the raw sensor data to provide more stable and reliable outputs, such as quaternions or Euler angles, which are essential for applications like orientation tracking.

I2C Communication: It communicates with microcontrollers (like Arduino, Raspberry Pi, etc.) using the I2C (Inter-Integrated Circuit) protocol, which is a simple two-wire interface. This makes it easy to integrate into various electronic projects.

Applications: Due to its small size, low power consumption, and integrated nature, the MPU-6050 is used in a wide range of applications, including:

Drones and Quadcopters: For flight stabilization and control.

Robotics: For balancing robots, navigation, and gesture control.

Wearable Technology: In smartwatches and fitness trackers to detect movement and orientation.

Gaming: In controllers and virtual reality headsets for motion tracking

General hobbyist projects where motion and orientation sensing are required.

TASK 1:

temperature sensor

LM35: Vcc → 5V, GND → GND, OUT → A0 of Arduino

DHT11: Vcc → 5V, GND → GND, Data → D2 (with 10k pull-up resistor)

Servo motor

Vcc → 5V (separate supply recommended if servo is large)

GND → Common GND with Arduino

Signal → D9

Push button

One side → D3

Other side → GND

Pull-up resistor (10k) or use Arduino’s internal INPUT\_PULLUP

🧠 Control Logic

Continuously read temperature.

If auto mode (default):

If temp > 30°C → servo rotates to open vent (say 90°).

If temp ≤ 30°C → servo rotates to close vent (say 0°).

If button pressed (manual override):

Toggle servo state (open ↔ close), ignoring temperature until button is pressed again.

TASK 2:

**Use MPU-6050 sensor** → it measures motion (acceleration + rotation).

**Make an Arduino Shield PCB** → this will sit on top of Arduino UNO.

**Add Power Circuit** → 5 V from Arduino → 3.3 V regulator → MPU-6050.

**Connect I²C** →

Arduino A4 = SDA

Arduino A5 = SCL

Use level shifter (because MPU-6050 is 3.3 V, Arduino is 5 V).

**Interrupt Pin** → MPU INT → Arduino D2.

**Add Debug LEDs** →

Power LED (shows 3.3 V OK).

Status LED (Arduino D8 → blinking).

Error LED (Arduino D9 → lights if problem).

Arduino talks to MPU-6050 using I²C.

MPU-6050 sends motion data (accel + gyro).

LEDs help you see:

Power ON,

System running,

Error if something is wrong.

Schematic circuit of auduino:

