

1. Controller Model and Price

The controller used in this project is the **ESP32 DevKit V1 (ESP-WROOM-32)**.

- **Model:** ESP32 DevKit V1 – ESP-WROOM-32
- **Price:** Approximately ₹500 to ₹600

2. Number of I²C and UART Channels

The **ESP32 DevKit V1 (ESP-WROOM-32)** has 2 I²C and 3 UART communication channels:

- **I²C Channels:** 2 (I²C0 and I²C1)
 - Any GPIO can be configured as SDA/SCL using software.
 - The I²C bus can connect multiple I²C devices using just 2 wires:
SDA (Data Line)
SCL (Clock Line)
- **UART Channels:** 3 (UART0, UART1, and UART2)
 - UART0 is typically used for programming and debugging.
 - UART1 and UART2 can be used for other serial communication tasks.

3. Will the I²C and UART Overlap During Operation? Why?

No, I²C and UART will not overlap during operation, as long as different GPIO pins are assigned to each interface.

- The ESP32 has a flexible pin mapping system, allowing you to assign I²C and UART functions to different pins.
- Overlap occurs only if the same GPIOs are mistakenly assigned to both interfaces at the same time.
- If configured properly, I²C and UART can operate simultaneously without interference.

4. What Happens if the Internet or Cloud Server Goes Down? Does It Reconnect Automatically or Require Manual Trigger?

The **ESP32** can **automatically reconnect** to Wi-Fi and cloud services with proper programming.

- **Wi-Fi Reconnection:**
 - The ESP32 can be programmed to **auto-reconnect to Wi-Fi** if the connection drops.
 - This is typically handled in the code using `WiFi.begin()` with checks in the `loop()` to reconnect if `WiFi.status() != WL_CONNECTED`.

- **Cloud Server Reconnection:**

- For protocols like **HTTP or MQTT**, the ESP32 can retry failed requests or automatically reconnect to the broker/cloud using built-in libraries.
- For example, MQTT libraries like PubSubClient offer **auto-reconnect functions**.

No manual intervention is needed if reconnection logic is implemented properly in the firmware.

5. Will the SHT31 Sensor Work in Condensed State? How Does It React to Non-Condensing Conditions? How to Overcome It Economically?

- **Condensed State:**

- The SHT31 will be affected by condensation. When water droplets form on the sensor surface, it may cause temporary incorrect readings or slow response due to moisture interference.
- Prolonged exposure to condensation may also reduce sensor lifespan.

- **Non-Condensing Conditions:**

- In dry or non-condensing environments, the SHT31 provides accurate and stable humidity and temperature readings.

- **Economic Solution to Overcome Condensation:**

- Periodically enable the heater and exhaust to prevent condensation buildup.
- Enclose the sensor in a breathable yet water-resistant membrane (e.g., Gore-Tex or sintered filter cap) to protect it while still allowing airflow.

6. Available Digital Output GPIOs After Essential Connections

Assuming the following essential connections:

- **SHT31 Sensors (x2)** – I²C uses **2 GPIOs total** (SDA and SCL shared)
- **Heater Control (Relay)** – 1 GPIO (Digital Output)
- **Exhaust Fan Control (Relay)** – 1 GPIO (Digital Output)

ESP32 DevKit V1 has around 25 usable GPIOs, but after excluding reserved and sensitive pins, about 17 are practically usable.

- **Total used GPIOs:**

- I²C Bus: 2
- Relays (Heater + Exhaust): 2
→ Total used: 4 GPIOs

- **Available digital output GPIOs:**

- ~13 GPIOs remaining, most of which can be used as digital outputs.

7. Will the Digital Output of ESP32 Exceed 3.3V?

No, the digital output voltage of the ESP32 will not exceed 3.3V.

- The ESP32 operates at 3.3V logic levels, meaning:
 - HIGH (logic 1) = ~3.3V
 - LOW (logic 0) = ~0V
- It cannot output 5V directly from its GPIO pins.
 - Applying more than 3.6V to a GPIO pin can damage the ESP32.

8. What Is the Driving Current of ESP32-WROOM-32 Controller? Will It Be Sufficient to Actuate Relay Modules (Heater and Exhaust)? If Not, How to Achieve It?

- **ESP32-WROOM-32 GPIO Driving Current:**
 - Each GPIO pin can safely source/sink up to ~12 mA
 - Absolute maximum per GPIO is ~20 mA (not recommended continuously)

Is It Sufficient for Relay Modules?

If you're using bare relays without a driver:

- These typically need 70–100 mA or more, which is beyond ESP32's capability.
- ESP32 cannot drive bare relays directly.

How to Achieve It Safely?

- Use a transistor switch circuit (e.g., using NPN like BC547 or MOSFET like IRFZ44N) to control the relay or heater with higher current requirements.
- Always include a flyback diode across the relay coil.

9. Can the ESP32-WROOM-32 Controller Handle HTTP GET and PUT Requests? How Does It Work?

Yes, the ESP32-WROOM-32 supports HTTP GET, POST, PUT, and DELETE requests through its built-in Wi-Fi and TCP/IP stack.