**1. Limited GPIO Pins on ESP32-CAM Module**

* The **ESP32-CAM module only exposes about 9 usable GPIO pins** (excluding special-purpose pins).
* Your setup requires pins for:
  + 3 ultrasonic sensors (each sharing trig/echo on a single GPIO): **3 pins**
  + MPU6050 I2C (SCL, SDA): **2 pins**
  + PCF8574 I2C (same bus as MPU6050): **2 pins shared on I2C bus**
  + 2 L298N EN pins for UP/DOWN (GPIO4, GPIO14)
  + 2 L298N EN pins for FWD/BWD (GPIO2, GPIO16)
  + PCF8574 I/O expansion module controlled via I2C bus (pins shared with MPU6050)

Total GPIOs claimed explicitly:

* Ultrasonic: 3 pins
* I2C (MPU6050 + PCF8574): 2 pins (shared)
* L298N enable pins: 4 pins  
  Total = 9 pins used for peripherals

**2. GPIO Pins Used for Motor Control:**

* **L298N Motor Drivers:**
  + Each L298N needs 4 IN pins for direction control + 2 EN pins for speed control.
  + 2 L298N boards, so 8 IN pins + 4 EN pins = **12 pins**.
  + Your setup uses the PCF8574 I/O expander for the 8 IN pins of the L298Ns, which helps reduce GPIO pin usage on the ESP32.
  + EN pins for the L298N motor drivers are directly connected to the ESP32 GPIO (GPIO4, GPIO14, GPIO2, GPIO16).

Offloaded 8 motor control pins to PCF8574, saving 8 GPIOs on ESP32.

**EN pins still consume 4 GPIOs.**

**3. I2C Bus for PCF8574 and MPU6050:**

* Both use same I2C bus on GPIO0 (SCL) and GPIO3 (SDA).

**4. Ultrasonic Sensors (AJ-SR04M) on GPIO12, GPIO13, GPIO15:**

* Each sensor shares trigger and echo on the same GPIO pin.
* Using GPIO12, GPIO13, GPIO15 uses **3 pins total**.

**5. Summary of GPIO Usage:**

| **Peripheral** | **Pins Used** | **Pin Count** |
| --- | --- | --- |
| Ultrasonic Sensors | GPIO12,13,15 | 3 |
| MPU6050 (I2C) | GPIO0 (SCL), GPIO3 (SDA) | 2 |
| PCF8574 (I2C) | Same as MPU6050 I2C pins | 0 extra |
| L298N EN pins | GPIO4, GPIO14, GPIO2, GPIO16 | 4 |
| PCF8574 (8 output pins) | offloaded (no ESP32 pins) | 0 |
| **Total GPIO Pins Used** |  | **9** |

**Potential Problems and Limitations:**

1. **GPIO15 is often used for boot mode selection on ESP32-CAM**
   * GPIO15 is a strapping pin, must be LOW on boot to boot from flash.
   * Using GPIO15 for ultrasonic sensor input could cause boot instability unless carefully managed.
2. **I2C bus clock stretching and bus conflicts**
   * With MPU6050 and PCF8574 sharing I2C, the bus may be busy or blocked if either device hangs.
   * You must handle errors or bus resets in software.
3. **Limited PWM pins for motor speed control**
   * EN pins on L298N are for PWM speed control; ESP32-CAM PWM capability is limited on some pins.
   * You only have 4 EN pins assigned; all must support PWM.
4. **Power considerations for motors and sensors**
   * Not an IO issue but critical: motors draw high current; wiring and power stability must be solid to avoid brownouts on ESP32.

| **Main Problem** | **Explanation** |
| --- | --- |
| **Pin shortage on ESP32-CAM** | Only ~9 GPIO pins usable, almost fully occupied. |
| **GPIO15 usage risk** | Pin 15 is a strapping pin; using it for sensor input risks boot failures. |
| **No spare pins for expansion** | Setup leaves no free pins for extra features. |
| **I2C bus sharing risks** | MPU6050 and PCF8574 on same bus may cause communication conflicts. |
| **PWM pin constraints** | EN pins need PWM capable pins, limited on ESP32-CAM. |