Microcontroller Viva Questions & Solutions

Frequently Asked Viva Questions

Q1: What is the difference between a microcontroller and a microprocessor?

A **microcontroller** is a compact IC containing CPU, RAM, Flash memory, and I/O ports on a single chip, suitable for embedded systems. A **microprocessor** contains only the CPU and requires external memory and peripheral chips to function.

Q2: What is the role of GPIO pins in microcontrollers?

GPIO (*General Purpose Input/Output*) pins allow the MCU to interact with external devices:

- Configured as **input** → read sensor/button states.
- Configured as **output** → drive LEDs, motors, or buzzers.

Q3: Why do we use a pull-down resistor with a button input?

Prevents the input pin from floating (undefined voltage) when the button is not pressed. A **pull-down resistor** holds the pin at logic LOW. Pressing the button connects the pin to VCC (logic HIGH).

Q4: What is switch bouncing? How is it handled?

Mechanical switches may generate multiple rapid ON/OFF transitions (bouncing).

- **Software debounce:** Add a delay or filter signal in code.
- **Hardware debounce:** Use a capacitor or Schmitt-trigger circuit.

Q5: Explain the difference between delay-based and interrupt-based LED control.

- **Delay-based:** Uses `delay()` → simple but blocks CPU.
- **Interrupt-based:** Responds to events (button press) → efficient and responsive.

Q6: Why might the LED not respond if the button is pressed quickly?

Due to **switch bouncing** or long `delay()` in the program. Solution → apply debouncing or check input state more frequently.

Q7: How do you make an LED stay ON for 5 seconds after a button press?

Use a delay after turning ON the LED:

```
if (digitalRead(buttonPin) == HIGH) {
  digitalWrite(ledPin, HIGH);
  delay(5000); // LED ON for 5s
  digitalWrite(ledPin, LOW);
}
```

Or use `millis()` for non-blocking timing.

Q8: What is the function of the Serial Monitor in Arduino IDE?

Allows communication between MCU and PC via UART. Useful for debugging and monitoring variable states.

Q9: Give one real-life application of microcontrollers.

Examples:

- Home appliances: washing machines, microwave ovens.
- IoT devices: smart bulbs, thermostats.
- Automotive systems: engine control units, airbags.

Q10: What is PWM? How is it used in microcontrollers?

PWM (*Pulse Width Modulation*) generates analog-like signals using digital pins:

- Control LED brightness
- Control DC motor speed
- Generate audio tones

Q11: What is an ADC in a microcontroller?

ADC (*Analog to Digital Converter*) converts analog voltages into digital values. Used to read sensors like temperature, light, or potentiometers.

Q12: What is I2C and how does it differ from SPI?

- **I2C:** 2-wire communication (SDA + SCL), supports multiple devices, slower.
- **SPI:** 4-wire communication (MISO, MOSI, SCK, CS), faster, supports multiple slaves using separate CS pins.

Q13: What is deep sleep mode in ESP32 or similar MCUs?

Low-power mode where the main CPU is off, peripherals may be partially active. RTC or timers can wake the MCU at scheduled intervals to save energy in battery-powered applications.

Q14: How can you measure the execution time of a code segment on a microcontroller?

Use `millis()` or `micros()` functions to measure elapsed time. Useful for profiling or precise timing tasks.

Q15: Explain the difference between volatile and non-volatile memory in MCUs.

- **Volatile:** RAM → loses data when power is off.
- **Non-volatile:** Flash/EEPROM → retains data without power.

Q16: What is the function of an RTC (Real-Time Clock)?

Keeps track of time and date even when MCU is in deep sleep. Used for alarms, periodic tasks, and energy-efficient timekeeping.

Q17: What is the purpose of UART RTS and CTS pins?

Used for hardware flow control in serial communication:

- **RTS** → Request to Send (output from MCU)
- **CTS** → Clear to Send (input to MCU)

Q18: Name a few safety considerations when working with microcontrollers.

- Avoid overvoltage on pins
- Never short VCC and GND
- Use current-limiting resistors for LEDs/motors
- Follow ESD precautions