

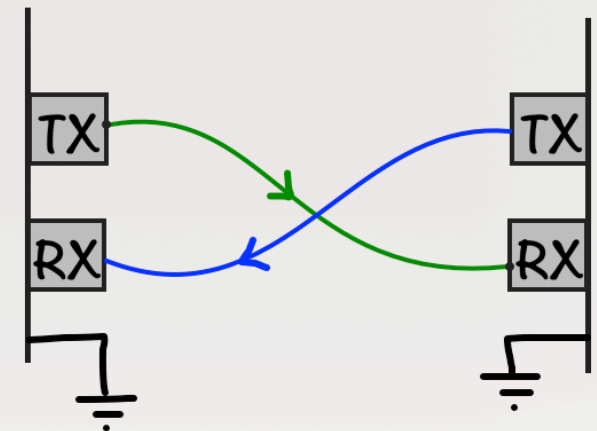
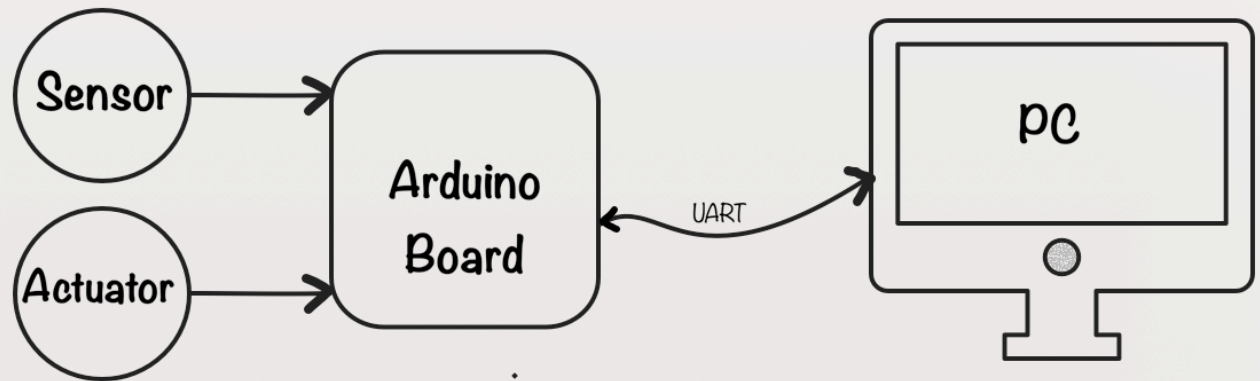
Communication Protocols

UART

Use Case: Arduino to PC

- **Key Features:**

- Peer-to-peer communication without a master-slave relationship.
- **Asynchronous:** No dedicated clock line; communication uses a predefined baud rate (commonly 9600).
- **Lines:**
 - TX (Transmitter), RX (Receiver), and GND (Ground).
- Devices require individual power, typically 5V (VCC).

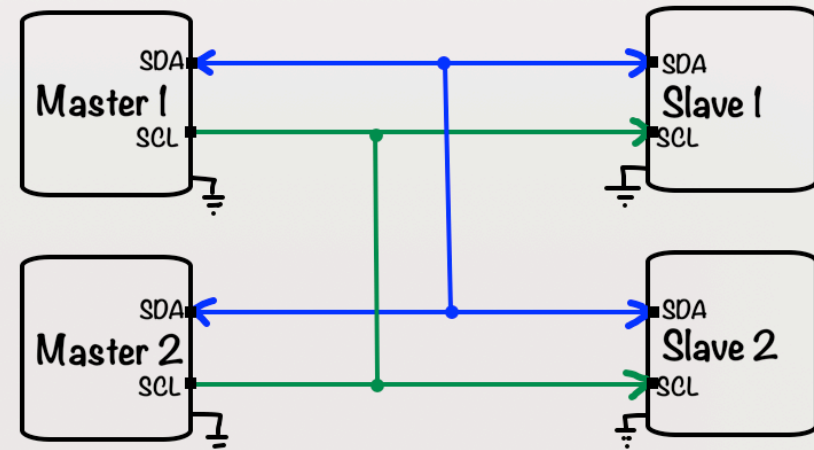


I2C

Use Case: Arduino to LCD

- **Key Features:**

- Operates in a **master-slave architecture**: Supports multiple slaves and multiple masters.
- **Synchronous**: Communication uses a dedicated serial clock line (SCL) set by the master.
- **Lines:**
 - SDA (Serial Data), SCL (Serial Clock), and GND (Ground).
- Devices require individual power, typically 5V (VCC).

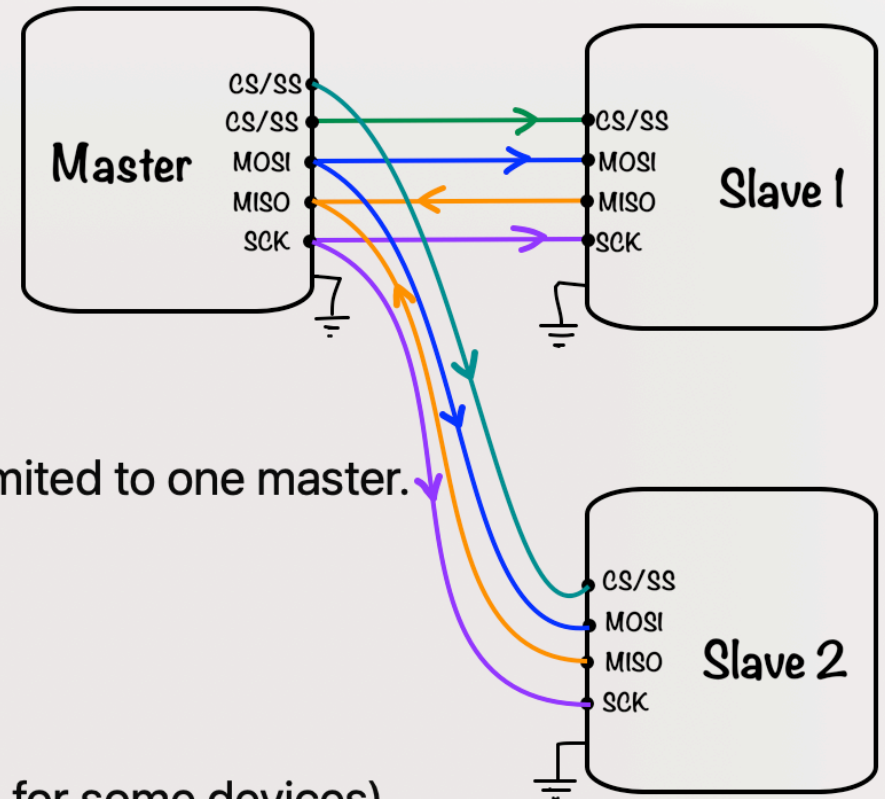


SPI

Use Case: Arduino to RFID

- **Key Features:**

- Master-slave protocol, supports multiple slaves but limited to one master.
- **Synchronous:** Uses a serial clock (SCK) for timing.
- **Lines:**
 - CS/SS (Chip/Slave Selector, also referred to as SDA for some devices).
 - SCK (Serial Clock), MOSI (Master Out Slave In), MISO (Master In Slave Out), and GND (Ground).
- Devices require individual power, typically 5V (VCC).
- Optional **RESET** line for reinitializing the clock state.



Combination of Protocols

Scenario:

An RFID reader communicates with a microcontroller using SPI, data is displayed on an LCD via I2C, and the microcontroller transfers data to a PC over UART.

Diagram Elements:

- **RFID Reader (SPI):**

Connected via CS/SS, SCK, MOSI, MISO, and GND.

- **Microcontroller:**

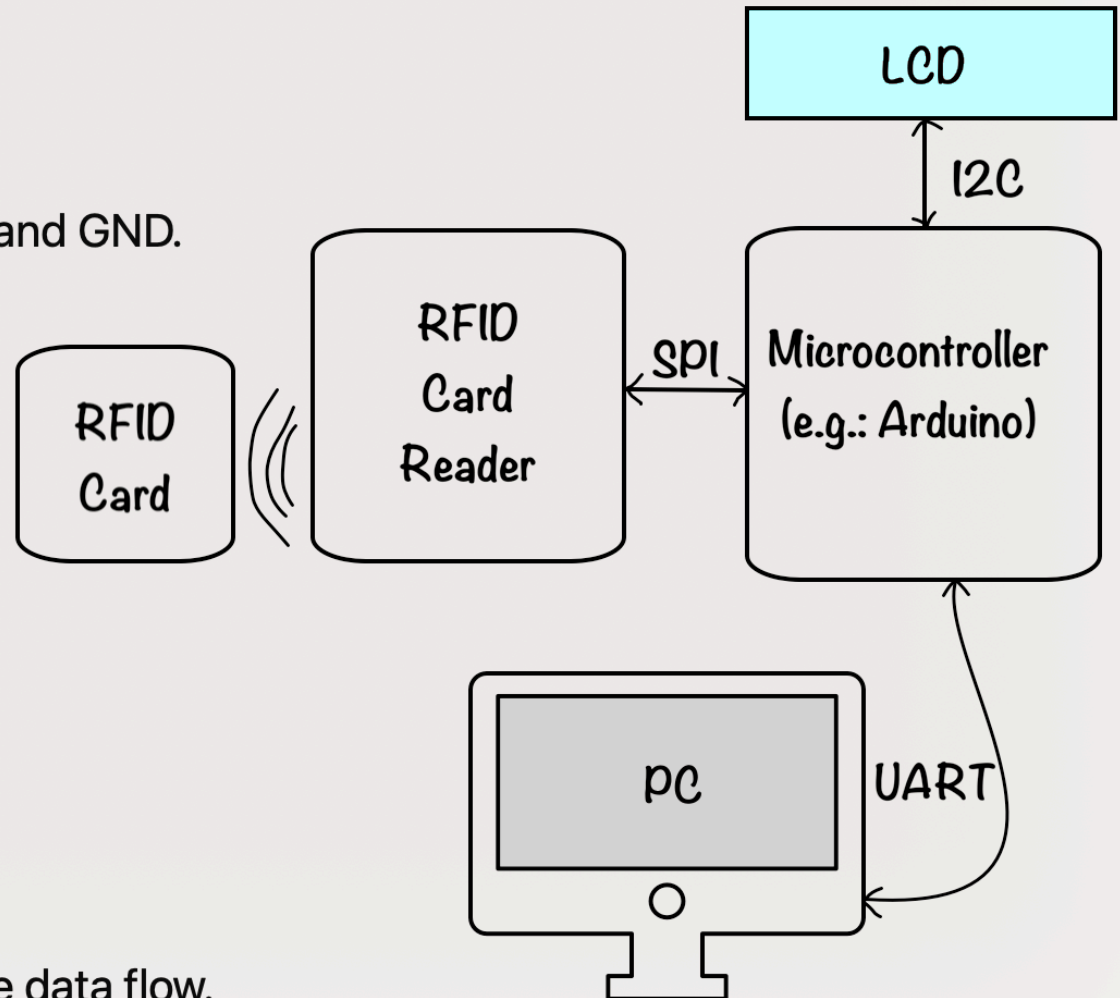
Central hub for communication.

- **LCD (I2C):**

Connected via SDA, SCL, and GND.

- **PC (UART):**

Connected via TX, RX, and GND.



Key Note:

Each protocol handles a unique part of the data flow.