Communication

Chirag Sangani

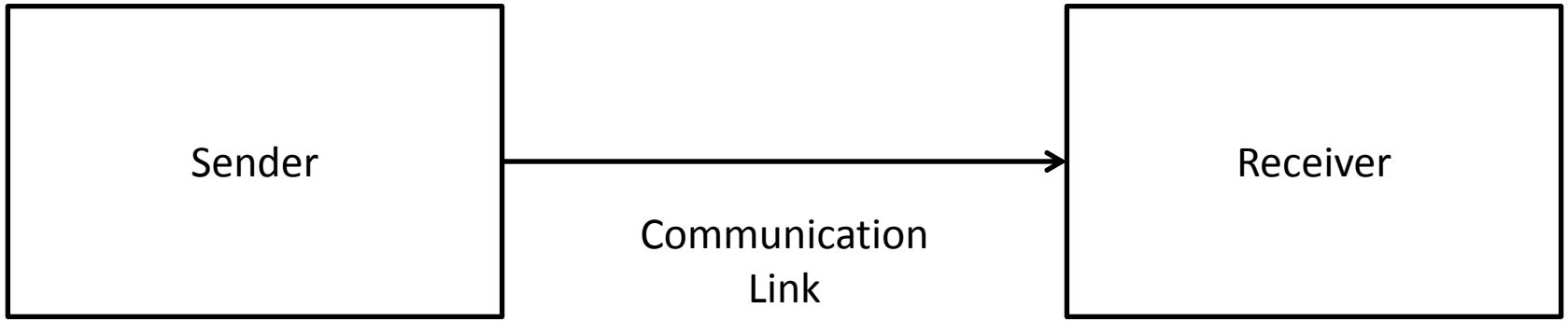
Scope of Communication

- Telephones and cell phones.
- Satellite networks.
- Radio and DTH services.
- Campus LAN and wireless.
- Internet.
- Intra-galactic communication.

Essentials of Communication

- The basic task: to convey a message (data) from one end to the other end.
- Sounds simple enough, however, there are a lot of assumptions that we're used to which need to be relooked at.
- The biggest assumption usually is that the message will reach the destination correctly.

Essentials of Communication



Assumptions in Communication

- The communication link exists.
- The communication link is sound.
- The sender and receiver are the correct nodes.
- The sender is sending the correct data.
- The receiver is able to correctly interpret the incoming data.

Protocols in Communication

- Assumptions in communication are harmful, yet important.
- To deal with these assumptions, we need an elaborate mechanism to ensure correctness of communication.
- These elaborate mechanisms have been codified and standardized as protocols.

Advantages of Protocols

- Standardized, so interoperability is ensured.
- Usually include error-detection and error-correction mechanisms.
- Are available as implemented chips that can be directly used.

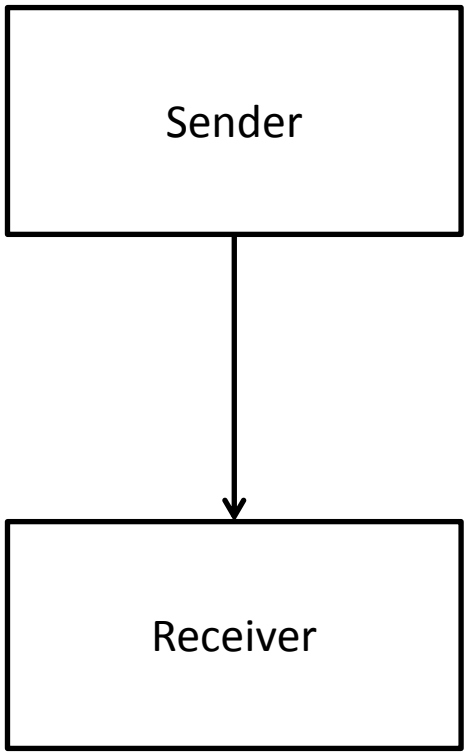
Types of Protocols

- There are different ways of categorizing protocols.
- Protocols can be categorized technically as serial mode transfer or parallel mode transfer.
- Protocols can also be categorized as asynchronous mode transfer or synchronous mode transfer.

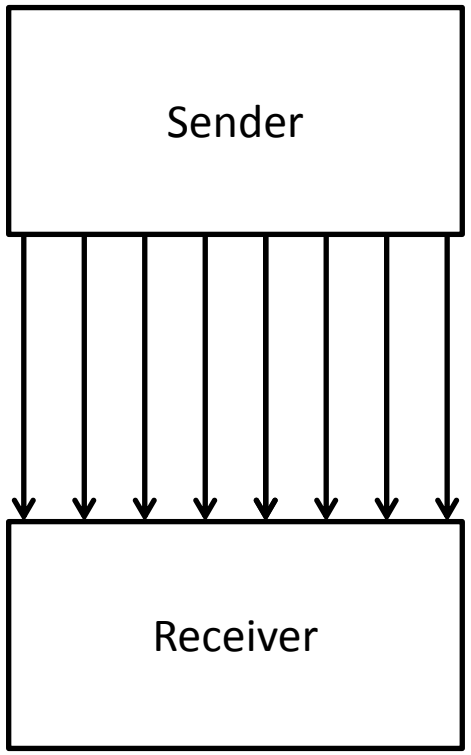
Serial and Parallel Mode

- Remember that we want to transfer data which, at its essence, is a binary number.
- We shall constrict ourselves to 8-bit binary numbers.
- Parallel protocols transfer 8 bits by 8 parallel wires.
- Serial protocols transfer 8 bits by using only one or two wires.

Serial and Parallel Mode



Serial Mode



Parallel Mode

Advantages and Disadvantages

Serial Mode

- Advantages:
  - Reliable
  - Low-cost
  - Low-power
- Disadvantages:
  - Slow
  - Requires complex control

Parallel Mode

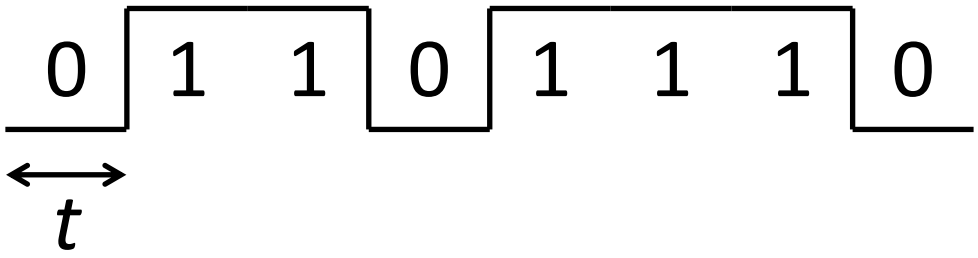
- Advantages:
  - Very fast
  - Simple to implement
- Disadvantages:
  - Unreliable
  - Short-range
  - Expensive
  - Draws more power

Synchronous and Asynchronous Mode

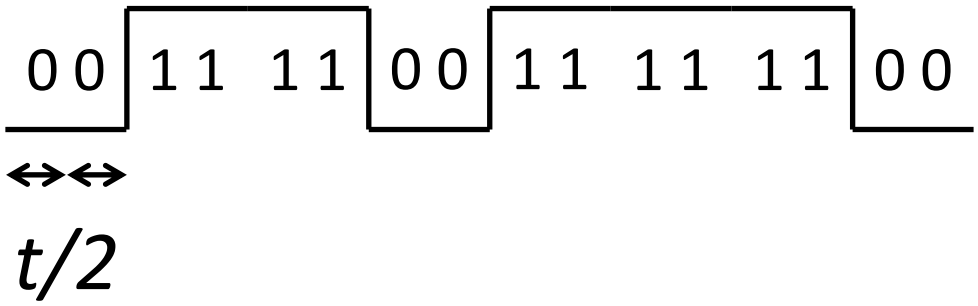
- Pertains to sender-receiver synchronization.
- Sender sends data at a certain speed. For flexibility, protocols allow for multiple speeds.
- If receiver does not know the speed or judges it incorrectly, errors may occur.

Perils of Desynchronization

Sender:

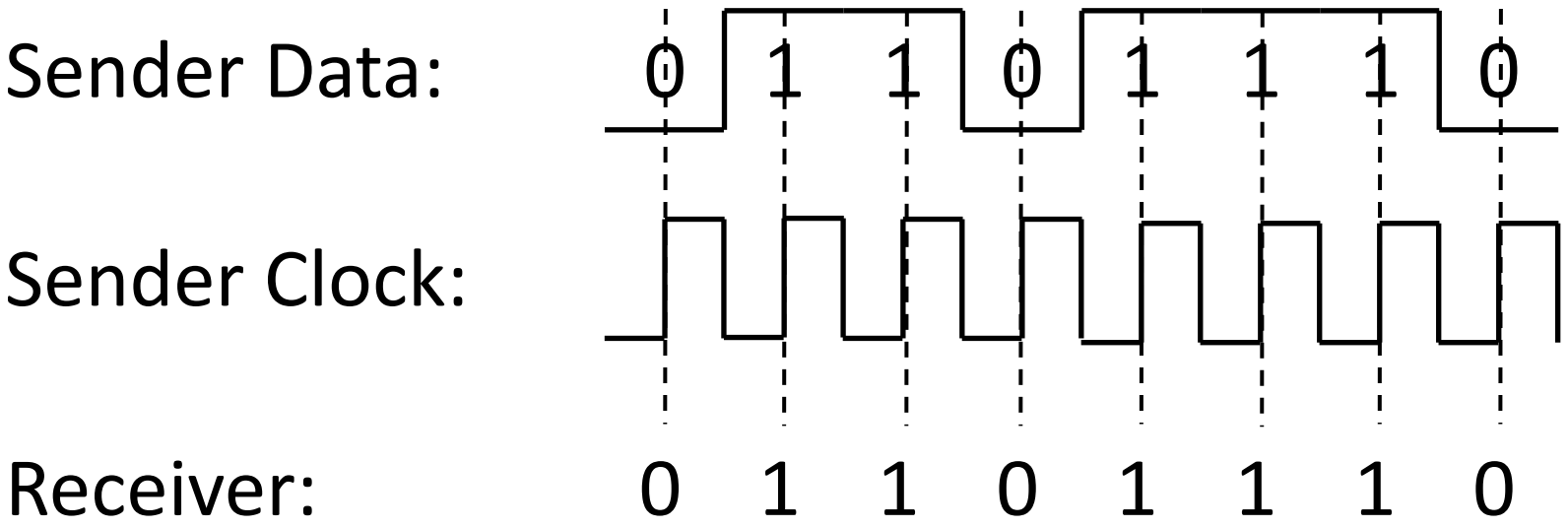


Receiver:



Synchronous Mode

- Sender sends a clock signal along with data: at every rising / falling edge of the clock, the data value is read by the receiver.



Asynchronous Mode

- There is no clock signal.
- The receiver and the sender communicate at a predetermined speed (bauds or bits per second).

Advantages and Disadvantages

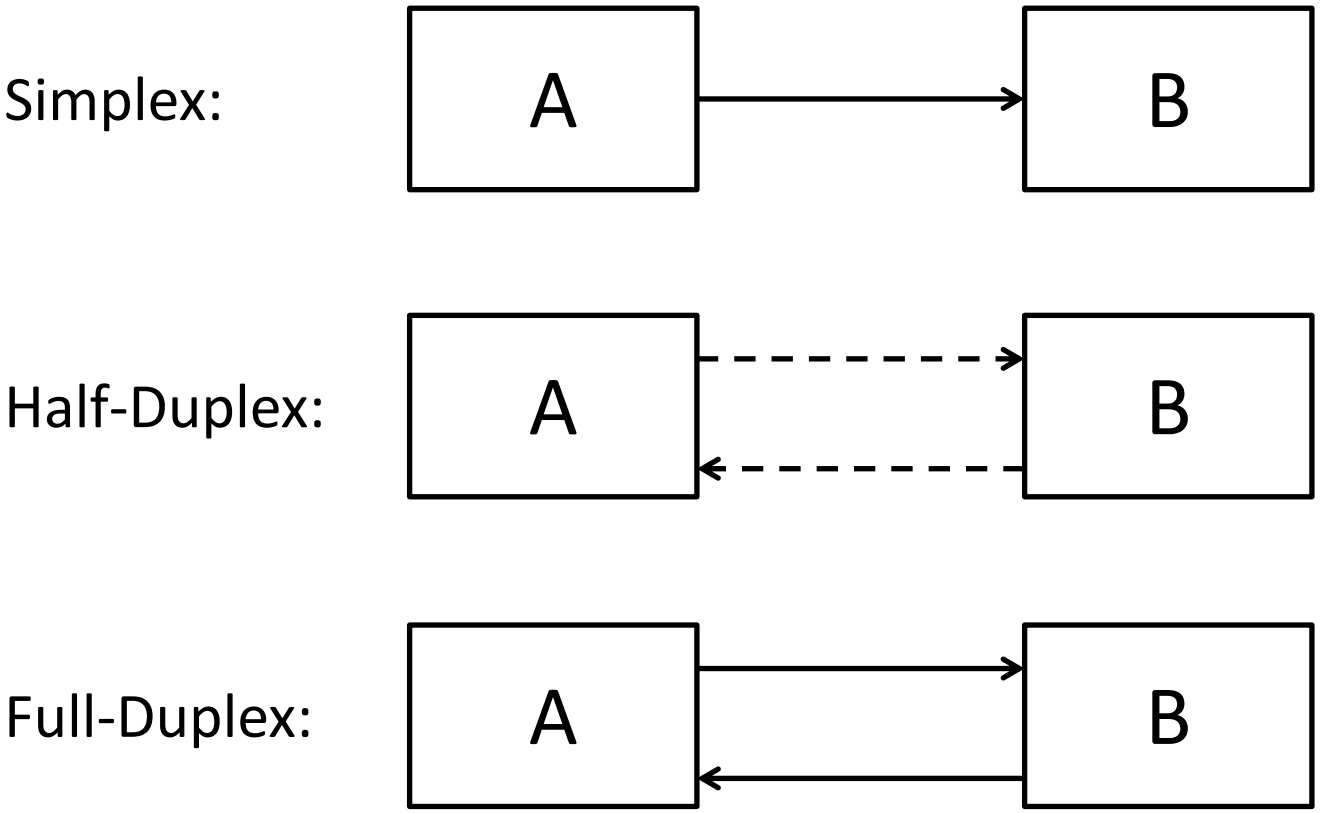
Asynchronous Mode

- Advantages:
  - Simple
  - Inexpensive
- Disadvantages:
  - High overhead
  - Error-prone

Synchronous Mode

- Advantages:
  - Efficient
  - Reliable
- Disadvantages:
  - Complicated
  - Expensive

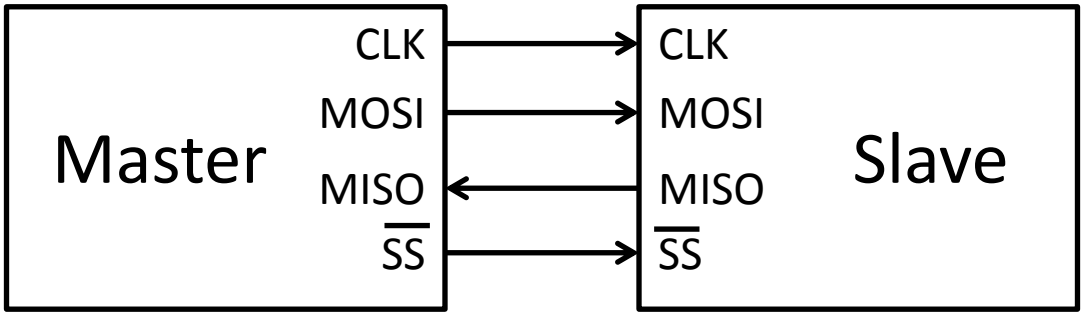
Transmission Modes



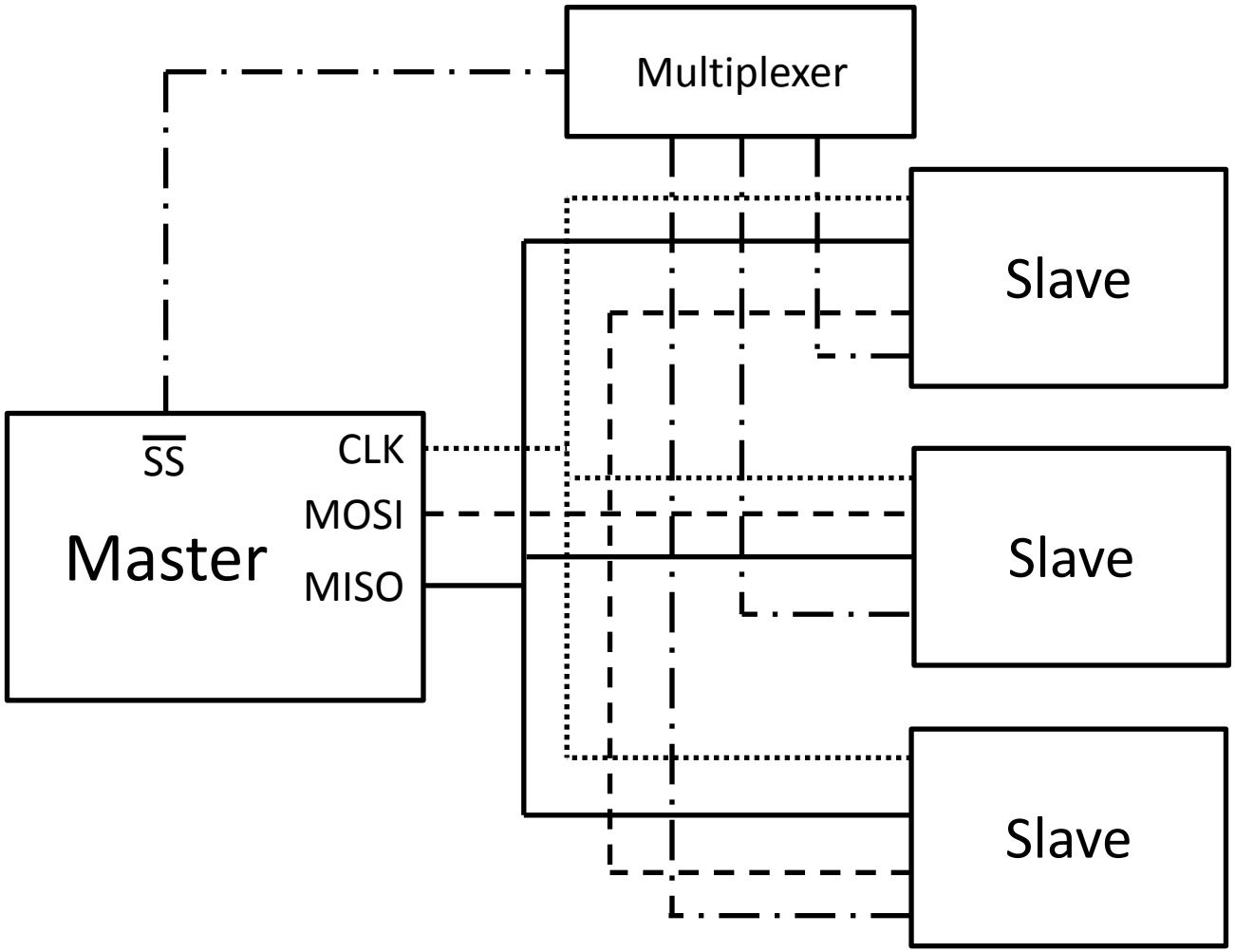
Serial Peripheral Interface (SPI)

- It is a serial, synchronous, full-duplex protocol.
- Sender and receiver follow a master-slave relationship.
- There may be multiple nodes in the network. One node is master, the rest are slaves.
- The slaves can communicate only with the master.
- Master decides when communication will occur.

SPI Schematics: Single Slave



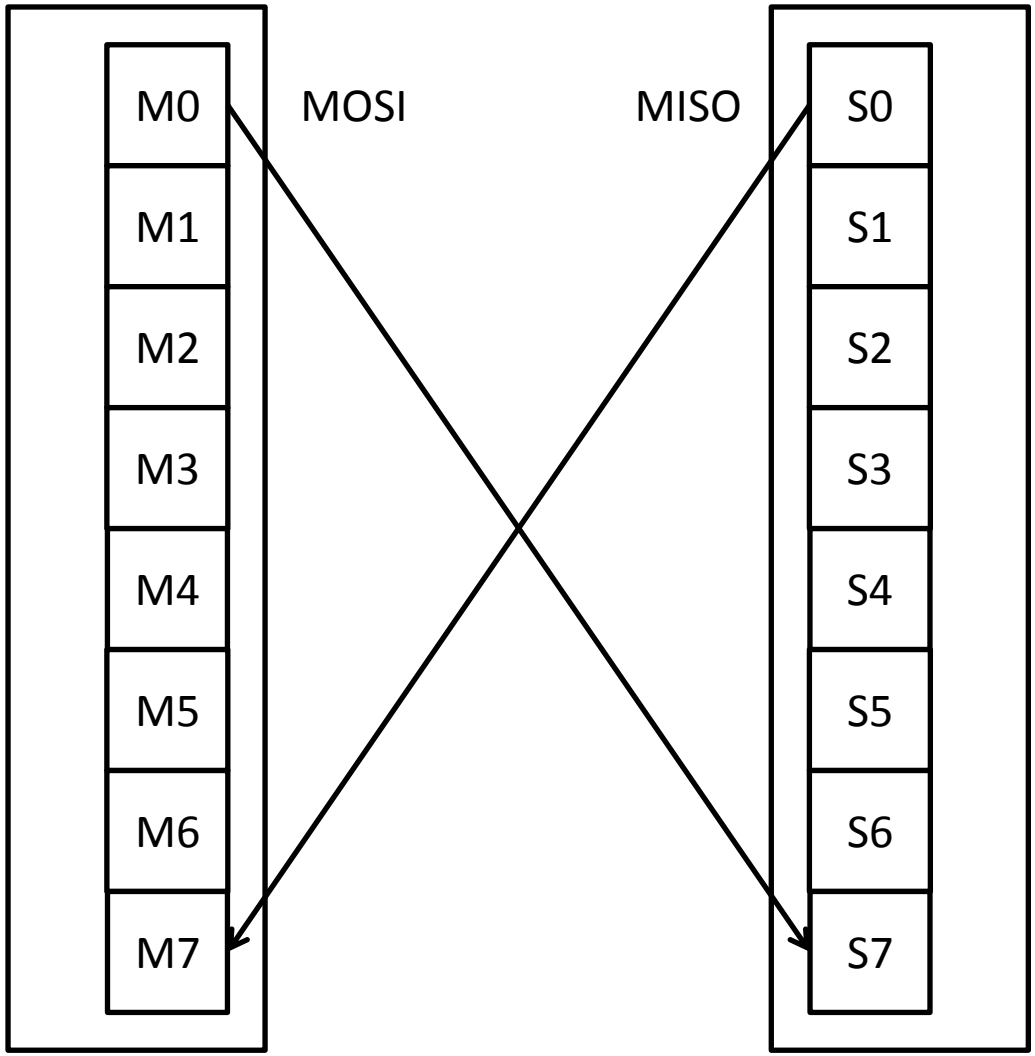
SPI Schematics: Multiple Slaves



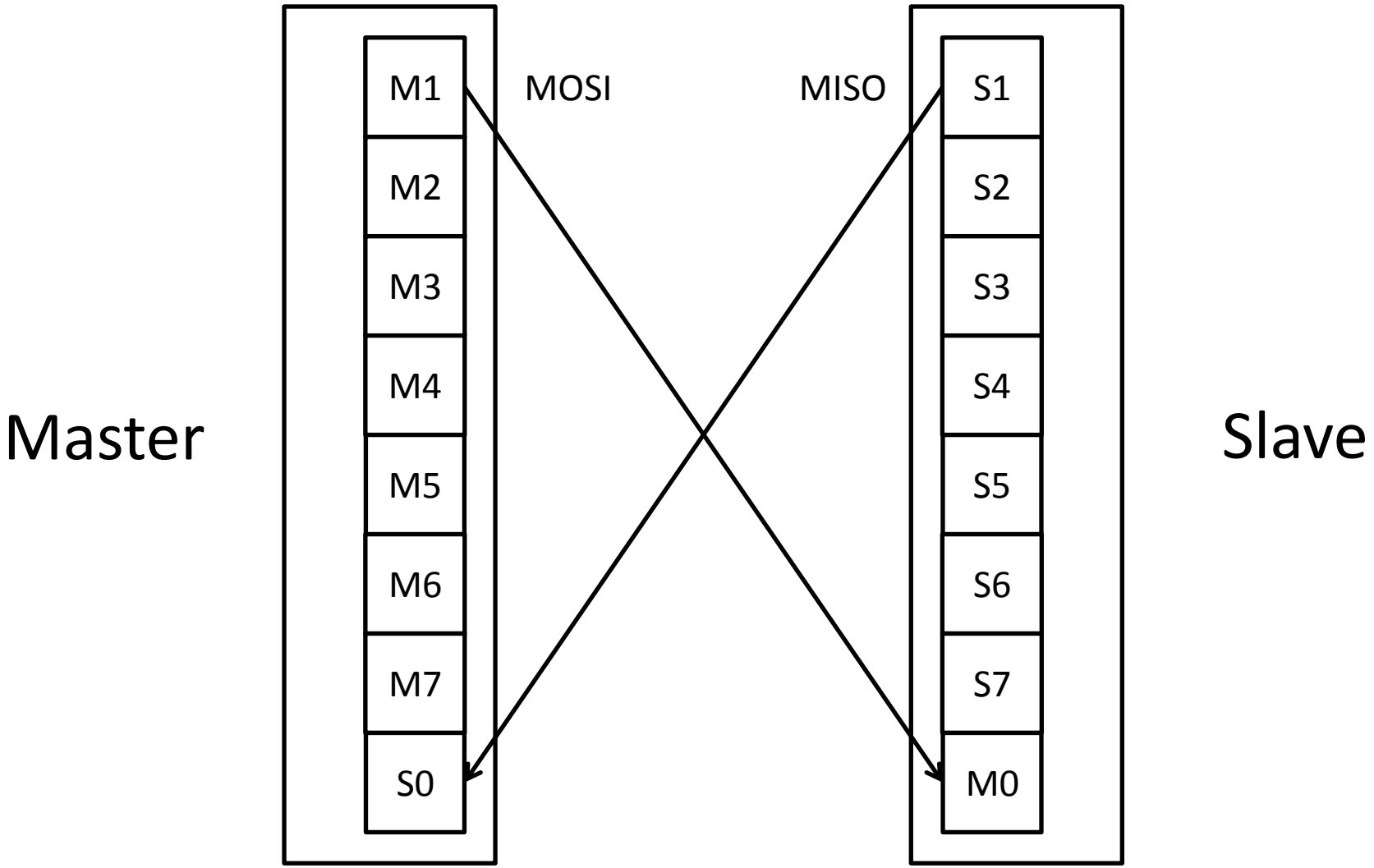
Pins in SPI

- CLK is generated by Master.
- MOSI is Master Out Slave In: Data sent by Master to Slave.
- MISO is Master In Slave Out: Data sent by Slave to Master.
- $\overline{SS}$ is slave select: Slave communicates with Master only if this pin's value is set as LOW.

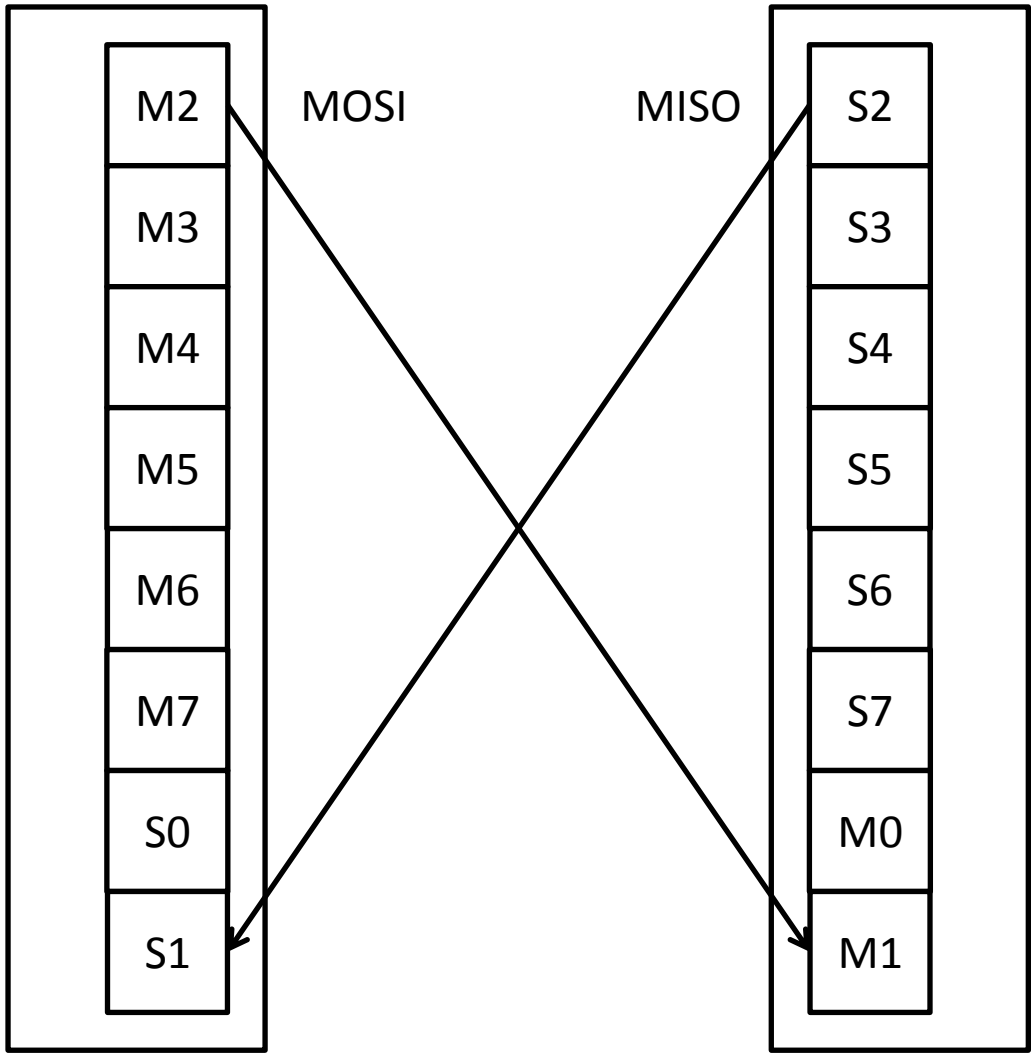
Data Transfer in SPI



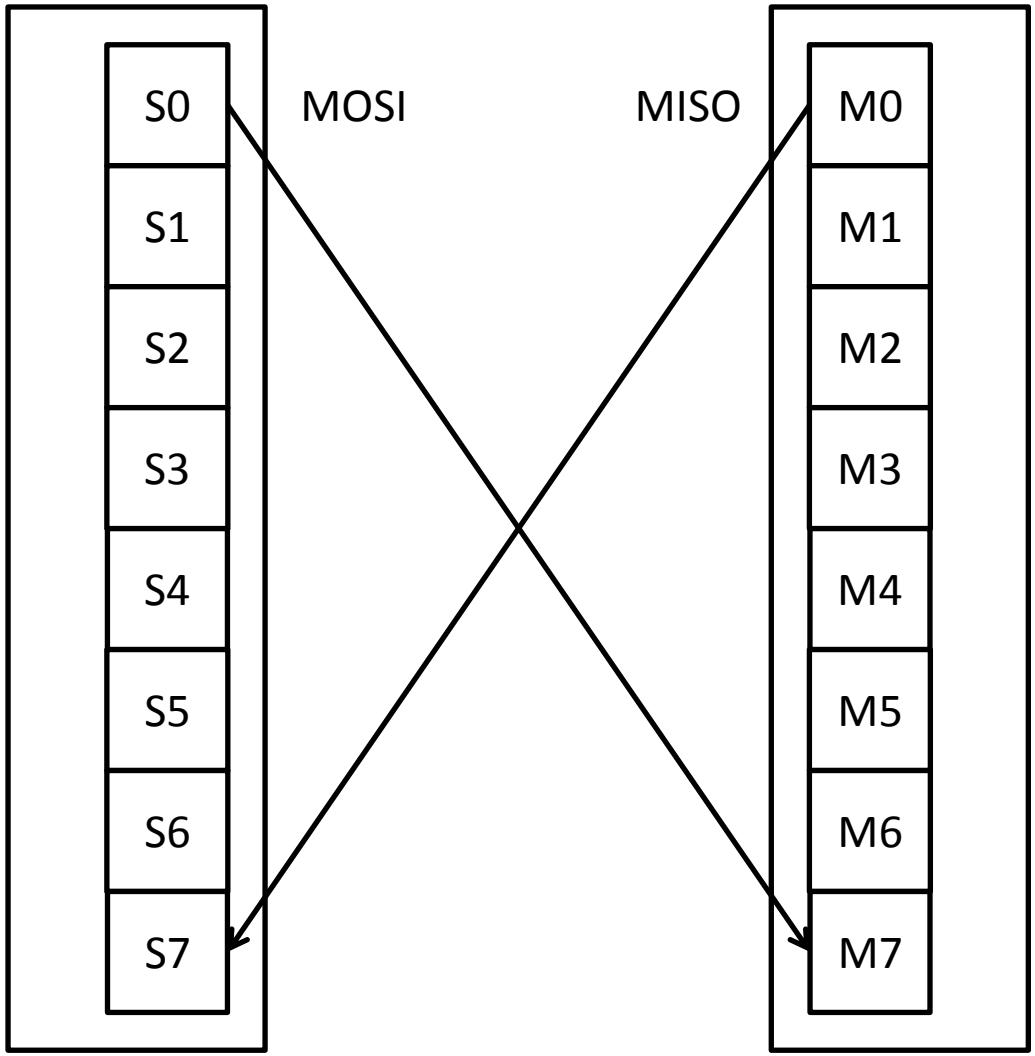
Data Transfer in SPI



Data Transfer in SPI



Data Transfer in SPI



SPI Code

Master

```
char A = spi('c');
lcd_putchar(A);
```

Slave

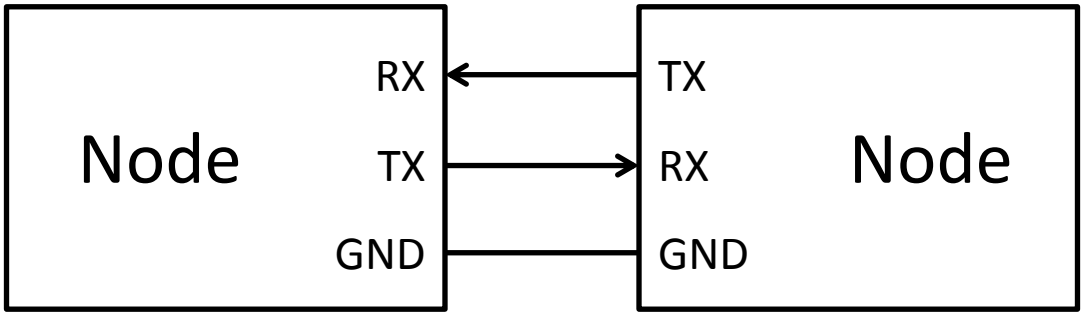
```
char B = spi('1');
```

This code will display '1' on the LCD.

Universal Asynchronous Receiver – Transmitter (UART)

- UART is a simple half-duplex, asynchronous, serial protocol.
- Simple communication between two equivalent nodes.
- Any node can initiate communication.
- Since connection is half-duplex, the two lanes of communication are completely independent.

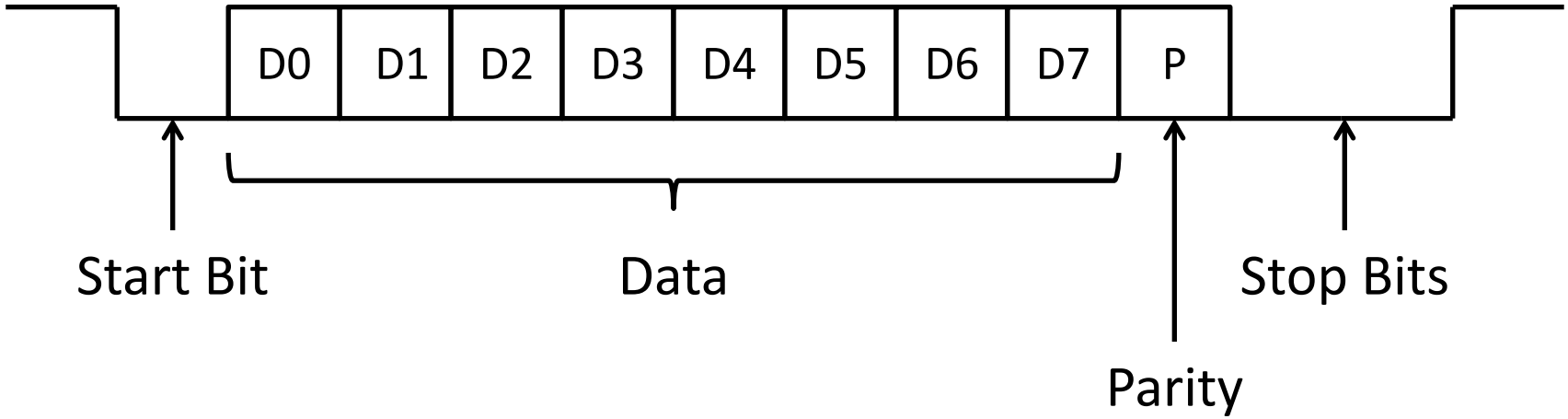
UART schematics



UART Characteristics

- The speed of communication (measured in bauds) is predetermined on both ends.
- A general rule of thumb is to use 9600 bauds for wired communication.
- UART implements error-detection in the form of parity bit.

UART Packet Structure



Parity Bit

- Parity bit is HIGH when number of 1's in the Data is odd.
- Respectively, it is LOW when number of 1's in the Data is even.

UART Code

Node A

```
putchar('a');
/*    Transmit the
*     character 'a'
*/
```

Node B

```
char B = getchar();
/*    getchar() waits
*     for transmission
*/
lcd_putchar(B);
```

This code will display 'a' on the LCD.

Thank you.