Embedded Communication

Introduction

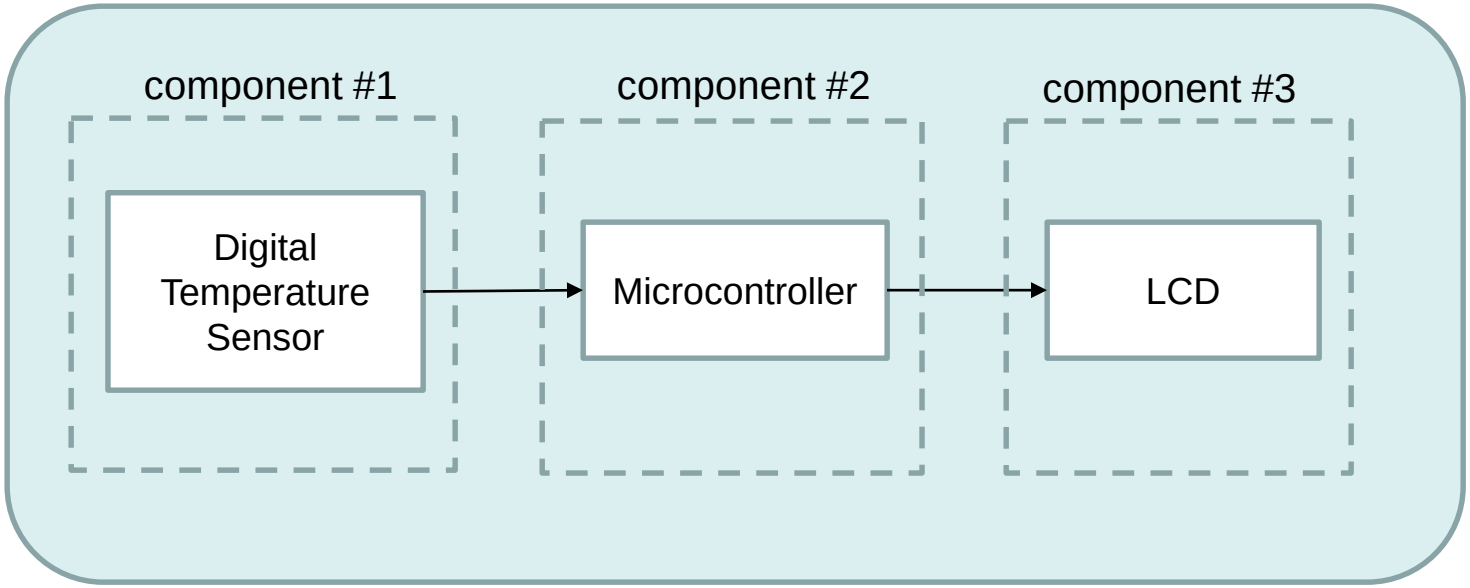
Chapter 8
“Mastering STM32” by Carmine Noviello
https://leanpub.com/mastering-stm32

What is an Embedded System?

- An “Embedded System” refers to a processor with a specific function that is part of a larger mechanical/electrical system
- Microcontrollers are components of an “Embedded System”
- Embedded Systems can be found in mp3 players, DVD players, GPS receivers, printers, washing machines, dishwashers, ….

What is Embedded Communication?

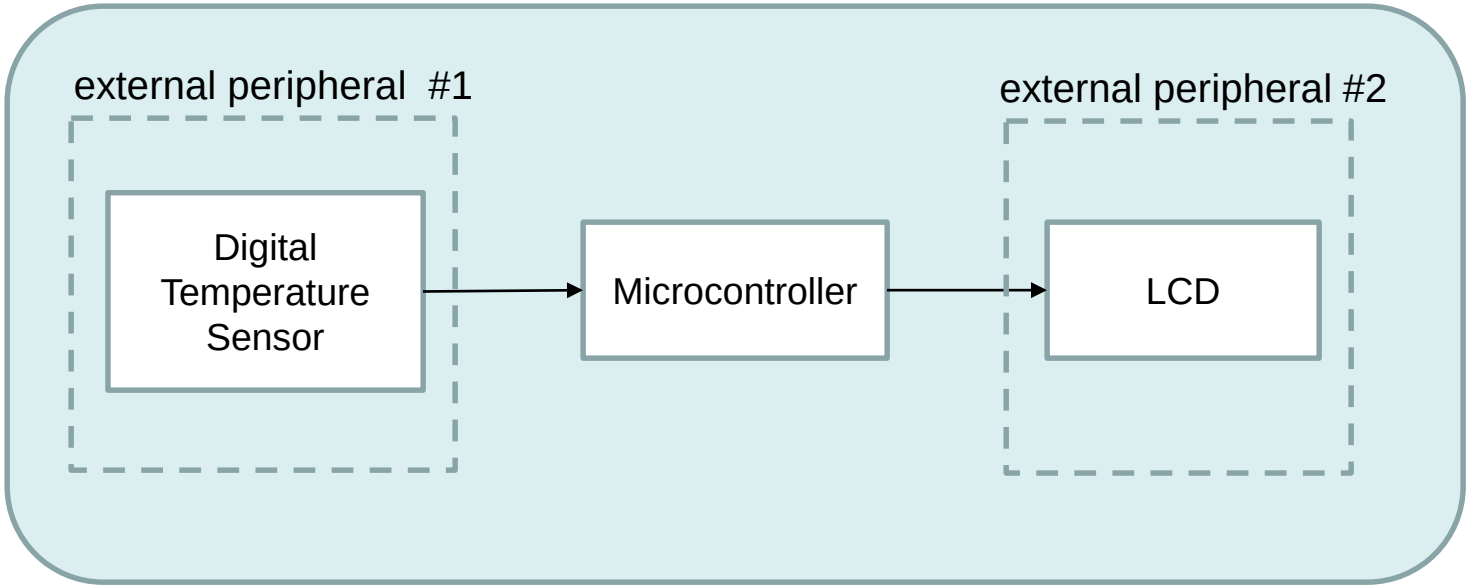
- “Embedded Communication” refers to the transfer of data between components of an embedded system



A basic embedded system

Embedded Communication
Why is it important?

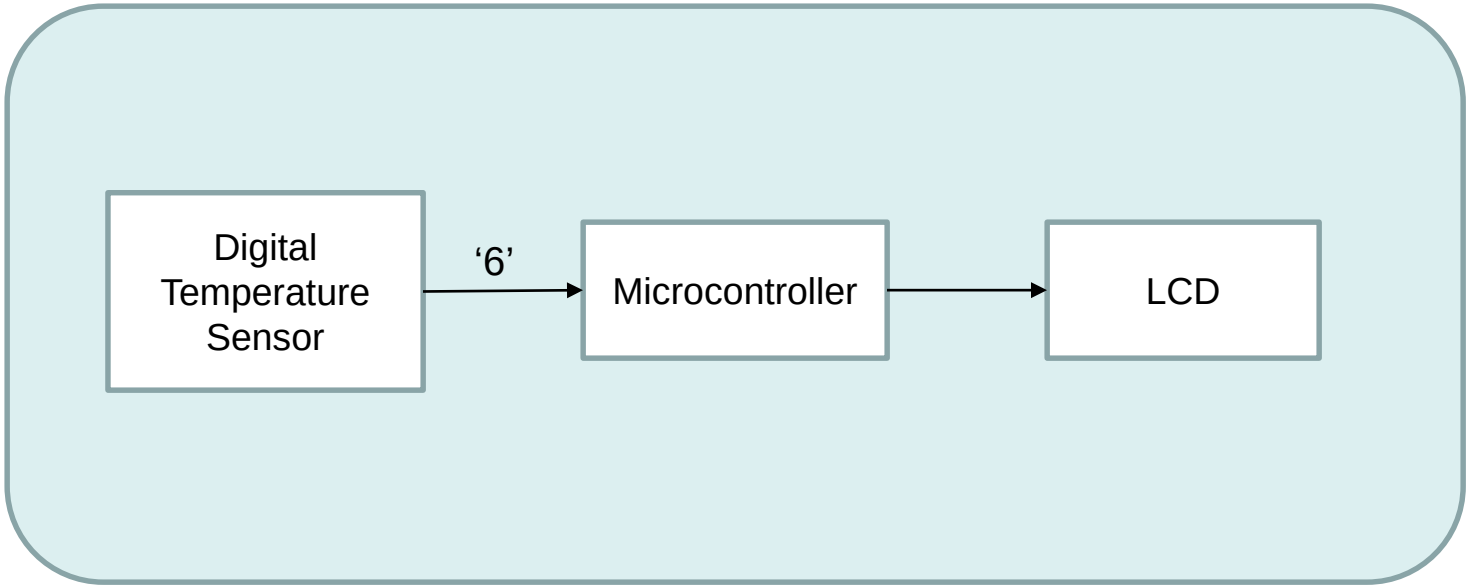
- Embedded Communication allows the microcontroller and the external peripherals to exchange data



A basic embedded system

Embedded Communication
Conceptual example

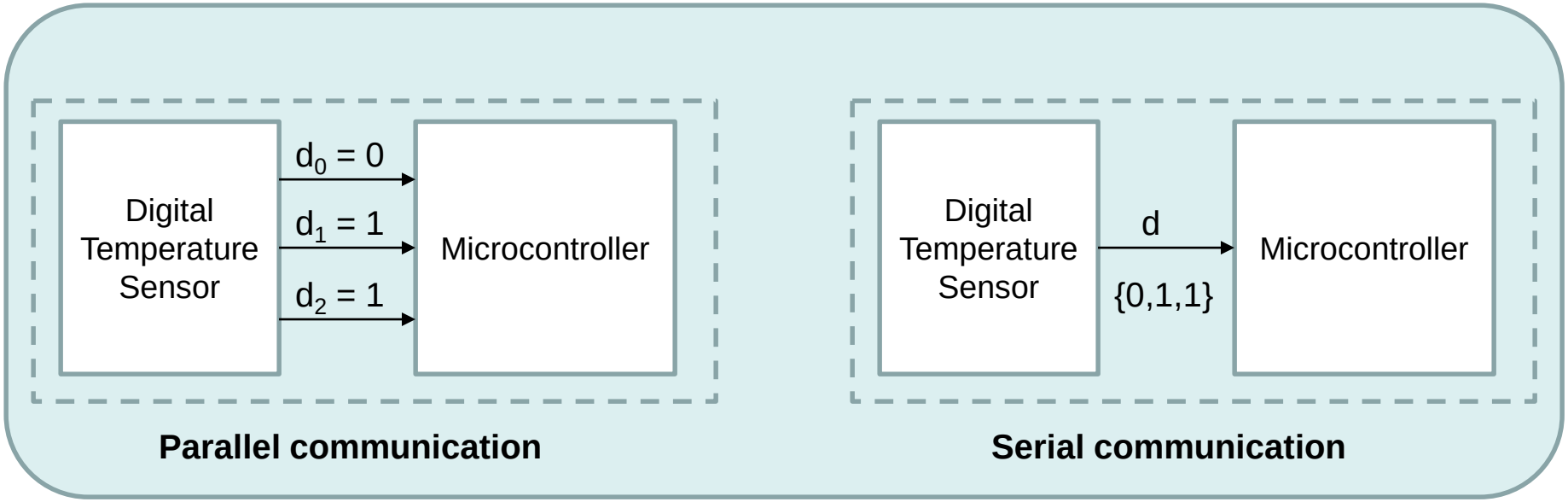
- Temperature sensor sends the value ‘6’ to the microcontroller that represents a temperature of 28°C



A basic embedded system

Embedded Communication
Types of data exchange

- Parallel communication
  - All data bits d_0, d_1, d_2 are transferred simultaneously
- Serial communication
  - One bit of data is transferred at a time
  - First LSB (‘0’), then ‘1’ and last MSB ‘1’

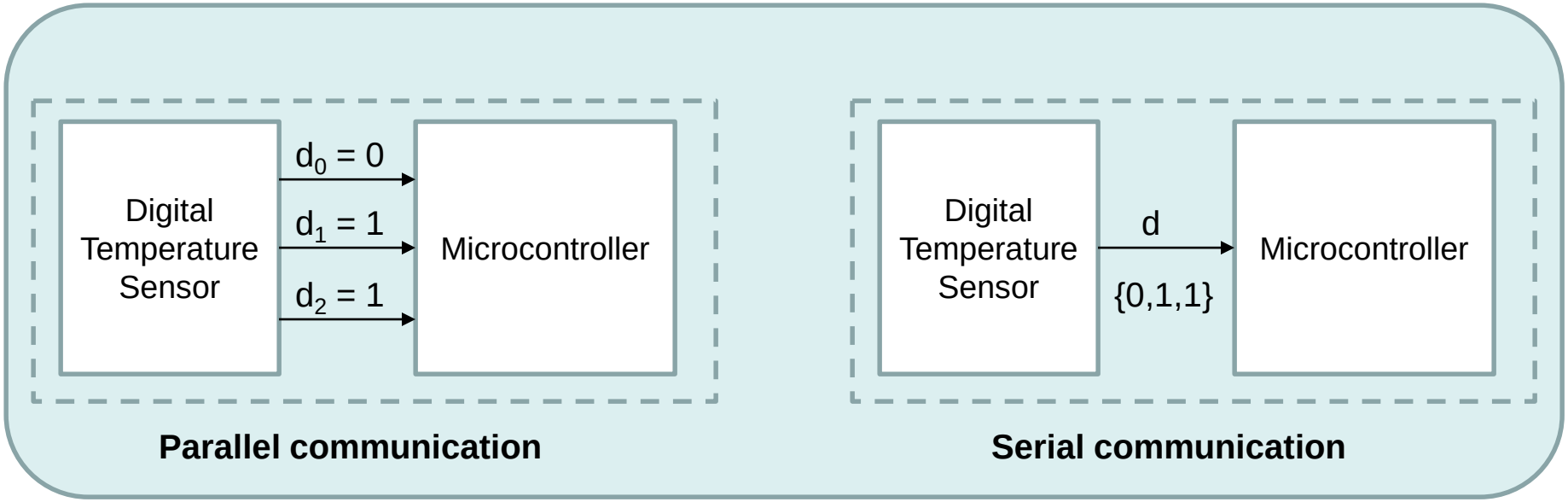


The temperature sensor transferring a value of ‘6’ to the microcontroller

Embedded Communication
Types of data exchange

- Parallel communication
  - All data bits d_0, d_1, d_2 are transferred simultaneously
- Serial communication
  - One bit of data is transferred at a time
  - First LSB (‘0’), then ‘1’ and last MSB ‘1’

In EEE3096, the scope is only serial communication



The temperature sensor transferring a value of ‘6’ to the microcontroller

Embedded Communication

Serial Communication: timing of data exchange

- Asynchronous transmission
  - Two devices need to agree on how long it takes to transmit one bit of information. No clock signal is shared between the two devices.
- Synchronous transmission
  - Two devices share a common clock that influences timing

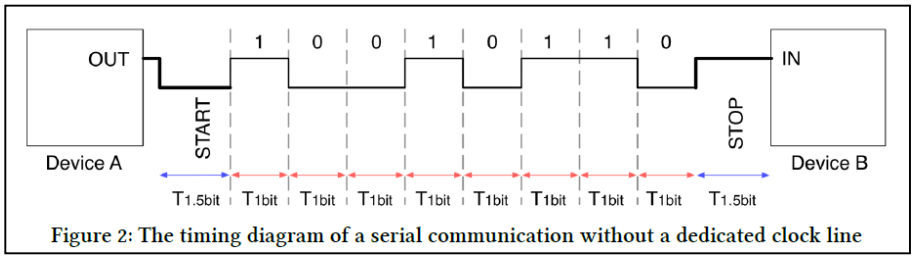


Figure 2: The timing diagram of a serial communication without a dedicated clock line

Asynchronous serial communication

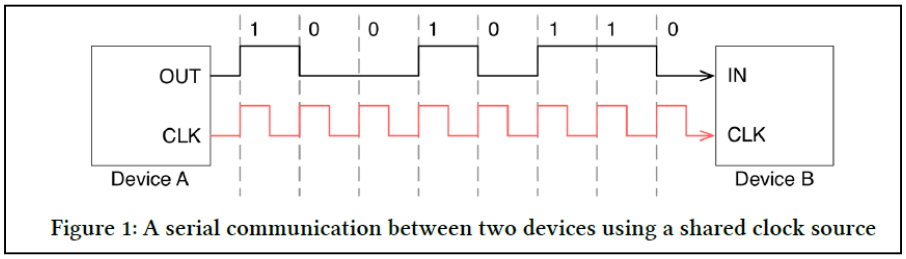


Figure 1: A serial communication between two devices using a shared clock source

Synchronous serial communication

[Images from “Mastering STM32” by Noviello]

Embedded Communication
Serial Communication: transmit/receive data

- Full Duplex: two devices can communicate simultaneously
  - Separate data lines for transmit (Tx) and receive (Rx)
- Half Duplex: two devices communicate, one at a time
  - Transmit and receive share the same data line
- Simplex: two devices communication in one directly only
  - Unidirectional communication (ie. only transmit or only receive)

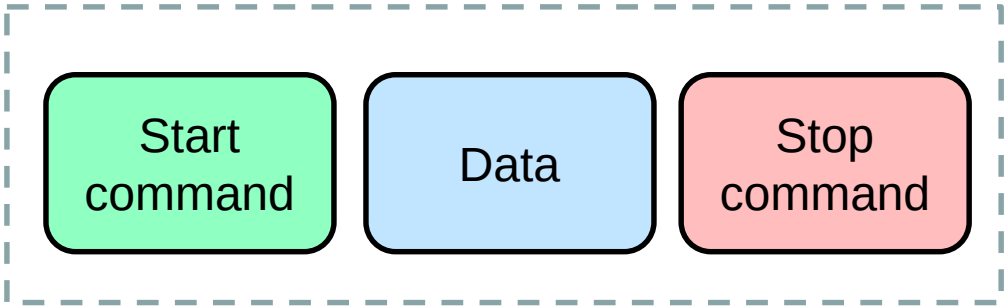
Full duplex

Half duplex

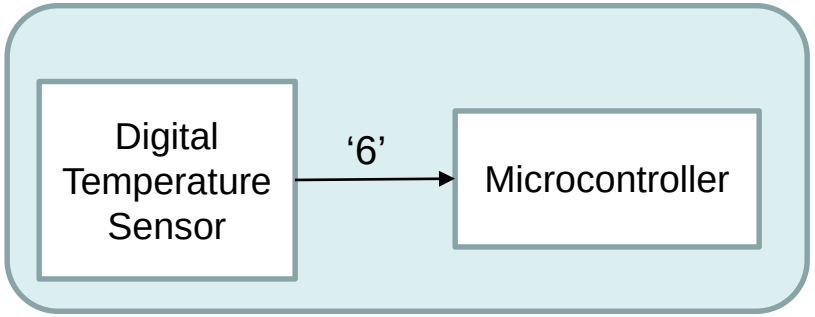
Simplex

Embedded Communication
A Basic Message Structure

- Data is transferred as part of a bigger message structure
  - **Start command**: signals the start of data transfer
  - **Data**: the data is transferred between devices
  - **Stop command**: signals the end of data transfer



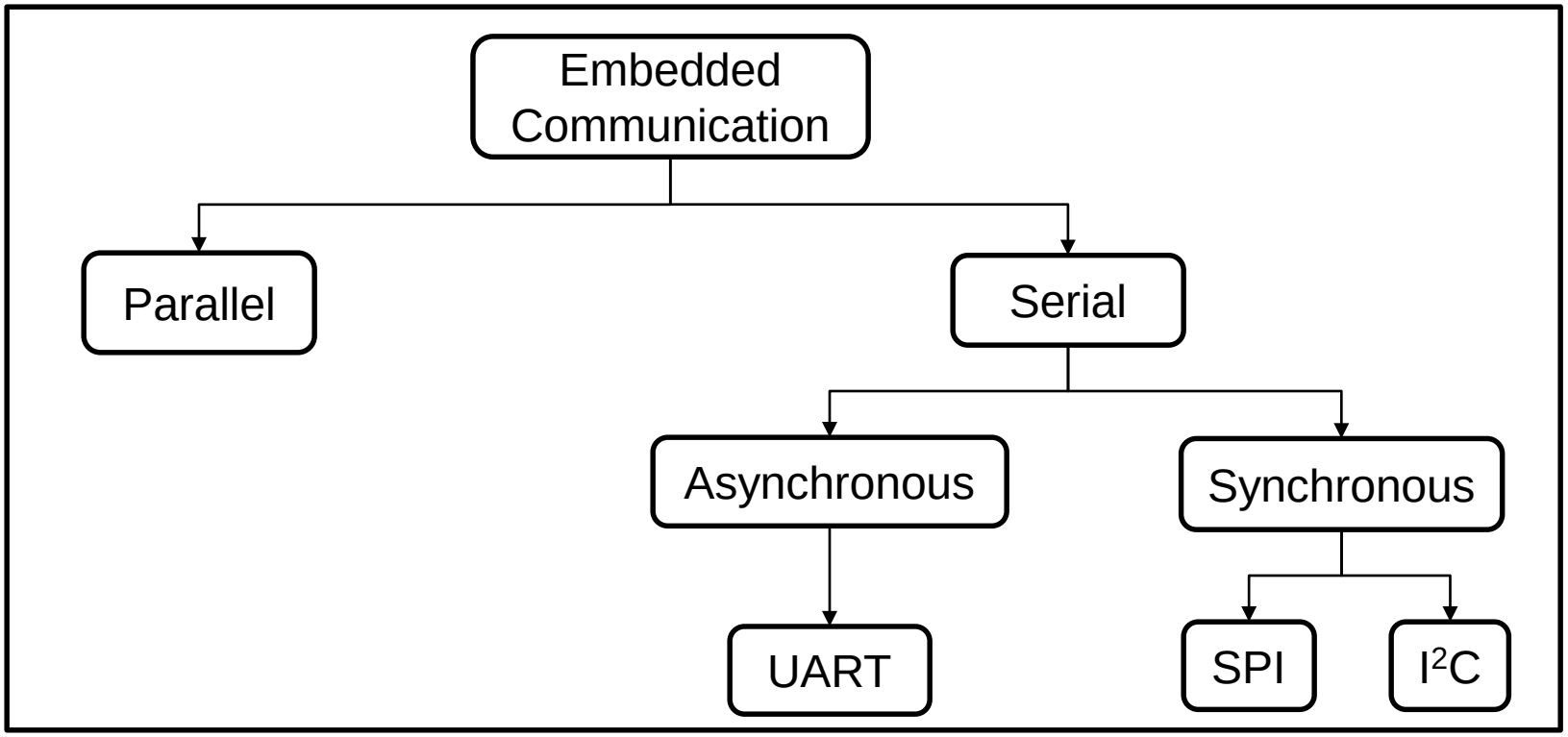
A basic message structure



A basic embedded system

Embedded Communication
Protocols

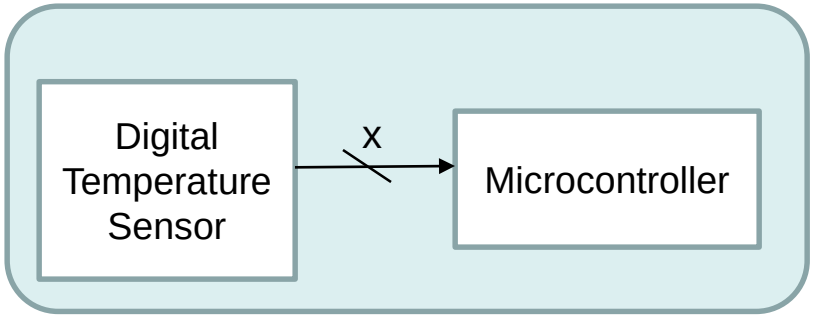
- Serial Protocols
  - **UART**: Universal Asynchronous Receiver Transmitter
  - **I²C**: Inter-Integrated Circuit
  - **SPI**: Serial Peripheral Interface



Overview of embedded communication

Embedded Communication
Properties of Serial Protocols

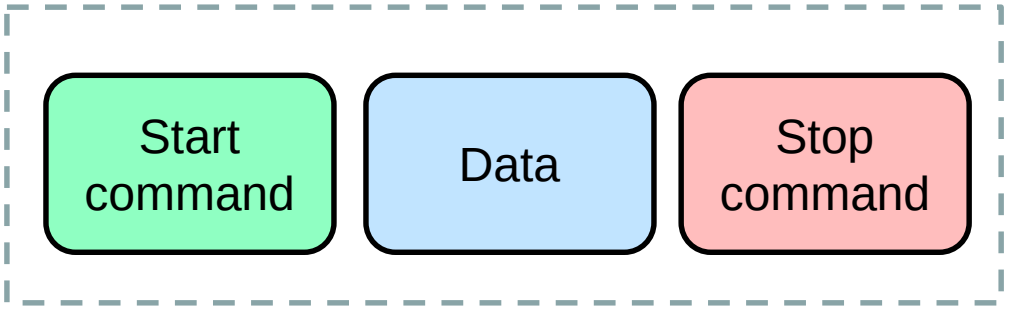
- Physical interface
  - Number of ‘lines’ x (data, clock, chip select, …)
  - Voltage levels
  - Connectors



Physical interface between two devices

Embedded Communication
Properties of Serial Protocols

- Physical interface
  - Number of ‘lines’ x (data, clock, chip select, …)
  - Voltage levels
  - Connectors
- Message structure
  - Start command
  - Data
  - Stop command
  - Other



A basic message structure

Embedded Communication
Properties of Serial Protocols

- Physical interface
  - Number of ‘lines’ x (data, clock, chip select, …)
  - Voltage levels
  - Connectors
- Message structure
  - Start command
  - Data
  - Stop command
  - Other
- Timing
  - When the next bit of data is sent
  - When the received signal is sampled

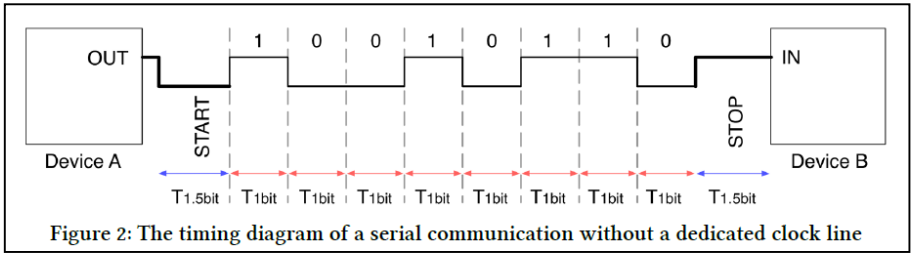


Figure 2: The timing diagram of a serial communication without a dedicated clock line

Asynchronous serial communication

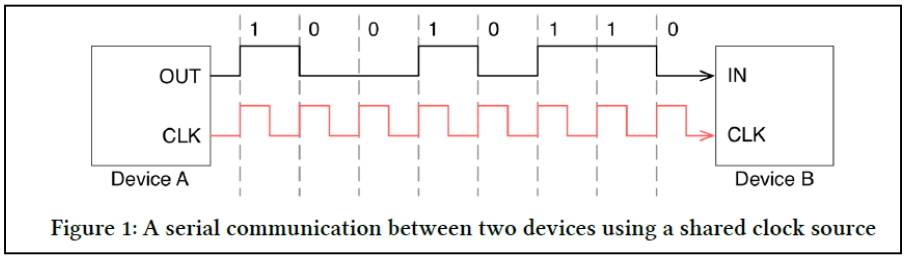


Figure 1: A serial communication between two devices using a shared clock source

Synchronous serial communication