MUNI FI

PV198 - UART I

One-chip Controllers

Daniel Dlhopolček, Marek Vrbka, Jan Koniarik, Oldřich Pecák, Tomáš Rohlínek, Ján Labuda, Jan Horáček, Matúš Škvarla

Faculty of Informatics, Masaryk University

8/2023

Intro

- Switch the branch to *Week_08*!
- Discussion of HW7

Embedded communication buses

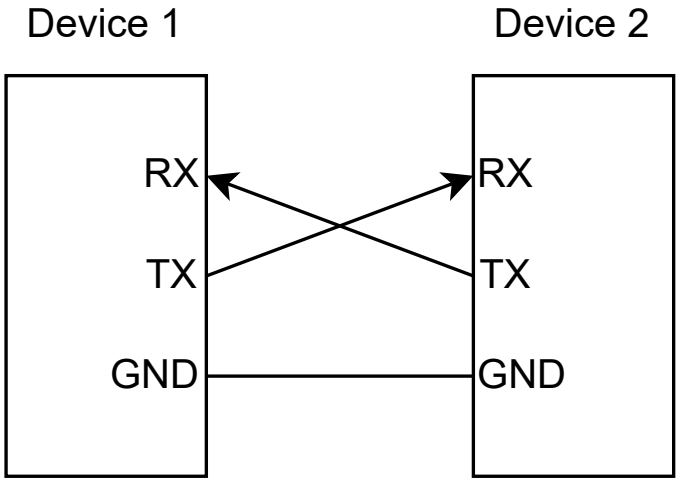
- SPI – Serial Peripheral Interface
- I^2C – Inter-Integrated Circuit
- **UART – Universal asynchronous receiver / transmitter**
- CAN – Controller Area Network
- 1-Wire
- RS-485
- RS-232

UART

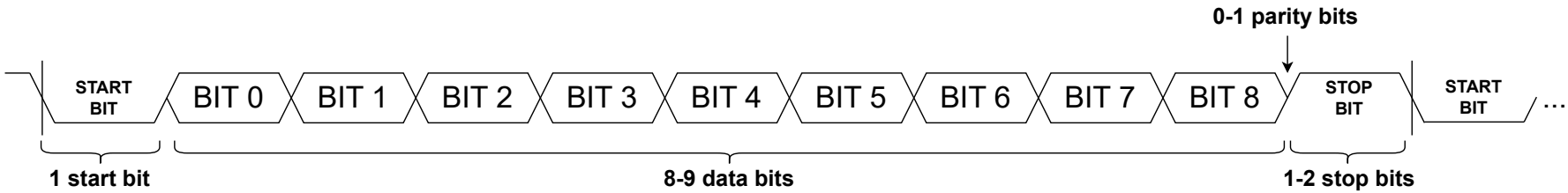
- UART = **U**niversal **A**synchronous **R**eceiver-**T**ransmitter
- Serial communication
- "Serial port" – old D-SUB 9 connector (UART = RS232 TTL)
- Application: intra-board communication
  - sensors
  - GPS
  - bluetooth
  - modems

UART Principles

- 2 wires
  1. RX – receive
  2. TX – transmit
- 1 to 1 communication
- No master nor slave, just 2 equal communication participants
- Works without clock signal
- Requires same UART configuration for both communicating devices (baud rate, parity, etc.)
- Asynchronous
- Full-duplex



UART Message



UART Settings

- Baud rate (typical 9600 baud/s – 115 200 baud/s)
- Number of data bits (8–9)
- Number of stop bits (1–2)
- Parity bit (disabled / odd / even)

Note – Baudrate vs. bitrate

- Bitrate = bits per seconds.
- Baudrate = symbols per second.

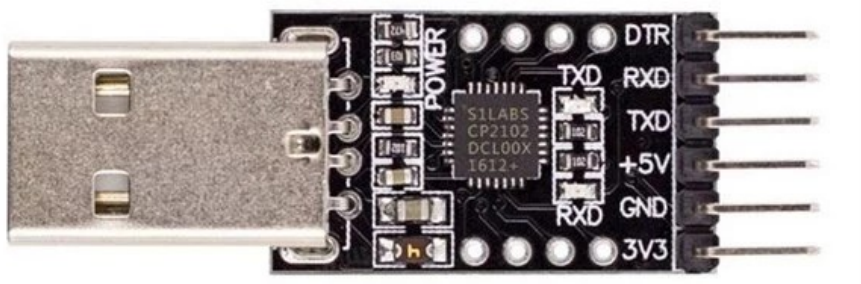
When bus allows more than 2 states (0/1) at one time, bitrate is higher than baudrate. E.g. bus with 4 voltage levels (2 bits at one time) with baudrate 1 kBaud/s has bitrate 2 kBits/s.

FRDM-K66F UART

- 5 UART modules
- RS-485 support
- Hardware flow control (RTS/CTS)
- 9-bit UART support
- Interrupts
- DMA support
- TX/RX FIFO
- MCUXpresso SDK API Reference Manual – UART driver

USB to UART Bridge CP2102

- We use it to connect MCU's UART to PC.
- Specification



Seminar task I

- Create an application that reads data from UART and sends the data back to PC.
- Update your code to rotate received character +2.

Seminar task I – Step-by-step guide

1. Create an empty FRDM-K66F project.
2. Setup pins routing & peripherals for UART communication.
   - PTB11 as UART3_TX, PTB10 as UART3_RX
   - UART3 with 8 bit data, 1 stop bit, no parity, 115200 baudrate
3. Connect “USB to UART bridge” to a board (based on pins routing).
   - Connect **just** RXD, TXD, GND.
   - Do not forget to swap RX & TX.
4. Connect “USB to UART bridge” to a PC, new COM port appears in Device Manager in “Ports (COM & LPT)”.
5. Program desired functionality.
   - MCUXpresso SDK API Reference Manual – UART driver
6. Open terminal application (e.g. PuTTy or Terminal view in MCUXpresso IDE) and connect to correct COM port with UART settings configured in MCU. Try programmed functionality.

Seminar task II

Task II

Write a PC application which communicates with your MCU.

1. Use same MCU program as in Task I.
2. Use Python, use `import serial`.
   - In case pyserial is not available:
     `python.exe -m pip install pyserial`
3. Send string to the device.
4. Read "encrypted" string back.
5. Check if device correctly "encrypted" the string.

Homework – RGB over UART

- On device side you will receive 3 bytes.
- These 3 bytes represent RGB values in order RGB.
- Your goal is to set RGB LEDs colors according to received values.
- Use PWM.
- Due to testing, set the Timer Output Frequency in the FTM peripheral to “262 144 Hz”

MUNI
FACULTY
OF INFORMATICS