
Chapter 10

Serial Communication



Serial Communication



What's the serial communication?

- It's a communication mechanism takes place using the **asynchronous protocol**
- In this communication 2 devices exchange data **serially**, 1 bit at a time
- These 2 devices exchange data using **2 shared data lines** and **1 common ground**



Serial Communication



- The **protocol is asynchronous** because **none of the shared lines carries clock**
- But both (transmitter and receiver) **have to agree prior to the communication** that how fast data will be sent/received
- This protocol allows **duplex communication** which means both can send data simultaneously.



Serial Communication



Why we need this protocol? What's the use-case?

- We are using this protocol to exchange data between MCU and host computer (i.e. our laptops)
- Unlike ITM protocol, this protocol let us send data from laptop to MCU



Serial Communication

Now a question raise. How fast we can send data using this protocol?

- This protocol works with **frames**
- Each frame has **1 start** bit, **1-2 stop** bits and **5-9 data** bits
- Speed of protocol is known as **baud rate** and it's quoted in **baud per seconds (bps)**
- Few baud rate **examples** are : 9600, 19200, 38400, 57600 and 115200 bps



Serial Communication

- To actually answering the question
- With a common configuration of 1-start bit, 8-data bits, 1-stop bit and a **baud rate of 115200 bps**
- Theoretically, we can send 11,520 frames per second
- But practically, data rates would probably be lower because of processing time on the slower side of communication (the MCU)



Serial Communication

Now the problem is our laptops don't support serial communication protocol!

- It means we **can't connect our laptop directly to the MCU**
- Here is the **serial module come in picture**
- Our **module will sit between the 2 devices and expose a serial interface to the MCU and a USB interface to the laptop**
- This is how both MCU and laptop will see each other as a serial device



Linux Tooling



Linux Tooling



It's time to configure our module to work with our board and laptop

- First step is to connect the module to laptop using a mini-B USB cable
- Run this command : `$ dmesg | grep -i tty`

```
[ 3063.967949] usb 3-4: FTDI USB Serial Device converter now attached to ttyUSB0
```

- **ttyUSB0** is a file created and associated with our module

```
crw-rw-rw-+ 1 root plugdev 188, 0 Feb  3 17:52 /dev/ttyUSB0
```



Linux Tooling



- If you are following along then you can test your configuration by sending a message to the module by running the command below
- `$ echo 'Hello, world!' > /dev/ttyUSB0`
- Now if all okay then you should see an led blink on the module as you enter this command (LED blink once)



Minicom



Minicom

Interacting with serial module using **echo** command is not a good approach. For this purpose we will use **minicom**

- We have to configure minicom before we can use it
- For configuring minicom we will create a file named “.minirc.dfl” in home directory
- After successful creation we will copy the content below in the file

```
pu baudrate 115200
pu bits 8
pu parity N
pu stopbits 1
pu rtscts No
pu xonxoff No
```



Minicom



NOTE Make sure this file ends in a newline! Otherwise, `minicom` will fail to read it.

Let's read the `minirc.dfl`

- `pu baudrate 115200`. Sets baud rate to 115200 bps.
- `pu bits 8`. 8 bits per frame.
- `pu parity N`. No parity check.
- `pu stopbits 1`. 1 stop bit.
- `pu rtscts No`. No hardware control flow.
- `pu xonxoff No`. No software control flow.

Now we configured the file, let's launch the minicom



Minicom

For launching minicom we will run this command :

```
$ minicom -D /dev/ttyUSB0 -b 115200
```

This command will tell minicom to open serial device at
`/dev/ttyUSB0`

Also it tells to set baud rate to 115200 and pop up a text-based user interface (TUI)



Minicom

```
Welcome to minicom 2.7

OPTIONS: I18n
Compiled on Sep  6 2015, 19:49:19.
Port /dev/ttyUSB0,

Press CTRL-A Z for help on special keys
```

```
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7 | VT102 | Offline | ttyUSB0
```

- Now we can send data using keyboard, go ahead and type something
- Now you'll observe, as you type in an LED on module will blink on each keystroke from keyboard



Minicom Commands

Minicom exposes commands via keyboard shortcuts. Few of them are:

- `Ctrl+A + Z`. Minicom Command Summary
- `Ctrl+A + C`. Clear the screen
- `Ctrl+A + X`. Exit and reset
- `Ctrl+A + Q`. Quit with no reset



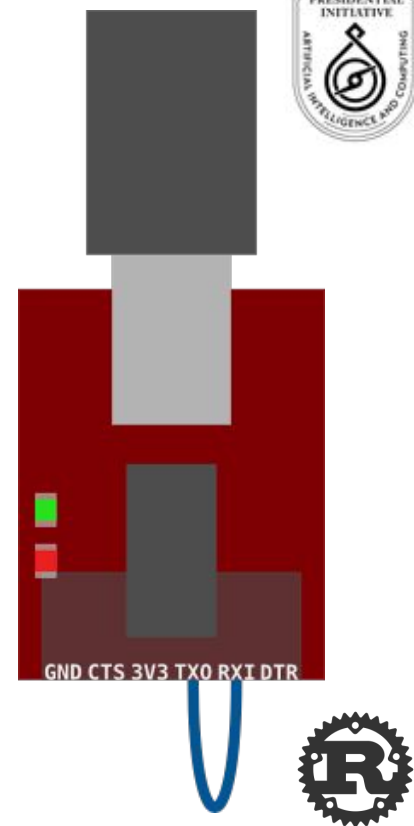
Loopbacks



Loopbacks

We already tested sending data to the module.
Now we try receiving data from serial module.

- For that first we have to connect **Tx** and **Rx** pins of the module using a jumper wire
- Once connected launch the minicom using the previous command
- Now if you hit any key you'll see that instead 1, now 2 leds are blinking
- Also now you can see keystrokes on your screen



Summary