

1 Communication Protocol

Communication occurs over a multiplexed UART interface.

In the following, each side of a block that can be connected to, will be called "port".

The amount of ports of a block will be denoted as P .

The communication can be extended to arbitrarily many ports.

1.1 Pinout

The following are the required pins on the microcontroller side:

- UART RX and TX
- $\lceil \lg(P) \rceil$ Pins for selecting which port is active
- P Pins for Interrupts

The following are the required pins between the blocks:

- VCC
- GND
- UART TX
- UART RX

1.2 Communication Process

For the following section the TX and RX of the microcontroller and block connection will be qualified by "uC" for the microcontroller and "bc" for the block connection respectively.

At any given time one of the ports will be selected (excluding during port selection transitions).

The selected port will have its bc_TX connected to uC_RX and bc_RX to uC_TX thus enabling full-duplex communication.

For the ports that are not selected the following configuration is used:

The bc_RX line will be pulled to high to indicate to the other block that a block is connected.

At the same time a floating detector is used on the bc_TX line to check if a block is connected.

The interrupt pin will be configured with a falling edge interrupt.

If you have data to send to the another block you set the given block to the active port and send a connection request frame.

Then you set a timer for 1ms during which you wait for a UART response frame.

If the other block does not respond before the timer runs out, you can move on to a different block or retry the current block.

(This should probably done with SYN -> ACK -> SYN-ACK where if the timer runs out you restart with SYN)

Here we have to make a case distinction:

- If the other block has this port as the selected port:
It will immediately get the frame and communication can start immediately.
- If the other block does not have this port as the selected port its RX will be connected to the interrupt pin.
The UART start bit of the frame will pull the pin low and trigger a falling edge interrupt on the other block.
If the other block is currently idle it will then change its selected port to the port where the frame was sent.
If the other block is currently communicating with another block, it will finish this communication with this block and then reinitiate the communication with the block that sent the request.
- If the block is currently changing the selected port from this block to another block, it will miss the communication request.
This will be caught by the timer reinitiating the request after some time.

1.3 Circuit Design

I did create a circuit design on paper but was not able to convert it into LaTeX Code yet.