Netze und Verteilte Systeme

Programmierprojekt

Dmitrii Polianskii, Lukas Lamminger

Universität Salzburg

Usage TX

C:

```
./TX portTX portRX packet_size packet_block_size send_delay file_name
```

java:

```
java TX portTX portRX packet_size packet_block_size send_delay file_name
```

- **portTX** - port to recieve acknowlegments (default: 4700)
- **portRX** - port to send datagrams (default: 4711)
- **packet_size** - size of a packet in Bytes (default: 1000)
- **packet_block_size** - amount of packets between delay (default: 100)
- **send_delay** - delay in microsec between blocks (default: 200)
- **file_name** - name of a file to transmit (default: to_send.jpg)

Usage RX

C:

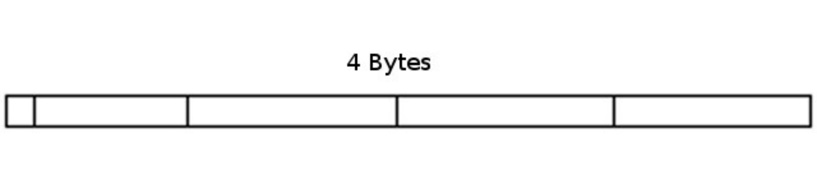
```
./RX portTX portRX
```

java:

```
java portRX portRX
```

- **portTX** - port to send acknowlegments (default: 4700)
- **portRX** - port to recieve datagrams (default: 4711)

Header structure



A header consists of 4 bytes.

First bit is used to indicate the last packet.

31 bits left for sequence number

TX description

TX description

1. Read file
    1.1 Read file in buffer
    1.2 Calculate CRC32 and add to filebytes
    1.3 Split filebytes in packages
2. Initialize UPD Socket
3. Initialize Acknowlegments array to obtain transmitted packets
4. Transmit one block of packets
    4.1 For every packet in block check if acknolegment was recieved, if not - send a packet.
    4.2 If last packet is reached, then start with first again,
5. Wait for acknowlegments {DELAY} microseconds.
    5.1 Write every sequence number from acknowlegment packet in Acknowlegments array,
    5.2 If all packets were acknowlegment - end transmission. Else - goto punkt 4.

RX description

RX description

1. Initialize UPD Socket
2. Listen for incomming packages
   2.1 Write databits from package in a memory
   2.2 If last-package-bit was seen, the size of file and Amount of packets can be defined.
   2.3 If not all of package were recieved, then goto punkt 2.
3. Assemble a file
4. Calculate CRC32 and compare with recieved one.