10/21/2012

SPANG

PROTOCOL DOCUMENTATION

Version 1.0 | Spang

Table of Contents

The connection procedure

First we divide the connection process into two parts; the connector, and the listener.

The listener waits for an incoming connection and the connector connects to the listener. See Figure 1 Connection procedure to the right.

Messages

Unordered Protocol

Header:

0-9	9-10	10+
Udp header.	Flag bits. With the 0x01 bit active.	Data
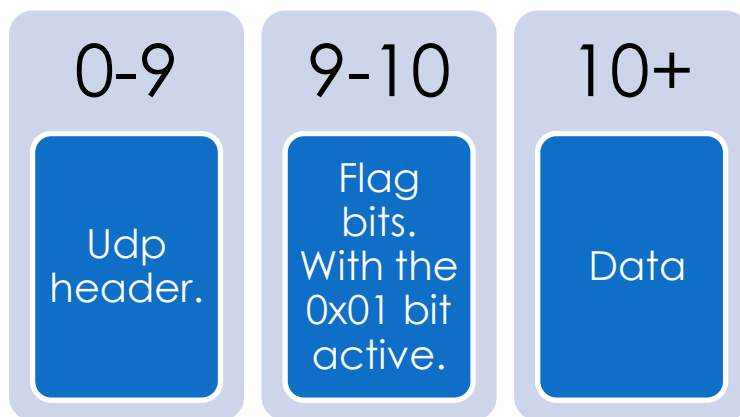
No check is made if the message arrived or not and nothing is done if messages arrive out of order.

This is simply UDP with an extra header byte to conform to the protocol standard.

Ordered Protocol

Header:

0-9	9-10	10-13	14+
Udp header	Flag bits. With the 0x02 bit active.	Sequence numbers.	Data

1. The listener listens for a connection request at a specific port.

2. The connector sends a connection request to the listener.

3. The listener receives a connection request.

4. The listener creates a new UdpSocket with an available port number.

5. The listener sends the port number of the newly created UdpSocket to the connector.

6. The connector receives the port number and from it connects the underlying UdpSocket to the new port.

7. The connector sends a message aknowledging that a port has been received to the server.

8. The connector creates a Connection from the UdpSocket and the Connection is done onthe connecting side.

9. The connector sends a connection complete message

10. The listener recives the compleate message.
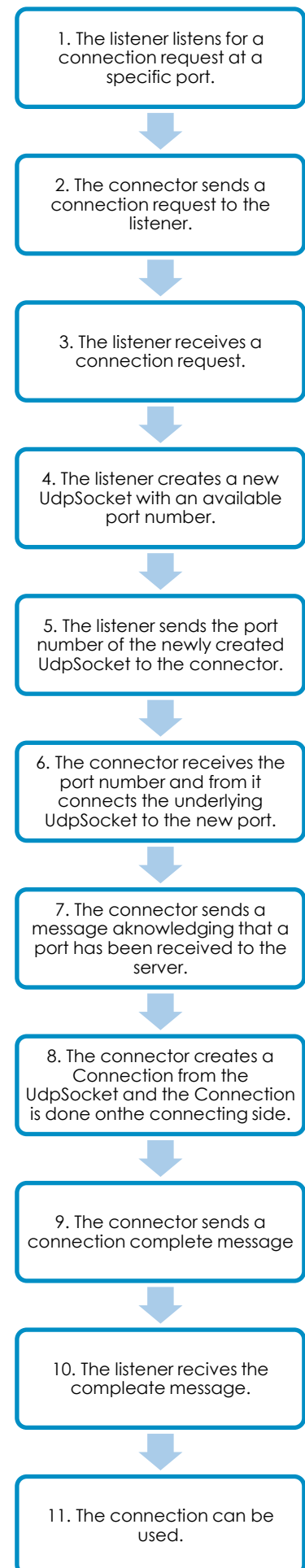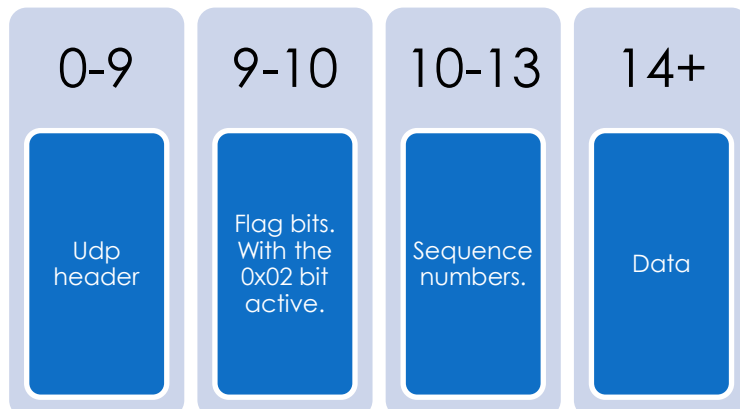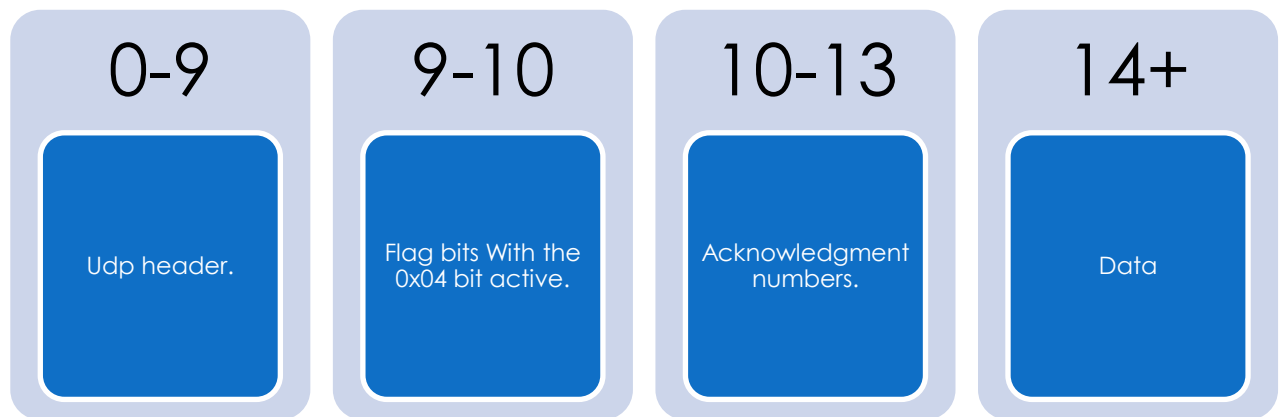
11. The connection can be used.

Figure 1 Connection procedure

The sequence numbers starts at 0 and increases by 1 for each message sent. The message is discarded if a message is received out of order. Other than that it's simply normal UDP.

Reliable Protocol

Header:

0-9	9-10	10-13	14+
Udp header.	Flag bits With the 0x04 bit active.	Acknowledgment numbers.	Data

The reliable protocol ensures that messages are received and that no duplicate messages are processed (messages with same acknowledgment numbers are duplicate messages).

Acknowledgment message

When a message is received a callback message is sent back called the "ACK message" (Acknowledgment message).

The "ACK message" is a reliable message that has the same acknowledgment numbers as the message received with the ACK bit (0x10) active.

Sending

Like the sequence numbers from *Ordered* Protocol the acknowledgment numbers start at 0 and increase by 1 for every message.

When a message is sent it is stored in a temporary collection until an "ACK message" for the sent message has been received from the connection endpoint. If no "ACK message" has been received during a certain interval, the message is resent. This procedure continues until an "ACK message" has been received.

Receiving

When a message is received, an "ACK message" is sent in response. After this is done a check is made to ensure that the message has not been received before. If the message is a duplicate the message gets discarded.

OrderedReliable Protocol

Header:

0-9	9-10	10-13	14+
Udp header	Flag bits with the 0x08 bit active	Acknowledgment/ Sequence numbers	Data.
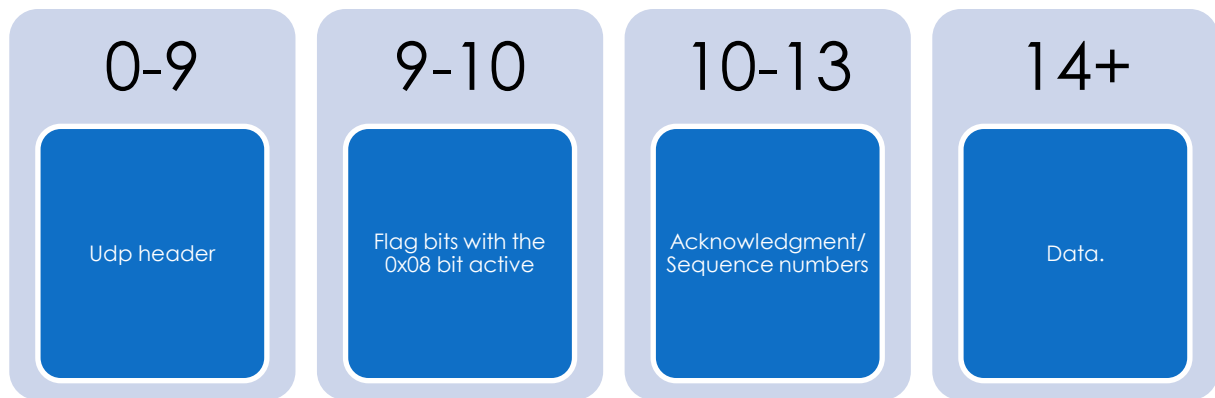
The *OrderedReliable* protocol acts much like the *Reliable* protocol with the exception that incoming messages are only processed if they arrive in order. To achieve this, received messages that arrive out of order are stored until the missing message(s) arrives.

In this protocol acknowledgment numbers act as sequence numbers and through them the receiving end can deduce if the message is in order or out of order.

Shutdown

When an endpoint wants to shut down, a shutdown message is sent to the other endpoint. A shutdown message is simply a message whose SHUTDOWN bit (0x20) is enabled.

This message can be sent using any protocol but is typically sent using *Unordered*, since the shutdown should be completed as soon as possible.

When a shutdown message is received the connection shuts down and is no longer usable. It is appropriate to notify the user of this through an exception or an error code.