Author: Thorsten Sommer, M.Eng.
Contact: vektoren@googlemail.com

Howto: Send OSC-UDP data to Max MSP

1. Send text messages

First it is important, that the entire data block begins with a string. The easiest way for that is to use an address at the beginning: `/0` for example. This address must be inside a 4 byte block, that means as string: `/0\0\0` or as byte array `2F 30 00 00`!

Let's say that we want to send this message: `This is a test!`. OSC understand that of a series of 4 strings: `This`, `is`, `a`, `test!`. To tell OSC that now 4 strings follow, we have to note this in the data block: `,ssss`. But this type information must be also inside a 4 byte block or must be aligned to 4 byte blocks. In this case, `,ssss` has the length 5. To align it we make that: `,ssss\0\0\0` or as byte array `2C 73 73 73 73 00 00 00`.

Okay, and now we add the message. But between the 4 strings we have to add at least one zero byte `00` and not a space! Attention please: Each single word must be inside a 4 byte block or it must be aligned! Let's check:

`This` has the length of 4 bytes, ok. But we need at least one zero byte to the next word. But this would be 5 bytes, which is not possible - we have to align to 8 bytes: `This\0\0\0\0` or `54 68 69 73 00 00 00 00`!

`is` has the length of 2 bytes, so we fill it up to 4 bytes: `is\0\0` or `69 73 00 00`. The same with `a`: `a\0\0\0` or `61 00 00 00`. `Test!` has 5 bytes so with make 8 bytes: `Test!\0\0\0` or `54 65 73 74 21 00 00 00`.

The entire data block looks like that as string:

`/0\0\0,ssss\0\0\0This\0\0\0\0is\0\0a\0\0\0Test!\0\0\0`

Or as byte array:

`2F 30 00 00 2C 73 73 73 73 00 00 00 54 68 69 73 00 00 00 00 69 73 00 00 61 6E 00 00 54 65 73 74 21 00 00 00`.

Another rule: If you have an address or list of types with the length 4, you have nevertheless to add at least one zero byte and align it again. Example: We want to use `/abc`, then we need to use `/abc\0\0\0\0` or `2F 61 62 63 00 00 00 00`.