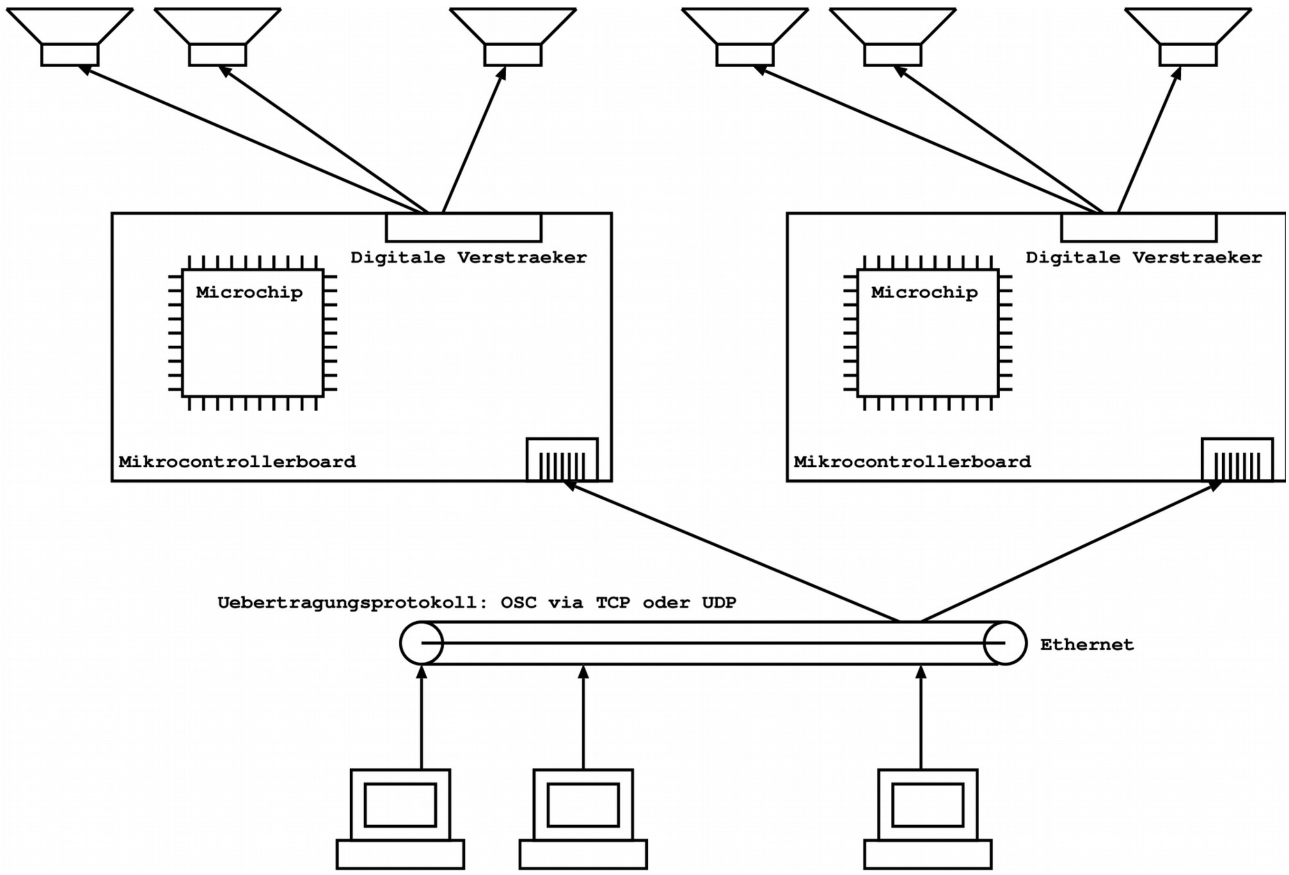
Audio over Internet using OSC

for

Computer and Embedded Microcontroller

(preliminary abstract, internal – 20.4.2009)

Wolfgang Jäger, Winfried Ritsch



First of all, there has to be developed an adequate definition for the OSC based protocol. Since the data is transported via UDP or TCP/IP there is no possibility of guaranteeing the duration which a packet takes to arrive at the receiver or if the order of the received data is in it's original sequence. Therefore a time stamp is required. Fortunately the OSC-Bundle contains a precise TimeTag and regulation rules for processing the sequence.

Furthermore a structure for the content of each audio-data-bundle has to be found. To keep the framework as general as possible, the following solution has been derived:

The first message of each bundle contains the sampling rate, block size, overlapping factor and the number of channels which follow in the audio-data-bundles content. The first three parameters are based on the [block~]-object of PureData. So the block size of each audio-data message may be varied, as well as the sampling rate at the receiver side or for more redundancy even an overlapping of the audio-data may be implemented. These parameters are particularly required if the audio-data is transmitted in the spectral domain.

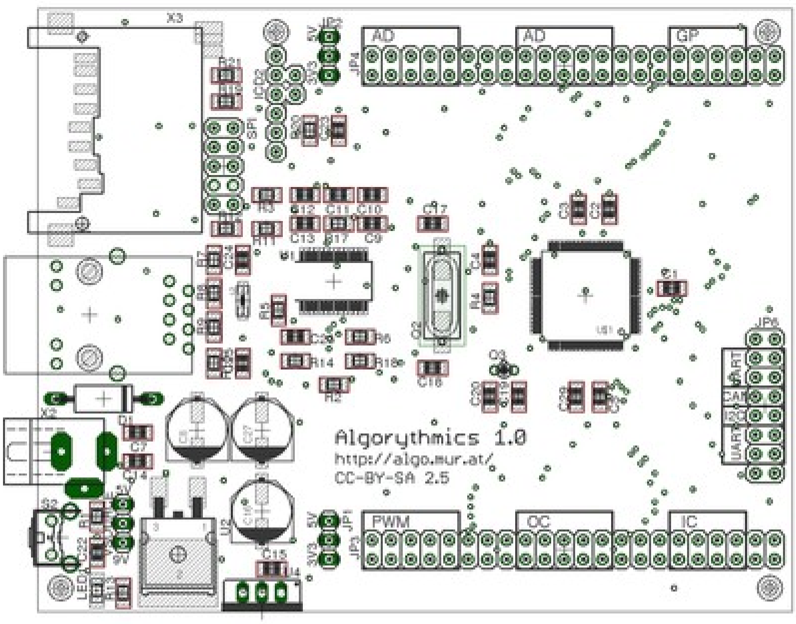
After the first OSC-Message, which contains the instruction data, the actual data is transmitted. The address is followed by the required OSC-TypeTag, an resampling factor (proportional to the sampling rate argument in the instruction message) to transmit single channels with a reduced bandwidth and at the end of the message an OSC-Blob, which contains the audio-data with its resolution.

The blob has been chosen, because it's the most flexible solution within the OSC-Definition. The arrangement of the audio-data could be arbitrary, limited only by the restriction that its size has to be a multiple of four bytes.

After developing the principle structure of transmitting audio over OSC, what happened until now?

In cooperation with devloper of the OSC-library for Pure-Data, Martin Peach, the "blob" type within the [packOSC]-object of PureData has been made. An [blob]/[unblob]-object has been written, which packs/unpacks the audio-data, according to the required resolution and prepare it for the transfer through the [packOSC]-object.

As a further approach, this functionalities should be implemented in the “Escher” Microcontroller, at the moment a 32F in conjunction with the FET-drivers for 8-16 channels. For this the uOSC library from CNMAT Berkely should be used, if available.



escher microcontroller board “Algorythmics”