

1 Help on reading audio files into Matlab

You can read audio files into a Matlab-vector using the following command

```
audio_vector=auread('filename.au');
```

Note that this only works for MuLaw encoded files which have the file-extension “.au”. *Auread* is a built-in Matlab function.

2 Help on playing audio files

In order to play audio files within Matlab, you can use the following command

```
sound(audio_vector);
```

By default *sound* assumes a sampling rate of 8192Hz. A different sampling rate may be specified with a second argument. For example, if the signal was sampled at a rate of 16 KHz, the following command could be used.

```
sound(audio_vector, 16000);
```

Another useful command, *soundsc*, can be used to prevent clipping that may occur if the values in the audio signal are too large. This command rescales the audio signal before playing it in order to place it within the dynamic range of the hardware. The syntax is the same as for the *sound* command.

3 Help on writing audio files from Matlab

You can write the audio vector y to a MuLaw encoded file by typing

```
auwrite(y,'filename.au');
```

Note that you have to use the file-extension ‘.au’ in order to be able to load the signal back using *auread*. *Auwrite* is a built-in Matlab function.

4 Example

Suppose we have an audio file *jimi.au* which we want to load into Matlab. We use

```
x=auread('jimi.au');
```

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

Now lets filter the signal.

```
y=zeros(1,length(x));
y(1)=0.2*x(1);
for i=2:1:size(x)
    y(i)=0.8*y(i-1)+0.2*x(i);
end
```

We can then play the filtered signal using

```
sound(y);
```

We can save the filtered signal to a new file using

```
awrite(y,'jimi_low.au');
```

5 Matlab Help on sound

SOUND Play vector as sound.

SOUND(Y,FS) sends the signal in vector Y (with sample frequency FS) out to the speaker on platforms that support sound. Values in Y are assumed to be in the range $-1.0 \leq y \leq 1.0$. Values outside that range are clipped. Stereo sounds are played, on platforms that support it, when Y is an N-by-2 matrix.

SOUND(Y) plays the sound at the default sample rate of 8192 Hz.

SOUND(Y,FS,BITS) plays the sound using BITS bits/sample if possible. Most platforms support BITS=8 or 16.

Example:

```
load handel
sound(y,Fs)
```

You should hear a snippet of Handel's Hallelujah Chorus.

See also soundsc, wavplay.