Playing Sounds in Excel Using VBA #1

by George Lungu

– In order to create sound effects in Excel models we need to download various sound files off the internet, store them (*preferably in the same directory with the excel file*) and have a VBA script access and play them when certain logic conditions in the worksheet or in other VBA macros occur.

– There are ways to play mp3 files of midi files but the most straightforward type of file to play from VBA is .wav

– You can find plenty of sites which offer free .vaw sound effects by typing "free sound effects wav" on Google

– You need to start the VBA code with the PlaySound API function call, which all has to be typed in one line (an underscore will physically break the line in two but will keep the 1-line code functionality):

```
Private Declare Function PlaySound Lib "winmm.dll"_
    (ByVal  lpszName As String, ByVal hModule As Long, ByVal dwFlags As Long) As Long
```

Arguments:

Indicates the file name and file path to play

Specifies options for playing the sound using one or more of the flags explained in the next page

Besides the first argument which is the file name and path, this function has flag arguments by which its run options can be controlled. You could declare these as constants as follows (or just use the number):

SND_SYNC = &H0 - The sound is played synchronously and the function does not return until the sound ends.

SND_ASYNC = &H1 - The sound is played asynchronously and the function returns immediately after beginning the sound.

SND_NODEFAULT = &H2 - If the sound cannot be found, the function returns silently without playing the default sound.

SND_LOOP = &H8 - The sound will continue to play repeatedly until PlaySound is called again. You must also specify the SND_ASYNC flag to loop sounds (use the compound condition: SND_ASYNC or SND_LOOP).

SND_NOSTOP = &H10 - If a sound is currently playing, the function continue playing the old sound and it will immediately return False without playing the newly requested sound.

<www.excelunusual.com>

Downloading sound files:

- Create a new folder called "Sounds". In this folder you need several .wav files downloaded off the internet.

- I recorded and saved the sounds from **AT&T Labs Natural Voices® Text-to-Speech Demo**:

(http://www2.research.att.com/~ttsweb/tts/demo.php)

- I also used:

- http://freesoundvault.com/FreeSoundPack.html, http://probangers.com/ http://simplythebest.net/sounds/WAV/sound_effects_WAV/index.html, http://www.webplaces.com/html/sounds.htm, http://www.wavsource.com/ http://www.pachd.com/sounds.html, http://www.hongkiat.com/blog/55-great-websites-to-download-free-sound-effects/

- I saved seven files as following: one.wav, two.wav three.wav, four.wav, five.wav, six.wav, seven.wav.

- There are also there other files there: effect_1.wav, effect_10.wav and effect_14.wav. These are short sounds used in the pong game.

AT&T Labs Natural Voices® Text-to-Speech Demo

2

Three basic sound macros:

- Open a new Excel workbook and save it in the same folder (Sounds).
- Insert a module (Module1) and after the function and constants declaration in Module1 write the three macros shown to the right.
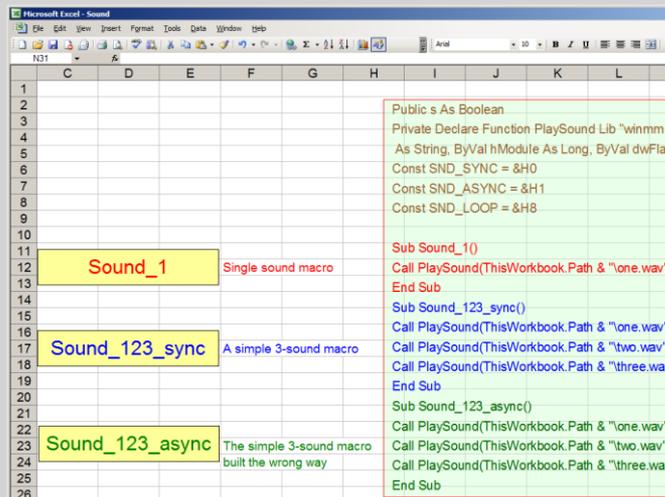

```
Public s As Boolean
Private Declare Function PlaySound Lib "winmm.dll" (ByVal lpszName
 As String, ByVal hModule As Long, ByVal dwFlags As Long) As Long
Const SND_SYNC = &H0
Const SND_ASYNC = &H1
Const SND_LOOP = &H8

Sub Sound_1()
Call PlaySound(ThisWorkbook.Path & "\one.wav", 0&, SND_SYNC)
End Sub
Sub Sound_123_sync()
Call PlaySound(ThisWorkbook.Path & "\one.wav", 0&, SND_SYNC)
Call PlaySound(ThisWorkbook.Path & "\two.wav", 0&, SND_SYNC)
Call PlaySound(ThisWorkbook.Path & "\three.wav", 0&, SND_SYNC)
End Sub
Sub Sound_123_async()
Call PlaySound(ThisWorkbook.Path & "\one.wav", 0&, SND_ASYNC)
Call PlaySound(ThisWorkbook.Path & "\two.wav", 0&, SND_ASYNC)
Call PlaySound(ThisWorkbook.Path & "\three.wav", 0&, SND_ASYNC)
End Sub
```

- Draw three rectangles labeled with the name of each macro and assign each of the macros to them.
- The first macro is the simplest sound playing macro.
- The second and the third macros are almost identical except for sync/async option. You can play them and see that while macro "Sound_123_sync" sounds right the last macro only plays the sound "three". This is because in this last macro a sound function returns from the beginning and the next function takes over. This is what happens with all the PlaySound functions except for the last one which can finish playing because there is no outer sound function to take over.

<www.excelunusual.com>

to be continued…