

DAC TESTING DOCUMENTATION

Current as of April Fools Day, 1979
Filed on: [IVY] <Note>Dac>DacTestingDoc.press

For the benefit of those who are testing, tuning, adjusting, debugging, amplifying or otherwise expending their efforts on the NoteTaker DAC, here is a bit of documentation on the dac programs I have worked on which I feel may be of some help to you and on the Smalltalk interface which I use to call them.

WHERE IT IS ALL TO BE FOUND

All of the programs and files referred to here can be found somewhere on [IVY] <Note>, but the subdirectory will depend on just the kind of program or file it is.

THE SMALLTALK INTERFACE

I use Smalltalk messages to initialize and invoke all my 8086 dac programs. The necessary code (plus a bit more) will be found on [IVY] <Note>Dac>Emulator-Dac-Music.st. This file will need to be filed into your Smalltalk system. (Ask any Smalltalk user for help on how to do that.)

PRODUCING A SINGLE TONE

There are two pieces of machine code which are intended to produce sustained tones at something approaching maximum volume. These are found on [IVY] <Note>Diag>MB>ToneDacTest.mb and [IVY] <Note>Diag>MB>ToneDacTest6.mb. The second file plays prettier tones but the first one should produce very pure tones (simple sine wave) suitable for adjusting the amplifier or whatever. The first of these files is invoked by a Smalltalk command like:

```
self dacPitch:28.
```

or:

```
self dacTestPitch:28 forDuration:15.
```

All pitches are given in discrete integers and are not frequencies, but rather surrogate note names, giving the number of the note counting up from 1—the 3rd 'A' below 'middle C'. (Thus, note #28 is middle C). Note numbers greater than 61 are not permitted. The duration (in the second type of command) is something like seconds.

The second program mentioned above (the one with the prettier tone) is called by one of the following types of Smalltalk messages:

```
self dacPitch6:28.
```

or:

```
self dacTestPitch6:28 forDuration:15.
```

(See the pattern?)

TESTING THE WHOLE FREQUENCY RANGE AT DIFFERENT VOLUMES

I suggest doing what the previous line says, to get an idea of what the amplifier does to the dac's signal at different frequencies. This is done by the various programs found in the dump file: [IVY] <Note>Diag>MB>BigSlideMbs.dm. Load that file onto your disk. The various programs can now be invoked by Smalltalk commands such as:

```
self dacSlide:80.
```

or:

```
self dacSlide:24.
```

The 80 and the 24 here mean that the signal produced by the dac should be exactly 80 percent or 24 percent of its maximum level.

The various possible percentages at which you may test are: 100, 80, 48, and 24. Supplying any other values should result in a Smalltalk error message.

TESTING AT A PRECISE DESIRED FREQUENCY

What if you want to find out whether the dac is sampling at just the right speed? The program on [IVY] <Note>Diag >MB >DacAmplitudeTest.bca allows you to specify the precise frequency (between 30 Hz and 16000 Hz) that you want to hear and also to choose between two different volumes (hence the slightly misleading file name). It is invoked by a Smalltalk command like:

```
self dacTestAmplitude: 100 freq: 440 dur: 15.
```

This command should have the result of producing a 15 second tone at 440Hz at full (100%) volume. The volume options for now are only 100 and 48. As mentioned above, the frequency may be any number between 30 and 16000.

PLAYING A TUNE

Alright, enough of this boring simple tone generation. A tune can be played in either of the tone qualities mentioned above. The pure sine tone instrument will play your piece if you properly invoke [IVY] <Note>Dac >TuneDacTest.mb, and the prettier instrument will play it using [IVY] <Note>Dac >TuneDacTest6.mb. See if you can guess which of the Smalltalk commands calls which program.

```
self playFile: 'GodRest.mu' load: 'y'.
```

```
or:
```

```
self playFile6: 'GodRest.mu' load: 'y'.
```

No trouble.

[IVY] <Note>Dac >MuFiles.dm is a dump file of one-voice (melody only) pieces suitable for playing by these two programs. Load any of them you want to hear onto your disk. One of them is 'God Rest Ye, Merry Gentlemen' and the file name is 'GodRest.mu'. To play a piece, you must include its file name (enclosed in single quotes) in the Smalltalk command, as shown above. If you have not played this song recently, the 'argument' after 'load:' should be a 'y' (enclosed in single quotes as shown in the examples). If you have just just played the tune (so that it is still in its assigned place in the NoteTaker's memory), you may put an 'n' where the 'y' is and save yourself the wait for the piece to get loaded.

In case you are interested, the format of a piece is as follows. Each note is represented by two bytes, the first representing the pitch (as above, some number between 0 and 61, where 0 represents a rest) and the second number representing the duration (in something like ninetieths of a second). You are invited to make up your own pieces if you have the patience. I would be happy to receive requests for more pieces (which I can make up more easily than you can) sent to me over the msg system (msg McCall). We ought to have a NoteTaker theme song.

Any questions or suggestions happily entertained by Kim McCall at PARC (ext.4339).