Lab 6 - Winter 2017 - EECS 363 Digital Filtering – formally due Friday March 10.

The formal due date is the last day of classes. You may hand it in and give me the demo any time before then. This lab depends on the results of the last Matlab problem I assigned.

In this lab we implement an IIR filter using the DSP capabilities of the board's aic3204 codec. We also implement the loopback aspect via the codec, so main does little more than configure the codec registers. Once the I/O within the codec starts, the C program and the C5535 chip become irrelevant.

Start this project as if you were repeating Lab 2; you can revert to the `lnkx.cmd` of Lab 2, as the DSPLIB is not used here. There is no ISR feature in this lab.

The source program `aic3204_init()` is actually called twice in the existing `main` from Lab2, first directly to configure the codec and a second time indirectly via the statement `set_sampling_frequency_and_gain` , which configures the codec a second and final time. We are interested in modifying only that second configuration to do both loopback and filtering. The direct calls, in `main`, to `aic3204_init` and `aic3204_hardware_init`, can be commented out.

In addition, in the major `while(1)` loop in `main` you need little. Get rid of all the I/O statements and just blink the diode at about the rate seen in Labs 2 and 3.

The basic reference is the long document “TLV320AIC3204 Application Reference Guide”, which you can get from the TI web site as PDF document slaa557.

The configuring is done by setting 8-bit registers using hexadecimal representations of integers, two such characters to a register. Since filter coefficients can use 24 bits, the Q23 representation is used for them, so each coefficient requires three registers. The filter could be implemented either at the A/D end or the D/A end. We choose the latter. Recall that I have posted the m-file `convert2q.m` on the web site. Though the coefficients are implemented in 24 bits, the signal is still represented in 16 bits.

According to p. 44 of the reference, here is how we can modify the configuring of the codec's DAC Biquad A to realize the transfer function

$$H(z) = 0.5\frac{1 - z^{-1} + z^{-2}}{1 - 1.25z^{-1} + 0.78125z^{-2}}$$

on the left channel and 0.5 on the right (the numerator coefficients are halved because +1 is not representable in Q23). The transfer function has a zero at 1/6th the sampling frequency and a near pole at 1/8th the sampling frequency. The first statement, and last two statements, are already there, and the rest must be inserted:

```
AIC3204_rset( 19, 0x82 );  // Power up MADC and set MADC value to 2
/* set up IIR filter in DAC */
AIC3204_rset( 0x51, 0x00 );  // Power down left and right ADC
AIC3204_rset( 0x3f, 0x14 );  // Power down left, right data paths and set channel
AIC3204_rset(60, 1);  // PRB_R1
AIC3204_rset(0, 44);  // select page 44
AIC3204_rset(12, 0x40); //Left N0 = 1/2
AIC3204_rset(13, 0x00);
AIC3204_rset(14, 0x00);
AIC3204_rset(16, 0xe0); //N1 = -1/4; coefficient halved – to be explained
AIC3204_rset(17, 0x00);
AIC3204_rset(18, 0x00);
AIC3204_rset(20, 0x40); //N2 = 1/2
AIC3204_rset(21, 0x00);
AIC3204_rset(22, 0x00);
AIC3204_rset(24, 0x50); //D1 = 5/8; halved and negated
AIC3204_rset(25, 0x00);
AIC3204_rset(26, 0x00);
AIC3204_rset(28, 0x9c); //D2 = -25/32; negated
AIC3204_rset(29, 0x00);
AIC3204_rset(30, 0x00);

AIC3204_rset(0, 45);  // select page 45
AIC3204_rset(20, 0x40); //Right N0 = 1/2
AIC3204_rset(21, 0x00);
AIC3204_rset(22, 0x00);
AIC3204_rset(24, 0x00); //N1 = 0
AIC3204_rset(25, 0x00);
AIC3204_rset(26, 0x00);
AIC3204_rset(28, 0x00); //N2 = 0
AIC3204_rset(29, 0x00);
AIC3204_rset(30, 0x00);
AIC3204_rset(32, 0x00); //D1 = 0
AIC3204_rset(33, 0x00);
AIC3204_rset(34, 0x00);
AIC3204_rset(36, 0x00); //D2 = 0
AIC3204_rset(37, 0x00);
AIC3204_rset(38, 0x00);
/* end set up IIR filter in DAC */
/* DAC ROUTING and Power Up */
AIC3204_rset( 0, 1 );      // Select page 1
```

In addition, the codec must be configured for loopback, and I will leave you to consult the reference to see how that is done. It involves only the addition of one statement, which should be put in a plausible place, i.e. where the right “page” has been selected. See pp. 101-102 for “Stereo ADC output is routed to Stereo DAC input”. It is a matter of setting an 8-bit register D7D6….D0 to other than its default value with a hexadecimal representation of the new setting.

After you have gotten your example filter running, set yourself up to implement the notch filter determined in the final Matlab assignment on the left channel. Q23 accuracy is more than enough, since the signal values remain accurate only to 16 bits.

I will require a demo in the lab and a printout of your `aic3204_init` file.

I have been setting deadlines for the labs to assure we get through all of them. Since this is the last lab, it is unimportant to me when you get it done provided it is done, and the demo given, by

Friday March 10, the last day of classes. Please email me on the morning of the day you will give it, so I can see that the MG18 lab is set up. However, I prefer you give me the demo on March 10. My plan at this point is to have, instead of a regular class on March 10, a hopefully brief demo session in MG18. Then we can have a special session of homework and other problem discussion the week of Monday March 13, before March 17 (the day of the final exam).

2/15/2017