# Midterm — November 17, 2004

Total: 320 pts

**Write and sign the honor pledge:**

1. (2 pts) Print your name and UMID# on every sheet.

2. (2 pts each)

   (a) The amount of DARAM present on the C5510 is _____ Kwords.
   (b) The amount of SARAM present on the C5510 is _____ Kwords.
   (c) The C5510DSK CPU minimum cycle time is _____ ns.
   (d) The AIC23 codec A/D and D/A word size is _____ bits.
   (e) The C5510 has _____ auxiliary registers.
   (f) The eXtended auxiliary registers each contain _____ bits.
   (g) The C5510 has _____ temporary registers.
   (h) The technical name for the norm that we used to scale for overflow in our IIR filter designs is the "_____ norm".
   (i) The C5510 always fetches _____ bytes from program memory every clock interval.
   (j) C5510 instructions vary in length and use from 1 to _____ bytes.
   (k) There are _____ accumulators in the C5510.
   (l) Each accumulator is made up of _____ registers.
   (m) The total number of bits contained in the registers forming an accumulator is _____.
   (n) There are _____ status registers in the C5510.

3. Denote the sampling rate by $f_s$

   (a) (3 pts) What is the purpose of anti-aliasing filtering?

   (b) (3 pts) How does it work?

   (c) (3 pts) Are anti-aliasing filters digital filters or analog filters?

4. Again denote the sampling rate by $f_s$

   (a) (3 pts) What is the purpose of anti-imaging filtering?

   (b) (3 pts) How does it work?

   (c) (3 pts) Are anti-imaging filters digital filters or analog filters?

5. Consider the group delay of a filter.

   (a) (3 pts) Is group delay a function or a value?

   (b) (3 pts) What is the physical interpretation of group delay?
   (Note: We are not asking for the definition of group delay.)

   (c) (3 pts) Is group delay always negative for a realistic filter? Justify your answer.

6. Lab 6 IIR filtering.

   (a) (3 pts) What Q format did we use in our program to denote the biquad coefficients?

   (b) (3 pts) Why did we divide the coefficients $a_1$ and $b_1$ of biquad sections by 2?

   (c) (3 pts) What is the major problem of TI's iircas5.asm compared to our MyDF2IIR.asm?

7. Consider an int variable a in C.

   (a) (3 pts) write an one-line C code to set bit 0 and bit 2 of a to be 1.

   (b) (3 pts) Write an one-line C code to set bit 1 and bit 3 of a to be 0.

   (c) (4 pts) Write the C code (at most 3 lines) to write the value of bit 4 of a into another int variable b such that b takes only values of 0 or 1.

8. One of the C functions used in our Lab 4:

```
void AIC23_IO(unsigned port, int LeftValue, int RightValue)
{
   McBSP_reg(port, McBSP_DXR2) = LeftValue;           // line 1
   McBSP_reg(port, McBSP_DXR1) = RightValue;          // line 2
   while((McBSP_reg(port, McBSP_SPCR1)&0x0002) == 0); // line 3
   LeftInput = McBSP_reg(2, McBSP_DRR2);              // line 4
   RightInput = McBSP_reg(2, McBSP_DRR1);             // line 5
}
```

(a) (3 pts) What do line 1 and line 2 do?

(b) (3 pts) What does line 3 do?

(c) (3 pts) What do line 4 and line 5 do?

9. Given the zero-pole plot of $H(z)$ as shown in Fig. 1.

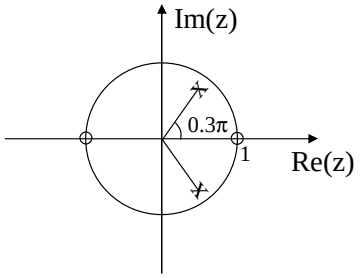
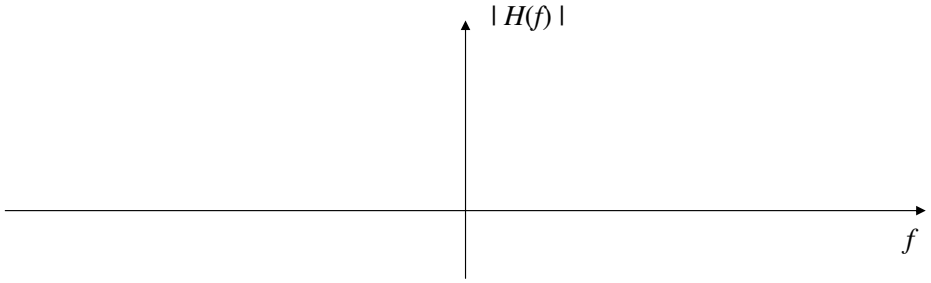



Figure 1: Zero-pole plot

(a) (2 pts) What kind of filter is it? Lowpass, bandpass, or highpass?

(b) (4 pts) Assume the sampling rate is 48 KHz. Roughly sketch $|H(f)|$. Label the frequencies of the sharp dips and peaks.



10. Consider the four IIR filter prototypes used in FDAtool to design a lowpass filter: Butterworth, Chebyshev I, Chebyshev II, and elliptic.

    (a) (2 pts) Which one generates the most smooth magnitude response?

    (b) (2 pts) Which one generates $H(z)$ with the sharpest transition?

    (c) (2 pts) Which one generates $H(z)$ of the highest order?

    (d) (2 pts) Which one generates $H(z)$ of the lowest order?

11. (a) (4 pts) What are the tradeoffs between DF1 and DF2?

    (b) (3 pts) What is the advantage of TDF2 over DF2?

    (c) (3 pts) Is TDF2 by all means the best implementation among DF1, TDF1, DF2, and TDF2? Explain why or why not.

12. Assume that a student is going to write a program for C5510 using large memory model.

    (a) (3 pts) Which RTS library should he/she use?

    (b) (3 pts) Assume the program is perfectly written, all supporting files and the right library files are included, yet lots of error messages occur during the compilation. If you are the 452 GSI, what would you check first?

(c) (3 pts) After you do the step above, there are still some errors though you are absolutely sure that there should not be any. What CCS setting will you check next? And how would you set it?

(d) (3 pts) The student uses FarPeek( ) and FarPoke( ) in the program. Does it make sense to you? Explain why yes or why no.

13. (a) (3 pts) What is the complexity of radix 2 decimation-in-time FFT for $N$ point DFT?

(b) (3 pts) What was the main effort of the FFT algorithms in the old days?

(c) (3 pts) What is the main effort of the FFT algorithms nowadays?

(d) (3 pts) What is wrong to say "please give me the FFT of $x(0), x(1), \cdots x(N-1)$"?

(e) (3 pts) If DFT size $N$ is a prime number, can we have any improvement at all in computing DFT?

14. (a) (3 pts) In Lab 7, why does the real-time FFT display of a sine wave have a serious fluctuation problem?

(b) (3 pts) What is the most effective way to reduce the fluctuation problem?

(c) (3 pts) Among the three windows we used in Lab 7, which one has the smallest sidelobes?

15. (a) (3 pts) Give an application example to which circular addressing is useful.

(b) (3 pts) Does Assembly programming on C5510 have circular addressing support in nature?

(c) (3 pts) Does C programming on C5510 have circular addressing support in nature?

16. (10 pts) Consider the following: CPLD, Timer, AIC23, DPLL, and McBSP.

(a) Which of those belong to the C5510 DSP processor?

(b) Which of those are on the C5510 DSK but not on the C5510 DSP processor?

17. Give 3 reasons why people sample at a higher rate than the Nyquist rate in practice.

(a) (3 pts)

(b) (3 pts)

(c) (3 pts)

18. (4 pts) Explain what does the calibration in Lab 6 do and why it is necessary.

19. In Lab 7, we used `cfft32_NOSCALE( )` and `cbrev32( )` to compute the standard DFT (without $\frac{1}{N}$). Assume that array x=$\{a_0, a_1, a_2, a_3\}$, and DFT$\{a_0, a_1, a_2, a_3\} = \{A_0, A_1, A_2, A_3\}$. We are given the following C code:

```
.
.
cfft32_NOSCALE(x, 4);     // line 1
cbrev32(x, X, 4);         // line 2
cfft32_NOSCALE(X, 4);
cbrev32(X, y, 4);         // line 3
.
.
```

(a) (3 pts) After line 1 is executed, x=?

(b) (3 pts) After line 2 is executed, X=?

(c) (3 pts) After line 3 is executed, y=?

20. Consider the following MyTDF2IIR.asm (read through the whole code first):

```
.
.
rptblocal L_outer-1
mov     *ptr_x+,T0          ; get sample value into T0
rptblocal L_inner-1         ; loop through sections
mov     ?A?,ac0             ; ac0 = w1
macm    *ptr_h+,T0,ac0      ; ac0 = w1+b0*x
mov     ?B?,mmap(T1)        ; output ac0 into T1
mpym    *ptr_h+,T0,ac0      ; ac0 = x*b1/2
```

```
        masm    *ptr_h+,T1,ac0      ; ac0 = x*b1/2-y*a1/2
        sfts    ?C?,?D?             ; ac0 = b1*x-a1*y
        add     *ptr_w-<<#16,ac0    ; ac0 = b1*x-a1*y+w2
        mov     rnd(hi(saturate(ac0))),*ptr_w+ ; update w1
        mpym    *ptr_h+,T0,ac0      ; ac0 = b2*x
        masm    *ptr_h+,T1,ac0      ; ac0 = b2*x-a2*y
        mov     rnd(hi(saturate(ac0))),*ptr_w+ ; update w2
        ||mov   ?E?,T0              ; move y to x
L_inner:
        mov     T1,*ptr_y+ ; filter cascade output value
        mov     txptr_w,xptr_w              ; reset xptr_w
        mov     txptr_h,xptr_h              ; reset xptr_h
L_outer:
        popboth txptr_w
        popboth txptr_h
        .
        .
```

(a) (3 pts) ?A? should be?

(b) (3 pts) ?B? should be?

(c) (3 pts) ?C? should be?

(d) (3 pts) ?D? should be?

(e) (3 pts) ?E? should be?

21. We want to compute a non-polynomial function $f(x)$ on C5510 using the approximation

$$f(x) \approx 1.23x^3 + 2.41x^2 + 8.92x + 2.31$$

(a) (3 pts) What is the reordered form of the polynomial according to Honer's method?

(b) (3 pts) Why use Honer's method?

(c) (3 pts) If we want to write an Assembly program to compute the polynomial and Q15 format is used for $x$. The coefficients are placed into 16-bit words using Qn format. What is the largest n we can use?

22. We want to implement DDS on C5510 using the following C code:

```
int sinetable[128] = { 0, ... }; // actual values go here
unsigned long ac0 = 0;
unsigned long FTV = ?????;  // line 1
unsigned index;
unsigned forever = 1;
        .
        .
while (forever) {
    ac0 = ?????;            // line 2
    index = ?????;          // line 3
    value = *(sinetable+index);
    CodecOut(value);
}
```

(a) (3 pts) If sampling rate $f_s = 24KHz$ and we want to generate a 20KHz sine wave using the DDS, what should FTV be in line 1?

(b) (3 pts) What should line 2 be?

(c) (3 pts) What should line 3 be? Note the size of the sine table.

23. (12 pts) Function fn was written in Assembly and called by C. The prototype of fn is:

void fn(long l1, long l2, long l3, int *p4, int *p5, int *p6, int *p7, int *p8, int i9, int iA);

Specify the content of the following C5510 registers:

(a) AC0 _____

(b) XAR1 _____

(c) T0 _____

(d) T1 _____

24. Consider the following C code for FIR filtering:

```
long y, output;
h_ptr = coeffs; db_ptr = &db[0];    // line 0
while(1) {                          // line 1
    *db_ptr = sample;               // line 2
    y = 0;                          // line 3
    for (i = 0; i < 4; i++) {       // line 4
        y += *h_ptr * *db_ptr;      // line 5
        h_ptr++;                    // line 6
        db_ptr--;                   // line 7
    }
    db_ptr++;                       // line 8
    output = y;                     // line 9
}
```

We are going to use this code to implement an order 3 FIR filter with

$$h(n) = 2^n, n = 0, 1, 2, 3,$$

which are stored in array coeffs[ ].

(a) (3 pts) What should the size of delay buffer db[ ] be for this program?

(b) (3 pts) When we execute the program, it does not work. Why?

(c) (6 pts) We need to add several lines of code to fix the problem. Specify where (i.e. right after which line) and what to add to make the program work.

25. (a) (3 pts) Why do sigma-delta AD converters work better than AD converters with oversampling only?

(b) (3 pts) If we want to have a 8-bit 1st order sigma-delta ADC to work as good as a 20-bit ADC sampling at 48KHz, the sampling rate of the 1-bit ADC should be greater or equal to?

(c) (3 pts) If we want to have a 8-bit 2nd order sigma-delta ADC to work as good as a 20-bit ADC sampling at 48KHz, the sampling rate of the 1-bit ADC should be greater or equal to?

26. Assume that the SNR at (1) in Fig. 2 equals to $K$.
Note that SNR = total signal power / total quantization noise power.

(a) (3 pts) What is the SNR at (2)?

(b) (3 pts) What is the SNR at (3)?

27. Consider the zero-pole plot of a IIR filter as shown in Fig. 3.

(a) (12 pts) If we do the biquad decomposition to the filter as the way we did in Lab 6, we have (continue on page 14)

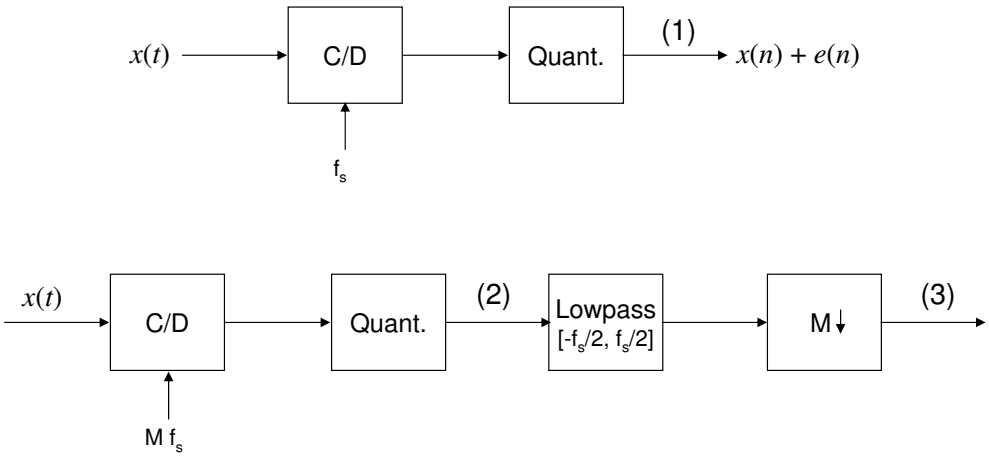



Figure 2: ADC with oversampling

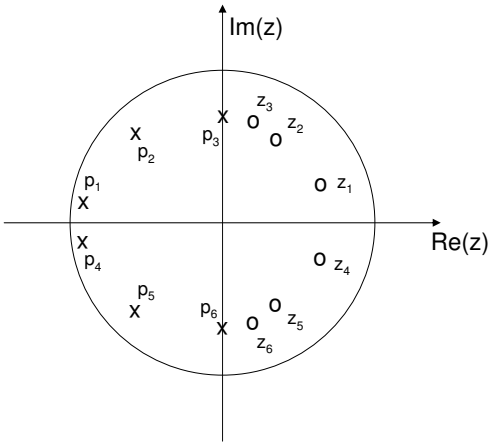



Figure 3: Zero-pole plot

Biquad 1: $\frac{(z-(\quad))(z-(\quad))}{(z-(\quad))(z-(\quad))}$

Biquad 2: $\frac{(z-(\quad))(z-(\quad))}{(z-(\quad))(z-(\quad))}$

Biquad 3: $\frac{(z-(\quad))(z-(\quad))}{(z-(\quad))(z-(\quad))}$

(b) (4 pts) What is the point of doing biquads decomposition?

(c) (4 pts) Should we filter using the biquad section with poles closest to the unit circle first or last? And why?

28. Consider a signal as shown in Fig. 4 which contains sine waves of 3 different frequencies. We feed the signal to a lowpass filter with cutoff frequency 3000Hz. Given the phase response $\theta(f)$ ($f$ in Hz, $\theta$ in radian) has the following derivative values:$\theta'(1000) = -0.06\pi$, $\theta'(2000) = -0.3\pi$, $\theta'(4000) = -0.4\pi$, $\theta'(8000) = -0.8\pi$,

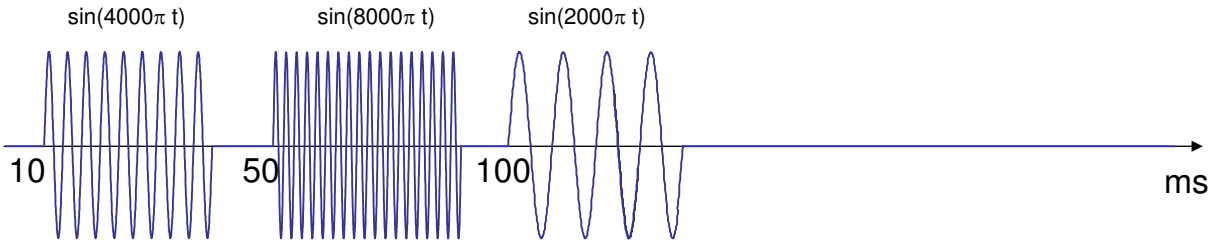



Figure 4: Input wave

(a) (3 pts) What is the unit of $\theta'(f)$?

(b) (9 pts) What are the group delay (in sec.) experienced by $\sin(2000\pi t)$, $\sin(4000\pi t)$, and $\sin(8000\pi t)$?

(c) (3 pts) Is the filter a linear phase filter?

(d) (3 pts) Roughly sketch the filter output. Remember to label each sine wave as well as the starting time.

29. (20 pts) An addressing exercise.

The initial state of each instruction is shown here ...

| | | |
|---|---|---|
| | 0x070203 | 0x21 |
| x = | 0x070204 | 0x22 |
| | 0x070205 | 0x23 |
| | 0x070206 | 0x24 |
| | | |
| | 0x070214 | 0x25 |

```
XAR1   0x070204
XDP    0x070204
T0     1
.dp x
```

Below, write down the state after each instruction . **Only those boxes whose contents have changed are to be filled in!**.

| | AR1 | AC0 | T1 | 0x070204 |
|---|---|---|---|---|
| MOV @(x+1),AC0 | | | | |
| MOV @(x+0x81),AC0 | | | | |
| MOV T0,*AR1+ | | | | |
| MOV *(#x),AC0 | | | | |
| MOV #5,@(x+128) | | | | |
| MOV *(AR1+T0),T1 | | | | |
| MOV *AR1(T0),AC0 | | | | |
| MOV *AR1(#0x10),T1 | | | | |
| MOV @(x+130),AR1 | | | | |
| MOV *+AR1(#1),AC0 | | | | |

30. (4 pts) If you have to bother about the battery life, which version of FIR would you recommend? The asm function from the DSP library or the C function that we wrote? And explain in what way does it save the battery power.