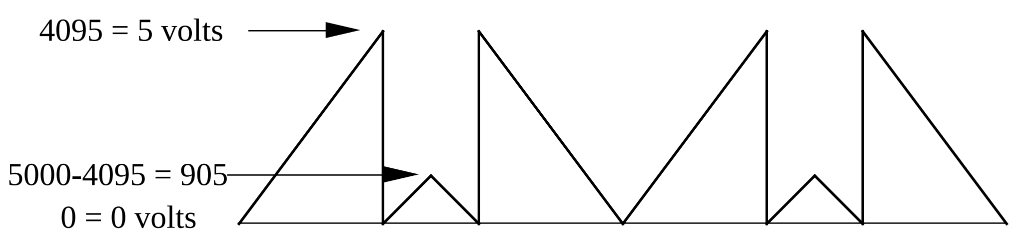
UNIVERSITY OF CALIFORNIA
College of Engineering
Electrical Engineering and Computer Sciences Department

145M Microcomputer Interfacing Lab

Final Exam Solutions May 17, 1996

**Problem 1a**

- The 12-bit D/A converter can only see the lowest 12 bits of the binary number.
- When the rising sequence reaches 4095 (0000 1111 1111 1111), the D/A outputs 5 volts.
- The next number is 4096 (0001 0000 0000 0000 ), so the D/A sees input = 0 and the output drops to 0 volts.
- The next number is 4097 (0001 0000 0000 0001), so the D/A sees input = 1, and the output begins to rise. This continues as shown in the figure below:



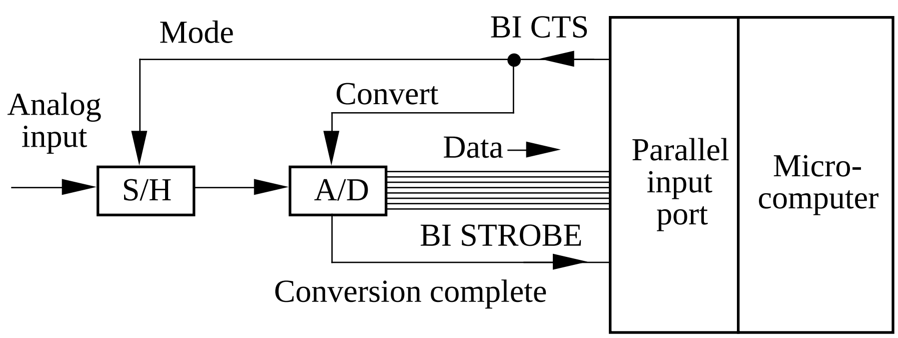

[7 points off if output is shown to rise to 6.1 volts at a D/A input of 5000]
[5 points off if the output is 5 volts or 0 volts when the number is between 4095 and 5000]

**Problem 1b**

The desired triangle wave is most easily produced by sending the sequence 0, 1, 2, . . ., 4094, 4095, 4094, . . ., 2, 1, 0, 1, 2, . . ., 4094, 4095, 4094, . . ., 2, 1, 0, etc. to the D/A.

**PROBLEM 2**

**2a**



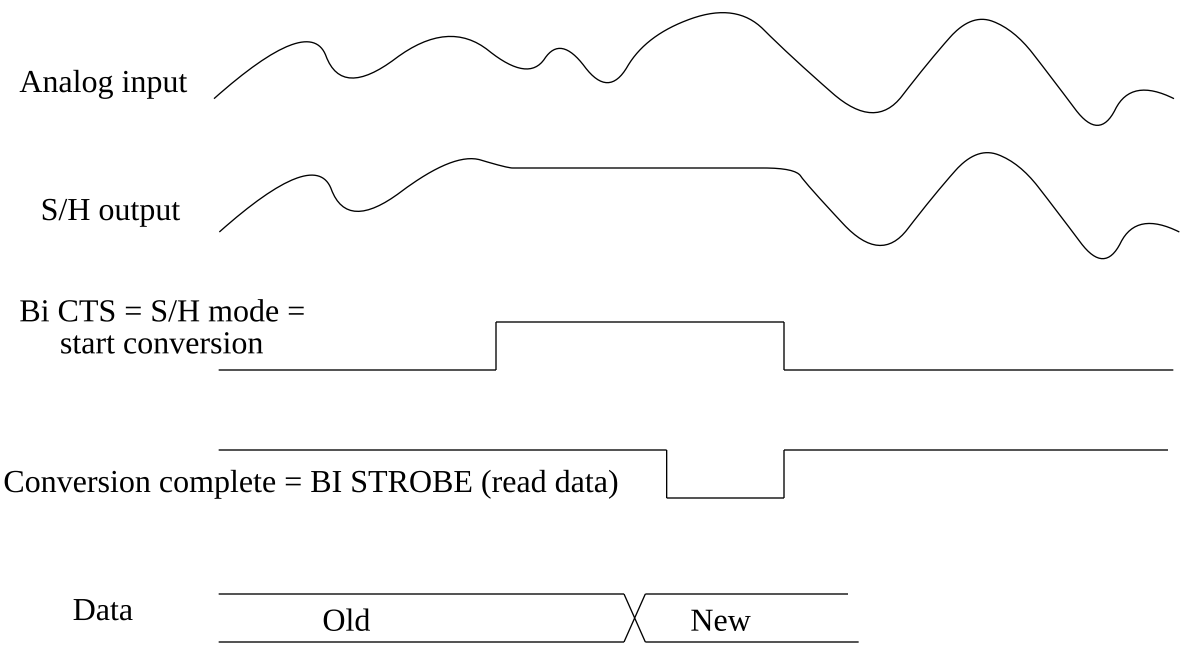

**2b**

BI CTS = start conversion is normally low
BI STROBE = conversion complete is normally high

1 Program makes BI CTS high
2 BI CT high puts S/H into hold mode
3 BI CTS low-high edge begins A/D conversion.
4 When A/D ends conversion, it makes BI STROBE low
5 When program detects BI STROBE low, it reads the input port registers
6 Program makes BI CTS low, and A/D sets BI STROBE high

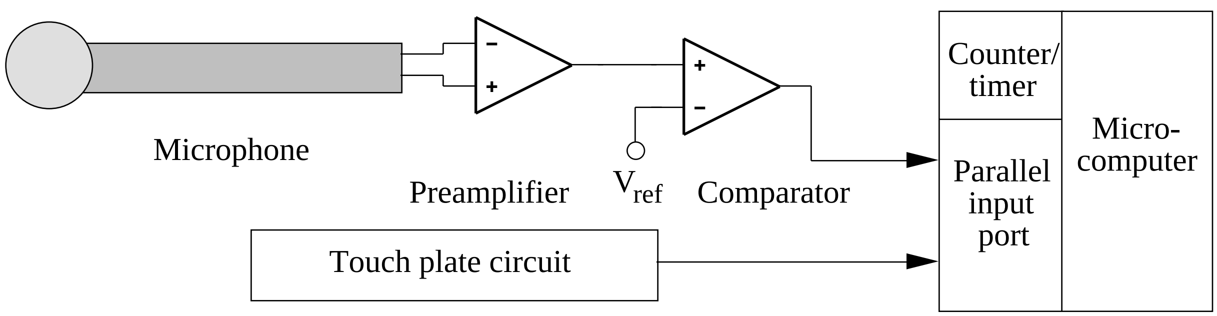
[2 points off for omitting the S/H] [2 points off for omitting step 6]

**Problem 2c**




[3 points off if no analog input and S/H output]  [2 points off if no A/D output data]

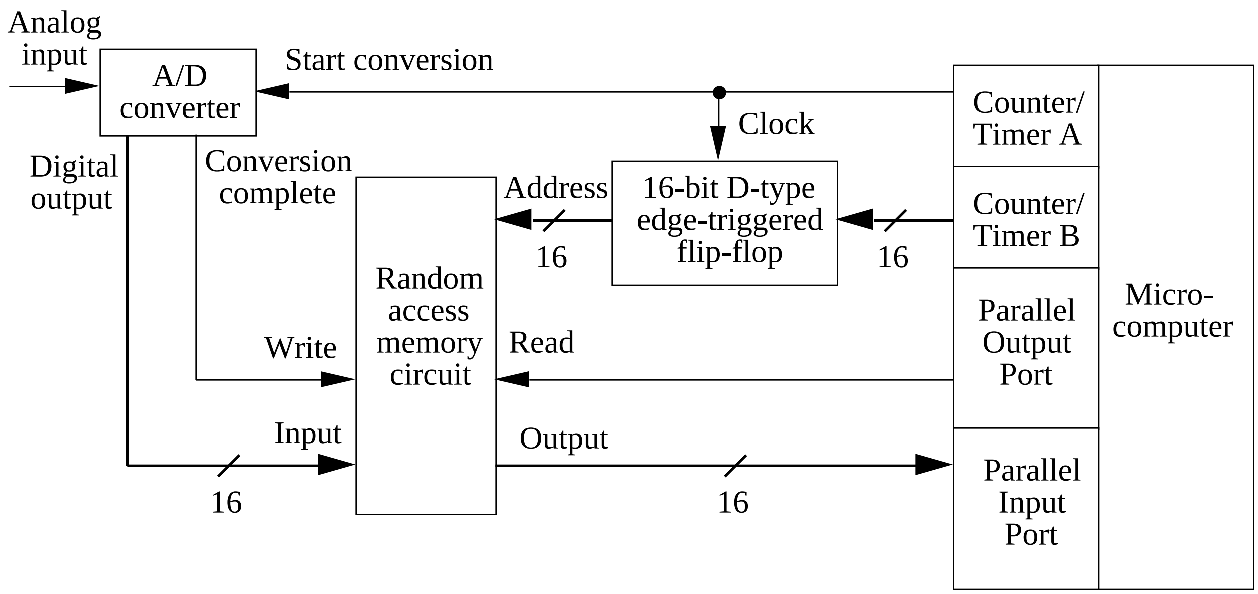
**Problem 3a**




[5 points off for adding analog inputs and reading comparator on one bit- poor design to put different signals on the same bit when there are 15 other bits available.]
[5 points off for connecting preamp output (unknown analog voltage) to parallel input port (needs 5 volts, which the comparator was intended to produce)]

**Problem 3b**
Loop until start comparator goes high
Zero counter/timer
loop until touch plate circuit goes high
Read counter/timer
take difference, divide by 1 million, write to screen

**Problem 4a**




[4 points off if flip-flops not used between counter B out and memory circuit address in, since counter B out is only valid on leading edge of pulse A out]

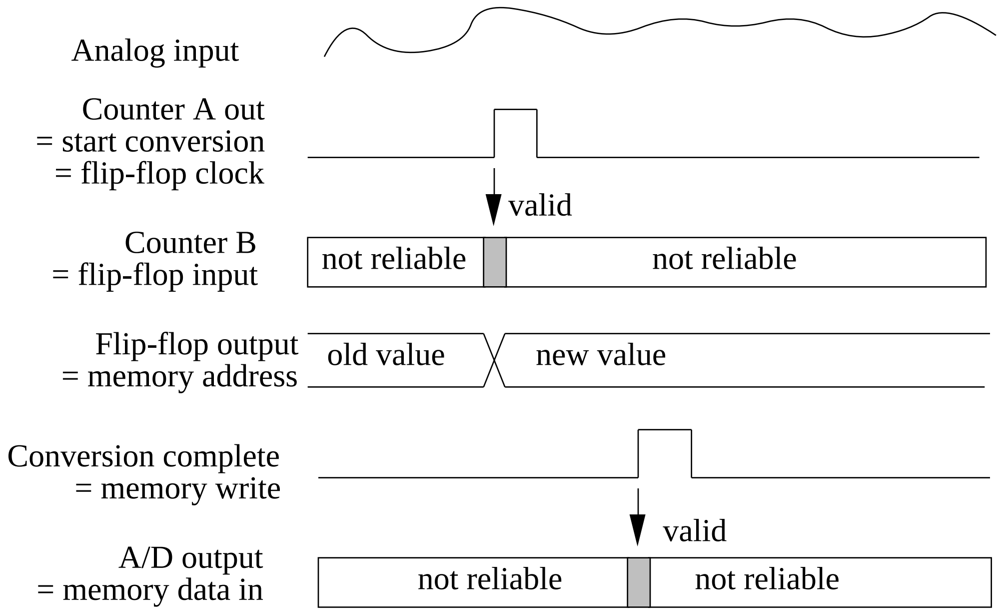
Note: since 0.1 µs (sampling period) < 1 µs (input port read time), the computer can not be involved in acquiring individual samples or handshaking- the counter/timer and the external circuit must do all the heavy work.

**Problem 4b**

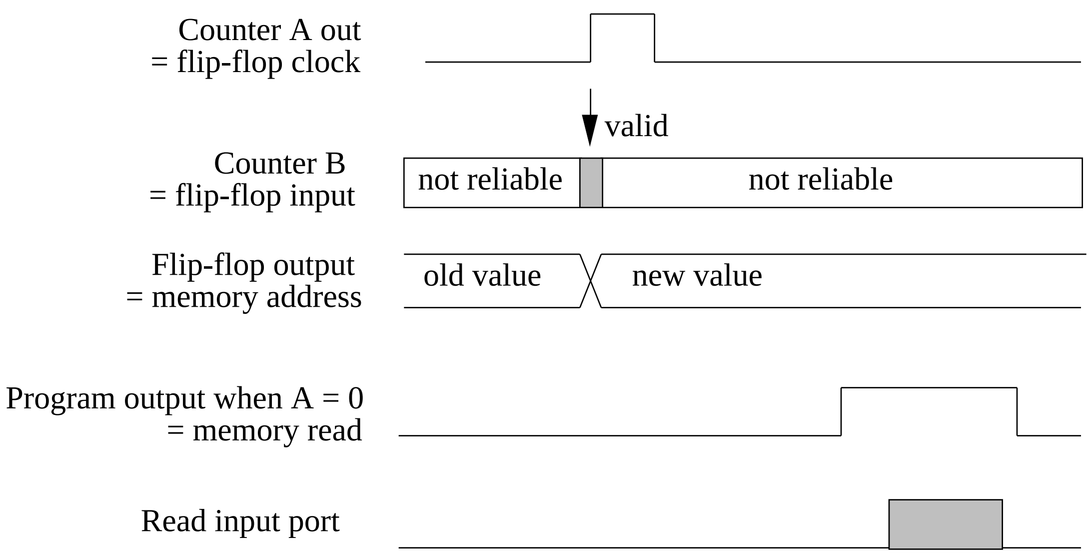
1 Program asks user for number of samples N, sampling period T (in µs), and a filename
2 Program loads N into counter B and sets for downcounting, and to stop when reaching zero.
3 Program loads 10T into counter A and sets for downcounting, reload 10T when reaching zero
4 Program starts counters to begin analog data sampling
5 When counter A reaches zero, it decrements counter B, produces an external pulse, reloads its initial value, and begins to count down
6 External pulse A latches the value of counter B on the flip-flops and starts A/D conversion
7 When A/D finishes conversion, “conversion complete” = “write” latches the A/D output number into the memory location specified by the address lines
8 Steps 5-7 are repeated until counter B reaches zero and stops
9 Program loads N into counter B and sets for downcounting, and to stop when reaching zero.
10 Program loads M into counter A and sets for downcounting, reload M when reaching zero. M is chosen to provide enough time to detect when counter B has changed (1 µs), generate a memory circuit read pulse (1 µs) and to read the parallel input port (1 µs). M = 60 would provide a safe 6 µs.
11 When counter A reaches zero, it decrements counter B, produces an external pulse, reloads its initial value, and begins to count down
12 External pulse A latches the value of counter B on the flip-flops
13 Program reads counter B in a loop until it changes
14 Program produces a memory circuit read pulse via the parallel output port and reads the parallel input port into computer memory
15 Steps 11-15 are repeated until counter B is zero
16 Program writes the data to the specified file

**Problem 4c**

For data acquisition, the timing diagram is:

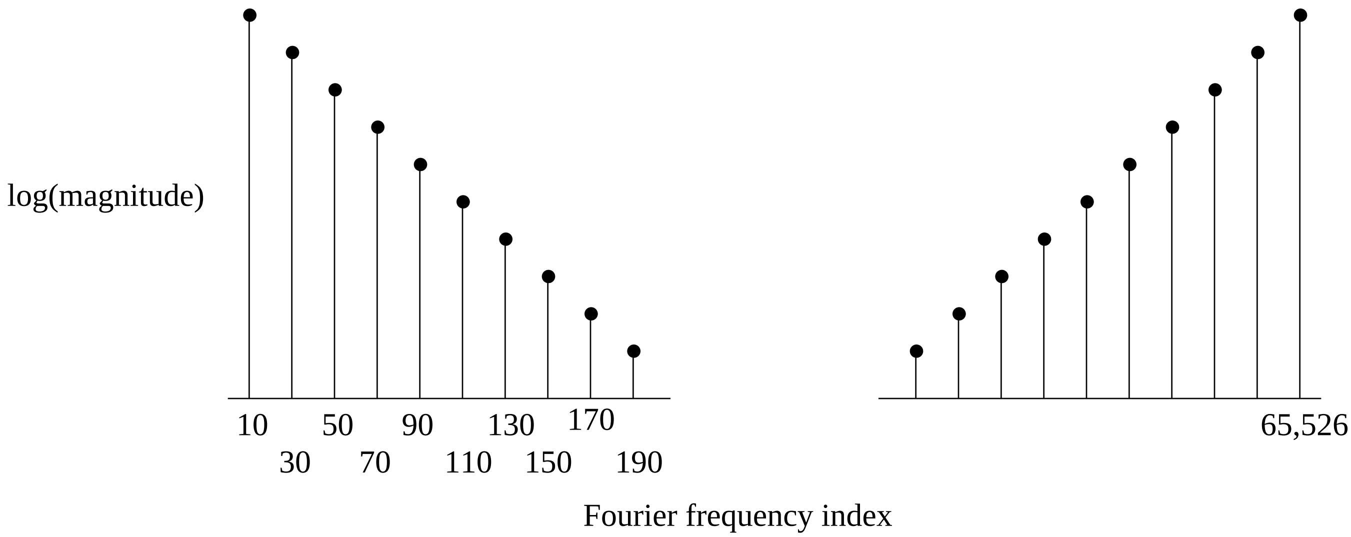



For program readback of memory circuit:




**Problem 5a**

- Since exactly 10 cycles are sampled, the first harmonic at 10 Hz appears at Fourier coefficient $H_{10}$ and at $H_{65,536-10}$.
- Since the square wave was symmetric, only odd harmonics will appear. The mth harmonic will have frequency 10m Hz and appear at Fourier coefficient $H_{10m}$, and at $H_{65,536-10m}$.
- The magnitudes of the harmonics will fall as 1/m (–6 dB per octave goes as 1/f).




[5 points off for not displaying discrete harmonics]

**Problem 5b**

Let a(t) be the desired 10 Hz square wave and b(t) be the output of the amplifier with a square wave input. First let us compute the impulse response of the amplifier, then use the impulse response to determine the desired input (as in laboratory exercise 24).

**Step 1: determine the impulse response**

These are related to c(t), the unknown impulse response of the amplifier by:

b(t) = a(t) convolved with b(t)

This means that the corresponding Fourier transforms have $B_n = A_n C_n$, a simple multiplication.

The impulse response can then be computed as the inverse FFT of $B_n / A_n$.

**Step 2: determine the input that produces a square wave**

We know that a(t) = d(t) convolved with c(t), where a(t) is the desired square wave output, c(t) is the impulse response we determined in step 1, and d(t) is the unknown input waveform that produces the square wave a(t).

d(t) = inverse FFT of $A_n/C_n$ = inverse FFT of $A_n (A_n/B_n)$.

In the case where we want to produce a 10 H square wave, we know exactly the Fourier coefficients that are needed in the above equation (i.e. those corresponding to 10 Hz, 30 Hz, etc.)

**Problem 5c**

We only have the response of the amplifier to a 10 Hz square wave input, which provides information on its response only at 10 Hz, 30 Hz, 50 Hz, etc. However, to generate a 20 Hz output square wave, we need to know $B_n$ at 20 Hz, 60 Hz, 100 Hz, etc. One solution is to interpolate the real and imaginary Fourier components between the known frequency values.

**Problem 5d**

- Method is unreliable at 5 Hz because amplifier response is not measured at 5 Hz when using 10 Hz square wave input. Needed response at 15 Hz, 25 Hz, etc. would have to be interpolated.
- Method is not reliable at 10 kHz because amplifier response is down by a factor of 1000 from its low frequency response. A/D resolution and noise limit accuracy.

**145M Final Exam:**

| Problem | 1 | 2 | 3 | 4 | 5 | Total |
|---|---|---|---|---|---|---|
| Average | | | | | | |
| rms | | | | | | |
| Maximum | | | | | | |

**145M Numerical Grades:**

| | Lab total | Lab Partic. | Midterm #1 | Midterm #2 | Final | Total |
|---|---|---|---|---|---|---|
| Average | | 100 | | | | |
| rms | | 0 | | | | |
| Maximum | 500 | 100 | 100 | 100 | 200 | 1000 |

The average scores of the two lab report graders differed by 1.4 points. 5.6 points were added......

**145M Letter Grade Distribution (passing students only)**

| Letter Grade | Course Totals (1000 max) |
|---|---|
| A+ | |
| A | |
| A– | |
| B+ | |
| B | |
| B– | none |
| C+ | |
| C | none |
| C– | none |
| D+ | none |
| D | none |
| D– | |

* graduate students