University of California, Los Angeles
Henry Samueli School of Engineering and Applied Science
Department of Electrical Engineering

ECE 216B | D. Marković

The exam is closed-book. You are allowed one page of notes.

ECE 216B: SPRING 2024—MIDTERM

Tuesday, May 7, 10am-11:50am

MSOL: times/dates vary

NAME	SOLUTION
	Last First

SID	

Please write answers in the box provided, answers elsewhere will not be graded.

If you need extra space, draw a box around your "final" answer for each question.

You have 110 minutes for the exam plus 20 minutes for upload logistics (if remote, 110 total if in-class).

Read the questions carefully.
If you get stuck, move on.

Good luck!

Problem 1   _____/6
Problem 2   _____/16
Problem 3   _____/18
Problem 4   _____/12

Total (52)	

PROBLEM 1: Dark Silicon (6 pts)

A 7nm processor chip operates at 0.8V. Under power-limited scaling, maximum chip utilization in the 7nm technology is 20%. The processor is scaled to a 5nm technology and it operates at 0.75V, under the same power budget as the 7nm counterpart. Additionally, scaling allows new features to be added, while keeping the chip area the same. In other words, assume that the chip area and the total power budget are the same across the two technologies. Estimate maximum chip utilization in the 5nm technology.

Power is limited and general scaling model applies:

Geometry scaling: $S = 7/5 = 1.4$

Voltage scaling: $U = 0.8/0.75 = 1.067$

For a fixed chip area (scaling is economic trend and more quantity is expected),

Chip utilization drops by: $S^2/U^2 = 1.72$

Chip utilization in 5nm = $20\% * U^2/S^2 = 11.63\%$

Max chip utilization at 5nm (%) = 11.63%

PROBLEM 2: Multiplier (16 pts)

An array multiplier consists of rows or adders, each producing partial sums that are subsequently fed to the next adder row. We consider the effect of pipelining such a multiplier by inserting registers.
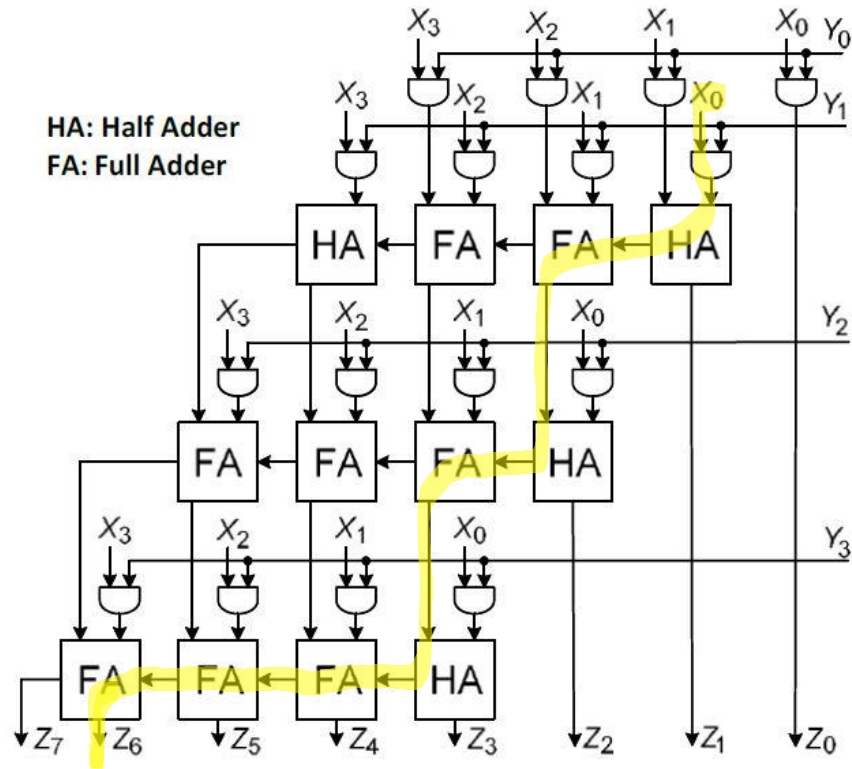
a) Indicate critical path of the non-pipelined multiplier on the diagram above. Redraw the standard array multiplier (on the next page) by inserting word-level pipeline registers as to maximize throughput for a 4x4 multiplier. Assume that X_{0-3} and Y_{0-3} come from registers. (**8 pts**)

critical path is marked on the schematic above

HA: Half Adder
FA: Full Adder
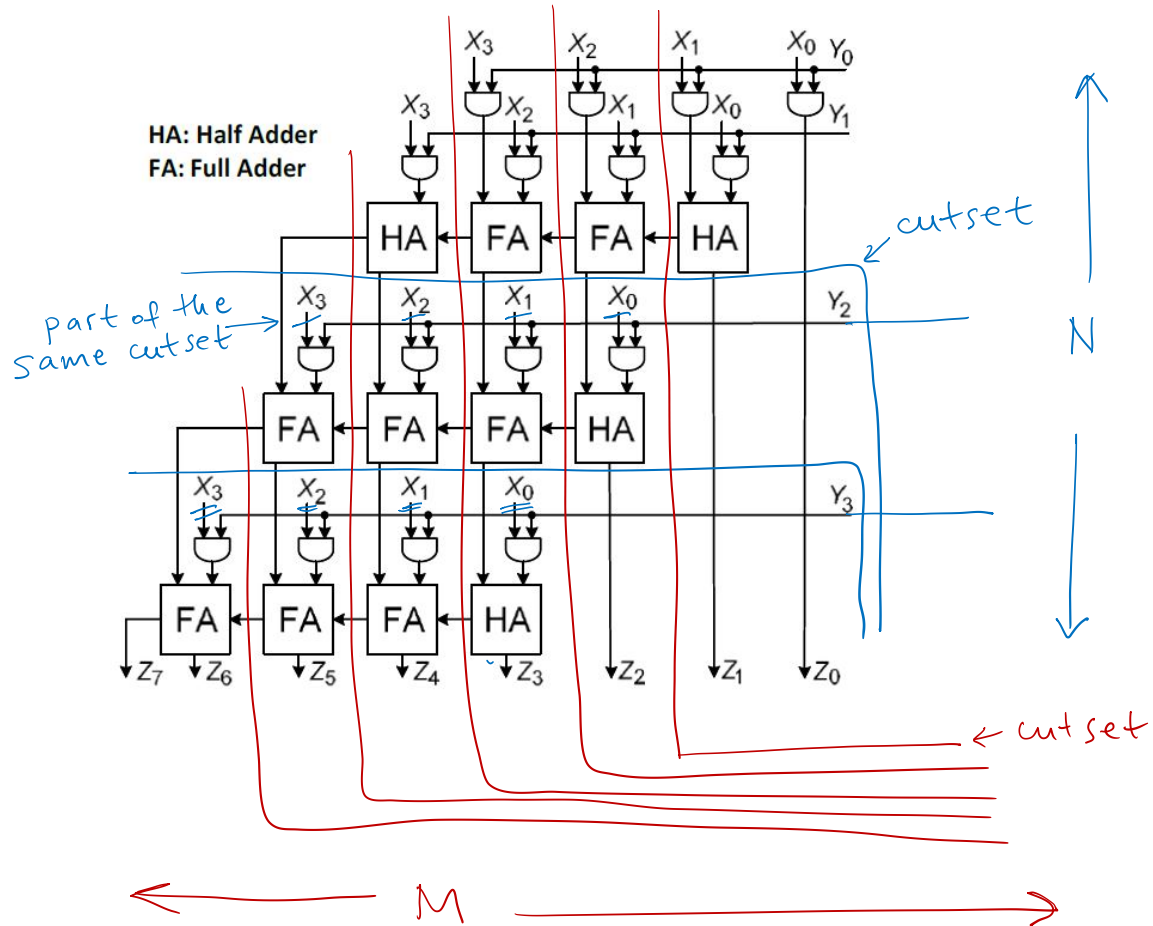
b) What is the critical path and latency for the non-pipelined and pipelined implementation? (**8 pts**)

Non-pipelined: $t_{crit\text{-}path} = [(M-1)+(N-2)]\, t_{carry} + (N-1)t_{sum} + t_{and}$

   $M = N = 4$ Simplified ($t_{carry} = t_{sum} = t_{add}$): $t_{crit\text{-}path} = 8\, t_{add} + t_{and}$

Pipelined (see table and schematic above): Horizontal (H), Vertical (V), Horizontal + Vertical (H + V)

Design	Non-pipelined	Pipelined
Critical path	$5\, t_{carry} + 3\, t_{sum} + t_{and}$ (Simplified: $8\, t_{add} + t_{and}$)	H: $3\, t_{carry} + t_{sum} + t_{and}$ (Simplified: $4\, t_{add} + t_{and}$) V: $3\, t_{sum} + t_{and}$ (Simplified: $3\, t_{add} + t_{and}$) H+V: $\max\{t_{carry}, t_{sum}\} + t_{and}$ (Simplified: $t_{add} + t_{and}$)
Latency	0	H: 2 V: 5 H+V: 7

PROBLEM 3: Number Representation (18 pts)

Add v_1, v_2, and v_3 in two steps: 1) $v = v_1 + v_2$, 2) $s = v + v_3$. Assume $v_1 = 0.6875$, $v_2 = 0.8125$, and $v_3 = -0.5625$. Use **5-bit** 2's complement representation of v_1, v_2, v_3 and final sum s (intermediate result v could have more bits, if needed). You may need to sign-extend v_3 in Step 2, if v dictates so.

a) **Step 1**   (**3 pts**)

v_1 in 2's complement → **0.1011** ($=2^{-1} + 2^{-3} + 2^{-4}$)

v_2 in 2's complement → **0.1101** ($=2^{-1} + 2^{-2} + 2^{-4}$)

$v = v_1 + v_2$ → **01.1000** (sign since C_{in}(MSB) $=1 \neq C_{out}$(MSB) $= 0$

b) **Step 2**   (**3 pts**)

v_3 in 2's complement → **11.0111** (magnitude $= 2^{-1} + 2^{-4}$ → 2's comp. = 1.0111, sign extend)

Copy v from **Step 1** here → **01.1000**

$s = v + v_3$ → **100.1111** (discard this sign bit; C_{in}(MSB) = C_{out}(MSB) = 1, so no overflow, <u>fits 5-bit format</u>)

c) Express the following 2's complement numbers in decimal (base-10) notation. (**2 pts**)

0110.1101

6.8125
($= 2^2 + 2^1 + 2^{-1} + 2^{-2} + 2^{-4}$)

1011.0011

−4.8125 (sign bit = 1, magnitude = 4.8125)

d) Truncate decimal format in (c) to base-10 integers, and binary format to base-2 integers. (**4 pts**)

Decimal →

6	−4

Binary →

0110	1011
$=(6)_{10}$	$=(-5)_{10}$

e) Round decimal format in (c) to base-10 integers, and binary format to base-2 integers. (**4 pts**)

Decimal →

7	−5

Binary →

0111	1011
$=(7)_{10}$	$=(-5)_{10}$

f) Apply sign extension to the binary format in (c) in base-2.   (**2 pts**)

Answer box →   **0**0110.1101   **1**1011.0011
(**extended sign** shown in red)

PROBLEM 4: Filter Architectures (12 pts)

Consider a 4-tap FIR filter shown below.
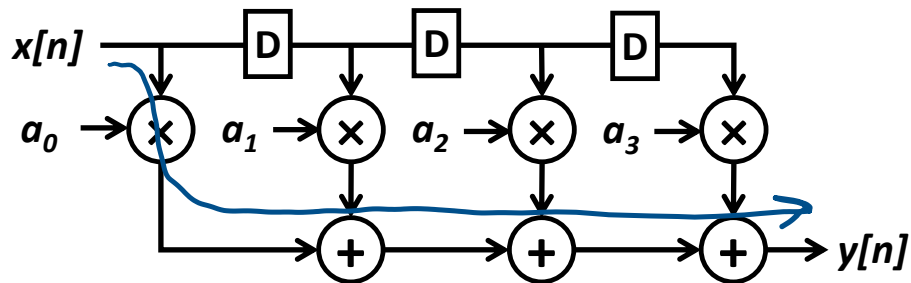
a) (**3 pts**) What is the maximum throughput ($1/t_\text{critical-path}$) of this architecture? Assume $t_\text{adder} = 200\text{ps}$, $t_\text{multiplier} = 800\text{ps}$, and registered x[n] and y[n].

Critical path = $t_\text{multiplier} + 3t_\text{adder}$ = 1.4ns

Throughput ≤ 714.3 MS/s

> **Max Throughput = 714.3 MS/s**

b) (**9 pts**) Suggest an architecture that maximizes throughput. You can use any architectural technique that you see fit (direct, transposed, pipelined), but you can add no more than 4 pipeline registers in total. Slicing through combinational logic inside the adders and multipliers is allowed (i.e. low-level pipelining). What is the maximum achievable throughput? Sketch the modified architecture.

Baseline transposed architecture has 3 registers
➔ Three pipeline registers are added to balance logic delay
➔ Low-level pipelining within multipliers is applied to match t_1 and t_2 logic delays
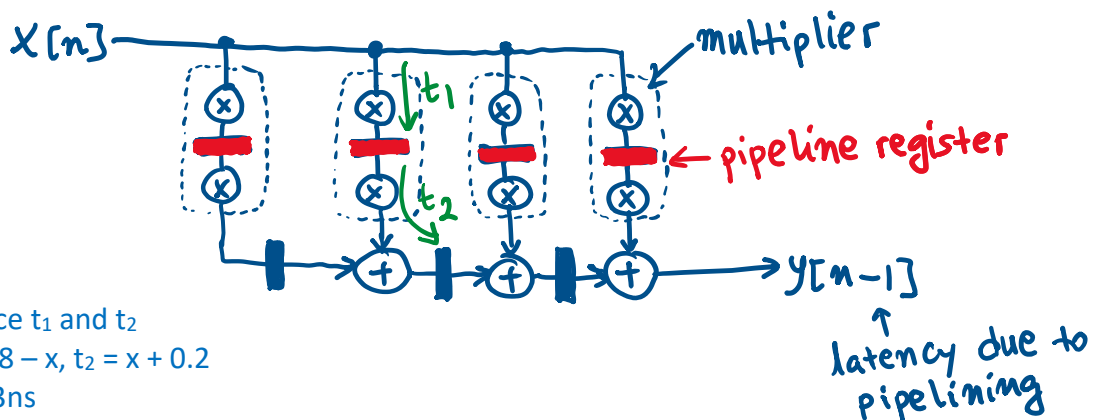
Balance t_1 and t_2
$t_1 = 0.8 - x$, $t_2 = x + 0.2$
$x = 0.3$ns
$t_1 = t_2 = 0.5$ns (a 2.8x speed up)

> **Max Throughput = 2 GS/s**