MichiganEngineering

EECS 427
Discussion 5

Brian Cline

Tuesday, October 14, 2008

Administrative

- Quiz 1 TOMORROW!!
  - Date & time: Wednesday, October 15, 6:30-8:00pm
  - Location: EECS 1303 (same place as lecture)
- CAD5 due a week from Friday; October 24, 2008
  - Shifter

Quiz 1

- Quiz 1 will cover all lecture material through last Thursday (process variation)
  - CMOS Fabrication flow
  - Layout (design rules, antenna rules, latch-up, etc.)
  - Process Variation (RET techniques, variation sources, variation sample spaces, etc.)
  - CMOS (inverter operation, delay calculation, sizing, etc.)
  - Adders
  - Shifters
  - Logical Effort
- Bring your favorite writing utensil, a calculator, notes, and books

CAD5 – the Shifter

- Due Friday, October 24, 2008 at 7pm
- Shifter also important to modern processors
  - Aligns or scales data
  - Used to manipulate bits or bytes
  - Could be used as the root of a multiplier – shift+add unit
- Use the shifter to implement baseline logical shift
  - LSH and LSHI in the ISA
  - Logical shift shifts in zeros to MSB or LSB (depending on shift directions)

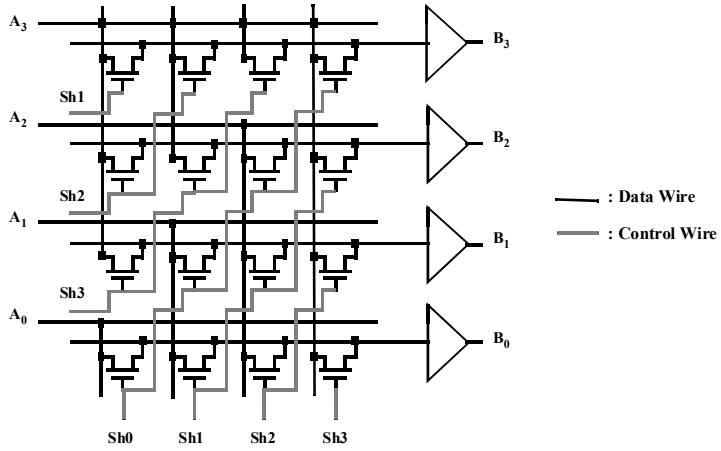
Shifter

- Again, only use up to metal3 (metals 4, 5, and 6 saved for global routing and power grids)
- Bit-slice pitch/width should match up with RF and ALU
- DRC should be clean, including grid checks at top level (ignore grid label violations at lower levels of hierarchy)
- Start with speedpath (significant routing capacitance – estimate capacitance using tables from lecture)
- Finally, run Analog simulations with parasitics

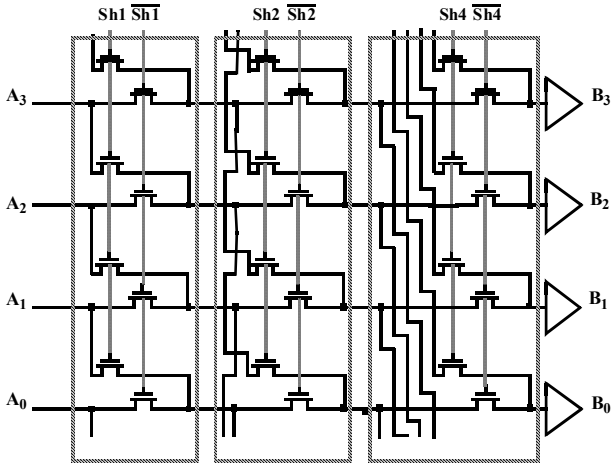
Requirements

- Schematic design
  - Will want to start with a speedpath
- HSpice waveforms showing critical path delays
- NCVerilog simulations showing successful shifter operation for a couple left shifts, right shifts, and “no” shifts
- Clean DRC report (excluding lower level gridchecks)
- Clean LVS report
- README

Barrel Shifter



Logarithmic Shifter



Shifter Comparison

N	K	Barrel		Logarithmic	
		Width	Speed	Width	Speed
		2 N p_m	1 + N diffs	$p_m(2^K+2K-1)$	K + 2 diffs
8	3	16 p_m	1 + 8	13 p_m	3 + 2
16	4	32 p_m	1 + 16	23 p_m	4 + 2
32	5	64 p_m	1 + 32	41 p_m	5 + 2
64	6	128 p_m	1 + 64	75 p_m	6 + 2

Barrel shifter needs an K x 2^K shift amount decoder

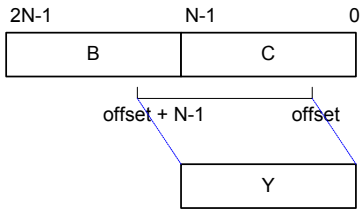
Table 1: Optimal designs for each architecture and gate type

arch and gate type	order	buffers	delay (ns)	avg power (mw)	power * delay
log pass transistor	high	1	9	2.56	23.0
log transmission	high	1	9	2.79	25.1
log static	low	-	7	5.85	40.9
log dynamic	high	-	11	6.65	73.2
array pass transistor	-	-	11	3.55	39.0
array transmission	-	-	11	4.37	48.1
array static	-	-	10	4.80	48.0

[Acken, ISLPED'96]

Funnel Shifter

- A funnel shifter can do all six types of shifts
- Selects N-bit field Y from 2N-bit input
  - Shift by k bits ($0 \leq k < N$)



Funnel Shifter Operation

Table 10.10 Funnel shifter operation

Shift Type	*B*	*C*	Offset
Logical Right	0…0	$A_{N-1}…A_0$	k
Logical Left	$A_{N-1}…A_0$	0…0	$N-k$
Arithmetic Right	$A_{N-1}…A_{N-1}$ (sign extension)	$A_{N-1}…A_0$	k
Arithmetic Left	$A_{N-1}…A_0$	0	$N-k$
Rotate Right	$A_{N-1}…A_0$	$A_{N-1}…A_0$	k
Rotate Left	$A_{N-1}…A_0$	$A_{N-1}…A_0$	$N-k$

- Computing N-k requires an adder

Simplified Funnel Shifter

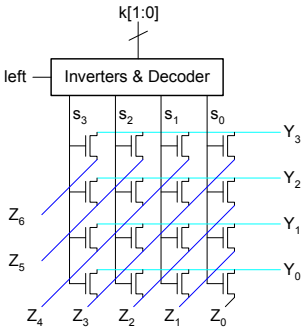
- Optimize down to 2N-1 bit input

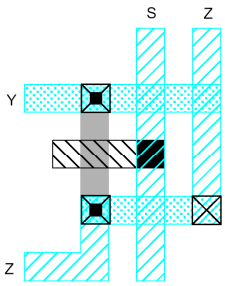
Table 10.11 Simplified funnel shifter

Shift Type	*Z*	Offset
Logical Right	0..0, $A_{N-1}…A_0$	k
Logical Left	$A_{N-1}…A_0$, 0..0	$\overline{k}$
Arithmetic Right	$A_{N-1}…A_{N-1}, A_{N-1}…A_0$	k
Arithmetic Left	$A_{N-1}…A_0$, 0..0	$\overline{k}$
Rotate Right	$A_{N-2}…A_0, A_{N-1}…A_0$	k
Rotate Left	$A_{N-1}…A_0, A_{N-1}.A_1$	$\overline{k}$

Funnel Shifter Design 1

- N N-input multiplexers
  - Use 1-of-N hot select signals for shift amount
  - *n*MOS pass transistor design (V_t drops!)





Funnel Shifter Design 2

- Log N stages of 2-input muxes
  - No select decoding needed