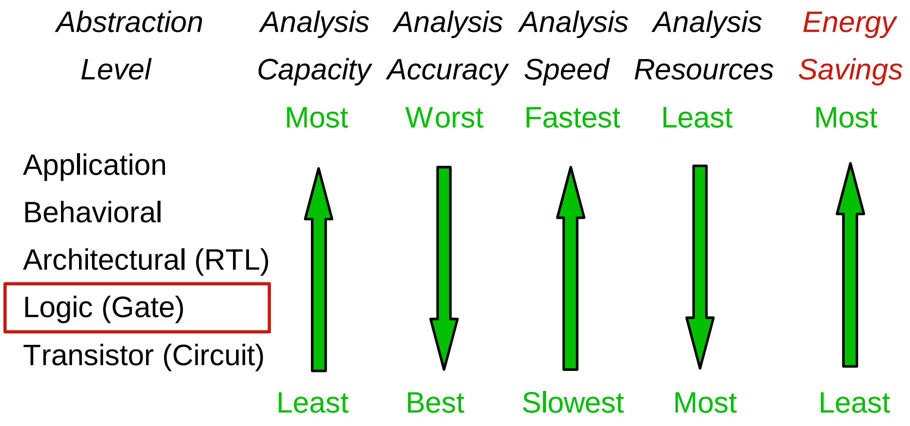
Tutorial Outline

8:30 - 8:45	Introduction and motivation
8:45 - 9:05	Sources of power in CMOS designs
9:05 - 9:30	Power analysis tools and techniques
9:30 - 10:30	Gate & functional unit design issues & techniques
10:30 - 10:50	BREAK
10:50 - 12:15	Architectural level issues and techniques
12:15 - 1:30	LUNCH
1:30 - 2:30	Low power memory system design
2:30 - 3:30	Software level issues and techniques
3:30 - 3:50	BREAK
3:50 - 4:30	Software level issues and techniques, con't
4:30 - 4:45	Future challenges

Design Levels



Abstraction Level	*Analysis Capacity*	*Analysis Accuracy*	*Analysis Speed*	*Analysis Resources*	*Energy Savings*
	Most	Worst	Fastest	Least	Most
Application	↑	↓	↑	↓	↑
Behavioral					
Architectural (RTL)					
Logic (Gate)					
Transistor (Circuit)					
	Least	Best	Slowest	Most	Least

Basic Principles of Low Power Design

$$P = C_L V_{DD}^2 f_{0\rightarrow 1} + t_{sc} V_{DD} I_{peak} f_{0\rightarrow 1} + V_{DD} I_{leakage}$$

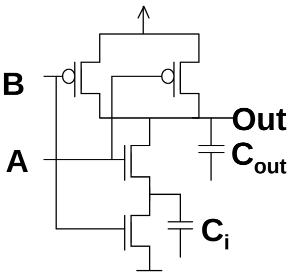
- Reduce switching (supply) voltage
  - » quadratic effect -> dramatic savings
  - » negative effect on performance
- Reduce capacitance
- Reduce switching frequency
  - » switching activity
  - » clock rate
- Reduce glitching
- Reduce short circuit currents (slope engineering)
- Reduce leakage currents

Low Energy Gates: Transistor Sizing

- Use the smallest transistors that satisfy the delay constraints
  - » slack time - difference between required time and arrival time of a signal at a gate output
    - – Positive slack - size down
    - – Negative slack - size up
- Make gates that toggle more frequently smaller
- Size for slope engineering to reduce short circuit currents

Low Energy Gates: Transistor Pin Ordering

- Logically equivalent inputs may not have identical energy/delay characteristics

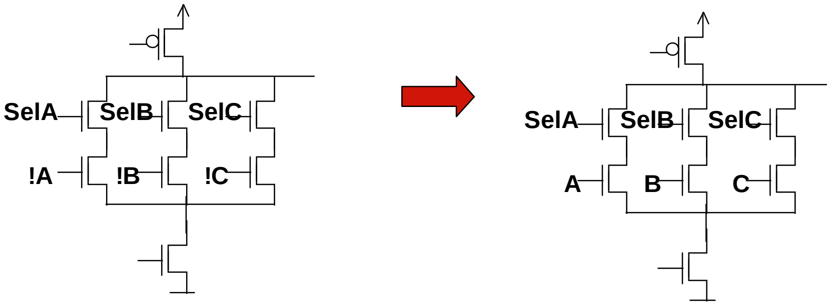


- To conserve energy (and improve speed), connect inputs so that most active input is nearest output

- Need to know signal statistics

Low Energy Gates: Dynamic Gate Pin Ordering

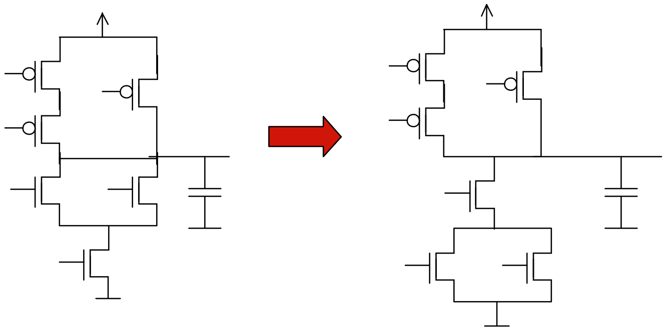
- Dynamic gates exhibit higher switching activity (and add to clock load) but are fast



If A, B, and C have low signal probability

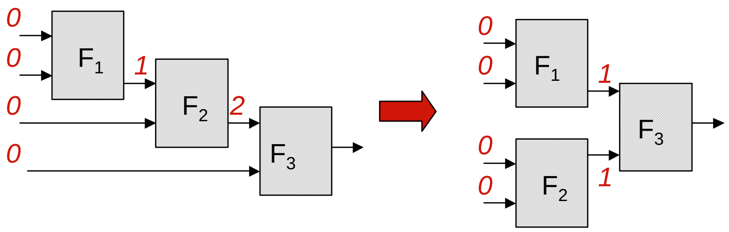
Low Energy Gates:
Gate Restructuring

- Logically equivalent CMOS gates may not have identical energy/delay characteristics



Low Energy Gate Networks:
Balanced Delay Paths

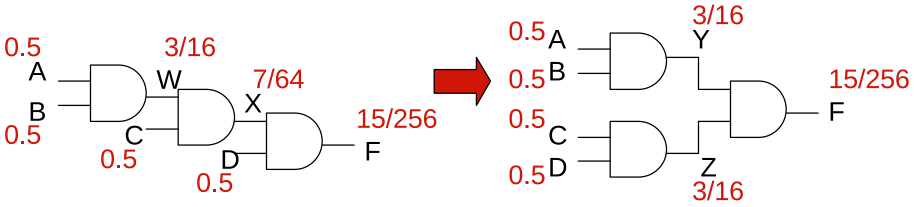
- Reduce glitching by balancing the delay path



Equalize lengths of timing paths through logic

Low Energy Gate Networks: Network Restructuring

- Consider logic topology alternatives



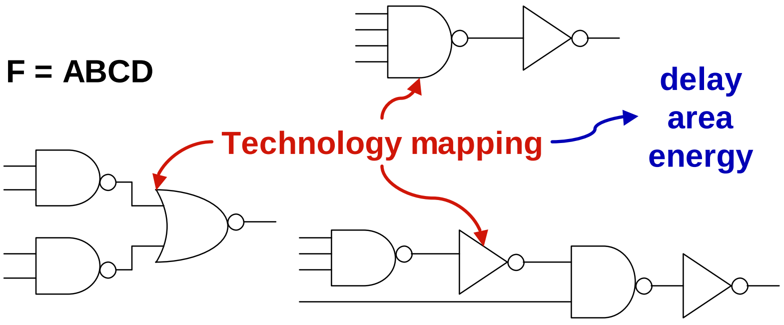
Chain implementation has a lower overall switching activity than the tree implementation

Ignores glitching effects

Network Restructuring, con't

- Logically equivalent gate networks may not have identical energy/delay characteristics

F = ABCD

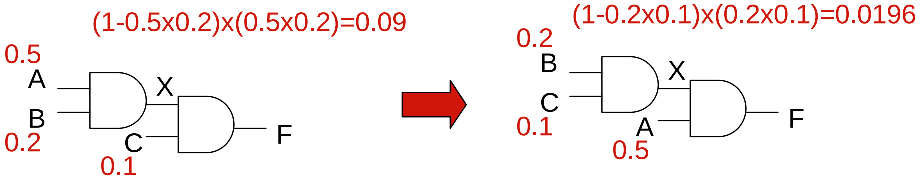


Technology mapping

delay
area
energy

Low Energy Gate Networks: Network Input Ordering

- Input ordering



Beneficial to postpone the introduction of signals with a high transition rate (signals with signal probability close to 0.5)

Dual Supply Voltages

- Use two V_{DD}'s (e.g., 2.5V and 1.5V)
  - » use the higher supply for gates on the critical path
  - » use the lower supply for gates off the critical path
- Reduces energy without a performance loss
- Cons
  - » slight area penalty
  - » increased design time
  - » need level converters to interconnect gates on different supplies (to avoid static currents)

Dual Threshold Voltages

- Use two V_T's (e.g., 0.6V and 0.3V for V_{DD} = 2.5V)
  - use the lower threshold for gates on the critical path
  - use the higher threshold for gates off the critical path
- Improves performance without an increase in power
- Cons
  - increased fabrication complexity
  - increased design time
  - beware of increased leakage in low V_T portion of the circuit - could end up with increased power!

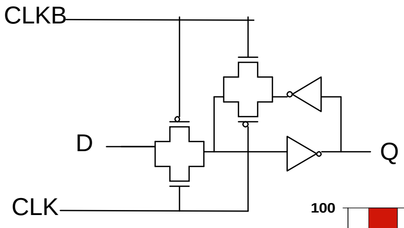
Functional Unit Energy Optimization

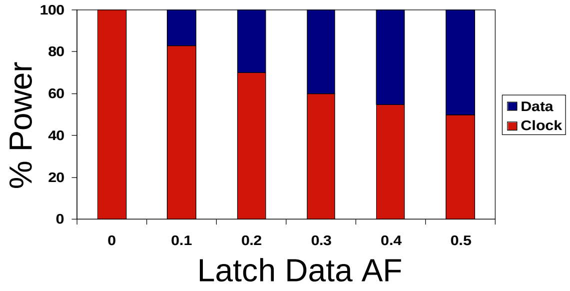
- Key processor core functional units
  - latches and (pipeline) registers
  - ALUs - adders, multipliers, barrel shifters
  - control logic (FSMs)
  - interconnect
  - multi-ported register file
- On-chip memories (ROMs, caches, SRAMs, eDRAMs)
- MMU, TLB
- Clock generation and distribution
- Off-chip interconnect (pads)

Flipflops and Pipeline Registers

- Consume a lot of energy because they are clocked every cycle
  - » Clock energy (E_c)
    - – energy dissipated when the ff is clocked with stable data
  - » Data energy (E_d)
    - – energy dissipated when the ff is clocked and the data has changed so that the ff changes state
  - » Typically the data rate (f_d) is much lower than the clock rate (f_c)
- Also impacts clock energy since a large portion of clock energy is used to drive the sequential elements

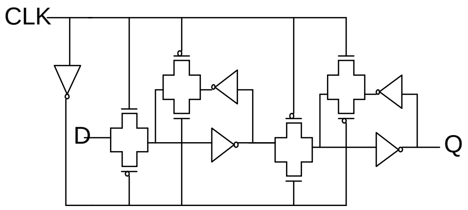
Power Consumption in Latches



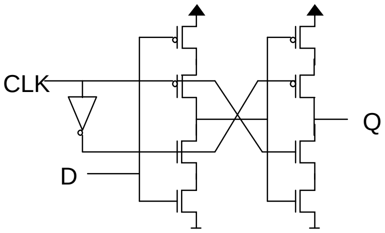


From Tiwari, 1998

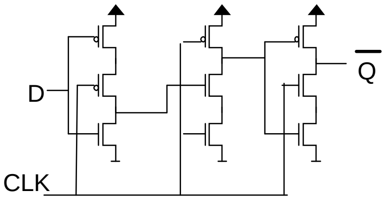
Some Typical CMOS FFs



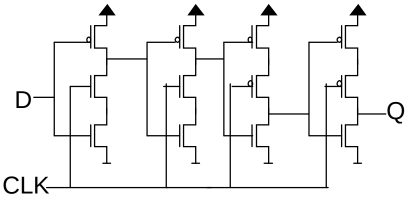
Static TG FF



Dynamic C2MOS FF

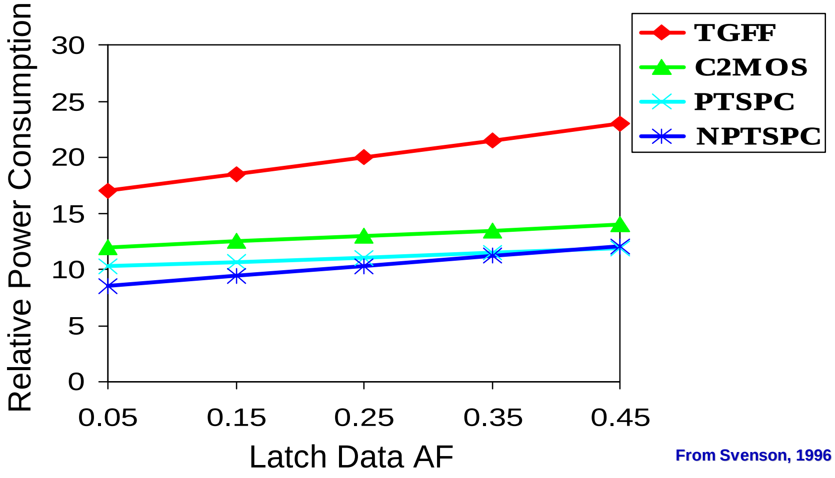


Dyn Precharged TSPC FF



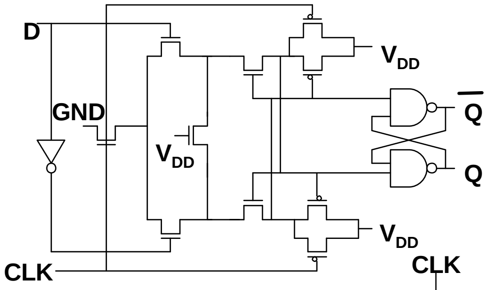
Dyn Non-Precharged TSPC FF

FF Power Comparison



From Svenson, 1996

Energy Efficient Flipflops

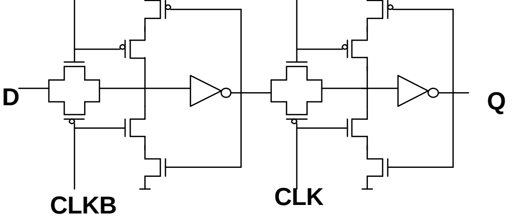


StrongArm SA110 FF

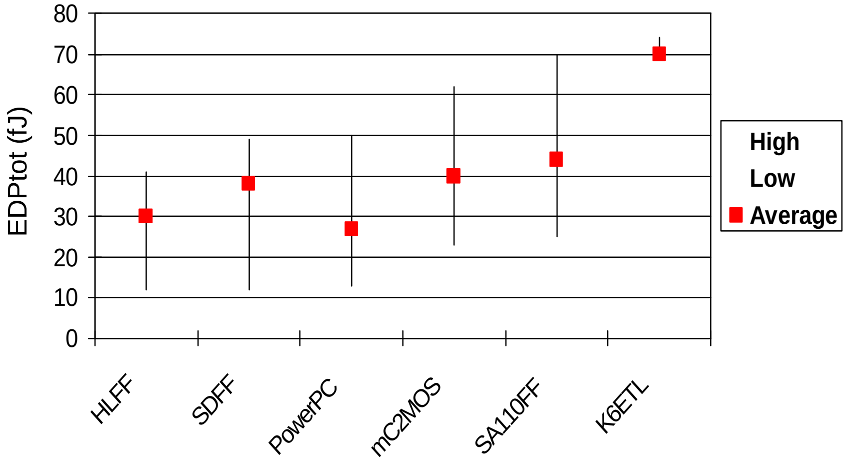
20 transistor
CLK
3 clock loads

16 transistor
CLK & CLKB
4 clock loads each

Power PC 603 FF



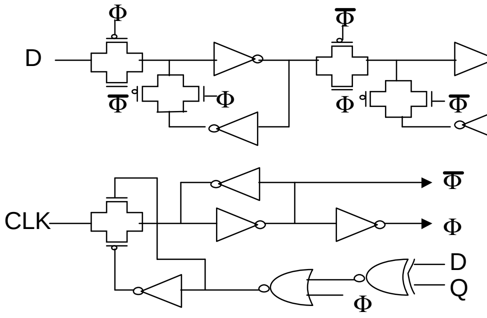
EDP of Some Low Power FFs



From Stojanovic, 1998

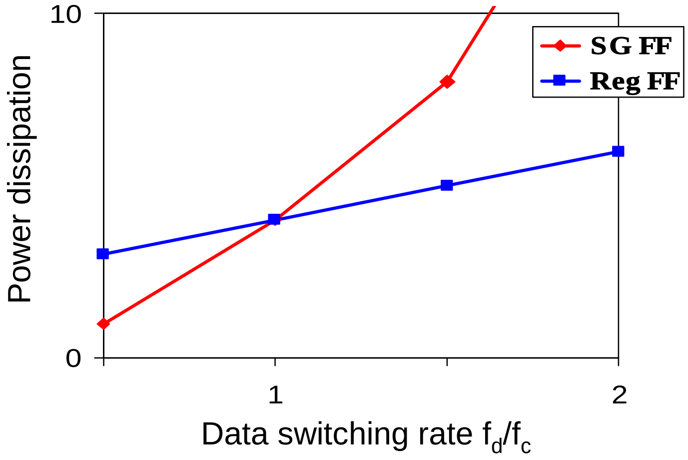
Self-Gating FF

- When ff input is equal to its output, suppress internal clocking to conserve energy
  - » gating function is derived within the FF



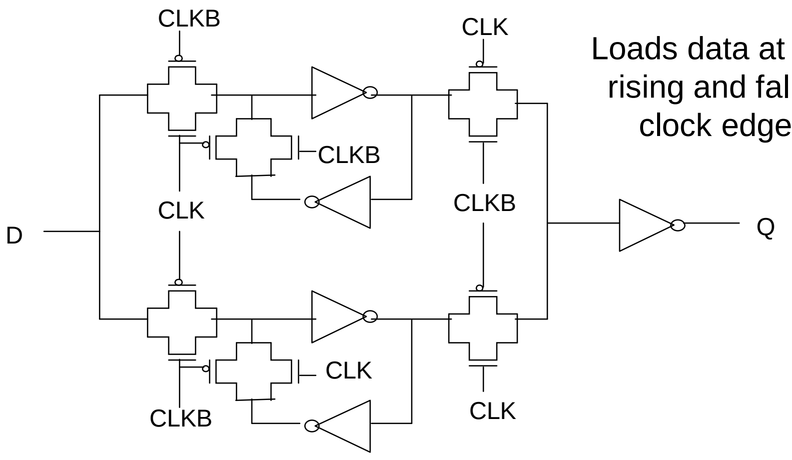
Strict rules
on when D can
change wrt CLK

Power of Self-Gated FF



From Reyes, 1996

Double Edge Triggered FF

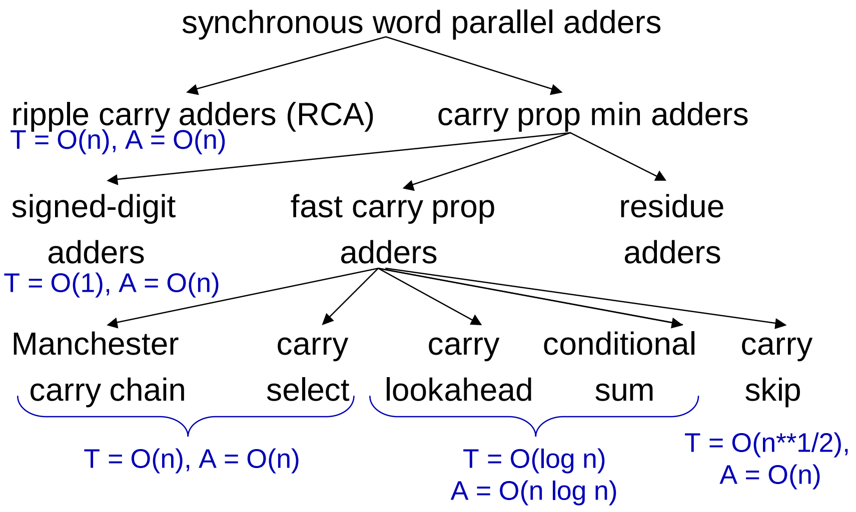


Loads data at both
rising and falling
clock edges

DETFF Pros and Cons

- Advantages
  - » Clock frequency can be halved to achieve the same computational throughput: $P_d = 0.84P_s$
  - » Also get a 2X energy savings in the clock network
- Disadvantages
  - » About 15% larger in transistor count
  - » Maximum operating frequency less
  - » Strict requirements on clock skew
  - » Requires a strict 50% duty cycle
  - » Larger clock load

Adders (Subtractors)



synchronous word parallel adders

ripple carry adders (RCA)
T = O(n), A = O(n)

carry prop min adders

signed-digit adders
T = O(1), A = O(n)

fast carry prop adders

residue adders

Manchester carry chain

carry select

carry lookahead

conditional sum

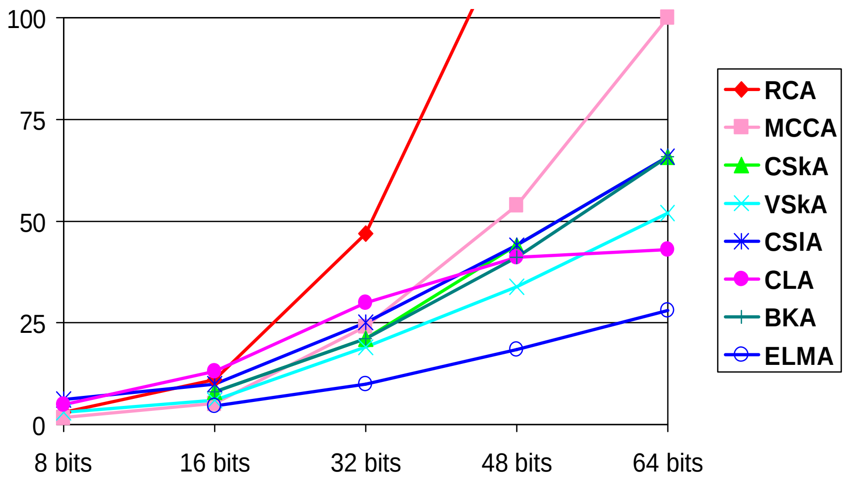
carry skip

T = O(n), A = O(n)

T = O(log n)
A = O(n log n)

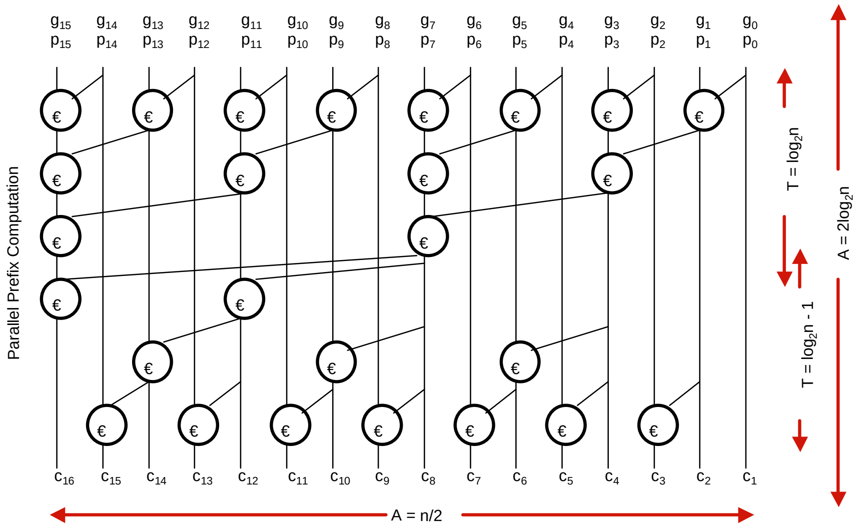
T = O(n**1/2),
A = O(n)

PDP of Different Adders

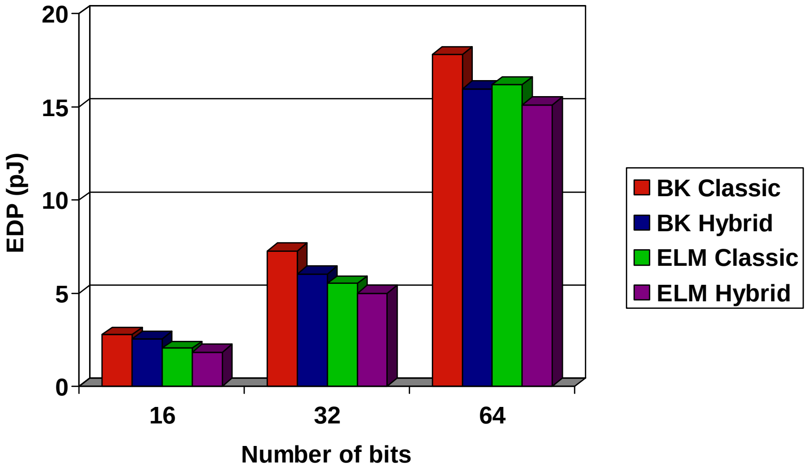


From Nagendra, 1996

Brent-Kung (CLA) Adder



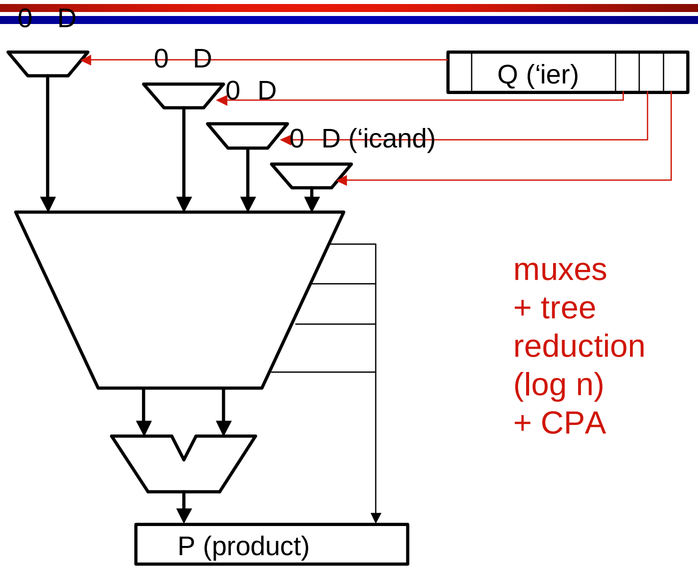
BK and ELM Adder Optimization



Parallel Multipliers

- Form partial product array in parallel and add it in parallel
  - » can use multiplier recoding to reduce the high of the partial produce array by half
  - » recoding may cost more energy than it saves!
  - » use delay balancing to reduce glitching
- Array multipliers (regularity)
- Pipelined multipliers (higher throughput, longer latency, less glitching but adds to clock load)

Parallel Multiplier Structure



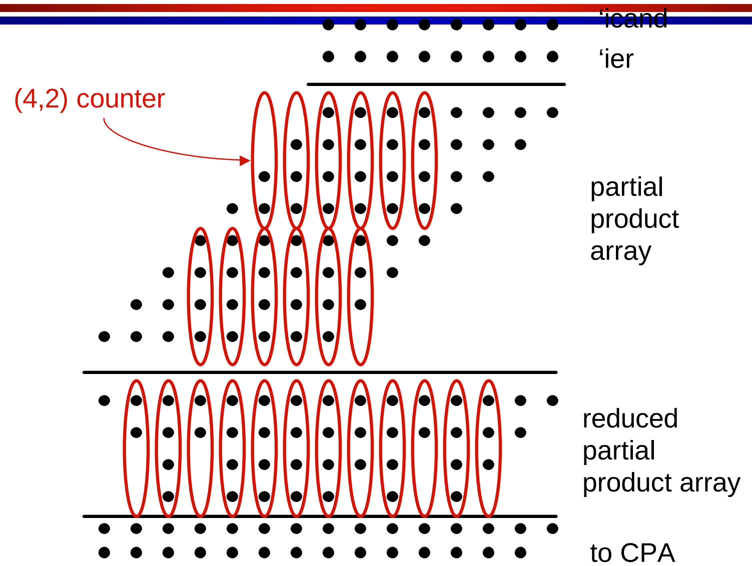
multiple forming circuits

partial product array reduction tree

fast CPA

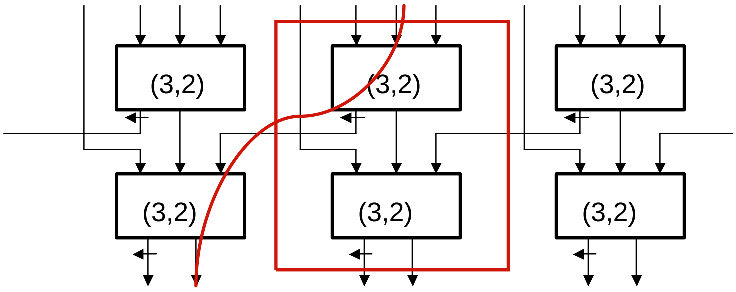
muxes
+ tree
reduction
(log n)
+ CPA

PP Array Reduction Process



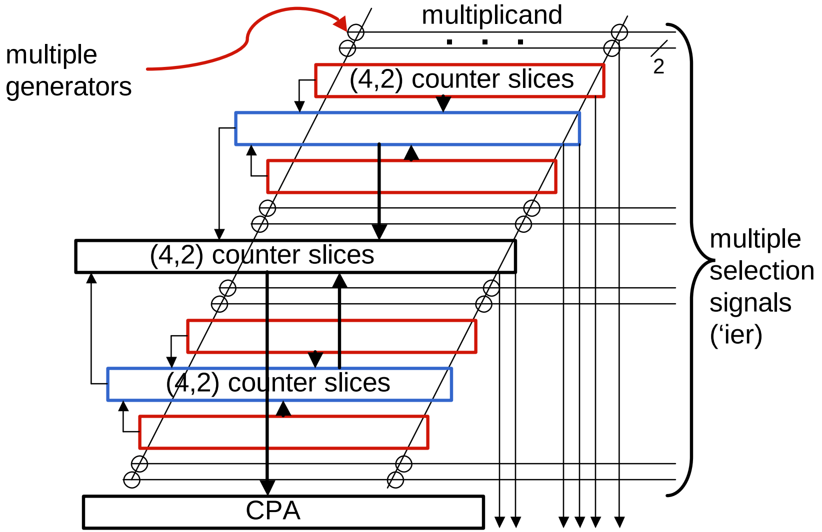
(4,2) Counters

- Built out of (3,2) counters (FA’s)



- Tiles with neighboring (4,2) counters
- Can use delay balancing in cell design and interconnect to reduce glitching

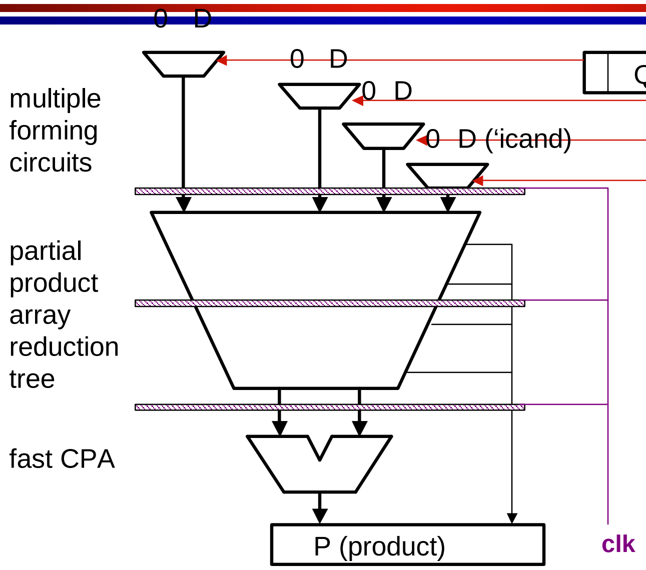
PP Array Reduction Tree Structure



Glitch Reduction by Pipelining

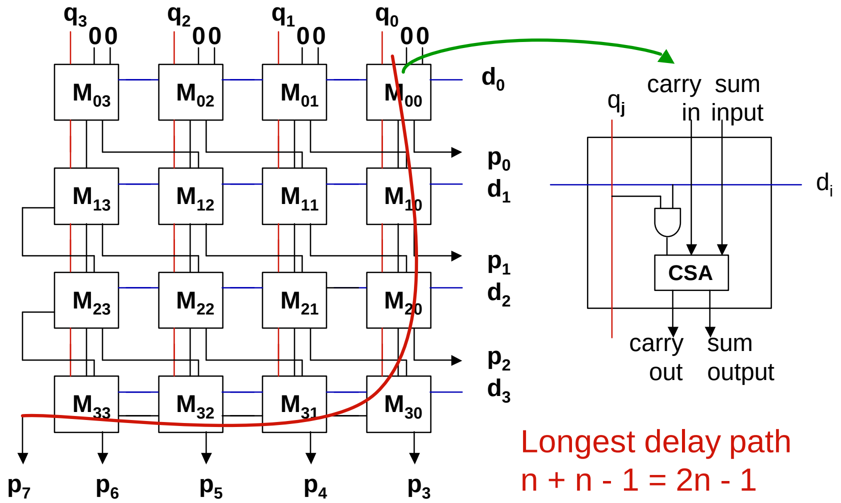
- Glitches are dependent on the logic depth of the circuit
- Nodes logically deeper are more prone to glitching
  - » arrival times of the gate inputs are more spread due to delay imbalances
  - » usually affected by more primary input switching
- Reduce depth by adding pipeline registers

Pipelined Parallel Multiplier



helps to reduce glitching but adds to the clock load

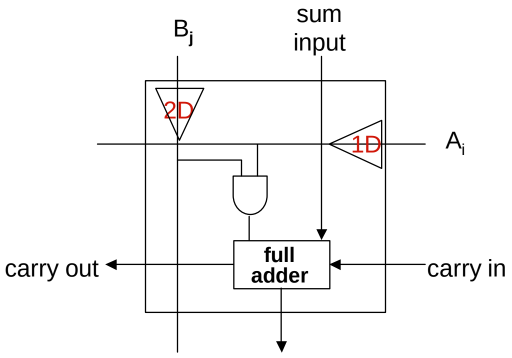
CSA Array Multiplier



Longest delay path

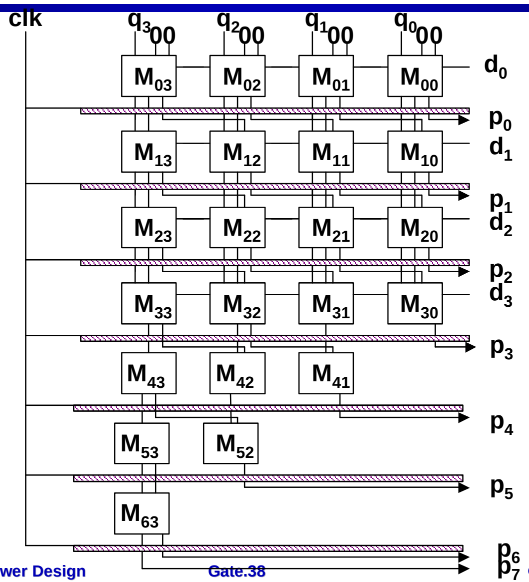
n + n - 1 = 2n - 1

Multiplier Cell Structure

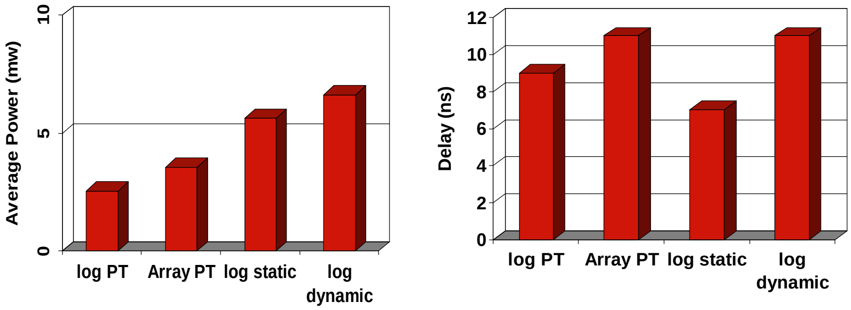


add delay elements to minimize glitching

Pipelined CSA Array Multiplier



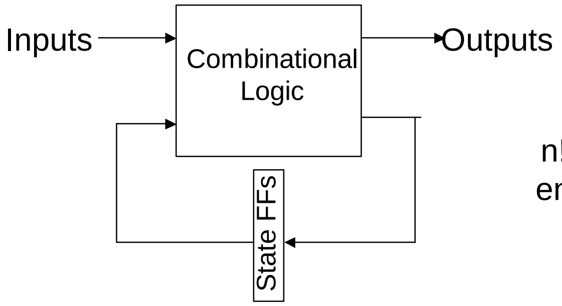
Barrel Shifters



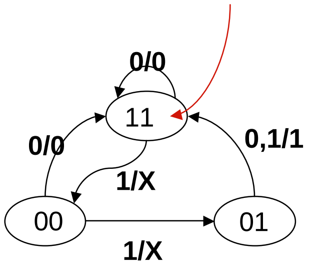
Influence of architecture: Logarithmic, Array
and Gate types: Pass Transistor, Dynamic/Static Mux

From Acken, 1996

Control Unit Design



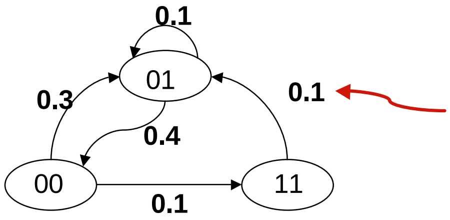
n! different possible
encodings (n states)



State Encoding
One of most important factors determining area, speed, and energy of resulting control logic

Energy State Encoding Heuristic

- Area driven -> try to reduce the distance in Boolean n-space between related states
- Energy driven -> try to minimize number of bit transitions in the state register
  - » fewer transitions in state register
  - » fewer transitions propagated to combinational logic

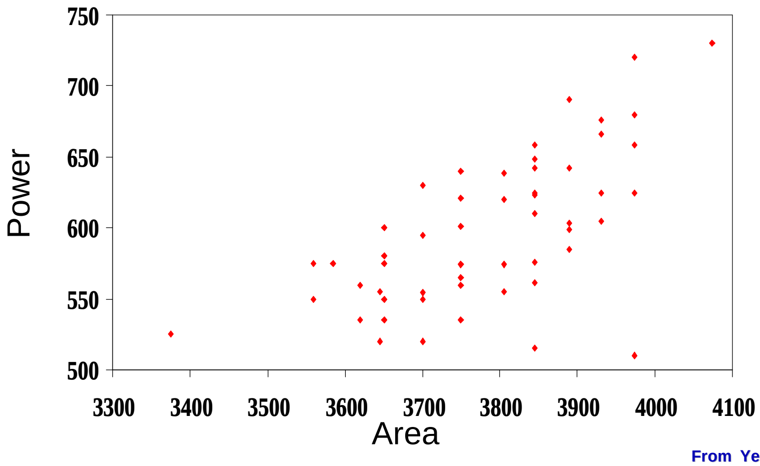


probability that a transition will occur (sum of all edges equals unity)

Caveat

- Lowest E[M] may not be lowest in energy → it could require more gates and/or signal transitions in the combinational logic
- Experiments show that the area and energy dissipation of a state machine are correlated when the state encoding is varied

State Encoding Effects



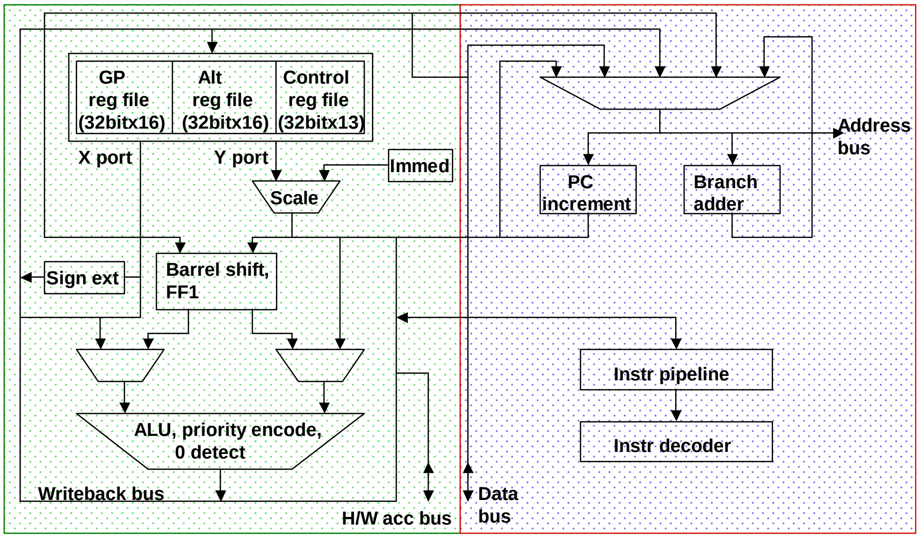
From Yeap, 1997

Practical Considerations

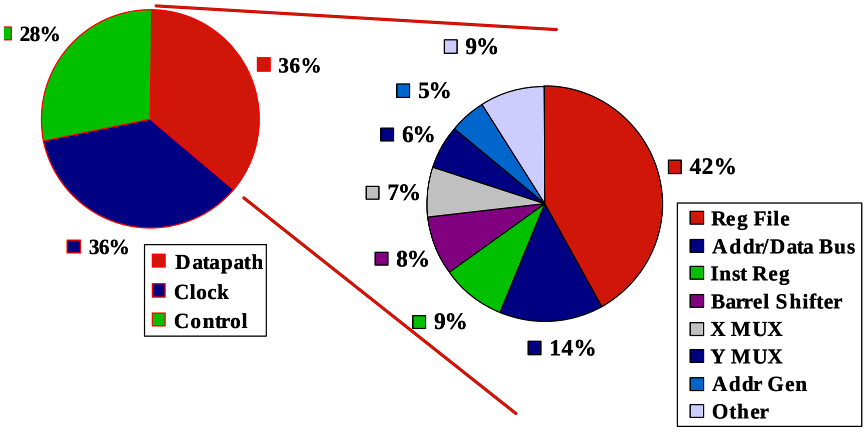
- Balance area-energy by forced encoding of only a subset of states that span the high probability edges
  - » leave assignment of remaining states to the logic synthesis system for area optimization
  - » fortunately, in practice, most state machines have this characteristic
- Unlike area encoding, energy encoding requires knowledge of probabilities of state transitions and input signals

A Low Power Processor Core Example

M•CORE Architecture



M•CORE Power Distribution



Key References

Hossain, Low power design using double edge triggered flipflop, *IEEE Trans. on VLSI Systems*, 2(2):261-265, 1994.

Motorola, *M ·CORE Architecture microRISC Engine, MCORE 1/D*, www.mot.com/SPS/MCORE/info_documentation.htm

Mutsunori, Low power design method using multiple supply voltages, *SLPED*, 1997.

Rabaey, *Digital Integrated Circuits*, Prentice-Hall, 1996.

Reyes, *Low Power FF Circuit and Method Thereof*, Patent No 5,498,988, 1996.

Roy, Power analysis and design at the system level, *Low Power Design in Deep Submicron Electronics*, Nebel and Mermet, Ed., Kluwer, 1997.

Sakuta, Delay balanced multipliers for low power, *SLPE*, 1995.

Scott, Designing the Low-Power M•CORE Architecture, *Proc. Inter. Symp. Computer Architecture Power Driven Microarchitecture Workshop*, June 1998.

Stojanovic, A unified approach in the analysis of latches and FFs for low power systems, ISLPED, 1998.

Tiwari, Reducing power in high-performance microprocessors, *DAC*, 1998.

Yeap, CPU controller optimization for HDL logic synthesis, *CICC*, 1997.

Yeap, *Practical Low Power Digital VLSI Design*, KAP, 1998.