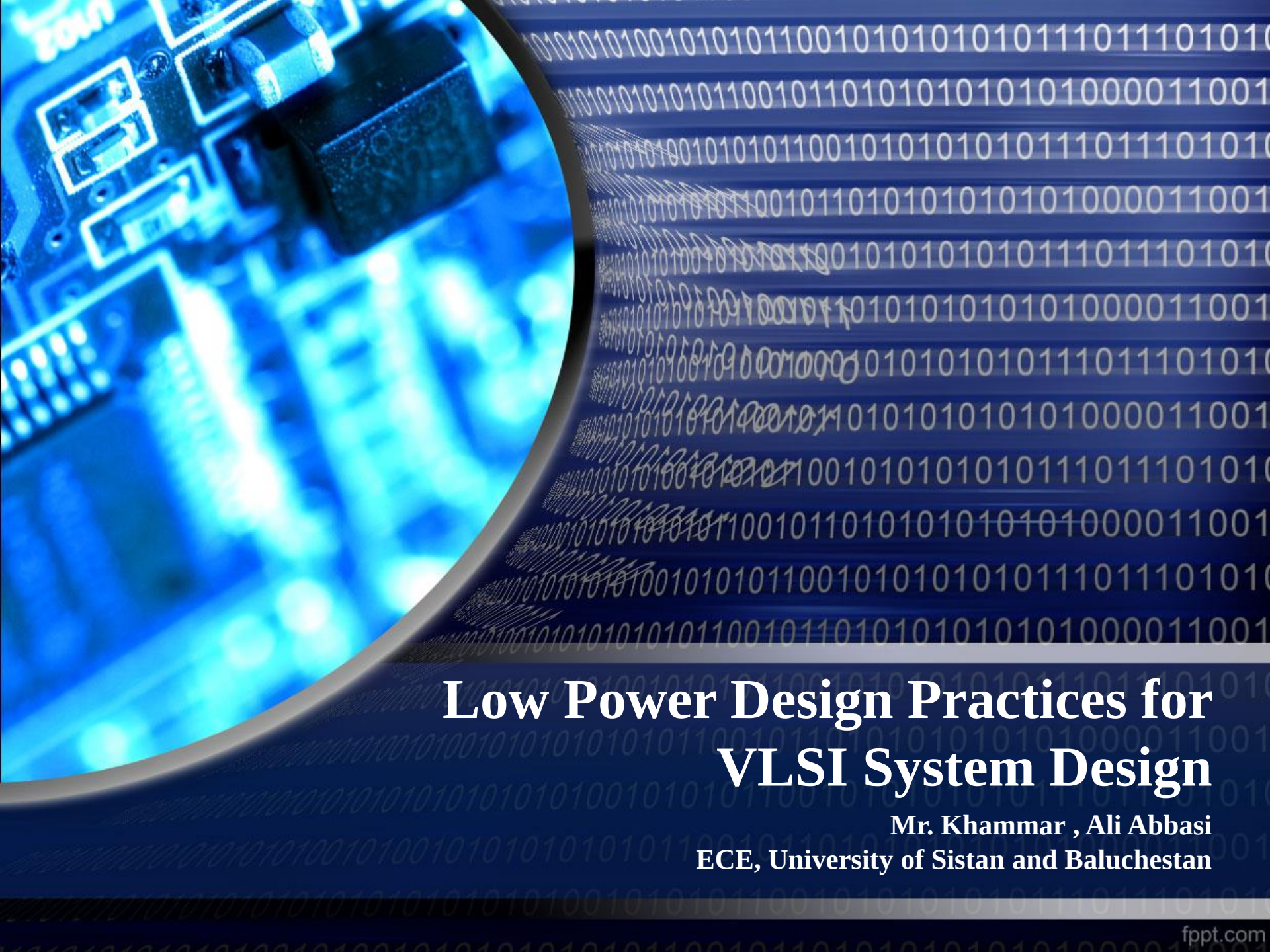


Low Power Design Practices for VLSI System Design

Mr. Khammar , Ali Abbasi
ECE, University of Sistan and Baluchestan

Contents

- Overview of Integrated Circuit Design
- Overview of VLSI Design
- Power Challenge of VLSI
- Sources of Power Dissipation
- Low Power Design Practices

Integrated Circuits Complexity

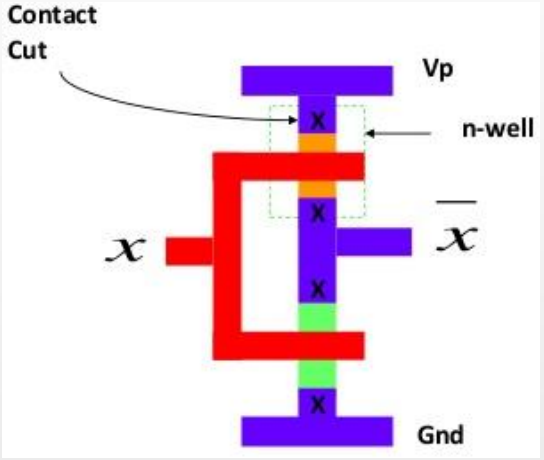
Name	Signification	Year	Transistors number	Logic gates number
SSI	small-scale integration	1964	1 - 10	1 - 12
MSI	medium-scale integration	1968	10 - 500	13 - 99
LSI	large-scale integration	1971	500 – 20,000	100 – 9,999
VLSI	very large-scale integration	1980	20,000 – 1,000,000	10,000 – 99,999
ULSI	ultra large-scale integration	1984	1,000.000 and more	100,000 and more

Integration Scales Examples

Scale	Device	Transistor Count
SSI	Logic Gates	2 - 8
MSI	256-bit DRAM	256
LSI	Zilog Z8000	17,500
VLSI	1-Mb EPROM	1,048,576
ULSI	Project Scorpio	8,000,000,000

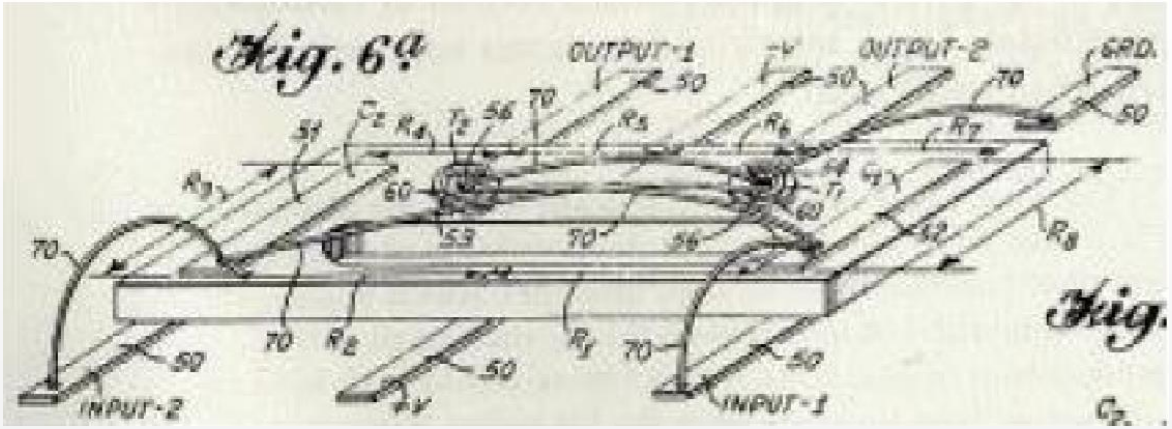
What is VLSI?

- VLSI Stands for Very Large Scale Integration
- Craver Mead of Caltech pioneered the filed of VLSI in the 1970's.
- Digital electronic integrated circuits could be viewed as a set of geometrical patterns on the surface of a silicon chip.



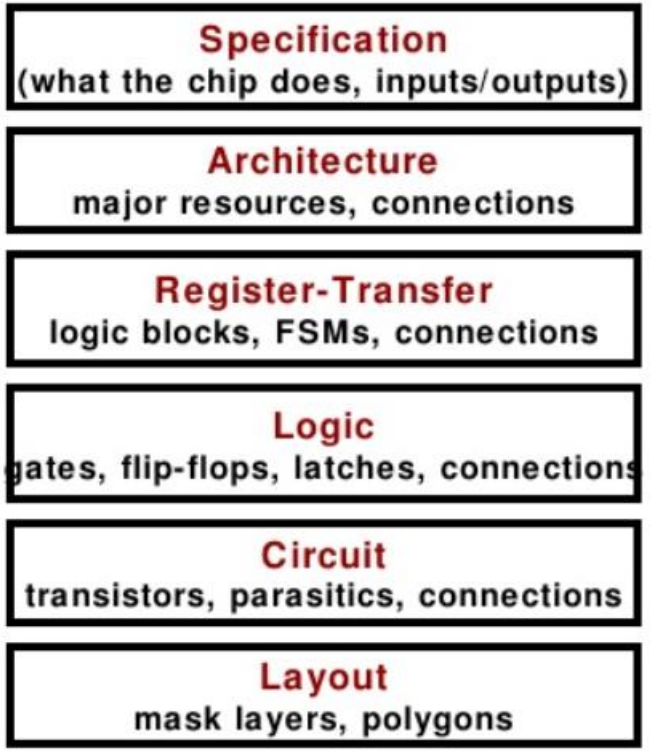
- Complexity could thus be dealt with using the concept of repeated patterns that were fitted together in structured manner.

Have you ever heard?

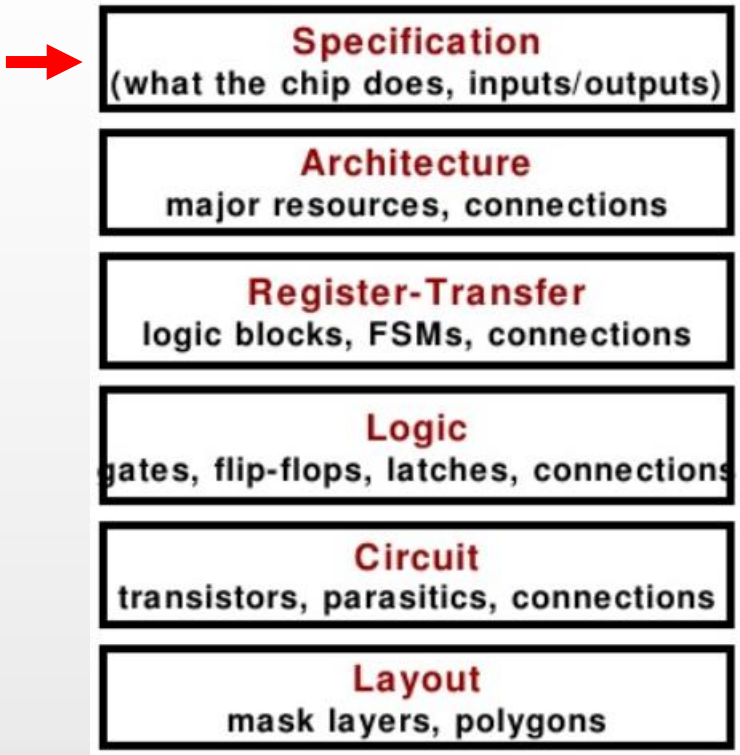


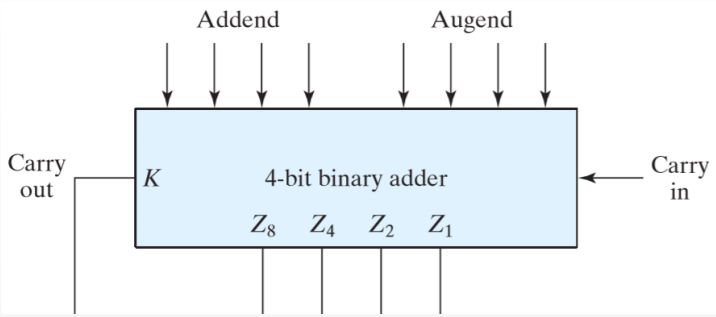
Jack Kilby was the first who dreamed up the idea of IC (in July 1959). By the end of the year, he had designed and constructed several examples including the flip-flop shown above!

VLSI Levels of Abstraction



VLSI Levels of Abstraction



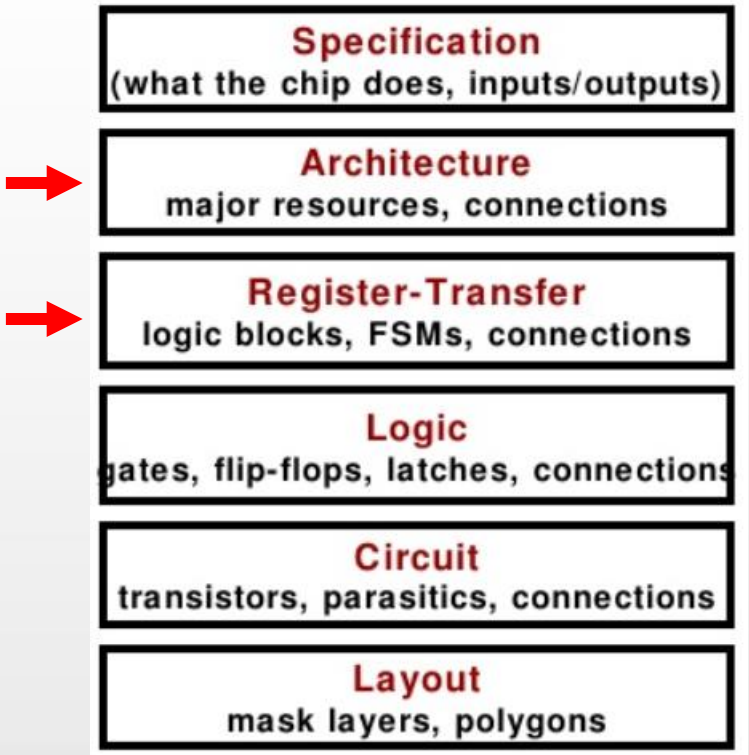


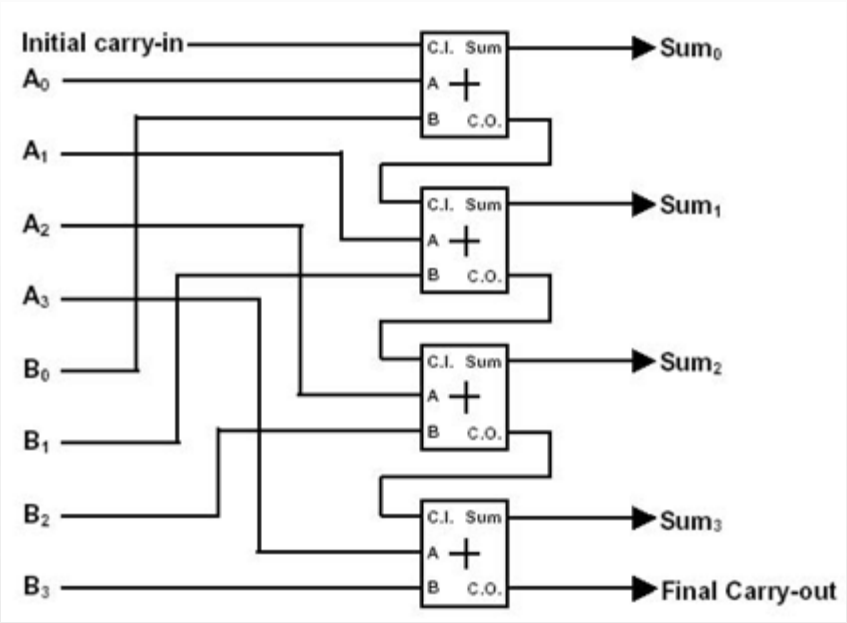

```
module binary_adder (
  output [3: 0]          Sum,
  output                 C_out,
  input [3: 0]           A, B,
  input                  C_in
);

  assign {C_out, Sum} = A + B + C_in;
endmodule
```

Picture Source: Digital Design, M. Morris Mano, page 164, with modifications

VLSI Levels of Abstraction





VLSI Levels of Abstraction

Specification
(what the chip does, inputs/outputs)

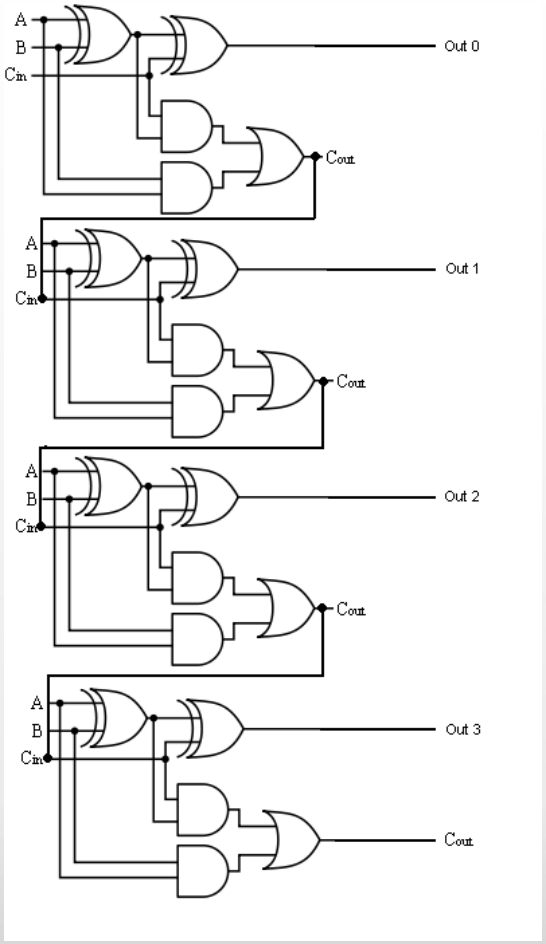
Architecture
major resources, connections

Register-Transfer
logic blocks, FSMs, connections

Logic
gates, flip-flops, latches, connections

Circuit
transistors, parasitics, connections

Layout
mask layers, polygons



VLSI Levels of Abstraction

Specification
(what the chip does, inputs/outputs)

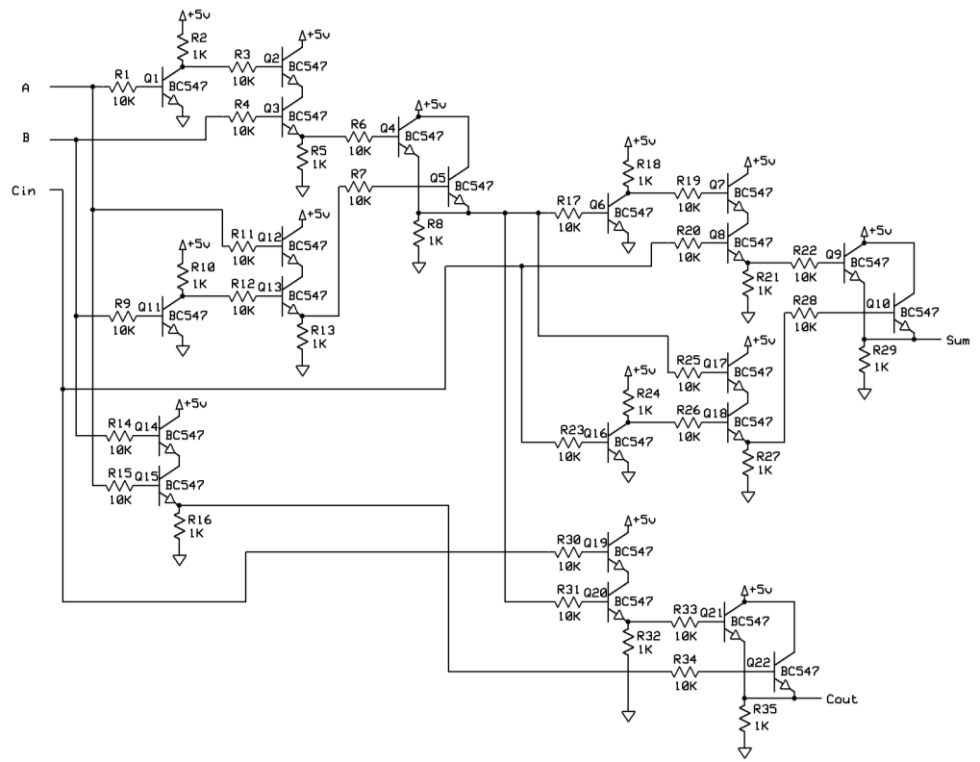
Architecture
major resources, connections

Register-Transfer
logic blocks, FSMs, connections

Logic
gates, flip-flops, latches, connections

Circuit
transistors, parasitics, connections

Layout
mask layers, polygons



VLSI Levels of Abstraction

Specification
(what the chip does, inputs/outputs)

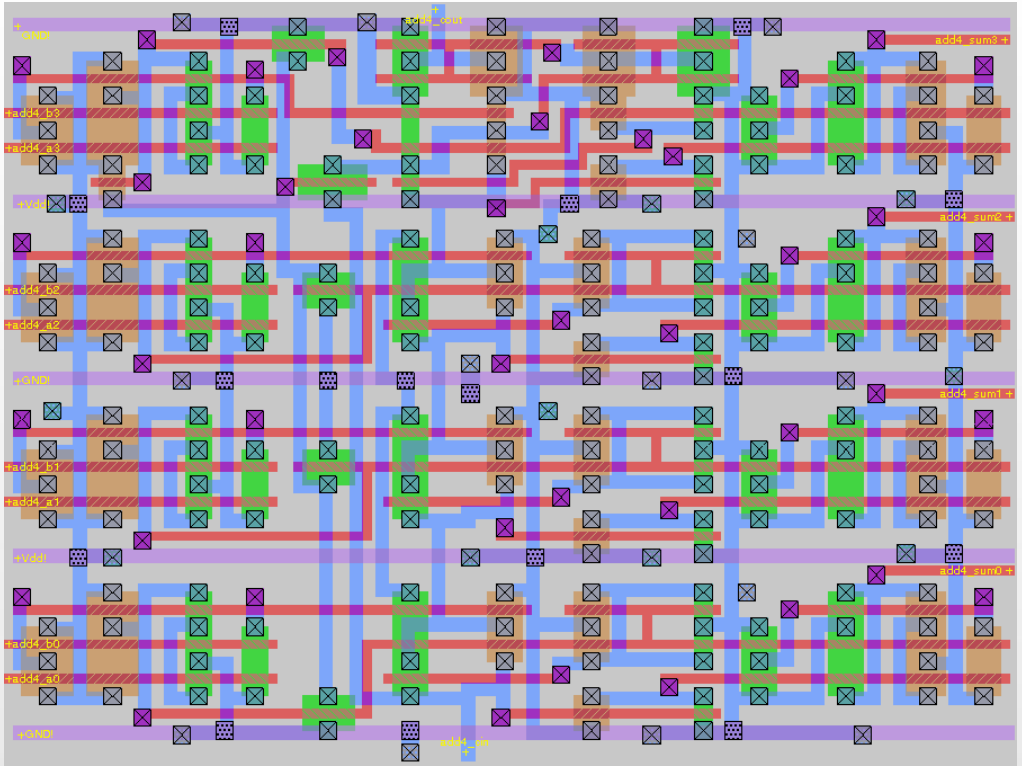
Architecture
major resources, connections

Register-Transfer
logic blocks, FSMs, connections

Logic
gates, flip-flops, latches, connections

Circuit
transistors, parasitics, connections

Layout
mask layers, polygons



Power Challenge of VLSI

I. Technology models 0.5µm, 0.35µm, 0.25µm, 0.18µm, 0.13µm, 90nm, 65nm, 45nm, 32nm, 20nm, ???

II. Scaling factor of 0.7 in the dimension exist from generation to the next one

III. Scaling in Area = 0.7 X 0.7 = 0.49 ≈ 0.5. This means the transistor density doubles every generation.

Power Challenge of VLSI (cont.)

IV. Electrical nodes in a given area increase by 100%

V. Die size grows by 14% every two years

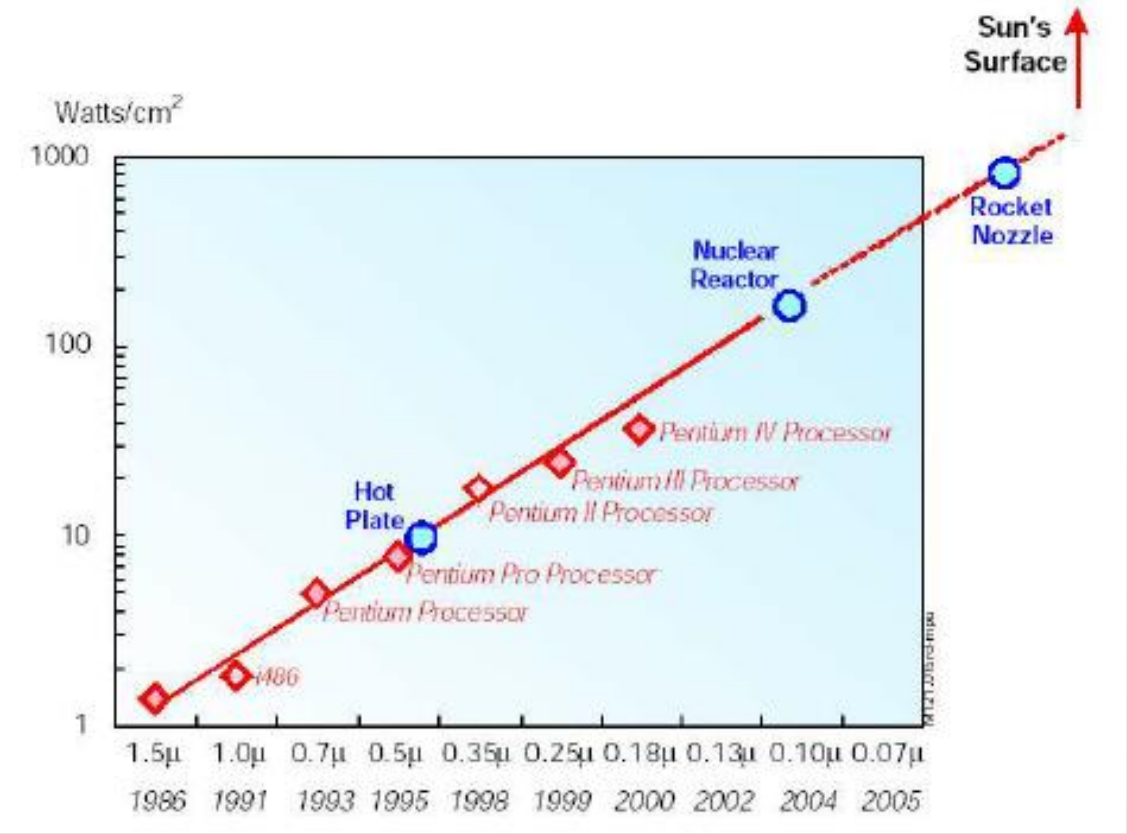
VI. Supply voltage reduces by 15% every two years

VII. Capacitance per node reduces by 30%

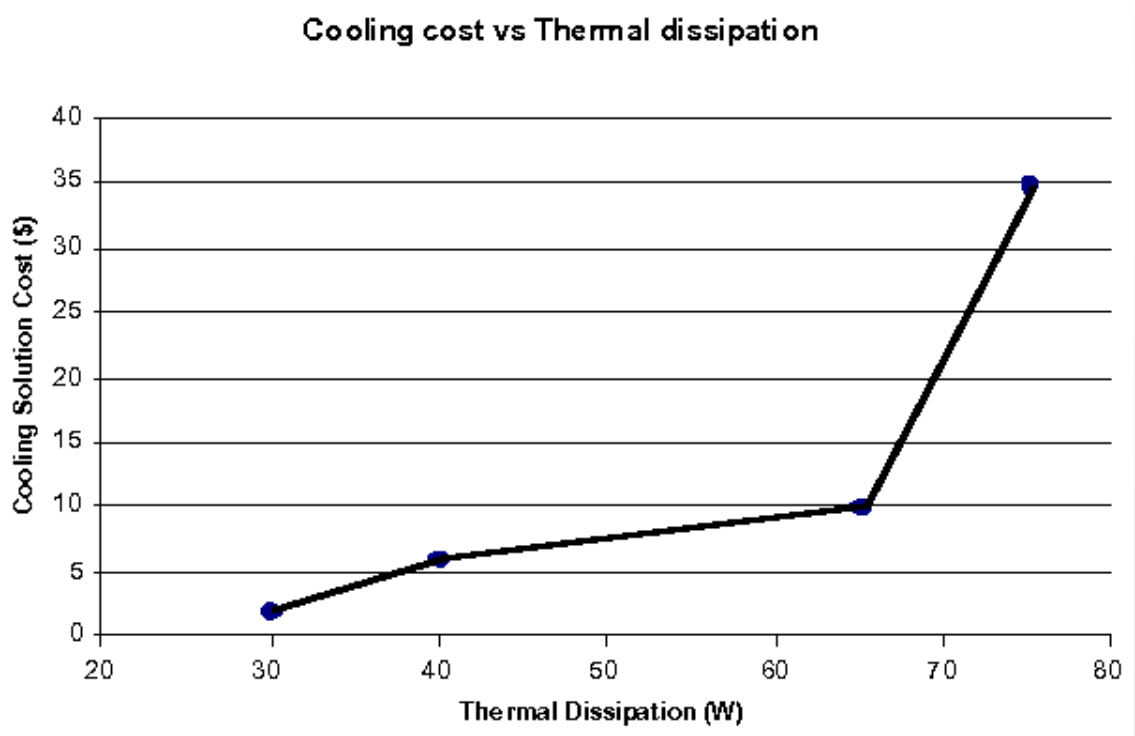
VIII. Frequency increases by 100%

Up to 2.7X increase in active power of the lead microprocessor every two years

Power Challenge of VLSI (cont.)



Power Challenge of VLSI (cont.)



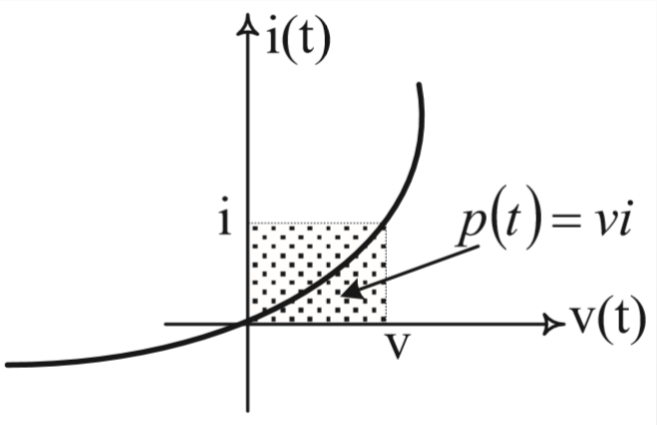
Some Useful Information About Power, Voltage, Current, and Performance!

Let's Get Back to the Electrical Circuits

$$P(t) = V(t).I(t)$$

$$Power \propto Performance$$

$$Performance \propto \frac{1}{Felexibility}$$

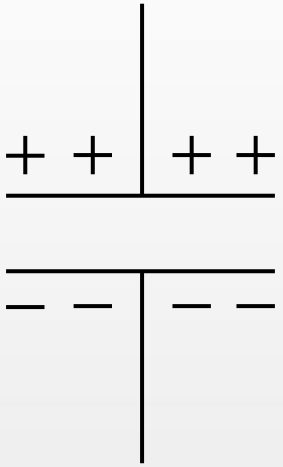


Picture Source: 2424224 at USB, Lecture 6, by Dr Jafari, with modifications

Sources of Power Dissipation

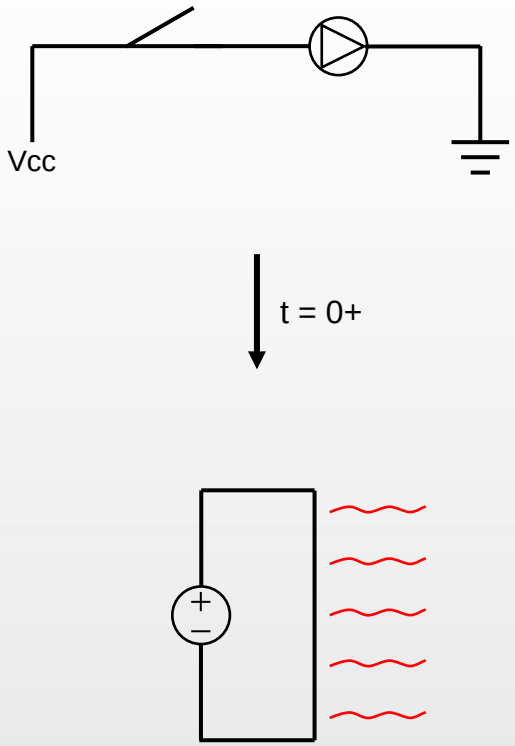
Sources of Power Dissipation

I. Switching



Sources of Power Dissipation

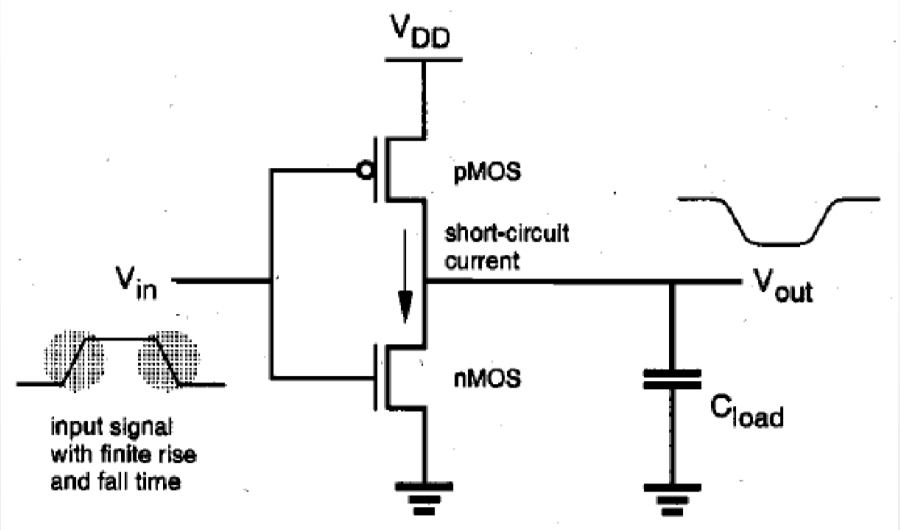
I. Switching
II. Short Circuit



Sources of Power Dissipation

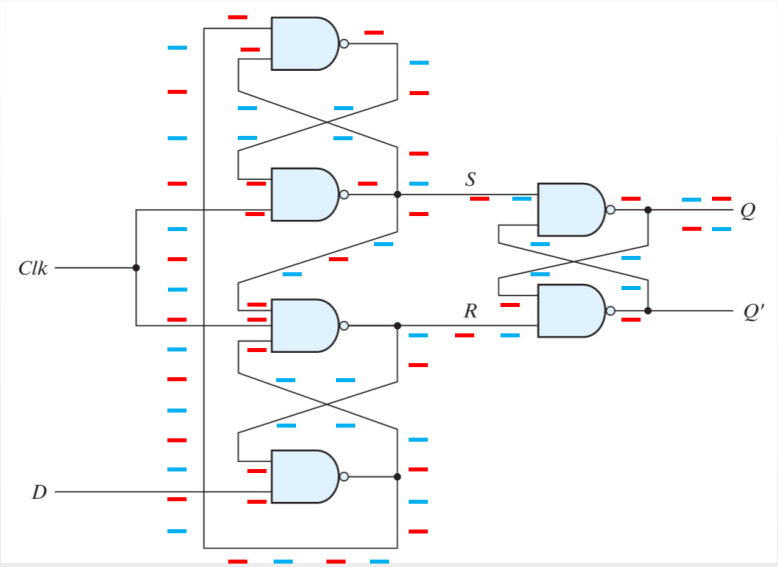
I. Switching

II. Short Circuit



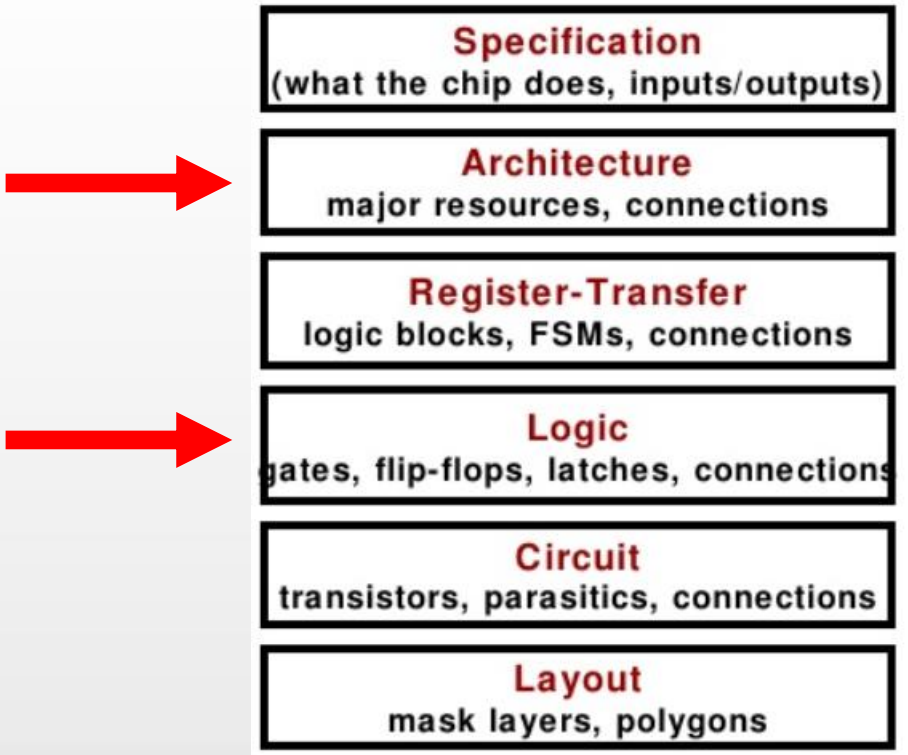
Sources of Power Dissipation

I. Switching
II. Short Circuit
III. Leakage



Picture Source: Digital Design, M. Morris Mano, page 199, with modifications

VLSI Levels of Abstraction



Logic Level Power Optimization Techniques

Logic Level Power Optimization Techniques

1. Gate Reorganization
2. Signal Gating
3. Logic Encoding
4. State Machine Encoding
5. Precomputation Logic
6. Path Equalization

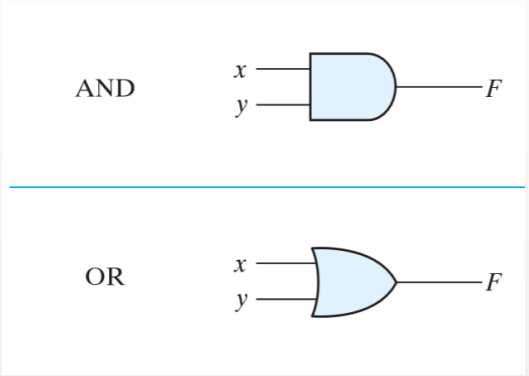
Logic Level - Gate Reorganization

- Combination
- Decomposition
- Duplication
- Removing Wire
- Adding Wire

Logic Level - Signal Gating

Methodes:

1. Simple AND/OR Gate

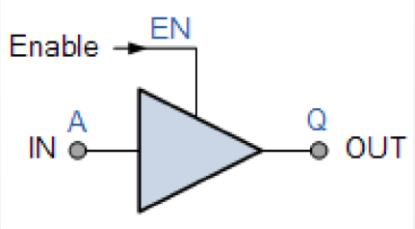


Picture Source: Digital Design, M. Morris Mano, page 61, with modifications

Logic Level - Signal Gating

Methodes:

1. Simple AND/OR Gate
2. Tristate Buffer

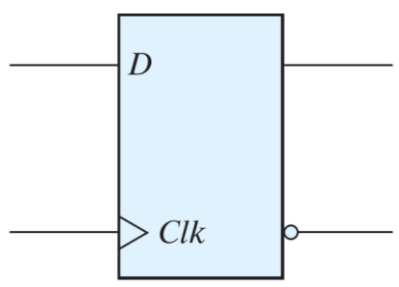


Picture Source: 2422445 at USB, Lecture 14, by Dr Khammar, with modifications

Logic Level - Signal Gating

Methodes:

1. Simple AND/OR Gate
2. Tristate Buffer
3. Latch/FF

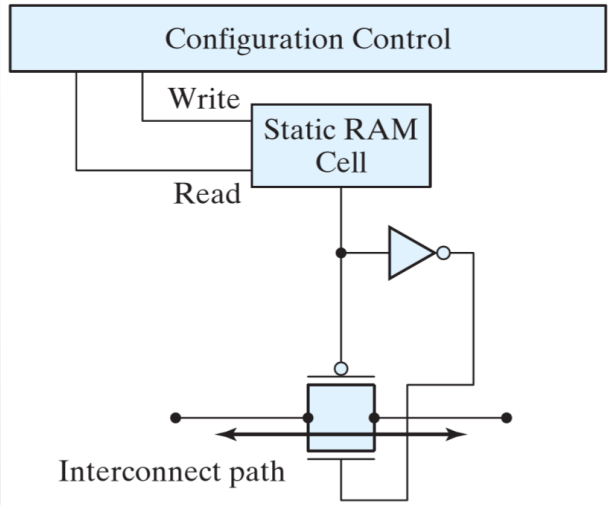


Picture Source: Digital Design, M. Morris Mano, page 200

Logic Level - Signal Gating

Methodes:

1. Simple AND/OR Gate
2. Tristate Buffer
3. Latch/FF
4. Transmission Gate



Picture Source: Digital Design, M. Morris Mano, page 565

Logic Level - Logic Encoding

- The amount of power consumption is tightly related to the number of transitions.
- A combination of bits create a concept for a digital signal.
- We may take the advantage of the properties of this concept to save the number of transition that we need to communicate it.
- So, What is the solution?

  Alternative approaches to represent a value, Instead of using a general-purpose binary format (such as unsigned or 2's complement).

  These approaches are **hot-coding** and **gray-coding**.

Logic Level - Logic Encoding (cont.)

- Assume **n** as the count of bits, then the number of transitions in a n-bit **binary counter** is equal to:

$$N.T. = 2(2^n - 1)$$

- **Hot Coding:** Every codeword has exactly one bit equal to 1. then the number of transitions is:

$$N.T. = n$$

- **Gray Coding:** Consequent binary numbers are mapped to codeword with unit distance. Thus, N.T. is equal to:

$$N.T. = 2^n$$

Logic Level - Logic Encoding (cont.)

- Performances of Gray code and binary code for representing data and instruction addreesses:

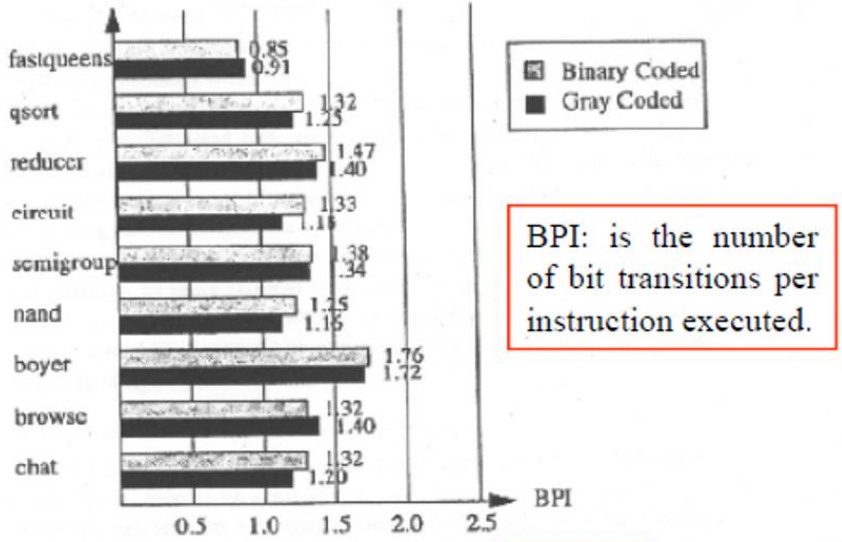


Figure 7.4: Temporal transition activity comparison for data addresses [Su94]. (© 1994 IEEE)

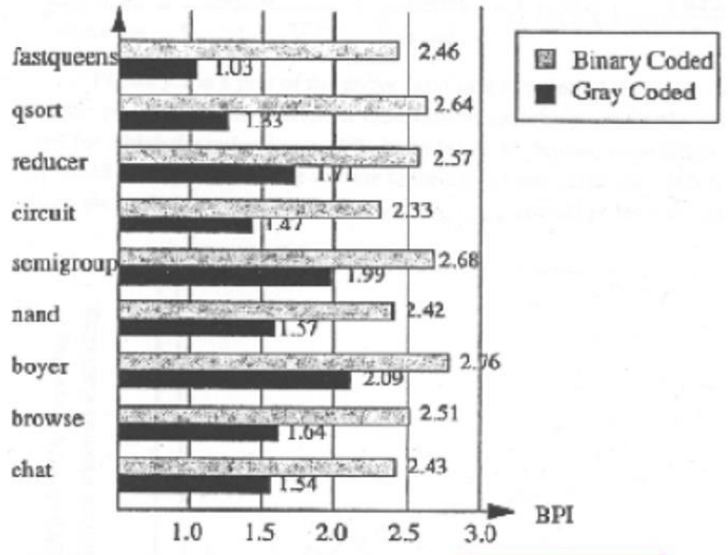
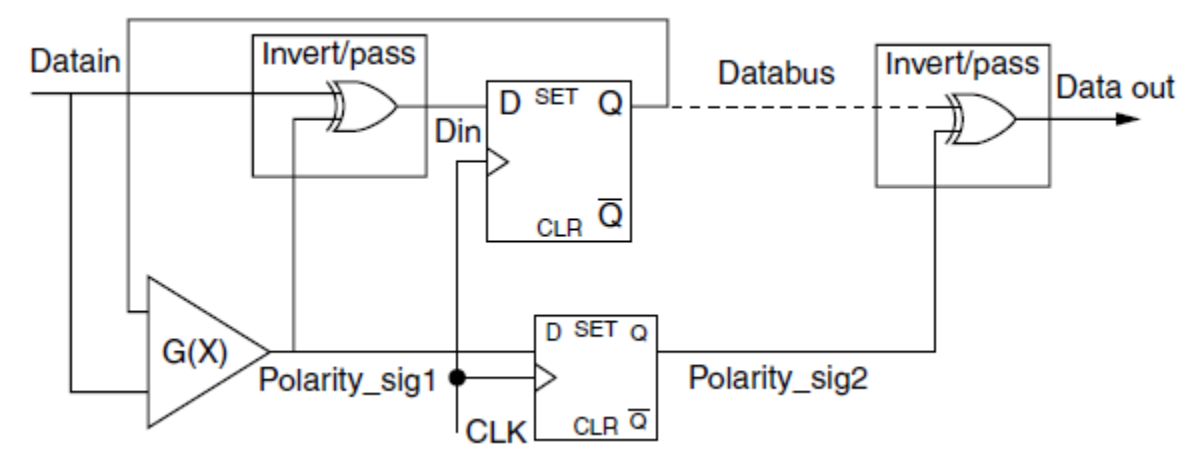


Figure 7.3: Temporal transition activity comparison for instruction addresses [Su94]. (© 1994 IEEE)

Logic Level - Logic Encoding (cont.)

- Bus Invert Encoding Technique



For example,

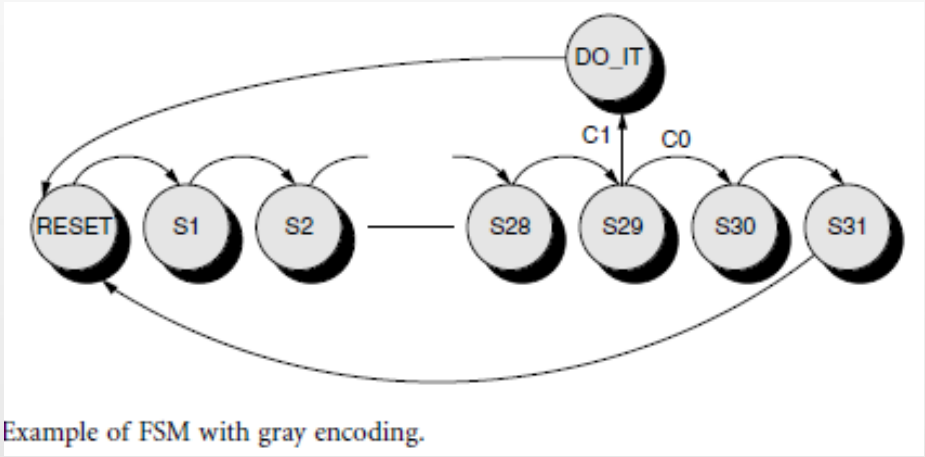
Current value of the bus is 0000, and the next value is 1110,

It will send the complement of the next value (0001) and assert the polarity signal.

- Complexity: Hamming Distance between two consecutive codes
- However, what about the overhead?

Logic Level - State Machine Encoding

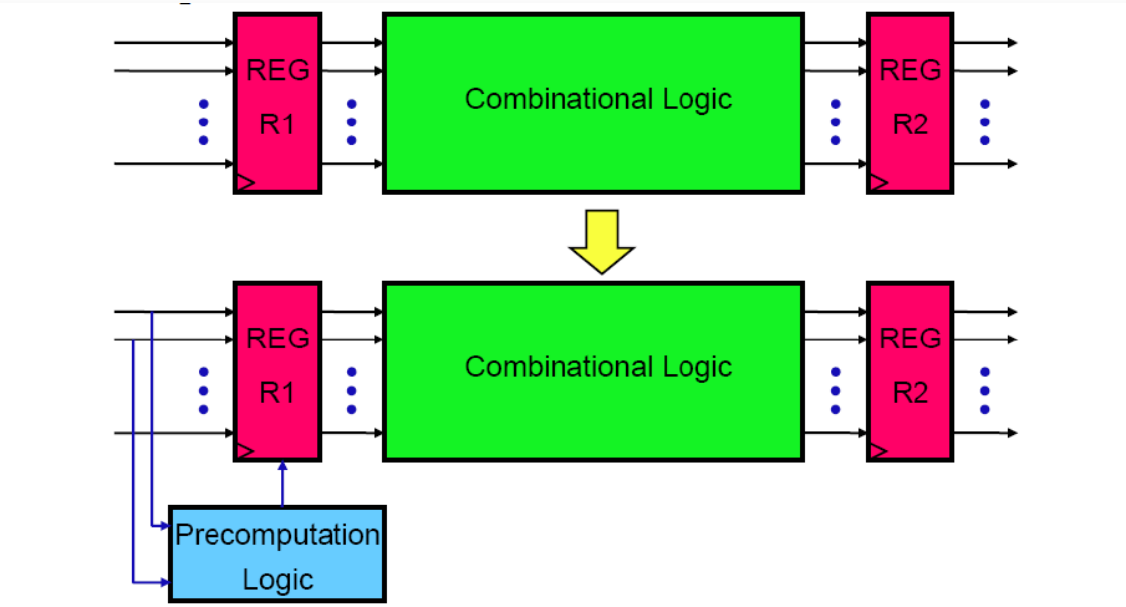
- Most of the times, the code we use to represent a state is arbitrary.
- We should minimize the hamming distance of the transition with high probability.



Example of FSM with gray encoding.

Logic Level - Precomputation Logic

- The principle of precomputation logic is to identify logical conditions at some inputs to a combination logic that does not affect the output.

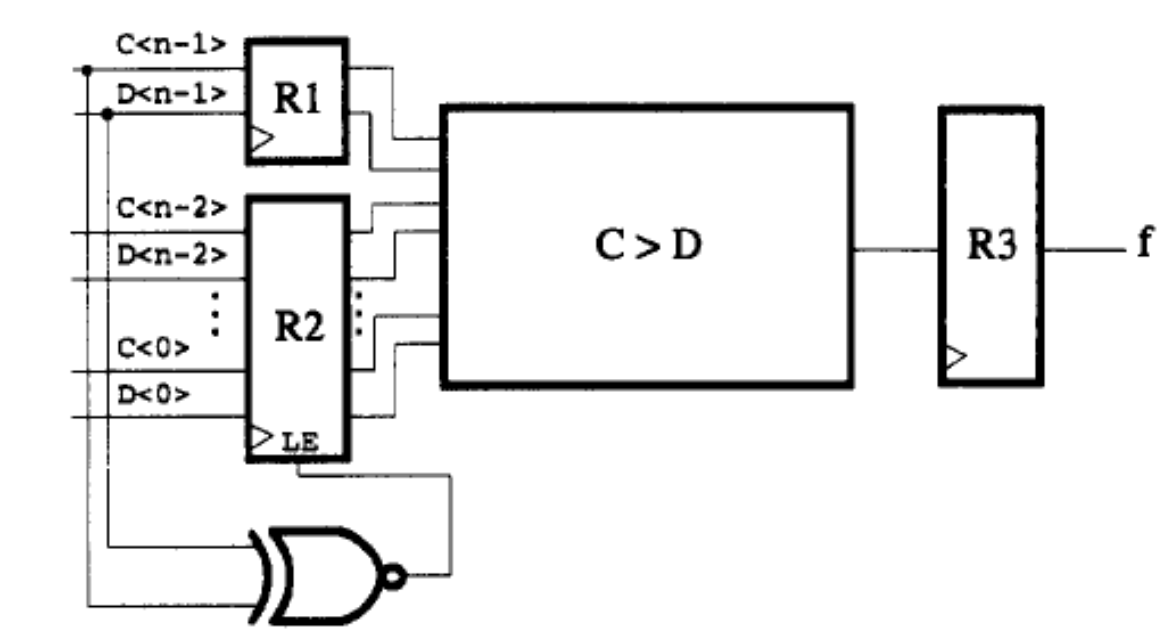


- For example: A Magnitude Comparator

Picture Source: Computer Architecture, University of Tehran, by Ali Afzali-Kusha, with modifications

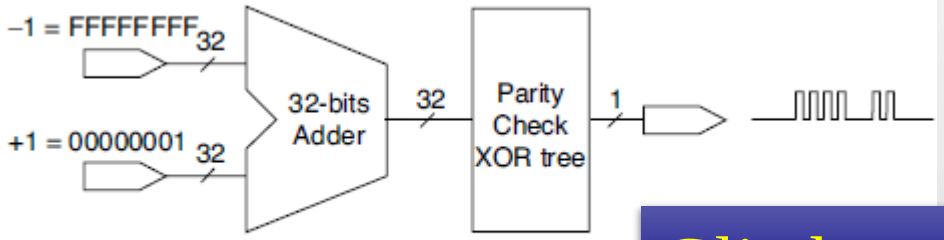
Logic Level - Precomputation Logic (cont.)

- Magnitude Comparator using precomputation logic



Logic Level - Path Equalization

- Path equalization ensures that signal propagation from inputs of a logic network to its outputs follows optimal or similar length paths.
- Something called Glitch, which means the output of a combinational logic settles to the right value after a number of transitions between 1 and 0.
- Example: Parity of the output of a ripple carry adder when it adds ‘111111’ with ‘000001’.

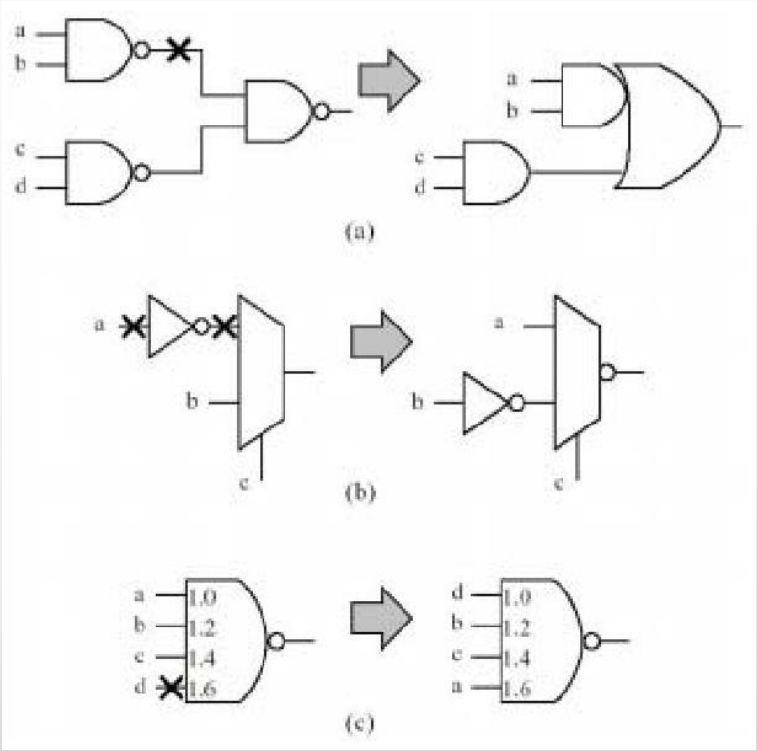


Example of glitches generation.

Glitch propagates!

Logic Level - Path Equalization (cont.)

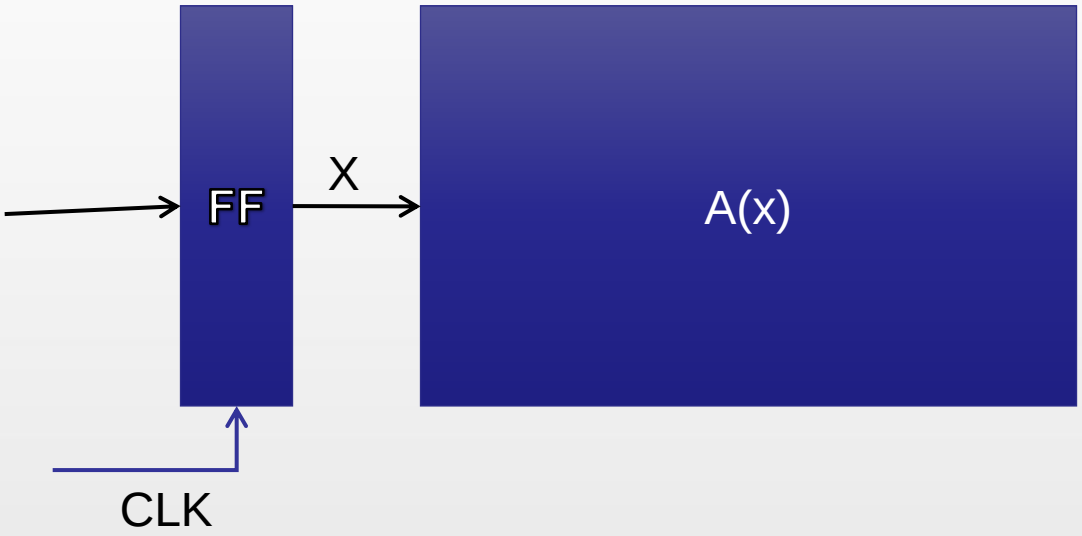
- Also, the path equalization can be done based on VLSI design methods.



Architecture Level Power Optimization Techniques

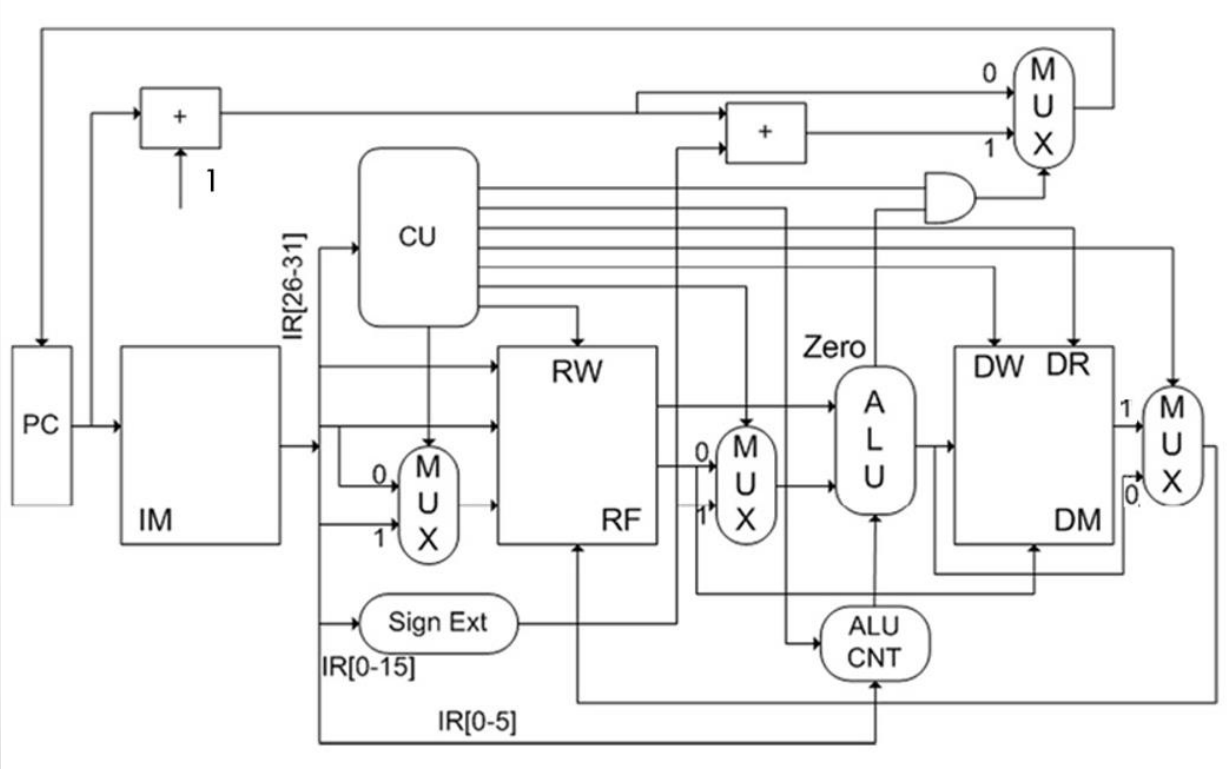
Architecture Level - Clock Gating

- We can gate the clock of FFs if the output value of A is not needed.



Architecture Level - Clock Gating (cont.)

- Assume a Multi Cycle MIPS processor:



Picture Source: 2422307 at USB, Lecture 10, by Dr Sargolzaei

References

[1] Bellaouar and Elmasry 2012, Yeap 2012, Chakraverty, Harisankar et al. 2016) Bellaouar, A. and M. Elmasry (2012). Low-power digital VLSI design: circuits and systems, Springer Science & Business Media.

[2] Chakraverty, M., et al. (2016). Low power design practices for power optimization at the logic and architecture levels for VLSI system design. 2016 International Conference on Energy Efficient Technologies for Sustainability (ICEETS), IEEE.

[3] Devadas, S. and S. Malik (1995). A survey of optimization techniques targeting low power VLSI circuits. Proceedings of the 32nd annual ACM/IEEE Design Automation Conference, ACM.

[4] Horowitz, M., et al. (1994). Low-power digital design. Proceedings of 1994 IEEE symposium on low power electronics, IEEE.

[5] Mano, M. M. and M. Ciletti (2013). Digital design: with an introduction to the Verilog HDL, Pearson.

[6] Patterson, D. A. and J. L. Hennessy (2013). Computer organization and design MIPS edition: the hardware/software interface, Newnes.

[7] Piguet, C. (2005). Low-power CMOS circuits: technology, logic design and CAD tools, CRC press.

[8] Segars, S. (2001). Low power design techniques for microprocessors. International Solid-State Circuits Conference Tutorial.

[9] Yeap, G. K. (2012). Practical low power digital VLSI design, Springer Science & Business Media.

Any Question?!

Thanks for your kind attention!