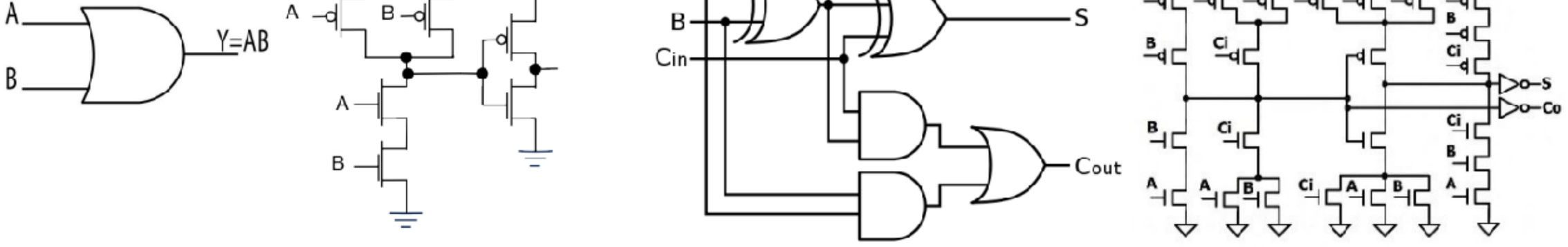
Standard Cells

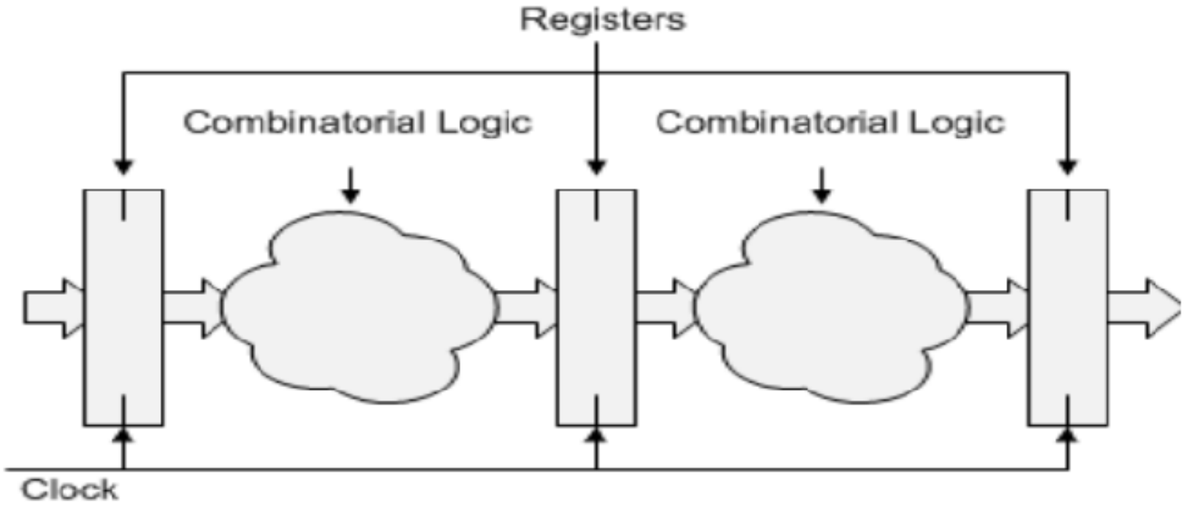
Digital Design (STD Cells)



- Digital Circuits are made of basic building blocks (gates, Mux, Decoder, Adders, Registers, …)
- Instead of designing the same circuit each time, carefully design the standard cell and "instantiate" a copy when needed.
- Beginning of abstraction where digital designers get isolated from circuit details (transistors, sizing, layout, etc …)

Digital Design (RTL Modeling)

- The simple concept behind "synchronous" circuits enables the description of digital circuits as data-flow between registers "Register Transfer Logic"
- No transistors, no logic gates, only registers (variables) and operation on the data (programming constructs)
- Digital designers are increasingly isolated from circuits, even sometimes confuse RTL modeling with programming!!!
- RTL works perfectly with synchronous circuits (under the assumption of worst case design). ***What about other circuit classes (Asynchronous)?***



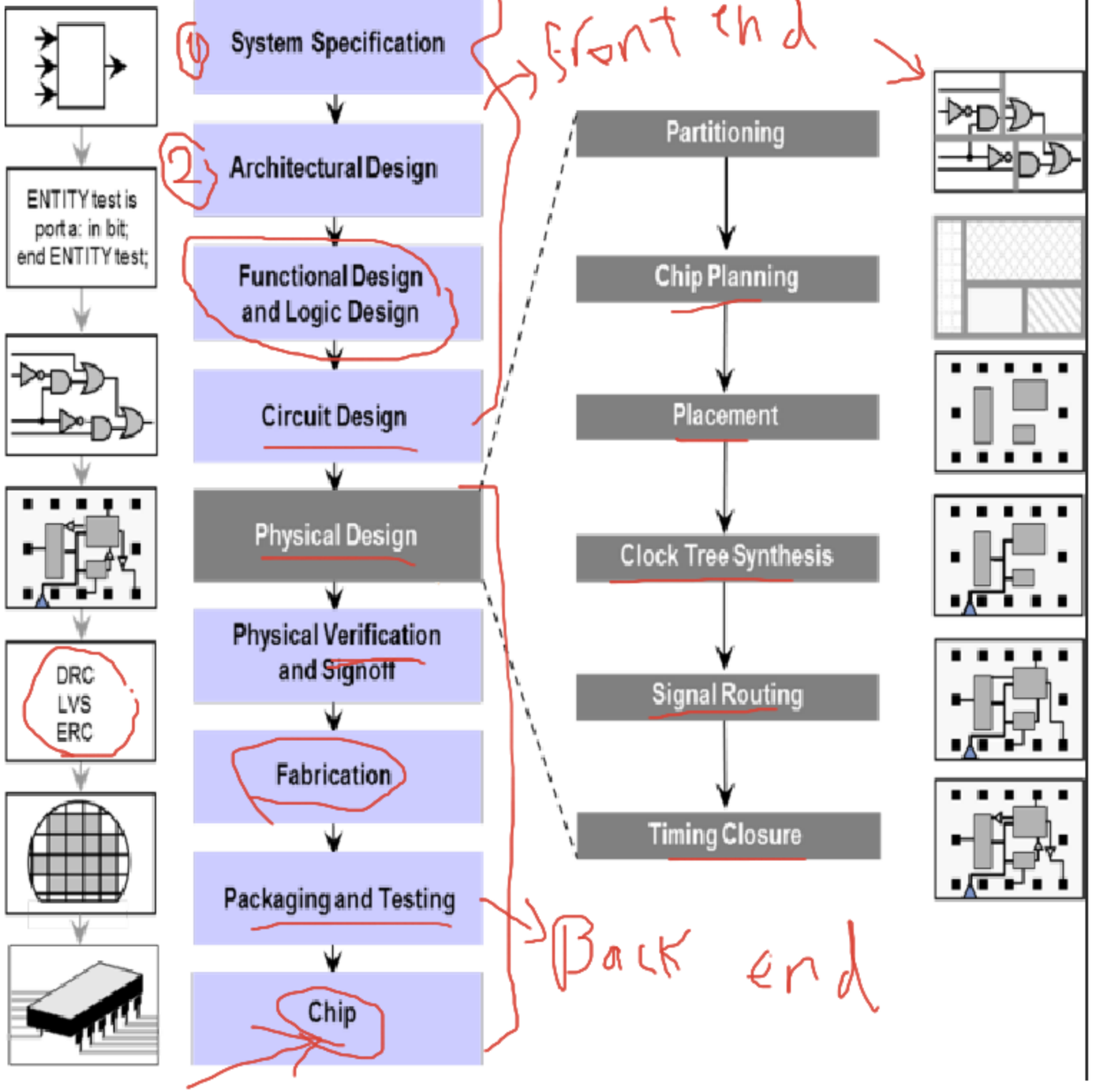
Digital Design (Synthesis)

- Digital designs were created by human designers, where they model their systems in different ways (equations, truth tables, state diagrams, etc), and then use some minimization methods (such as Karnaugh maps) to optimize the logic.
- As the complexity of digital designs grown, we needed more error-proof, automation friendly methods
- Logic Synthesis "transforming desired circuit ***behavior*** to ***logic gates***"
- Designs now are described on an abstracted level
- Digital designers are more isolated from implementation details, which increase productivity

What about efficiency?

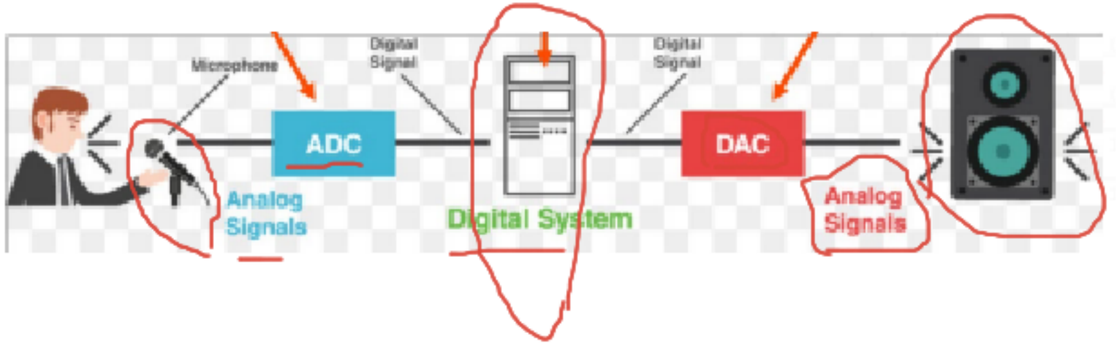
Physical Implementation

16 32
ALU



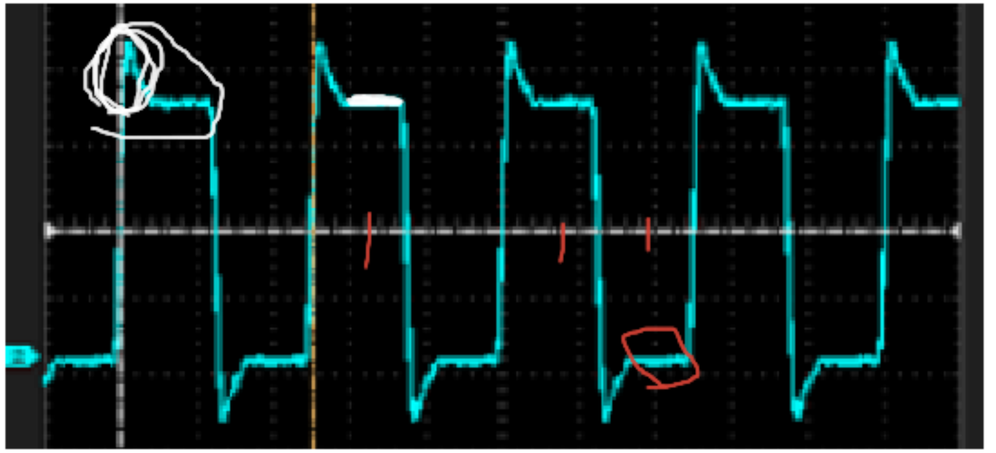
Analog

- Continuous in time and amplitude
- Sensitive to noise



Digital

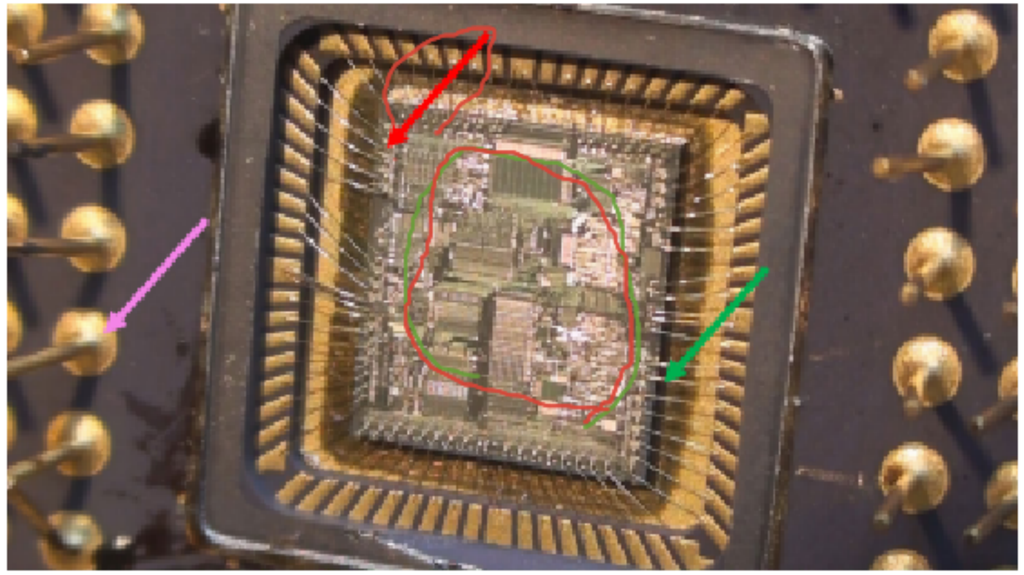
- Discrete in time and amplitude
- Noise Immunity
- Easy storage, and processing

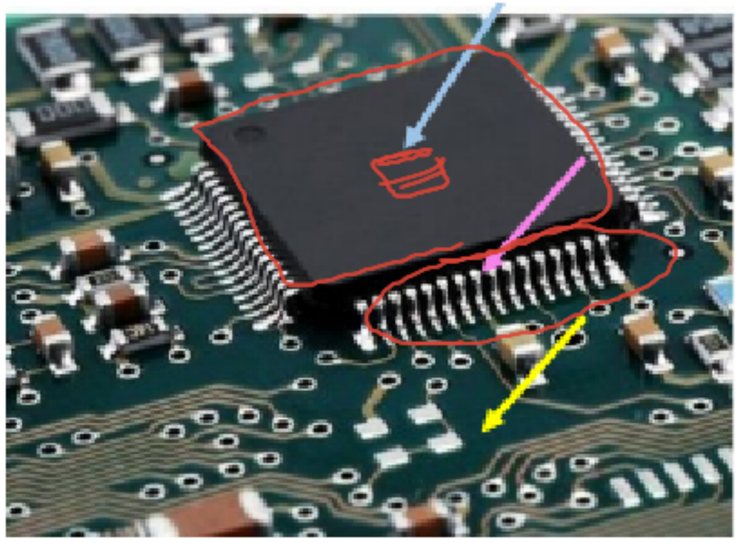


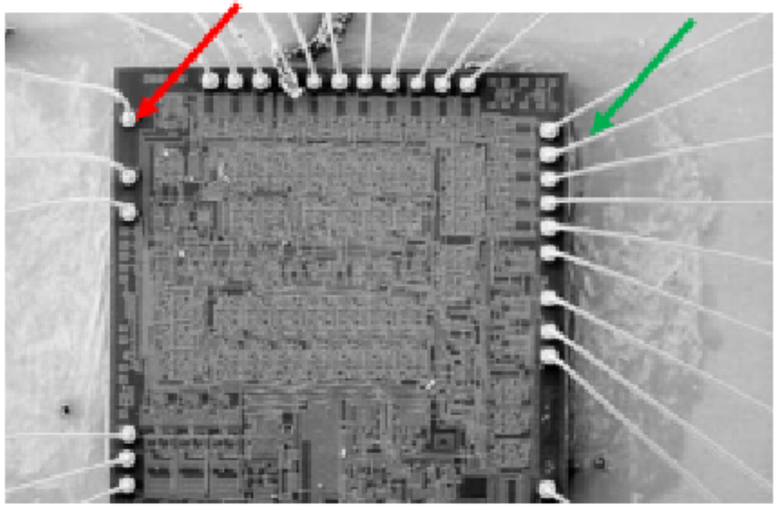
- Continuous not discrete
- We are supposed to read them only when they are at specific voltage level (read them at a specific time)

Digital System

- I/O pads
- Wire bonding
- Package
- Package pins
- PCB

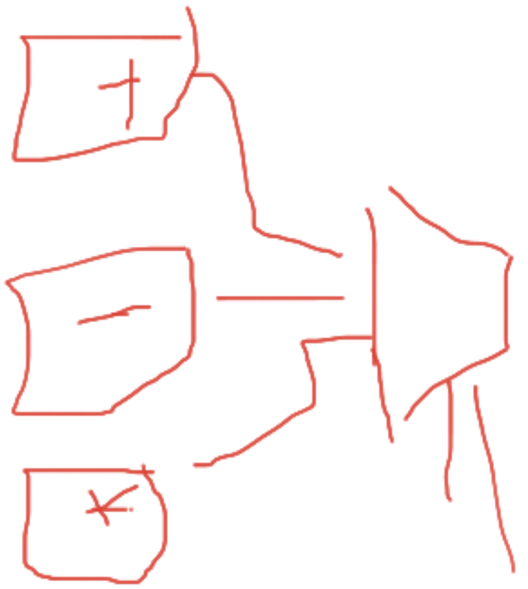






Design ALU

1- System specification
2- I/O Set
3- Control Signals
4- Block Diagram
5- Timing Diadram
6- Logic Description (HDL, Schematic..etc)
7- Testing plan



1- Specifications:

32 or 16 bits?

shift left or right?

internal or external memory?

2- I/O Set

input and output buses

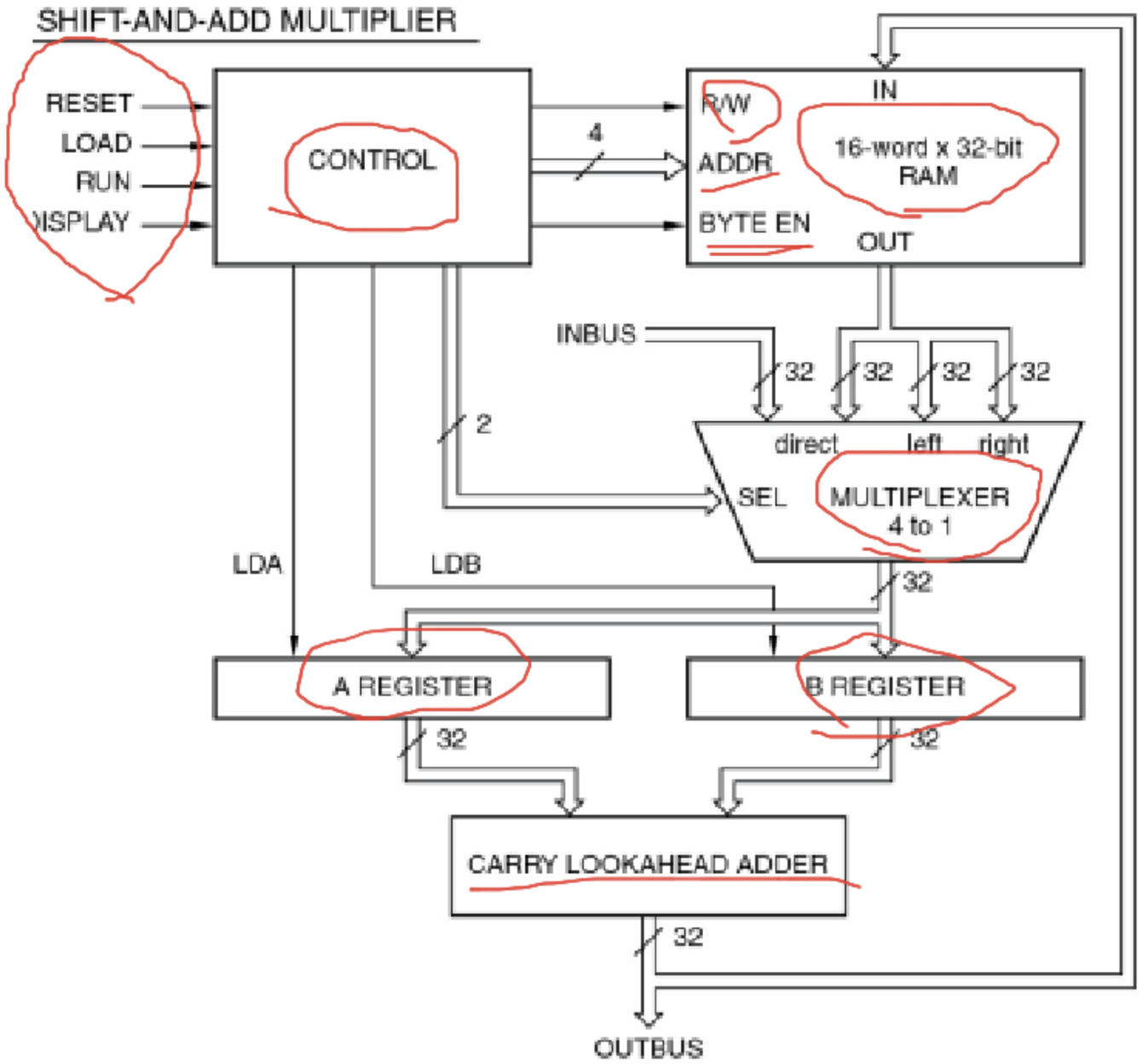
3- Control Signals

Reset

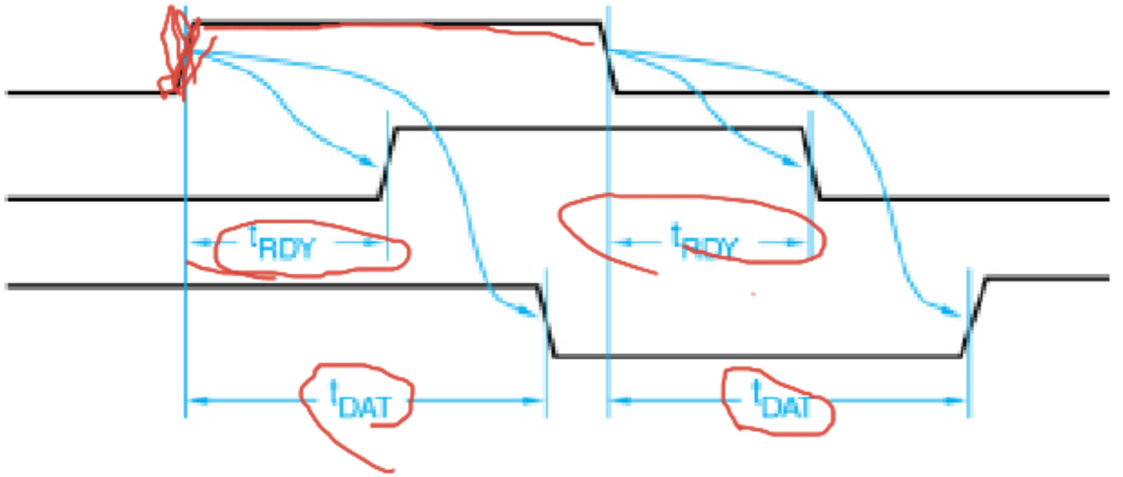
Load

Display

4- Block Diagram



5- Timing Diagram



6- Logic Description (HDL)

```
module Add_full (sum, c_out, a, b, c_in);
 input a, b, c_in;
 output c_out, sum;
 Add_half M1 (w1, w2, a, b);
 Add_half M2 (sum, w3, c_in, w1);
 or(c_out, w2, w3);
endmodule
```

7- Testing Plan