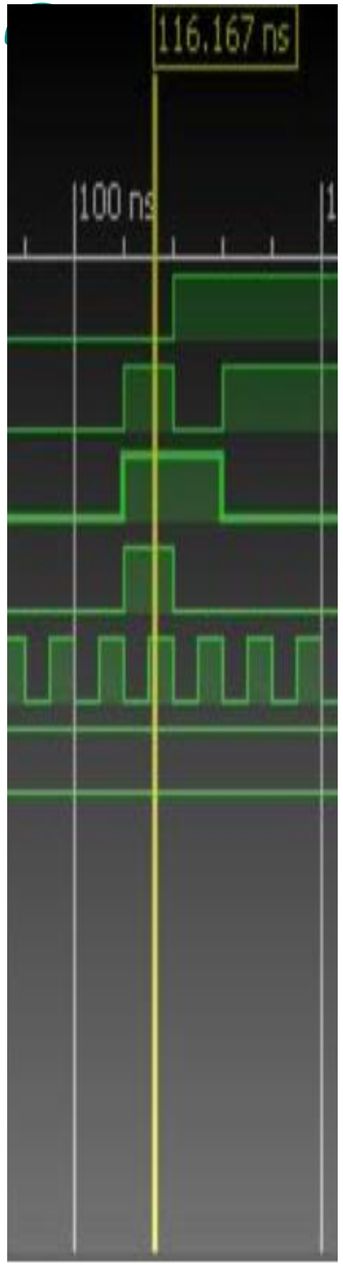
Serial Adder

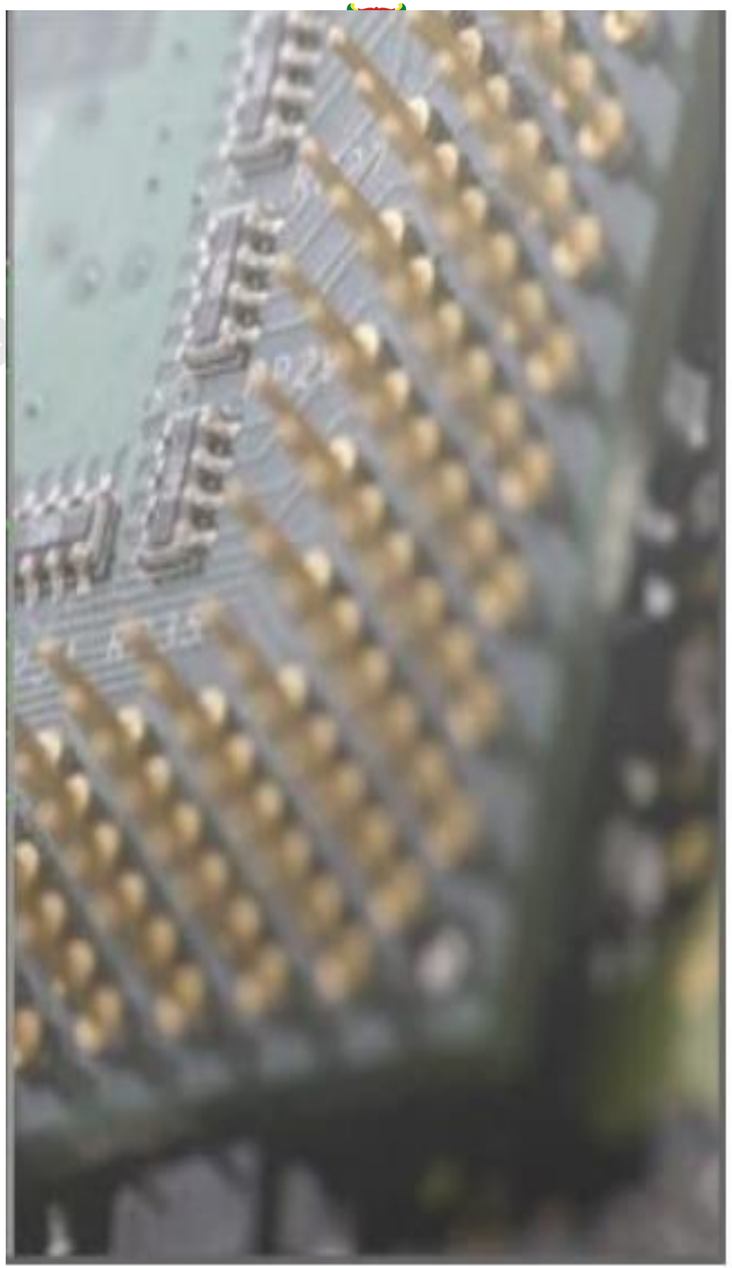
MC. Martin González Pérez

Agenda

- Serial adder

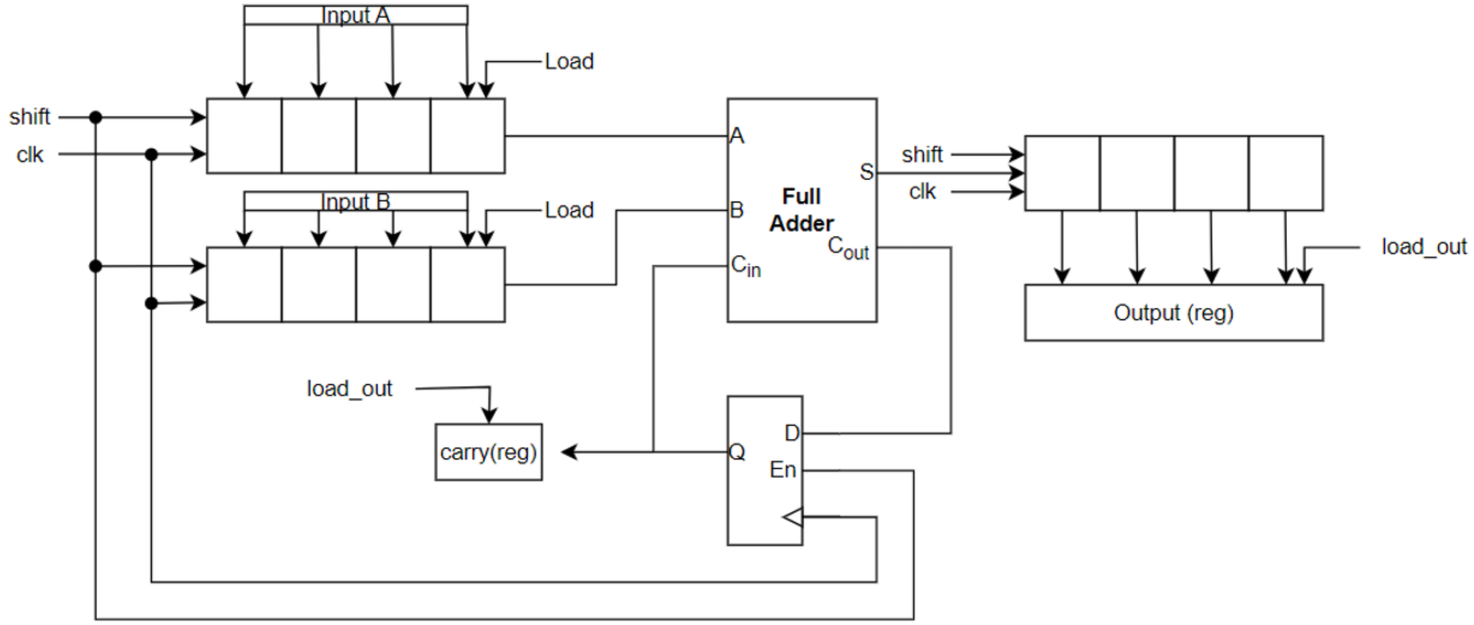


Serial adder



Serial adder

A serial adder is an arithmetic circuit that adds two numbers bit by bit on each clock cycle, using a shift register to process the operands sequentially. It is an efficient hardware alternative when optimizing resource usage is desired, rather than using a parallel adder.



Serial adder: Ports

Design a 4-bit serial adder with the following characteristics:

The adder must have the following inputs and outputs.

Inputs.

- **clk:** Clock.
- **rst:** Synchronous reset.
- **start:** Signal that initiates the addition process.
- **in_a:** 4-bit bus used to provide the summand “a”.
- **in_b:** 4-bit bus used to provide the summand “b”.

Outputs

- **sum:** 4-bit bus that reflects the result of a+b.
- **c:** Carry resulting from the addition of a+b.

Serial adder: Operation

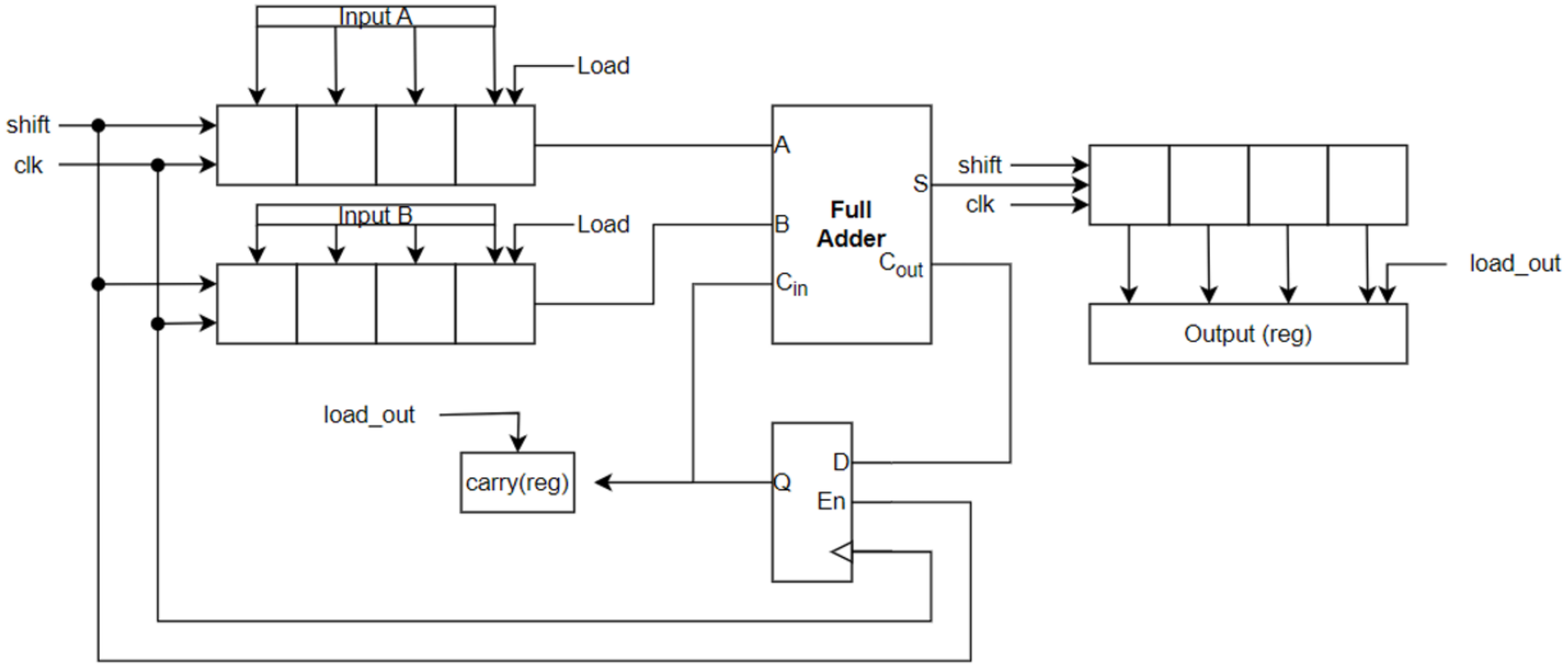
Design a 4-bit serial adder with the following characteristics:

General operation.

- Sum initialization:
  - The start signal should normally be high.
  - When **start** is low, the values of **in_a** and **in_b** are captured by the circuit.
  - When **start** is low and transitions to high, it initiates the addition process through bit shifting.
- Result: Once the addition is completed, the result must be held in an output register until another addition is performed and the output register is overwritten.
- Reset: The synchronous reset signal (**rst**) must restart the addition process when active.

Serial adder: Microarchitecture

The figure describes the data flow of the serial adder; however, it includes signals such as **shift** and **load_out** that do not correspond to the input signals. Therefore, it is necessary to create a control circuit that generates these signals and controls addition sequence.



Serial adder: Microarchitecture

For the design of the serial adder, the use of multiple submodules is proposed, the proposed modules are described below.

Datapath.

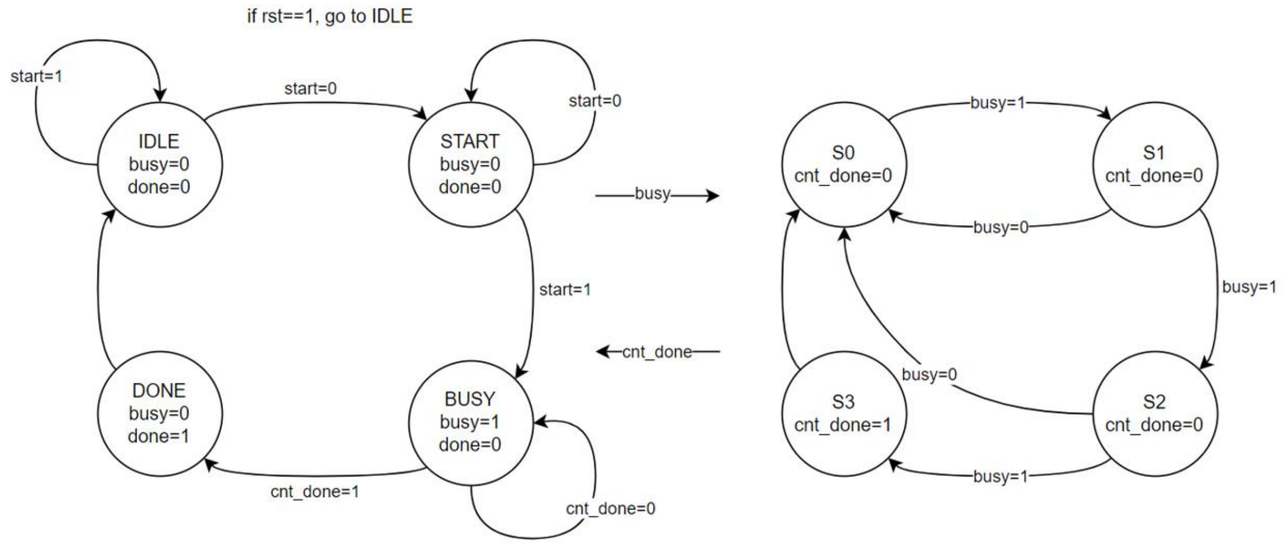
- **input_reg:** A shift register with enable and parallel load. This register loads the value of **in_a** (or **in_b**) in parallel when **start** is low, and it performs the bit shift when **control_fsm** indicates it. The output of this module is the least significant bit (LSB) of the register.
- **full_adder:** A 1-bit full adder.
- **output_reg:** A shift register with enable. The outputs of this module are all the bits of the register.
- **carry_logic:** A simple register that takes as input the carry_out of the adder and the output is connected to the carry_in of the adder.
- **output_registers:** Registers that store the result until the next addition occurs.

Serial adder: Microarchitecture

For the design of the serial adder, the use of multiple submodules is proposed, the proposed modules are described below.

Control logic

- **control_fsm:** A finite state machine that initializes the addition sequence and generates the **shift** and **load_out** signals.
- **counter:** A counter that manages the bit shifting process.



Serial adder

Note: In the data flow and control logic diagrams, the signals may have different names. For example, the 'busy' signal in the control logic corresponds to the 'shift' signal in the data flow.