Pipeline Processing

Book of John P. Hayes

pp: 275 - 283

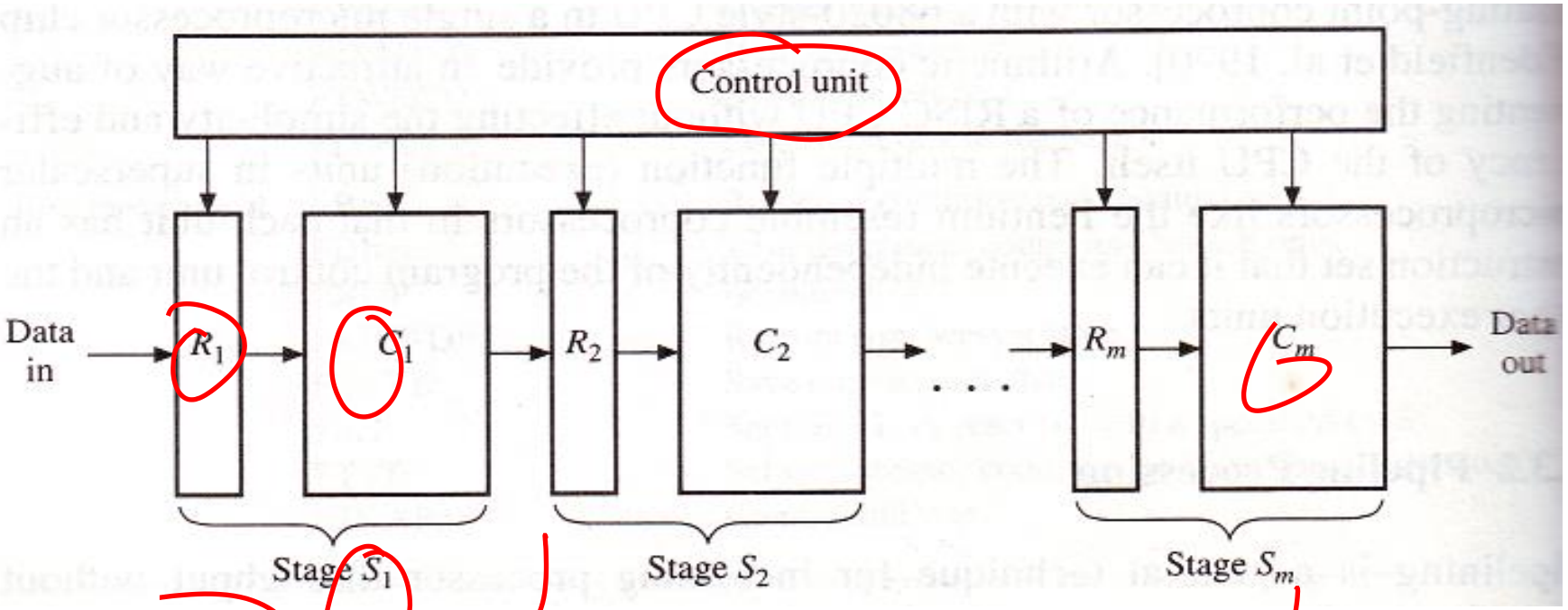
Pipeline Processing

- ✔ It is a general technique for increasing processor throughput without requiring large amount of extra hardware.
- ✔ Used in the design of the complex datapath units such as multiplier and floating-point adders.

Pipeline Processor

- ✔ It consists of a sequence of m-data processing circuits, called stages or segments, which collectively perform a single operation on a stream of data operands passing through them.
- ✔ Some processing is performed at each stage, but a final result is obtained only after an operand set has passed through the entire pipeline.
- ✔ In each clock cycle, every stage transfer its previous results to the next stage and computes a new set of results.

Structure of a Pipeline Processor



Pipeline Processor

- ✔ An m-stage pipeline can simultaneously process up to m independent sets of data operands.
- ✔ Suppose that each stage of the m-stage pipeline takes *T* seconds to perform its local sub-operation and store its results. Then *T* is pipeline's clock period.
- ✔ The ***delay or latency*** of the pipeline, is the time to complete a single operation, that is *mT*.
- ✔ Any operation that can be decomposed into a sequence of sub-operations of about the same complexity can be realized by a pipeline processor.

Example

- The addition of two normalized floating-point numbers x and y can be implemented by four step sequence:

  1. compare the exponents. 2. Align the mantissa. 3. Add the mantissa. 4. Normalize the result.

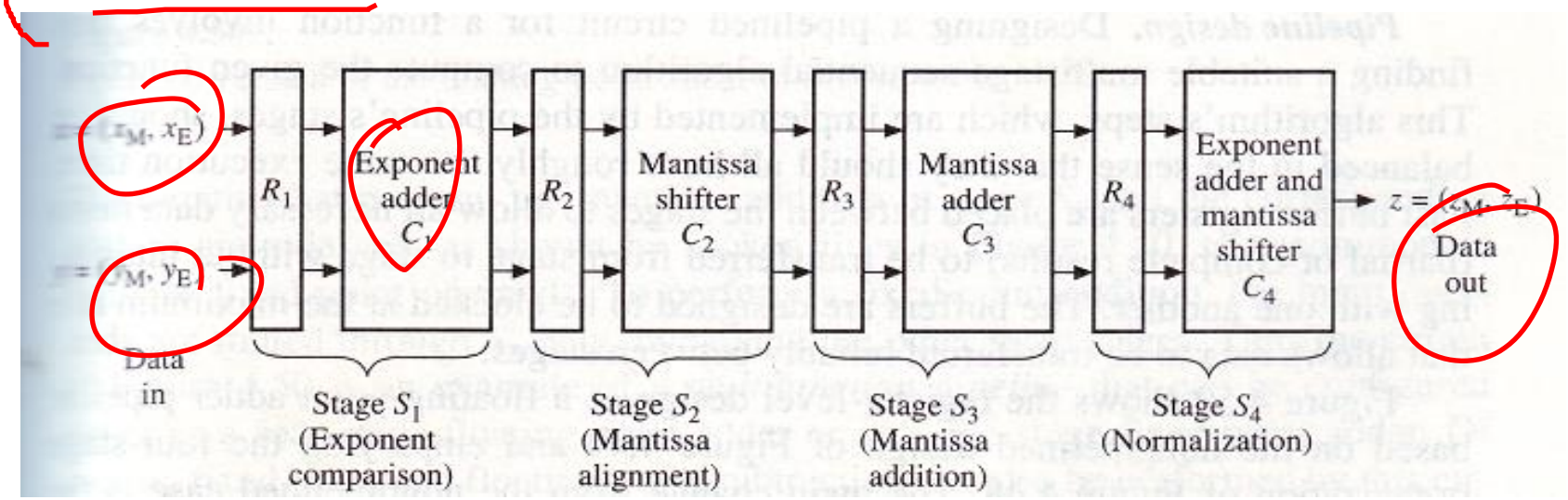
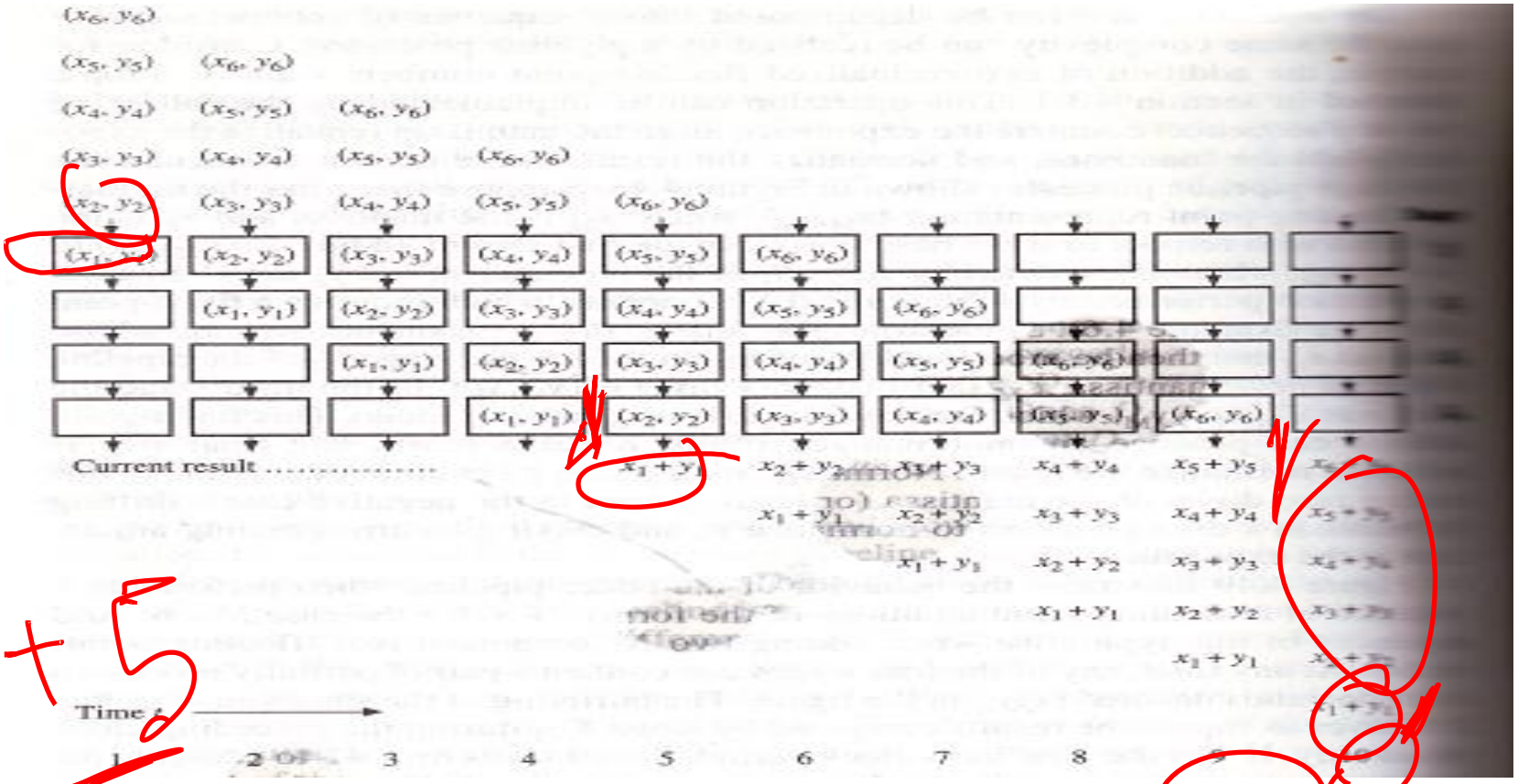


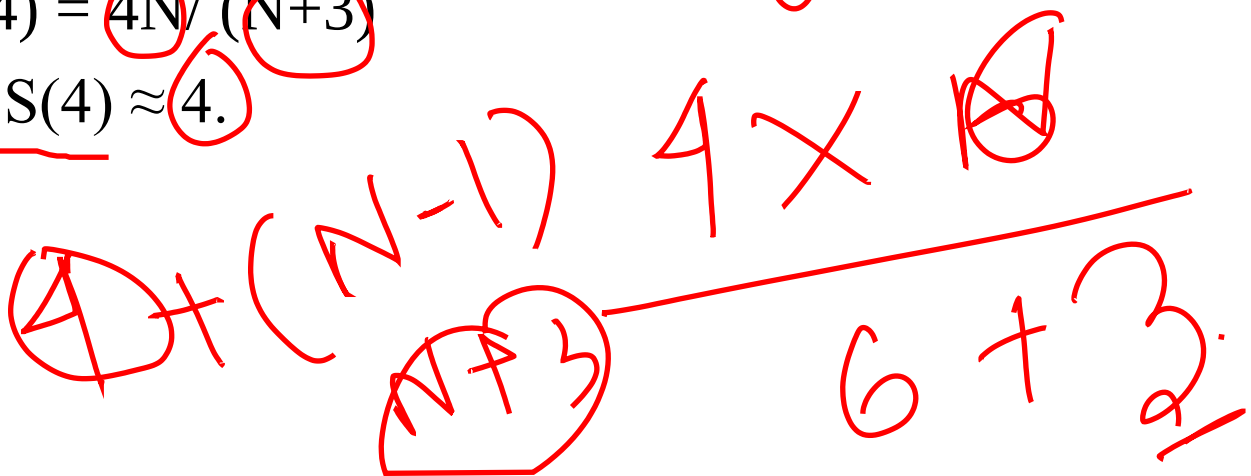
Figure: Four stage floating-point adder pipeline

Operation of the Four Stage Floating-point Adder Pipeline



Performance Analysis of Four Stage Floating-point Adder Pipeline

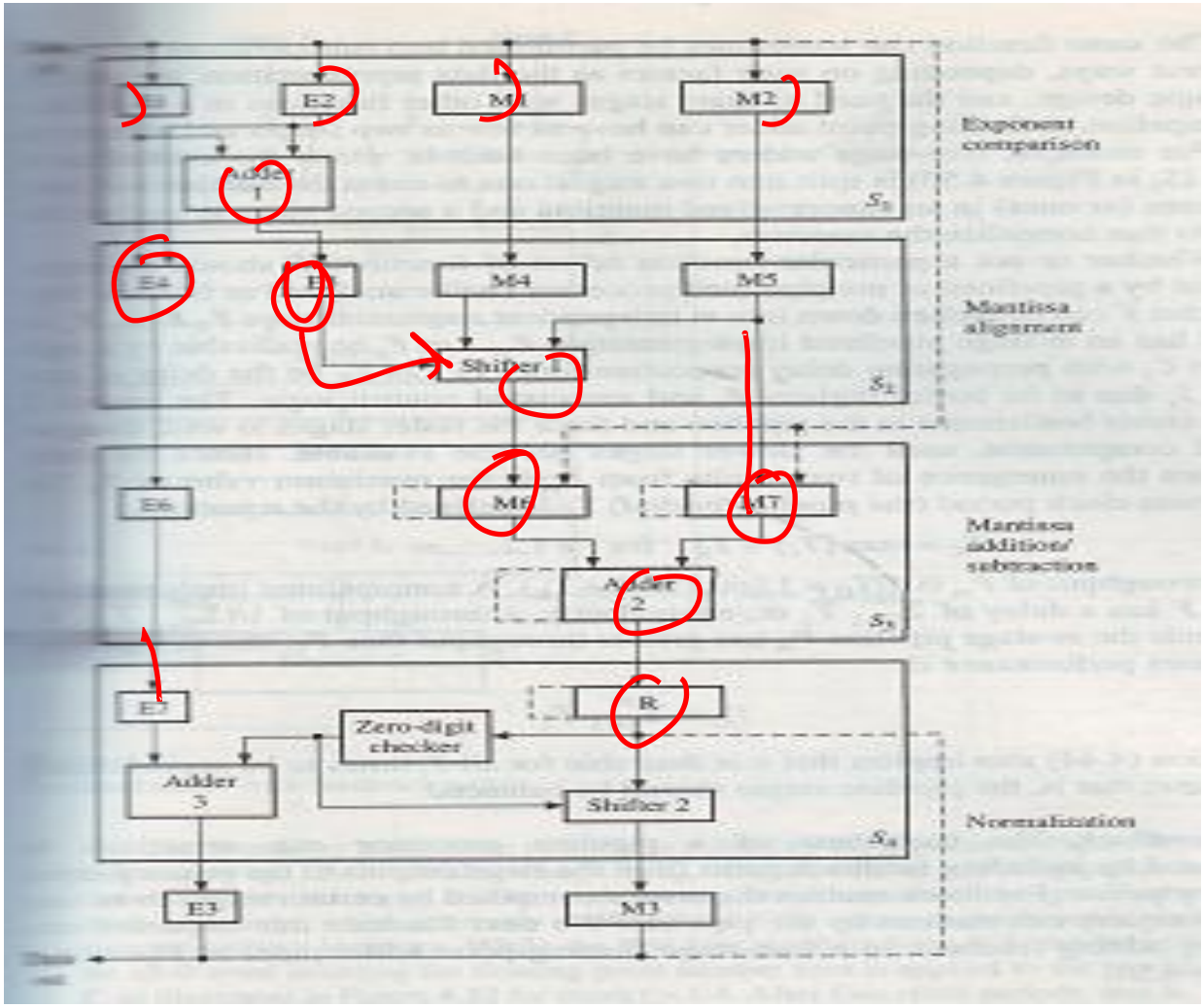
- ✔ If *T* is the pipeline's clock period, then the pipeline's delay is *4T*.
- ✔ *4T*= time required to do one floating-point addition using a non-pipelined processor plus the delay due to the buffer registers.
- ✔ *N* consecutive additions can be done in time *(N+3)T* using the pipelined processor.
- ✔ Speed up S(4) = 4N/ (N+3)
- ✔ For large N, S(4) ≈4.



Pipeline Design

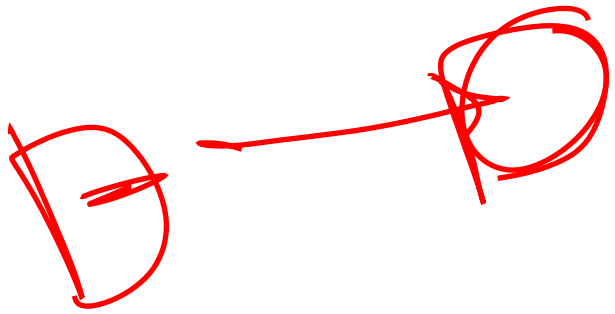
- ✔ Designing a pipelined circuit for a function involves first finding a suitable multistage sequential algorithm to compute the given function.
- ✔ This algorithm steps should be balanced.
- ✔ Fast buffers are placed between the stages.
- ✔ The buffers are designed to be clocked.

Pipelined Version of the Floating-point/ Fixed-point Adder



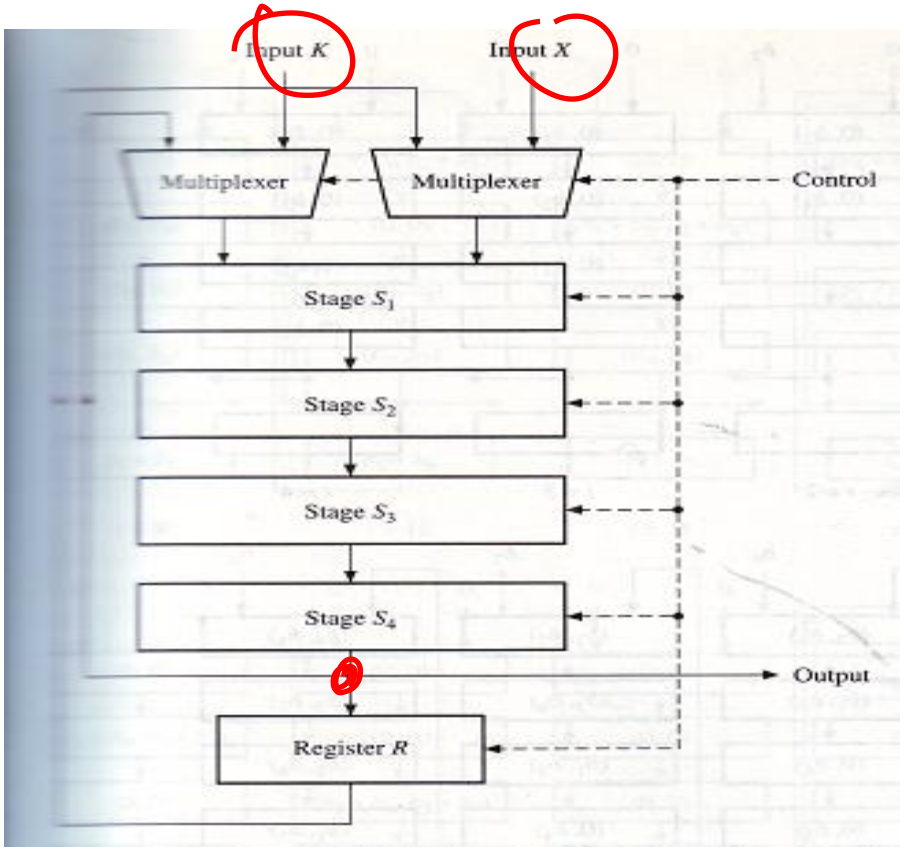
Feedback

- ✔ The usefulness of a pipeline processor can be enhanced by including feedback paths from the stage outputs to the primary inputs of the pipeline.
- ✔ It enables the results computed by certain stages to be used in subsequent calculations.

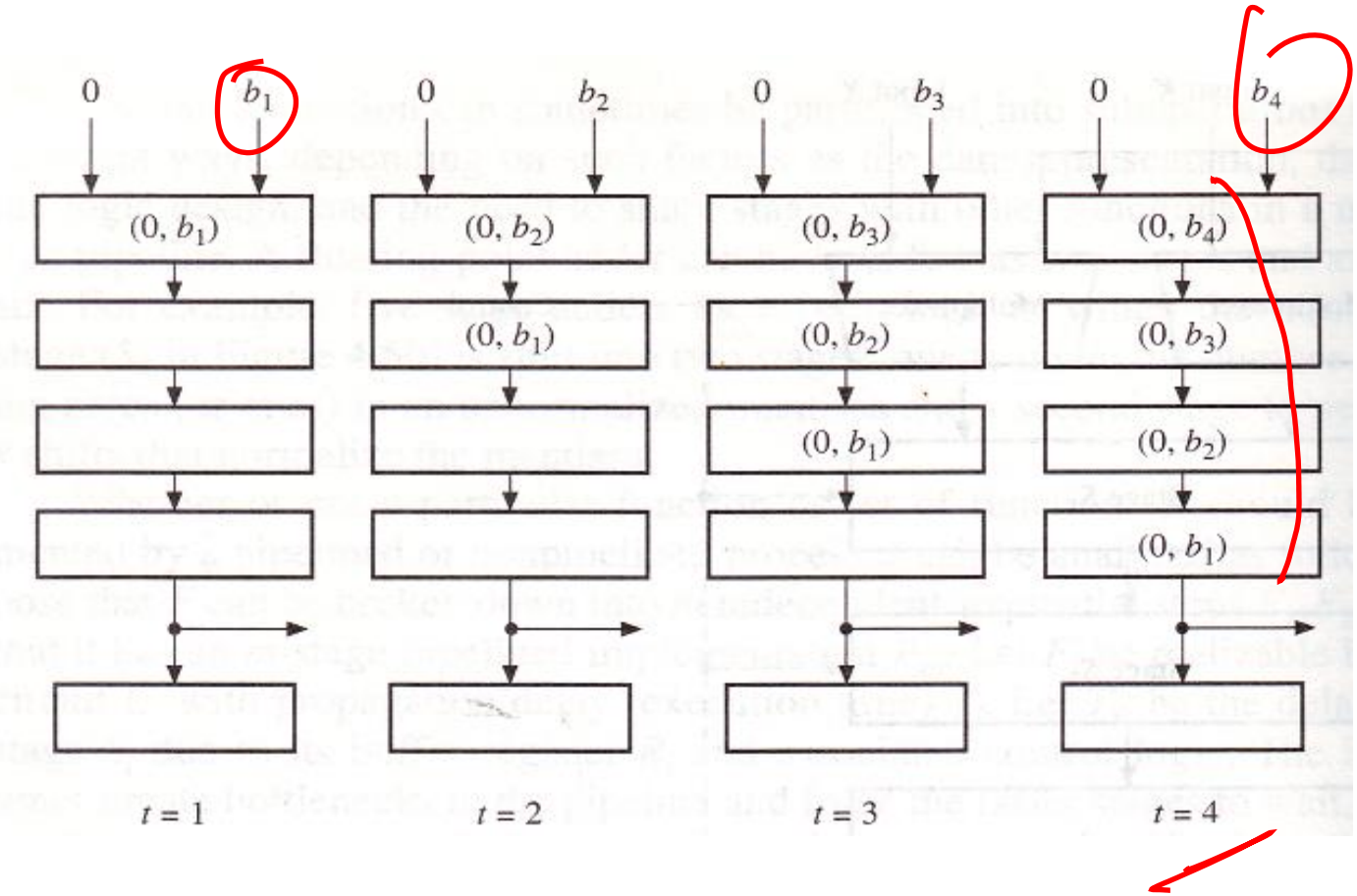


Summation by a Pipeline Processor

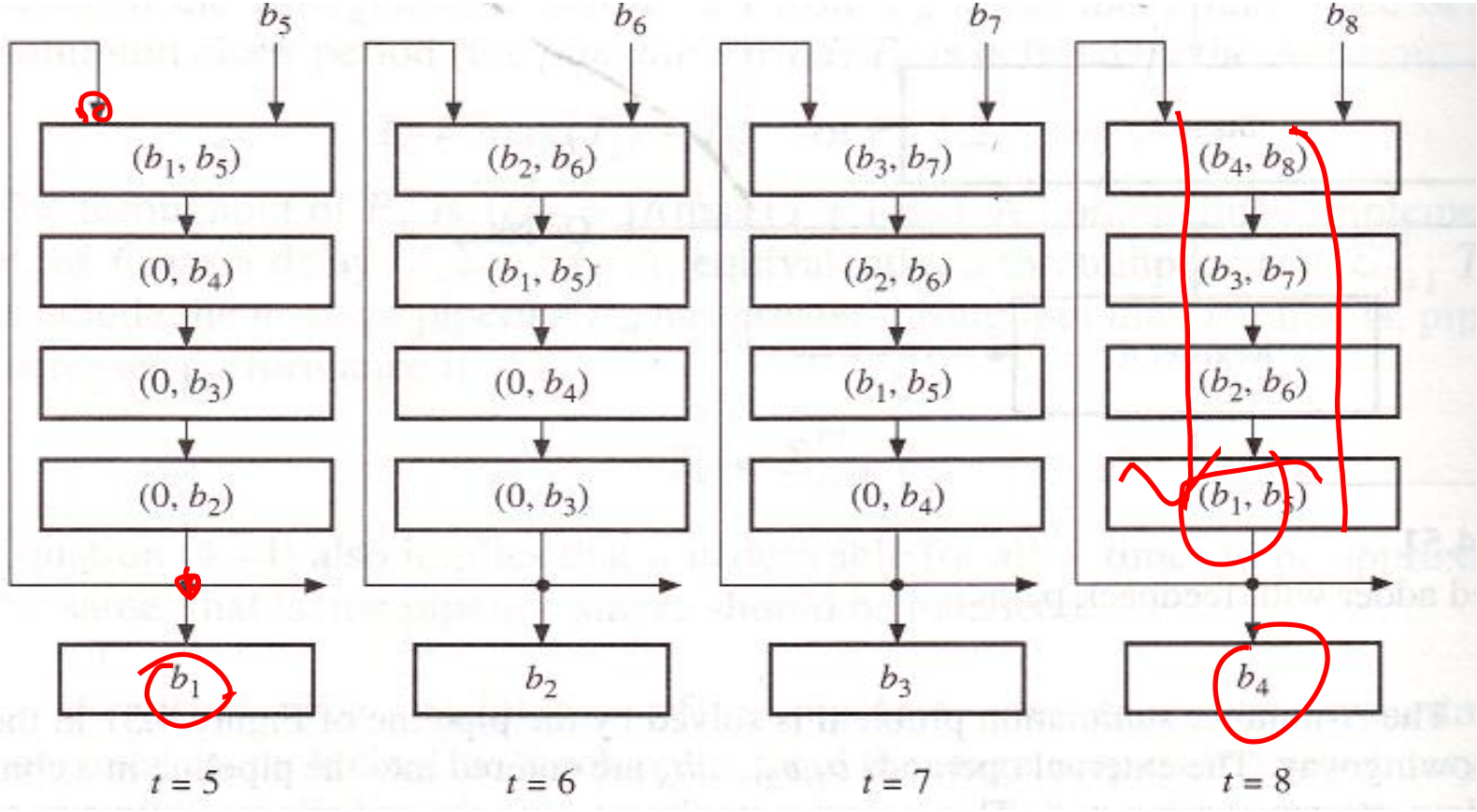
✔ Problem: compute the summation of 8 floating-point number b_1, b_2, b_3, b_4, b_5, b_6, b_7 and b_8.



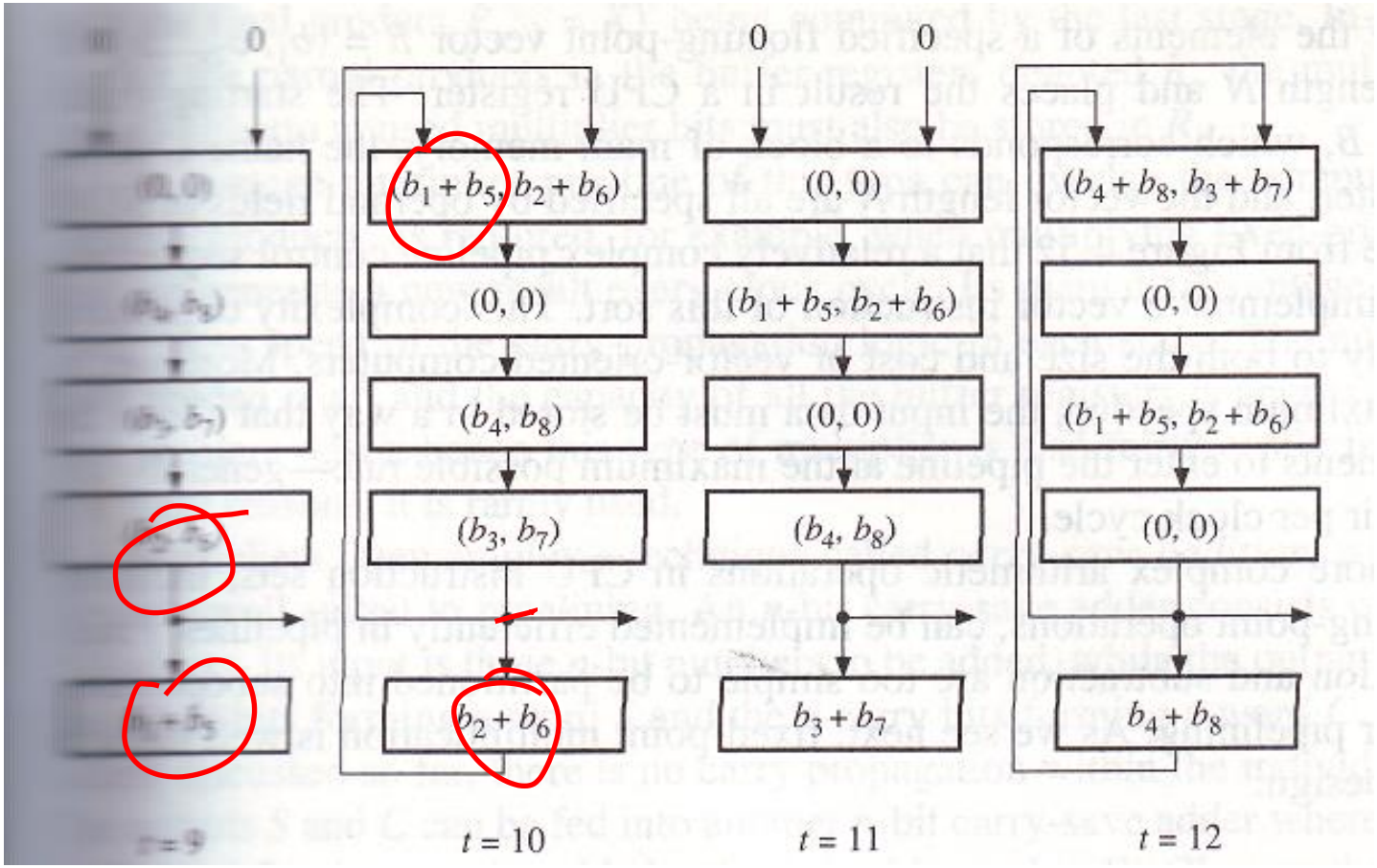
Summation by a Pipeline Processor



Summation by a Pipeline Processor



Summation by a Pipeline Processor



Summation by a Pipeline Processor