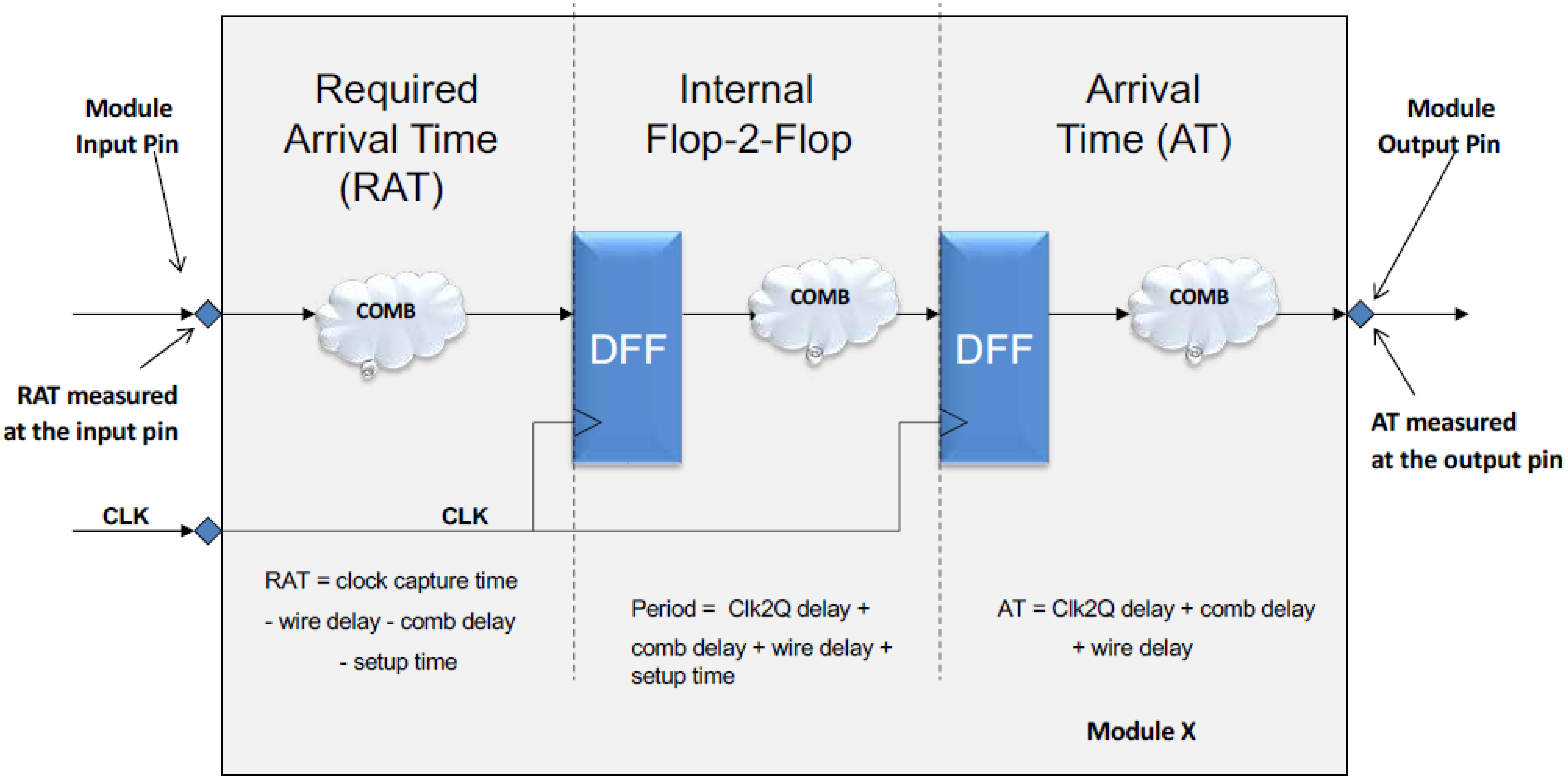
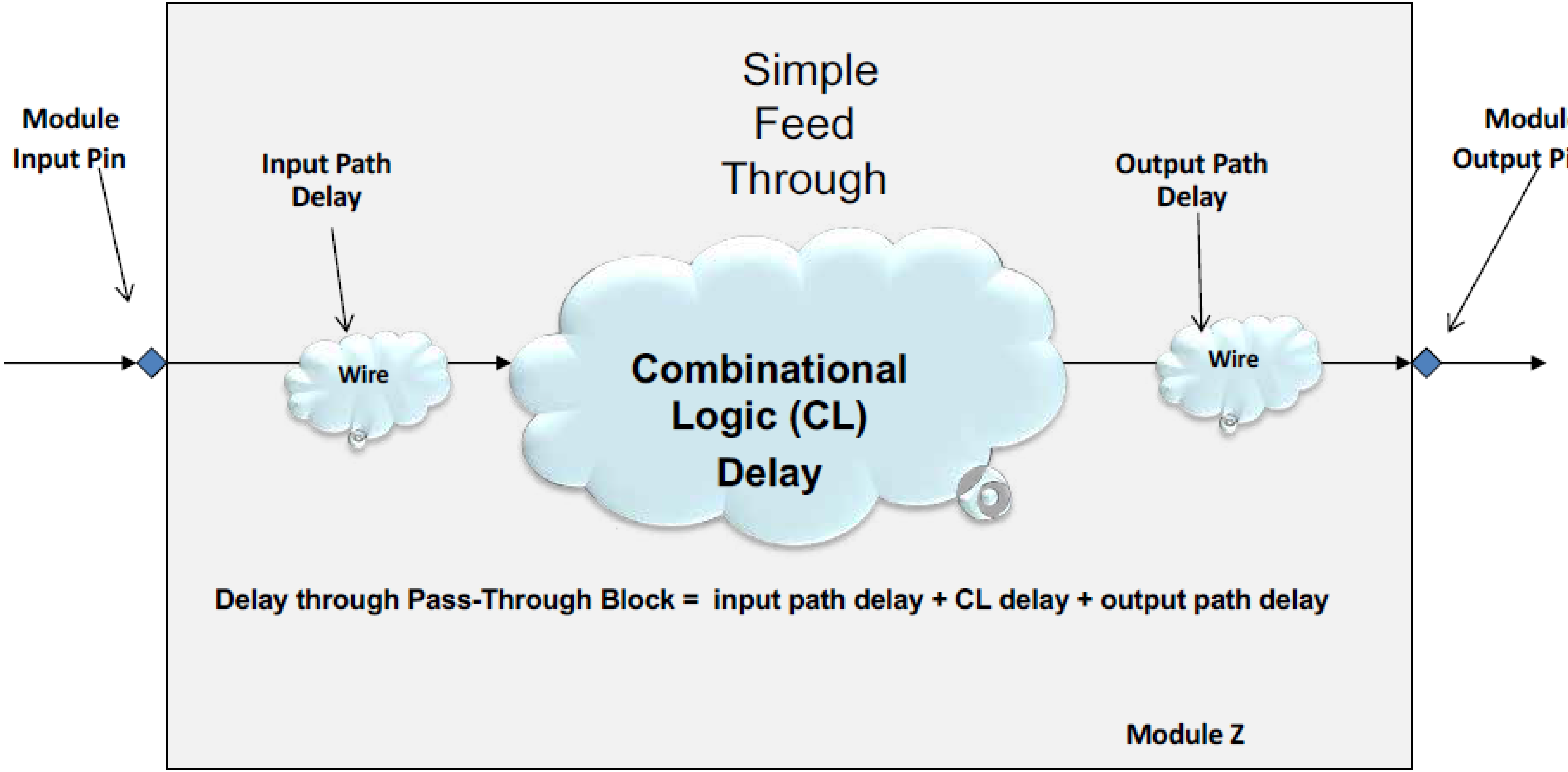
ASIC Timing

Basics of Timing

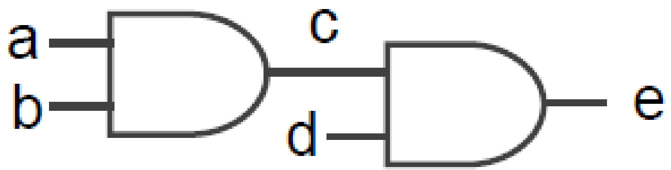


Basics of Timing: Pin-2-Pin (Pass-through)

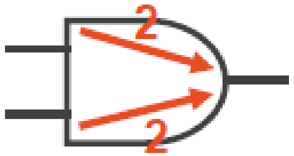


Simple path representation

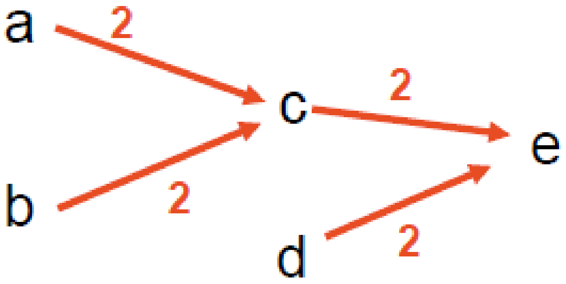
- **Consider a circuit**



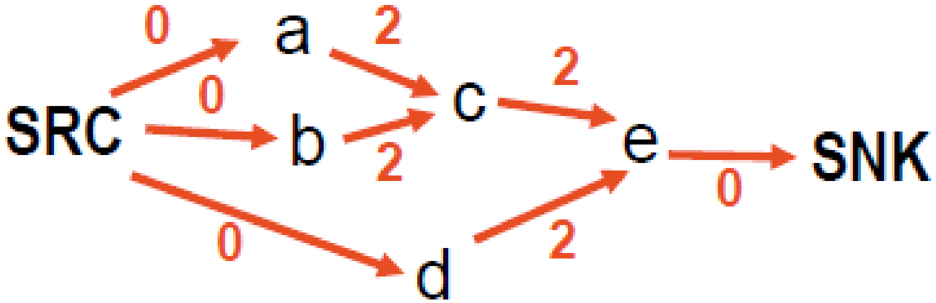
- Timing model of AND gate



- **Build a graph:**
  - Wires are Vertices - 1 per gate output and 1 for each PI and PO
  - Gates are Edges - input pin to output pin,1 edge per input with a delay for each edge



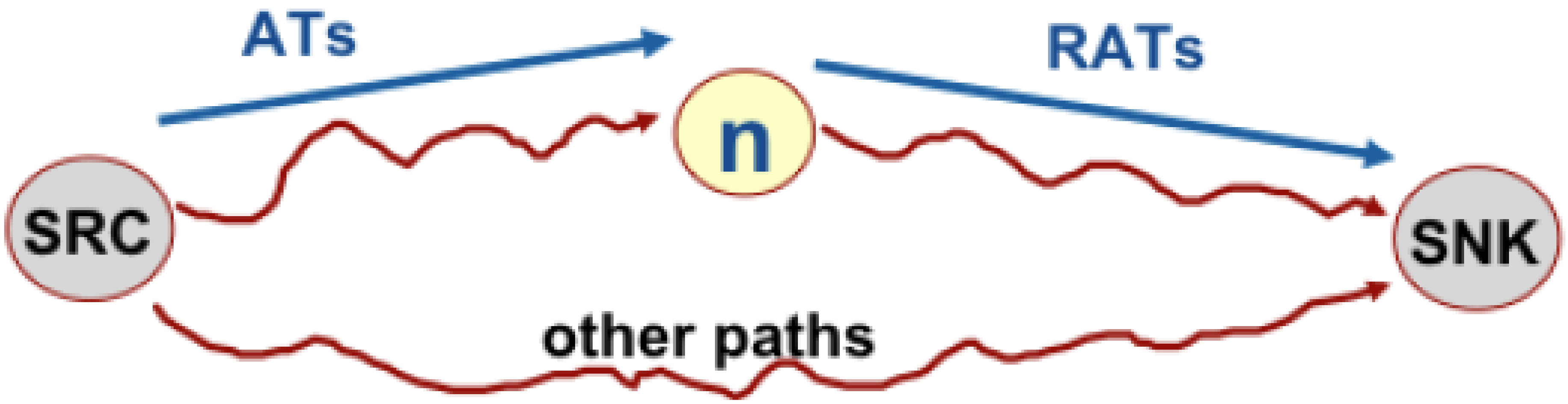
  - **Add Source/Sink Nodes:**
  - 0-weight edge to each PI and from each PO.



- All paths start and end at a single node
- Add interconnect delay – if available

Node Oriented Timing Analysis

- Enumerate every path, number of paths get exponentially bigger
- Instead, use node-oriented timing analysis
  - For each node, find the worst delay to the node along any path
  - Define two important values:
  - Arrival Time at a node (AT): the longest path from the source to the node
  - Required Arrival Time at node (RAT): the latest time the signal is allowed to leave the node to make it to the sink in time
- **Slack at node n is defined as:**
- Slack(n) = RAT(n) –AT(n)

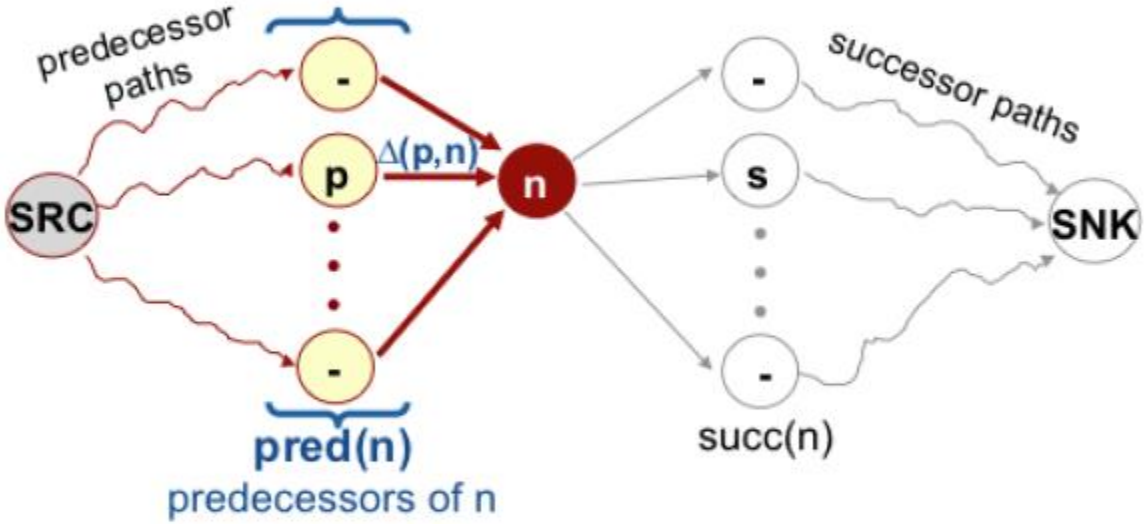


Compute ATs and RATs

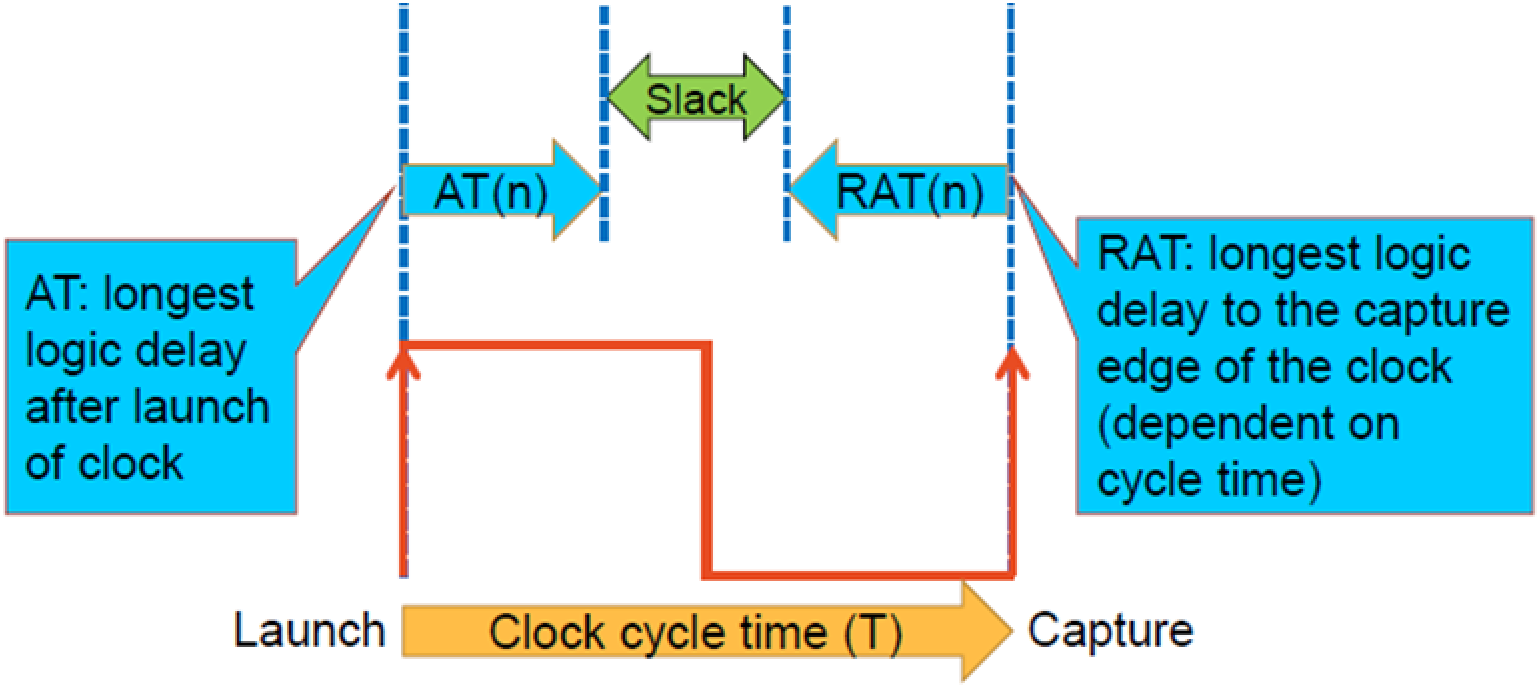
- Recursively
- **Arrival Time** at a node is just the maximum of the ATs at the predecessor nodes plus the delay from that node
- **Required Arrival** Time to a node is just the minimum of the RATs at the successor nodes minus the delay to that node

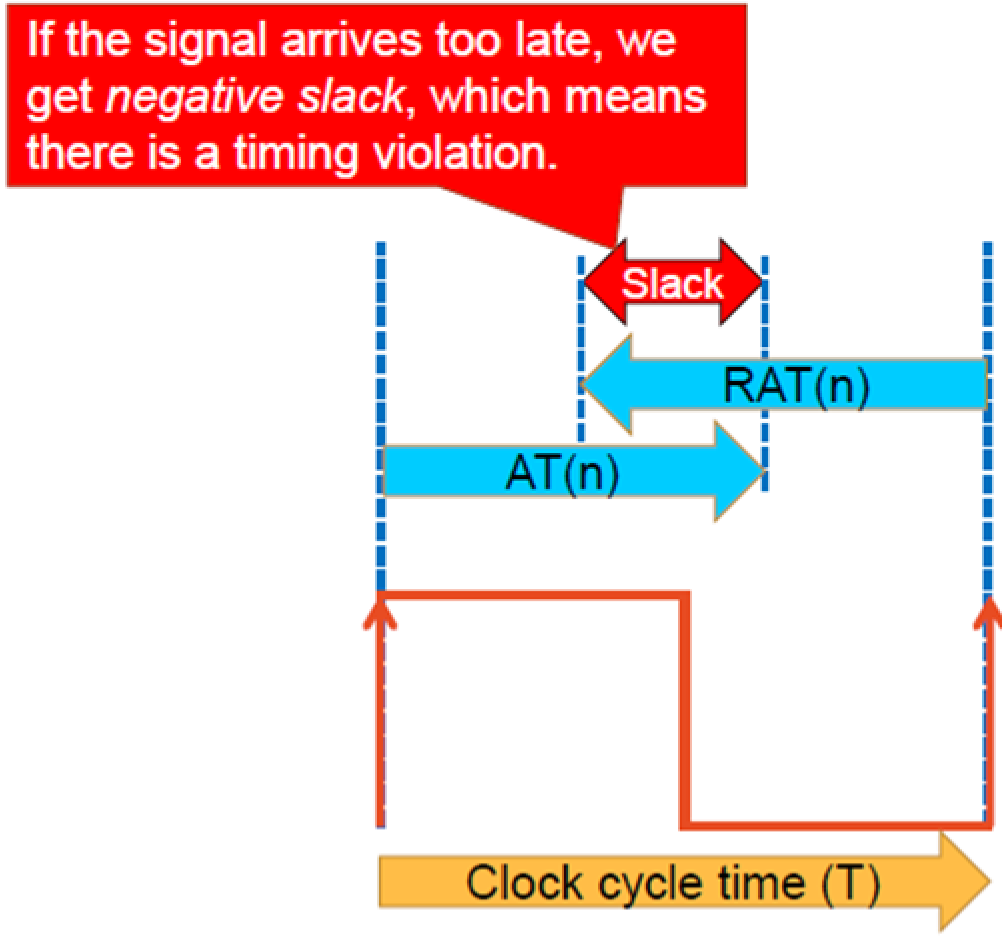
$$\mathrm{AT}(n) = \begin{cases} 0 & n = \mathrm{SRC} \\ \max\limits_{p \in \mathrm{pred}(n)} \left[\mathrm{AT}(p) + \Delta(p,n) \right] & n \neq \mathrm{SRC} \end{cases}$$

$$\mathrm{RAT}(n) = \begin{cases} T & n = \mathrm{SNK} \\ \max\limits_{s \in \mathrm{succ}(n)} \left[\mathrm{RAT}(s) + \Delta(n,s) \right] & n \neq \mathrm{SNK} \end{cases}$$

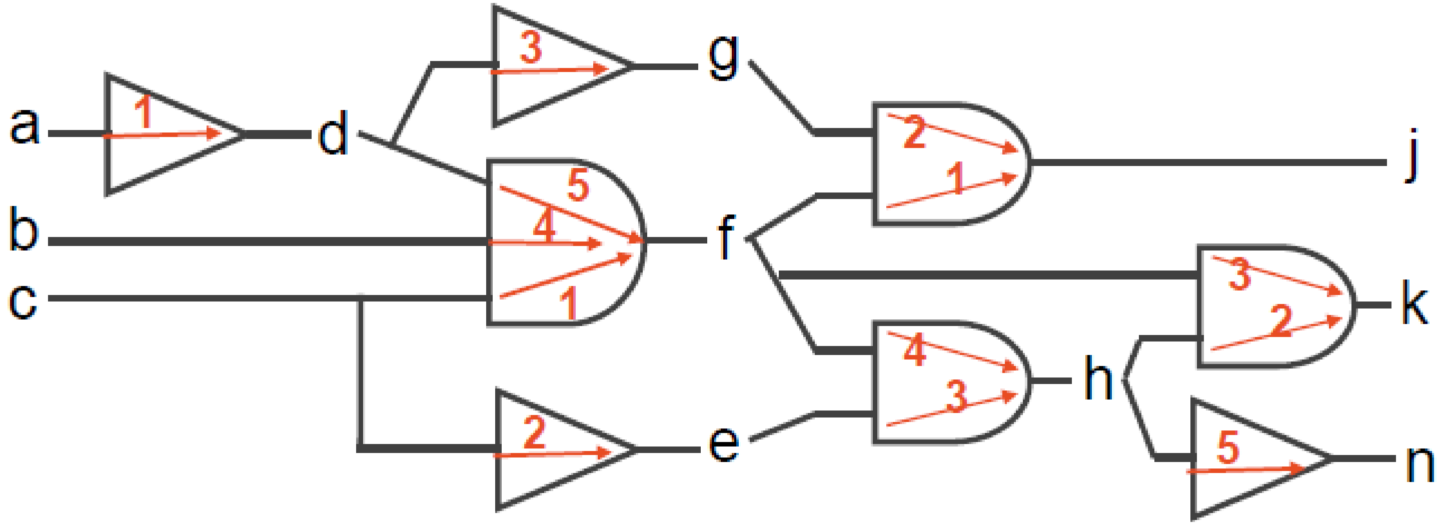


AT, RAT, and Slack





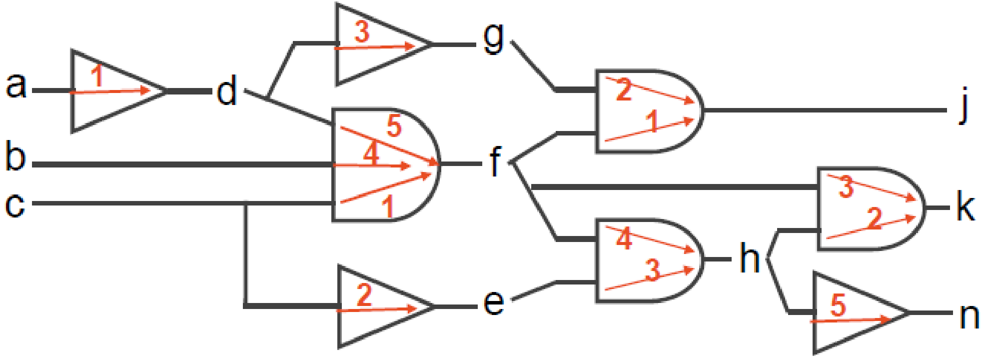
Example

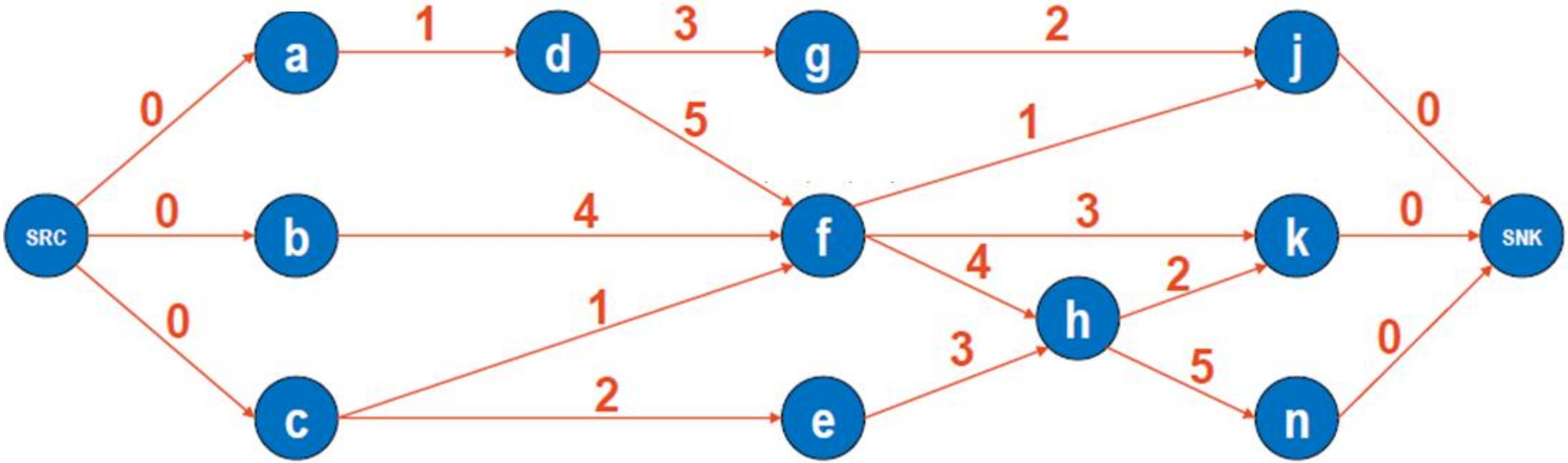


- Does it meet a cycle time of T=12?
- **Fill in the RAT, AT, and SLACK of each node**
  - Find out whether timing is met
  - Figure out what the worst path is

Example

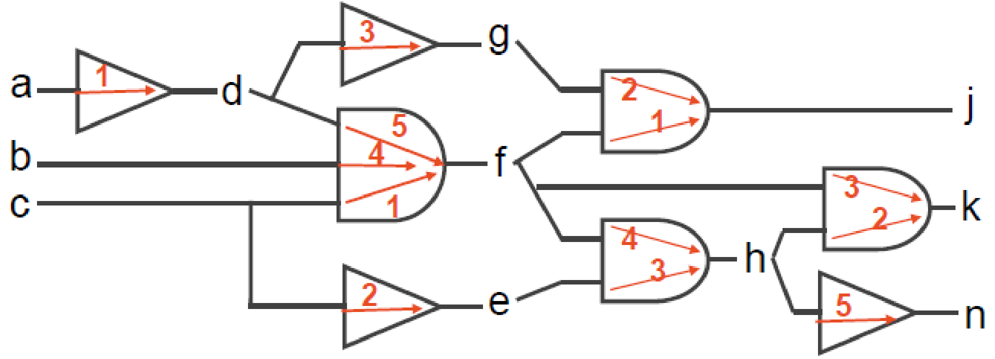
- Start by representing it as a directed acyclic graph (DAG)
- Compute ATs from SRC to SNK

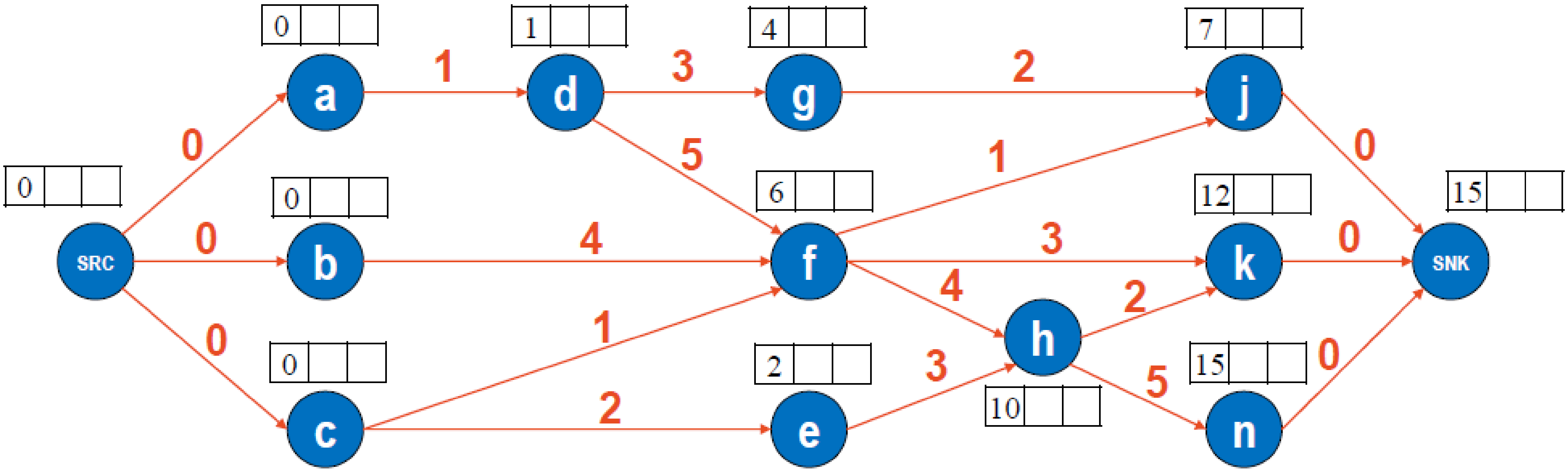




Example

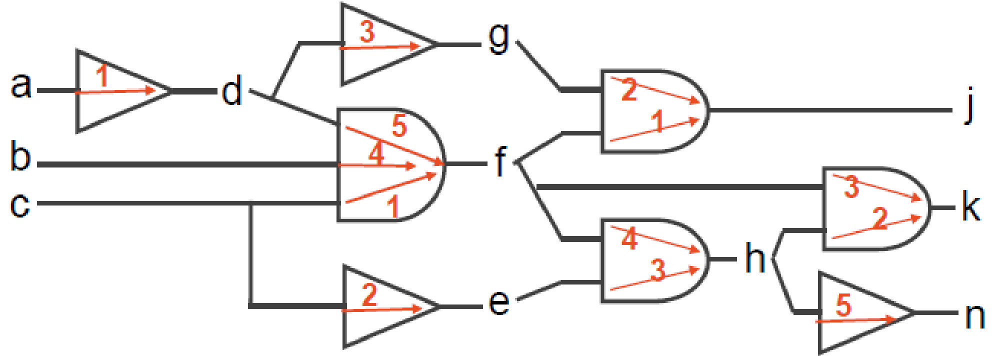
- Start by representing it as a directed acyclic graph (DAG)
- Compute ATs from SRC to SNK

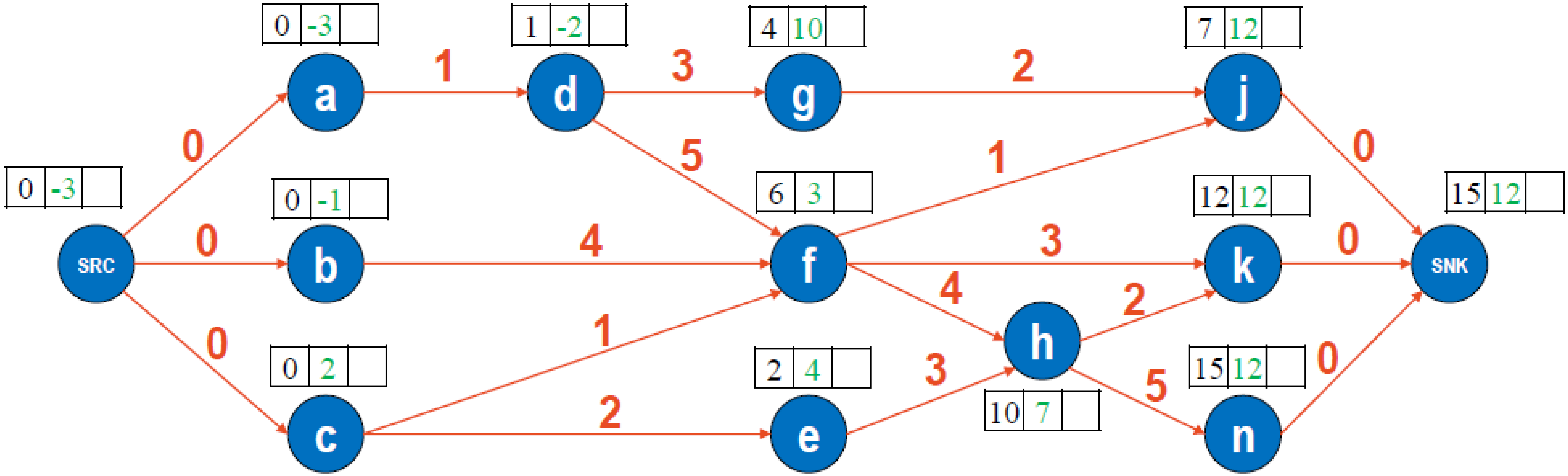




Example

- Add RAT to SNK from SRC





Example

- Calculate the slack - find the critical path