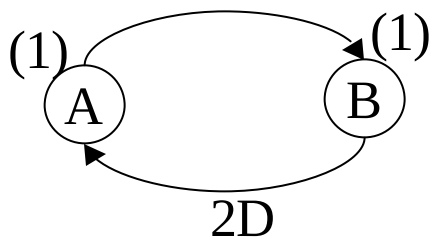
Chapter 5: Unfolding

Keshab K. Parhi

- Unfolding ≡ Parallel Processing



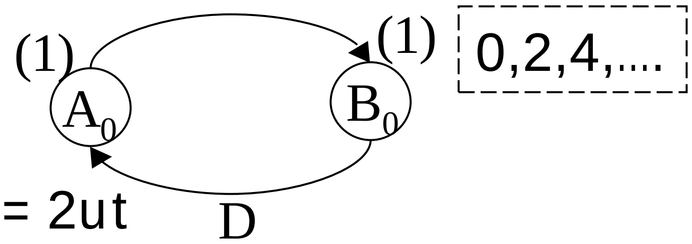
$A_0 \rightarrow B_0 \Rightarrow A_2 \rightarrow B_2 \Rightarrow A_4 \rightarrow B_4 \Rightarrow .....$

$A_1 \rightarrow B_1 \Rightarrow A_3 \rightarrow B_3 \Rightarrow A_5 \rightarrow B_5 \Rightarrow .....$

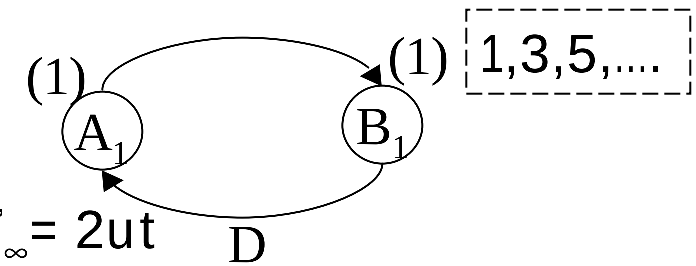
2 nodes & 2 edges

$T_\infty = (1+1)/2 = 1ut$

2-unfolded



$T'_\infty = 2ut$



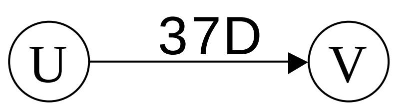
$T'_\infty = 2ut$

4 nodes & 4 edges

$T_\infty = 2/2 = 1ut$

- In a '*J*' unfolded system each delay is *J*-slow => if input to a delay element is the signal $x(kJ + m)$, the output is $x((k-1)J + m) = x(kJ + m - J)$.

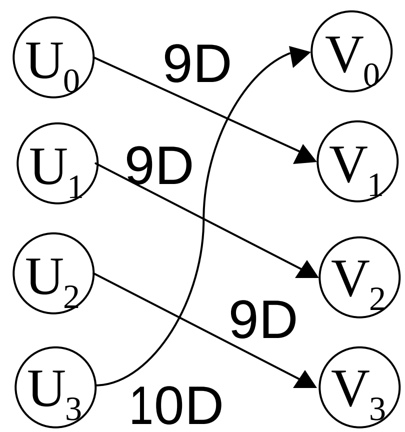
- Algorithm for unfolding:
  - For each node U in the original DFG, draw J node U_0 **,** U_1, U_2,..., U_{J-1}.
  - For each edge $U \rightarrow V$ with w delays in the original DFG, draw the J edges $U_i \rightarrow V_{(i+w)\%J}$ with $\lfloor (i+w)/J \rfloor$ delays for $i = 0, 1, \ldots, J-1$.



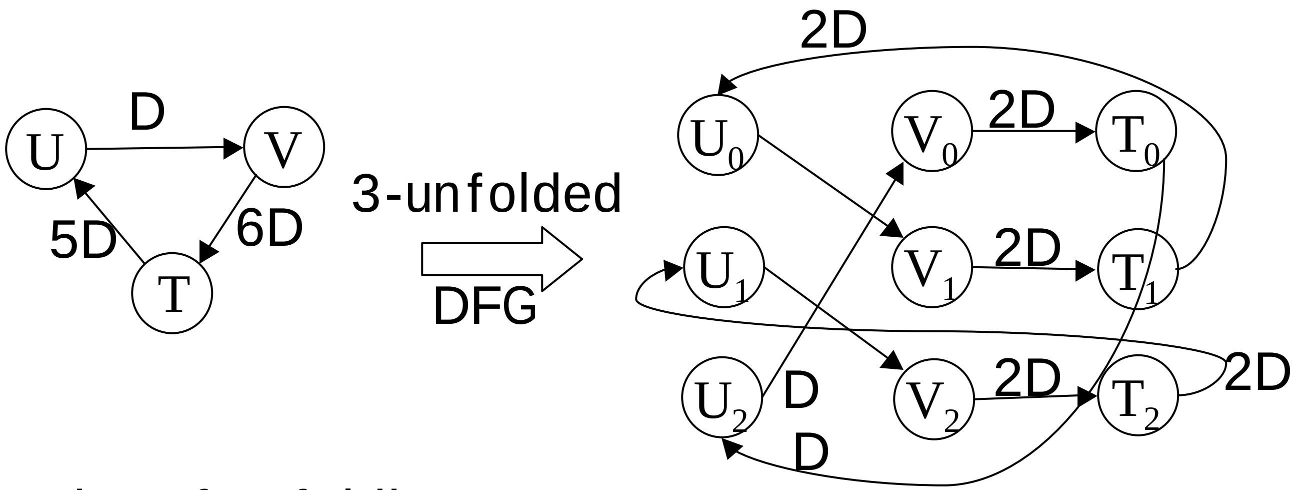
$w = 37$

$\Rightarrow \lfloor (i+w)/4 \rfloor = 9,\ i = 0,1,2$

$= 10,\ i = 3$



- Unfolding of an edge with w delays in the original DFG produces J-w edges with no delays and w edges with 1delay in J unfolded DFG for $w < J$.
- Unfolding preserves precedence constraints of a DSP program.



Properties of unfolding :

- *Unfolding preserves the number of delays in a DFG.*

  This can be stated as follows:

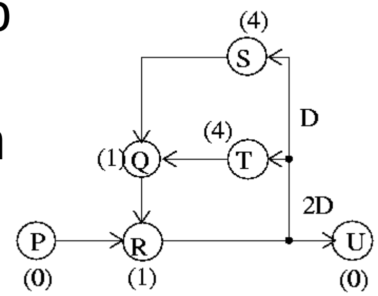
$$\lfloor w/J \rfloor + \lfloor (w+1)/J \rfloor + \ldots + \lfloor (w+J-1)/J \rfloor = w$$

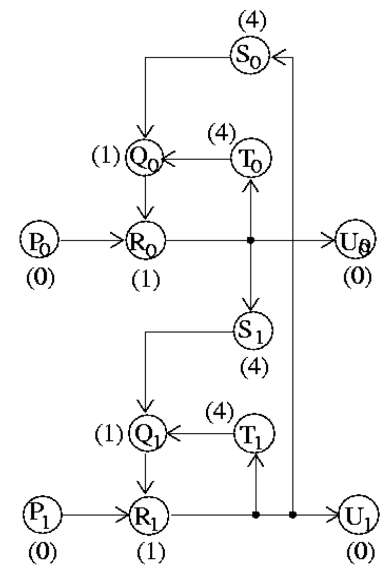
- *J-unfolding of a loop l with w_l delays in the original DFG leads to gcd(w_l, J) loops in the unfolded DFG, and each of these gcd(w_l, J) loops contains w_l/ gcd(w_l, J) delays and J/ gcd(w_l, J) copies of each node that appears in l.*
- *Unfolding a DFG with iteration bound T_∞ results in a J-unfolded DFG with iteration bound JT_∞ .*

- Applications of Unfolding
  - Sample Period Reduction
  - Parallel Processing
- Sample Period Reduction
  - Case 1 : A node in the DFG having computation time greater than T_∞.
  - Case 2 : Iteration bound is not an integer.
  - Case 3 : Longest node computation is larger than the iteration bound T_∞, and T_∞ is not an integer.

Case 1 :

- The original DFG cannot have sample period equal to the iteration bound because a node computation time is more than iteration bound

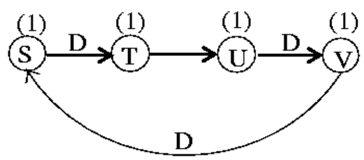


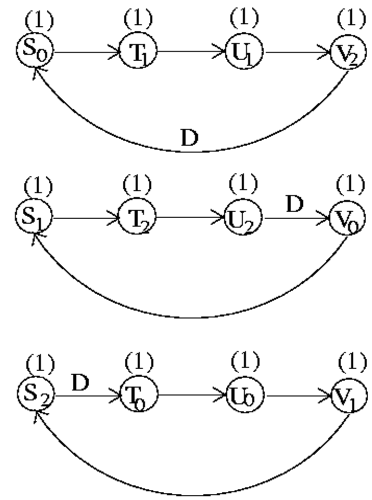


- If the computation time of a node 'U', t_u, is greater than the iteration bound T_∞, then $\lceil t_u/T_\infty \rceil$ - unfolding should be used.
- In the example, $t_u = 4$, and $T_\infty = 3$, so $\lceil 4/3 \rceil$ - unfolding i.e., 2-unfolding is used.

- **Case 2** :

➢The original DFG cannot have sample period equal to the iteration bound because the iteration bound is not an integer.



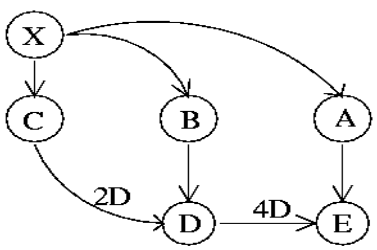


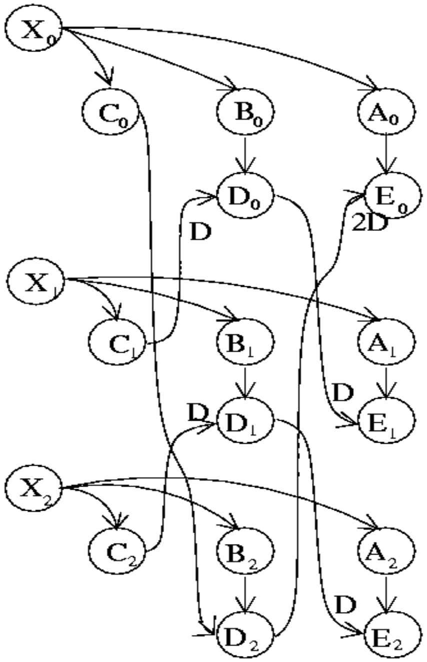
➢If a critical loop bound is of the form t_l/w_l where t_l and w_l are mutually co-prime, then w_l-unfolding should be used.

➢In the example $t_l = 60$ and $w_l = 45$, then t_l/w_l should be written as 4/3 and 3-unfolding should be used.

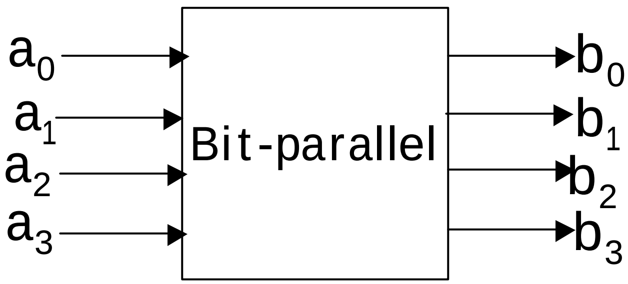
- Case 3 : In this case the minimum unfolding factor that allows the iteration period to equal the iteration bound is the min value of J such that JT_∞ is an integer and is greater than the longest node computation time.

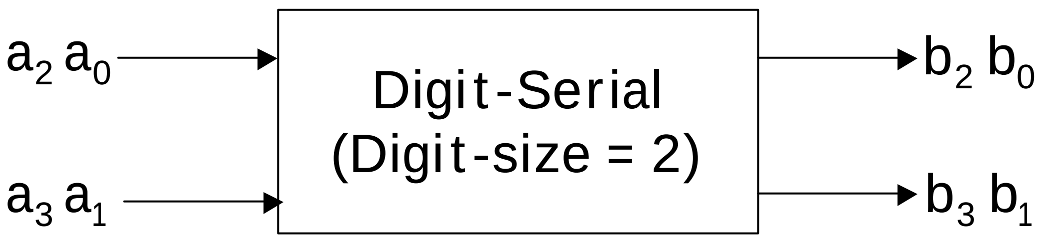
- Parallel Processing :
  - ➢ Word- Level Parallel Processing
  - ➢ Bit Level Parallel processing
    - ❖Bit-serial processing
    - ❖Bit-parallel processing
    - ❖Digit-serial processing

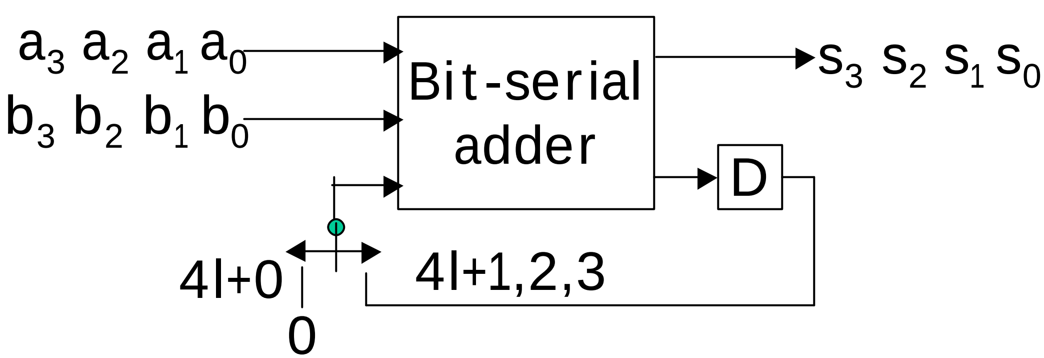




- Bit-Level Parallel Processing





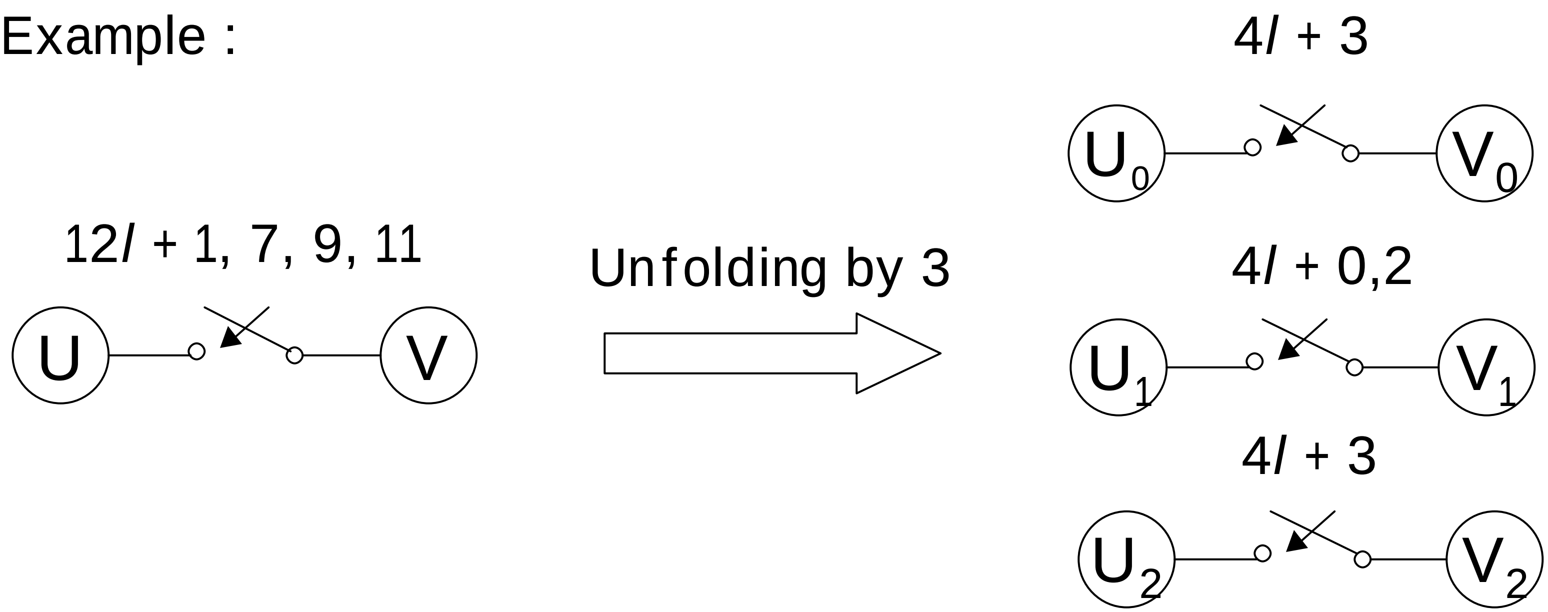


- The following assumptions are made when unfolding an edge $U \rightarrow V$:
  - The wordlength W is a multiple of the unfolding factor J, i.e. W = W'J.
  - All edges into and out of the switch have no delays.
- With the above two assumptions an edge $U \rightarrow V$ can be unfolded as follows :
  - Write the switching instance as

$$Wl + u = J(W'l + \lfloor u/J \rfloor) + (u\%J)$$

  - Draw an edge with no delays in the unfolded graph from the node $U_{u\%J}$ to the node $V_{u\%J}$, which is switched at time instance $(W'l + \lfloor u/J \rfloor)$.

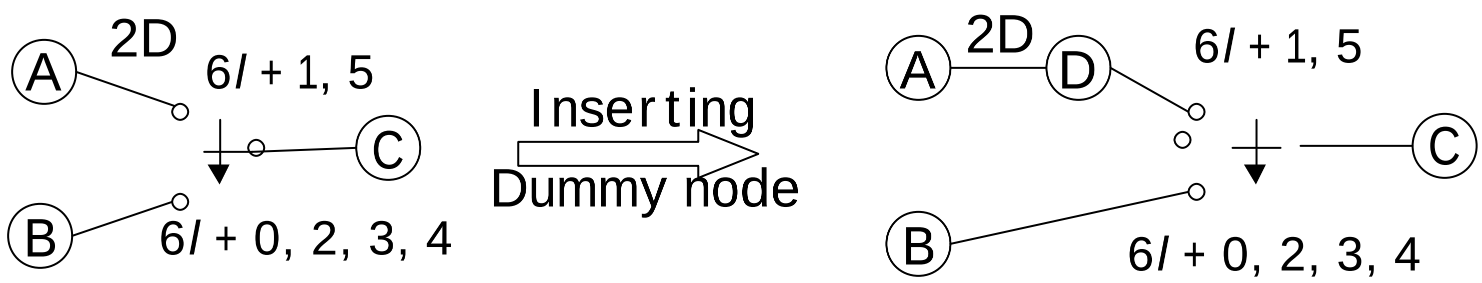
Example :

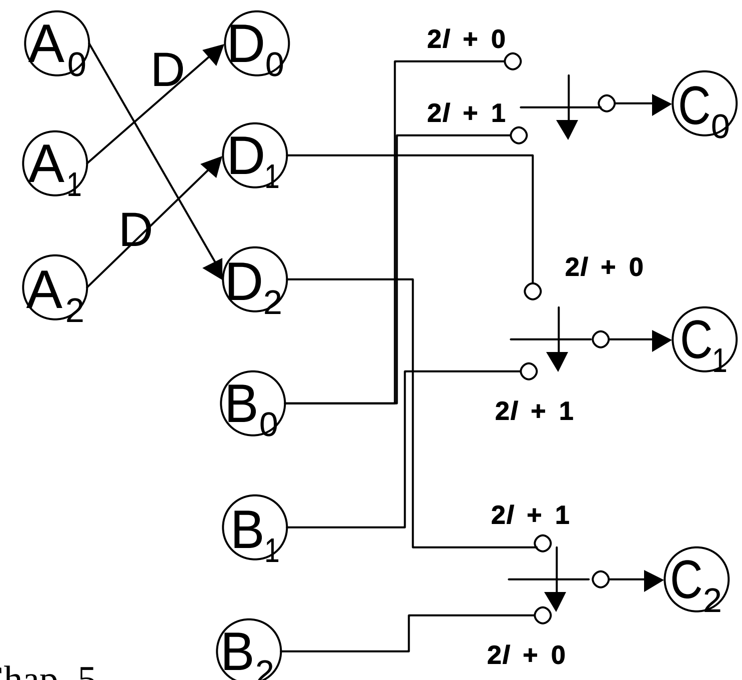


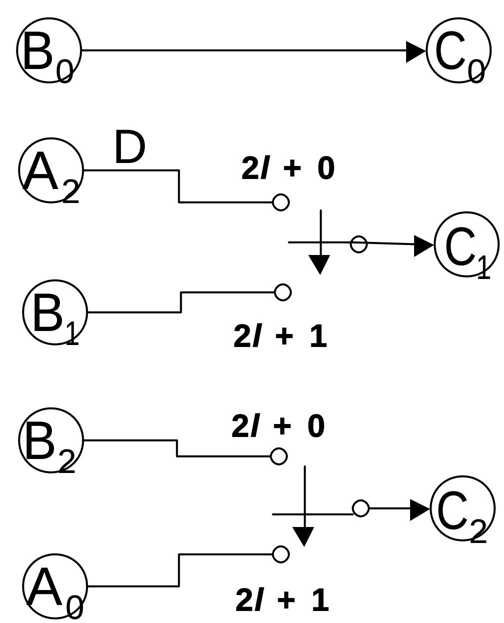
To unfold the DFG by J=3, the switching instances are as follows

$$12l + 1 = 3(4l + 0) + 1$$
$$12l + 7 = 3(4l + 2) + 1$$
$$12l + 9 = 3(4l + 3) + 0$$
$$12l + 11 = 3(4l + 3) + 2$$

- Unfolding a DFG containing an edge having a switch and a positive number of delays is done by introducing a dummy node.







- If the word-length, W, is not a multiple of the unfolding factor, J, then expand the switching instances with periodicity lcm(W,J)
- Example: Consider W=4, J=3. Then lcm(4,3) = 12. For this case, 4l = 12l + {0,4,8), 4l+1 = 12l + {1,5,9}, 4l+2 = 12l + {2,6,10}, 4l+3 = 12l + {3,7,11}. All new switching instances are now multiples of J=3.