ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

TIK Institut für
Technische Informatik und
Kommunikationsnetze

Prof. L. Thiele

Hardware/Software Codesign - HS 15

Exercise Sheet 2

Issue Date:	23. September 2015
Discussion Date:	30. September 2015

2.1 Model of Computation - Kahn Process Network

2.1.a)

Similar to the Fair Merge mentioned in the lecture, here we consider again a merge process that merges data tokens from two input channels into one output channel, as shown in Figure 1. In this case, the merge process can peek but at most one data token at a time from inputs, which means the process can check the arrival and the status of the most recent incoming token. There are two different versions for this One Peek Merge process, the semantics of which are shown in Algorithm 1 and Algorithm 2, respectively.
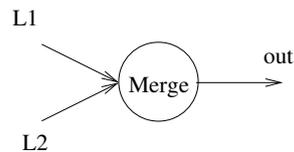

Figure 1: One Peek Merge.

Algorithm 1	Algorithm 2
loop	loop
X=Peek(L1)	X=Peek(L1)
Y=Peek(L2)	Y=Peek(L2)
if X$\neq \phi$ and $Y \neq \phi$ then	if X==ϕ or Y==ϕ then
out[X,Y], Del(L1), Del(L2)	no operation
else if X$\neq \phi$ and Y==ϕ then	else if X == Y then
out[X], Del(L1)	out[X,Y], Del(L1), Del(L2)
else if X==ϕ and Y$\neq \phi$ then	else if X < Y then
out[Y], Del(L2)	out[X], Del(L1), Del(L2)
else if $X == \phi$ and $Y == \phi$ then	else if X > Y then
no operation	out[Y], Del(L1), Del(L2)
end if	end if
end loop	end loop

Peek(X) returns the element at the head of the queue for channel X; or returns ϕ if the queue is empty. Del(X) deletes the element at the head of the queue for channel X. In Algorithm 2, we assume that tokens on channels L1 and L2 are integers.

- Examine the determinacy of these two algorithms. Prove or disprove your conclusion.
- Examine the fairness of these two algorithms.

2.1.b)

Draw a Kahn process network that can generate the sequence of quadratic numbers $n(n + 1)/2$. Use basic processes that add two numbers, multiply two numbers, or duplicate a number. You can also use initialization processes that generate a constant and then simply forward their input. Finally, you can use a sink process.

2.2 Model of Computation - Synchronous Data Flow

2.2.a)

Given the SDF graph in Figure 2.
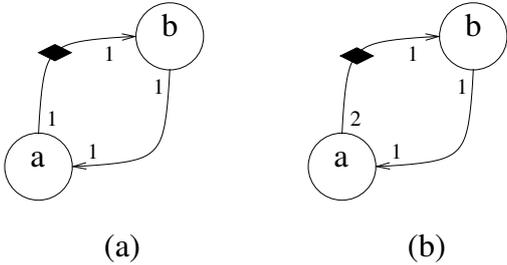

(a)   (b)

Figure 2

- Determine the topological matrix of these two SDF graphs.
- Are these two graphs consistent?
- If yes, determine the number of firings of each node, which leads for a periodic execution. How often each node must fire thereby at least?

2.2.b)

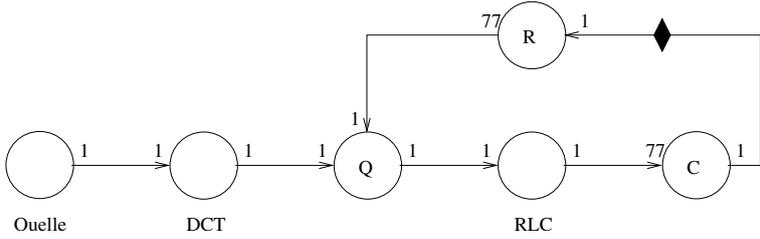Given the SDF graph in Figure 3

Figure 3

- Determine the topological matrix of this SDF graph.
- Examine the consistency.
- Determine the relative number of node firings, which leads for periodic execution at node firings.