

UC Berkeley computer science

Subject: Solutions to GasP Exercises from Oct 5, 2005
Date:
Authors: Mike Holenderski
Number:

References:

- [1] Igor Benko: "GASP Exercises", UCIB# 2005-ib05
- [2] Jo Ebergen: "Squaring the FIFO in GASP", ASYNC 2001
- [3] Ivan Sutherland and Scott Fairbanks: "GasP: A Minimal FIFO Control", ASYNC2001

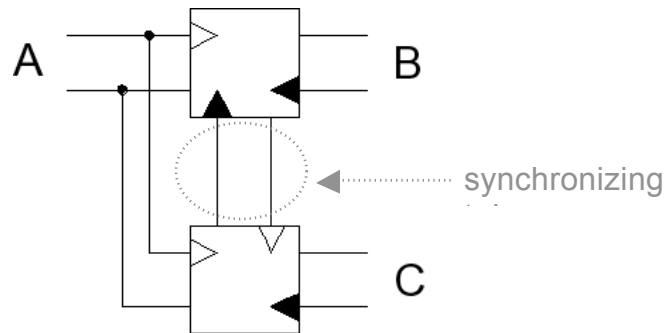
FIFO

We are asked to draw a GasP diagram for a control circuit for a FIFO SHIP with capacity 4.



ALTERNATOR

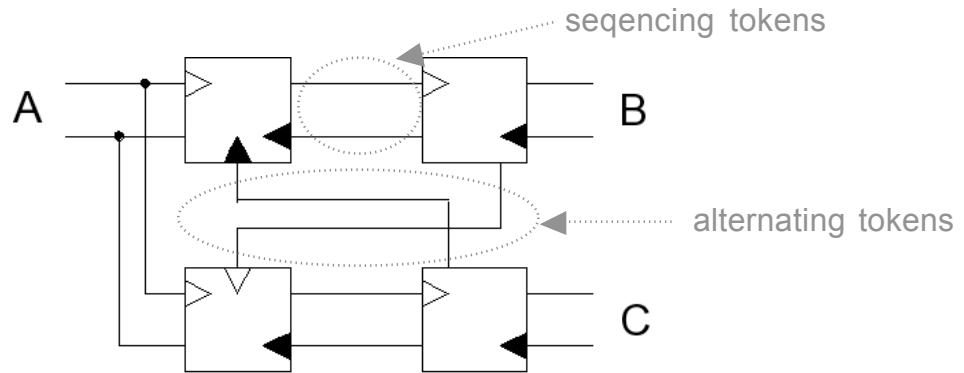
We are asked to draw a GasP circuit for an alternator SHIP. One solution is shown below.



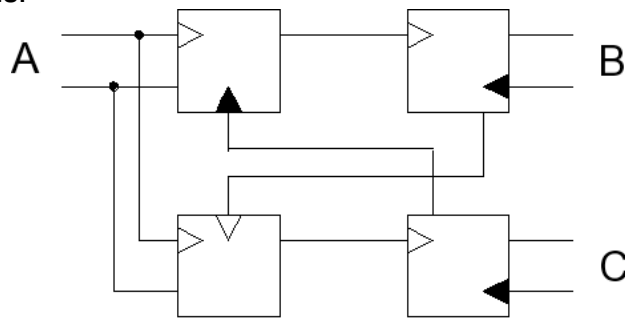
As you will notice, a race condition on the A input wire may occur. Assume inputs A, B and C are active. The top transition fires (as it is the only one allowed to fire) and sets all its outputs, including the synchronizing token. While doing this it clears the A input. However, it may happen that the synchronizing token arrives at the bottom transition before the A input is cleared, in which case the bottom transition will initiate firing (since the C input is also active). Since the top transition already set the A output, the A component might start clearing the A output wire while the lower transition is setting it, which will lead to a short circuit.

This document is a product of a collaboration between Sun Microsystems and the University of California at Berkeley. The ideas contained herein are freely available for any academic purpose.

One solution to this problem is to introduce a 1-stage delay as shown below.

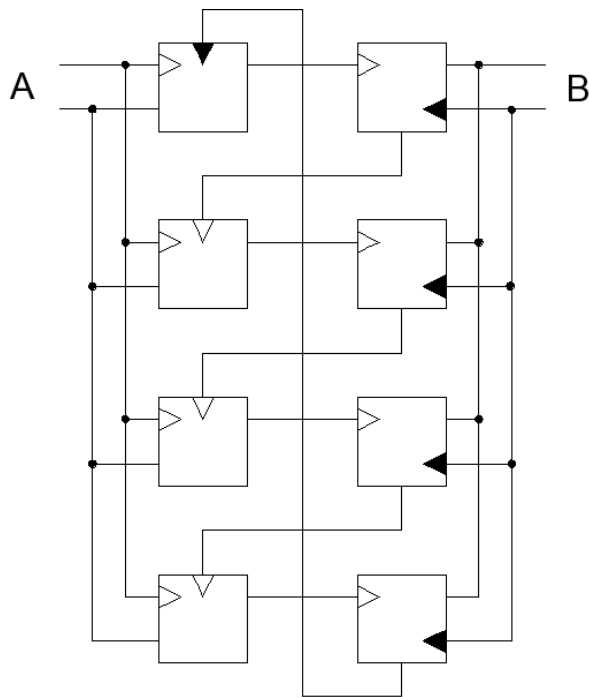


The 1-stage delay postpones the synchronizing token until the top transition has consumed the A input. Note that there are two sets of synchronizing tokens: one set sequences the two stages and one is responsible for alternating the B and C outputs. Since the “alternating” tokens suffice to sequence the stages, we can remove one output from the 1-stage delay transitions.



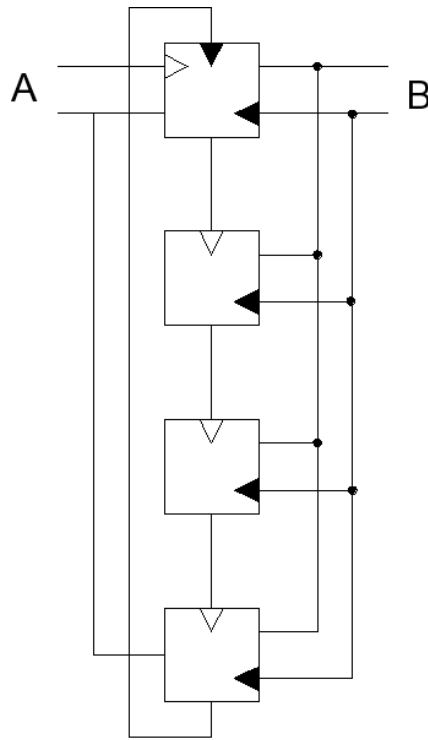
ROUND ROBIN

We are asked to draw a GasP circuit for a round robin alternator SHIP. The round robin is a generalization of the alternator depicted in the previous section.



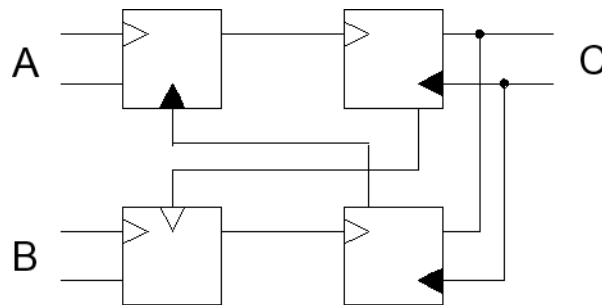
SPLITTER

We are asked to draw a GasP circuit for a splitter SHIP.



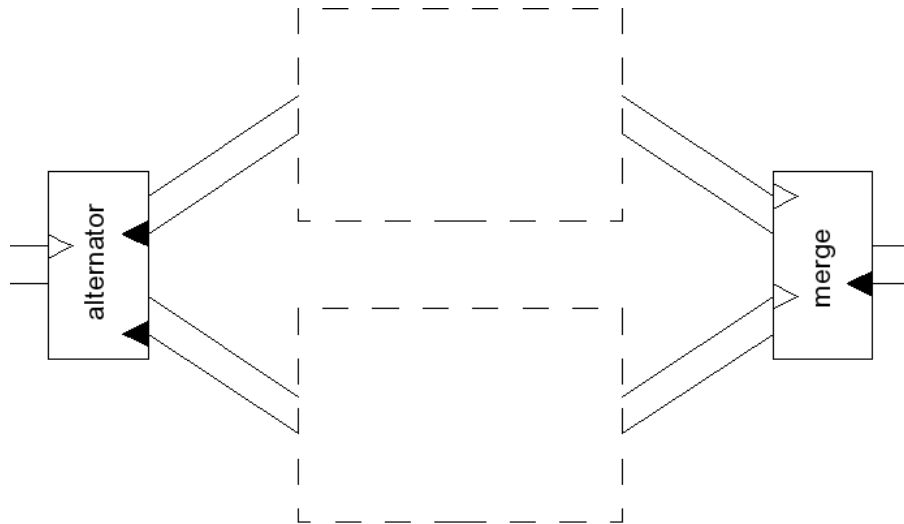
MERGE

We are asked to draw a GasP circuit for a merge SHIP. Merge is a mirror image of the alternator circuit shown above.

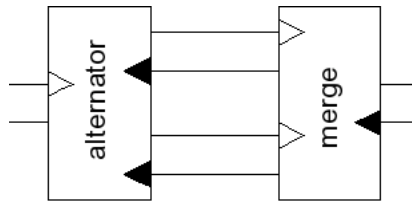


FIFO with logarithmic latency

We are asked to draw a GasP circuit for a FIFO SHIP with capacity N and logarithmic latency, using the block defined above. The logarithmic latency restriction suggests a binary tree implementation. The solution below uses an “alternator horn” for storing data in the FIFO and a “merge funnel” for retrieving the data from the FIFO.

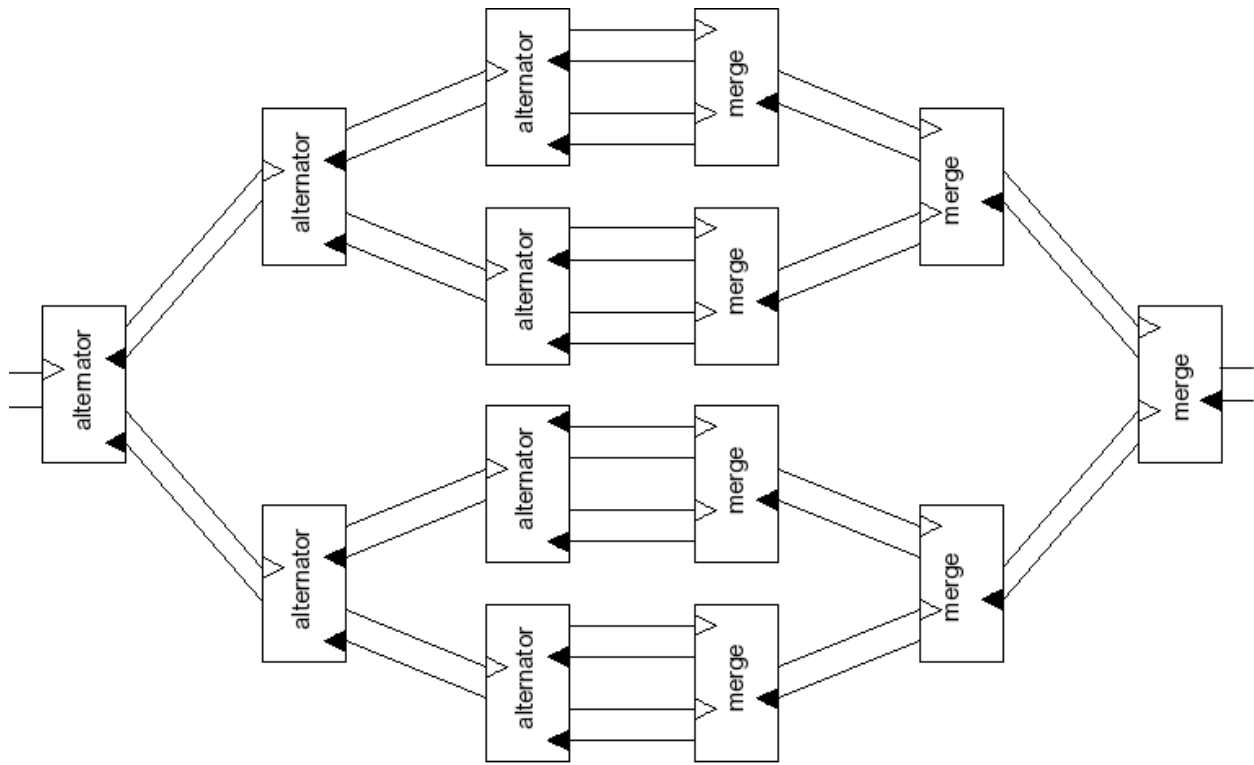


The dashed rectangle represents recursive composition of the circuit, with base element shown below.



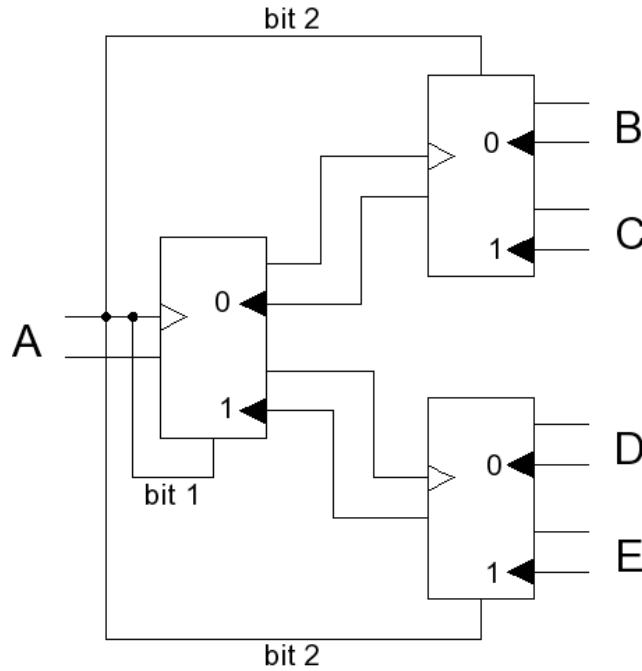
Note that the alternating behavior of both the alternator and the merge components guarantees the FIFO order, assuming correct initialization.

A FIFO with capacity $2 \leq N$ requires $\log(N) - 1$ iterations of the recursive step. An example of a logarithmic latency FIFO with capacity 8 is shown below.



DISTRIBUTOR

We are asked to draw a GasP circuit for a data distributor SHIP steering data into four destinations, using the DATA direct output component.



This SHIP is easily extendable to N destinations by attaching more DATA direct output components to the leaves of the circuit.