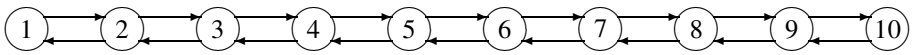
T-79.231 Parallel and Distributed Digital Systems
Homework 3, October 14, 2002 **Marko Mäkelä**

Designing a Telecommunication Protocol

Consider a system, in which devices are connected together by a bus, so that each device in the middle of the bus is connected to two neighbours, and the devices at the end of the chain are connected to a single neighbour.



The task is to design a protocol by which the devices attached to the bus are able to send messages to each other. Each device holds a single *message frame* in memory, which can be either empty, or contain the address of the receiver and the message data. A device may send a new message to the network only if its frame is empty.

If a device is the recipient of the message frame, it will read the message and replace the message frame with an empty frame. Otherwise it will try to move the frame to the left or the right along the bus, depending on the receiving address on the frame and the device's own address. The moving of a frame is carried out by switching frames between the current device and the one next to it. Notice that for the switching to take place the neighbour's message frame must be empty.

1. Design a High Level Net that implements the above protocol, and convert the net to MARIA language, when there are n devices on the bus. Find the shortest path to a deadlock. In what situations will the system deadlock? (Hint: Pick a small n to start with).

2. Change the protocol so that one empty frame will be named a 'hole'. A hole can not be replaced with a new message - remember that other empty frames normally can be replaced with a new message. Does the system deadlock now? How about if we allow the protocol to place a new message only in place of a hole?

3. Determine the state space sizes for the corrected, non-deadlocking protocol for different values of n. (Your reachability graph should have only one strongly connected component, otherwise there is a mistake.)

4. Examine whether your corrected protocol guarantees that a message sent to a recipient will eventually reach the recipient. Is it possible that a device will have to wait forever to send a next message? Show the temporal logic formulae you used in the verification. If the properties do not hold, show a counterexample and explain it.

5. How convinced are you that these properties will hold for large networks, i.e. for arbitrarily large n ? In your opinion, how useful/efficient is the protocol?

Return the answer to the mailbox located between rooms B 336 and B 337 in the Computer Science Building, 3rd floor, by 4 PM on November 18. You may also return a postscript or pdf version of your answer to Jukka.Honkola@hut.fi.