T-79.231 Parallel and Distributed Digital Systems
Homework Assignment 3, October 20, 2003 **Marko Mäkelä**

Verifying a Communication Protocol

Many data communication networks, such as the Ethernet, are based on contention for the communications medium. In CSMA-CD (carrier sense multiple access with collision detection), the sender may notice a competing transmission while sending and address the problem by repeating its own transmission after some amount of time. Thus, the message in transit may reach the recipient zero, one or many times.

In this assignment, you will verify some aspects of the alternating bit protocol when it is connected to different types of channels.

1. The file `abp-m.pn` contains a modular MARIA model of the alternating bit protocol. The transitions and places have been divided into modules, which are connected to each other via shared transitions. The modules of the channels are `msg_ch` and `ack_ch`. Add transitions to them for losing a message or an acknowledgement. Enclose your additions between `#ifdef LOSSY` and `#endif`. In this way, the channels will be lossy only if `maria` is invoked with the parameter `-DLOSSY`.

2. The receive transition of the message channel may leave the received message in the channel when `maria` is invoked with the parameter `-DDUPLICATING`. Modify the acknowledgement channel in similar way.

3. Generate the reachability graphs of your model with all four combinations of `-DLOSSY -DDUPLICATING`. Each graph should be strongly connected (`strong` should report one strongly connected component). If not, there is something wrong with your model. How many reachable states and enabled transitions are there in each case?

4. There are separate producer and consumer processes in the model. The parameter `-DTESTER` makes the producer produce sequences of messages according to the regular expression a^*b, that is, an arbitrary number of a followed by one b. Implement a check in the consumer process that alerts if a b is followed by anything else. This ensures that the consumer will receive each produced data item only once, even if the channels duplicate messages. Generate again four reachability graphs. How big are they?

5. Each of the reachability graphs of the previous question should contain the same number of deadlocks. How many did you get? Observe the shortest paths to the deadlocks of the case `-DTESTER -DLOSSY -DDUPLICATING` and explain what causes the deadlocks. For instance, you may draw a message sequence chart of the traffic between the producer, sender, receiver and the consumer.

Return the answer to the mailbox located between rooms B 336 and B 337 in the Computer Science Building, 3rd floor, by 8 p.m. on November 10, 2003. You may also return your answer in Postscript or PDF format to Jukka.Honkola@hut.fi.