LOTOS-based Verification of Modular Asynchronous Circuits

Michael Yoeli, Prof. Emeritus

Abstract

Let IMPL and SPEC denote LOTOS-processes describing the implementation and specification of an asynchronous circuit. Assume they have the same alphabet, which is partitioned into an input part and an output part.

We define: IMPL realizes SPEC (notation: IMPL |= SPEC) iff the following conditions are met.

1. (IMPL || SPEC) obs.equiv. SPEC.
   Here ‘||’ denotes full synchronization, except for the unobservable action ‘i’. In case both IMPL and SPEC are deterministic, cond(1) can be replaced by
   1a) L(SPEC) is_subset_of L(IMPL).
   Cond(1) ensures that IMPL is capable of performing any action sequence specified by SPEC, but may also be more powerful.

2. IMPL is free of livelocks.

3. IMPL does not produce any "undesirable" output. Namely, let w be a word in L(SPEC), and assume that w;z (where z is an output) is a word in L(IMPL) but not in L(SPEC). We consider this to be a case of an "undesirable" output.

Using CADP, the LOTOS-oriented toolbox developed at IMAG, Grenoble, the checking of conditions (1) and (2) is straightforward. On the other hand, cond(3) is more difficult to be checked. We now propose the following method. Let iIMPL be the extension of IMPL, obtained by replacing each output symbol, say z, by i;z. Then cond(3) is met iff (iIMPL || SPEC) contains no deadlock.

The paper also presents an alternative definition of IMPL|= SPEC, based on the theory of finite (non-deterministic) automata and provides a proof that the two definitions are equivalent. Yet another approach to defining the concept of realization, this time based on Petri nets, is the topic of the above TR CS0959.