The long term scope of SAVE

1. Participants

KTH: Axel Jantsch
LiTH: Zebo Peng
Saab: Ingemar Söderquist
FMV: Gunnar Ericson
Cybertech: Lars Ödman

2. Moderators

Ingemar Söderquist, Lars Ödman

3. Purpose

The task was to review and discuss concerns about the long term scope, objectives and feasibility of the SAVE approach.

4. Discussion

Three areas were brought up for discussion;

4.1 Cut verification cost!

The cost of verification is forecasted to be a dominant part of the total cost of electronic systems development in the near future. From the product developer's point of view, verification of function and performance of the final implementation is the essential objective.

In SAVE, the use of formally verified transformations are supposed to eliminate, or at least ease, the verification strain along the route of the design flow. Where in the design flow can we completely rely on the formal verification? Where are complementary methods necessary?

In the SAVE approach, the verification task has two separate sides:

- Verification of design transformations

- Verification of the design

Petru Eles has provided the following clarification:

As part of this project, the following main aspects concerning formal verification are handled:

- Verification of safety and liveness properties. In general, we verify that certain (desirable) states are reached and other (non-desirable) are not reached.

- Timing properties. We verify that certain timing requirements are satisfied.

These verifications are performed by model checking.

- The system also provides a set of semantic-preserving transformations to be used during the design process. The correctness of these transformations is verified by theorem proving, based on the semantics of the specification language (Haskell and respectively PRES).

By this methodology, verification time can be drastically reduced because:

- It is extremely difficult and time consuming to verify by simulation timing, safety, and liveness properties. By performing formal verification, this time can be very much reduced.

- By using previously checked design transformations, further verification during and after design, can be avoided or much reduced.

What is not handled in this project is formal verification of functional properties (the relation between input-output values). This is a particular research issue which is not part of this project. There are some solutions to the problem, based on theorem proving and language semantics, but they are very difficult to apply in practice. This means that functional correctness is to be verified by simulation. This is, however, not the most difficult and time consuming part of the process and there are quite mature techniques to do this.

4.2. Intellectual Property (IP)

The IP problem is addressed only partly within the research portion of SAVE. In order to make progress, the study of an example of IP for use within SAVE should be of value to further explore and reveal the real technical problems.

IP can appear in many forms. One conclusion was that IP primarily is a verification issue. Can Saab, in the procurement of IP on commercial terms, request that the supplier fulfills certain properties?

Another conclusion was that IP to a large extent is a business problem (legal conditions, support, etc.) and not a technical research issue. However, in order to clarify the problem, an explorative study of a real world example is recommended. In this case, well known or simple classes of IP, like microprocessor cores or memory blocks, should be avoided in favor of more challenging applications. Protocol interface units or filters should be appropriate IP targets.

4.3. The gap to implementation tools

At this stage of the SAVE project, the application of the SAVE methodology into practical engineering environments is of increasing interest and importance. It is, however, a common experience that a gap may exist between high level, systems engineering tools and tools for detailed implementation in HW and/or SW. Thus, is there a such a gap between the proposed ForSyDe flow and e.g. HW synthesis with VHDL?

The general view was that SAVE is aimed at exploring methods for high level formal design. If this approach is viable and successful, then practical solutions to bridge the gap could be worked out. This was considered a development issue, rather than research.