1450.4 meeting minutes – 10/22/09

Attendees: Jim O'Reilly, Ernie Wahl, Bruce Parnas, Markus Seuring
Not present: Ajay Khoche

Agenda:

- IEEE Meeting Preamble (No discussion of proprietary information)
- No meeting on 10/8 (cancelled by prior notice).
- No meeting last week on 10/15 (no quorum). Only Jim and Ernie were present. Discussed the following:
  - Content of status statement for IEEE TTTC.
  - Minutes from last week's meeting (10/01/09) are now available on the web.
- Discussion of spec/category/variable scoping proposal (see "Spec/Category/SpecVariable Handling Proposal" rev. 4 on web under "Docs for WG Meetings". This updated document includes pros and cons for opt. 1 only, opt. 2 only, and a user-selectable opt.1/opt. 2 combination). The proposal is to include opt. 2 only, but show usage guidelines that meet the constraints of opt. 1.
  - Unanimous voice vote in favor of option 2, with guidelines included in the text of the standard showing how to meet opt. 1 constraints
  - When referring to spec blocks (line 539), is spec_name optional? No, it required, but if it's already specified as part of the context, it does not need to be specified.
  - In the context of a category, the (const) attribute only modifies to the .meas subsection. All others (max, typ, min) are by definition const.
    - If no const attribute is specified, the .min, .typ, and .max fields are still const (i.e., they can only be changed by changing the spec block in which they're defined); the .meas field is in this case not const (i.e., it can be changed).
    - If the const attribute IS specified, then all fields (.min, .typ, .max, and .meas) are const, and cannot be modified. This is admittedly NOT a typical use – if there's no way to change the .meas field, then it cannot be filled in.
- Do we need a special STIL type for spec_var_name? (if so, it would include min, typ, max, meas, and perhaps units).
  - Consensus: Yes, a data type to hold a spec variable IS needed. A proposed form of the declaration is:

    SpecVariable VAR_NAME = *initial_value* { **Units** '*units_specifier*' }

    where initial_value applies to the .meas field.
    An alternative would be to include the initial value of the .meas field in the spec block definition (just as values are specified for min, typ, and max), and to have the declaration syntax as follows:

    SpecVariable VAR_NAME { **Units** '*units_specifier*' }

    In this case, the initial value of the .meas field is as specified in the spec block in which the spec being passed via this variable type was defined. The Units specifier is required in order to insure that we're passing in a spec with the correct type of units.
- Additionally, a variable type to hold a signal or list of signals or signal groups (i.e., a *sigref_expr*) is needed. We'll discuss that next week.
- Discussion of retest proposal
  - Did not get to this. Will discuss it next week, and/or via email.
- Other issues:
  - Issues:
    - Conversions from one data type to another – what are the rules? These need to be explicitly defined.
    - Limits as a specific data type?
      - How to specify an open limit on one end of a comparison?

        - Specify >=, >, <=, < - what does the syntax look like?
        - Limits x 0nS >= n <= 3nS
    - List of reserved words by tester type as an integral part of dot4? (should this be a dot3 addition?). Does NameChecks in dot3 cover this, or could it cover it with some extensions.
- Open issues - are there other open issues that should be considered? A review of the open issues list can guide us here.
    - http://spreadsheets.google.com/ccc?key=pEI1-gPUmt2ZTw_kcCTgnKw&inv=jim_oreilly@ieee.org&t=93304845348855187 1&guest).
    - If logged into your google account, can edit. If not, can only view
- Other?
- Next Meeting 10/29/09.

For reference STIL .4 information can be found at the IEEE STIL website: http://grouper.ieee.org/groups/1450/ (select the P1450.4 link from the table) or use the direct link http://grouper.ieee.org/groups/1450/dot4/index.html