1450.4 meeting minutes – 02/04/10

Attendees: Markus Seuring, Ernie Wahl, Jim O'Reilly, Oleg Erlich

Not present: Ajay Khoche, Bruce Parnas

Agenda:

- IEEE Meeting Preamble (No discussion of proprietary information).
- Discussion items
  - Discussion of operation of examples – remove spec block designation as Parameter, and add spec block as qualifier to the category name. (JO: I still need to think more about this – it seems that we could allow the prior form which specified both the Spec block name, and the Category name (without any spec block qualifier.)
  - Spec block, Category, Spec variable namespace resolution in Tests/Flows. (see GoogleDoc link below for examples).
    - Proposal: Namespace resolution precedence rules.
      - Local variables or Test/Flow Parameters of type SpecVariable
      - Spec variables in any spec block named in context (i.e. provided as a parameter, along with Category and Selector).
      - Global variables of type SpecVariable
    - Develop, if possible, rules for allowing dot0 compatibility (i.e., what happens if the Test or Flow doesn't specify a spec block name, and all spec/category blocks are developed according to dot0 rules? Do we want to allow resolution based solely on Category/Selector specification - assuming that, as per dot0, all category+variable names are unique. I need to think this one through, but thought I'd put it on the table. As I was thinking through the rules above, it occurred to me that we *might* be able to make this one work also.
    - Discussion:
      - Discussed and developed example code per namespace resolution examples document (see link below). Continuing to work on this.
      - Discussed what levels of scoping are allowed – i.e., if a Test has a parameter of Spec (a spec block), can one provide a Spec block name, then a Category name (without further qualification by the spec block name) to establish the <spec_block_name>.<category_name> scoping for any spec variables? This would be in addition to providing
      - Ernie asked about including some syntax which would give access to a spec variable used to specify the period of a waveform table in a timing block. The idea is that, if the period is set by a spec variable, one can, without any prior knowledge of the spec variable name, get that variable (not just its value), and use it to pass some value to a Test which is a spec search. This issue needs more discussion – no current language supports such capability – if you're doing a spec search (on the period, for instance), you MUST know the name of the spec variable used to program the period in the timing block, and you MUST be aware of the effects of varying that parameter).
      - Obviously, there must be some syntax allowing one to vary HW parameters (as expressed in STIL syntax, rather than in target tester syntax) – but that would vary ONLY the HW parameter in question, and not any other HW resources that would normally track that parameter as may occur if a spec were used (and those resources were programmed in terms of the spec to be varied).
  - Discussion of retest proposal (included at the end of the current syntax document).
    - No discussion of this.

- Open issues - are there other open issues that should be considered? A review of the open issues list can guide us here.
    - Issues list: http://spreadsheets.google.com/ccc?key=0AoKiPr1I9LY9dF95dkhSTVVqOU5GbWJyWFNhY0JPX0E&hl=en
    - Namespace resolution examples document: http://docs.google.com/Doc?docid=0AYKiPr1I9LY9ZGY4dmNjNTNfMGZkOGJ2bmZy&hl=en
    - If logged into your google account, can edit. If not, can only view.
- Next Meeting 02/11/10.

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