

1450.4 meeting minutes - 12/11/08

Attendees: Jim O'Reilly, Ernie Wahl, Bruce Parnas, Ajay Khoche

Not present: Doug Sprague

Agenda:

- Preamble:
 - Record Meeting (*2)
 - To listen to the meeting recording, do the following:
 - Call the (US) dial-in numbers 1-877-421-0003 (toll free) or 1-770-615-1374 (toll)
 - Enter the passcode code 747464
 - Once dialed in with the proper access code, enter *3 (star 3)
 - Then enter the file number 54834101 for this conference (this number will change each week).
 - Press 1 to listen to the conference
 - IEEE Meeting Preamble (No discussion of proprietary information).
 - Review proposed variable attributes. Ernie to walk through rationale for permissions and owner. Can these be combined (i.e., does RO imply that it would always be owned by the system? Does RRW (right-hand-side writable, variable itself should not be deleted) imply ownership by either system or user?
 - The purpose of the proposed variable attributes is to aid writers of test generation software. The proposed variable attributes are:
 - Permission:
 - RW: Variable can be altered, renamed, or deleted.
 - RRW: Variable cannot be renamed or deleted (i.e., the left-hand-side MUST remain unchanged). Its value can be changed, however (i.e., the right-hand-side can be changed).
 - RO: Variable cannot be renamed, deleted, or altered (i.e., both the left-hand-side and the right-hand-side must remain unchanged).
 - The default for the Permissions attribute (if not specified) is "RW".
 - Owner: The owner of a variable can be either "System" (i.e., owned by a library used by test generation software, or by test generation software itself), or "User" (owned by the user – i.e., the person creating the test program in STIL). The default, if not specified, is "User".
 - Description: A text string which can be used as a "tool-tip"-like comment to indicate the variable's purpose. It would typically be displayed in an input screen of test generation software. The default description (if none is specified) is an empty string.
 - How does RO interact with previously defined (Const) attribute?
 - The Permissions attribute is unrelated to the "Const" attribute. The Permissions attribute is intended to be used ONLY by test generation software, not to allow or deny any run-time notion of RW or RO.
 - Review proposed syntax relative to variable initialization syntax previously agreed to. Should initial value and size (of arrays) simply be additional attributes of variables (there IS precedent for this in dot1).
 - Discussed this in detail. The consensus was that the dot1 variable syntax would be allowed for those variable types defined in dot1 (assuming the "Design 2005" substatement was present in the "STIL { }" statement). We will not extend these to the various variable types defined for dot4, but instead, simply stick with the syntax previously agreed to for defining variable initial values and array sizes.

- Variable scoping and namespaces (issue #12 of open issues list – located at http://spreadsheets.google.com/ccc?key=pE11-gPUmt2ZTw_kcCTgnKw&inv=jim_oreilly@ieee.org&t=933048453488551871&guest)
 - Discussed how to specify variables which are global to the test program, and can be accessed by all flow subblocks (tests, flownodes, and flows). Up to now, we've been expecting that the TestProgram block is the last block specified – but if we do that, then how do we allow variables blocks specified in the TestProgram to be accessible to all subblocks?
 - We've been trying to avoid forward references, but further study (after the meeting) shows that there IS some precedent in dot0 for forward references (the PatternBurst block is typically specifies patterns which haven't yet been defined). This forward referencing was done in part because the dot0 WG wanted to have the pattern data (typically, the lengthiest part of a STIL file) be the last elements specified – and thus, forward-references from the PatternBurst to (yet-to-be-defined) patterns were necessary.
 - We can take a like approach to our problem, if we state that the TestProgram block MUST be specified BEFORE any other blocks (except, perhaps, a “SocketDef” or NameMap used for the purpose of signal to tester resource mapping). If we allow that, then TestProgram variables can be defined, and are therefore accessible to all Test or Flow blocks. The only forward references required in this case are forward references of EntryPoint names (tests or flows), and perhaps to the BinDef and BinMap blocks (although since those blocks don't use variables, we could require them to be defined before the TestProgram block).
 - By doing so, we can allow the variables mechanism as it currently stands to be used. We still need to define a set of test program variables which are required to be present, but at least this gets around the issue of needing a specific name for a variables block. The user can choose any name desired for the variables block, as long as:
 - that block is referenced by the “Variables” statement in the TestProgram block, and
 - The variables block specified and referenced contains the required variables.
- Category/Selector/Spec cross references to alter testing based on a measurement result (issue #10). Need to understand the rules in other parts of STIL for how to access (for reading current value, or updating value) of a spec variable. Issue still under study. If sufficient semantic rules don't exist in other parts of STIL, we'll need to create them.
- Next Meeting 01/08/09.

For reference STIL .4 information can be found at the IEEE STIL website:

<http://grouper.ieee.org/groups/1450/> (select the [P1450.4](http://grouper.ieee.org/groups/1450/P1450.4) link from the table) or use the direct link <http://grouper.ieee.org/groups/1450/dot4/index.html>