

Summary of Cliff's votes:

SVDB 699 - YES
SVDB 907 - YES
SVDB 1035 - YES
SVDB 1288 - YES
SVDB 1425 - YES
SVDB 1468 - YES (with friendly amendment?)
SVDB 1710 - YES
SVDB 1747 - NO-WAY !!!
SVDB 1846 - NO
SVDB 1940 - NO
SVDB 1949 - YES

=====

SVDB 699 - YES

SVDB 907 - YES

SVDB 1035 - YES

SVDB 1288 - YES

SVDB 1425 - YES

SVDB 1468 - YES (with friendly amendment?)

Paul Menchini used to nail me on the "correct" usage of "which" and "that" all the time.

To be perfectly correct,

WAS:

The key difference between these procedures is design intent **which** allows software tools to perform additional checks ...

PROPOSED:

The key difference between these procedures is design intent **that** allows software tools to perform additional checks ...

See the following web site (there are many more).

<http://www.worldwidewords.org/articles/which.htm>

I know this is not consistently used in the LRM, but I see no reason to add another minor grammatical mistake.

SVDB 1710 - YES

SVDB 1747 - NO-WAY !!!

I am strongly opposed to this proposal.

With passage of this proposal, engineers are going to be tempted to do the following:

```
`default_nettype none  
`default_nettype ports_only wire
```

They will then be forced to declare all internal nets but not have to declare port data types. This is at best only a small step towards insuring good declarations. The intent is to force declarations of internal nets with the hope that any typos in the code will be caught by a corresponding net declaration. This fails in all of the following cases:

- (1) If there is a typo in the net declaration, the compiler will flag an error for the good code (engineers will spend significant time debugging the good code only to later find the problem in the declaration - this is a compiler mis-direction - I ran into these problems while doing VHDL design).
- (2) The declaration of internal nets does not guarantee that the correct size must be declared. At least VHDL requires declarations with size-matching.
- (3) Declaration requirements are easily defeated by randomly declaring an identifier to match the typo in the actual code (I saw VHDL examples of this problem). This is clearly a user mistake but the false sense of security instilled by required declarations can make these bugs time consuming to find.
- (4) If you turn on these directives, they are active until turned off or changed. Because there is only a global version of these directives, we have to clean up at the bottom of each file or previously working later files could fail to compile.
- (5) If anything, we should be allowed to use variable types with this directive (I know, I know, variables are not nets) but ``default_nettype logic` (or `bit`) or even a user defined type would be infinitely more useful than this proposal.

In my opinion, required declarations are an archaic software concept that have been poorly implemented in Verilog and not well suited to finding the types of bugs they were intended to identify.

As I mentioned back in August, it is much more useful to check for connectivity (drivers, receivers, etc.). Connectivity is much harder to fool, requires fewer declarations and gives stronger checking for the unintentional insertion of typo-identifiers. My proposal was defeated because vendors were not ready to make the capability mandatory. Vendors suggested that connectivity was better left to linting tools.

I believe this half-baked proposal is also better left to linting tools. Add the following attribute and let the linting tools check this:

```
(* lint, default_nettype=none *)  
(* lint, default_porttype=wire *)
```

And attach the attributes to just the module header, not the entire Verilog input stream.

This proposal adds a mandatory compiler directive to potentially solve a very small condition while potentially introducing its own set of problems.

Connectivity is a much better way to check for typos and errors.

SVDB 1846 - NO

I just want to hear Steve Sharp say that he approves of this change. At one point, Steve said we would need -noconfig version of all of the 2005 and 2008 versions. If Steve is okay with this change, I have no objection.

SVDB 1940 - NO

I may not have a strong objection. I just want to know that scalar and vector nets are also covered elsewhere since they have been removed from the sections covered by this proposal.

SVDB 1949 - YES