36.16 Variables

ADD

30) The \texttt{vpiDecompile} and \texttt{vpiFullName} properties for variable objects which are members of structs, unions, or class vars shall include their struct, union, or class var name prefix. The \texttt{vpiName} property for these objects shall not include such prefixes. The \texttt{vpi\_handle\_by\_name} function shall require the \texttt{vpiDecompile} form of the name to properly resolve it for any non-top-level scope context, and the \texttt{vpiFullName} form shall be required for the top level. If the object is an indexed element or indexed sub-array (slice) of another object, those indices shall be included in \texttt{vpiDecompile}, \texttt{vpiName}, and \texttt{vpiFullName} properties for the object in order to distinguish it from its \texttt{vpiParent} object. Examples:

```verilog
module top;
    bit [7:0] arr1 [1:4][9:15];
    struct { integer i1; shortint j1; logic[1:4]vec[5:8]; } psarr;
    class cdef;
        int cvInt;
    endclass
    cdef cv = new;
endmodule

// Objects from above declarations-
\texttt{vpiFullName} : top.arr1[1][9]
\texttt{vpiDecompile} : arr1[1][9]
\texttt{vpiName} : arr1[1][9]

\texttt{vpiFullName} : top.psarr.il
\texttt{vpiDecompile} : psarr.il
\texttt{vpiName} : il

\texttt{vpiFullName} : top.psarr.vec[5]
\texttt{vpiDecompile} : psarr.vec[5]
\texttt{vpiName} : vec[5]

\texttt{vpiFullName} : top.cv.cvInt
\texttt{vpiDecompile} : cv.cvInt
\texttt{vpiName} : cvInt
```
36.15 Nets

ADD

29) The *vpiDecompile* and *vpiFullName* properties for net objects which are members of structs shall include their struct name prefix. The *vpiName* property for these objects shall not include such prefixes. The *vpi_handle_by_name* function shall require the *vpiDecompile* form of the name to properly resolve it for any non-top-level scope context, and the *vpiFullName* form shall be required for the top level. If the object is an indexed element or indexed sub-array (slice) of another net object, those indices shall be included in *vpiDecompile, vpiName, and vpiFullName* properties for the object in order to distinguish it from its *vpiParent* object. Examples:

```vhdl
module top;
  wire logic[7:0] warr1 [1:4][9:15];
  wire struct { integer i1; time t1; logic[1:4] vec[5:8]; } psarr;
endmodule

// Objects from above declarations-
  vpiFullName : top.warr1[1][9]
  vpiDecompile : warr1[1][9]
  vpiName : warr1[1][9]

  vpiFullName : top.psarr.i1
  vpiDecompile : psarr.i1
  vpiName : i1

  vpiFullName : top.psarr.vec[5]
  vpiDecompile : psarr.vec[5]
  vpiName : vec[5]
```