# Minor Corrections

These changes are related to the description of $isunknkown function.

## 1.1 Page 232 in Section 17.10

Assertions are commonly used to evaluate certain specific characteristics of a design implementation, such as whether a particular signal is “one-hot”. The following system functions are included to facilitate such common assertion functionality:

— `$onehot (<expression>)` returns true if only one bit of the expression is high.

— `$onehot0(<expression>)` returns true if at most one bit of the expression is high.

— `$isunknown (<expression>)` returns true if any bit of the expression is ~~‘x’~~ X or Z. This is equivalent to `^<expression> === 'bx`.

All of the above system functions have a return type of bit. A return value of `1'b1` indicates true, and a return value of `1'b0` indicates false.

~~If the specified clock tick in the past is before the start of simulation, the returned value from the `$past` function is a value of X.~~

## 1.2 Page 239 in Section 23.10

Assertions are commonly used to evaluate certain specific characteristics of a design implementation, such as whether a particular signal is “one-hot”. The following system functions are included to facilitate such common assertion functionality:

— `$onehot` returns true if one and only one bit of expression is high.

— `$onehot0` returns true if at most one bit of expression is high.

— `$isunknown` returns true if any bit of the expression is `X` or `Z`. This is equivalent to `^expression === 'bx`.

 