Property Equivalence Checking

Using the API

```
include /vols/jasper_users/gvamorim/workspace/pec/pec.tcl

set p1 {A |-> ##[0:$] B}
set p2 {A |-> s_eventually(B)}
set clk ""
set rst ""
set signal_list {A B}

prop_eq_checker $p1 $p2 $clk $rst $signal_list
```

```
Full equivalence between
'A |-> ##[0:$] B'
'A |-> s_eventually(B)'
```

Examples

Example 1

Input:

- p1: `a ##1 b ##1 c |=> d`
- p2: `a ##1 b |=> c ##1 d`
- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst a b c d}`

Output:

```
'a ##1 b ##1 c |=> d'
implies
'a ##1 b |=> c ##1 d'
```

Example 2:

Input:

- p1: `A[*4]`
- p2: `A ##1 A ##1 A ##1 A`

- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst A}`

Output:

```
Full equivalence between
'A[*4]'
'A ##1 A ##1 A ##1 A'
```

Example 3:

Input:

- p1: `(A ##1 B) [*2]`
- p2: `A ##1 B [*2]`
- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst A B}`

Ouput:

```
Full equivalence between
'(A ##1 B) [*2]'
'A ##1 B [*2]'
```

Example 4:

Input:

- p1: `(A) throughout (B ##1 C ##1 D)`
- p2: `(A)[*0:$] intersect (B ##1 C ##1 D)`
- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst A B C D}`

Ouput:

```
Full equivalence between
'(A) throughout (B ##1 C ##1 D)'
'(A)[*0:$] intersect (B ##1 C ##1 D)'
```

Example 5:

Input:

- p1: `(A) throughout (B ##1 C ##1 D)`
- p2: `(A)[*0:$] intersect (B ##1 C ##1 D)`
- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst A B C D}`

Ouput:

```
Full equivalence between
'(A) throughout (B ##1 C ##1 D)'
'(A)[*0:$] intersect (B ##1 C ##1 D)'
```

Example 6

Input:

- p1: `A |-> ##[0:$] B`
- p2: `A |-> s_eventually(B)`
- clk: `clk`
- rst: `rst`
- signal_list: `{clk rst A B}`

Output:

```
Full equivalence between
'A |-> ##[0:$] B'
'A |-> s_eventually(B)'
```

Example 7

Input:

- p1: `A`
- p2: `!A`
- clk: ``
- rst: ``
- signal_list: `A`

Output:

```
'A'
'!A'
conflict with each other
```