



Compare Operations with Various Quantizations

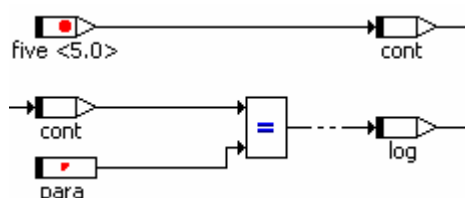
PROBLEM DESCRIPTION

If compare operations are executed in ASCET on elements for which the implementations or quantizations are different – the resulting code may be faulty.

EXAMPLE

In a block diagram the constant value 5 is assigned to a variable. The variable has been implemented with a quantization of 1.5: $f(\text{phys}) = 1.5 * \text{phys}$.

This variable is then compared with a parameter to which the value 5 is also assigned, but which has been quantized with **ident**: $f(\text{phys}) = \text{phys}$.



The generated code from this operation looks as follows:

```
_cont = (uint8)8;  
_log = (sint8)((uint16)_cont << 1 == _para * (uint16)3);
```

When the constant five <5.0> is assigned, it will be recalculated according to the implementation of the variable as **cont**: $f(5)=1.5 * 5 = 7.5$. This value will then be rounded to 8 so that it can be assigned to the integer variable.

In the subsequent comparison operation, a further re-calculation is done: **cont** is multiplied with 2 in a shift operation, and the parameter **para** is multiplied by 3. In that way, the differing quantization of factor 1.5 of the two compared variables is compensated for.

In this case, the value on the left side of the comparison will be 16 and on the right side 15. The comparison will thus yield a negative result, even though both inputs have the same values in the physical model.

SOLUTION

ASCET is not able to compensate for such problems based on rounding errors. Users are therefore asked to ensure that the implementations and values do not allow rounding errors. If necessary, users are advised to use <OR> operations.





APPLIES TO PRODUCTS

ASCET-MD

ADDITIONAL SEARCH TERMS

µController, Implementation, Target-Code

