

How to specify loop iterations in nested loops

IN THIS DOCUMENT

- ▶ Setting a global loop scope
 - ▶ Setting a local loop scope
-

version	1.1.1
scope	Example. This code is provided as example code for a user to base their code on.
description	How to specify loop iterations in nested loops
boards	Unless otherwise specified, this example runs on the SliceKIT Core Board, but can easily be run on any XMOSS device by using a different XN file.

In many cases the XTA will know the number of iterations for a given loop. However, in some cases, this information cannot be deduced by the tools so you must supply it.

This example describes some options for setting loop iterations for nested loops, and introduces the concept of the loop scope.

For example, compile the following code:

```
int f() {
    int i, j, k = 0;
    for (i = 0; i < 5; ++i) {
        #pragma xta label "outer_loop"
        for (j = 0; j < i; ++j) {
            #pragma xta label "inner_loop"
            k += i + j;
        }
    }
    return k;
}

int main() {
    f();
    return 0;
}
```

By default, the tool assumes that the iterations for loops are *relative*, meaning that the iterations for an inner loop will be multiplied by the iterations of enclosing loops.

However this is not sufficient to describe all loop structures. If this assumption is not correct (as is the case in the above code) a loop count can be set to *absolute*. The iteration count set on an absolute loop is not multiplied up by the iterations set on enclosing loops. For example, in this case, if you set the inner loop to have an absolute scope, and a loop count of 10, then this will correctly capture the behavior of the code.

1 Setting a global loop scope

Load the resulting executable into the XTA then click the 'Add define' button in the toolbar. In the dialog, switch to the *Loop Scope* tab, enter 'inner_loop' in the reference section and check the *make scope absolute* box. Now, any iterations set on 'inner_loop' will be treated as absolute.

To add a global loop scope using the command line XTA, or from an XTA script/embedded source command, the following can be used:

```
add loopscope inner_loop a
```

2 Setting a local loop scope

Load the resulting executable into the XTA then time the function 'f'. This will create the route. Right-click in the left hand side border of the editor on the source line inside the loop ($k += j + i$) and select 'Set loop scope'. In the resulting dialog check the *make scope absolute* box.

To add a local loop scope using the command line XTA, or from an XTA script/embedded source command, the following can be used:

```
set loopscope 0 inner_loop a
```

This will set a the inner loop for the route with an id of 0 to have a scope of *absolute*.



Copyright © 2013, All Rights Reserved.

Xmos Ltd. is the owner or licensee of this design, code, or Information (collectively, the "Information") and is providing it to you "AS IS" with no warranty of any kind, express or implied and shall have no liability in relation to its use. Xmos Ltd. makes no representation that the Information, or any particular implementation thereof, is or will be free from any claims of infringement and again, shall have no liability in relation to any such claims.