

How to use labels in inline assembly

version	1.0.1
scope	Example. This code is provided as example code for a user to base their code on.
description	How to use labels in inline assembly
boards	Unless otherwise specified, this example runs on the SliceKIT Core Board, but can easily be run on any XMOS device by using a different XN file.

It is invalid to write inline assembly code that branches to another inline assembly statement. However it is sometimes useful to write inline assembly that branches within the instructions contained in that `asm` statement.

A common mistake is to write the following:

```
asm(  
    "bt %1, .Lfoo\n"  
    "mov %0, %2\n"  
    "bu .Ldone\n"  
    ".Lfoo:\n"  
    "mov %0, %3\n"  
    ".Ldone:\n"  
    : "=r"(a)  
    : "r"(b), "r"(c));
```

If the statement containing the `asm` statement is duplicated (e.g. due to function inlining or loop unrolling) this will result in an error due to the labels `.Lfoo` and `.Ldone` being defined multiple times.

Instead you should write:

```
asm(  
    "bt %1, .Lfoo%=\n"  
    "mov %0, %2\n"  
    "bu .Ldone%=\n"  
    ".Lfoo%=: \n"  
    "mov %0, %3\n"  
    ".Ldone%=: \n"  
    : "=r"(a)  
    : "r"(b), "r"(c), "r"(d));
```

The `%=` escape sequence emits a number that is unique to each expansion of the `asm` statement, making the labels unique.