

Application Note: AN10104

# How to use the XTA to time a loop

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to use the XTA to time a loop.

---

## Required tools and libraries

This application note is based on the following components:

- xTIMEcomposer Tools - Version 14.0.0

## Required hardware

Programming how-tos are generally not specific to any particular hardware and can usually run on all Xmos devices. See the contents of the note for full details.

## 1 How to use the XTA to time a loop

The XTA can be used to determine the worst and best case timing for a loop, i.e. to determine the time taken for a single iteration of the loop. Compile the following code:

```
int loop() {
    int i, j = 0;
    for (i = 0; i < 10; ++i) {
        #pragma xta label "loop_label"
        j += i;
    }
    return j;
}

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

Load the resulting binary into the XTA.

In the xTIMEcomposer studio, select the *Analyze* button on the left-hand side navigation toolbar. Next select the *Analyze Timing* perspective via the main toolbar. A binary can be loaded into the XTA then via the *Load Binary into XTA* toolbar button.

From the command line, a binary can be loaded using the following xta command:

```
load a.xe
```

The loop can then be timed as follows:

In the xTIMEcomposer studio editor, right-click on the border of the line within the loop  $j += i$  and select 'Set loop point'. Next select the 'Analyze loop' button in the toolbar. The best and worst case timing for this loop will then be displayed in the *Routes* view in the left hand panel.

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

```
analyze loop loop_label
```

Note: In order to do this in a maintainable way, the above relies on the *xta label* pragma, which lets you associate a label with a lexical block in the source code.