Application Note: AN10116

How to write XTA command scripts

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to write XTA command scripts.

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.

Copyright 2015 XMOS Ltd.

1 How to write XTA command scripts

The XTA can be used interactively to investigate and measure the timing properties of you program. Once satisfied, it is often a requirement to capture these timing requirements in a script thus allowing the timing of your program to be automatically re-validated on future modifications. This can be achieved by placing XTA commands into a script file.

For example, compile the following code:

```
#include <stdlib.h>
#include <xs1.h>

port p1 = XS1_PORT_1A;
port p2 = XS1_PORT_1B;

int main() {
  int x;

  #pragma xta endpoint "input"
  p1 :> x;

  // Checks for errors..
  if (x == 1) {
    #pragma xta label "error_case"
    exit(1);
  }

  // do some computation here..

  #pragma xta endpoint "output"
  p2 <: 0;
  return 0;
}
```

Assume that there is a timing requirement between the *input* and the *output* of 100.0 ns. Assume also assume that you are not interested in the timing of the *error_case*.

Add the following lines to a file *script.xta*:

```
load a.xe
analyze endpoints input output
set exclusion - error_case
set required - 100.0 ns
print summary
exit
```

Note: In the above script, the '-' refers to the ID of the most recently created route. Using this construct can result in more maintainable scripts.

On the command line, this script can then be executed:

```
xta source script.xta exit
```

Which will display the following:

```
Route(0) endpoints: input to output
    Pass with 1 unknown, Num Paths: 1, Slack: 20.0 ns, Required: 100.0 ns, Worst: 80.0 ns, Min Core Frequency:
      ↪  320 MHz

Pass, Min Core Frequency: 320 MHz
```

Copyright © 2015, 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.