Application Note: AN10075

How to use a select function

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

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 a select function

Select statements can be implemented as functions, allowing their reuse in different contexts.

The declaration

```
select channel_input(chanend c_a, chanend c_b)
```

declares `channel_input` to be a select function that takes two arguments c_a and c_b. The body of the select function adheres to the same rules as the select statement.

```
{
  case c_a :> int chnl_input_a :
    printstr("Channel Input A Received ");
    printintln(chnl_input_a);
    break;
  case c_b :> int chnl_input_b :
    printstr("Channel Input B Received ");
    printintln(chnl_input_b);
    break;
}
```

You can call the select function from the relevant location in the program. In this example the select function is nested within a select statement where it is inputting data from either one of the two channel ends `c_a, c_b` or from the input port `port_in`

```
select {
  case channel_input(c_a, c_b);
  case port_in when pinsneq(port_input_data) :> port_input_data :
    do_port_input(port_input_data);
    break;
}
```

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.