How to use select with multiple resources

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

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 select with multiple resources

A select statement waits for one of a set of inputs to become ready, performs the selected input and then executes a corresponding body of code. Each input is proceeded by the keyword case and its body must be terminated with a break or return statement. Case statements are not permitted to contain output operations. Select statements can be used to tie together inputs from multiple resources, in this example we look at combining timers, channels and ports into a single select statement.

The select statement implemented below is used to wait for either an input on chnlend_a, an input on chnlend_b, an input from port_a or a timer value. When an input is received from any of the cases an message is printed to display which case has fired. The timer case can be used to wake up and execute at a particular rate, in this example it is every 10000 timer cycles.

```
select
{
  case chnlend_a ->  chnl_input_a :
    printstr("Channel Input A Received ");
    printintln(chnl_input_a);
    break;
  case chnlend_b -> chnl_input_b :
    printstr("Channel Input B Received ");
    printintln(chnl_input_b);
    break;
  case port_a when pinseq(1) -> port_input_a:
    printstr("Port Input A Recieved ");
    printintln(port_input_a);
    break;
  case tmr_a when timerafter(start_time + 10000) -> start_time:
    printstrln("Timer A Fired ");
    break;
  }
}
```