**Application Note: AN10025**

# How to pass pointers over interface calls

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to pass pointers over interface calls.

## 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.

 www.xmos.com

XM007292

# 1 How to pass pointers over interface calls

Interface functions can take pointer arguments:

```
interface my_interface {
  void msg(int *p);
};
```

The client end can then pass a pointer into the function:

```
void task1(client interface my_interface c)
{
  int a[5] = {0,1,2,3,4};
  int *p = &a[0];
  c.msg(p);
}
```

On the server end the select case can access memory via the pointer.

```
void task2(server interface my_interface c)
{
  select {
  case c.msg(int *p):
    printintln(*p);
    printintln(*(p+2));
    break;
  }
}
```

Since passing the pointer implies that both tasks need access to the same memory space, if an interface includes a function that passes pointers, you cannot use it between tasks on different tiles.