

Application Note: AN10120

How to read/write to the console during execution

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to read/write to the console during execution.

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 read/write to the console during execution

The xTIMEcomposer tools can handle system calls on behalf of the target application. This handling is enabled by default.

For example, compile the following code snippet:

```
#include <stdio.h>
#include <print.h>

int main() {
    char c;
    printstr("Please enter a character, then press enter:");
    c = getchar();
    printstr("The character input is: ");
    printcharln(c);
    return 0;
}
```

When the resulting executable is run, the code blocks in the call to `getchar()`. Once a character has been supplied from the console, execution continues and the character is displayed:

```
> xsim a.xe
x
The character input is: x
```

Internally xSIM is monitoring the system call location in the code. When that location is reached it stops the simulation, performs the required system call on the host then restarts the simulation. This is similar to the way system calls are handled on hardware via JTAG. However, in this case no debug interrupt is required thus making the system call handling less intrusive.

This is also supported on hardware targets although, as mentioned above, the support is somewhat more intrusive on the execution of the target application due to having to drop into debug mode to process the system call on the host machine:

```
> xrun --io a.xe
x
The character input is: x
```