

How to read/write to the console during execution

version	1.1.1
scope	Example. This code is provided as example code for a user to base their code on.
description	How to read/write to the console during execution
boards	Unless otherwise specified, this example runs on the SliceKIT Core Board, but can easily be run on any XMOS device by using a different XN file.

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
```



Copyright © 2013, 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.