

Application Note: AN10065

How to run XGDB commands from a file

This application note is a short how-to on programming/using the xTIMEcomposer tools. It shows how to run XGDB commands from a file.

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 run XGDB commands from a file

XGDB commands can be placed in a file, and the tool can be run in a mode which executes the given commands on startup. For example, compile the following code ensuring that debug is enabled (-g):

```
int f() {
    int i, j = 0;;
    for (i = 0; i < 5; ++i) {
        j += i;
    }
    return j;
}

int main() {
    int retval = f();
    return retval;
}
```

Now place the following in a file named *cmds.txt*:

```
connect -s
break f
run
finish
next
print retval
continue
quit
```

These commands, when passed to XGDB will cause the simulator to be used as the target. It will then break at function *f*. The *finish* statement will cause the function to be executed to completion, at which point you can step over the *retval* assignment then print the result. This can be run from the command line using the *-command* argument, and will produce the following result:

```
> xgdb --command=cmds.txt a.xe
GNU gdb (XGDB) 12.1.0 (build 7669)
...etc...
Breakpoint 1, f () at running_commands_from_a_file.xc:10
10   int i, j = 0;;
main () at running_commands_from_a_file.xc:18
18   int retval = f();
Value returned is $1 = 10
19   return retval;
$2 = 10

Program exited with code 012.
```