How to display the resources used by a program

IN THIS DOCUMENT

version	1.1.1
scope	Example. This code is provided as example code for a user to base their code on.
description	How to display the resources used by a program
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.

You can use the xTIMEcomposer tools to show the resources used by a given executable. For example, compile the following code:

```
#include <print.h>

int main() {
  printstr("Hello World!\n");
  return 0;
}
```

1 From within the xTIMEcomposer

Double-click on the resulting binary from within the *Project Explorer*. The executable is opened in the *Binary View*. This gives a graphical view of the resources used by the program (in the *Resources* tab), and the sizes/locations of functions and global data objects (in the *Function Table* and *Data Table* tabs).

2 From the command line

You can view the resources used by the resulting executable from the command line using *xobjdump*:

```
xobjdump --resources a.xe
```

This will produce the following output:

```
....
tile[0] (node "0", tile 0) stack usage,  upper bound: 208
tile[0] (node "0", tile 0) program size,  upper bound: 1092
tile[0] (node "0", tile 0) free memory,  lower bound: 64236
tile[0] (node "0", tile 0) thread usage,  upper bound: 1
tile[0] (node "0", tile 0) unused threads,  lower bound: 7
tile[0] (node "0", tile 0) timer count,  upper bound: 0
tile[0] (node "0", tile 0) unused timers,  lower bound: 10
tile[0] (node "0", tile 0) channel end usage,  upper bound: 0
tile[0] (node "0", tile 0) unused channel ends,  lower bound: 32
Node "0" routing id =  0x0000
Node "0" PLL configuration register value =  0x00002700
Node "0" reference clock divider register value =  0x00000003
Node "0" system frequency (Hz) =  400000000
```

You can also display the code and data section sizes as follows:

```
xobjdump --size a.xe
```

This will produce the following output:

```
Loadable 1 for tile[0] (node "0", tile 0):

      text      data       bss     total
       680        84        64       828
```