

ETS-EVO - LAB1

In this LAB we'll learn how to:

- build an executable to be run on the ETS-EVO
- control the LED on the ETS-EVO front panel

What is required:

- basic knowledge of the C programming language
- Software SDK¹ and associated tools
- ETS-EVO register mapping¹

1. Create a new project

Go to the petalinux installation directory and navigate to software/user-apps folder

Create a new project with:

```
petalinux-new-app lab1
```

This creates an empty project with two important files:

lab1.c

Makefile

In order to control the LEDs we have to know the locations they are mapped to.

This is clearly depicted on the ETS-EVO register mapping document

COUNTER_CTRL 0x81400038

Read-write

[...]

[31..28] output leds

Since the hardware uses a MMU is not possible to directly write to this location, but a mapping has to be done first.

A wrapper utility (mmu.c) has been included to handle the internals of this mapping.

So the user is able to issue:

```
*(volatile unsigned long *) MMU_ADDR(0x81400038) = 0xF0000000;
```

to turn on all the LEDs.

The mmu.o object has to be added to the Makefile and some additional calls have to be issued to the wrapper utility (do_mmap and do_munmap) in the lab1.c file.

For simplicity please overwrite the project folder with the one attached to this document and issue a *make* command to compile the executable.

¹ Can be downloaded from the Digital Instruments website

2. Run the executable on the board

Copy the executable to the target board with the command `putftp lab1 192.168.200.2` (change IP address accordingly).

Log in on the board via telnet (credentials are usr: root - pwd: root)

The board will be probably running its default firmware so it is better to temporarily stop it during the lab execution, by running the following commands:

```
killall microref_loop  
killall microref
```

It is possible to check if the microref executable has been successfully stopped by issuing a `ps` command

Now in order to run the freshly built program is necessary to make it executable by issuing

```
cd /var/ftp  
chmod 755 lab1
```

And launch it with

```
./lab1
```

You should get an output like this:

```
/var/ftp # ./lab1  
/dev/mem opened.  
Memory 0x80000000 (size 0x10000000) mapped at address 0x4818b000  
HW version: 22  
Toggling LEDs.. (press CTRL-C to quit)
```

To stop it simply press CTRL-C

```
^C  
/var/ftp #
```

Congratulations! You were successful in running your first custom program on the ETS-EVO board.