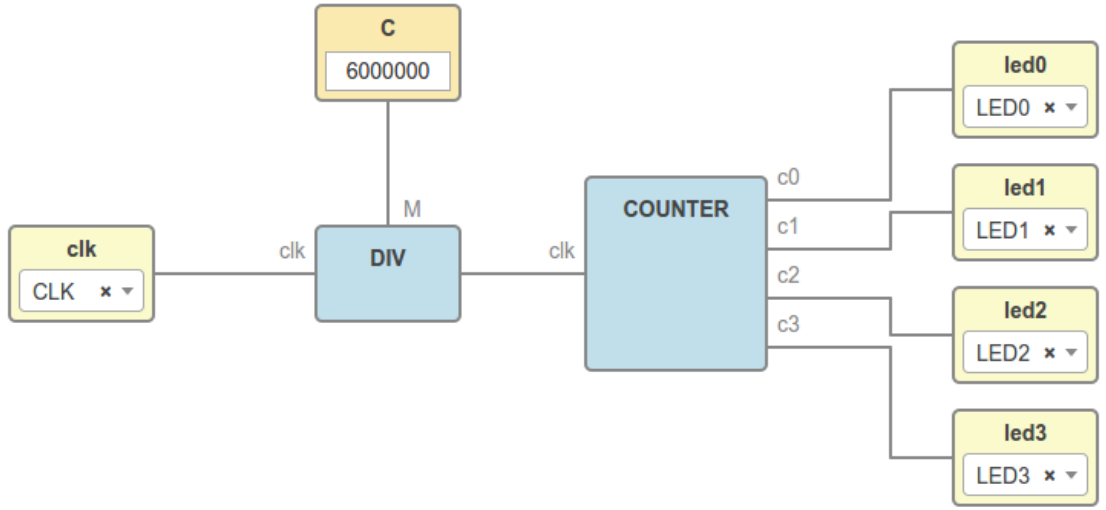
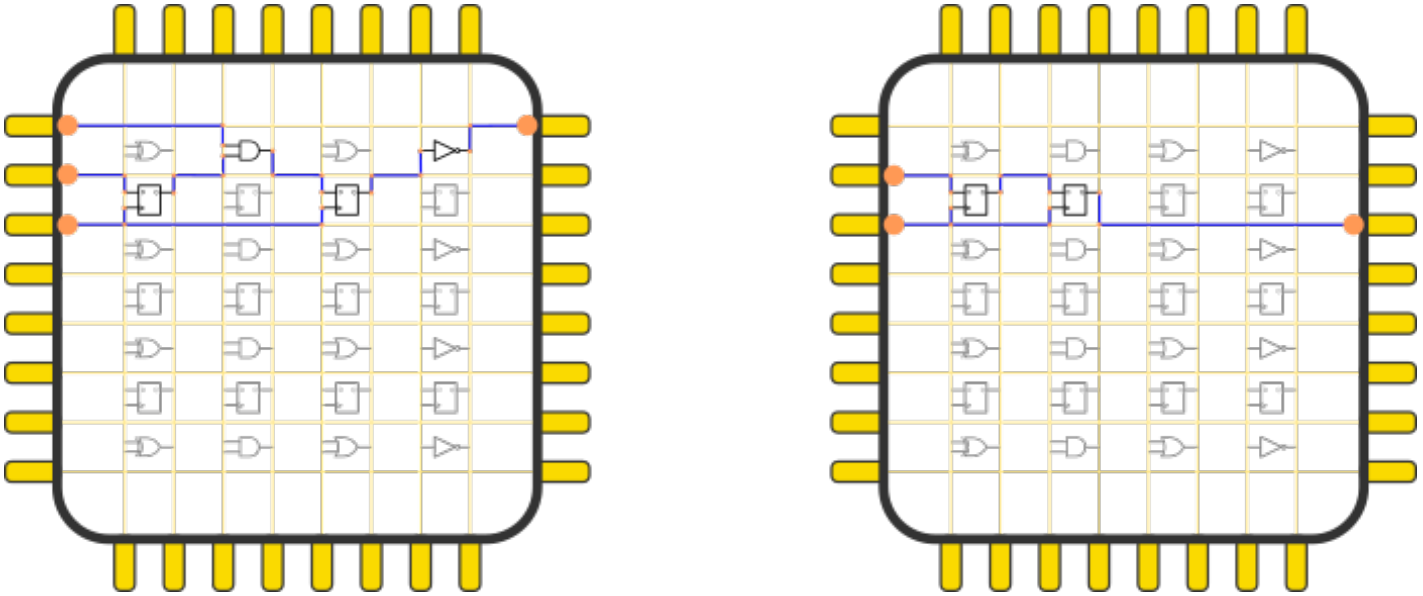
Workshop - Open FPGA tools



Jesús Arroyo Torrens

https://github.com/Jesus89

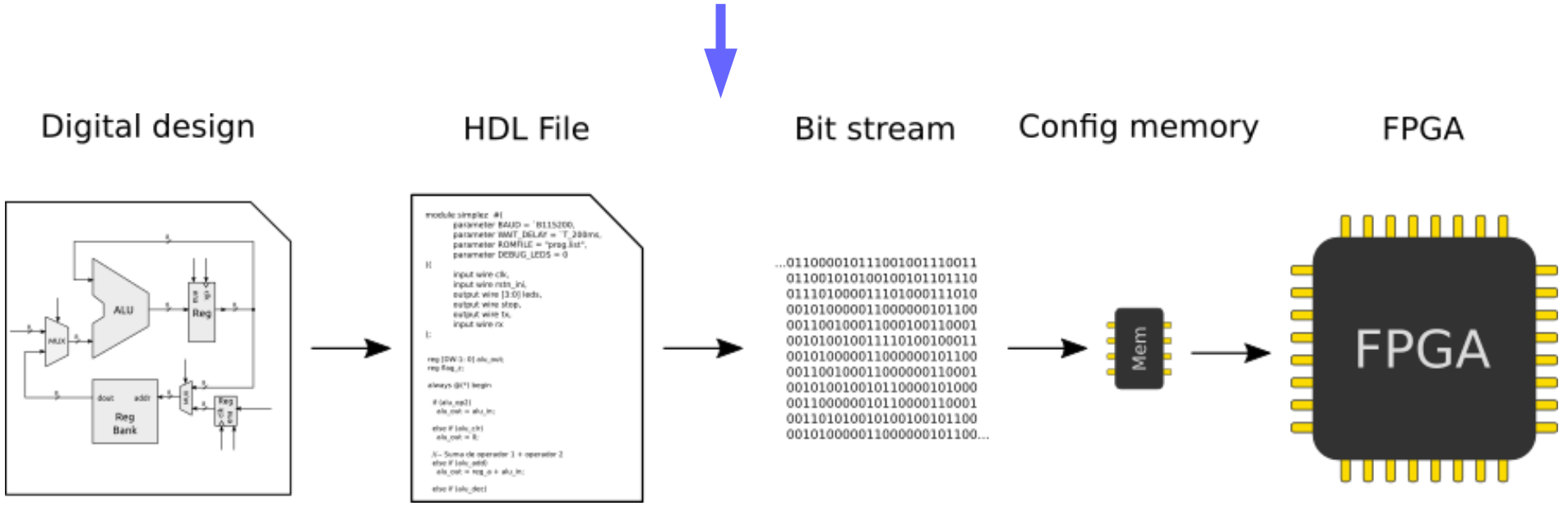
Field Programmable Gate Array



Open FPGA toolchain

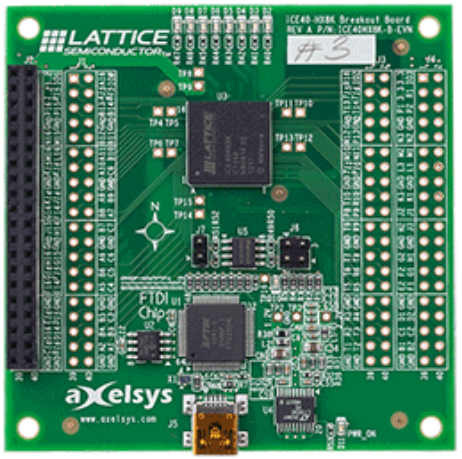
IceStorm Project

by Clifford Wolf

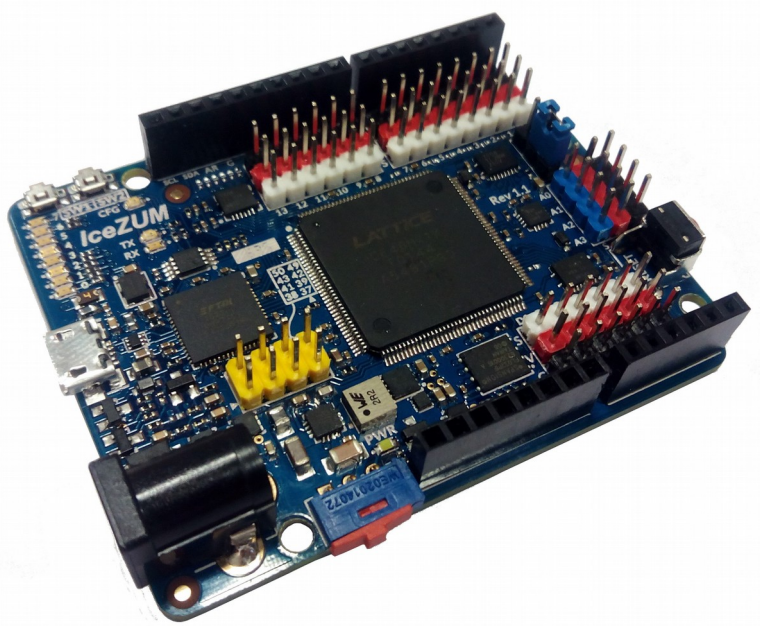


Open FPGA boards

Icestick



iCE40-HX8K Breakout Board



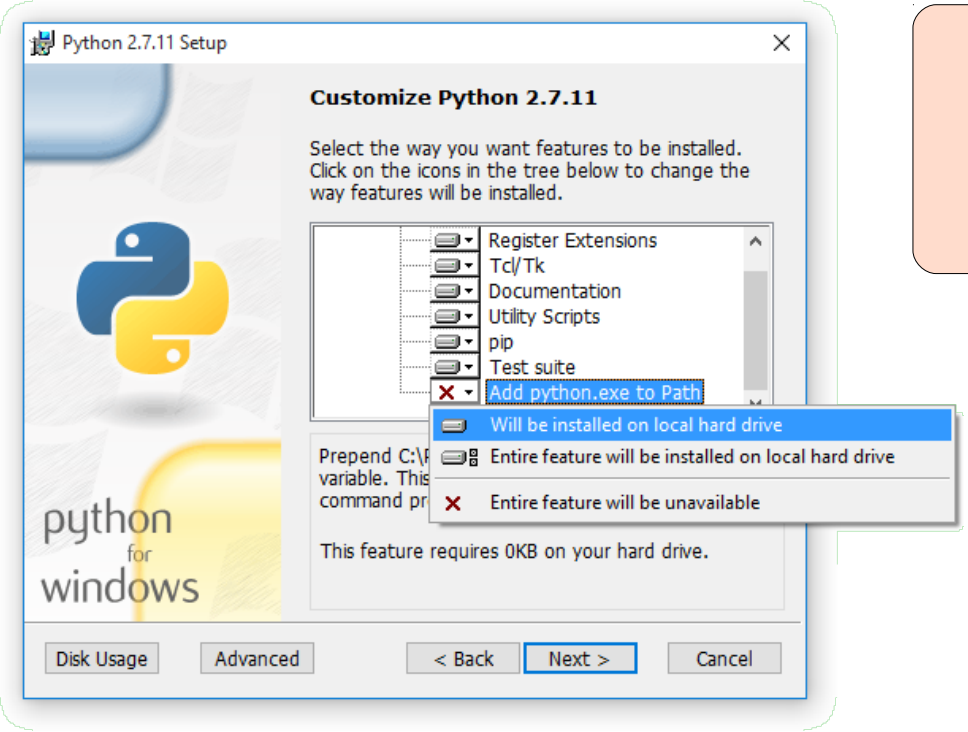
IceZUM Alhambra

Open FPGA tool stack

Open FPGA tool stack

Requirements

1. **Python 2.7** https://www.python.org

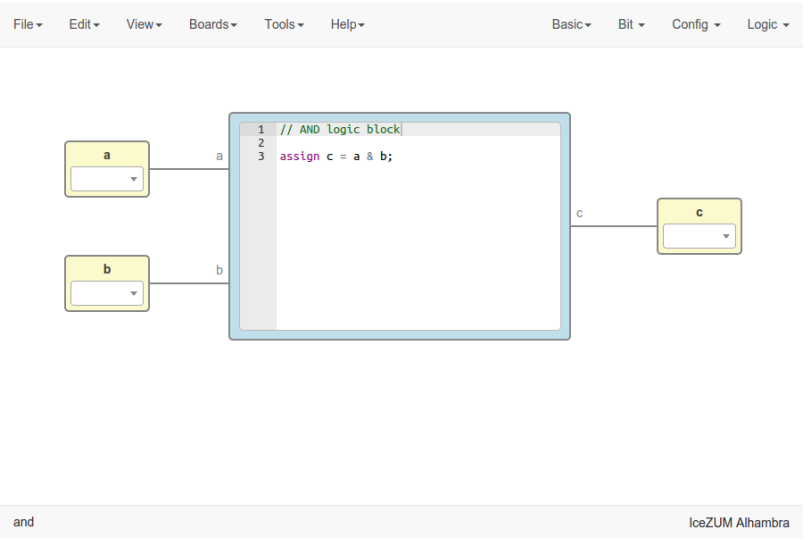


Windows Users: DON'T FORGET to select `Add python.exe to Path` feature on the "Customize" stage

Check installation: open a console and type **`python`**

Icestudio

https://github.com/FPGAwars/icestudio

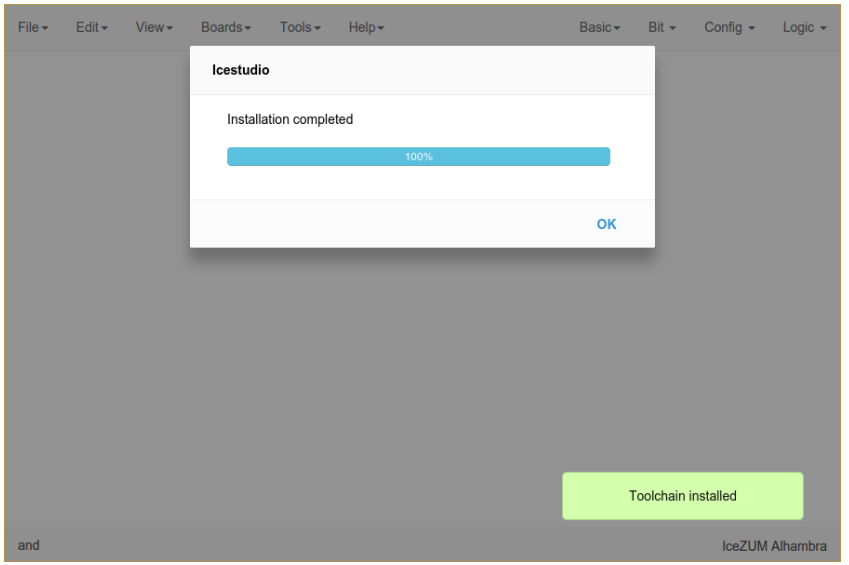


Experimental **graphic editor** for open FPGAs. Created with HTML and JS

Icestudio

1. **Install**

- Copy *icestudio-0.2.4-dev.zip*
- Unzip the file
- Execute *icestudio*



2. **Setup**

- Install toolchain

  Tools > Toolchain > Install

- Install drivers

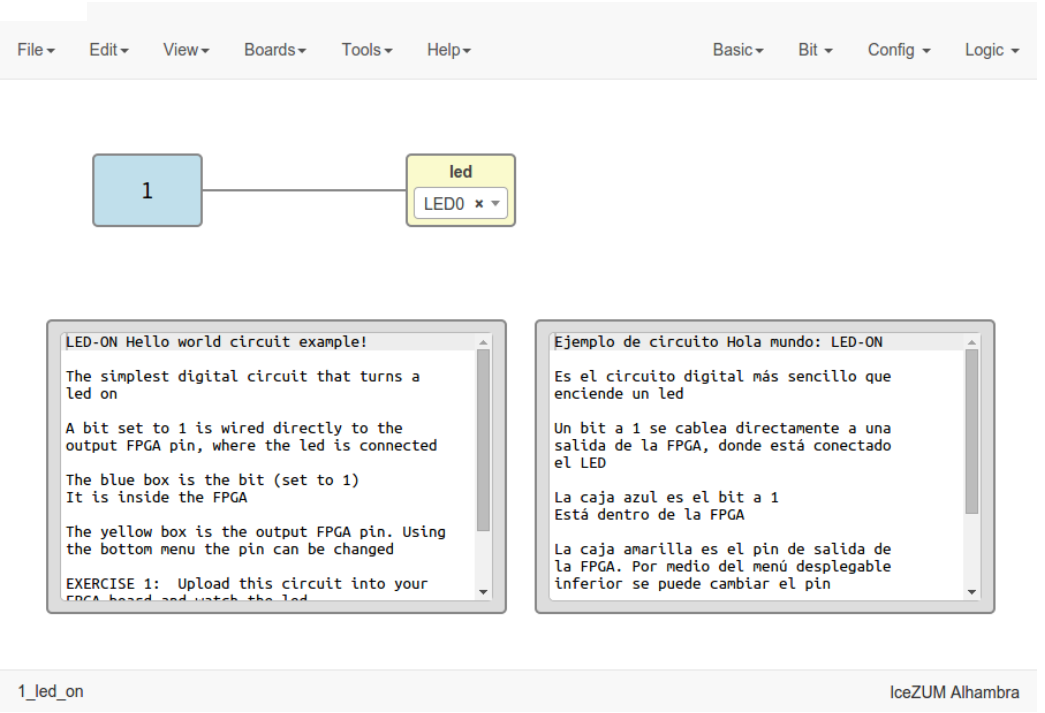
  Tools > Drivers > Enable

Drivers configuration requires administrative privileges
Follow the instructions in each OS

Icestudio

3. **Hello, world!**

- Load example
  1. Basic > 1. Led on
- Select board
  Board > IceZUM
- Select I/O pin
  Edit the combo
- Upload bit stream
  Tools > Upload

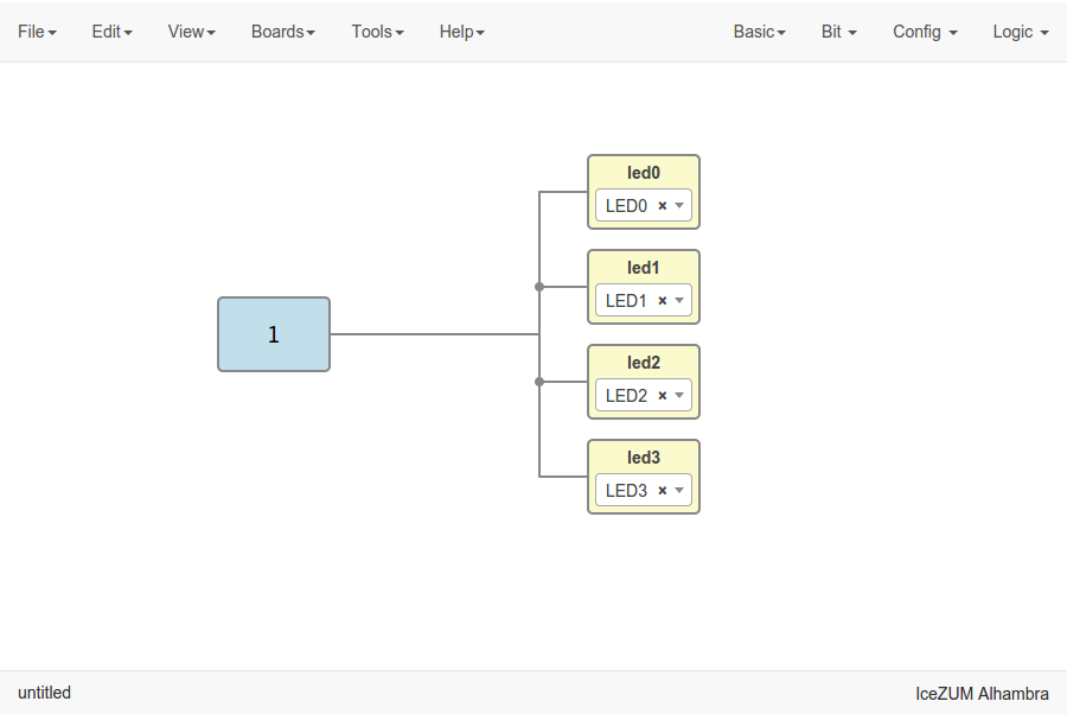


Enter in a block and edit a I/O label by *double clicking*

Icestudio

4. **More leds on**

- Create a project
  File > New project
- Add blocks
  Bit > 1
  Basic > Output
- Connect wires
- Upload bit stream
  Tools > Upload

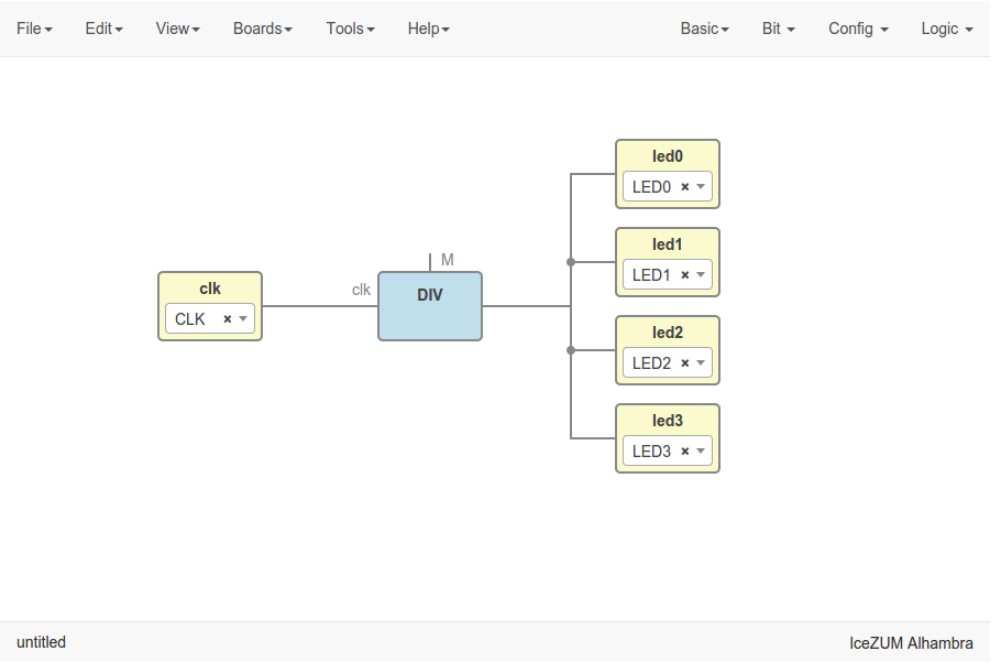


Multiple I/O blocks can be created, e.g. "*led0, led1, led2*"

Icestudio

5. **Blink**

- Import DIV block
  File > Import block
- Add clock input
  Basic > Input
- Connect wires
- Upload bit stream
  Tools > Upload

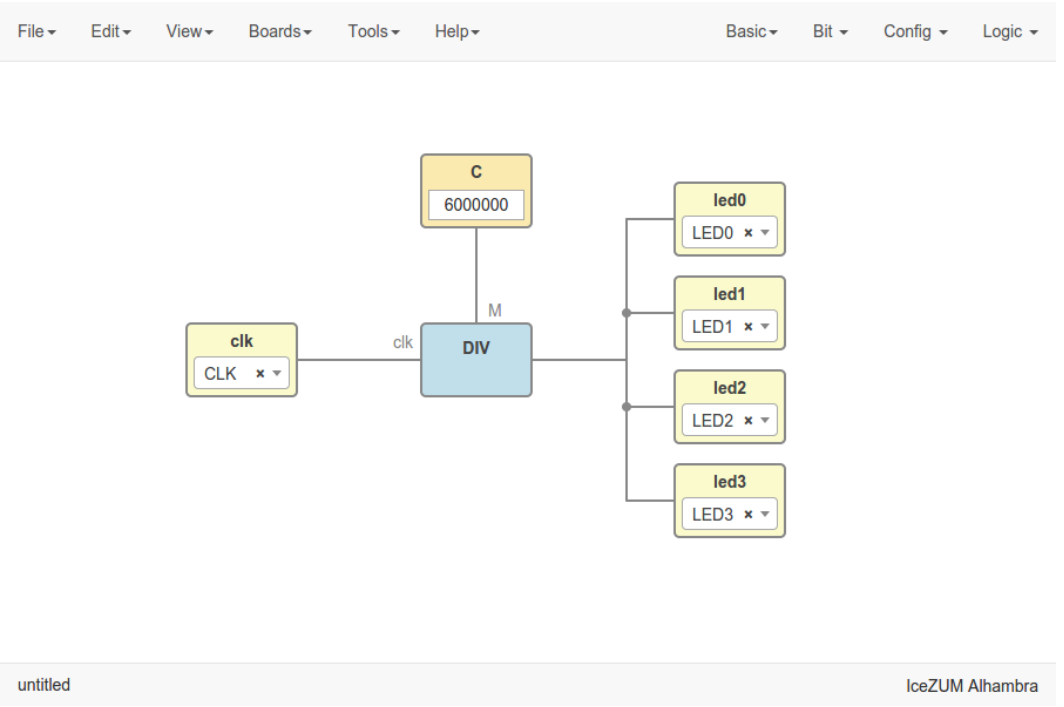


The clock is a 12 MHz signal

Icestudio

6. **Custom blink**

- Add constant
  Basic > Constant
- Insert a value
  6000000
- Connect wires
- Upload bit stream
  Tools > Upload



The clock is a 12 MHz signal

Icestudio

7. **Counter**

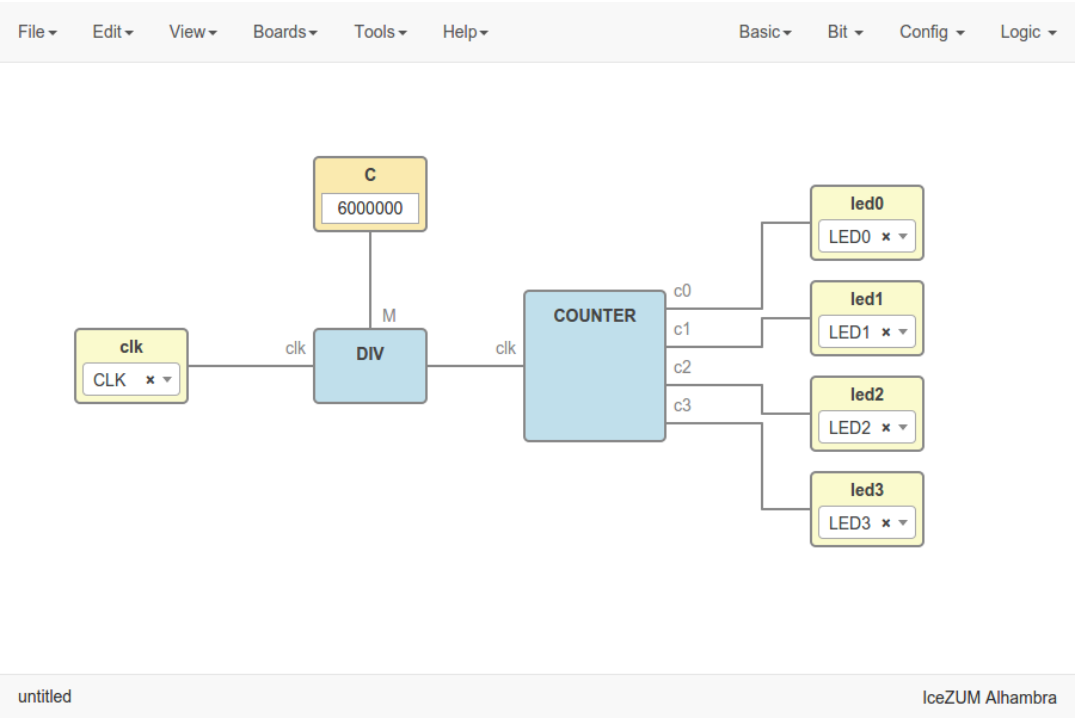
- Import counter block

  File > Import block

- Reconnect wires

- Upload bit stream

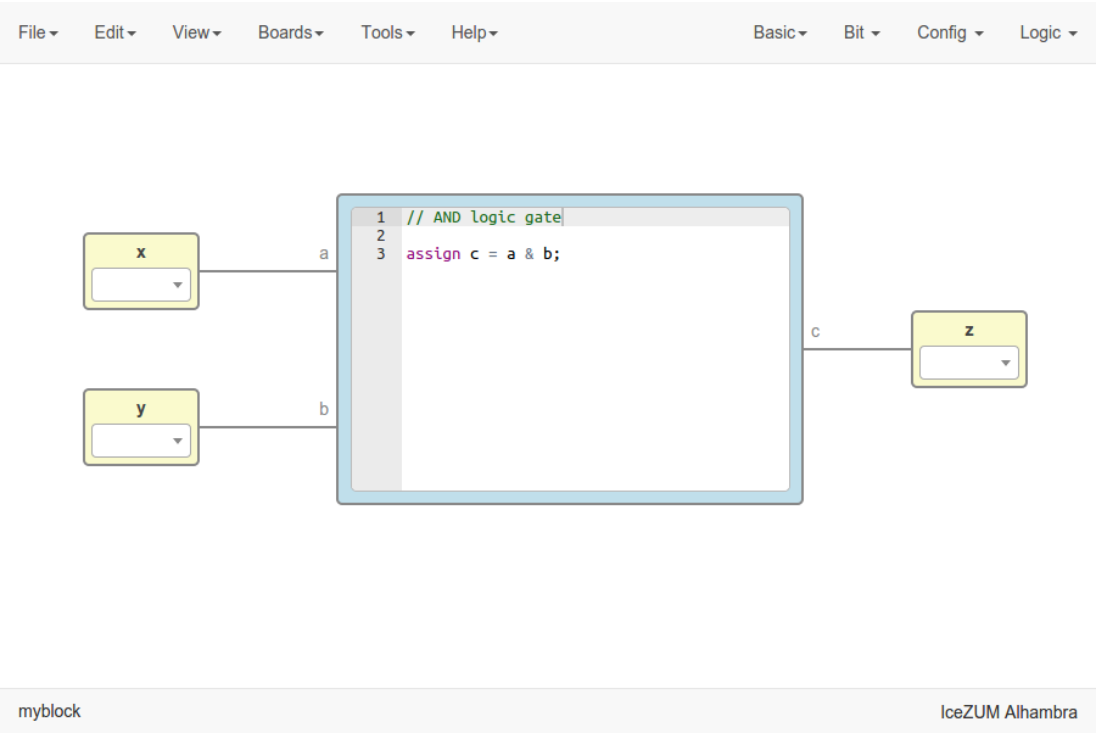
  Tools > Upload



Icestudio

8. **Let's code**

- Create a project
  File > New project
- Add blocks
  Basic > Code
  Basic > Input
  Basic > Output
- Connect wires
- Verify the design
  Tools > Verify



Icestudio

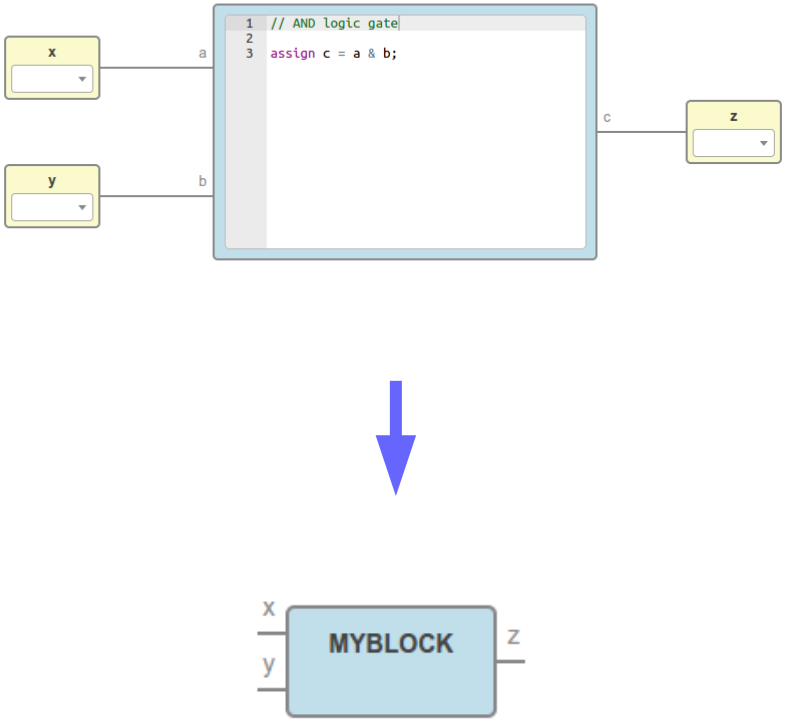
9. **Custom block**

- Load a project

  File > Open project

- Save project as block

  File > Export as block

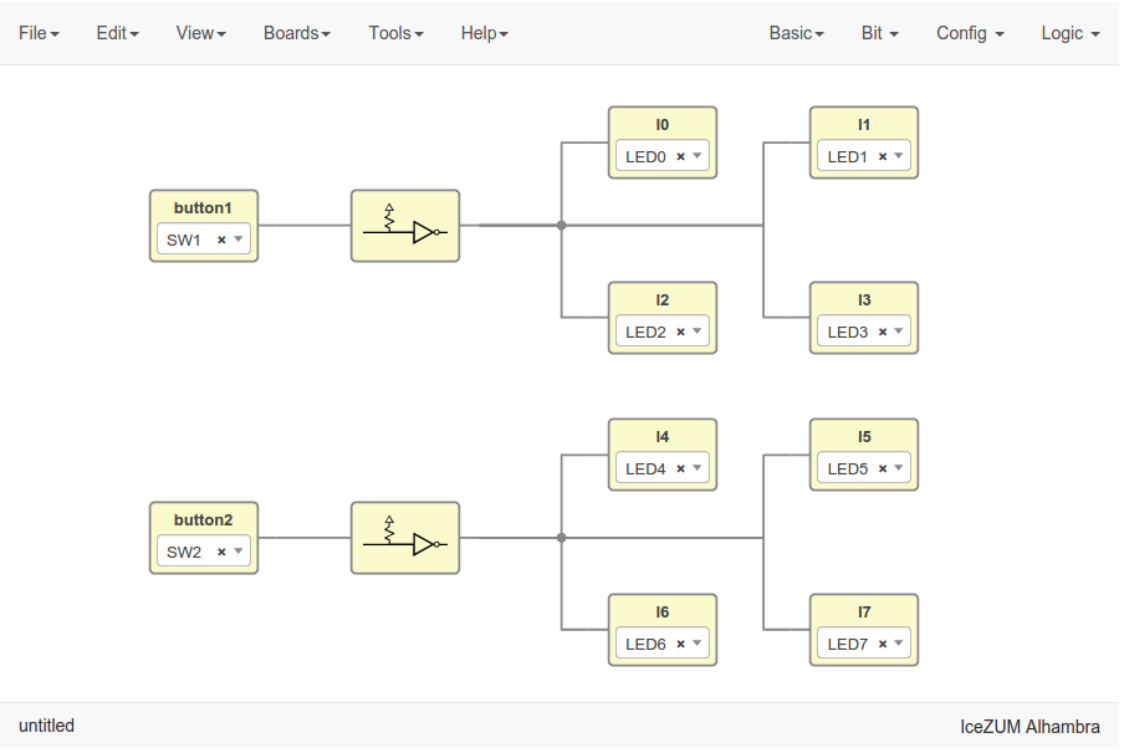


Input / Output pins will become in/out block ports

Icestudio

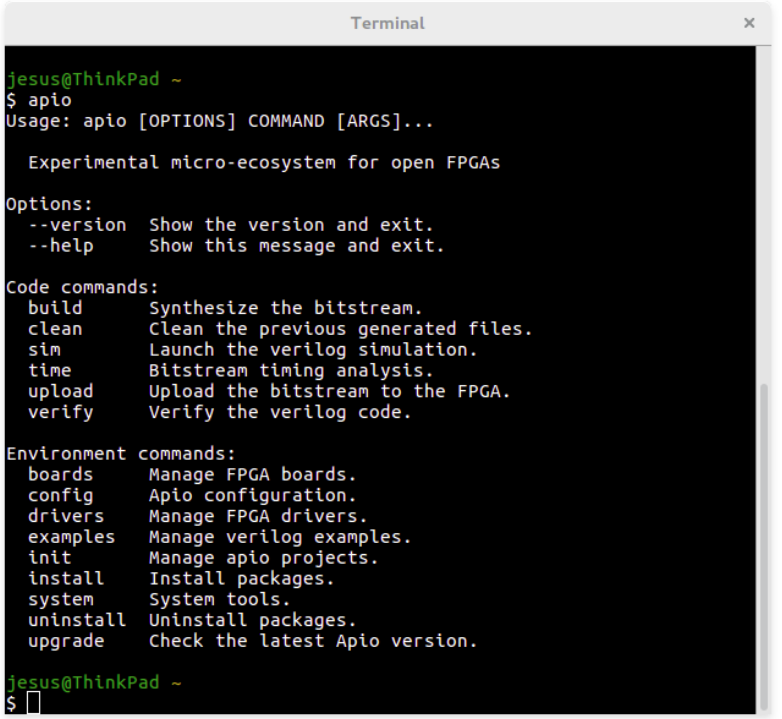
10. **Using buttons**

- Create a project
  File > New project
- Add blocks
  Basic > Input
  Basic > Output
  Config > Pull up inv
- Connect wires
- Upload bit stream
  Tools > Upload



Apio

https://github.com/FPGAwars/apio



Experimental open source **ecosystem** for open FPGAs. Created with Python

Apio

1. **Install**

- Open the console and execute

```
$ pip install -U apio
```

- Check apio

```
$ apio
```

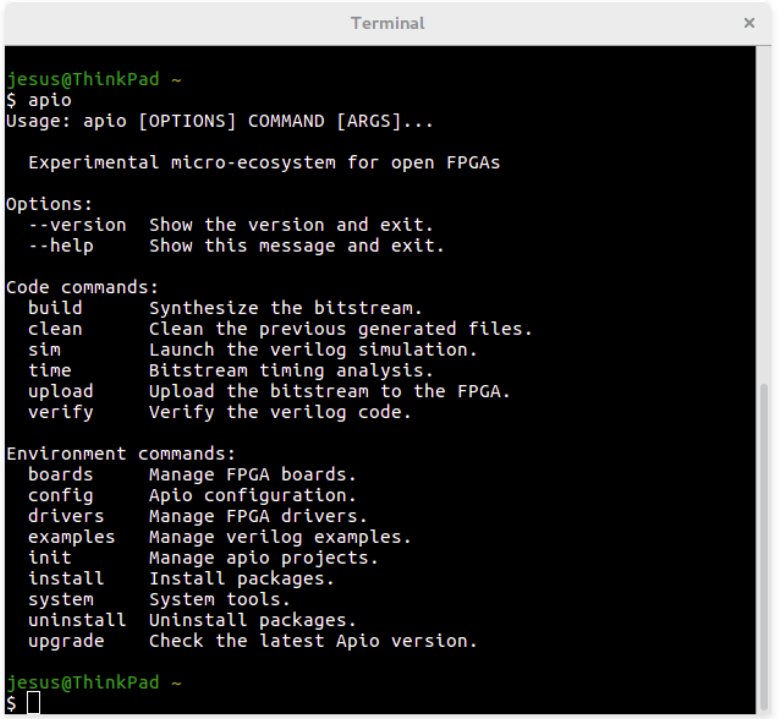
2. **Setup**

- Install toolchain

```
$ apio install --all
```

- Install drivers

```
$ apio drivers --enable
```



Drivers configuration requires administrative privileges
Follow the instructions in each OS

Apio

3. **Hello, world!**

- Load example

  ```
  $ apio examples -d icezum/leds
  ```

- Move to example

  ```
  $ cd icezum/leds
  ```

- Verify and simulate

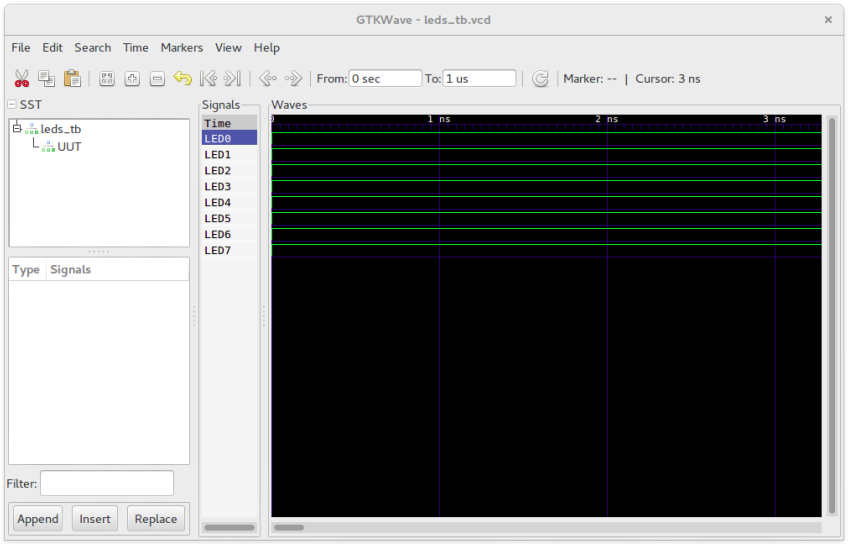
  ```
  $ apio verify $ apio sim
  ```

- Build and upload

  ```
  $ apio build $ apio upload
  ```

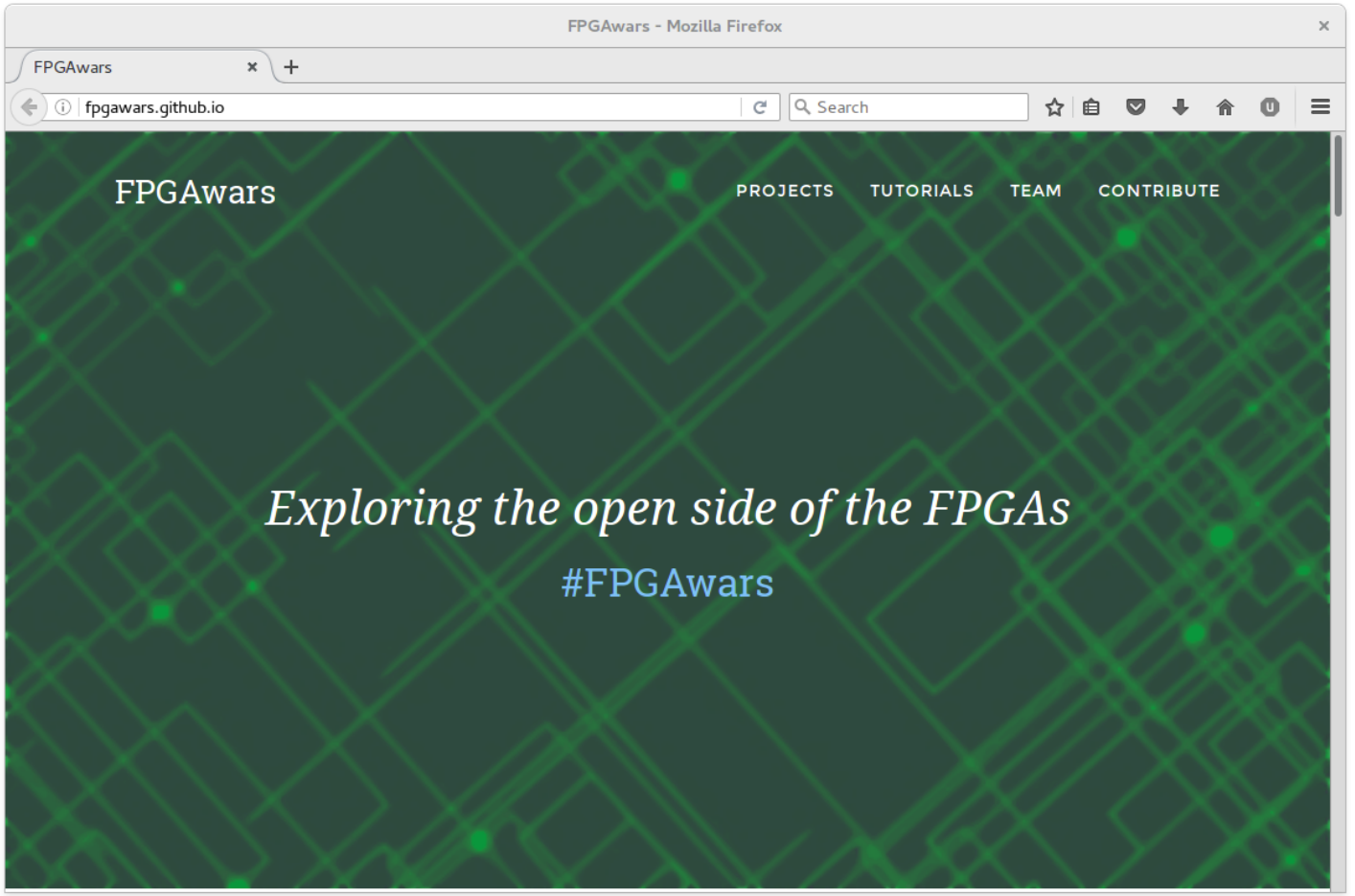
- Time analysis and clean

  ```
  $ apio time $ apio clean
  ```

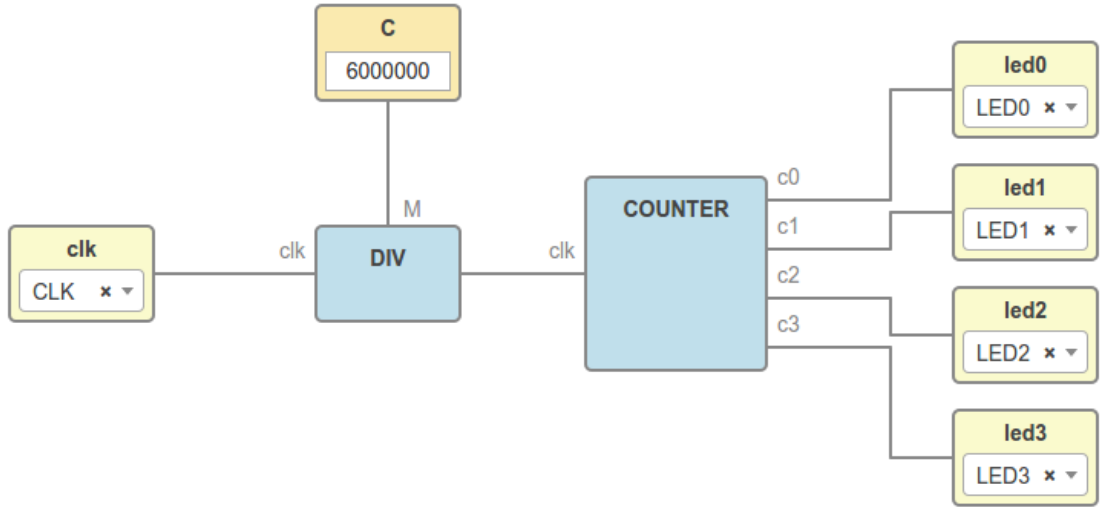


I want more!

http://FPGAwars.github.io



Workshop - Open FPGA tools



Jesús Arroyo Torrens

https://github.com/Jesus89