

Graphic design tools for Open Source FPGAs

Learn about the [Apio](#) and [Icestudio](#) projects



FOSDEM 2018

Jesús Arroyo Torrens

Presentation

Husband and father of two.

Engineer in software, robotics and electronics.

Creator of the open source tools [Apio](#) and [Icestudio](#) to bring FPGA technology to everyone.

Currently working at CARTO, an open source company of Location Intelligence, as a software engineer in Madrid (Spain).



GitHub [@Jesus89](#)

Twitter [@JesusArroyo89](#)

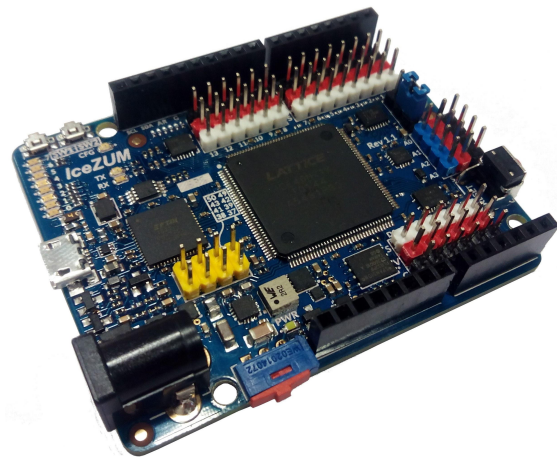
State of the art

Open FPGA boards

FPGA: field-programmable gate array

Open FPGA: FPGA chip that can be used with open source tools (Lattice iCE40)

Open FPGA board: open source electronic board containing an open FPGA as main chip



iCEstick Evaluation Kit	iCE40-HX8K Breakout Board	icoBOARD 1.0
Nandland Go board	IceZUM Alhambra	Kéfir I iCE40-HX4K
CAT Board	BlackIce	TinyFPGA B2

Open FPGA tools

Icarus Verilog*: simulation and synthesis tool

<http://iverilog.icarus.com>

GTKWave*: fully featured wave viewer

<http://gtkwave.sourceforge.net>

Icestorm: Verilog-to-Bitstream flow

<http://www.clifford.at/icestorm>

Yosys: logic synthesis

Arachne-pnr: place and route

Icestorm tools: package and upload

...

Open FPGA tools

Simulation example

```
iverilog -B "/path/to/lib/ivl" -o leds_tb.out "/path/to/cells_sim.v" leds.v leds_tb.v  
vvp -M "/path/to/lib/ivl" leds_tb.out  
gtkwave leds_tb.vcd leds_tb.gtkw
```

Synthesis & Analysis example

```
yosys -p "synth_ice40 -blif hardware.blif" -q leds.v  
arachne-pnr -d 1k -P tq144 -p leds.pcf -o hardware.asc -q hardware.blif  
icetime -d hx1k -P tq144 -C "/path/to/chipdb-1k.txt" -mtr hardware.rpt hardware.asc  
icepack hardware.asc hardware.bin  
icaprogram -d i:0x0403:0x6010:0 hardware.bin
```

Found issues

Build all the tools

- Requires time and knowledge
- Requires the build environment

Manual setup of the drivers

- Manage Serial & FTDI drivers
- Different approaches in each OS

Simulation & Synthesis parameters

- Board parameters
- Configuration paths

Upload a wrong bitstream

- Board is not verified before upload



Possible solutions

Create a high level multi-platform tool to manage every found issue

- Build all the tools
- + Package manager
- Simulation & Synthesis parameters
- + Simulation & Synthesis manager
- Manual setup of the drivers
- + Drivers manager
- Upload a wrong bitstream
- + Upload manager

APIO

Apio



A multi-platform cli toolbox for open FPGAs.

Written in Python.

```
pip install apio
```

Doc: <http://apiodoc.readthedocs.io>

Repo: <https://github.com/FPGAwards/apio>

A screenshot of a terminal window titled "jesus@jesus-ThinkPad: ~". The terminal shows the command "apio" being executed, which displays the usage and help information for the apio tool. The help text includes options like --version and --help, and lists code commands (build, clean, sim, time, upload, verify) and environment commands (boards, config, drivers, examples, init, install, system, uninstall, upgrade).

```
jesus@jesus-ThinkPad: ~  
File Edit View Search Terminal Help  
(venv) jesus@jesus-ThinkPad:~$ apio  
Usage: apio [OPTIONS] COMMAND [ARGS]...  
  
Experimental micro-ecosystem for open FPGAs  
  
Options:  
  --version  Show the version and exit.  
  --help    Show this message and exit.  
  
Code commands:  
  build      Synthesize the bitstream.  
  clean      Clean the previous generated files.  
  sim        Launch the verilog simulation.  
  time       Bitstream timing analysis.  
  upload     Upload the bitstream to the FPGA.  
  verify     Verify the verilog code.  
  
Environment commands:  
  boards     Manage FPGA boards.  
  config     Apio configuration.  
  drivers    Manage FPGA boards drivers.  
  examples   Manage verilog examples.  
  init       Manage apio projects.  
  install    Install packages.  
  system     System tools.  
  uninstall  Uninstall packages.  
  upgrade    Check the latest Apio version.  
(venv) jesus@jesus-ThinkPad:~$
```

Commands

Package manager

Commands: `apio install / uninstall`

Packages: `icestorm, iverilog, system, drivers, gtkwave, examples`

Drivers manager

Commands: `apio drivers`

Options: `--ftdi-enable, --ftdi-disable, --serial-enable, --serial-disable`

Simulation & Synthesis manager

Commands: `apio verify / sim / build / time / clean`

Commands

Upload manager

Commands: `apio upload`

Check platform

Check USB VID & PID

Check FTDI description

Auto-search Serial & FTDI devices

More...

`apio init / config`

`apio boards / examples`

`apio system --lsusb, --lsserial, --lsftdi`

Future work

- Improve examples manager
- Add Project manager
- Support new boards
- ...



Apio IDE

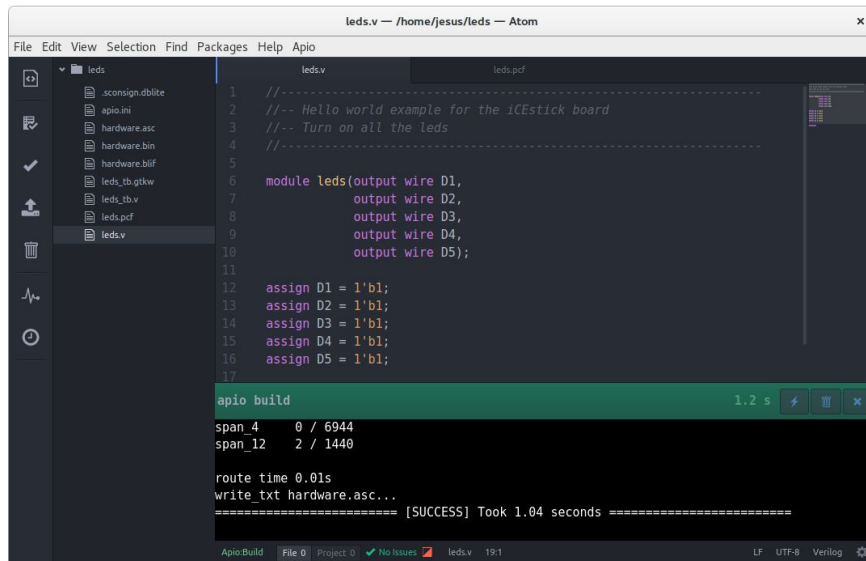


Atom plugin for Apio.

+ Verilog linter

+ Verilog/PCF syntax highlighting

Written in JavaScript and HTML.



Doc: <https://atom.io/packages/apio-ide>

Repo: <https://github.com/FPGAwards/apio-ide>

DEMO |

Nice, but...

I still need to **learn** and **write HDL**

Can I enjoy open FPGAs in an **easy way**?

Let's try to build **one more level**!

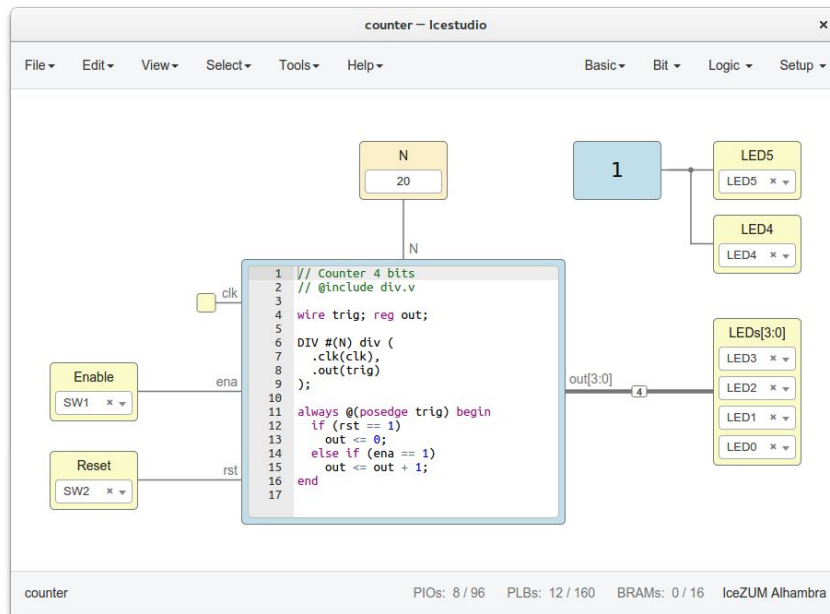


Icestudio



An experimental **graphic editor** for open FPGAs.

Written in **JavaScript** and **HTML/CSS**.



Doc: <http://icestudio.readthedocs.io>

Repo: <https://github.com/FPGAwards/icestudio>

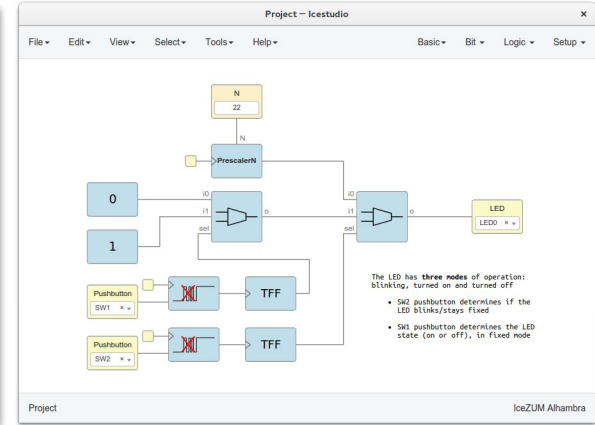
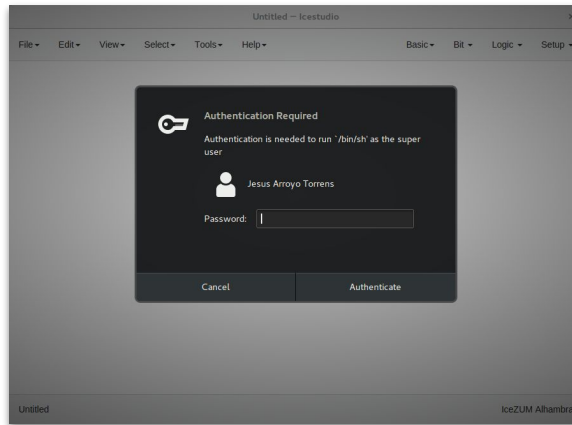
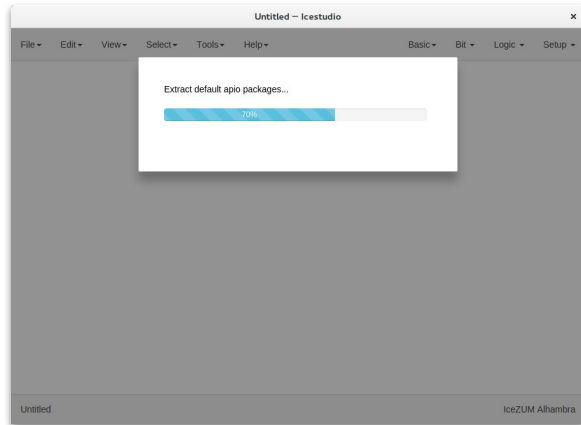
Easy setup



Multi-platform application: Appliance, Windows Installer, Mac OS DMG

Integrated toolchains: includes Apio and all the necessary tools by default

Drivers configuration: step-by-step guide to configure the drivers



Basic blocks



I/O blocks

Input/output ports.

Constant blocks

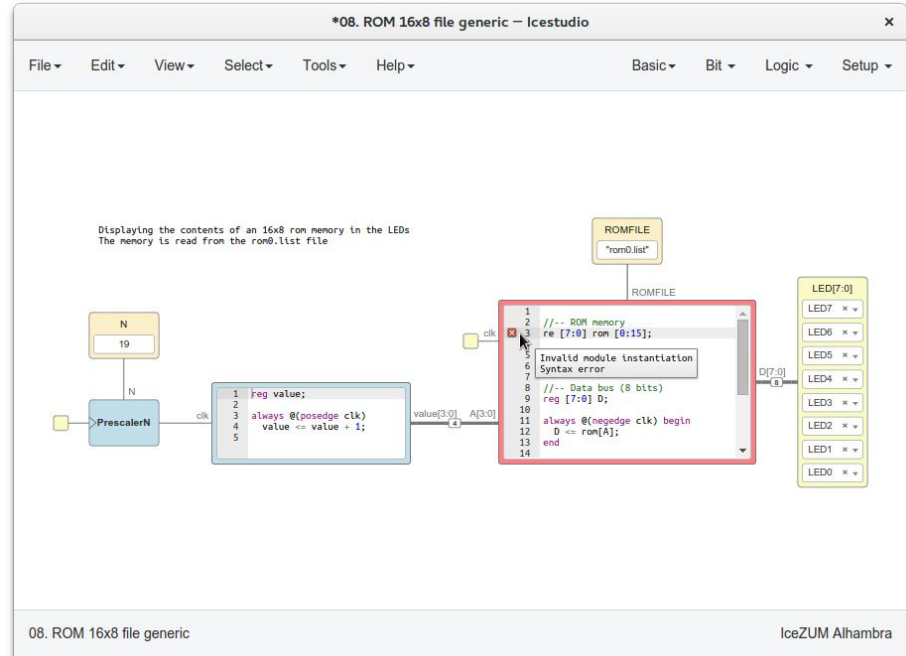
Parameters for other blocks.

Code blocks

Write Verilog code.

Information blocks

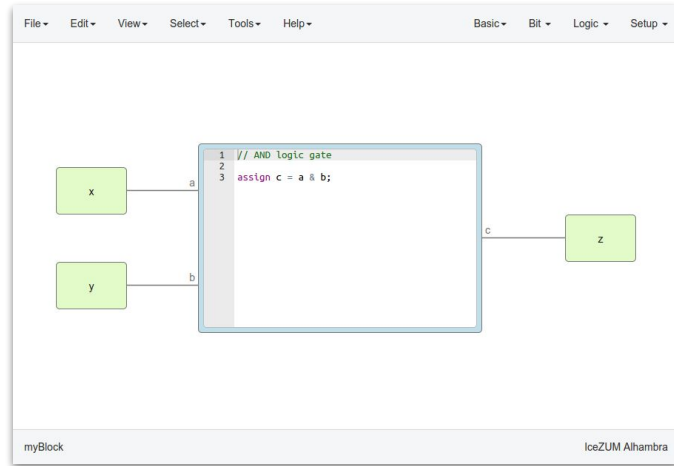
Write documentation in Markdown.



Custom blocks



Duality project / block. Each project can be used as a block (*.ice)



+

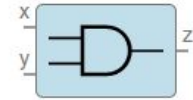


The form for creating a custom block is shown. It includes the following fields:

- Name: AND
- Version: 1.0.0
- Description: AND logic gate
- Author: Jesús Arroyo
- Image: A button to select an image, with a preview of an AND gate symbol.

Below the image field are three buttons: Open SVG, Save SVG, and Reset SVG. At the bottom right are two buttons: OK and CANCEL.

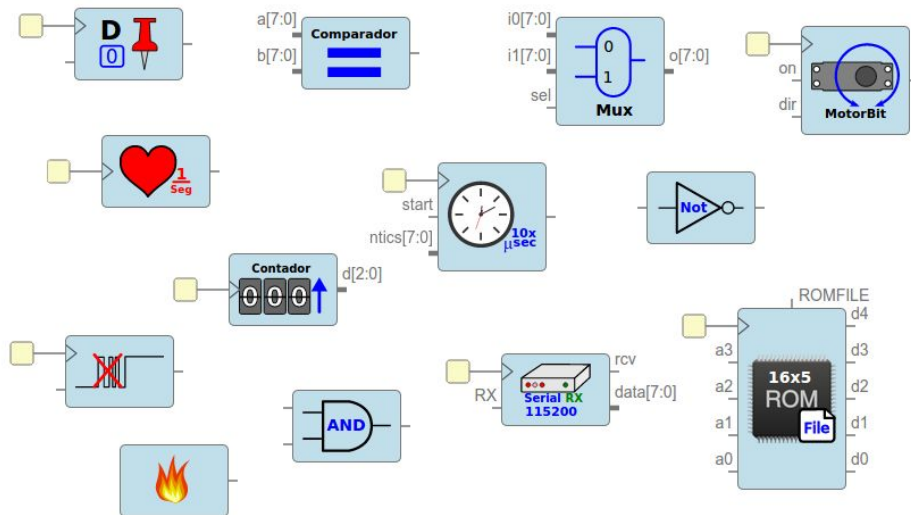
=



Collections



Group of [blocks](#) and [examples](#) with translations distributed in a ZIP file



More...



[Multiple boards support](#): included PCF, pinout
SVG, datasheet

[Multi language](#): English, Spanish, Galician,
Basque, French, Catalan

[Export](#) to Verilog, PCF, Testbench, GTKWave,
BLIF, ASC, Bitstream

[Error detection](#) and management

Include external files: v, vh, list

[Board rules](#) to define the default I/O behavior

Resize text blocks

Full [Undo/Redo](#) for all the components

Select, cut, copy and paste blocks

Pan & zoom

[Multi window](#) application

Remote host configuration

Show [FPGA resources](#)

Block examination

Block tooltips

Take [snapshots](#)

...

Future work



- Add memory and label blocks
- Add parametric blocks
- Add Iterative block edition
- Add testbench editor
- Integrate simulation
- Integrate routes viewer
- Add colored wires
- Support more boards
- New graphic interfaces
- Refactor internal architecture
- ...

DEMO II

FPGAwards

A community to share knowledge about open FPGAs



Web: <http://fpgawars.github.io>

GitHub: <https://github.com/fpgawars>

Group: <https://groups.google.com/forum/#!forum/fpga-wars-explorando-el-lado-libre>

- Around 555 members
- Mostly in Spanish

Thanks!