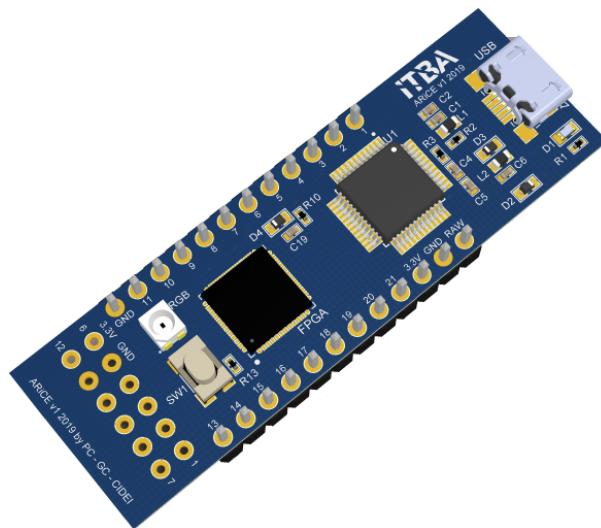


CENTRO DE INVESTIGACIÓN Y DESARROLLO EN ELECTRÓNICA INDUSTRIAL (CIDEI)



ARiCE Plattform

Getting Started Apio



AUTHORS: Dr. Ing. Pablo COSSUTTA - Ing. Gonzalo CASTELLI - Rodrigo DEVESA

CIUDAD AUTÓNOMA DE BUENOS AIRES

2018-2019

Contents

1	Introduction	3
2	Getting started	3
2.1	Downloading the software	3
2.2	Creating a new project	4
2.3	Programming the board	5
2.4	Model simulation	5
3	Pinout information	6
4	Open Source PCB	8
4.1	Bill of Materials	9
4.2	Schematics	9
5	License	13
6	Acknowledgments	13

1 Introduction

This project is an open-source platform, shown on Fig. 1, based on a low cost and low consumption FPGA chip from [Lattice Semiconductor](#), aiming to be used in a wide range of signal processing and control applications, both for education and the industry.

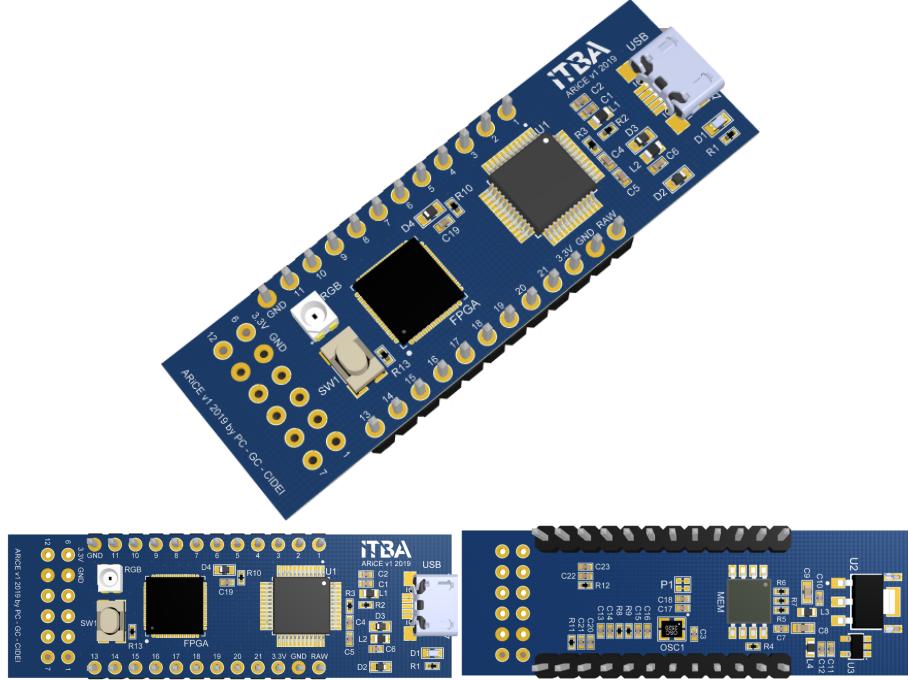


FIGURE 1: Board's view

2 Getting started

Without no previous knowledge, three main steps are necessary to use the board if you are starting from scratch. The first is to download and install the software, the second is to create a new project and include the code files needed and finally load the program to the onboard memory using a USB cable.

2.1 Downloading the software

Before using the board, it is necessary to set up the software to program the Lattice iCE40-UP5K FPGA chip. In this tutorial we will be utilizing the [Apio Open Source Ecosystem for FPGAs](#), which allows to verify the code and program the FPGA through a USB port. [Python 3.7+](#) should be previously installed in order to install Apio.

During Python installation, make sure to select "**Customize Installation**". Then, on the optional features screen, select to install pip, as shown in Fig. 2, as we will be using it to download the Apio packages. [2](#).

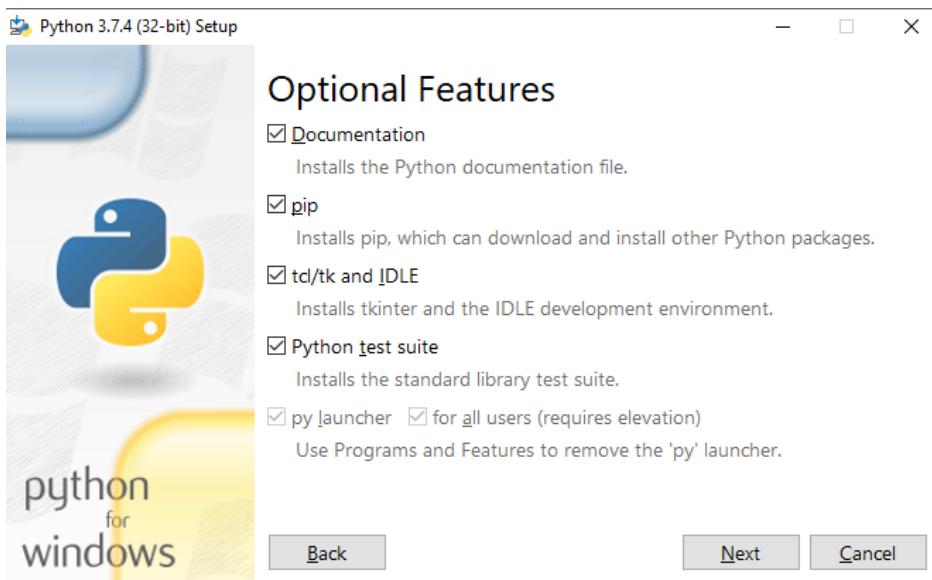


FIGURE 2: Setting up the Python installation

Once Python is successfully installed along with Pip, you can install Apio. On Windows, open your command line console and type:

```
pip install -U apio
```

When it's finished, you will need to install Apio's packages. Use the following command:

```
apio install --all
```

This installation might take a few minutes.

2.2 Creating a new project

After having installed the software, the next step is to create a new folder for the project and start writing the code. An example code that uses the onboard RGB LED is available for download at the GitHub repository of this project. After downloading the files, place them in an easily accessible folder from where we will run the code.

Use the `cd` command followed by the path to your project's folder to change directory to that folder. For example, if your project is on a folder called "fpga_project" in your Desktop, then use:

```
cd Desktop/fpga_project
```

Once in the right folder, you need to generate the right `.ini` configuration file for this board, as Apio supports multiple different FPGA platforms. To do this, use the command:

```
apio init --board upduino2
```

Now, to make sure the project is set up correctly, you could use the command:

```
apio build
```

The folder should contain at least 3 files, one .v file with the program's code itself, one .pcf file with pinout information for the program, and the .init file we just created. If the build process is successful, you can move on to programming the FPGA board.

2.3 Programming the board

Now that the project is fully set up, we can load the program into our board to run it. Connect the FPGA board to the PC using the USB cable, wait a few seconds for Windows to configure the drivers, then use the following command to program the board:

```
apio upload
```

The process will take a few seconds, and if everything went right, the FPGA board should start running the program and its onboard LED should start flashing in colors.

2.4 Model simulation

To simulate the model the following command is used:

```
apio sim
```

The [GTKWave](#) graphic environment should be load with the simulation data.

3 Pinout information

The board has multiple I/O pins, as well as power supply pins, distributed along the side and the front of the PCB. The next list gives a short description of the signals:

- IOT_XX are general I/O pins. In user mode, after configuration, these pins can be programmed as I/O in user function in the top (xx = I/O location)
- IOB_XX are general I/O pins. In user mode, after configuration, these pins can be programmed as I/O in user function in the bottom (xx = I/O location)
- RAW VCC. The input voltage to the FPGA board when it's using an external power source. The board can be supplied with power either from the USB connector or the RAW VCC pin. The nominal supply voltage is 5V. (Minimum is 4.5V and maximum is 6V)
- 3.3V. A 3.3 volt supply generated by the on-board regulator. Maximum recommended current draw from this pin is 500 mA
- GND. Ground pins
- Signal names with G1, G3 and G6 suffixes may be used as a General I/O or as a Global input used for high fanout, or clock/reset net. These pins drive the GBUF1, GBUF3 and GBUF6 global buffers respectively

Table 1 shows the mapping between the signals and the board pin numbers.

TABLE 1: Connections

Board Pin	FPGA Pin	Signal Name	Board Pin	FPGA Pin	Signal Name
Side Pins			Front Pins		
1	20	IOB 25B G3	1	4	IOB 8A
2	21	IOB 23B	2	3	IOB 9B
3	23	IOT 37A	3	47	IOB 2A
4	25	IOT 36B	4	44	IOB 3B G6
5	26	IOT 39A	5	-	GND
6	27	IOT 38B	6	-	3.3V
7	31	IOT 42B	7	48	IOB 4A
8	32	IOT 43A	8	45	IOB 5B
9	34	IOT 44B	9	38	IOT 50B
10	36	IOT 48B	10	42	IOT 51A
11	37	IOT 45A G1	11	-	GND
12	-	GND	12	-	3.3V
13	2	IOB 6A	Board Pin	FPGA Pin	Signal Name
14	6	IOB 13B	Not Connected		
15	9	IOB 16A	-	43	IOT 49A
16	10	IOB 18A	-	46	IOB 0A
17	11	IOB 20A	-	28	IOT 41A
18	12	IOB 22A	LED Color		
19	13	IOB 24A	FPGA Pin		
20	18	IOB 31B	Signal Name		
21	19	IOB 29B	Onboard RGB LED		
22	-	3.3V	Blue	39	RGB0
23	-	GND	Green	40	RGB1
24	-	RAW VCC	Red	41	RGB2

The differential pairs are shown in Table 2. They are grouped together and labeled in colors, white is positive and light blue is negative.

TABLE 2: Differential Pairs

Board Pin	FPGA Pin	Signal Name
3	23	IOT 37A
4	25	IOT 36B
5	26	IOT 39A
6	27	IOT 38B
8	32	IOT 43A
7	31	IOT 42B
11	37	IOT 45A G1
9	34	IOT 44B
2	21	IOB 23B
18	12	IOB 22A
2	3	IOB 9B
1	4	IOB 8A
4	44	IOB 3B G6
3	47	IOB 2A
8	45	IOB 5B
7	48	IOB 4A
10	42	IOT 51A
9	38	IOT 50B

4 Open Source PCB

The board files, circuit schematics and list of components are available in the GitHub repository. Altium project and Gerber files are available. The board is also published in open source EDAs just as [Circuit Maker](#) and [KiCad](#).

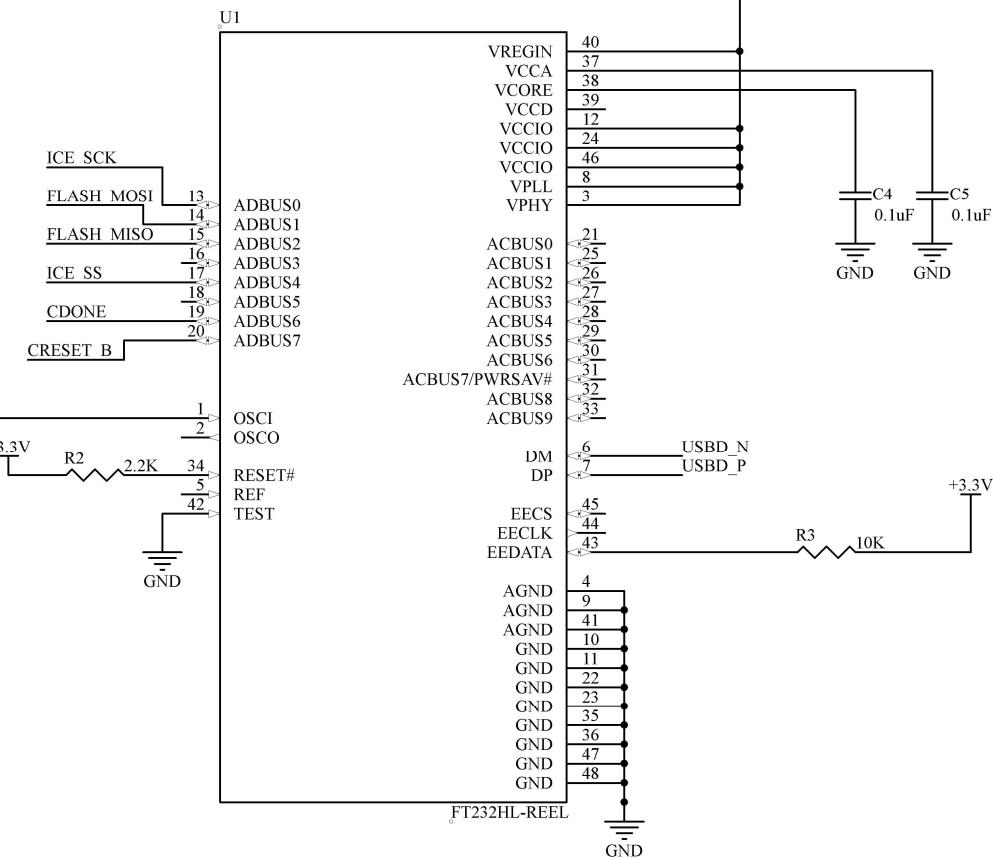
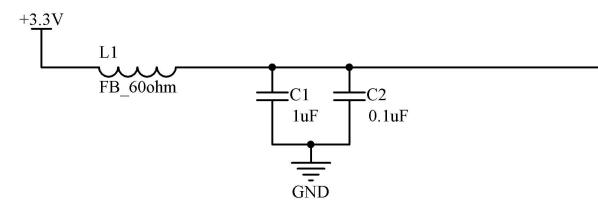
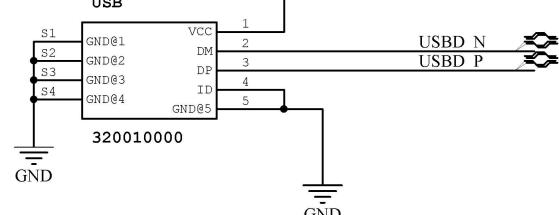
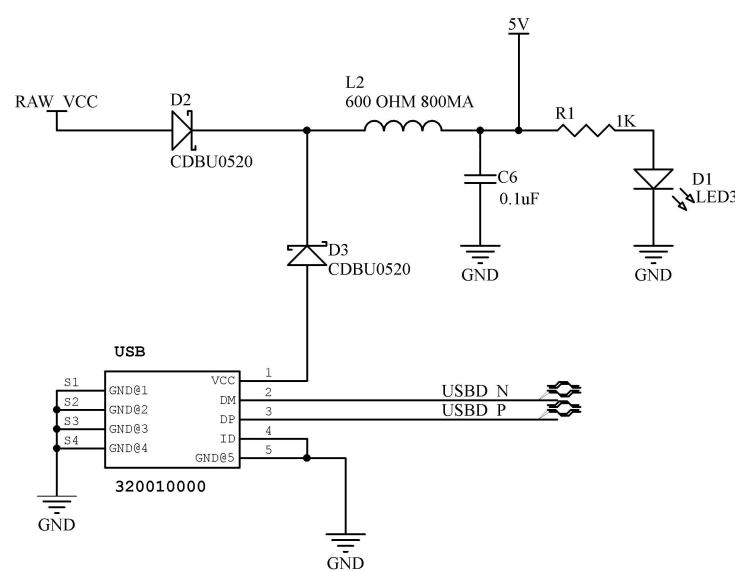
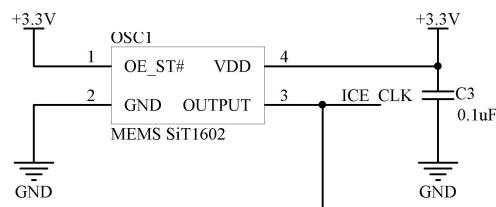
4.1 Bill of Materials

Table 3 includes all the components in the FPGA Board. Follow the hypertext links to the component's datasheets for further information.

TABLE 3: Bill of materials

Designator	Qty	Manufacturer Part Number	Value	Footprint
C2, C3, C4, C5, C6, C10, C12, C14, C16, C18, C19, C21, C23	13	CL05B104KA5NNNC	0.1uF	0402
C8, C9	2	CC0603KRX5R8BB105	1uF	0603
C1, C7, C11, C13, C15, C17, C20, C22	8	CGB2A1JB1E105M033BC	1uF	0402
R1	1	RC0402JR-071KL	1KΩ	0402
R2	1x	RC0402FR-072K2L	2.2kΩ	0402
R3, R4, R5, R6, R7, R13	6	RC0402JR-0710KP	10kΩ	0402
R8, R9, R10, R11, R12	5	AC0402JR-071RL	1Ω	0402
L1	1	HI0603P600R-10	10mH	0603
L2, L3, L4	3	BLM18HE601SN1D	10mH	0603
RGB	1	CLMVC-FKA-CL1D1L71BB7C3C3	4-PLCC	
D1	1	LTST-C190TBKT		0603
U1	1	FT232HL-REEL	48-LQFP (7x7)	
U2	1	TLV1117LV33DCYR		SOT-223-4
U3	1	LP5907MFX-1.2/NOPB		SOT-23-5
OSC1	1	SIT1602AC-73-33S-12.000000G		2.0X1-6MM
FPGA	1	ICE40UP5K-SG48ITR50	48-QFN- 7X7	
USB	1	10118193-0001LF	Micro B SMD	USB
MEM	1	W25Q32JVSSIQ		SOP8
D2, D3, D4	3	CDBU0520		0603/SOD- 523F
SW1	1	PTS810 SJM 250 SMTR LFS		SW4-SMD

4.2 Schematics



3

Title		FTDI Module	ITBA - CIDEI - PC - GC	Pablo Cossutta Gonzalo Castelli
Size		Number		Revision 2.0
A4				
Date:		12/14/2018	Sheet of	
File:		C:\Users...\FTDI.SchDoc	Drawn By:	

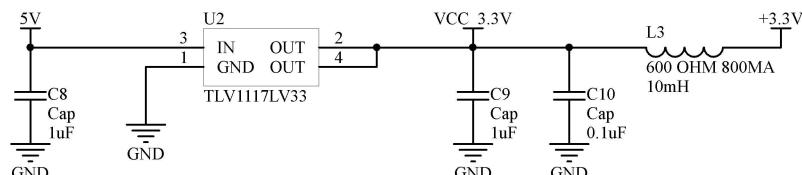
1

2

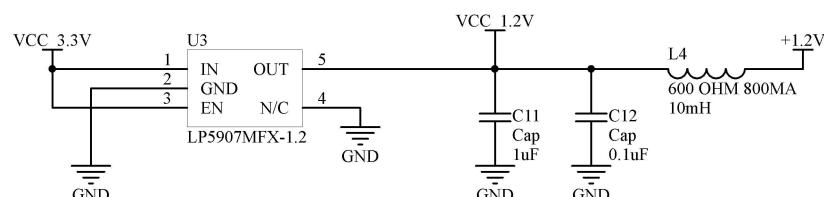
3

4

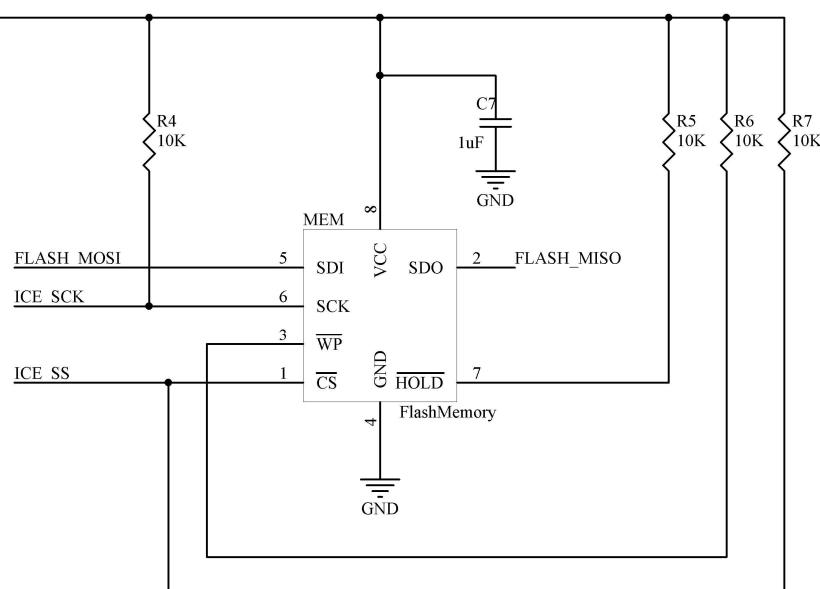
A



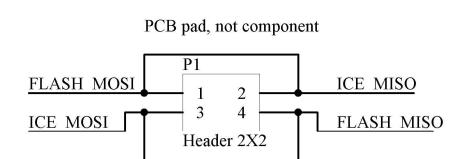
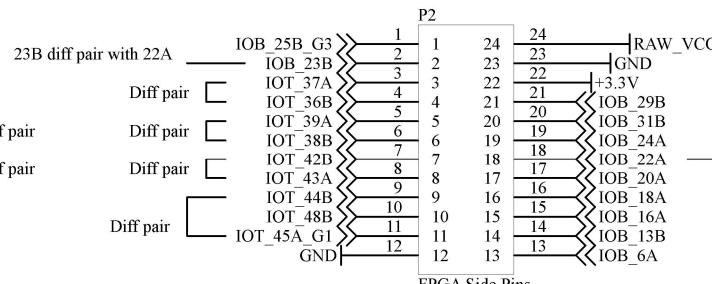
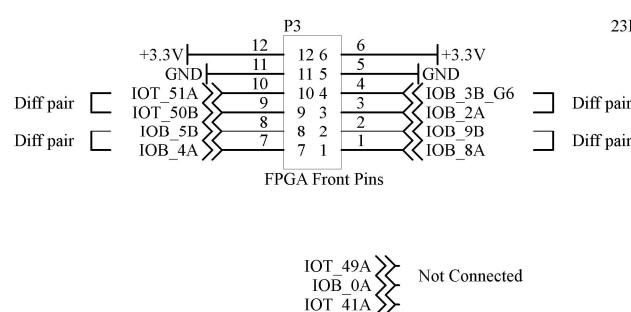
B



+3.3V



C



To program FLASH (default, wired) =

To program iCE FPGA cut traces and solder ||

Title		FLASH Memory, Supply, Connectors ITBA - CIDEI - PC - GC	Pablo Cossutta Gonzalo Castelli
Size	Number		Revision 2.0
A4			
Date: 12/14/2018	Sheet of 1	File: C:\Users...\Misc.SchDoc	Drawn By:

1

2

3

4

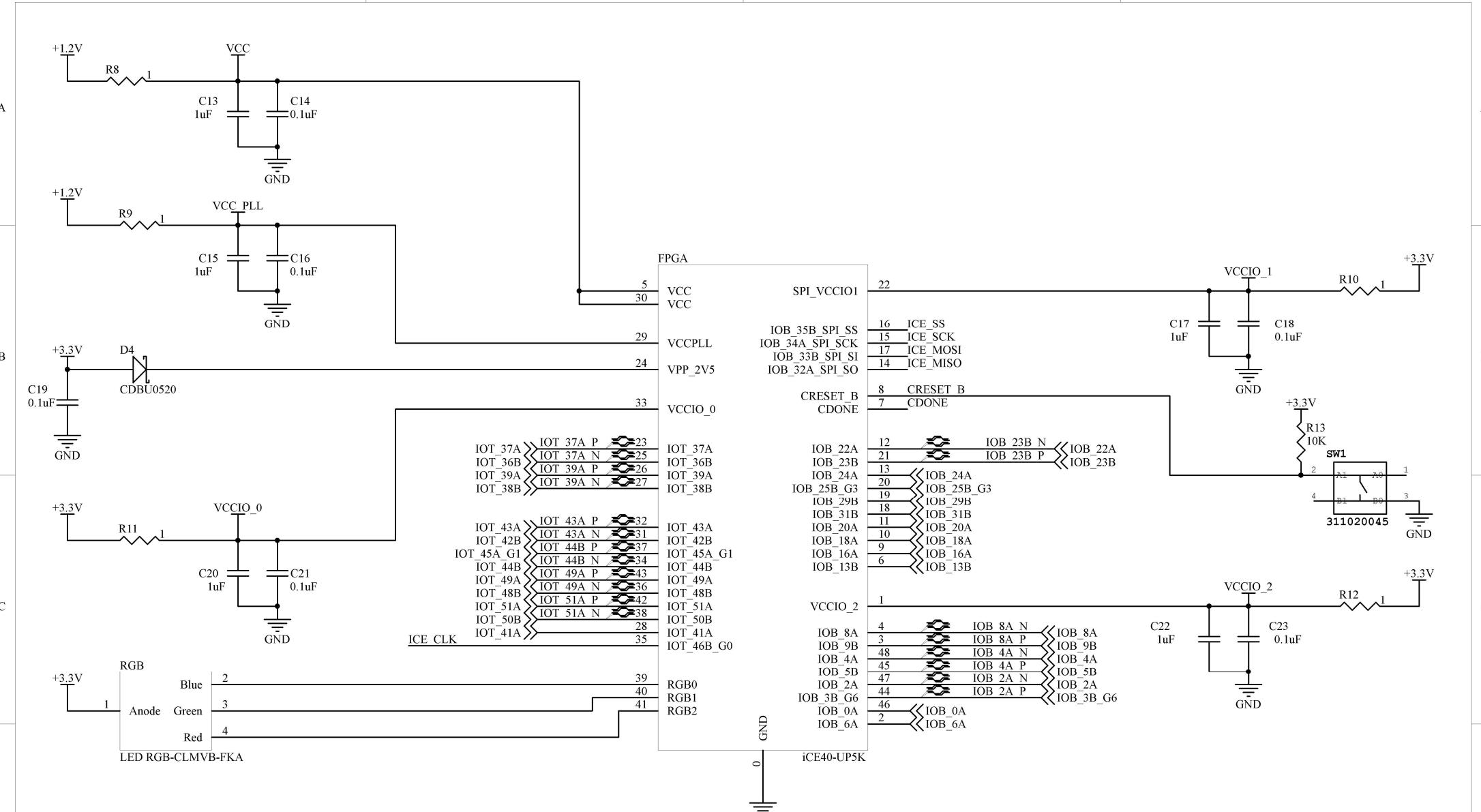
A

B

C

D

1 2 3 4



Title		FPGA	ITBA - CIDEI - PC - GC	Pablo Cossutta Gonzalo Castelli
Size		Number	Revision	
A4			2.0	
Date:		12/14/2018	Sheet of	
File:		C:\Users...\FPGA.SchDoc	Drawn By:	

1 2 3 4

5 License

This project is licensed under a Creative Commons Attribution Share-Alike license, meaning that you are free to use and adapt it for your own needs without asking for permission or paying a fee, even for commercial purposes, as long as you give appropriate credit and release the design under the same license. Visit the [Creative Commons Website](#) for further information.

6 Acknowledgments

The circuit was inspired by the official manuals and toolchain of the [iCE40 UltraPlus Breakout Board](#) from Lattice, and the documentation of the iCE40UP5K FPGA chip.