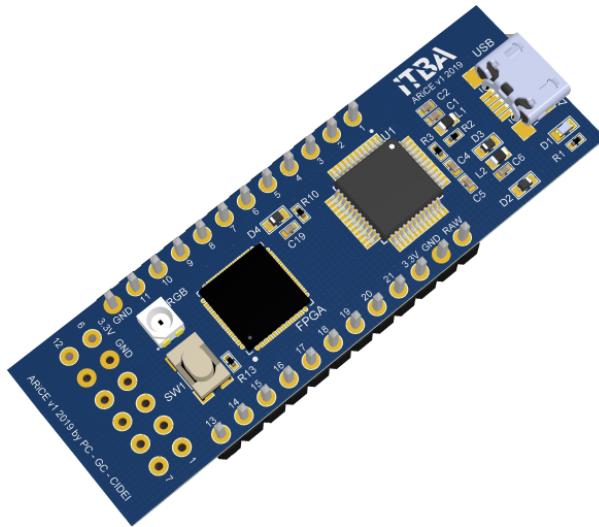


# CENTRO DE INVESTIGACIÓN Y DESARROLLO EN ELECTRÓNICA INDUSTRIAL (CIDEI)



# ARiCE Plattform

## Getting Started Radiant



**AUTHORS:** Dr. Ing. Pablo COSSUTTA - Ing. Gonzalo CASTELLI

# CIUDAD AUTÓNOMA DE BUENOS AIRES

## 2018-2019

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Getting started</b>	<b>3</b>
2.1	Download the software . . . . .	3
2.2	Create a new project . . . . .	4
2.3	Board Programming . . . . .	5
<b>3</b>	<b>Pinout information</b>	<b>7</b>
<b>4</b>	<b>Open Source PCB</b>	<b>9</b>
4.1	Bill of Materials . . . . .	10
4.2	Schematics . . . . .	10
<b>5</b>	<b>License</b>	<b>14</b>
<b>6</b>	<b>Acknowledgments</b>	<b>14</b>

# 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.

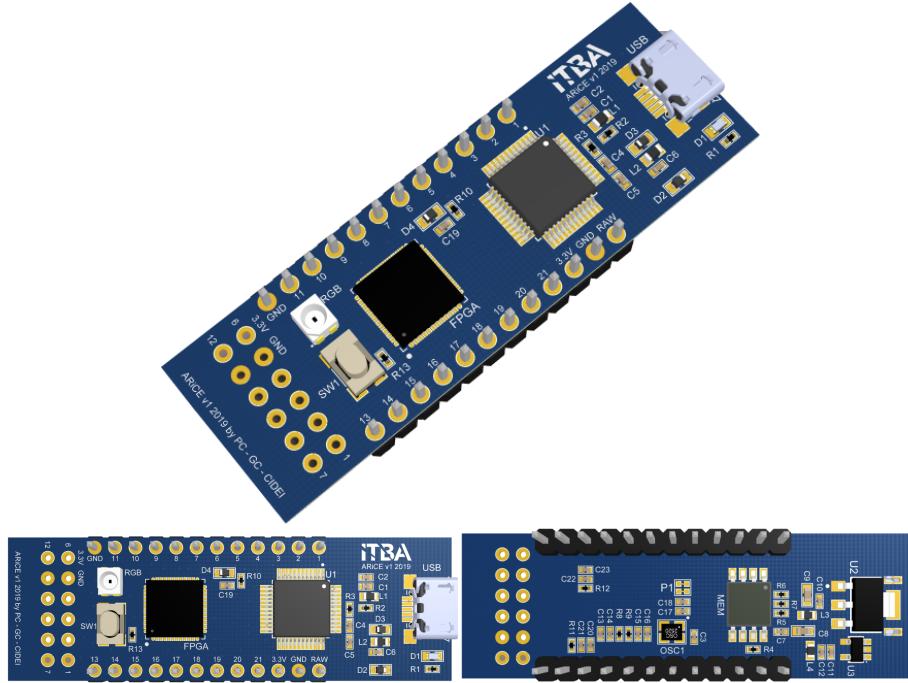


FIGURE 1: Board's view

## 2 Getting started

Three main steps are necessary to use the board if you are starting from scratch. The first is to download the software and get a license, 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 Download the software

Before using the board, it is necessary to set up the software to program the Lattice iCE40-UP5K FPGA chip. It is recommended to use [Lattice Radiant software](#), which supports iCE40 UltraPlus and offers all the best in class tools and features to develop edge applications effectively and efficiently. Free registration is required to download the software, and a license can be requested with no charge after providing the MAC address of the computer it will run on. This development also supports the [iceStorm project](#), Open Source toolchain by Clifford Wolf although its usage is not covered in this document.

## 2.2 Create a new project

After having installed and licensed the software, the next step is to create a new project in the Lattice Radiant Software and start writing HDL 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, open the Lattice Radiant application and go to:

File → Open → Project...

Browse the folder you downloaded from GitHub and select the `getting_started.rdf` file. You can now jump to the next section. To learn how to manually create a new project, click on the "New Project" icon in the start page, or by going to:

File → New → Project...

Follow the instructions of the wizard, and name the project "getting\_started". After clicking "Next", a new window will ask for source files. Click the "Add Source..." button and load the Verilog File you downloaded from the GitHub repository called "top.v". Check both boxes at the bottom of the window to generate all the necessary files in the project folder. The window should look like Fig. 2.

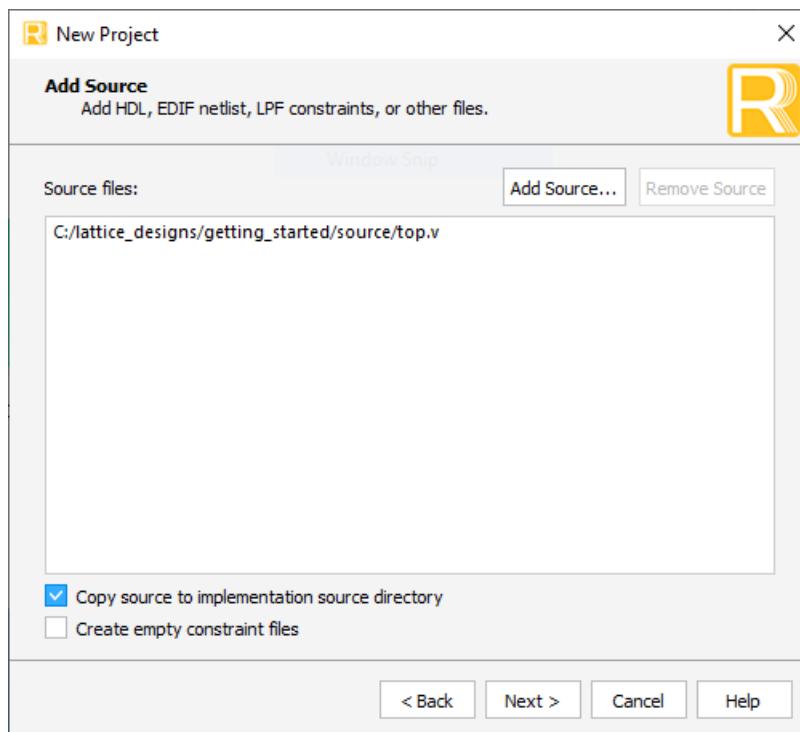


FIGURE 2: Adding source files to the new project

Click "Next", and select the "iCE40UP5K" device. In the package pull-down menu, choose "SG48" and in the Part Number menu, pick "iCE40UP5K-SG48I". The window should look like Fig. 3.

Then, click "Next" and "Finish" to conclude with the wizard. Now, replace the content of the `impl_1.ldc` and `impl_1.pdc` files with the ones from our example in the GitHub repository. The

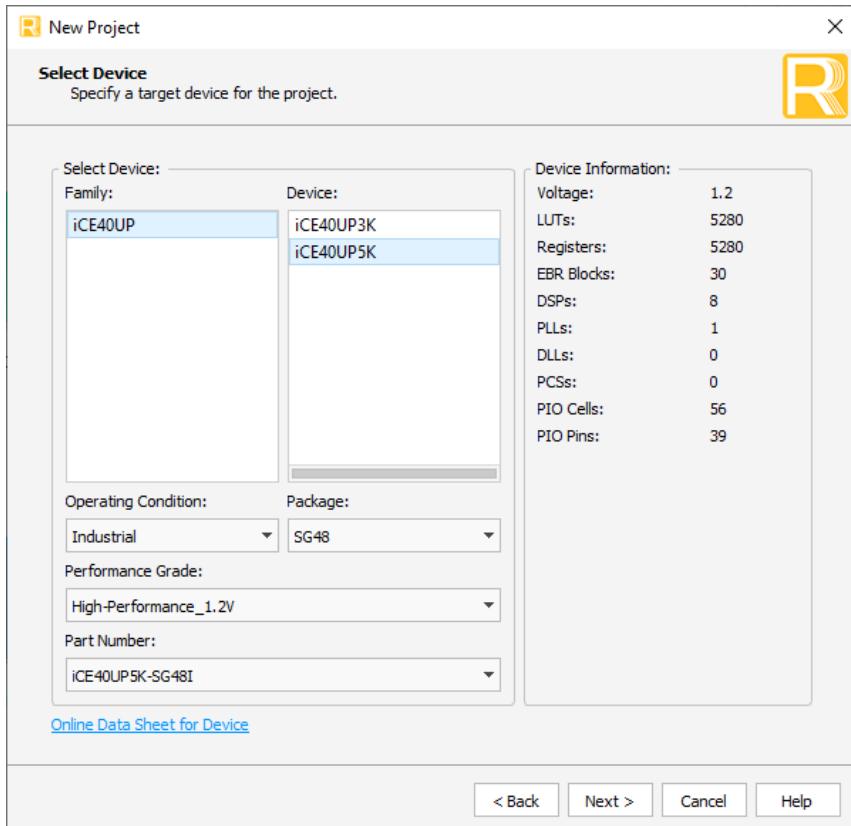


FIGURE 3: Selecting the device

project has now been set up, and the last step remaining is to load the code to the memory of the board.

### 2.3 Board Programming

Now that the project is fully set up, it is necessary to synthesize the design to create the export files. Click on the green play button on the upper-left corner to do so. The process should succeed, even though warnings will appear. Connect the FPGA board to the PC using the USB cable. To set up the device for programming, go to:

Tools → Programmer

In the RADIANT Programmer window, the iCE40UP5K device should be displayed in the list. Select it by clicking on it, and go to:

Edit → Device Properties...

Configure the window so it looks like Fig. 4. Do not forget to include the programming file path, which was generated when doing the synthesis and which has a ".bin" extension. In our case, the file is `getting_started_impl1.bin`.

Click "OK" to exit the window and to finally load the program to the board's memory go to:

Run → Program Device

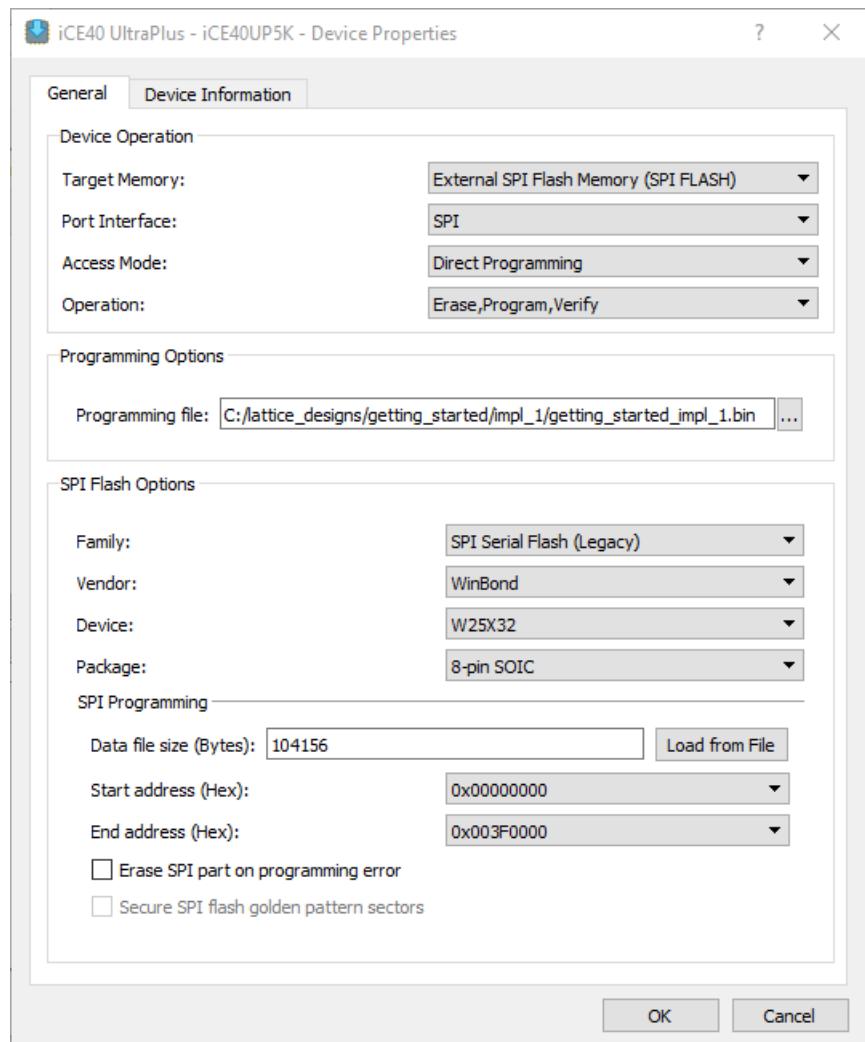


FIGURE 4: Programming properties

If done successfully, the RGB LED of the FPGA board should be flashing in colors.

### 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			
19	13	IOB 24A			
20	18	IOB 31B	LED Color	FPGA Pin	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](#).

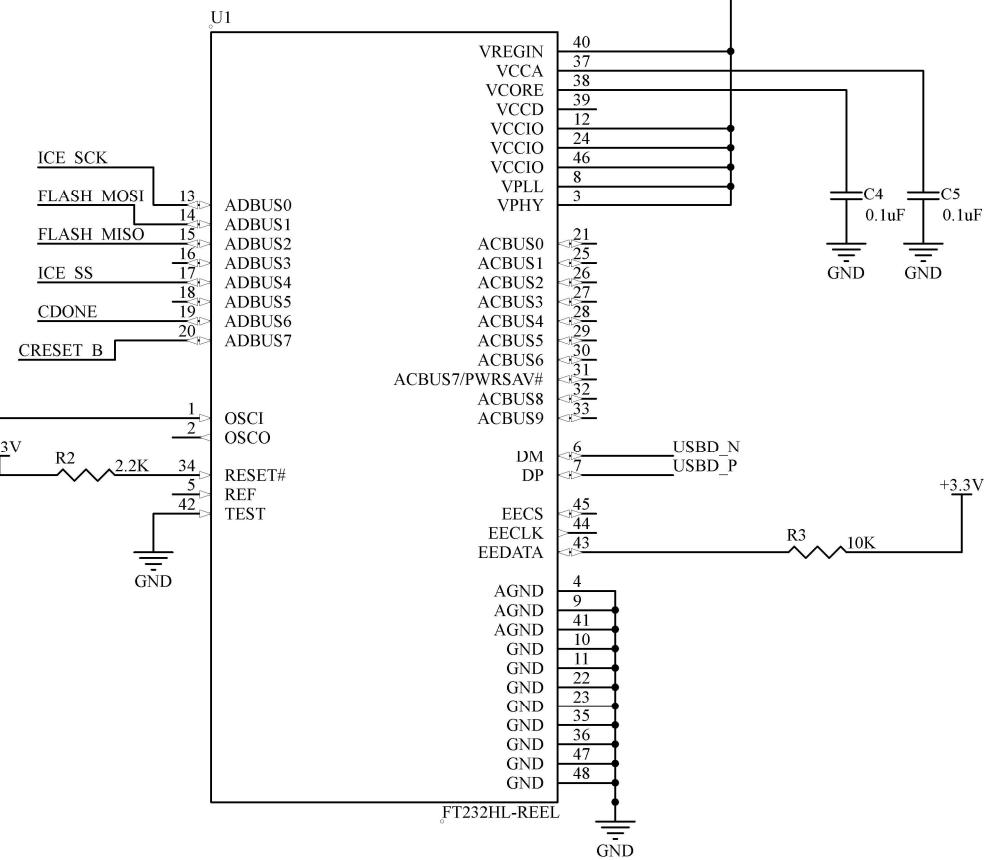
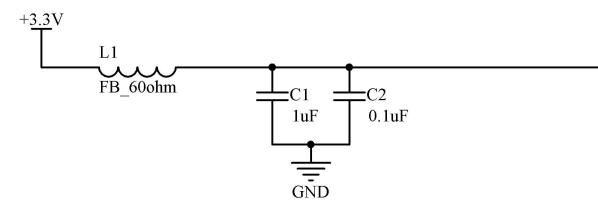
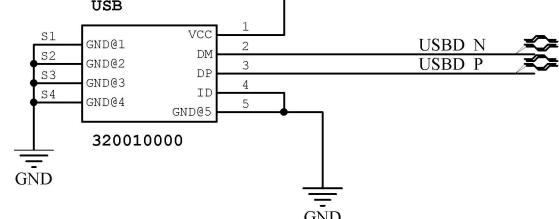
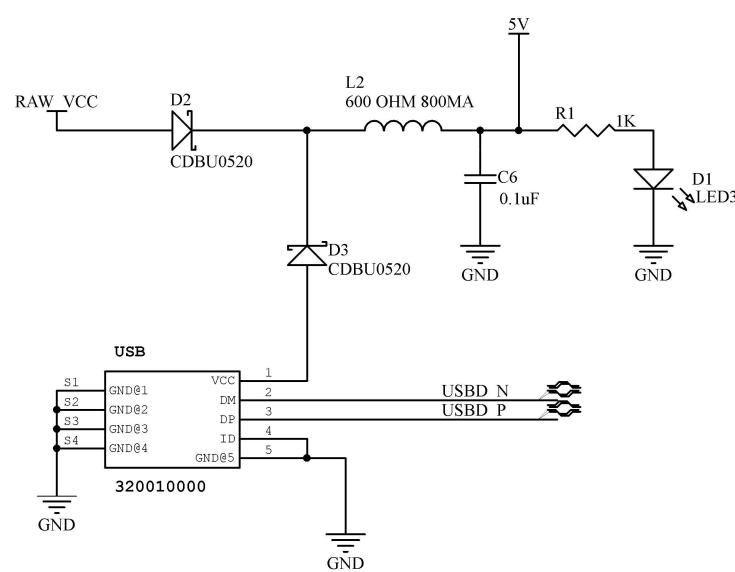
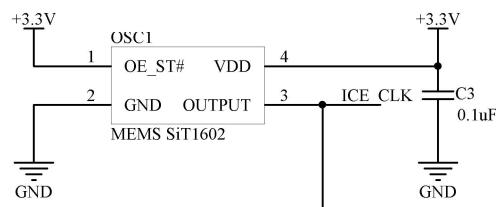
## 4.1 Bill of Materials

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

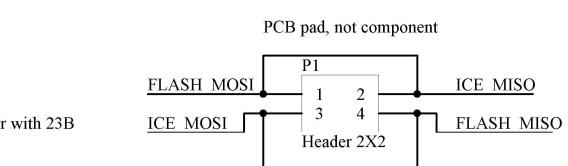
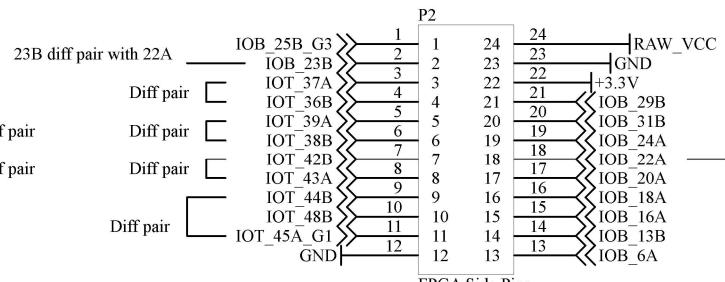
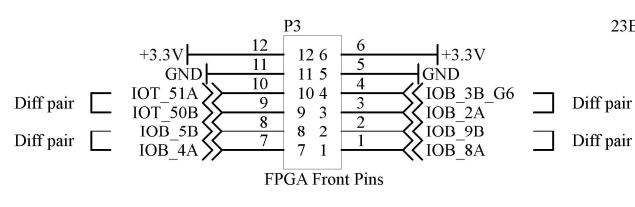
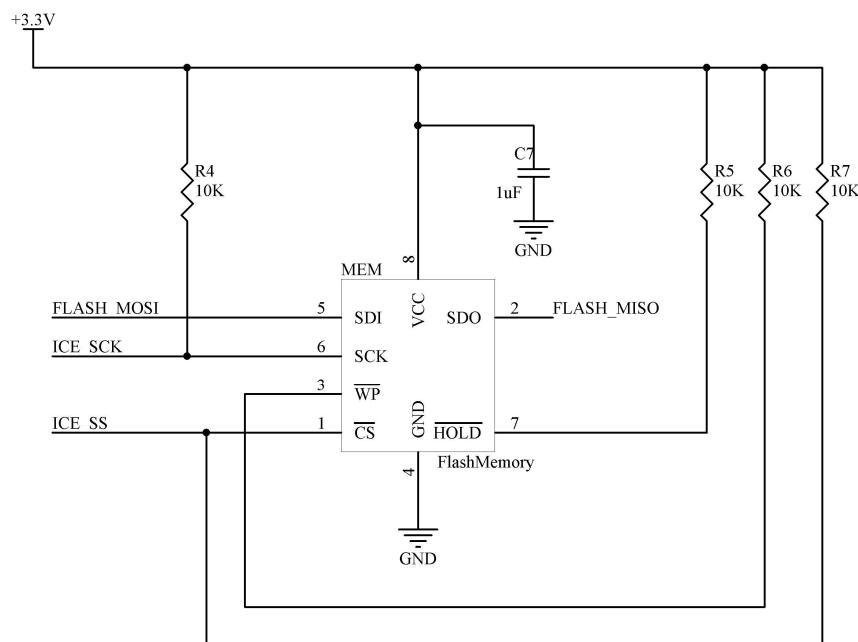
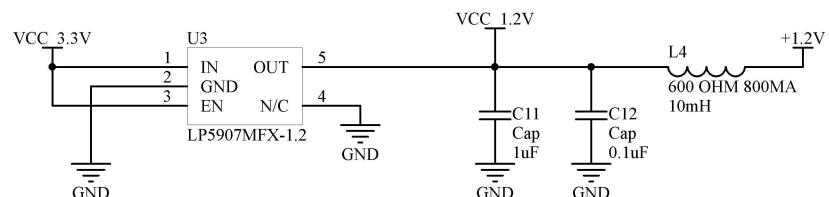
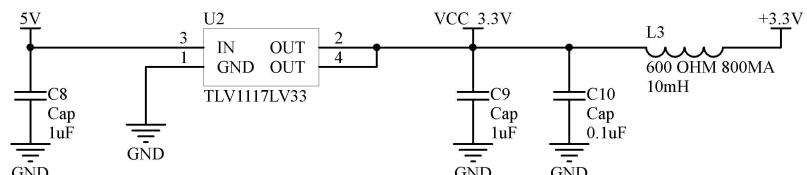
## 4.2 Schematics



**D**

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

Date: 12/14/2018  
File: C:\Users...\FTDI.SchDoc

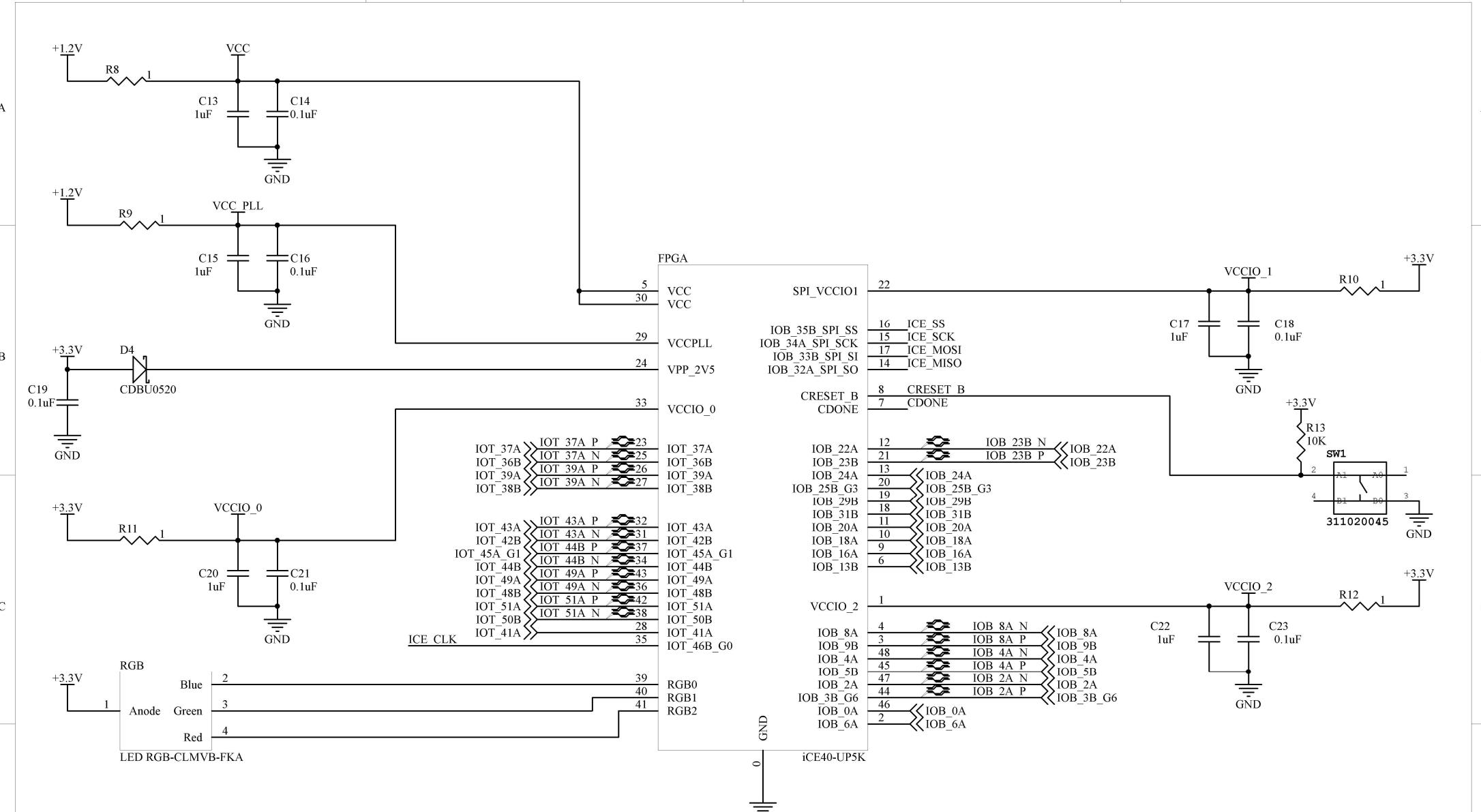


To program FLASH (default, wired) =

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
File:	C:\Users\...\Misc.SchDoc	Drawn By:	

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.