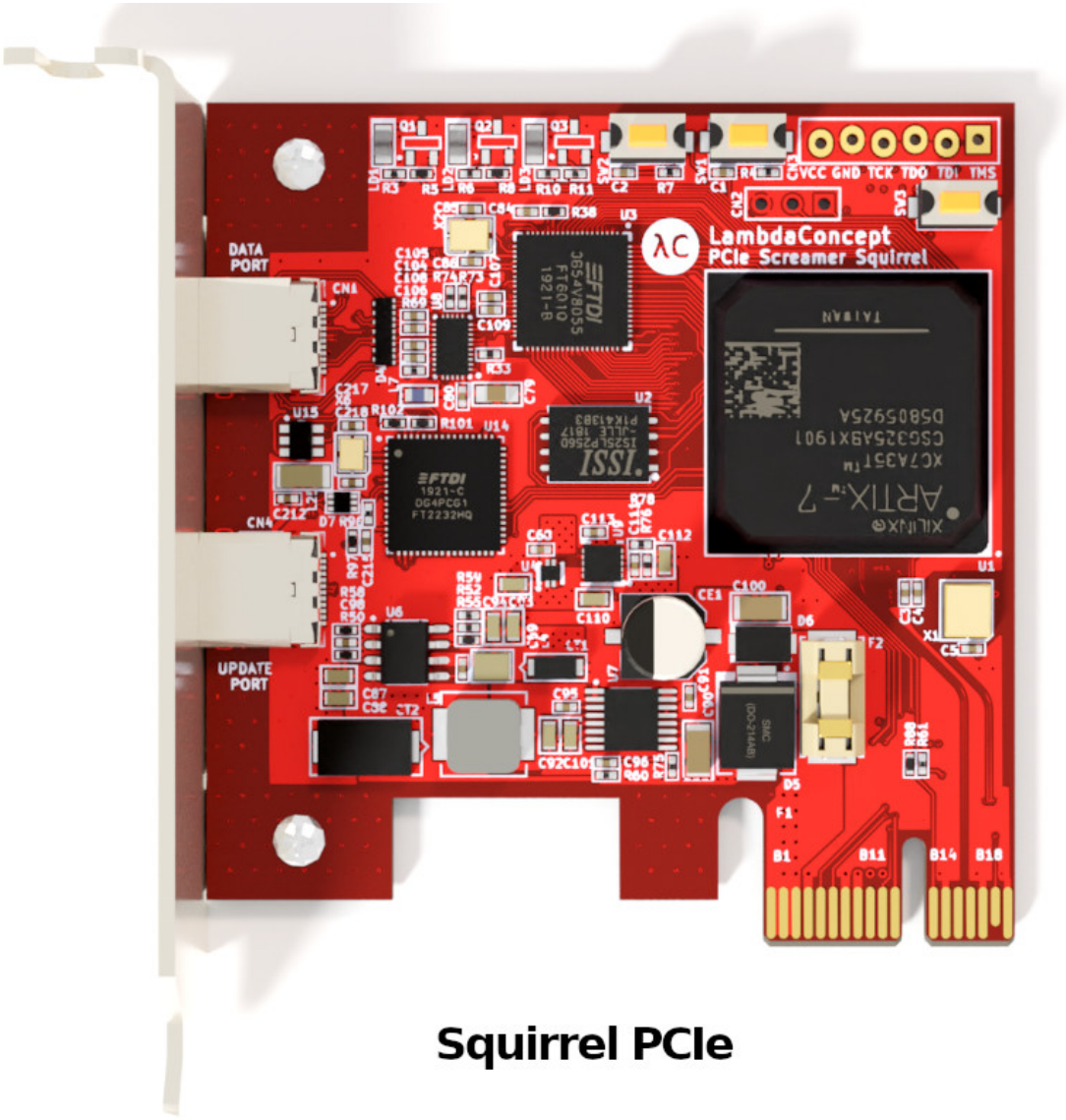



Squirrel PCIe

# Screamer Documentation

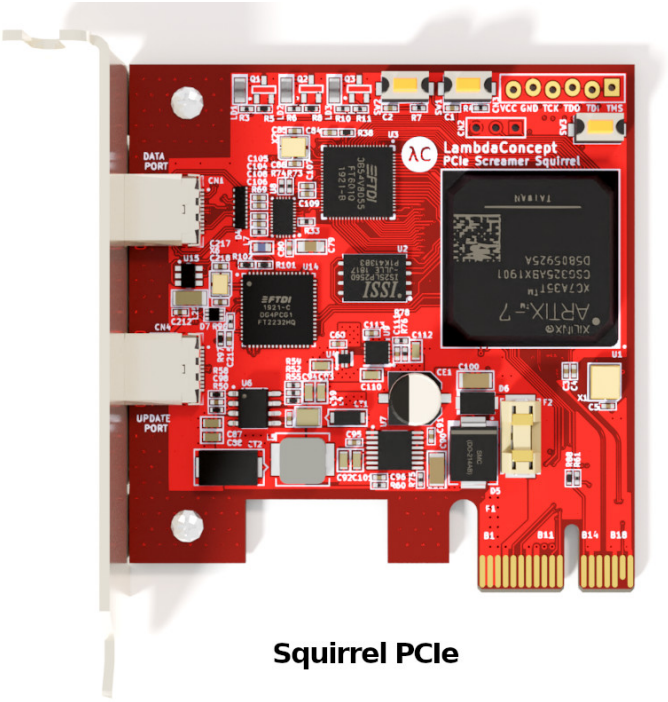
## LambdaConcept

**Aug 04, 2022**

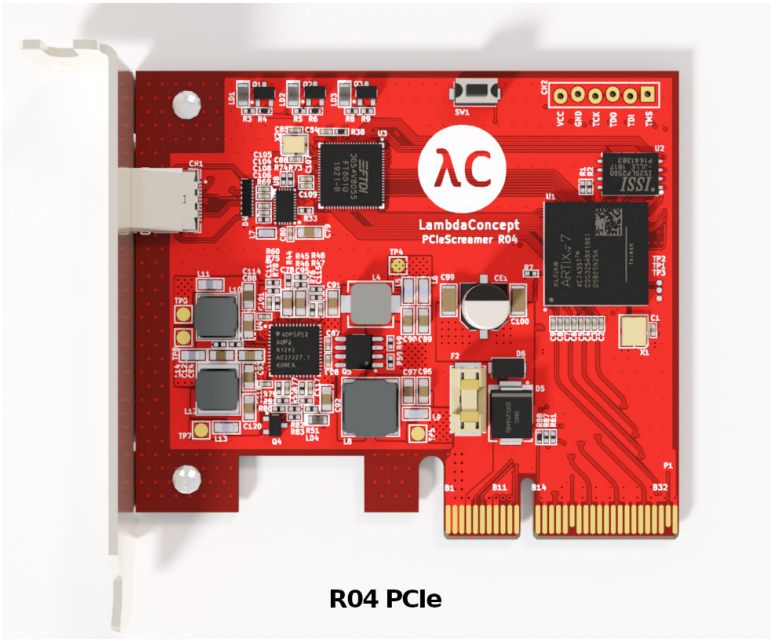
# CONTENTS

The Screamer is designed for DMA (Direct Memory Access) attacks over PCI Express.

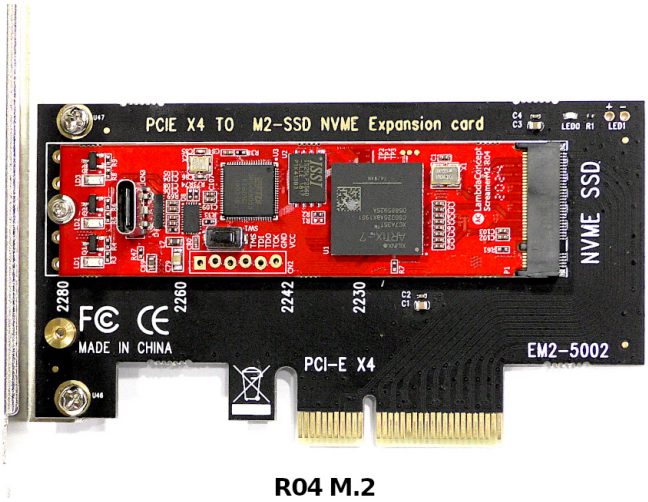
The PCIe bus is heavily used to interconnect chips in computers/embedded devices. Tools to interact with PCIe can be very expensive and often limited when doing security researchs. The Screamer aims to offer an alternative at a reasonable price.




**Squirrel PCIe**




**R04 PCIe**




**R04 M.2**

## HARDWARE REVISIONS

- Squirrel PCIe (2022-Active): Screamer PCIe Squirrel Edition with a Low-Profile form factor and PCIe x1 connectivity. Compatible with PCILeech gateware/software.
- R04 PCIe (2021-Discontinued): Screamer PCIe USB-C (R04) with a Low-Profile form factor and PCIe x4 connectivity. Compatible with Screamer M.2 gateware/software.
- R04 M.2 (2020-Discontinued): Screamer M.2 USB-C (R04) brings USB-C 3.1 connectivity to our successful Screamer M.2 board while keeping its M.2 form factor and PCIe x4 connectivity.
- R03 M.2 (2019-Discontinued): Screamer M.2 (R03) replaces PCIe Screamer R02 with an M.2 form factor and PCIe x4 connectivity.
- R02 PCIe (2018-Discontinued): PCIe Screamer with rerouted PCIe lanes for better signal stability.
- R01 PCIe (2017-Discontinued): PCIe Screamer first version in PCIe x1.

## BUY ON OUR WEBSHOP

Get yours at https://shop.lambdaconcept.com/

## PDF DOCUMENTATION

`screamer.pdf`

CHAPTER

ONE

# SPECIFICATIONS

- **FPGA Chipset:** Xilinx 7 Series XC7A35T
- **USB Chipset:** USB 3.0 FTDI FT601
- **Data Port:** USB Type-C 3.1 Gen 1
- **Update Port:** USB-C (USB 2 Integrated JTAG + Serial; Squirrel only)
- **PCIe Connector:** PCIe x1 (Squirrel); PCIe x4 (R04)
- **Bandwidth:** PCIe Gen 2.0: 5GT/s
- **M.2 Connector:** M.2 Key M (M.2 versions only)
- **System Clock:** Frequency: 100 MHz
- **User Interfaces:** 2 x LED (Green); 1 LED (Prog Done)
- **Input Voltage:** 3.3V (M.2 versions); 12V (PCIe versions)
- **GPIO (Testpoints):** 2 pins + GND
- **JTAG:** 6 pins JTAG connector for FPGA programming (Use the integrated Update Port on Squirrel)
- **Flash:** 256Mb SPI Flash for FPGA self-configuration
- **Dimensions:**
  - **Squirrel PCIe:** Low-Profile (68.90mm height x 67.00mm length)
  - **R04 PCIe:** Low-Profile (68.90mm height x 79.20mm length)
  - **R03/R04 M.2:** 2280 (22 mm x 80 mm)

CHAPTER

TWO

# APPLICATIONS

## 2.1 PCILeech (Recommended)

https://github.com/ufrisk/pcileech

PCILeech uses PCIe hardware devices to read and write target system memory.
This is achieved by using DMA over PCIe.
No drivers are needed on the target system.

| Screamer version | PCILeech bitstream |
|---|---|
| Squirrel | PCILeech Squirrel |
| R03/R04 M.2/PCIe | PCILeech ScreamerM2 |

**The Screamer board comes pre-flashed with the official PCILeech gateware:**

- **Squirrel:** Pre-flashed with PCILeech [v4.10,0100]
- **R04: (>Jan. 2021)** Pre-flashed with PCILeech [v4.7,0100]
- **R04: (>Oct. 2020)** Pre-flashed with PCILeech [v4.6,0100]
- **R04:** Pre-flashed with PCILeech [v4.3,0100]
- **R03:** Pre-flashed with PCILeech [v4.1,0100]

**Note:** Starting from the Squirrel version, our JtagSerial cable is no longer required to reprogram the Screamer with another gateware or update PCILeech to newer versions. Instead just connect through the integrated Update Port.

## 2.2 PCIeScreamer: (Example gateware)

https://github.com/enjoy-digital/pcie_screamer

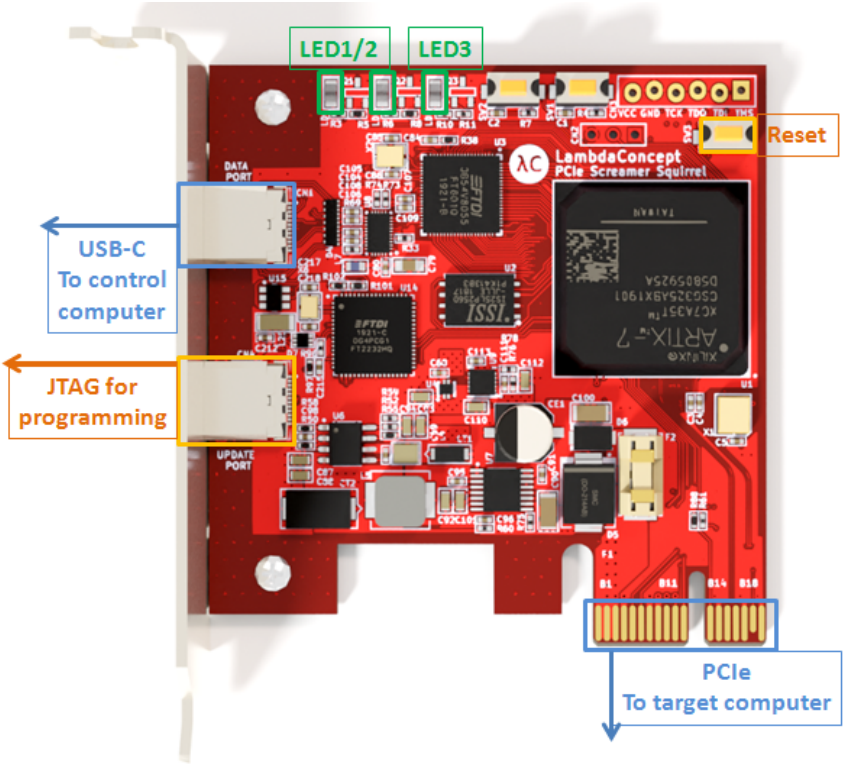
A example design built with Migen and LiteX that allows:
- Redirecting PCIe TLP requests to the Host, using the Host to analyze/generate the TLP completion and sending it to the PCIe bus.
- Generating PCIe TLP requests from the Host and redirecting the TLP completions to the Host.

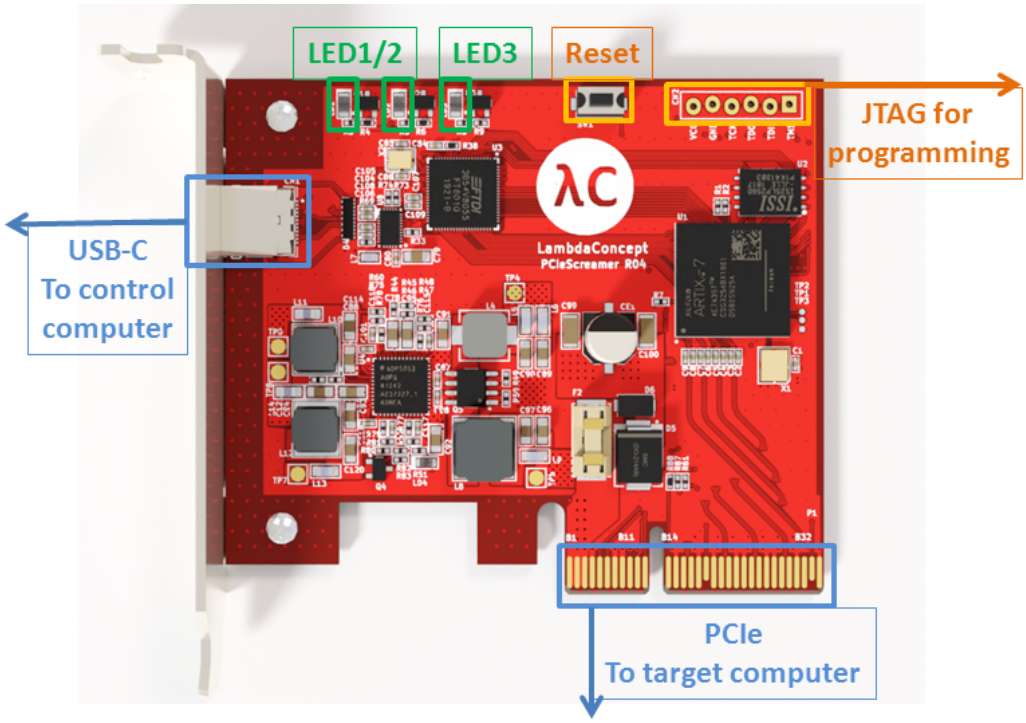
CHAPTER

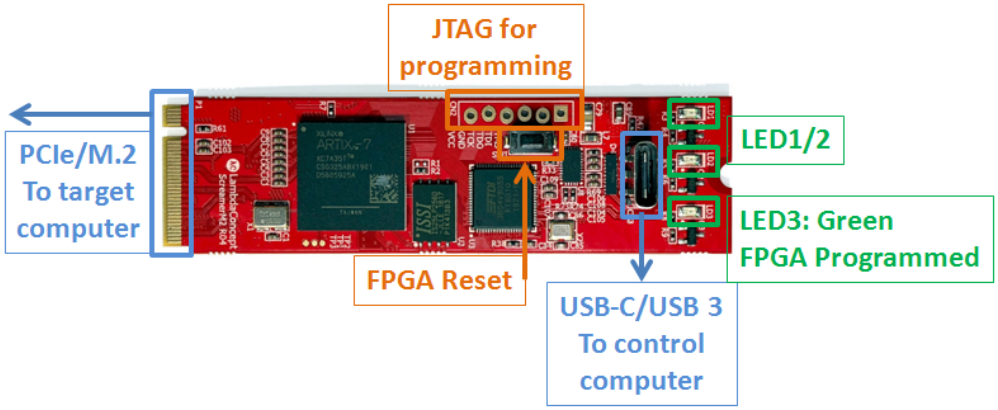
THREE

# GETTING STARTED

The Screamer board comes pre-flashed with the PCILeech FPGA gateware, ready for DMA attacks.






## 3.1 1. Disable IOMMU on the target system

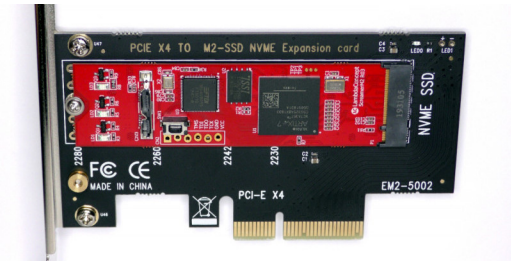
Check your motherboard BIOS settings on the target computer. (The one you want to read/write the memory).

-> IOMMU setting should be set to **disabled**.

## 3.2 2. Plug the Screamer

Power off your target computer and plug the Screamer.

-> Optionally for Screamer M.2 use the provided M.2/PCIe adapter cards to fit your target:




## 3.3 3. Boot the target system

Power up the target computer.

-> The Screamer is powered from the PCIe/M.2 connector, and will boot with the target computer.

---

**Note:** The JTAG Serial cable alone does not power the Screamer !

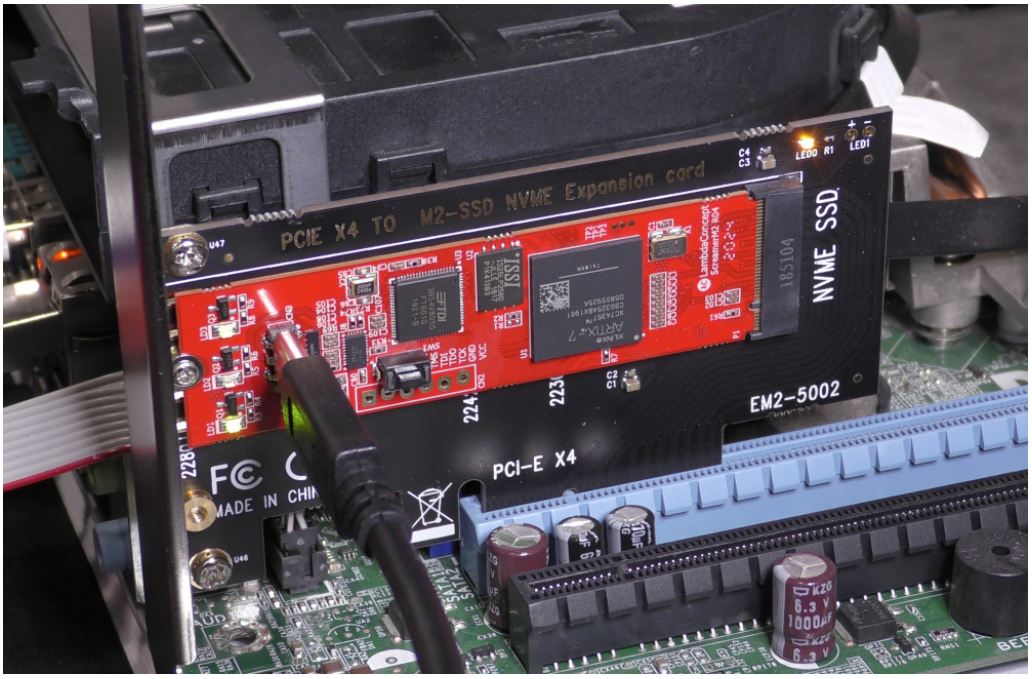
---

Once booted, the FPGA “Prog Done” LED LD3 will be green. If the LED LD3 is not turned on, check the power from the PCIe/M.2 slot, and make sure the FPGA is correctly programmed.

At this stage, the Screamer (PCIe side) should be visible from the device manager or lspci on the target system. (By default with PCILeech gateware, it is seen as Ethernet controller).

## 3.4 4. Connect with USB-C/USB 3

Plug the USB-C or USB 3 cable to the Screamer and to your control computer. Optionally use the provided USB-C to USB 3.1 Type-A Adapter if your control computer does not have any USB-C port.




Now the Screamer (USB3 side) should show up as USB FTDI device.

## 3.5 5. Run PCILeech

Install PCILeech on the control computer.

https://github.com/ufrisk/pcileech

Run PCILeech:

```
$ sudo ./pcileech probe -device fpga -v

[+] using FTDI device: 0403:601f (bus 2, device 5)
[+] FTDI - FTDI SuperSpeed-FIFO Bridge - serialNumber 000000000001
DEVICE: FPGA: PCIeScreamer M2 PCIe gen2 x1 [300,0,500] [v4.6,0100]
 Memory Map:
 START               END                 #PAGES
 0000000000000000 - 000000000009ffff  000000a0
 00000000000c0000 - 00000000caffffff  000caf40
 0000000100000000 - 000000012dffffff  0002e000

 Current Action: Probing Memory
 Access Mode:    Normal
 Progress:       4832 / 4832 (100%)
 Speed:          241 MB/s
 Address:        0x000000012E000000
 Pages read:     1019872 / 1236992 (82%)
 Pages failed:   217120 (17%)
Memory Probe: Completed.
```

CHAPTER

FOUR

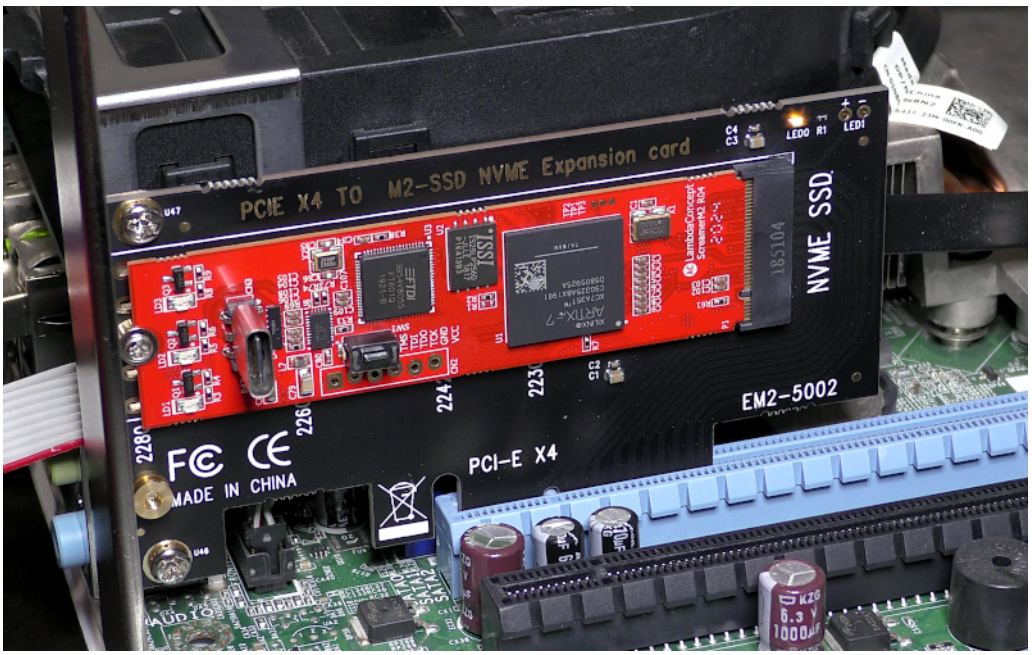
# UPGRADING

The FPGA on the Screamer runs a binary program, called gateware (or bitstream). This gateware is stored on the onboard SPI Flash memory chip, and loaded at power-on.

To change the gateware (for example to update PCILeech to a newer version, or replace it by your own custom gateware), the SPI Flash memory should be reprogrammed with the procedure below:

## 4.1 1. Power the Screamer

Power the Screamer from the PCIe/M.2 connector !




**Note:** The JTAG Serial cable alone does not power the Screamer !

The Screamer requires power from the PCIe/M.2 connector for all operations (programming or running PCILeech).

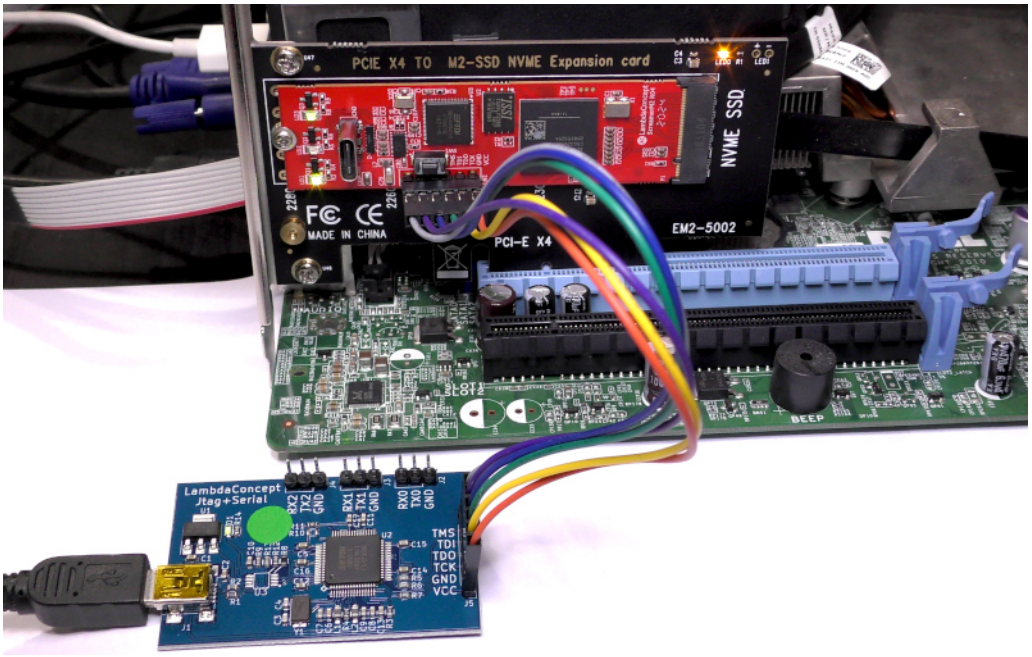
## 4.2 2. Connect the JTAG Cable

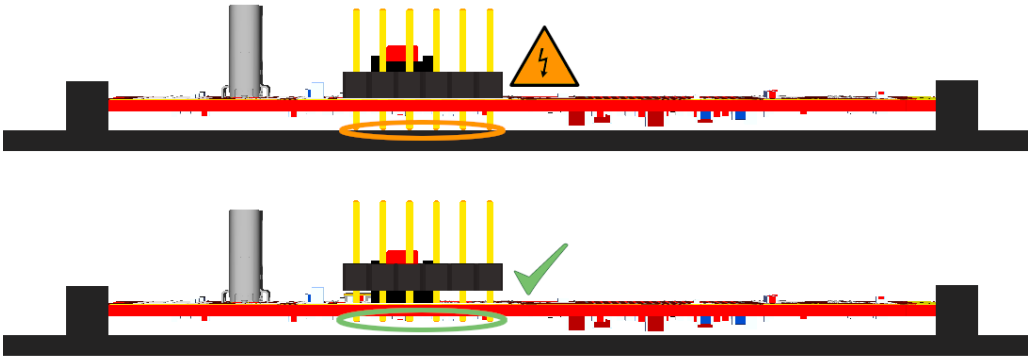
Connect the JTAG Cable to the Screamer, and to your computer.

### 4.2.1 Squirrel

**-> For the Squirrel version, the external JTAG cable is not required, instead directly connect through the Update Port.**

### 4.2.2 R03/R04



**Note:** Make sure to respect the pinout and not to short circuit the pins with the PCIe adaptor boards or your PC motherboard underneath !!

## 4.3 3. Get OpenOCD and flashing scripts

You need OpenOCD and a proxy bitstream. The proxy bitstream will be programmed temporarily on the FPGA and used by OpenOCD to program the SPI Flash with your final gateware.

### 4.3.1 Precompiled OpenOCD archive

- **Windows:** `openocd-win.zip`

  Unzip the content, it will create a "openocd" directory, with the executable file in "bin\openocd.exe"

- **Linux/Windows:** For compiling your own OpenOCD, refer to the *OpenOCD* section.

### 4.3.2 Get Proxy and flashing scripts

- **Scripts:** `flash_screamer.zip`

Unzip the content, it will create a "flash_screamer" directory with this content:

```
$ ls flash_screamer/

bscan_spi_xc7a35t.bit
flash_screamer_squirrel.cfg
flash_screamer_r03_r04.cfg
```

## 4.4 4. Get the new gateware

Download a pre-built gateware from the official PCILeech GitHub:

| Screamer version | PCILeech bitstream |
|---|---|
| Squirrel | PCILeech Squirrel |
| R03/R04 M.2/PCIe | PCILeech ScreamerM2 |

Put the gateware file inside the "flash_screamer" directory. Your "flash_screamer" directory will now look like this:

```
$ ls flash_screamer/

bscan_spi_xc7a35t.bit
flash_screamer_squirrel.cfg
flash_screamer_r03_r04.cfg
pcileech_screamer_m2_top.bin   [...or...]   pcileech_squirrel_top.bin
```

## 4.5 5. Run the flashing script

**Note:** For Windows, use Zadig as explained in "*WinUSB driver for Windows*" to associate the JTAG cable with the WinUSB driver.

- **Windows:**

  For Windows, specify the path to the precompiled openocd.exe, for example:

```
>cd flash_screamer

>..\openocd\bin\openocd.exe -f flash_screamer_squirrel.cfg
   [...or...]
>..\openocd\bin\openocd.exe -f flash_screamer_r03_r04.cfg
```

- **Linux:**

```
$ cd flash_screamer/

$ openocd -f flash_screamer_squirrel.cfg
   [...or...]
$ openocd -f flash_screamer_r03_r04.cfg
```

You will get the following output:

```
Open On-Chip Debugger 0.10.0+dev-01293-g7c88e76a-dirty (2020-07-02-19:28)
Licensed under GNU GPL v2
For bug reports, read
    http://openocd.org/doc/doxygen/bugs.html
Info : auto-selecting first available session transport "jtag". To override use
↪'transport select <transport>'.
Info : ftdi: if you experience problems at higher adapter clocks, try the command "ftdi_
↪tdo_sample_edge falling"
Info : clock speed 10000 kHz
Info : JTAG tap: xc7.tap tap/device found: 0x0362d093 (mfg: 0x049 (Xilinx), part: 0x362d,
↪ ver: 0x0)
Info : JTAG tap: xc7.tap tap/device found: 0x0362d093 (mfg: 0x049 (Xilinx), part: 0x362d,
↪ ver: 0x0)
Info : Found flash device 'issi is25lp256d' (ID 0x0019609d)
Warn : device needs paging or 4-byte addresses - not implemented
Info : Found flash device 'issi is25lp256d' (ID 0x0019609d)
Warn : device needs paging or 4-byte addresses - not implemented
Info : Found flash device 'issi is25lp256d' (ID 0x0019609d)
Warn : device needs paging or 4-byte addresses - not implemented
Info : Found flash device 'issi is25lp256d' (ID 0x0019609d)
Warn : device needs paging or 4-byte addresses - not implemented
Info : sector 0 took 113 ms
Info : sector 1 took 108 ms
Info : sector 2 took 113 ms
Info : sector 3 took 116 ms
Info : sector 4 took 125 ms
Info : sector 5 took 114 ms
```

(continues on next page)

(continued from previous page)

```
Info : sector 6 took 110 ms
Info : sector 7 took 98 ms
Info : sector 8 took 122 ms
Info : sector 9 took 118 ms
Info : sector 10 took 117 ms
Info : sector 11 took 114 ms
Info : sector 12 took 108 ms
Info : sector 13 took 102 ms
Info : sector 14 took 110 ms
Info : sector 15 took 92 ms
Info : sector 16 took 139 ms
Info : sector 17 took 105 ms
Info : sector 18 took 102 ms
Info : sector 19 took 104 ms
Info : sector 20 took 102 ms
Info : sector 21 took 104 ms
Info : sector 22 took 110 ms
Info : sector 23 took 112 ms
Info : sector 24 took 116 ms
Info : Found flash device 'issi is25lp256d' (ID 0x0019609d)
Warn : device needs paging or 4-byte addresses - not implemented
shutdown command invoked
```

Your board is flashed ! Remove the JTAG cable and reboot the target computer.

---

**Note:** “Warn : device needs paging or 4-byte addresses - not implemented” is NOT a problem.

“Warn : 4-byte addresses needed, might need extra command to enable” is NOT a problem.

---

CHAPTER

FIVE

# OPENOCD

## 5.1 Precompiled binaries

- **Windows:** `openocd-win.zip`

## 5.2 Compile on Linux

Get OpenOCD latest version:

```
git clone https://github.com/ntfreak/openocd.git
cd openocd

./bootstrap
./configure
```

Build:

```
make
sudo make install
```

## 5.3 Compile on Windows

We do not offer support for this.

## 5.4 Cross compilation for Windows

### 5.4.1 Build MXE toolchain

Full instructions here: https://mxe.cc/

Get the repo:

```
git clone https://github.com/mxe/mxe.git
```

Compile required files:

```
# Basic mingw32
make -j cc
# USB/FTDI libs for openocd
make -j libusb1 libftdi1
```

Install or set PATH:

```
export PATH=/path_to_my_directory/mxe/usr/bin:$PATH
```

You now have the required "i686-w64-mingw32.static" compiler.

### 5.4.2 Build OpenOCD

Get OpenOCD latest version:

```
git clone https://github.com/ntfreak/openocd.git
cd openocd

./bootstrap
./configure --prefix= --host=i686-w64-mingw32.static --enable-ftdi
```

Build:

```
make pkgdatadir=
```

Create a Windows package layout:

```
make pkgdatadir= DESTDIR=/tmp/openocd install
cd /tmp

ls -l openocd/
total 0
drwxr-x---  2 po po  60 Jan 16 13:57 bin
drwxr-x---  2 po po  60 Jan 16 13:57 OpenULINK
drwxr-x--- 11 po po 300 Jan 16 13:57 scripts
drwxr-x---  4 po po  80 Jan 16 13:57 share

zip -r openocd-win.zip openocd/
```

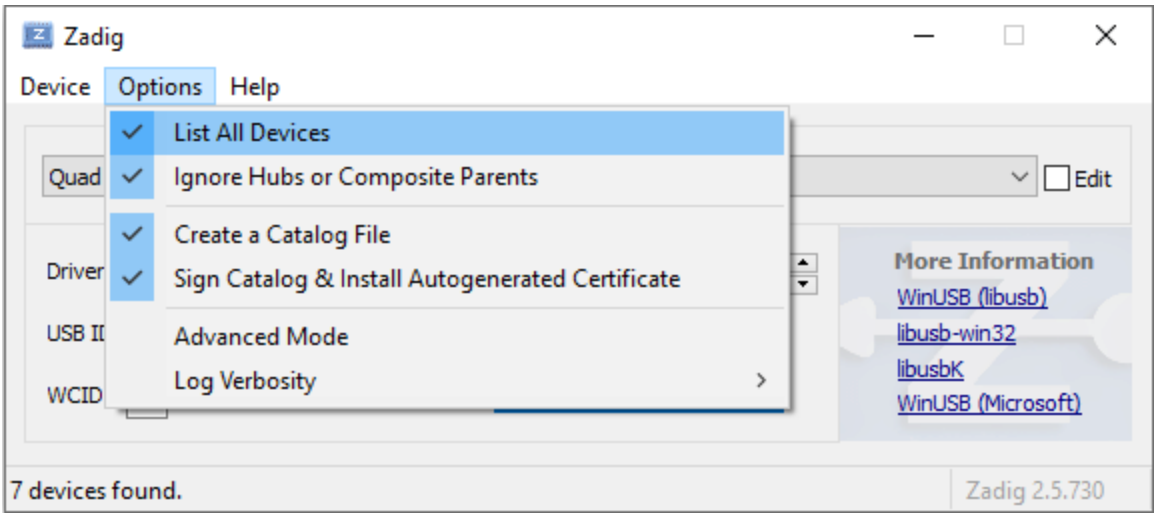
openocd-win.zip

## 5.5 WinUSB driver for Windows

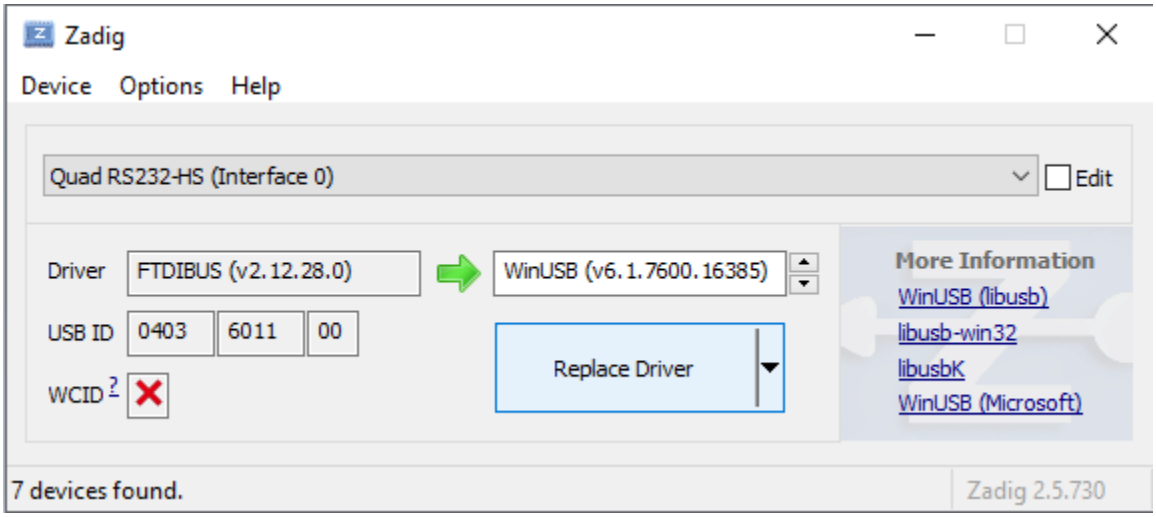
Before using OpenOCD with the JTAG Cable, you need to replace the default FTDI driver with WinUSB.

To do that, download Zadig and run it.

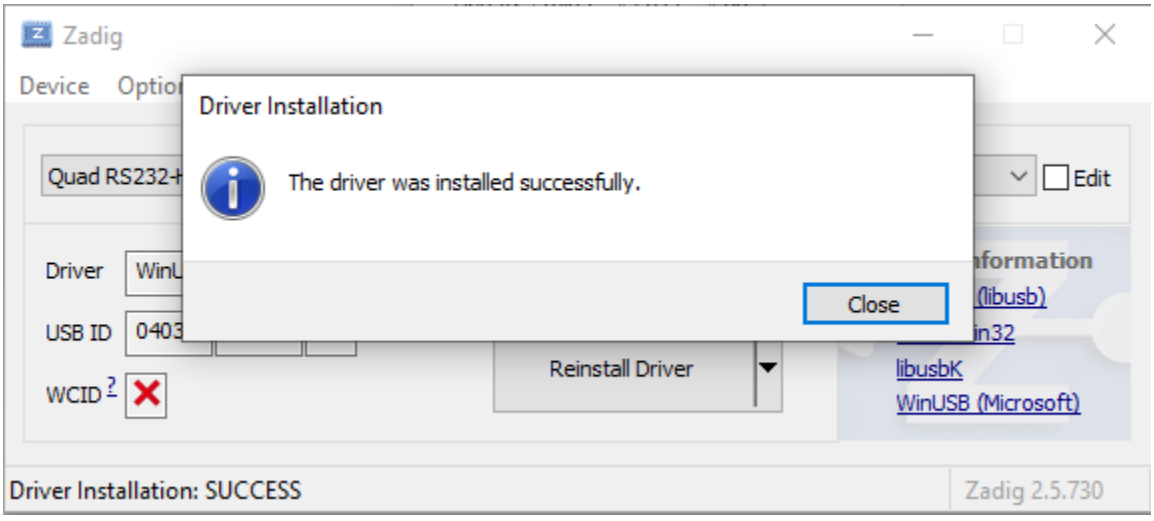
1. Connect the JTAG Cable to your computer.
2. List all devices in Zadig.




3. Assign *"WinUSB"* driver to the device *"Quad RS232-HS (Interface 0)"*.




Done !



CHAPTER

SIX

# TROUBLESHOOTING

## 6.1 Error: JTAG scan chain interrogation failed

```
$ sudo openocd -f flash_screamer.cfg

Open On-Chip Debugger 0.10.0+dev-01293-g7c88e76a-dirty (2020-06-30-17:21)
Licensed under GNU GPL v2
For bug reports, read
http://openocd.org/doc/doxygen/bugs.html
Info : auto-selecting first available session transport "jtag". To override use
↪'transport select <transport>'.
Info : ftdi: if you experience problems at higher adapter clocks, try the command "ftdi_
↪tdo_sample_edge falling"
Info : clock speed 10000 kHz

Error: JTAG scan chain interrogation failed: all ones
Error: Check JTAG interface, timings, target power, etc.

Error: Trying to use configured scan chain anyway...
Error: xc7.tap: IR capture error; saw 0x3f not 0x01
Warn : Bypassing JTAG setup events due to errors
Error: JTAG scan chain interrogation failed: all ones
Error: Check JTAG interface, timings, target power, etc.
Error: Trying to use configured scan chain anyway...
Error: xc7.tap: IR capture error; saw 0x3f not 0x01
Warn : Bypassing JTAG setup events due to errors
Error: Unknown flash device (ID 0x00ffffff)
```

**Note:** Your Screamer board is not powered properly !

**Note:** The JTAG Serial cable alone does not power the Screamer !

The Screamer requires power from the PCIe/M.2 connector for all operations (programming or running PCILeech).

## 6.2 Error: Unknown flash device

```
# openocd -f flash_screamer.cfg

Open On-Chip Debugger 0.10.0
Licensed under GNU GPL v2
For bug reports, read
http://openocd.org/doc/doxygen/bugs.html
none separate
adapter speed: 10000 kHz
Info : ftdi: if you experience problems at higher adapter clocks, try the command "ftdi_
↪tdo_sample_edge falling"
Info : clock speed 10000 kHz
Info : JTAG tap: xc7.tap tap/device found: 0x0362d093 (mfg: 0x049 (Xilinx), part: 0x362d,
↪ ver: 0x0)
loaded file bscan_spi_xc7a35t.bit to pld device 0 in 0s 218682us
Info : JTAG tap: xc7.tap tap/device found: 0x0362d093 (mfg: 0x049 (Xilinx), part: 0x362d,
↪ ver: 0x0)

Error: Unknown flash device (ID 0x00ffffff)
```

**Note:** Ensure you are using the latest version of *OpenOCD* from the git.

Or use the precompiled binaries from the *OpenOCD* section.

## 6.3 Error: contents differ

```
# openocd -f flash_screamer.cfg

[...]
sector 32 took 1 ms
Info : sector 33 took 0 ms
wrote 2228224 bytes from file top.bin in 15.640747s (139.124 KiB/s)
Info : Found flash device 'micron n25q256 3v' (ID 0x0019ba20)
Warn : device needs paging or 4-byte addresses - not implemented
read 2192012 bytes from file top.bin and flash bank 0 at offset 0x00000000 in 1.949110s␣
↪(1098.264 KiB/s)

contents differ

diff 0 address 0x000007a3. Was 0x00 instead of 0x40
diff 1 address 0x0000081a. Was 0x01 instead of 0x15
diff 2 address 0x0000081b. Was 0x80 instead of 0xc6
[...]
```

**Note:** Ensure you are using the latest version of *OpenOCD* from the git.

Or use the precompiled binaries from the *OpenOCD* section.

If you are using an older OpenOCD version (before Nov 12, 2021), you need the `flashid.patch` workaround to revert to 3-byte addresses commands.

**Important note:** After applying the patch, these warnings are fine as long as you dont get "contents differ" errors !

```
Warn : device needs paging or 4-byte addresses - not implemented
```

## 6.4 Error: libusb_open() failed with LIBUSB_ERROR_NOT_FOUND

For Windows only.

```
C:flash_screamer>..\openocd\bin\openocd.exe -f flash_screamer.cfg
Open On-Chip Debugger 0.10.0+dev-01293-g7c88e76a-dirty (2020-07-02-19:28)
Licensed under GNU GPL v2
For bug reports, read
        http://openocd.org/doc/doxygen/bugs.html
DEPRECATED! use 'adapter driver' not 'interface'
Info : auto-selecting first available session transport "jtag". To override use
↪'transport select <transport>'.
DEPRECATED! use 'adapter speed' not 'adapter_khz'
Error: libusb_open() failed with LIBUSB_ERROR_NOT_FOUND
Error: no device found
Error: unable to open ftdi device with vid 0403, pid 6011, description '*', serial '*'␣
↪at bus location '*'
```

**Note:** Use Zadig as explained in "*WinUSB driver for Windows*" to associate the JTAG cable with the WinUSB driver.

## 6.5 Contact support

If your issue is not listed here, please report to contact@lambdaconcept.com

CHAPTER

SEVEN

# OLDER VERSIONS

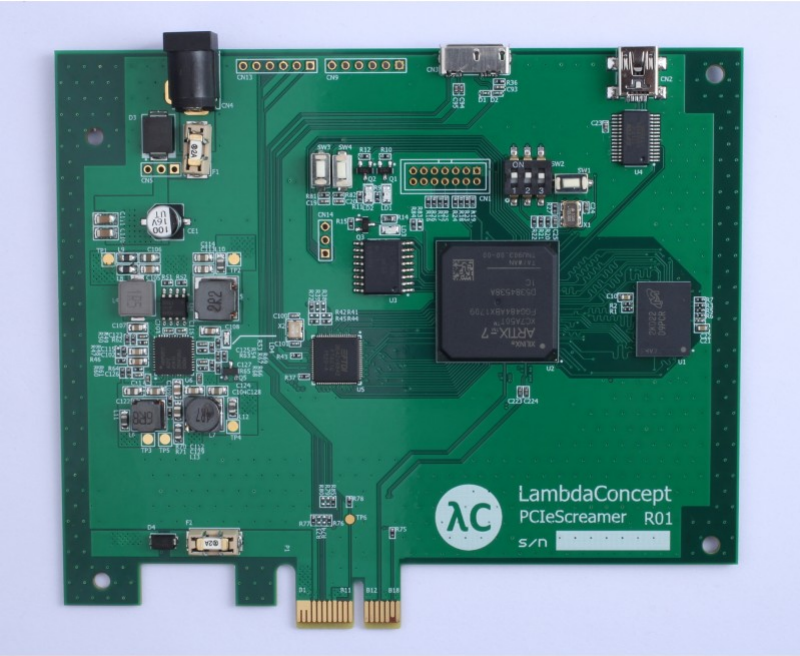
## 7.1 PCIe Screamer R02 (Discontinued)


PCIe Screamer R02 is a revised version of our successful PCIe Screamer.

Most notably PCIe lanes routing has been improved with better differential pairs impedance and length matching resulting in a better PCIe signal stability.

Existing software and gateware are fully compatible with this new PCIe Screamer R02 version.

## 7.2 PCIe Screamer R01 (Discontinued)




PCIe Screamer first version in PCIe x1.

## 7.3 Specifications (R01 & R02)

- **FPGA Chipset:** Xilinx 7 Series XC7A35T
- **Memory:**
  - 4Gb DDR3 DRAM MT41K256
  - 256Mb SPI Flash for FPGA self-configuration
- **USB Chipset:** USB 3.0 FTDI FT601
- **PCIe Bandwidth:** PCIe x1 Gen 2.0: 5GT/s
- **System Clock:** Frequency: 100 MHz
- **User Interfaces:**
  - 2 x LED
  - 2 x push button
- **Serial Interface:** USB to UART FTDI FT232
- **JTAG:** 6 pins JTAG connector for FPGA programming
- **Input Voltage:** 12V from PCIe or from external supply

- **GPIO:** 2 pins + GND
- **Dimensions:** 130 mm x 110 mm x 12 mm

## 7.4 Buttons

- **SW1:** FPGA Program/Reset
- **SW3:** User Button
- **SW4:** User Button

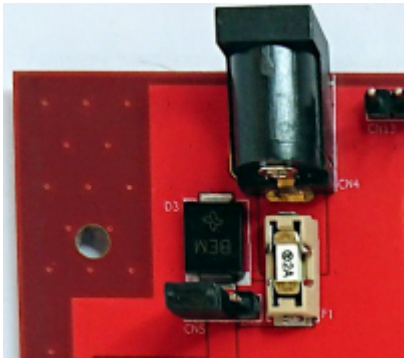
## 7.5 LEDs

- **LD1:** User Led (Green)
- **LD2:** User Led (Green)
- **LD3:** FPGA Prog Done (Green)
- **LD4:** Power Good (Red)

## 7.6 Quick Start

### 7.6.1 1. Select power source

- For PCIe power supply, put the jumper on the left position.
- For External supply via jack CN4, put the jumper on the right position. **Input voltage: 5V to 15V max.**

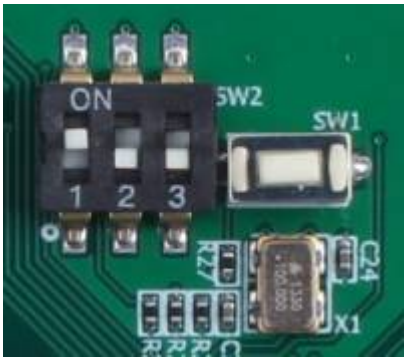



When the power supply is stable, the LED LD4 will be red.

### 7.6.2 2. Select Boot mode (R01 only)

SW2 selects the FPGA bootmode. Set it to Master SPI:

- 1: ON
- 2: OFF
- 3: OFF

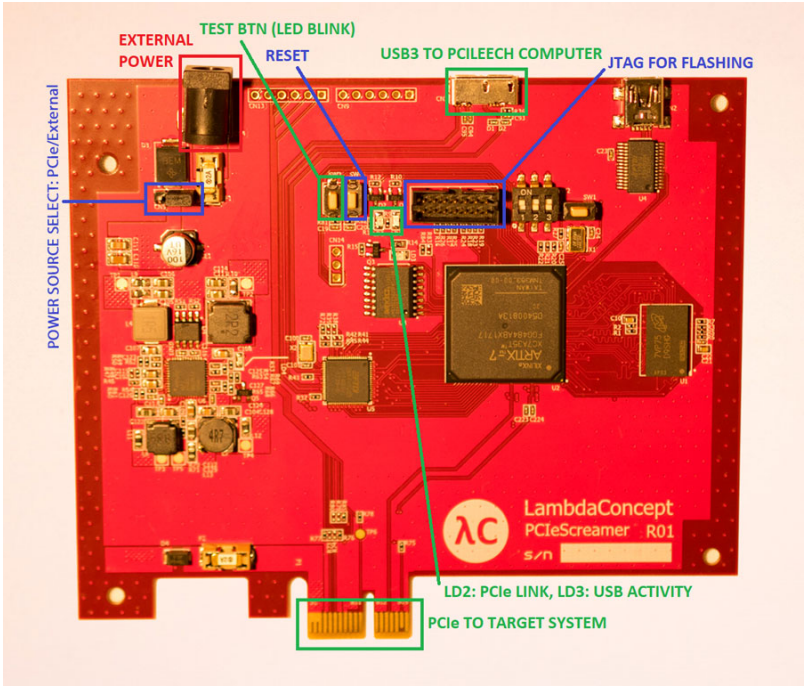



Once booted, FPGA prog done LED LD3 will be green.

## 7.7 Gateware & Software

**PCILeech recommended.**

Refer to the official PCILeech README and gateware for PCIeScreamer R01/R02



## 7.8 Flash Programming

Same as Screamer M.2. Refer to *Upgrading*.