

### Clock generator & Reset fanout

File: clock\_gen.kicad\_sch

PCIe switch

File: pcie\_switch.kicad\_sch

RC 4-lanes PCIe

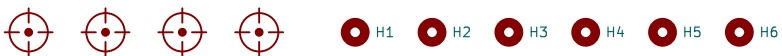
File: RC\_4-lanes-PCle.kicad\_sch

4 x 1-lanes PCIe

File: 4\_1-lanes\_PClc.kicad\_sch

## Direct PCIe

File: direct\_PClc.kicad\_sch



Chili.CHIPS

Sheet: /

File: openpci2-backplane.kicad\_sch

**Title:**

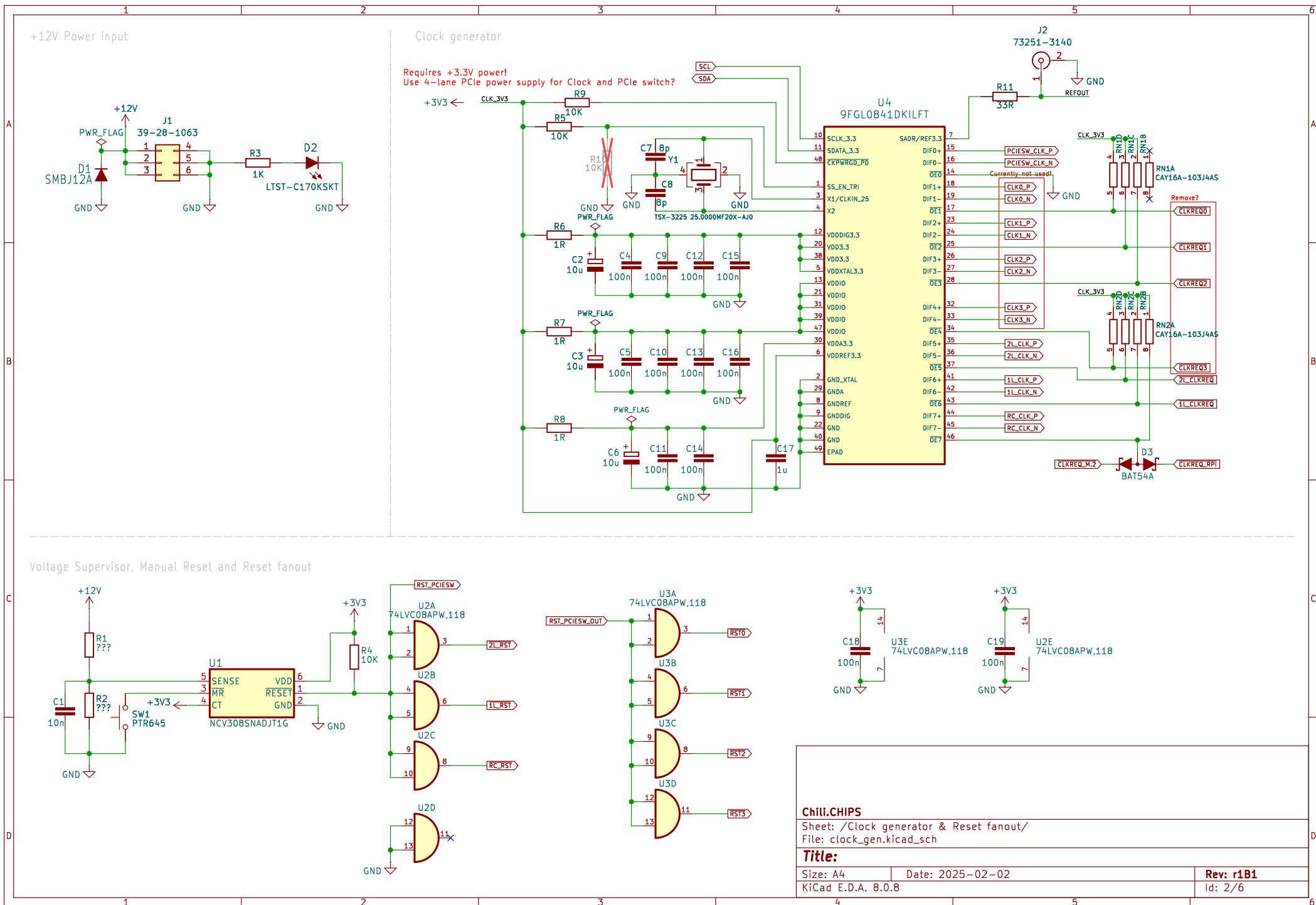
Size: A4

Date: 2025-02-02

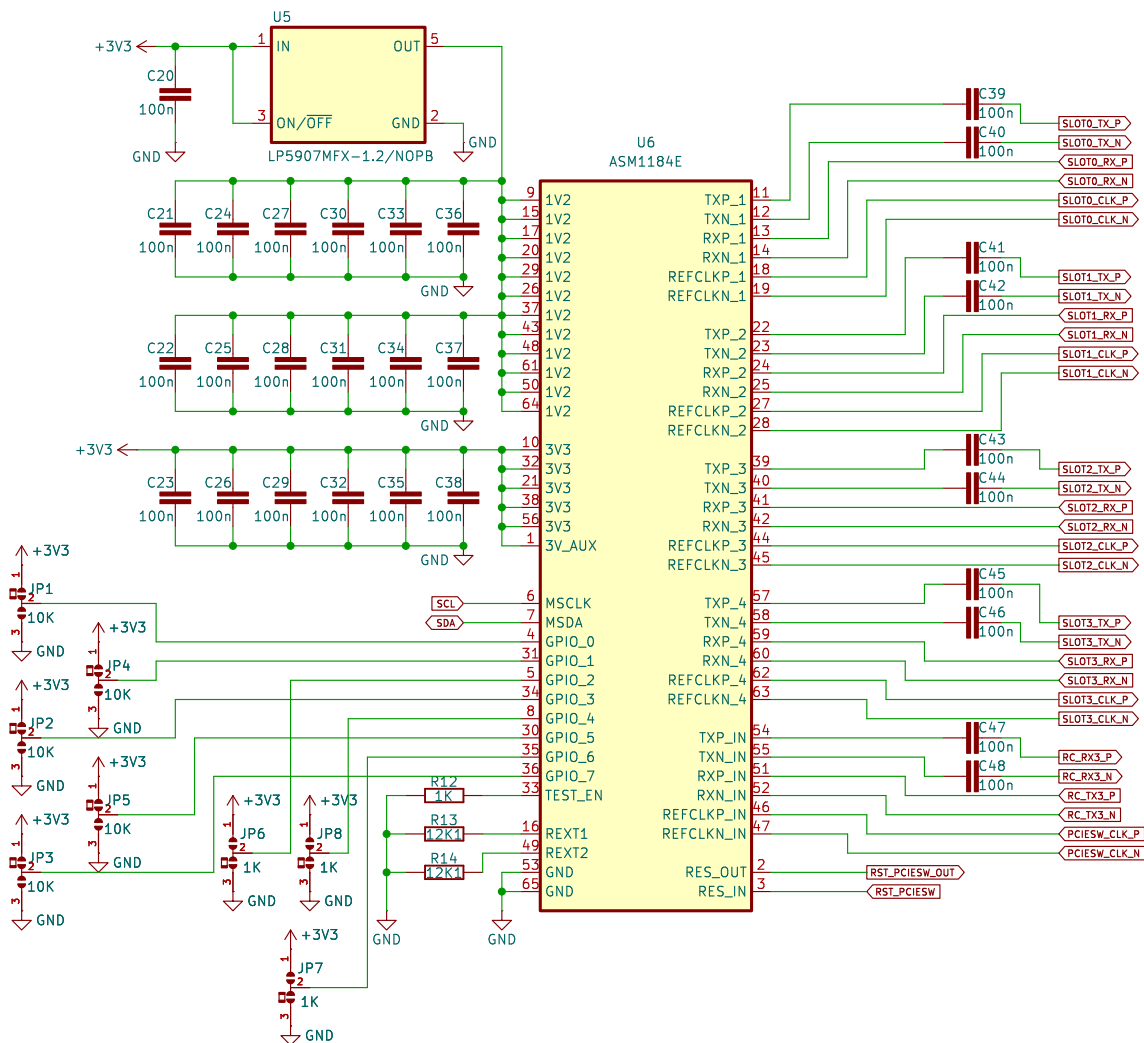
Rev: r1B1

KiCad E.D.A. 8.0.8

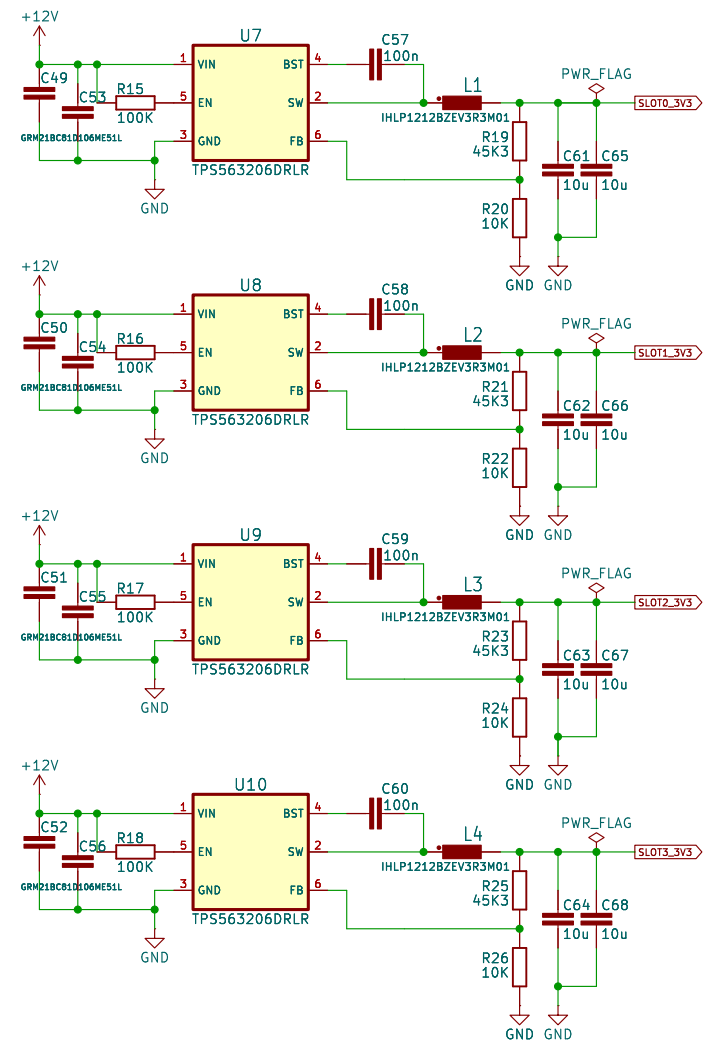
Id: 1/6



# PCIe 4-slot switch



# Point-of-load stepdown converters for 1-lanes PCIe



## Chili.Chips

Sheet: /PCIe switch/  
File: pcie\_switch.kicad\_sch

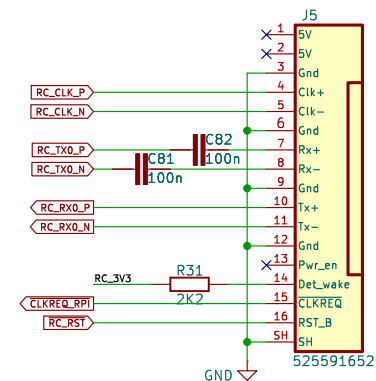
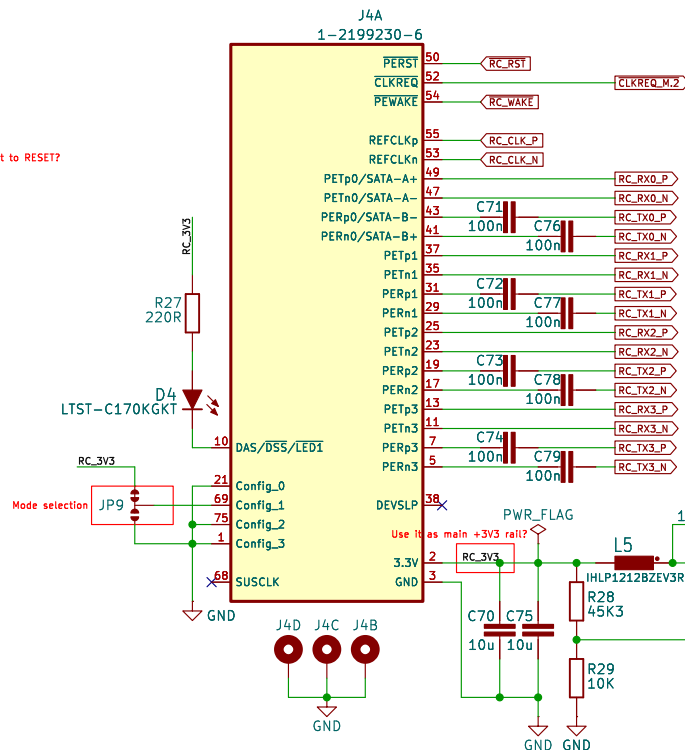
## Title:

Size: A4 Date: 2025-02-02  
KiCad E.D.A. 8.0.8

Rev: r1B1  
Id: 3/6

## 4-lane M.2 (Type M) for Root Complex

## Raspberry Pi FPC 16-pin connector



## 2.1. PCIe Signals

The PCIe signals are a single lane of PCIe Gen 2, including  $\overline{\text{CLKREQ}}$  and RST\_B sideband signals which operate at 3.3V.

### 2.1.1. Pwr\_en pin

This pin is a 3.3V output from the Raspberry PI to a HAT+ or other add-on board, and signals to the HAT+ to power up any supplies. For example, in the instance of the Raspberry PI M.2 M Key HAT+, this enables the M.2 3.3V power (which is generated from the incoming 5V). Provide a 100K low pull on this pin on any HAT+.

### 2.1.2. Det\_wake pin

This pin is a 3.3V input to the Raspberry PI. Pull high to 3.3V either from a resistive divider from 5V (3k6/6k8 giving 2.35k output impedance), or from permanently enabled 3.3V (using a 2.2K resistor). The Raspberry PI will detect this high pull at boot time, and will automatically probe the PCIe bus. Use the PCIe WAKE# to pull this low

Lines #	Device
0	2-lines slot + M.2, Raspberry Pi PFC
1	2-lines slot + M.2
2	1-lines slot + M.2
3	PCIe switch for 4 x 1-lines + 4 x M.2

Chili.CHIPS

Sheet: /RC 4-lanes PCIe/  
File: RC\_4-lanes-PCIe.kicad\_sch

**Title:**

Size: A4	Date: 2025-02-02
----------	------------------

KiCad E.D.A. 8.0.8

Rev: r1B1

Id: 4/6

