Chip Designer: Refresh and Backend Discussion

01. Design | 02. Review | 03. Prototype

Untitled Chip

Review

IP Library

Go to Category ▾

Filter

Mastering

AI ACCELERATOR

NVIDIA
NVIDIA NVDLA (Deep Learning Accelerator)
NVIDIA Deep Learning Accelerator
Open Source

GRAPHICS PROCESSOR

ThinkSilicon
NEMA®|pico GPU
2D/2.5D Graphics Accelerator
DS

HIGH SPEED INTERFACE

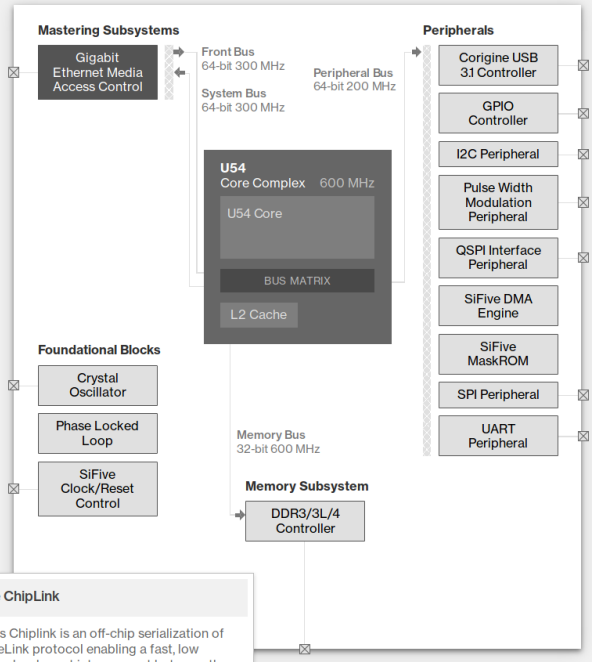
mobiveil
Mobiveil GPEX PCI-Express 4.0 Controller
PCI Express Gen4 Controller
DS

PCI Express Gen3 Controller
PCI Express Gen3 Controller

MISCELLANEOUS IP

SiFive
SiFive ChipLink
A coherent chip-to-chip interconnect

NETWORKING



SiFive ChipLink

SiFive's Chiplink is an off-chip serialization of the TileLink protocol enabling a fast, low overhead, coherent interconnect between the SoC and an external device.

Max: **1**
Vendor: **SiFive, Inc.**

SiFive

0 Added

+ Add

Chip Details

Platform: Freedom Unleashed
Process: TSMC 28nm
Base Design: Application Processor

Chip Settings

CPU Clock: **600 MHz**

100MHz | 500 | 900MHz

Peripheral Bus: **200 MHz**

50MHz | 200 | 350 | 500MHz

Mastering Bus Clock Ratio

2:1	3:1	4:1

Memory Port Width (bits)

32	64	128

CPU Clock speed and Bus Clock Ratio determine Front and System Bus speeds.

Bus widths are determined by the choice of Core IP.

Memory Bus speed is determined by IP in the Memory Subsystem.

```
ipSpecs = [
SLACLDO50mA

signableInputs": "",
signableOutputs": "",
asHas": "",
asReq": "1uAP, VREF",
ock": "SLACLDO50MA",
l": "LDO_50MA_TOP",
mments": "",
signShare": "N",
agramRegion": "Always-on Blocks",
splayCategory": "Power Management",
splayDescription": "50mA Configurable LDO",
splayName": "Low Dropout Regulator, 50mA",
hasis": "",
ternalPadphyTypes": "",
rdSoft": "Hard",
```

```
ge sifive.enterprise.ip.slac.ldo // Package for SLAC ldo

t chisel3.{Input, Output}
t Chisel._



ariables describe the LDO configuration
class SLACLDOConfig(
bleInit: Boolean = true,

mWidth:  Int = 4,
mInit:   Int = 4,

3Init:   Boolean = false,
5Init:   Boolean = false,
3Init:   Boolean = true,
5Init:   Boolean = true,
```

puts: *On-boarding IP*

- What format for IP?
- Common description between front end and back end?

tputs: *Exporting user design*

- What format for chip specs?

nstraints: *User experience vs. Fabrication/synthesis*

- How free is the user's design space?
- Where and how to encode constraints?

s sizes

s speeds, timing, clock ratios

ock generation

d map and pin muxing

ocess feature conflicts between IP (DRRC issues)

wer estimation

ea estimation

erfacing to blocks and pads

- Logic levels, PHYs, Muxing

sing

- Automated or user choice? What trade offs do we want to present to the user?

ensing

- Vendor restrictions and availability