Introduction to the SkyWater PDK
The New Age of Open Source Silicon

Tim Edwards
SVP Analog & Platform

efabless
efabless.com

Open Circuit Design
opencircuitdesign.com

October 8, 2021
Introduction to the SkyWater PDK
Introduction to the SkyWater PDK
Introduction to the SkyWater PDK
Introduction to the SkyWater PDK

130nm SCMOS
Introduction to the SkyWater PDK

https://github.com/google/skywater-pdk

130nm SCMOS
Introduction to the SkyWater PDK

efabless.com
Introduction to the SkyWater PDK

The "Caravel" harness chip
Introduction to the SkyWater PDK

The "Caravel" harness chip

RISC-V processor
Introduction to the SkyWater PDK

The "Caravel" harness chip

RISC-V processor

user project area
Introduction to the SkyWater PDK

Your project here!
Introduction to the SkyWater PDK

Projects on Google/SkyWater MPW-One

https://efabless.com/open_mpw_shuttle_project_mpw_one
Introduction to the SkyWater PDK

Your project here!
The SkyWater Open PDK

PDK = "Process Design Kit"
The SkyWater Open PDK

SKY130
The SkyWater Open PDK

SKY130

130 nm
The SkyWater Open PDK

SKY130

130 nm

minimum-length transistor*

transistor length = 130 nm

"feature size"

*caveat: for obscure reasons, the minimum size device in the SKY130 process is actually 150 nm. . .
The SkyWater Open PDK

Minnesota

Bloomington

Minnesoeta
The SkyWater Open PDK

Public repository

Documentation

PDK Library and files
The SkyWater Open PDK

Public repository

- Documentation
- PDK Library and files
- Community
The SkyWater Open PDK

Public repository

- Documentation
  https://skywater-pdk--136.org.readthedocs.build

- PDK Library and files

- Community
The SkyWater Open PDK

Public repository

Documentation
https://skywater-pdk--136.org.readthedocs.build

PDK Library and files
https://github.com/google/skywater-pdk

Community
The SkyWater Open PDK

Public repository

Documentation
https://skywater-pdk--136.org.readthedocs.build

PDK Library and files
https://github.com/google/skywater-pdk

Community

slack
https://join.skywater.tools
Open-Source EDA Tools
Open-Source EDA Tools

open_pdk

http://opencircuitdesign.com/open_pdk
https://github.com/RTimothyEdwards/open_pdk
Open-Source EDA Tools

open_pdfs

http://opencircuitdesign.com/open_pdfs
https://github.com/RTimothyEdwards/open_pdfs

Open-source PDK repository

open_pdfs

Makefile

magic
openlane
ngspice
xschem

...
Open-Source EDA Tools

**open_pdks**

Steps to installing the SKY130 PDK

1. Clone the repository
   
   
   "git clone https://github.com/RTimothyEdwards/open_pdks"

2. Run "cd open_pdks"

3. Run "configure --enable-sky130-pdk"

4. Run "make"

5. Run "sudo make install"
Open-Source EDA Tools
Tools Currently Supported by Open_PDKs

Magic

http://opencircuitdesign.com/magic
Open-Source EDA Tools
Tools Currently Supported by Open_PDKs

Klayout  https://www.klayout.de
Open-Source EDA Tools

Tools Currently Supported by Open_PDKs

Openlane

https://github.com/The-OpenROAD-Project/OpenLane
Open-Source EDA Tools
Tools Currently Supported by Open_PDKs

Xschem
https://github.com/StefanSchippers/xschem
Open-Source EDA Tools

Tools Currently Supported by Open_PDKs

Netgen

http://opencircuitdesign.com/netgen
Open-Source EDA Tools
Tools Currently Supported by Open_PDKs

Ngspice

https://ngspice.sourceforge.net
Open-Source EDA Tools

Tools Currently Supported by Open_PDKs

iverilog
https://iverilog.icarus.com

qflow
http://opencircuitdesign.com/qflow

IRSIM
http://opencircuitdesign.com/irsim

xcircuit
http://opencircuitdesign.com/xcircuit

xyce
https://xyce.sandia.gov

*fun fact: These slides were drawn with xcircuit!*
Open-Source EDA Tools

SkyWater SKY130 Libraries

1. Digital standard cells
   - sky130_fd_sc_sc_hd
   - sky130_fd_sc_sc_hdl
   - sky130_fd_sc_sc_hs
   - sky130_fd_sc_sc_ms
   - sky130_fd_sc_sc_ls
   - sky130_fd_sc_sc_lp
   - sky130_fd_sc_sc_hvl

2. Primitive devices / analog
   - sky130_fd_pr

3. I/O cells
   - sky130_fd Io

4. 3rd-party libraries
   - sky130_ml_xx_hd
   - sky130_sram_macros
Open-Source EDA Tools
PDK (e.g., SKY130) Installed Filesystem Structure

/usr/share/pdk/

sky130A/
Open-Source EDA Tools
SkyWater SKY130 Installed Filesystem Structure

```
/usr/share/pdk/
  sky130A/
  libs.tech/
  libs.ref/
```
Open-Source EDA Tools
SkyWater SKY130 Installed Filesystem Structure

/usr/share/pdk/

sky130A/

libs.tech/
libs.ref/

magic/
openlane/
xschem/
Open-Source EDA Tools
SkyWater SKY130 Installed Filesystem Structure

`/usr/share/pdk/`

```
sky130A/

libs.tech/
  magic/
  openlane/
  xschem/

libs.ref/
  sky130_fd_sc_hd/
  sky130_fd_pr/
  sky130_fd_io/
```
Open-Source EDA Tools
SkyWater SKY130 Installed Filesystem Structure

```
/usr/share/pdk/

sky130A/

libs.tech/  libs.ref/

magic/
openlane/
x schem/
ngspice/
sky130_fd_sc_hd/
sky130_fd_io/
sky130_fd_pr/

spice/
mag/
lib/
lef/
gds/
verilog/
```
Open-Source EDA Tools
Open PDKs Project Filesystem Structure

```
project_root/
```

e.g., "my_sky130_project"
Open-Source EDA Tools
Open PDKs Project Filesystem Structure

```
project_root/
```

e.g., "my_sky130_project"

```
xschem/
spice/
mag/
openlane/
verilog/
```
Open-Source EDA Tools

Open PDKs Project Filesystem Structure

```
/usr/share/pdk/

sky130A/

libs.tech/

magic/

sky130A.magicrc

xschem/

xschemrc

/project_root/ e.g., "my_sky130_project"

mag/

.magicrc

xschem/

xschemrc
```
Open-Source EDA Tools

Open PDKs Project Filesystem Structure

Project Management

/usr/share/pdk/scripts/project_manager.py

(work in progress)
Understanding the SkyWater PDK Layers

SKY130
Understanding the SkyWater PDK Layers

5 layers of aluminum metal
Titanium Nitride (TiN) aka "Local interconnect"
Understanding the SkyWater PDK

sky130_fd_sc_hd__nand2_2

local interconnect
- (blue)

local interconnect
- (blue)

strapped with metal1
- (purple)
Understanding the SkyWater PDK

Layers

poly contact

nitride poly cut layer (NPC)
Understanding the SkyWater PDK

**back-end metal stack**

- metal5: 1.26 µm
- metal4: 0.845 µm
- metal3: 0.845 µm
- metal2: 0.36 µm
- metal1: 0.36 µm
- mcon: 0.1 µm

local interconnect
Understanding the SkyWater PDK

front-end layers

- licon
- p-diffusion
- n-tap
- nwell
- pwell
- local interconnect
- p-tap
- n-diffusion
- polysilicon
Understanding the SkyWater PDK

front-end layers

- local interconnect
- polysilicon gate
- n-tap
- nwell
- p-tap
- pwell
- n-diffusion
- deep nwell
Understanding the SkyWater PDK

- high voltage layer (HVI)
- GPIO pad
- 3.3V busses
- 3.3V on pad
- 3.3V busses
- 1.8V busses
- 1.8V core signals
Understanding the SkyWater PDK

MiM cap layers

Top capacitor

Bottom capacitor
Understanding the SkyWater PDK

Redistribution layer

RDL (copper)

viaPI1

metal5

via4

metal4

4.0 µm

5.25 µm

1.26 µm

0.845 µm
Understanding the SkyWater PDK

Redistribution layer (RDL)

under-bump material (UBM) via PI2

solder bump
Understanding the SkyWater PDK

Devices
Understanding the SkyWater PDK

Devices

bipolar NPN

- base
- pwell in deep nwell
- collector
- n well
- emitter
- n diffusion
Understanding the SkyWater PDK

Devices

bipolar PNP

emitter

p diffusion

base

n well

collector

p substrate
Understanding the SkyWater PDK

Devices

polysilicon resistors

resistor length
Understanding the SkyWater PDK

Devices

diffusion resistors

n diffusion resistor on substrate

p diffusion resistor in nwell
Understanding the SkyWater PDK

Devices

pwell resistor

26 µm

p well in nwell
deep n well
n well
Understanding the SkyWater PDK

Devices

Discrete widths

- width = 0.35 \( \mu \text{m} \)
- width = 0.69 \( \mu \text{m} \)
- width = 1.41 \( \mu \text{m} \)
- width = 2.85 \( \mu \text{m} \)
- width = 5.73 \( \mu \text{m} \)
Understanding the SkyWater PDK

Devices

Reference layouts

RF 0.18V pFET
4 fingers
width = 3.0 µm
length = 0.15 µm
Understanding the SkyWater PDK

Devices

Hidden mask layers

![Diagram showing PSDM mask and POLYRES ID layer]
Understanding the SkyWater PDK

Libraries
Understanding the SkyWater PDK

Libraries

Digital standard cells

I/O cells

Primitive devices and models
Understanding the SkyWater PDK Libraries

Digital standard cells
Understanding the SkyWater PDK

Libraries

Digital standard cells

Naming convention:

```
sky130_vendor_library−type[name]  
```
Understanding the SkyWater PDK

Libraries

Digital standard cells

Naming convention:

\texttt{sky130\_vendor\_library\_type[\_name]}

\texttt{sky130\_fd\_sc\_hd}

- foundry
- standard cells
- high density
Understanding the SkyWater PDK Libraries

Digital standard cells

Naming convention:

\[
\text{sky130\_vendor\_library\_type[\_name]}
\]

library: \text{sky130\_fd\_sc\_hd}

cellname: \text{sky130\_fd\_sc\_hd\_nor2\_2}
Understanding the SkyWater PDK

Libraries

Digital standard cell abutment (when placed manually)

correct: abutment boxes touching

too far apart

too close together
Understanding the SkyWater PDK

Libraries

Digital standard cells

Documentation??
Understanding the SkyWater PDK

Libraries

I/O cells

sky130_fd_io

input/output

foundry

Figure 3: GPIO V2 Block Diagram
Understanding the SkyWater PDK

Libraries

I/O cells

sky130_ef_io

efabless

overlay connects clamps to power rail

overlay connects pad to power rail

sky130_ef_io__vccd_lvc_clamped2_pad
Understanding the SkyWater PDK

Libraries

primitive devices and models

sky130_fd_pr

foundry

primitives
Understanding the SkyWater PDK

Libraries

primitive devices and models

sky130_fd_pr

RF frequency reference layout

RF 0.18V pFET
4 fingers
width = 3.0 µm
length = 0.15 µm
Understanding the SkyWater PDK

Libraries

primitive devices and models

sky130_fd_pr

cap_vpp_06p8x06p1_l1m1m2m3_shieldpom4

vertical parallel plate capacitor
(vpp cap)
Understanding the SkyWater PDK

Libraries
- SRAM
  - sky130_sram_macros
  - sky130_fd_bd_sram

openRAM

foundry (not yet available)

dual-port SRAM bit cell
Understanding the SkyWater PDK Libraries

NVDRAM

SONOS nFET transistor

buried floating gate
Open Source Tools and Flows

or... How can you design something?
Open Source Tools and Flows

A simple manual design flow

1. Schematic
Open Source Tools and Flows

A simple manual design flow

1. Schematic

xschem
Open Source Tools and Flows

A simple manual design flow

1. Schematic

- xschem

- ngspice (analog simulation)

- gaw (waveform viewing)
Open Source Tools and Flows

A simple manual design flow

1. Schematic

  ???

  2. Layout
Open Source Tools and Flows

A simple manual design flow

1. Schematic

   export → netlist
   (SPICE format)

   simulate (ngspice)

2. Layout
Open Source Tools and Flows

A simple manual design flow

1. Schematic
   export

   netlist
   (SPICE format)

2. Layout
   generate (magic)

simulate (ngspice)
Open Source Tools and Flows

A simple manual design flow

Parameterized devices in magic
Open Source Tools and Flows

A simple manual design flow

Parameterized devices in magic

selection of devices from SKY130 PDK
Open Source Tools and Flows

A simple manual design flow

Parameterized devices in magic
Open Source Tools and Flows

A simple manual design flow

1. Schematic
   - export
   - netlist (SPICE format)
   - simulate (ngspice)

2. Layout
   - import
   - initial layout
   - generate (magic)
Open Source Tools and Flows

A simple manual design flow

Example initial layout of devices in magic, imported from schematic

transistors

capacitor

resistor
Open Source Tools and Flows

A simple manual design flow

Analog example layout work in progress in magic

resistor

transistors

capacitor
Open Source Tools and Flows

A simple manual design flow

1. Schematic export
(schematic capture)

   netlist
   (SPICE format)

2. Layout export
(extraction)

   netlist
   (SPICE format)
Open Source Tools and Flows

A simple manual design flow

1. Schematic export
   - netlist
     (SPICE format)

2. Layout export
   - LVS (netgen)
   - netlist
     (SPICE format)
Backend validation DIY

Running a simple example through the entire validation flow

1. xschem     Create a schematic
2. ngspice    Simulation validation
3. magic      Layout, DRC validation
4. netgen     LVS validation