Creating a World where a 14-year-old Designs a Chip

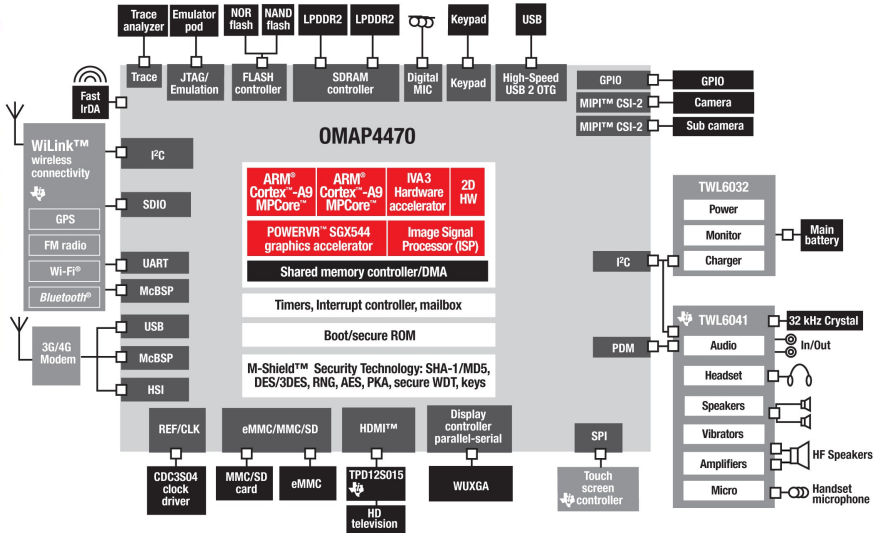
Mohamed Kassem

Cofounder & CTO, EFABLESS.COM

mkk@efabless.com

Mohamed

I designed chips for stuff



Kindle Fire HD

Kindle Fire

Droid Razr

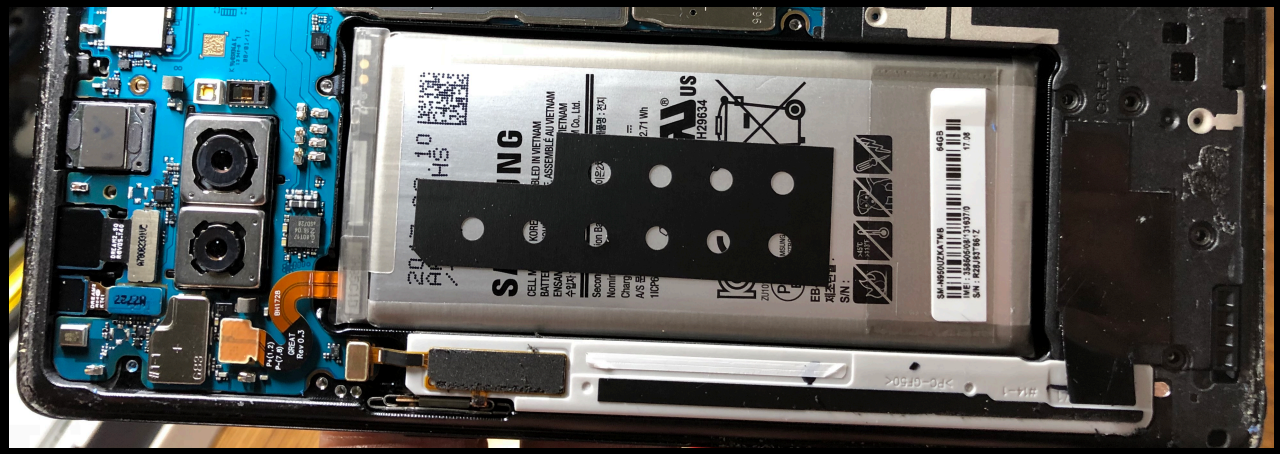
Droid X

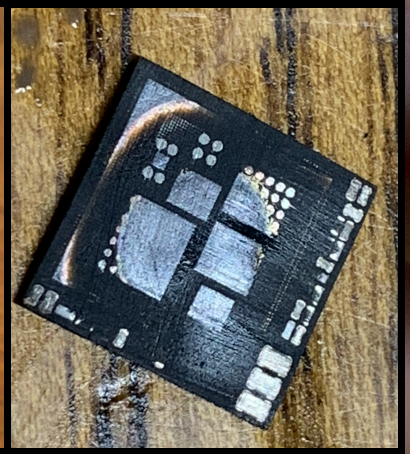
Huawei Ascend D1

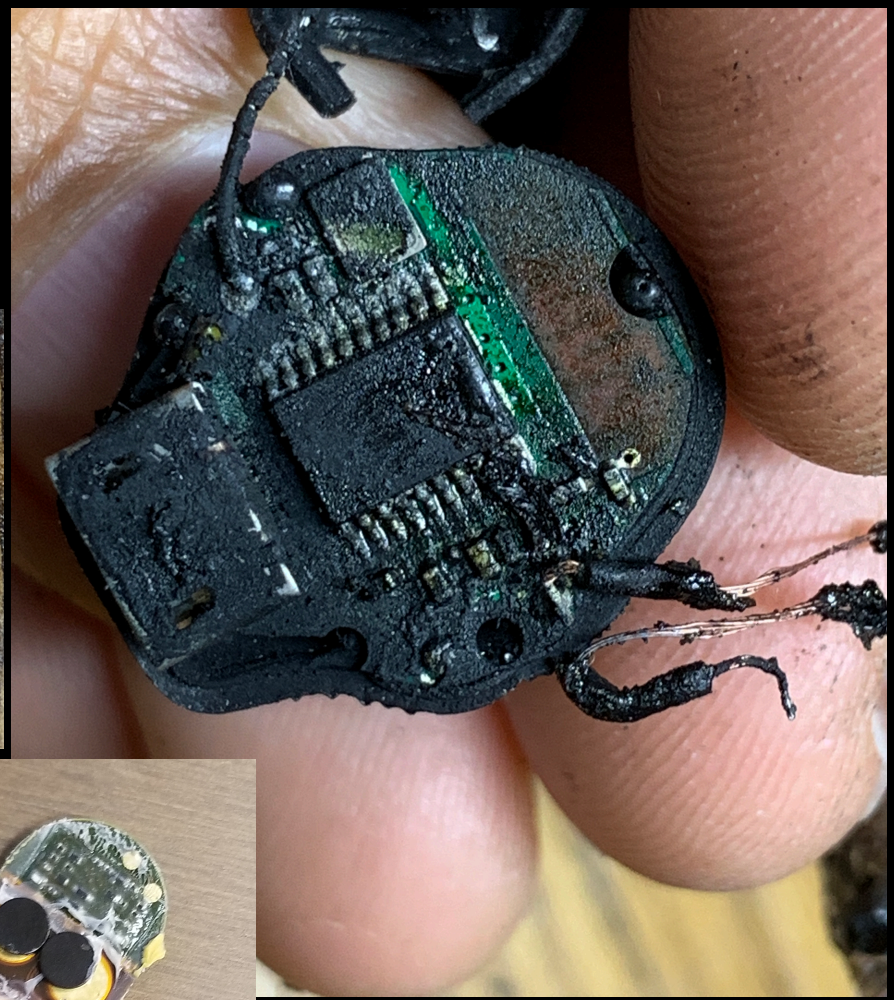
Motorola - Droid Milestone

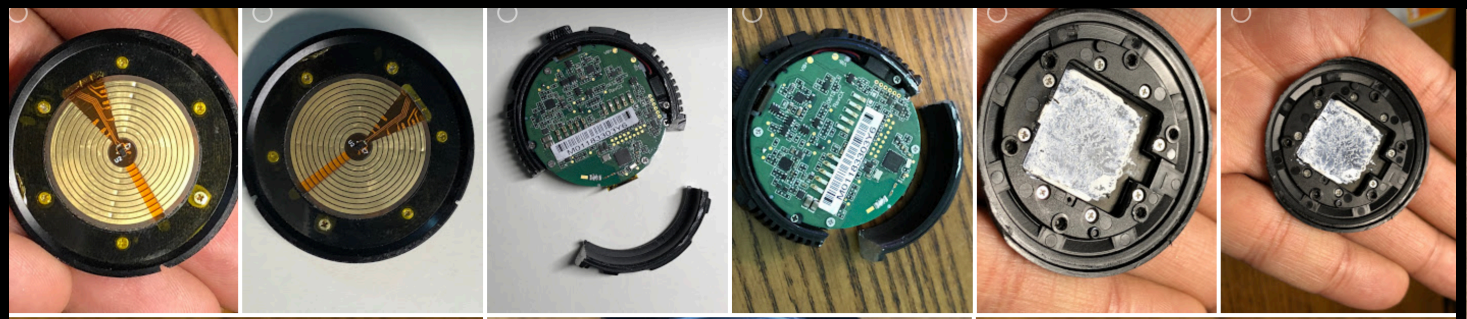
Motorola - Droid Bionic

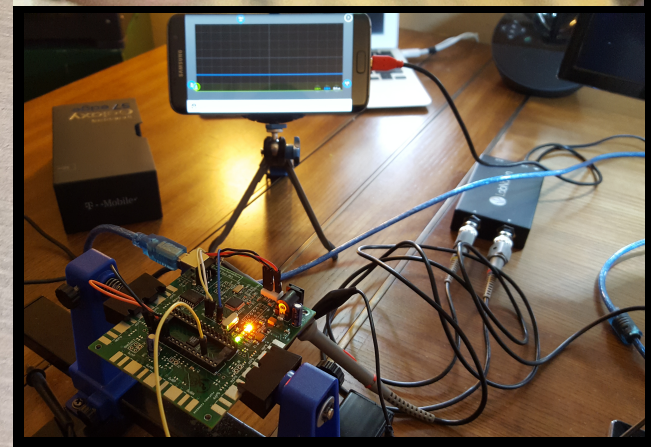
I break other stuff

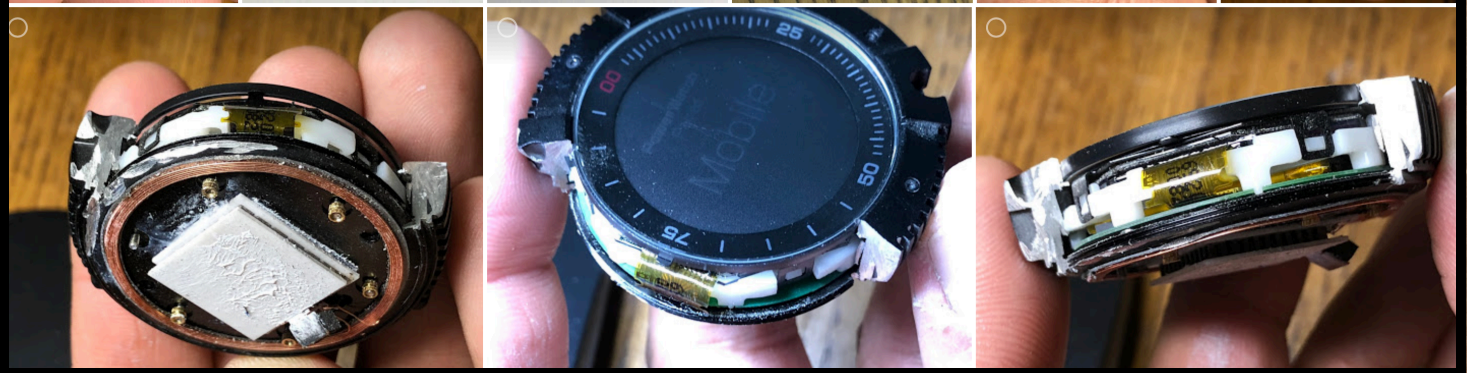








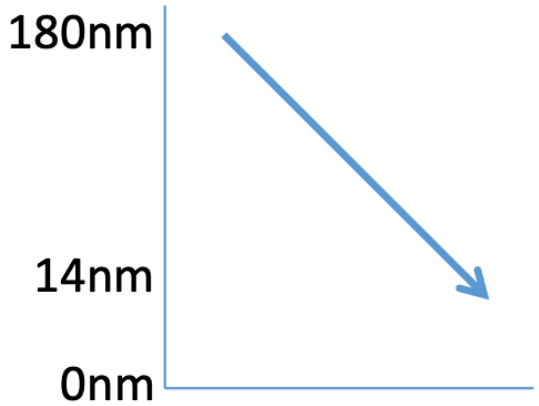




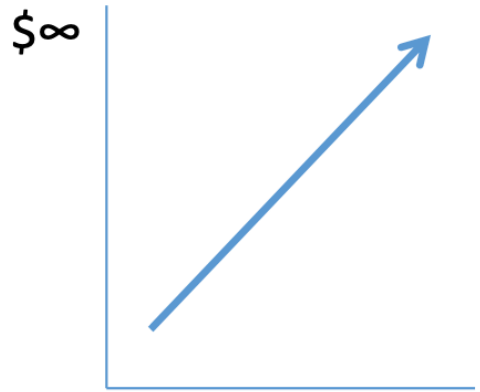
INDUSTRY CHALLENGES

Why is chip design and making has become less accessible to the world and stifling innovation?

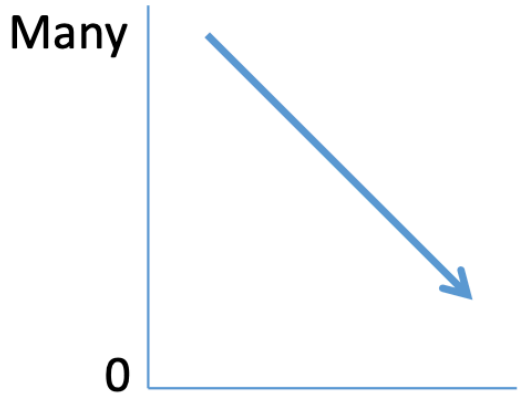
CHIP INDUSTRY TRENDS



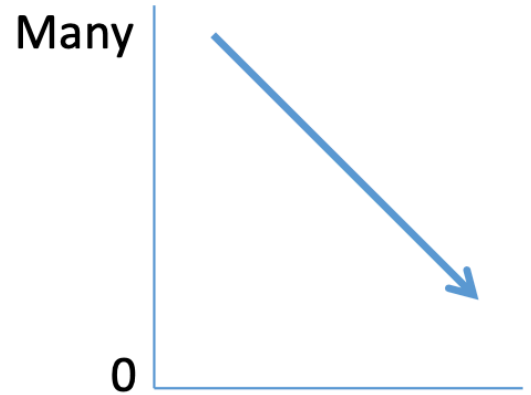
As Transistor Size Goes To Zero



Development Cost Goes to Infinity



Number of Customers Goes to Zero

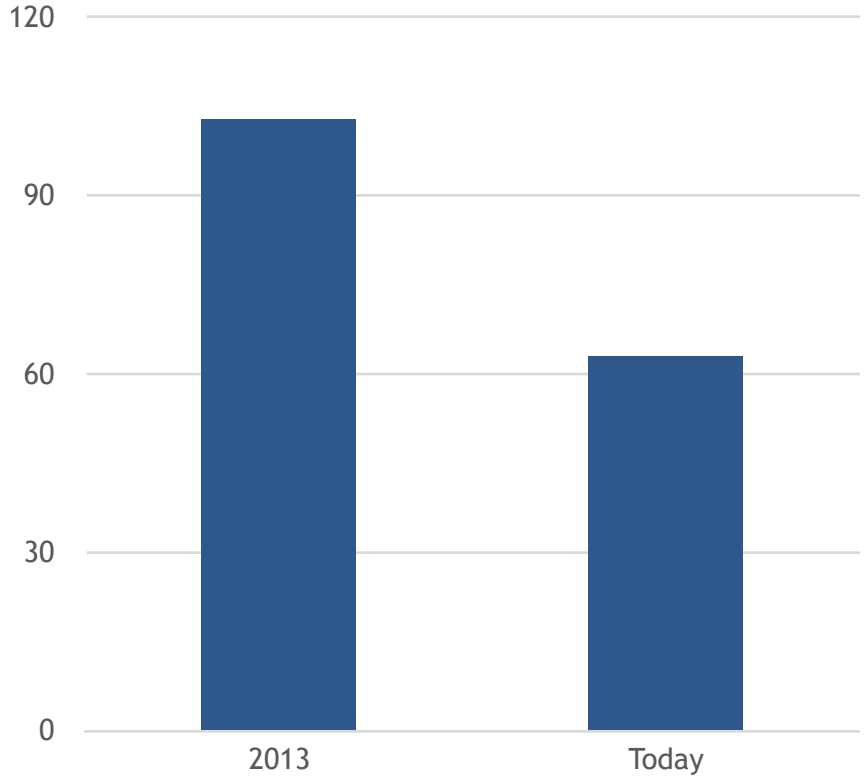


Number of Designers Goes to Zero

Source: Mike Noonen

CHIP INDUSTRY TRENDS

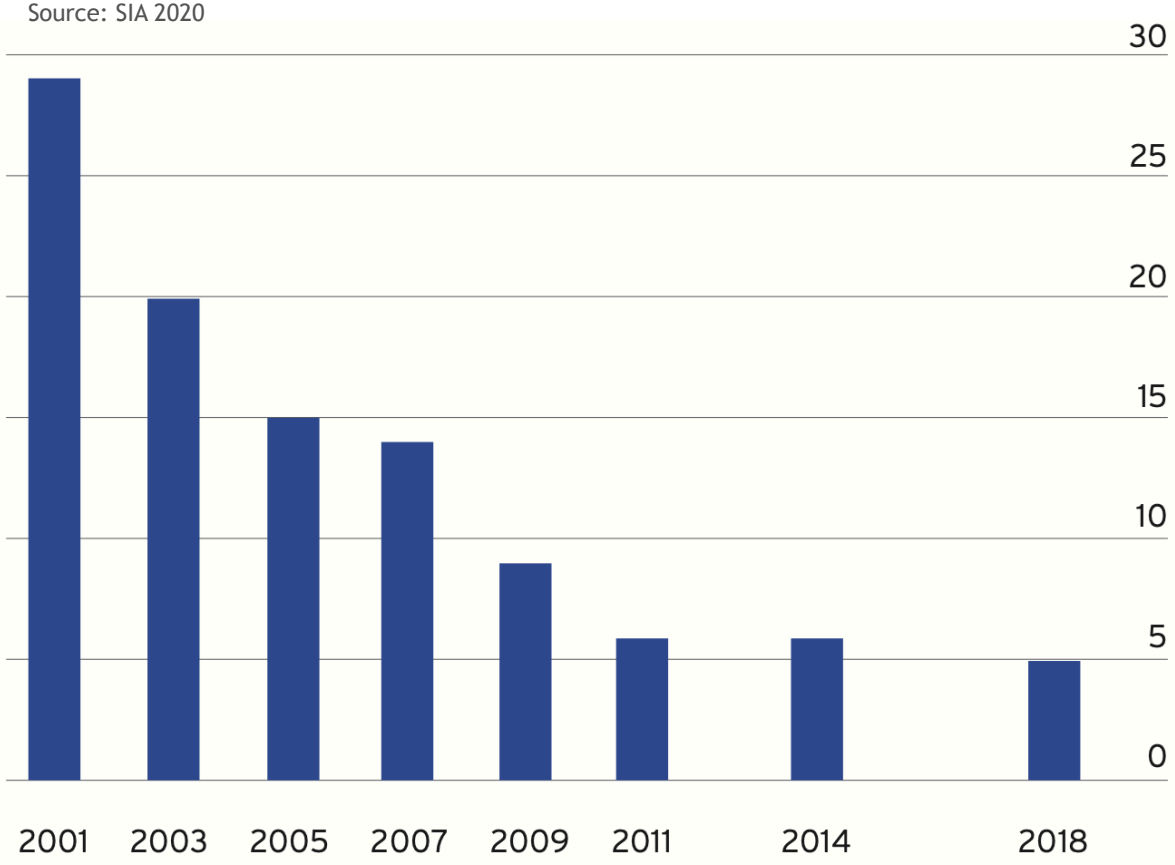
Source: Morgan Stanley



Semiconductor industry consolidation

Semi Companies Larger Than $100M

Source: SIA 2020



Semi Companies Delivering Lead-in Edge Tech

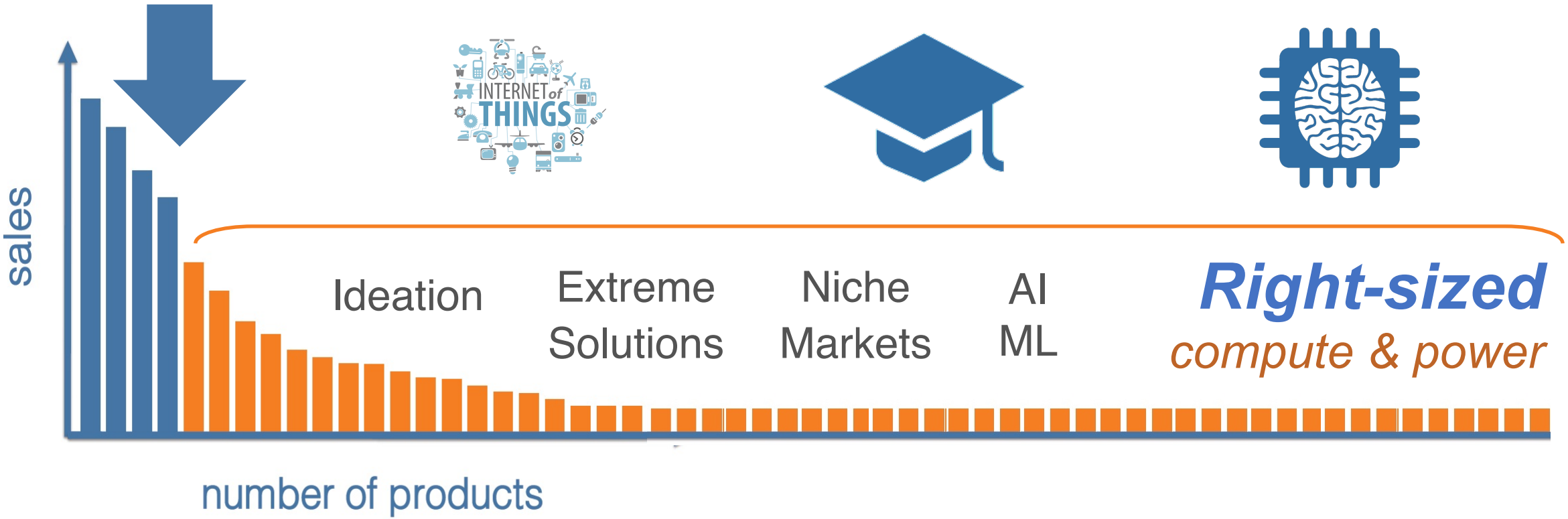
efabless.com

WE NEED A 1000X PRODUCT DEVELOPERS

Long-tail Innovation is required on a massive scale to meet demand

10,000's of Products

Markets Served By Traditional Methodology



efabless.com

Creating a World where a 14-year-old Designs a Chip

What about that?

“In the beginner’s mind there are many possibilities, but in the expert’s there are few”

— Shunryu Suzuki, Zen Monk 1904-1971

Why is it “hard” to design chips

Limited access to **knowledge**

Limited access **technology**

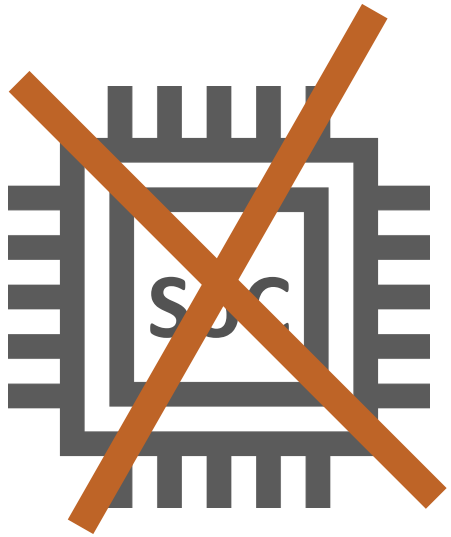
Costly **manufacturing**

Too many things to bring together ….. too complicated

..... too complicated

1. Market Access — $$
2. Expertise — $$+Time
3. IP — $$+NDA
4. EDA Tools — $$+NDA
5. PDK — NDA
6. AFFORDABLE MFG/ASSY/TEST — $$

At least these **6** things must exist to create useful chips



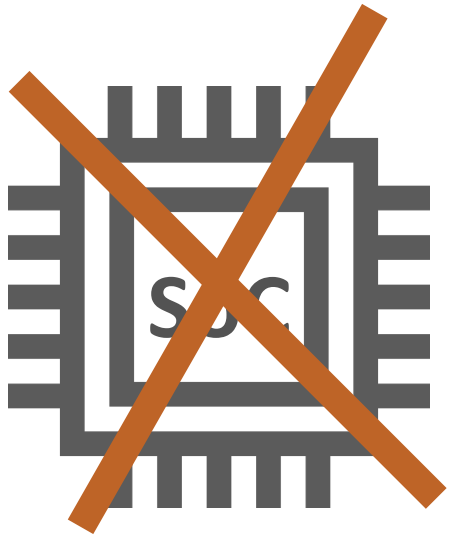
YOU **CAN'T** BUILD A USEFUL **CHIP**

efabless.com

….. still too complicated

1. Market Access — $$
2. Expertise — $$+Time
3. IP — $$+NDA
4. EDA Tools — $$+NDA
5. PDK — NDA
6. AFFORDABLE MFG/ASSY/TEST — $$

Partial solutions **don't work**



STILL **CAN'T** BUILD A USEFUL **CHIP**

efabless.com

We need to

Simplify the process of **Chip** **creation** and open it **to** **Everyone**

How would we simplify chip design?

Think what **app stores** did to software innovation.

Simplified *(democratized)*

the **development tools**

the **business process**

the **connection to customers**

Think what **app stores** did to software innovation.

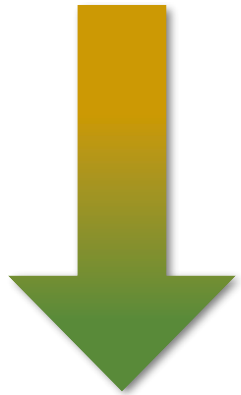
Simplified (democratized)

the **development tools**

the **business process**

the **connection to customers**

Thousands



At least

Millions

Including kids

efabless.com

Think that we apply the same approach to chip design

***Simplify** (democratize)*

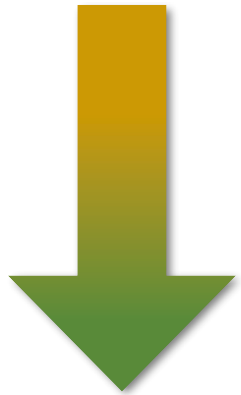
the **access** to chip design tools & PDK

the **business process**

the **connection to customers**

Think that we apply the same approach to chip design

Thousands



Simplify (democratize)

the **access** to chip design tools & PDK

the **business process**

the **connection to customers**

At least

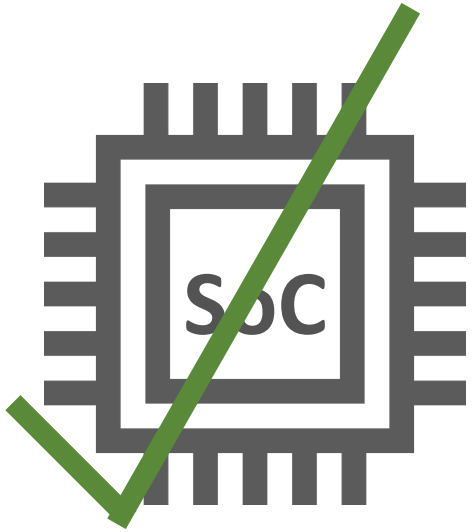
Millions

Including kids

..... what we need ...

1. **Market Access** — Open Online Marketplace for Chips Like the App Stores
2. **Expertise** — Open Community Collaboration Clusters of global teams
3. **IP** — No NDA or Access Fee Open Source or Obfuscated
4. **EDA Tools** — No Licensing Cost = Open Source Low Licensing Cost = Cloud-based
5. **PDK** — No NDA = Open Source or Obfuscated to protect IP
6. **AFFORDABLE** MFG/ASSY/TEST — No NDA = Open Source or Obfuscated to protect IP while enabling designers to use it

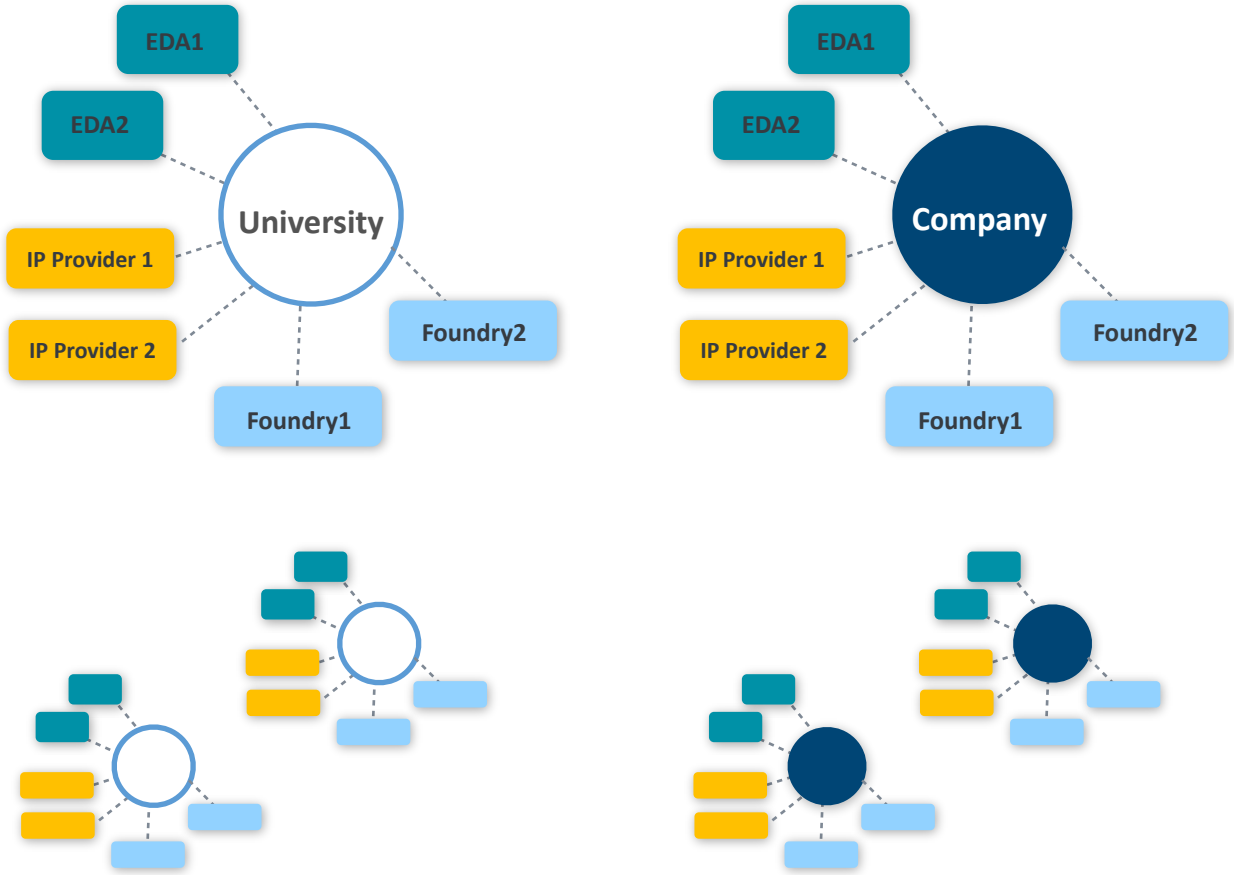
Solve **1** at time



CAN BUILD USEFUL **CHIPS**

efabless.com

TODAY'S APPROACH



Every design entity needs to establish **independent** business and contractual (including NDA's) relationships with **multiple** IP providers, EDA vendors, and foundries

efabless.com

CLOUD - BUT NOT FOR COMPUTE

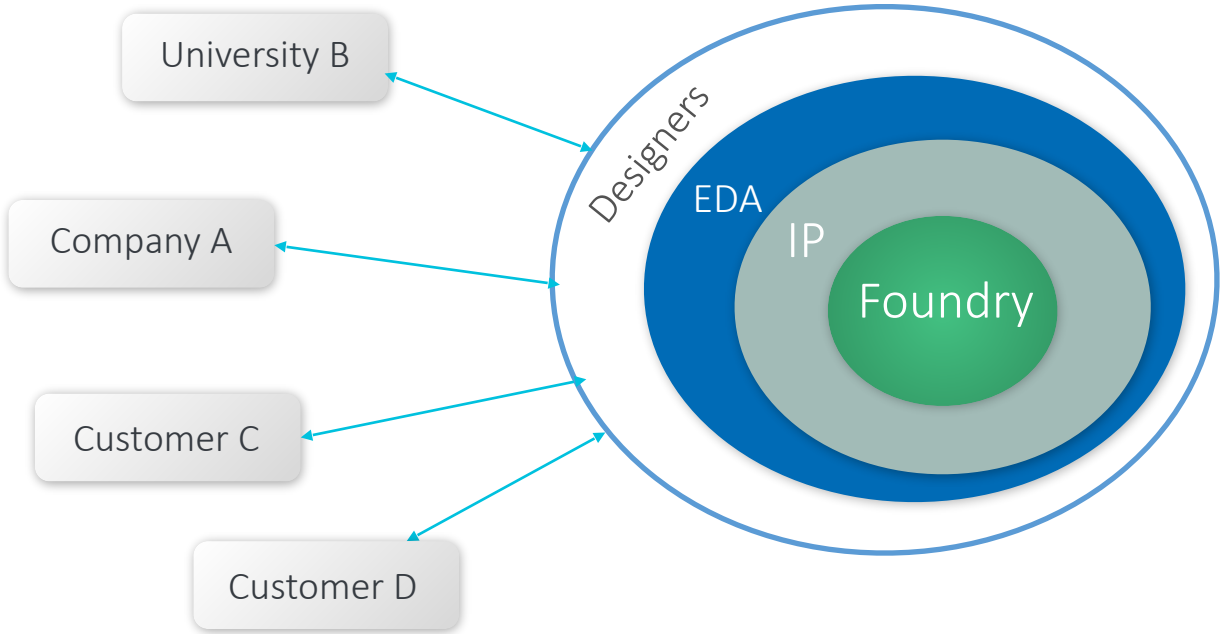
Secure, cloud-based platforms, all the resources and processes required to design ASICs and IP are offered in one place while maintaining the “fire-walling" of IP information and with enhanced traceability and on-demand elasticity.

Enabling a worldwide network of developers and customers to collaborate, model, and verify custom SoCs. When custom SoC creation becomes less risky and more cost-effective, innovation is unleashed.

Scalable & elastic design/development capacity

Facilitates Collaboration, reference designs and design re-use accelerates development and reducing costs

Risk reward sharing enabled by consolidated contribution tracking and design obfuscation



efabless.com

FOUNDRY DATA OBFUSCATION

Mask-geometry layout is foundry proprietary.

How can you design an entire chip and submit to the foundry for fabrication without signing an NDA, purchasing commercial tools, and installing PDKs?

All analog cells at the transistor level are abstracted views using information from the corresponding LEF files and simulation models

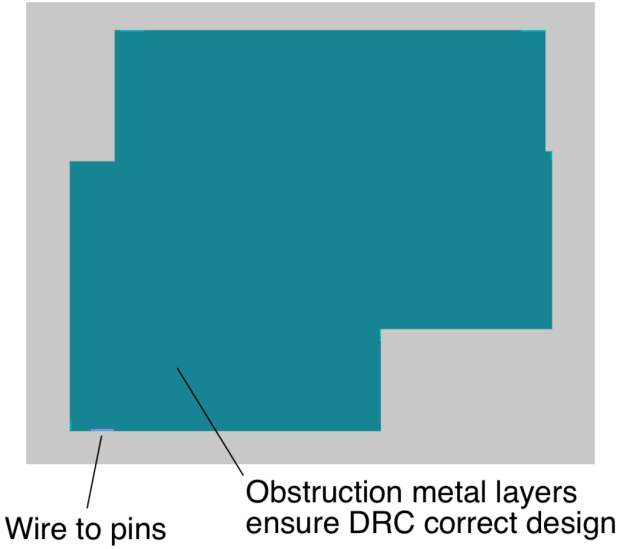
XFAB MIXED-SIGNAL FOUNDRY EXPERTS

GLOBAL FOUNDRIES

3.3V ADC, Original vendor layout:

(Layout blurred to protect the identity of the victim)

3.3V ADC, Abstracted layout (from LEF view):



efabless.com

DESIGN IP OBFUSCATION

The target process: **X-FAB XH018**

Base MOS LP (low power) option

6 metal stack (5 standard route layers, 1 thicker top

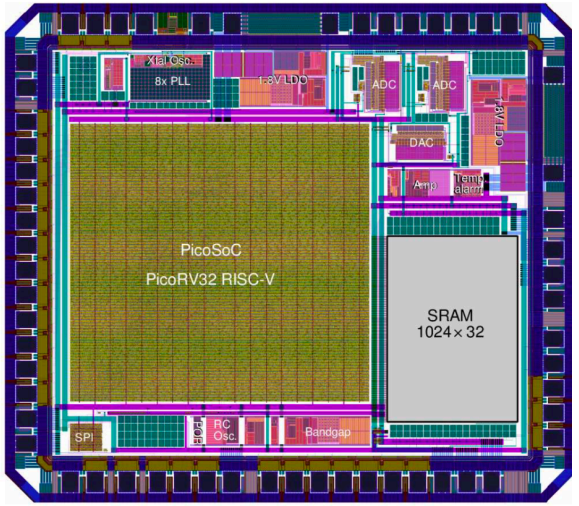
The proprietary data:

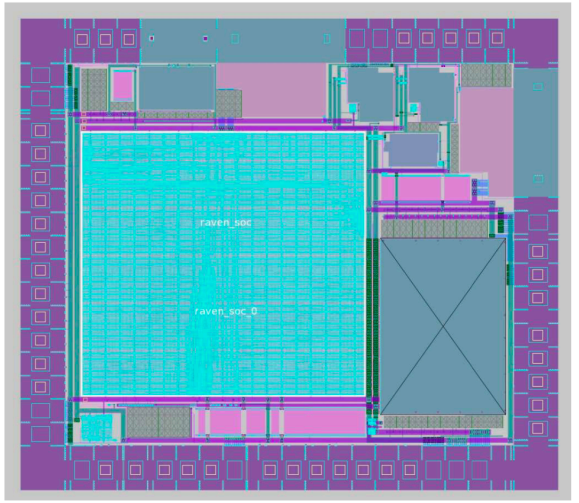
X-FAB digital standard cells

X-FAB I/O Cells(3.3V with both 3.3V and 1.8V core)

X-FAB Analog IP

X-FAB SRAM (from memory compiler)

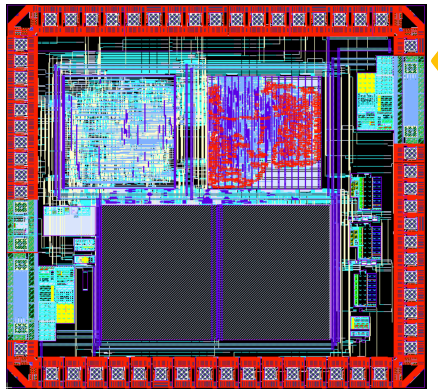


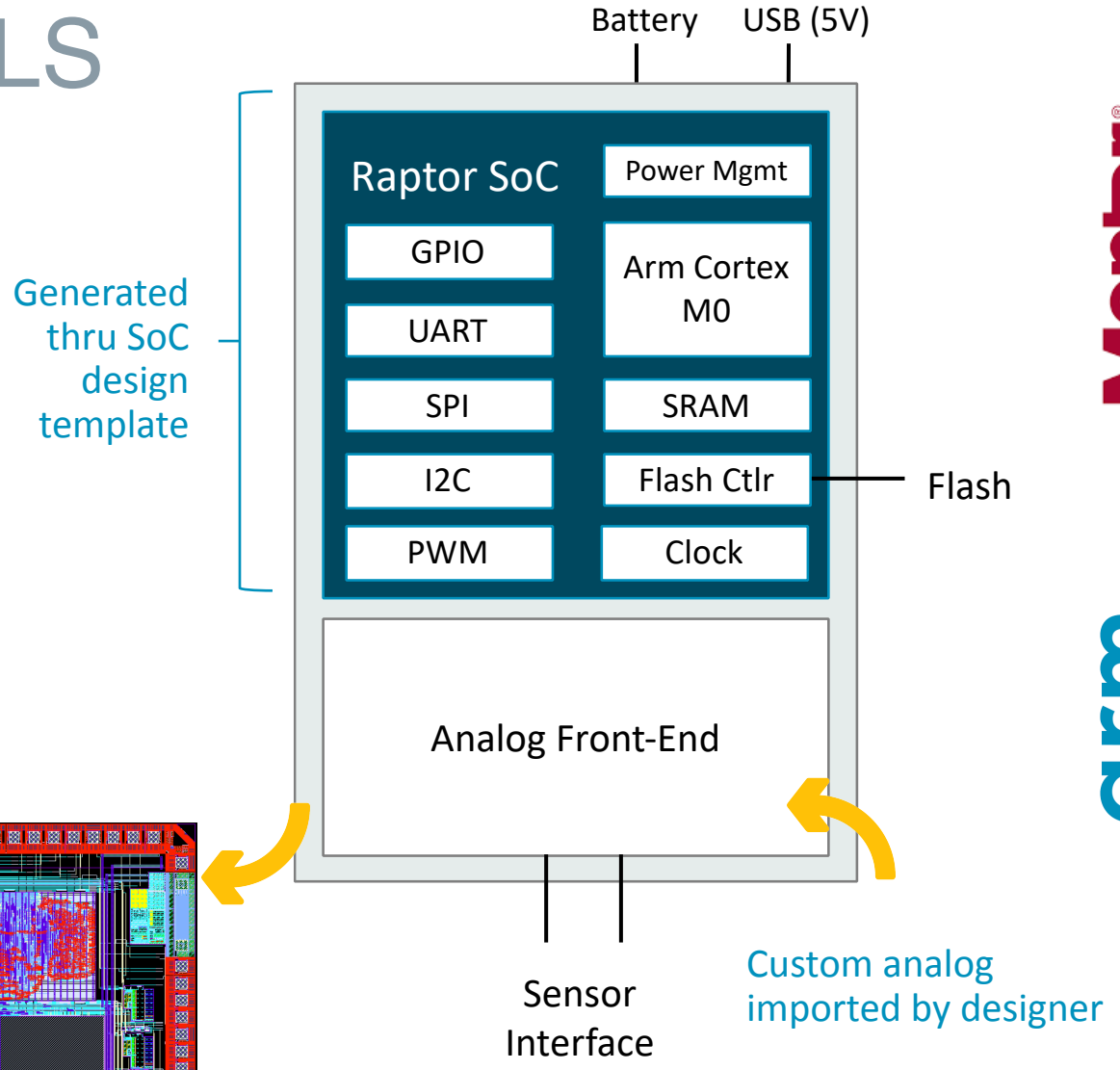


PROPRIETARY EDA TOOLS

SIEMENS EDA

- Seamless integration in **efabless** platform
- Complete Mentor/Tanner design flows
- Process technology support 65 through 22FDX
- Optimized SoC template implementation





KEY PLATFORM FEATURES

- **Instant access** to the platform - **less than 5 minutes**
- **Option for No-NDA** Access To Foundry Process Technology - efabless' obfuscation technology
- **No Upfront Cost** Microelectronics Design Tools (EDA) using proven Open Source & Proprietary SW
- **No IP licensing cost** for prototyping enabled by IP obfuscation
- **No cost Try Before Use/Buy** for proprietary designs while protecting IP information and ownership
- **Low cost prototype manufacturing** options as multi-project-wafers (MPW) including bench setup
- **Built-in project collaboration**, forums and management enabling **effective knowledge exchange**
- Provides **results-based reputation/certification** indicators based on real user design performance
- Supports online **project-based** - design, verification & validation of microelectronics
- **Elastic and scalable** infrastructure to support 10,000's of users
- Marketplace **connecting** designers to potential customers for their **unique design innovation**

How do we simplify chip design? **#1**

Open access to technology PDK

No NDA, nothing to sign - it's Open Source

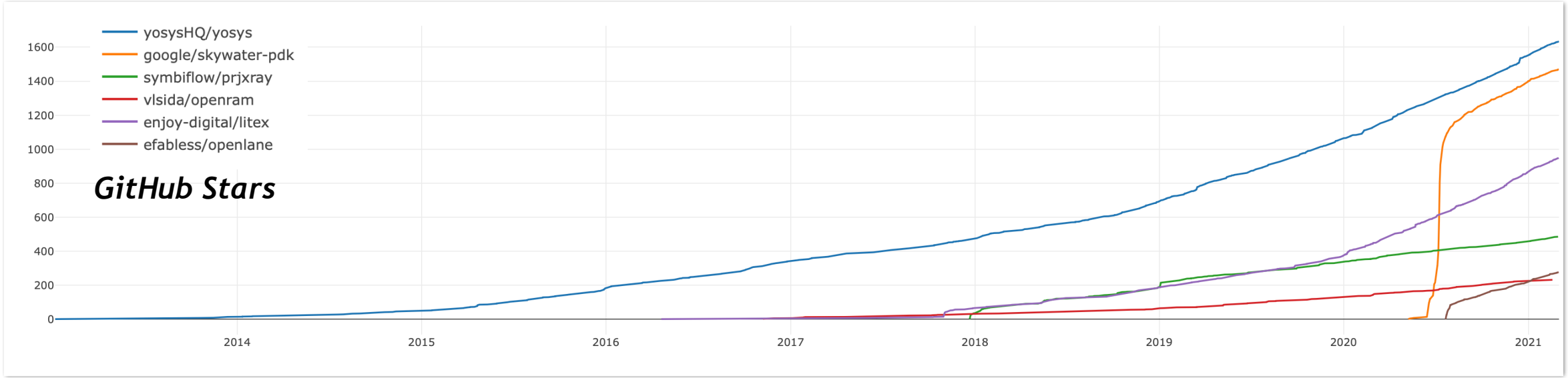
$ git clone https://github.com/google/skywater-pdk

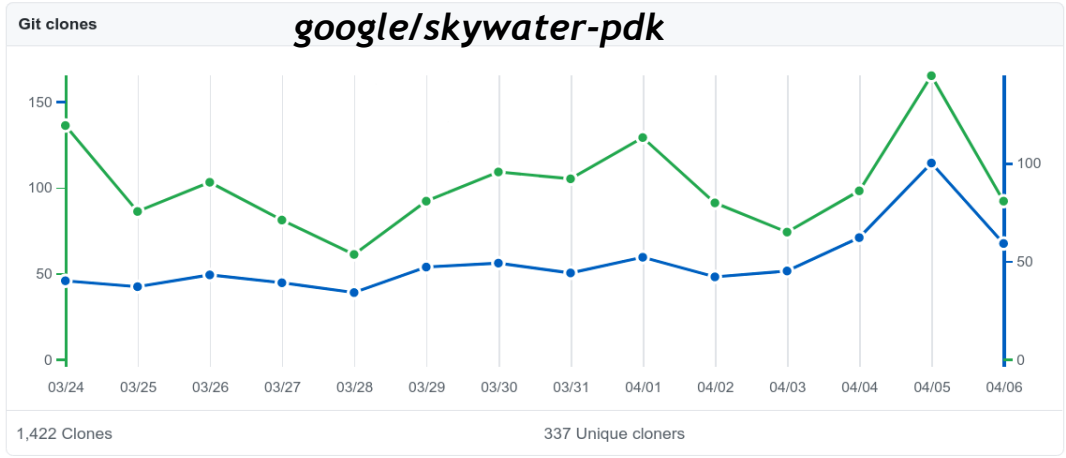
Technology information availability is virtually limitless

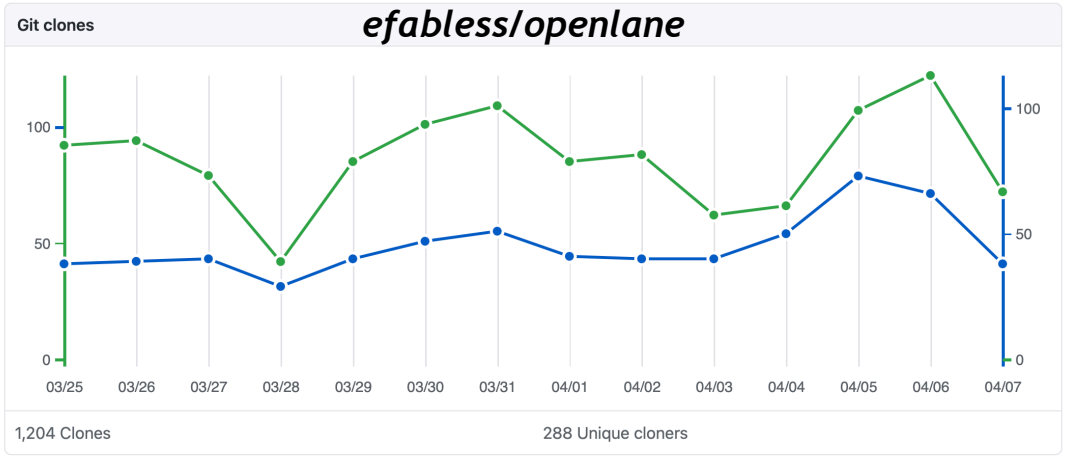
*Leading to **massive open collaboration***

efabless.com

SKY130 OPEN SOUCRE PDK WENT VIRAL







efabless.com

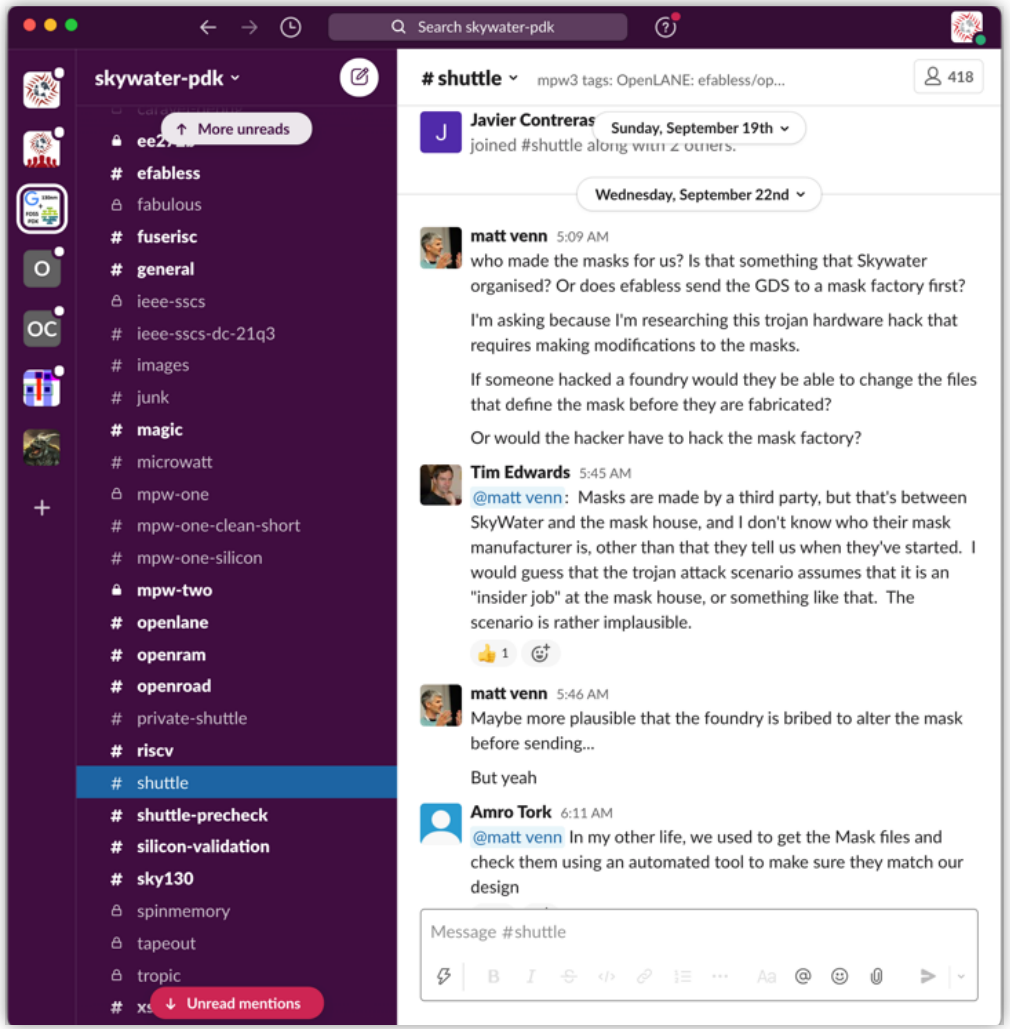
LIVE COMMUNITY SLACK SPACE

2,800+
COMMUNITY MEMBERS

100+
CHANNELS & TOPICS

Join SkyWater-PDK Community

https://join.skywater.tools



efabless.com

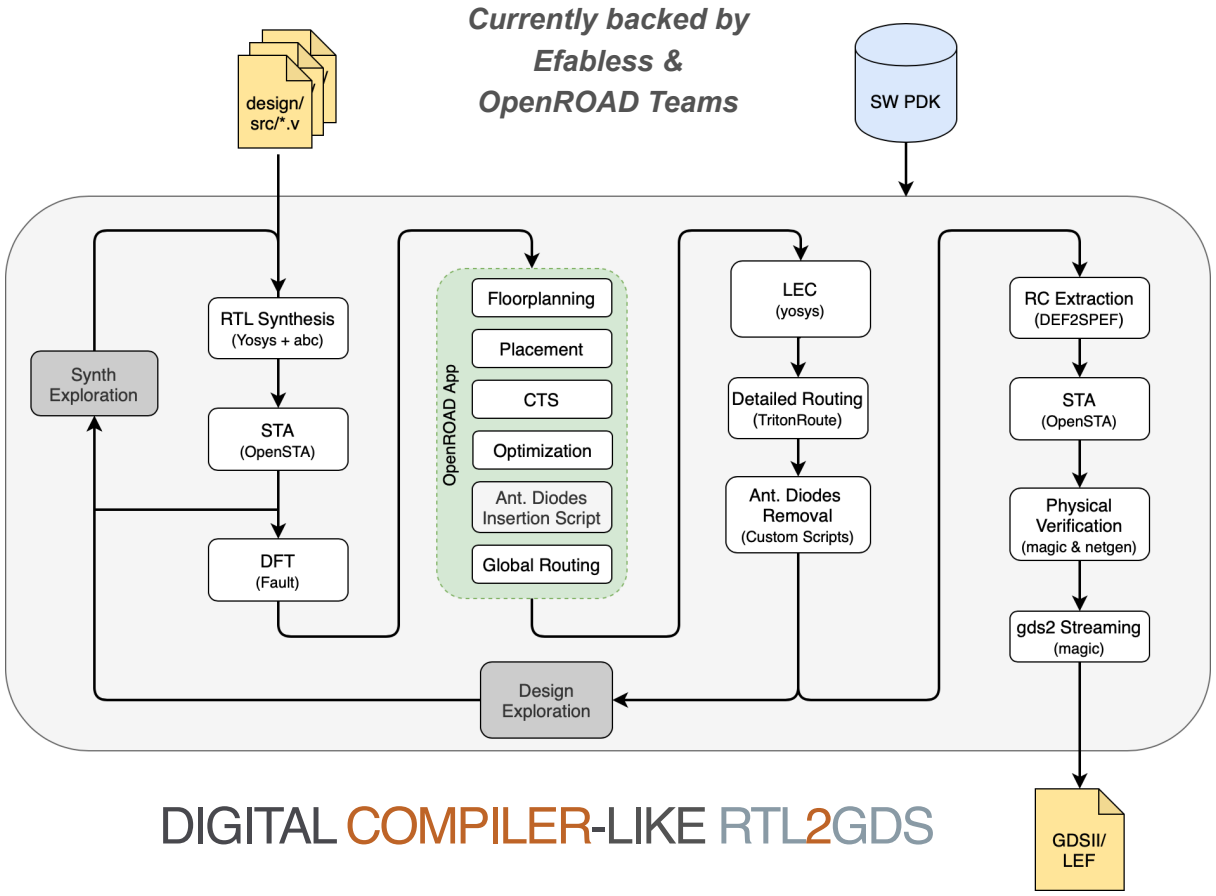
How do we simplify chip design? **#2**

Codify and abstract knowledge

Automate code-to-chip like a ***GNU software compiler***

It opens the door for software developers to generate hardware

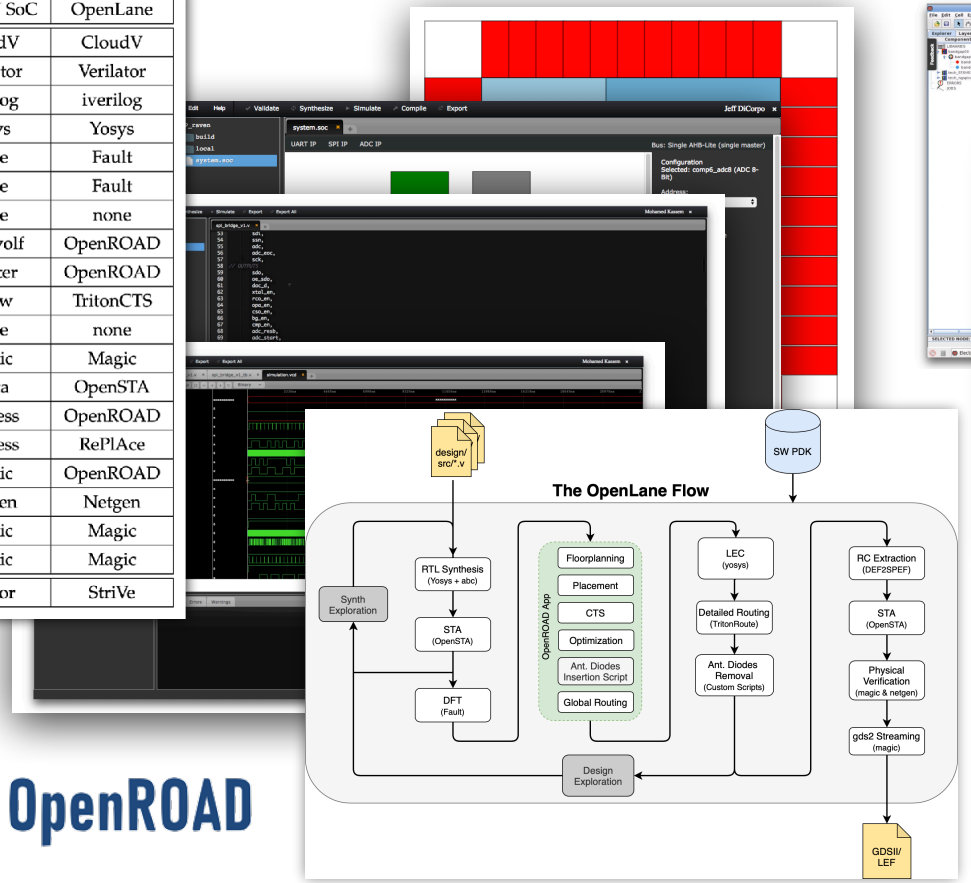
That's at least a **1000X** more potential designers!

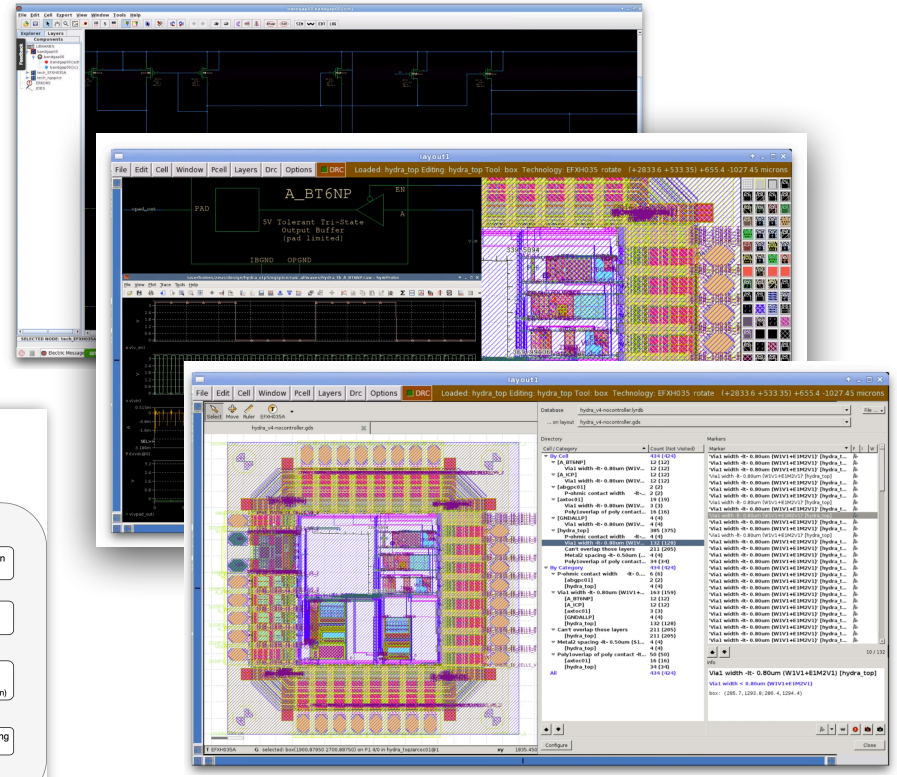


efabless.com

OPEN SOURCE DESIGN FLOWS & TOOLS

DESIGN STEP	Qflow	CloudV SoC	OpenLane
System Design	N/A	CloudV	CloudV
RTL Lint	Verilator	Verilator	Verilator
RTL Simulation	iverilog	iverilog	iverilog
Logic Synthesis	Yosys	Yosys	Yosys
DFT Scan Insertion	none	none	Fault
DFT ATPG	none	none	Fault
Formal Verification	none	none	none
Placement	graywolf	graywolf	OpenROAD
Routing	qrouter	qrouter	OpenROAD
CTS	Qflow	Qflow	TritonCTS
Dynamic EMIR	none	none	none
Extraction	Magic	Magic	Magic
Timing Analysis	Vesta	Vesta	OpenSTA
Floorplanning	Magic	efabless	OpenROAD
Top-Level Placement	Magic	efabless	RePlAce
Top-Level Routing	Magic	Magic	OpenROAD
LVS	Netgen	Netgen	Netgen
DRC	Magic	Magic	Magic
GDS	Magic	Magic	Magic
SoC	Raven	Raptor	StriVe





- SoC Editor
- RTL Simulation
- Synthesis
- GL Simulation

OpenROAD

- Schematic Capture
- SPICE Simulation

- Mixed-Mode Simulation
- Parasitic Extraction
- Physical Verification

Silicon Compiler Coriolis OpenLane

efabless.com

Why does it matter to have tools open-sourced?

***Limitless availability** which fosters **spontaneous idea realization** and open **collaborative** development by community*

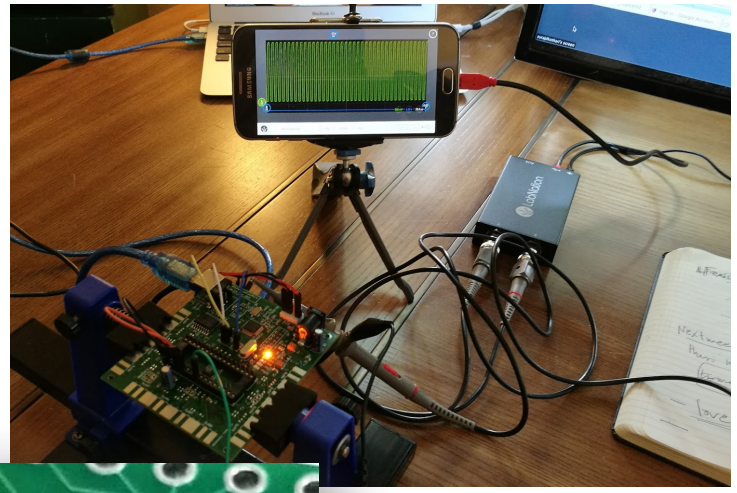
EXAMPLE OPEN SOURCE SoCs

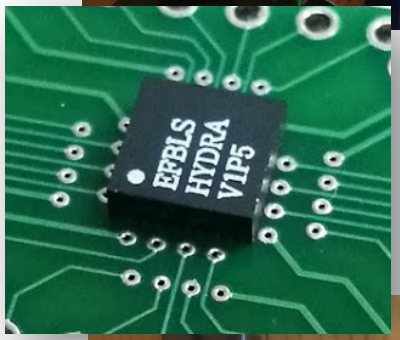
HYDRA REFERENCE ASIC

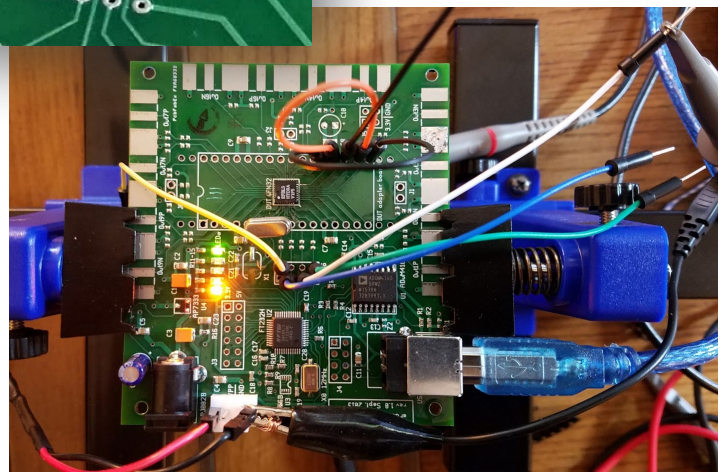
- **A reference ASIC platform for testing A/MS IP modules**

- **Features**
  - Based on a standardized pad frame - used for multiple core chip.
  - Designed along with it's companion test PCB & test SW.
  - Interfaces to a personal computer via USB.
  - Top-level is available to community members to modify & re-shuttle.
  - Intended as a design & test learning vehicle for analog/ms blocks

- **Included IP modules**
  - 10-bit SAR ADC - aadcc02
  - 10-bit voltage-scaling DAC - adacc02
  - GP Low Power BGAP - abgpc01
  - GP AMP - aopac01
  - GP Bias Cell - abiac01
  - SPI Controller - efdspi001
  - Dynamic Power-on-Reset - aporc01
  - LP CMOS Comparator - acmpc01
  - 14KHz RC Oscillator - arcoc01
  - 1-6MHz Crystal Oscillator - axtoc01







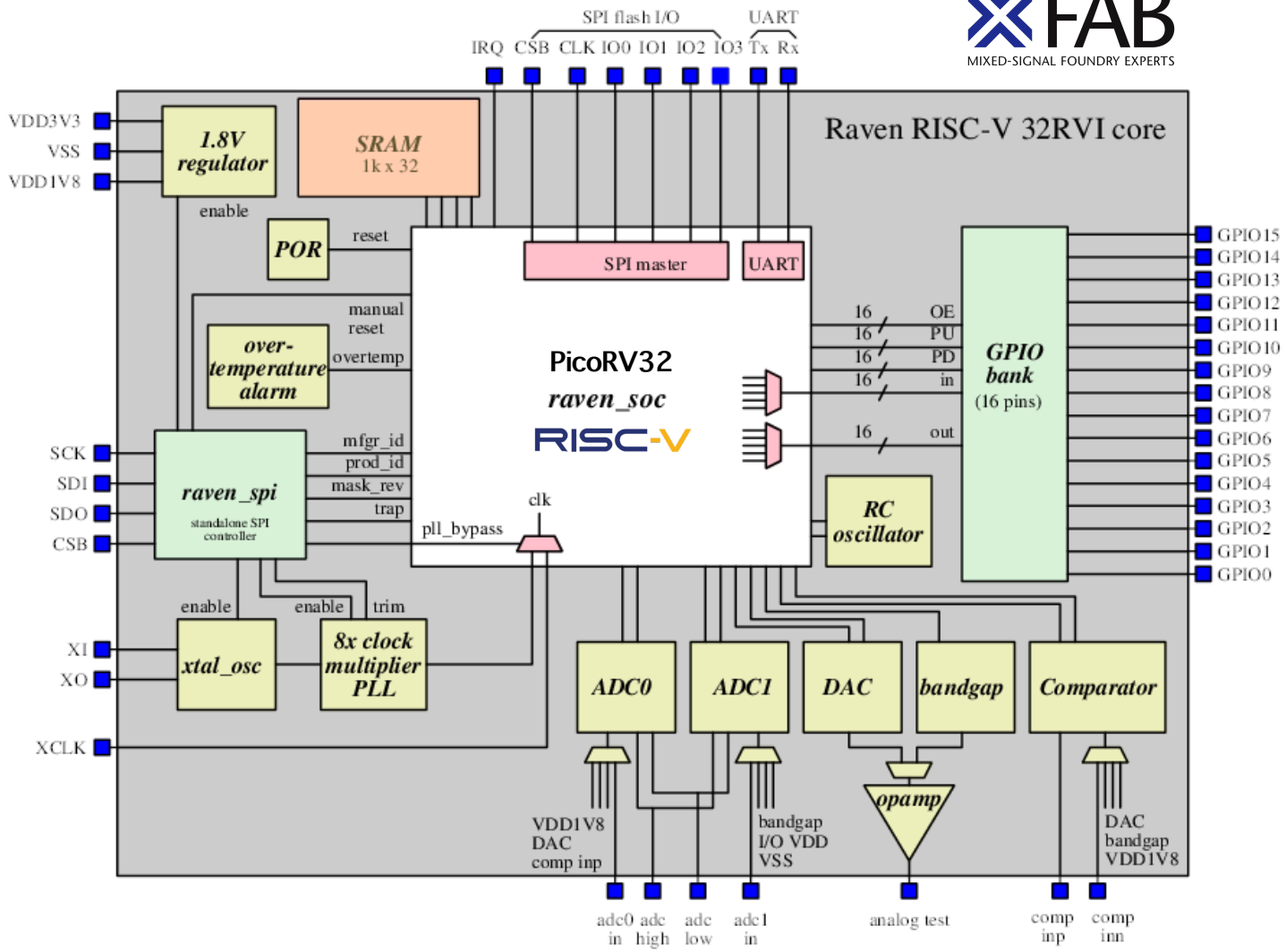
efabless.com

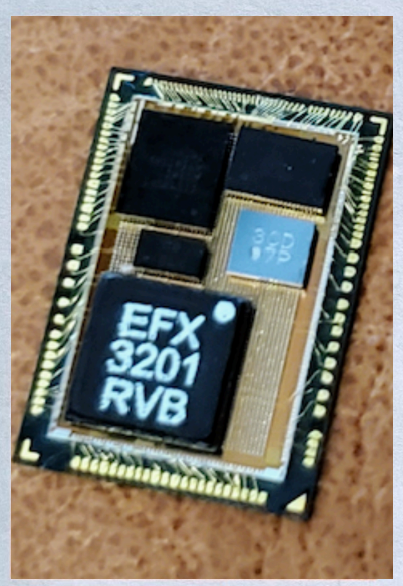
RAVEN - 32-bit RISC-V uC

Key Features

- RISC-V CPU (**PicoRV32**)
- SRAM 32x1024
- 100 MHz clock rate
- Programmable clock source
- 16 channels GPIO
- 2 ADCs
- 1 DAC
- 1 Comparator
- Over-temperature alarm
- 100 kHz RC oscillator
- Programmable functions on GPIO outputs
- Programmable interrupts on GPIO inputs

http://github.com/efabless/raven-picorv32





Chip Description

EFX3201
RISC-V SOC

iCE40UP5K
An ultra-low power FPGA

AT25SL321
A serial interface Flash memory device

SIT8021
The industry's smallest and the lowest power MHz oscillator.

W25Q80D
A serial Flash memory

GEM2

antmicro

efabless.com

zGlue

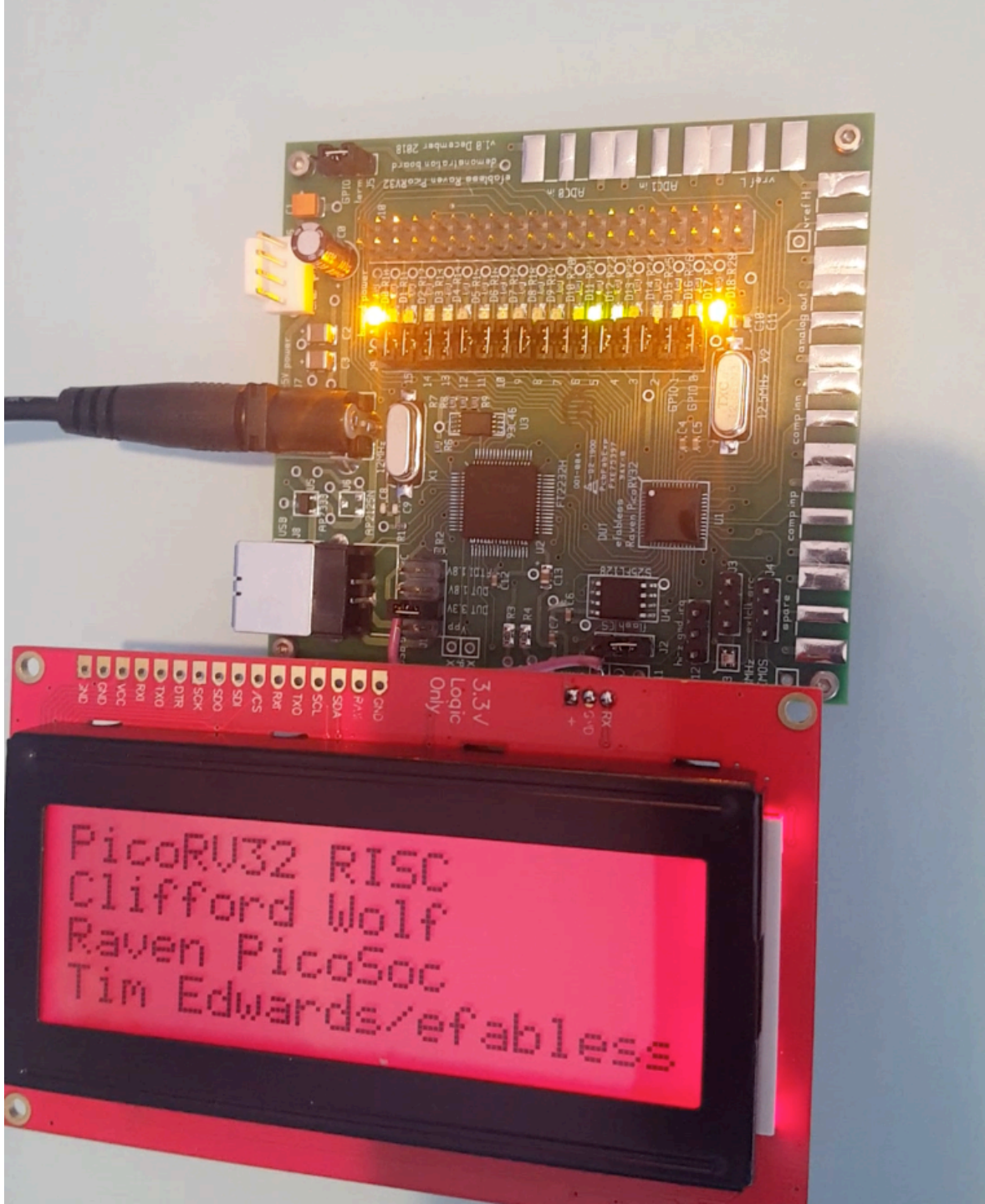
GEM2 is a RISC-V based heterogeneous, chip scale system integration solution targeting edge AI consumer electronics, industrial IoT, medical devices and mobile application. GEM2 consists a series of chiplets that, along with zGlue Smart Fabric, incorporates of iCE40 FPGA and Efabless Raven RISC-V chip which reduces the barrier of developing a custom SOC. It's tiny, secured and fast time to market.

efabless.com

RISC-V

FAB

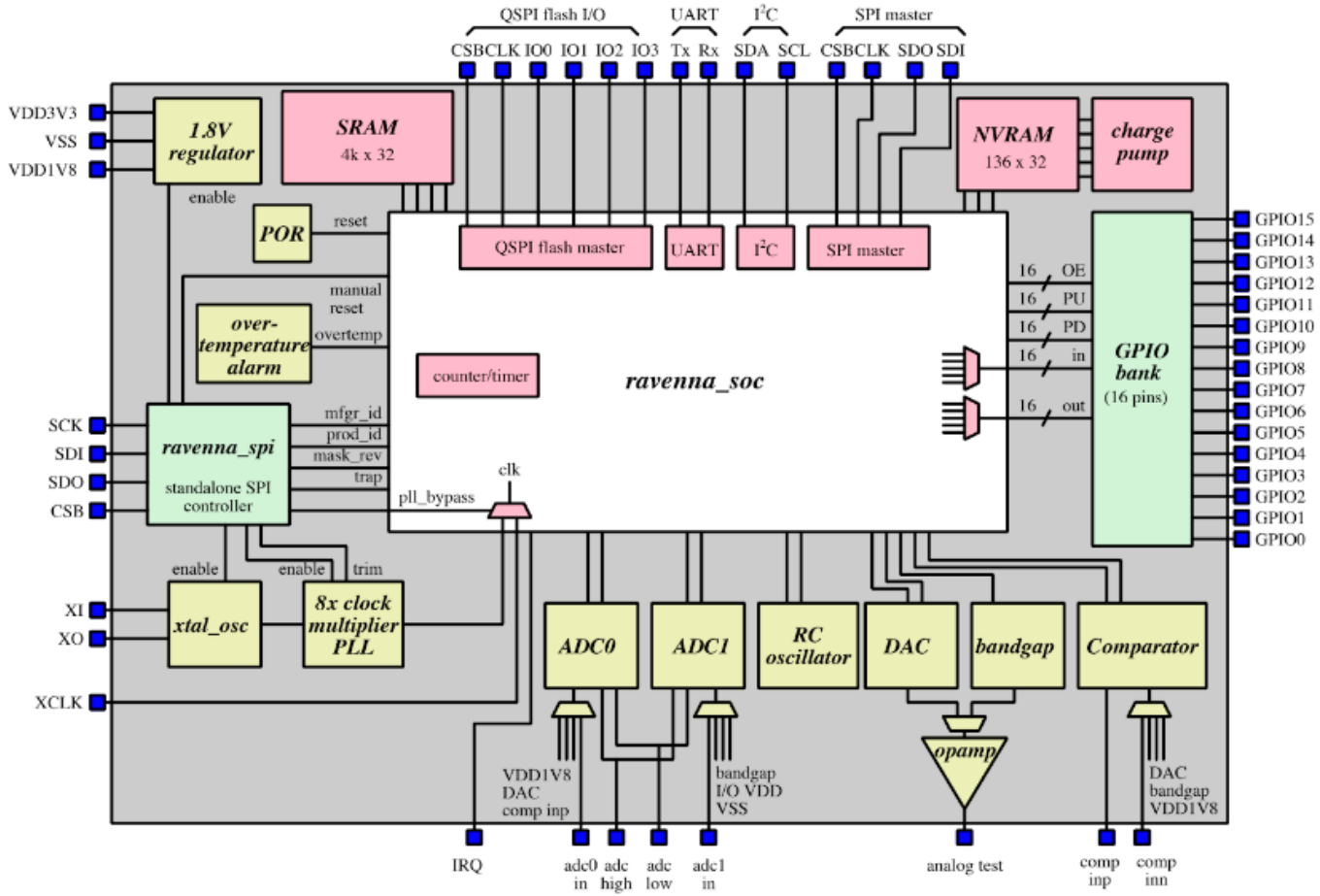
open hardware



RAVENNA - 32-bit RISC-V uC

Key Features

- RISC-V CPU (**PicoRV32**)
- 2 10-bit SAR ADCs
- 1 10-bit DAC
- 1 analog comparator
- 1 100kHz RC oscillator
- 1 1.235V band gap reference
- 1 high temperature alarm
- 1 QSPI flash controller 1 UART
- 1 SPI master
- 1 I^2C master
- 1 counter-timer
- 16 general-purpose digital input/output channels
- 4k word (4096 bytes × 32 bits) on–board SRAM
- 136 (128) word (128 bytes × 32 bits) on–board NVRAM



efabless.com

Example Of Commercial Interest in Open Source ASIC Design

Open Source Based Privacy Protection for IoT

Observation

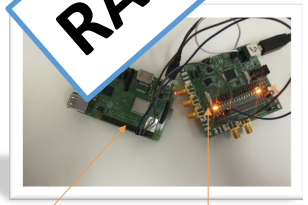
General Purpose Analog Functions can do the job

NEC

AES Authentication

RAVEN

- Authentication / Identification / Signature
  - Confirming the user is who or what is stated
  - chip encrypts data with an 'identity key'
  - Key is symmetric
- R-PI have authentication requests verified by the Raven processors externally



Source sends identity request command

Input: 128 bit

Input is encrypted with the identity key

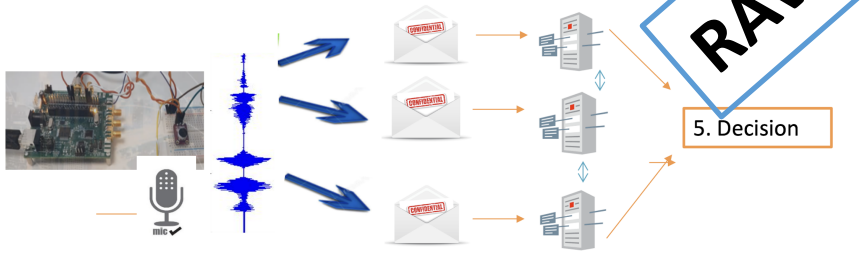
Source can validate results

Line of codes	Max RAM
1.2k	~128 bytes

Audio Sampling and Transfer

RAVEN

1. Data is recorded 8Hz
2. Turn data into digital stream
3. Split to 3 MPC 'random' shares
4. Sent to 3 different servers on cloud

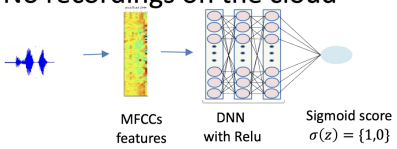


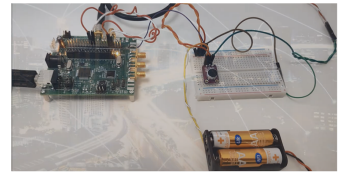
A/D #bits	Samples per second
8	8Khz

Audio Sampling and Transf

RAVEN

- Private voice detection
- Private keywork recognition
  - Keyword is located in the recording
  - No recordings on the cloud





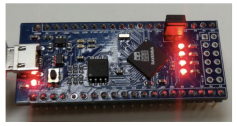
- sMPC random shares are just random numbers

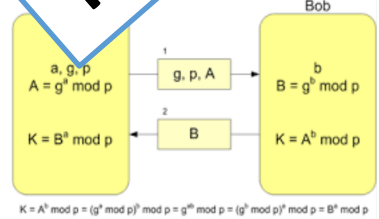
Line of codes	Max RAM
~100	<100 Bytes

PKI Implementation

RAVENNA

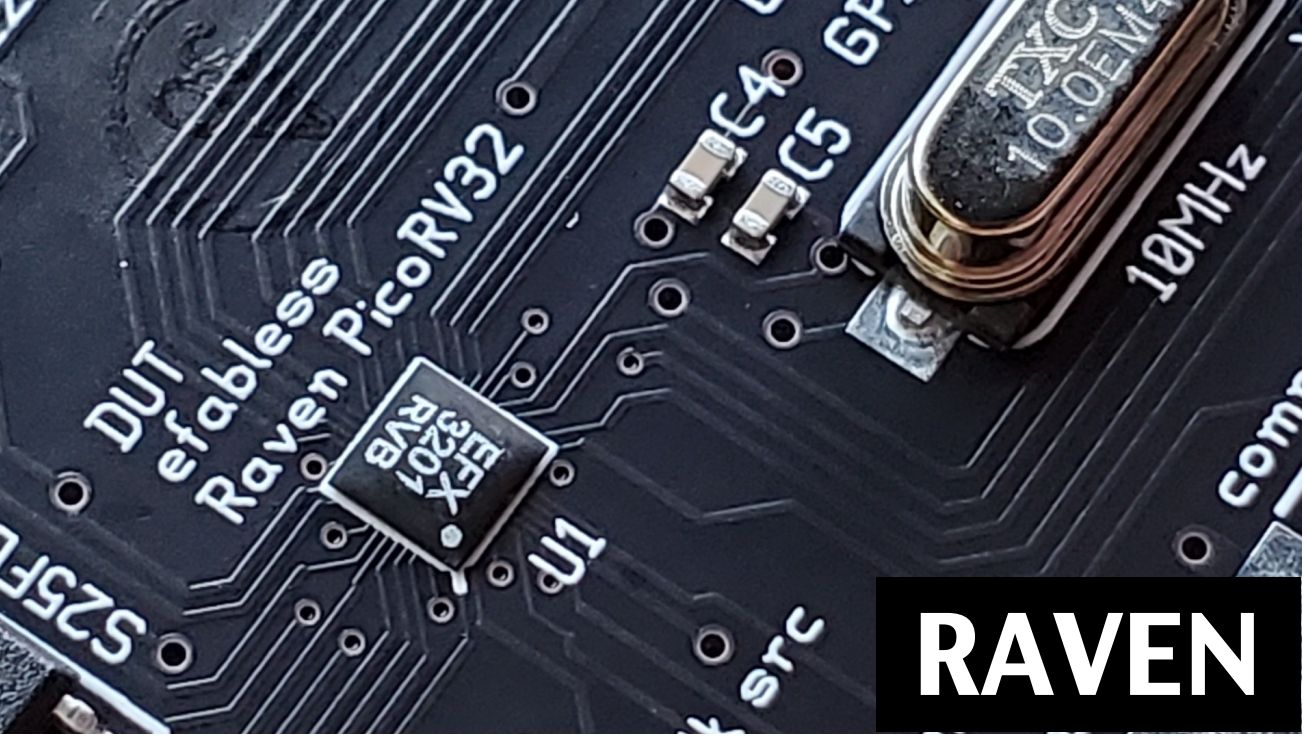
- Demo:
  - Standard key exchange Diffie–Hellm
  - Compute public and private keys
  - Ravenna chips



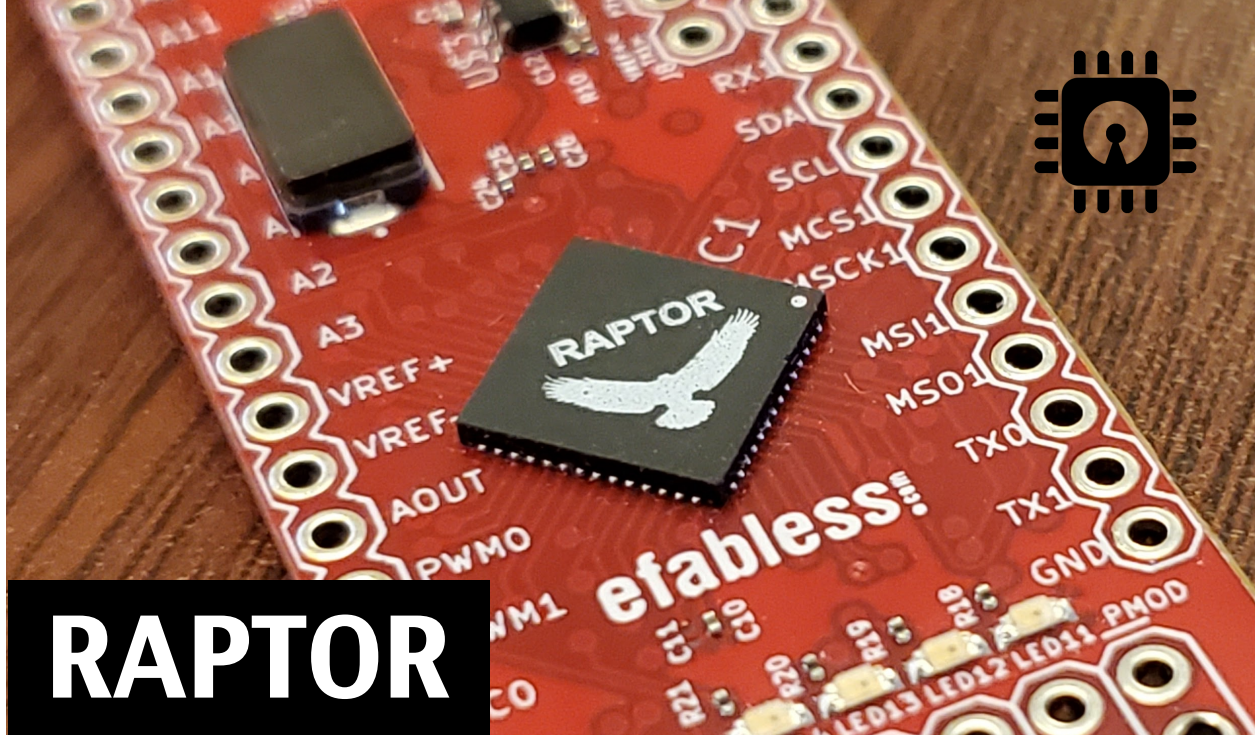


- Best performance
  - Ravenna : 96 bits, 75 seconds
  - Newer Ravenna: 128 bits, 60 seconds

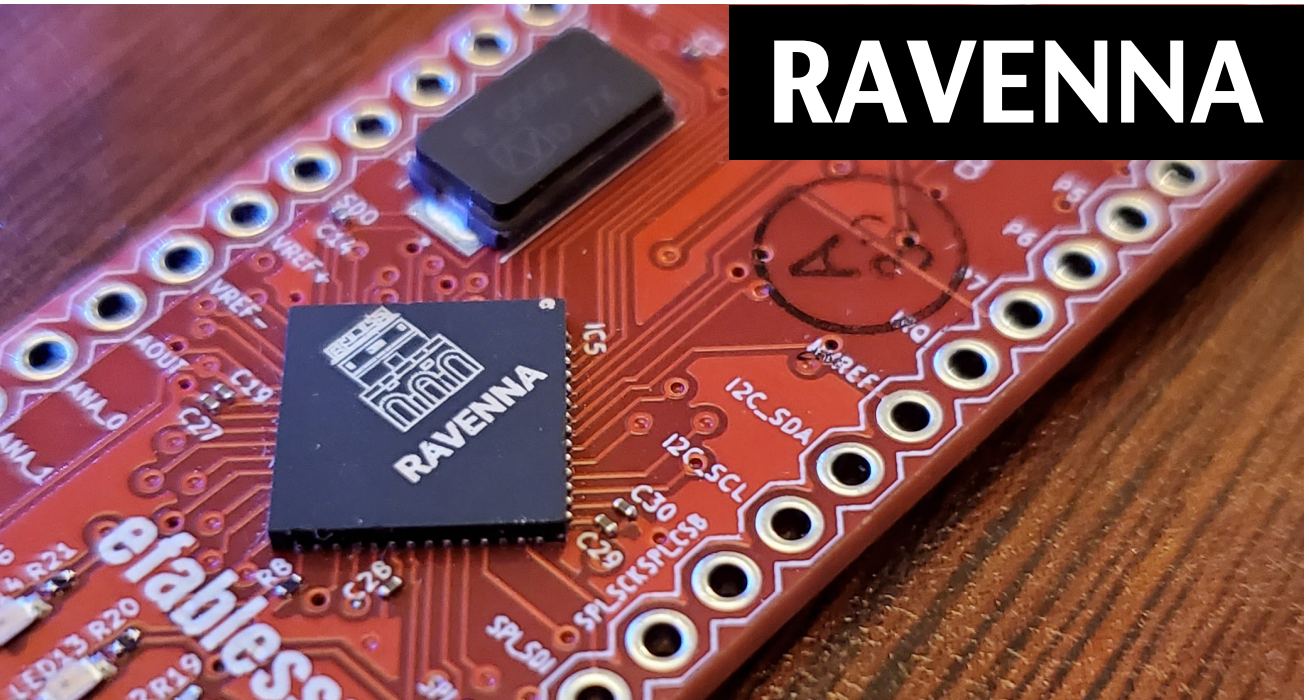
Line of codes	Max RAM
~400	<100 Bytes



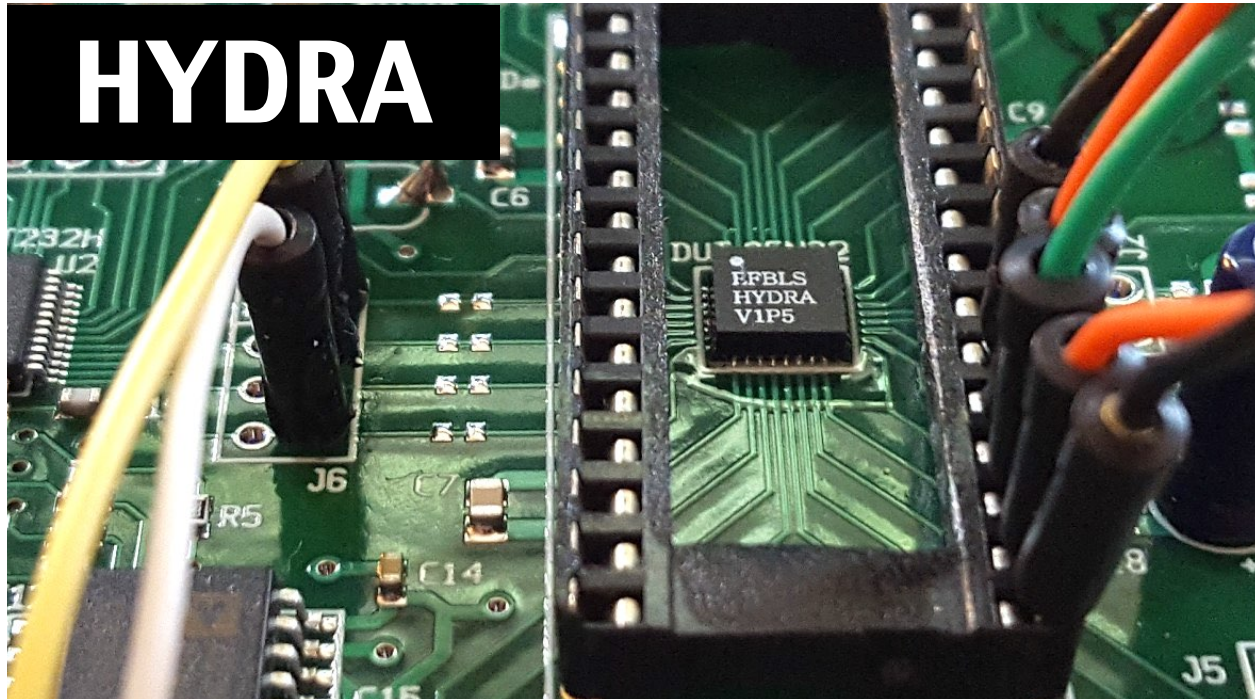
RAVEN



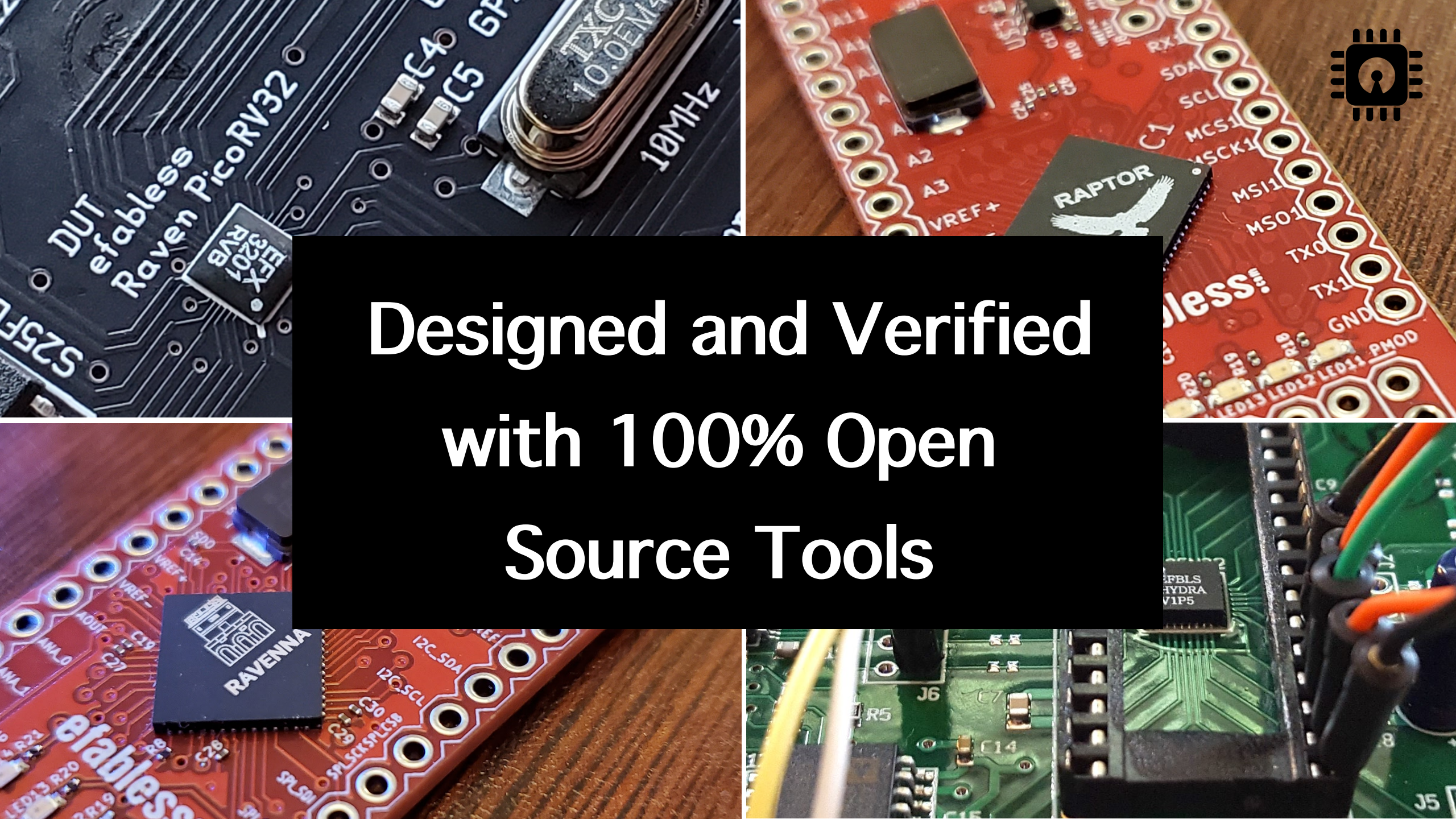
RAPTOR



RAVENNA



HYDRA



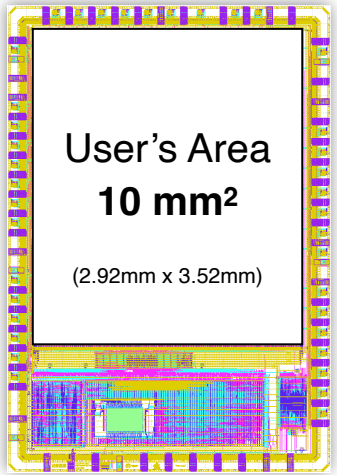
Designed and Verified with 100% Open Source Tools

How do we simplify chip design? **#3**

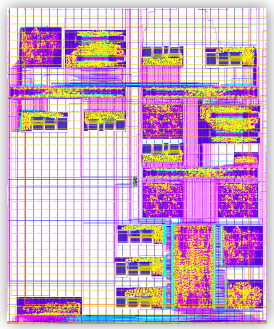
Start from the 15th floor to reach 20

*Build on existing foundational work by others - **CARAVEL***

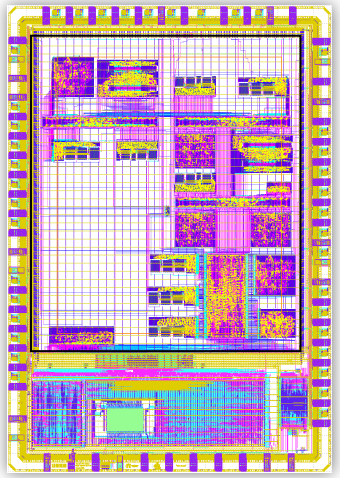
*You only need to know your design … or **your code***



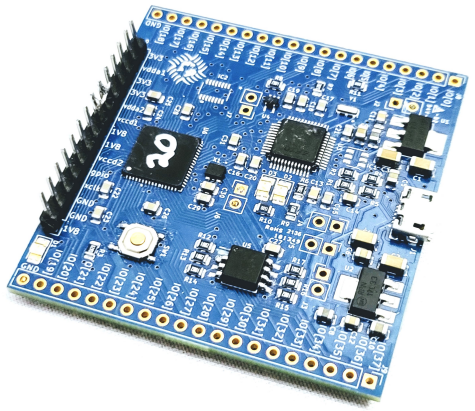
CARAVEL



My Design



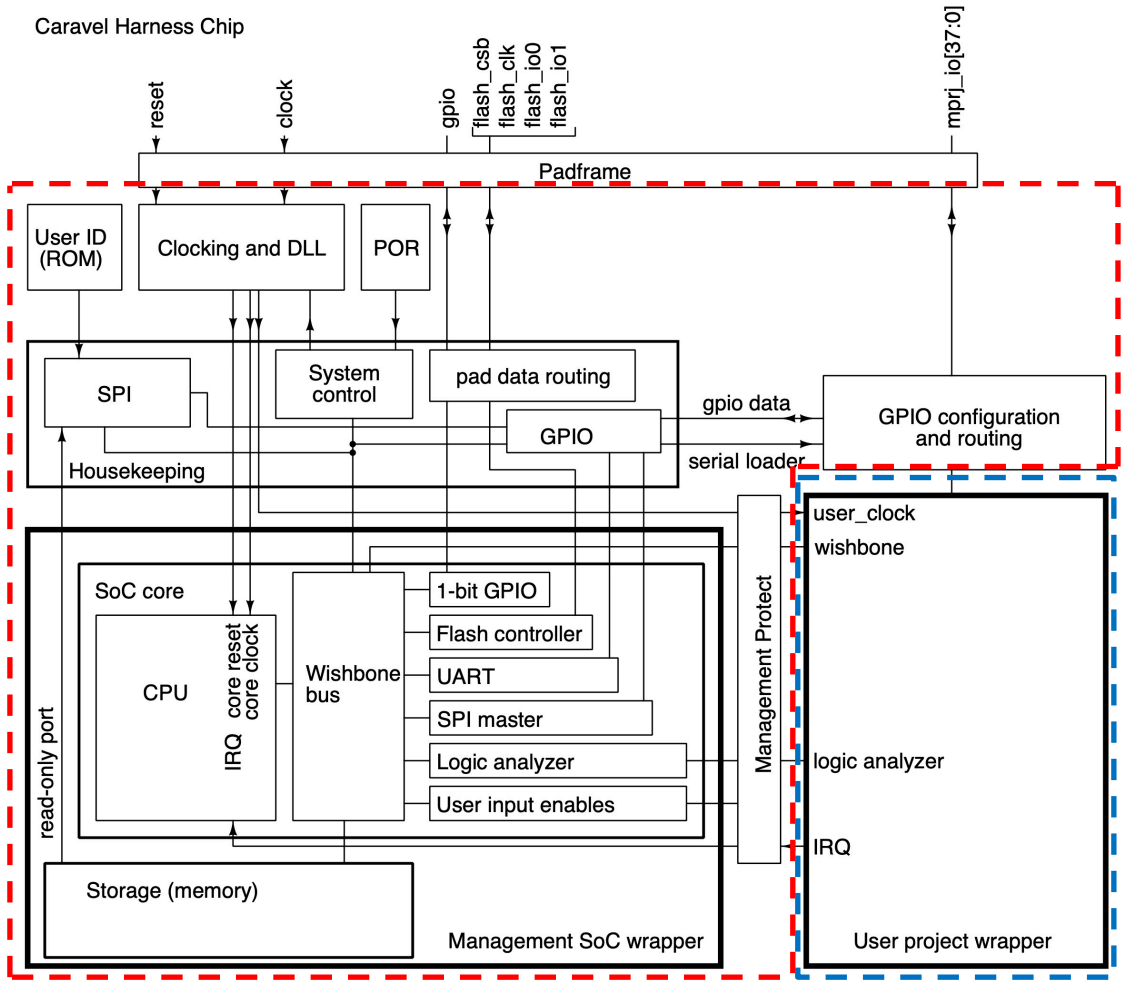
CARAVEL + My Design



Plug & Play Dev Board

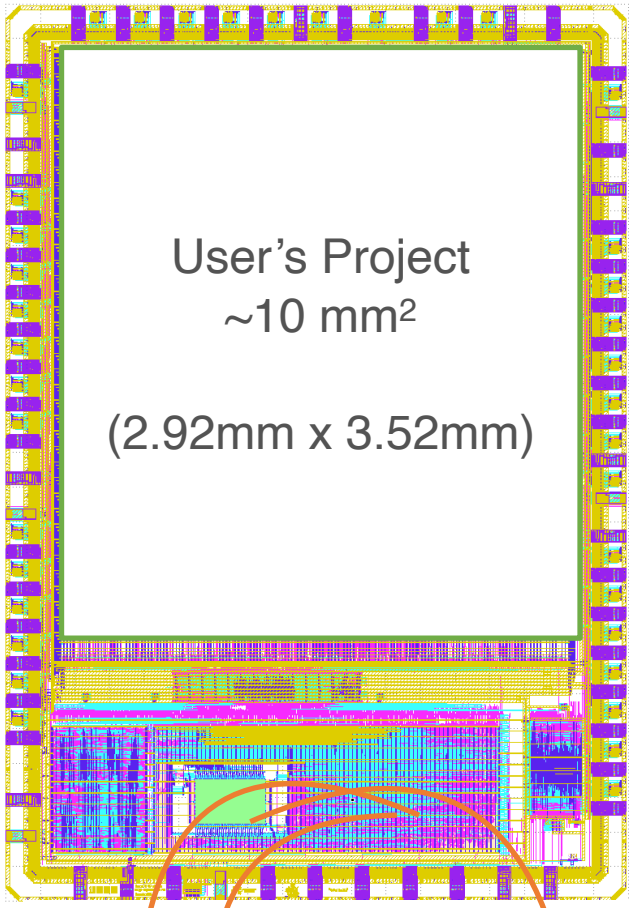
efabless.com

CARAVEL PLATFORM



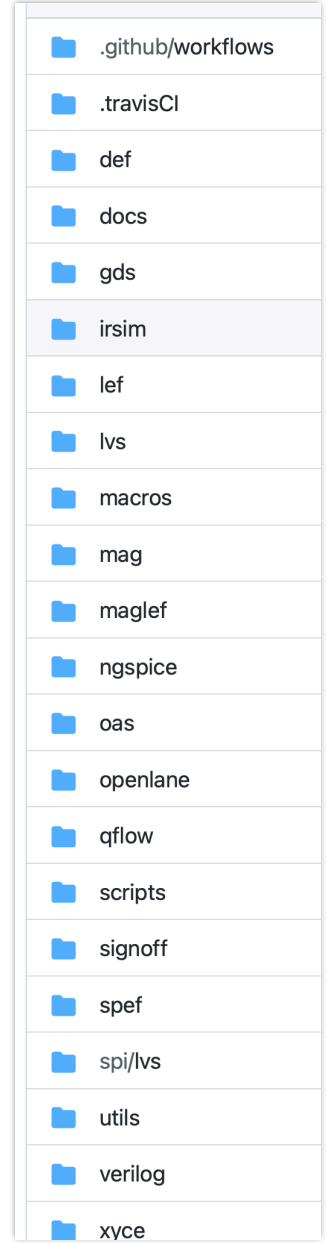
https://github.com/efabless/caravel

CARAVEL SoC Platform



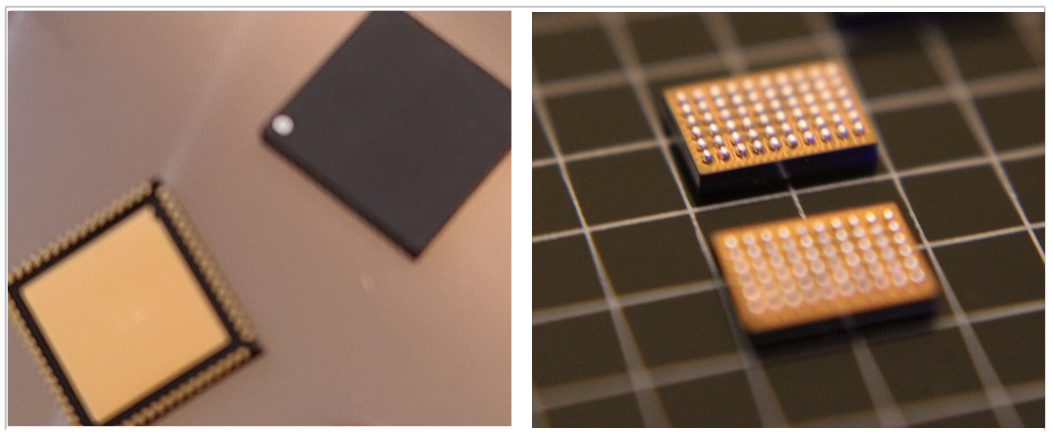
PicoRV32 OpenRAM
OpenROAD OpenLane

Uniform Open Source File Structure to ease reuse & modifications



efabless.com

SILICON TESTING & VALIDATION



Actual Chip and Board Images

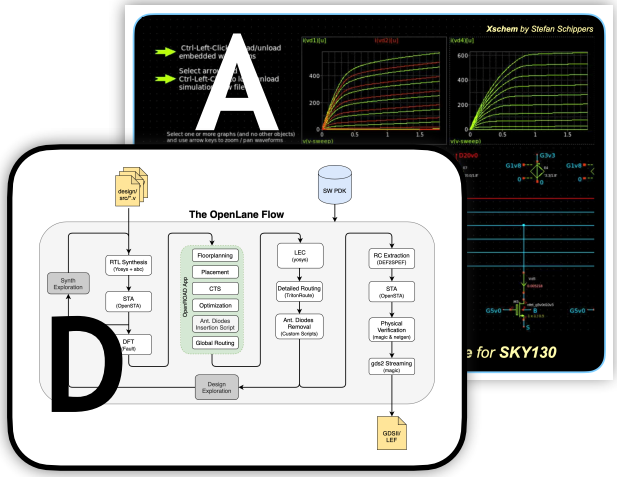
Designers receive **packaged chips** and **assembled 5 evaluation boards** with for each project

On-chip **open source** test framework with firmware to support the following:

- On-chip logic Analyzer
- Drivers for common peripherals
- Flash memory programming software utility
- Example firmware routines for common functions
- Instructions for customizing firmware for each project

efabless.com

OS EDA Digital & Analog

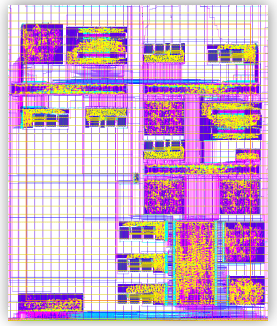


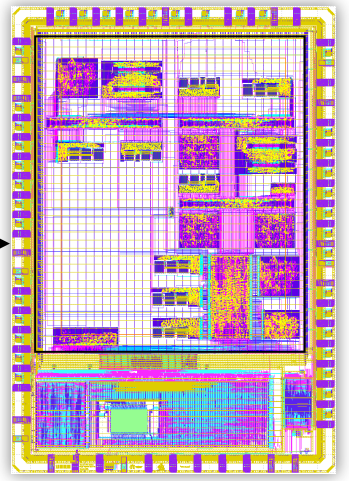
+

CARAVEL

+

User Design





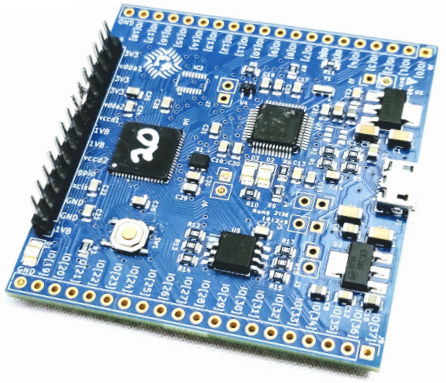
CARAVEL + User Design

MANUFACTURING

Start Here
https://ef.link/start-digital
https://ef.link/start-analog

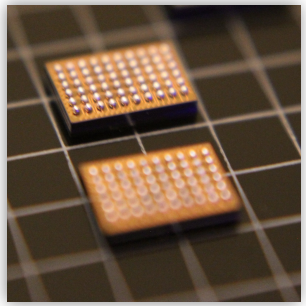
Software (FOSS)

+



5 Dev Boards

+



300 WCSP Parts

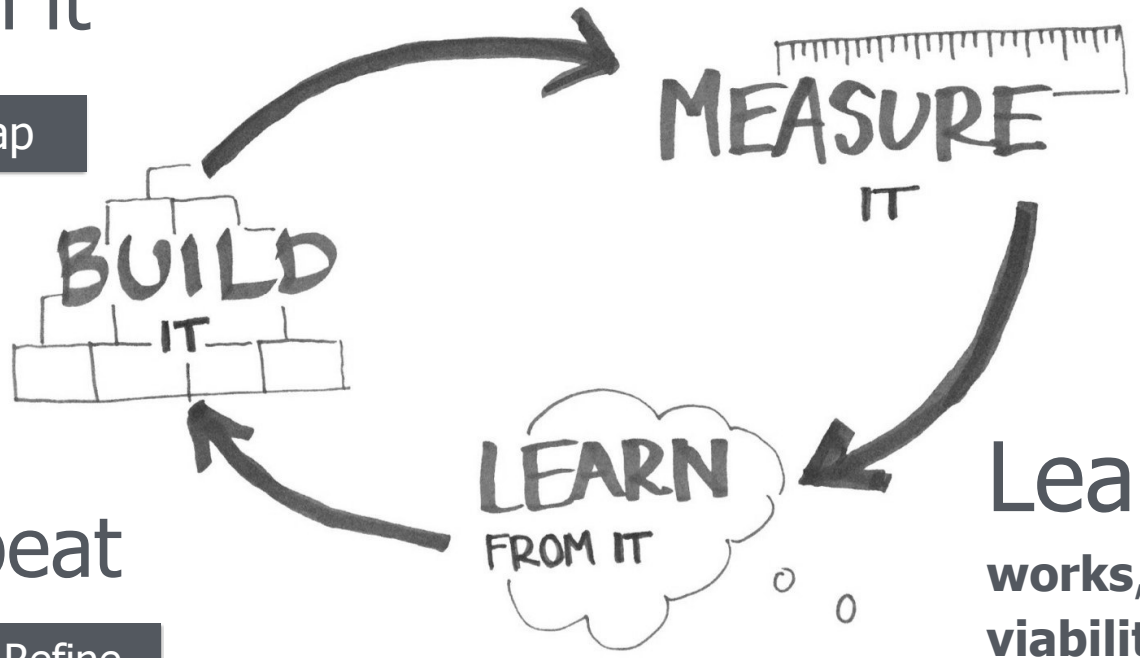
efabless.com

How do we simplify chip design? **#4**

Enable —Fail fast, learn and repeat

1 Design it

Quick & Cheap



Test It **2**

Learn from it, What **works**, What **doesn't**
viability and feasibility **3**

4 Repeat

Correct & Refine

efabless.com

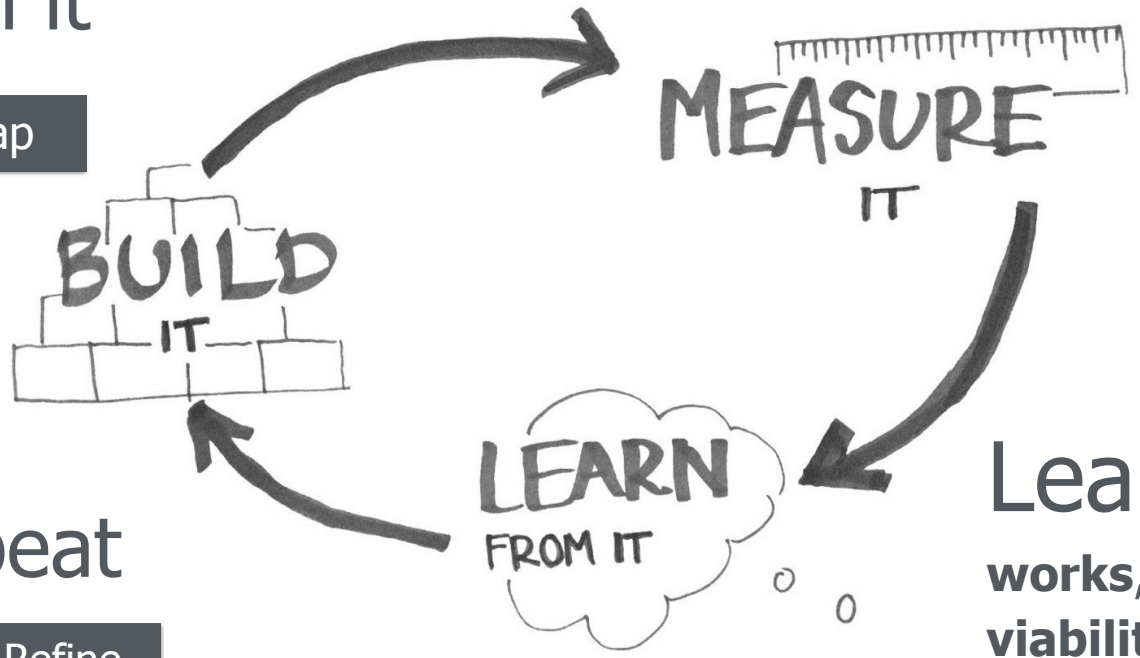
OK - sounds good but …

hardware iterations cost $$

Enable —Fail fast, learn and repeat

1 Design it

Quick & Cheap



Test It **2**

Learn from it, What **works**, What **doesn't**
viability and feasibility **3**

4 Repeat

Correct & Refine

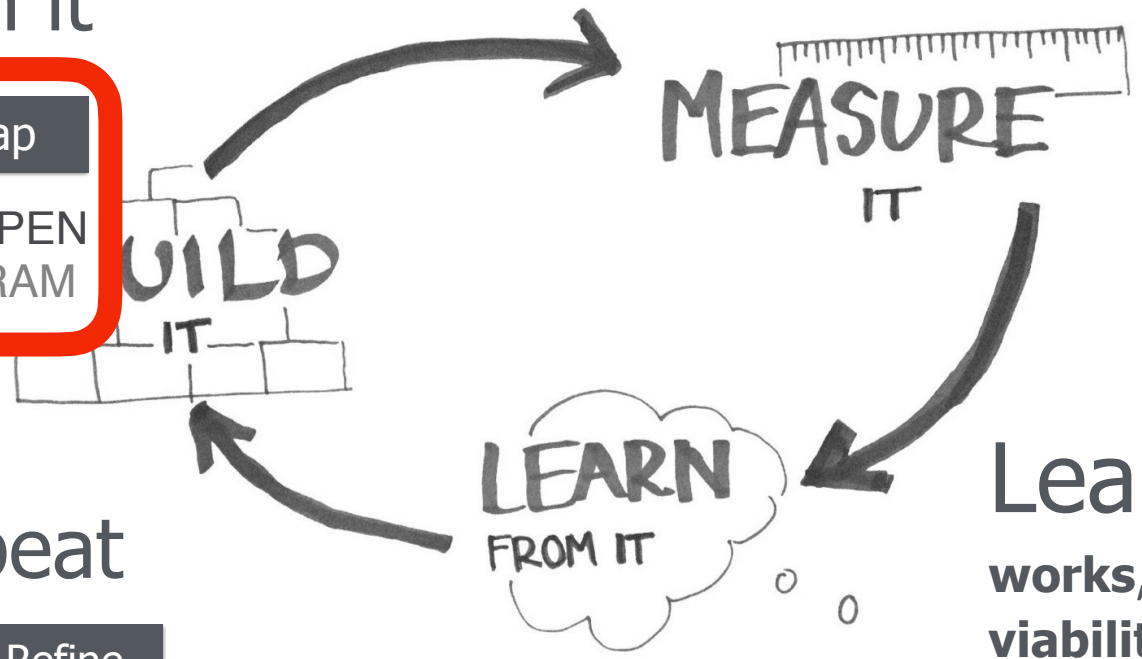
efabless.com

Enable —Fail fast, learn and repeat

1 Design it

Quick & Cheap

GOOGLE's OPEN MPW PROGRAM



Test It 2

Learn from it, What **works**, What **doesn't viability** and feasibility 3

4 Repeat

Correct & Refine

efabless.com

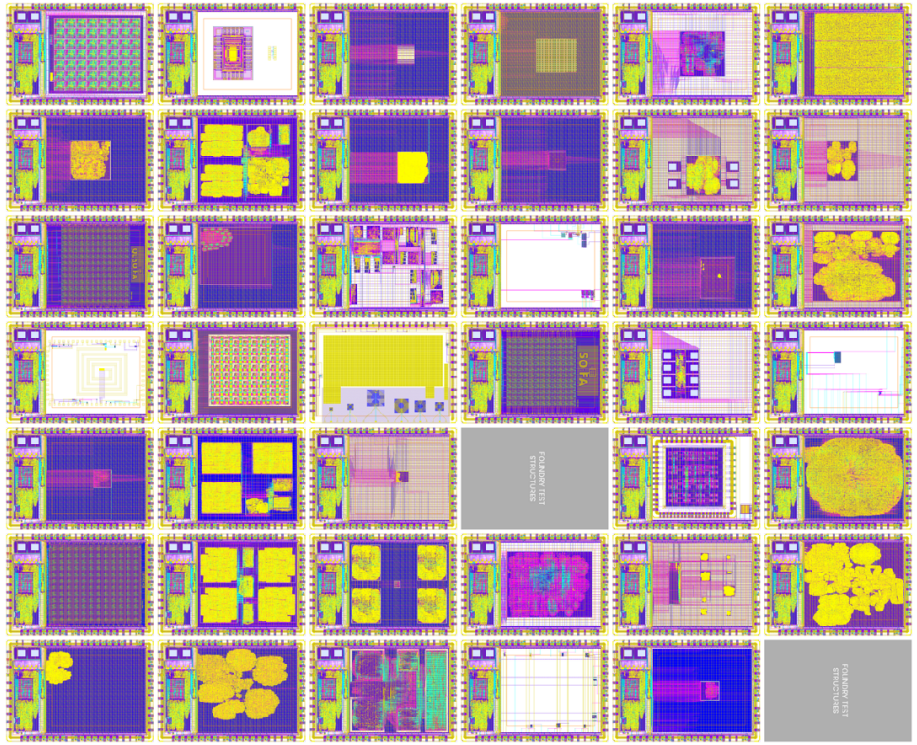
Start here https://ef.link/gmpw

FREE SILICON SPONSORED BY GOOGLE

- **Google** is funding **4** manufacturing runs in **2022**
- *A minimum of **40** designs each run*
- Participants use a Harness SoC (Caravel) - ***10mm2 open area***
- Use SKY130 process & OS *Digital & Analog Design Tools*
- Participants get ***5 dev boards + 300 WCSP-packaged parts***
- All designs must be public and under an ***open source license***
- All designs must contain information & files to ***reproduce the work***
- You are strongly encouraged to try new ideas, ***take risks and iterate***
- ***All skill and experience levels*** are welcome to participate

Join the community at https://ef.link/chat

The first shuttle was overbooked: 45 designs submitted in 30 days!



Open MPW Shuttle Program

efabless.com

Sponsored by Google

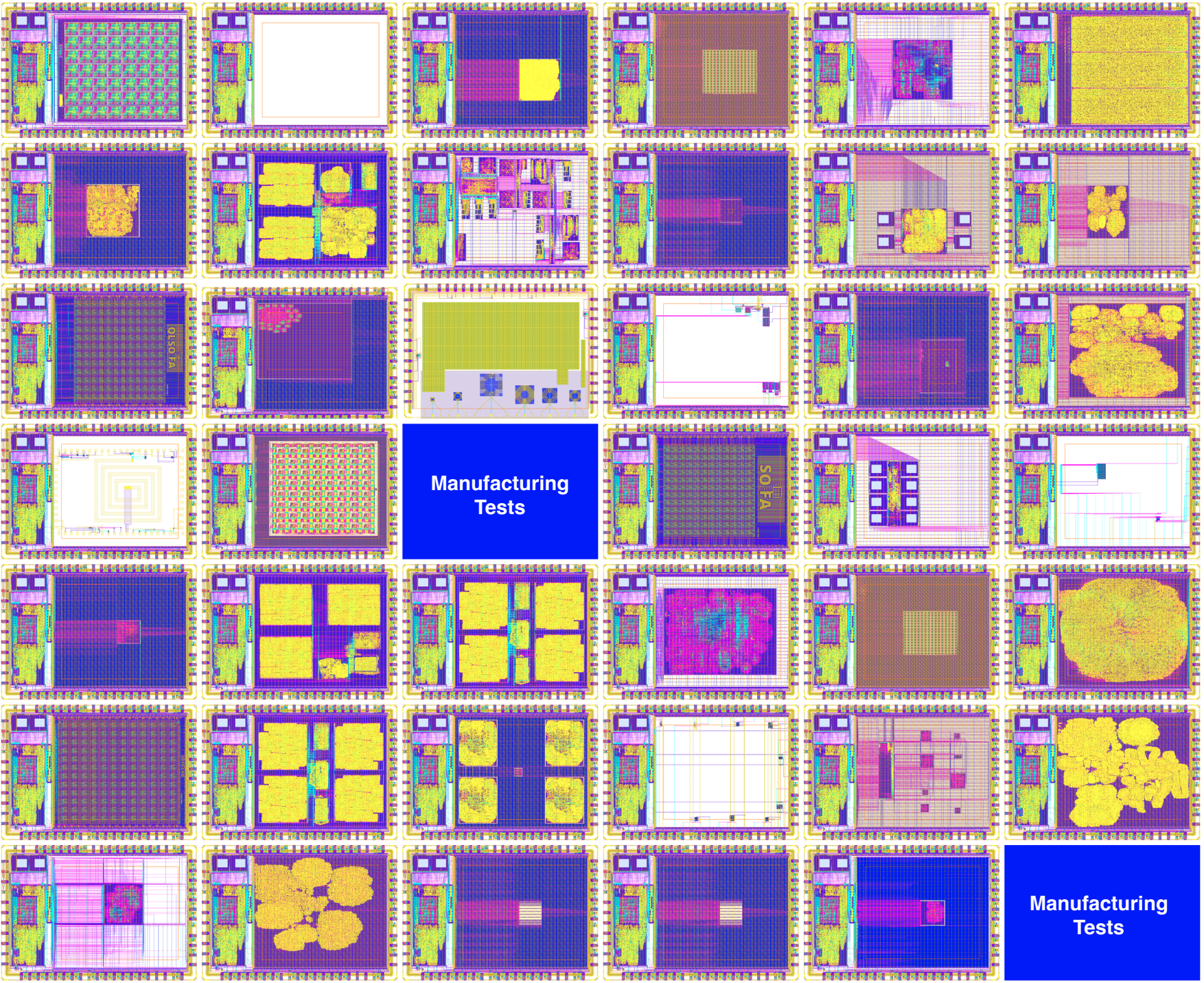
DESIGN TYPES

MPW ONE

- 9 x Open processor cores
- 9 x SoC's
- Crypto-currency Miner
- Robotic App Processor
- Amateur Satellite Radio Transceiver
- 7 x Analog/RF
- 5 eFPGA's

See Projects Here: https://efabless.com/projects/shuttle_name/MPW-1

efabless.com



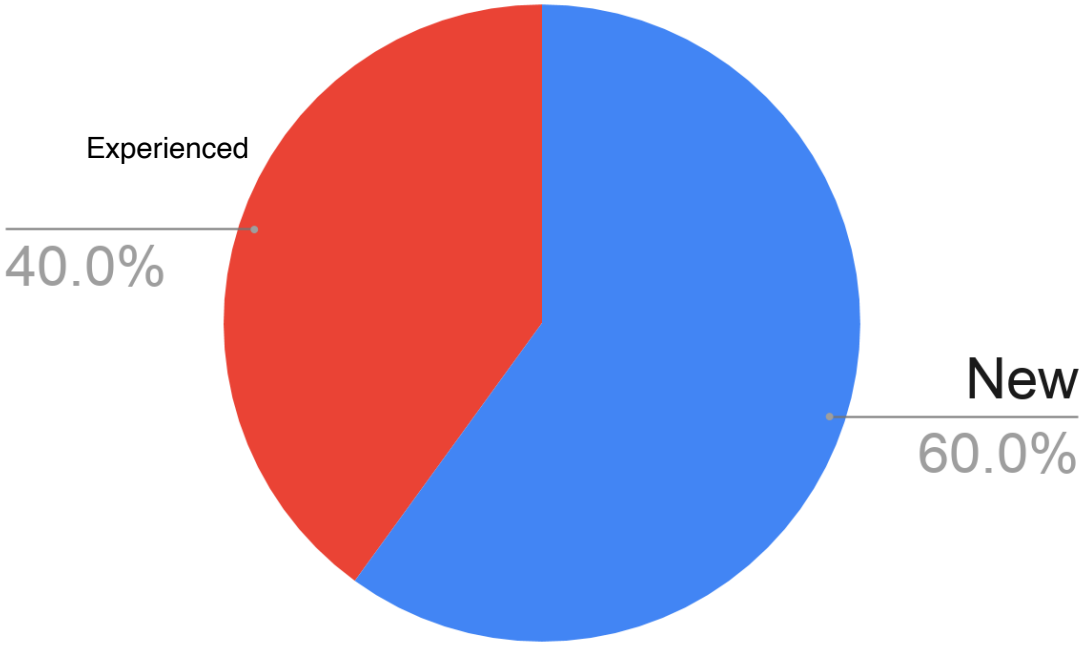
DESIGN TYPES

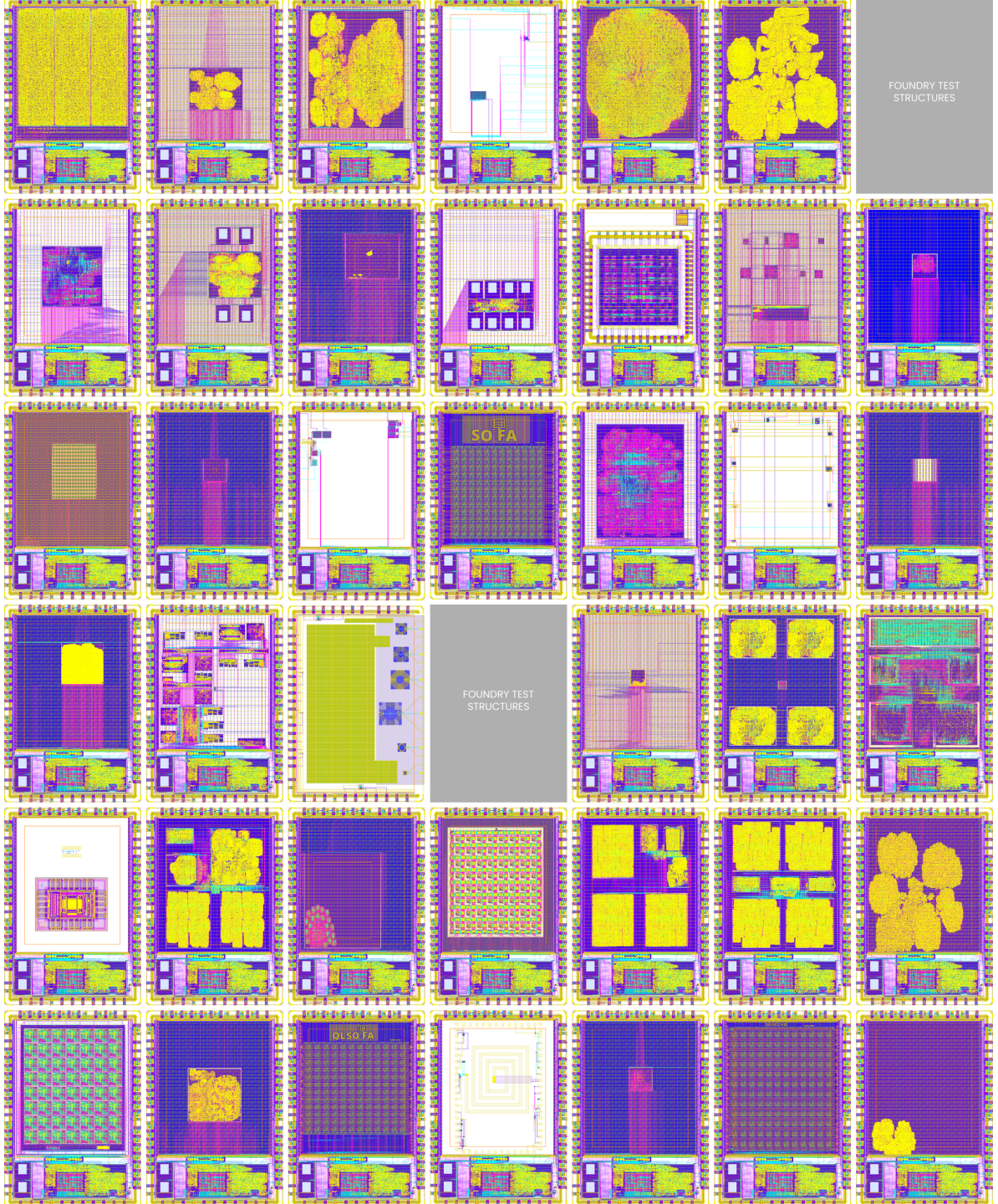
- 9 x Open processor cores
- 9 x SoC's
- Crypto-currency Miner
- Robotic App Processor
- Amateur Satellite Radio Transceiver
- 7 x Analog/RF
- 5 eFPGA's

COMPANIES

- IBM: OpenPOWER - MicroWatt
- QuickLogic - eFPGA
- Antmicro
- Western Digital - Swerv-EL2
- EFabless
- SpinMemory

60% by first time designers!



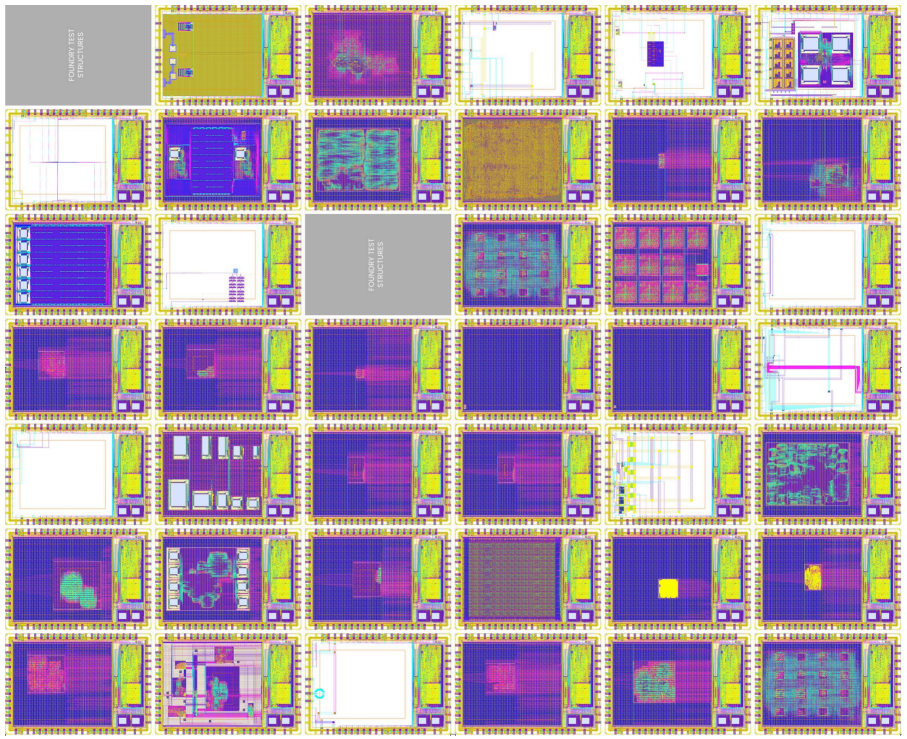


And so was the second:
56 designs submitted
in 30 days!

Open MPW Shuttle Program

efabless.com

Sponsored by Google



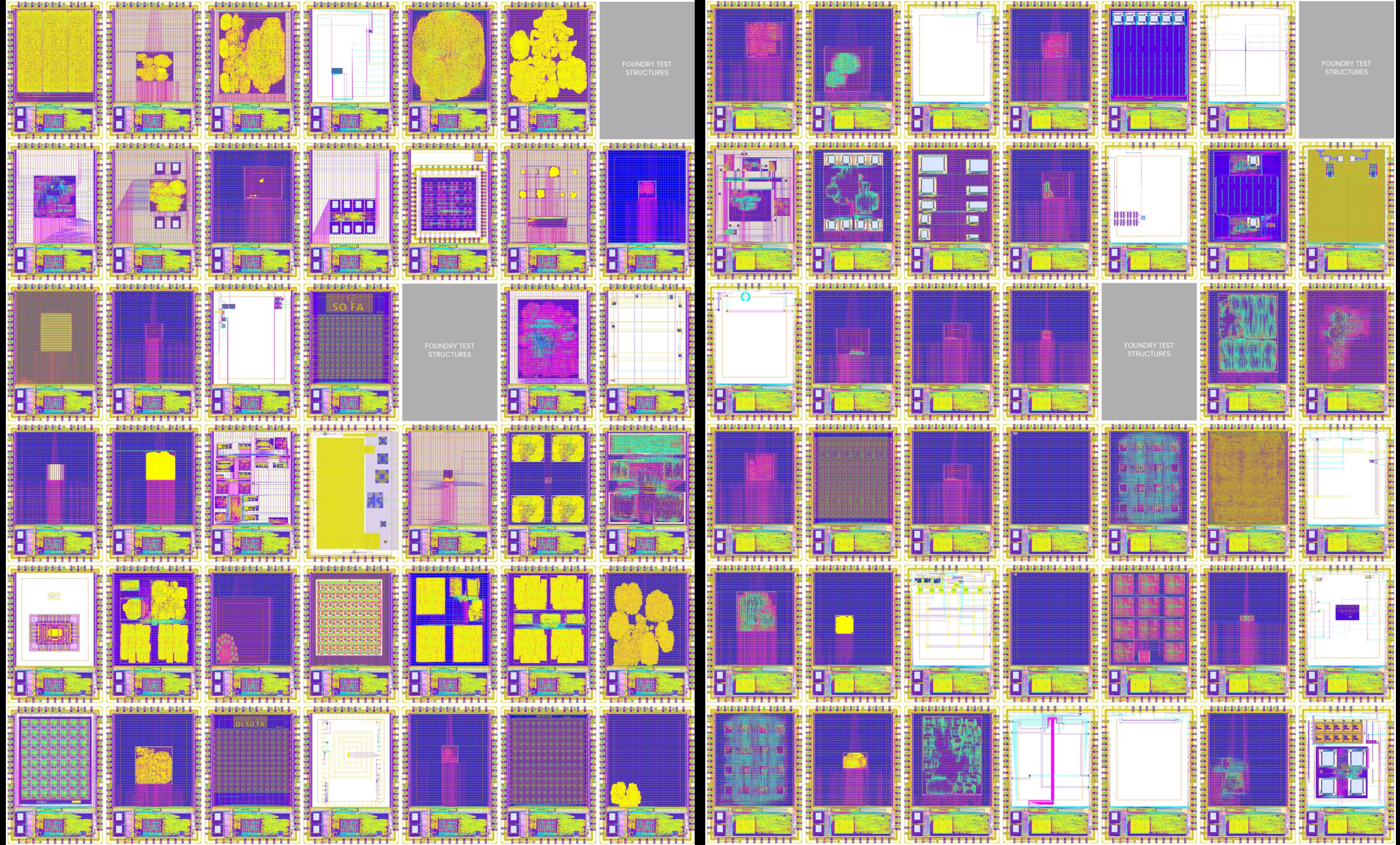
DESIGN TYPES

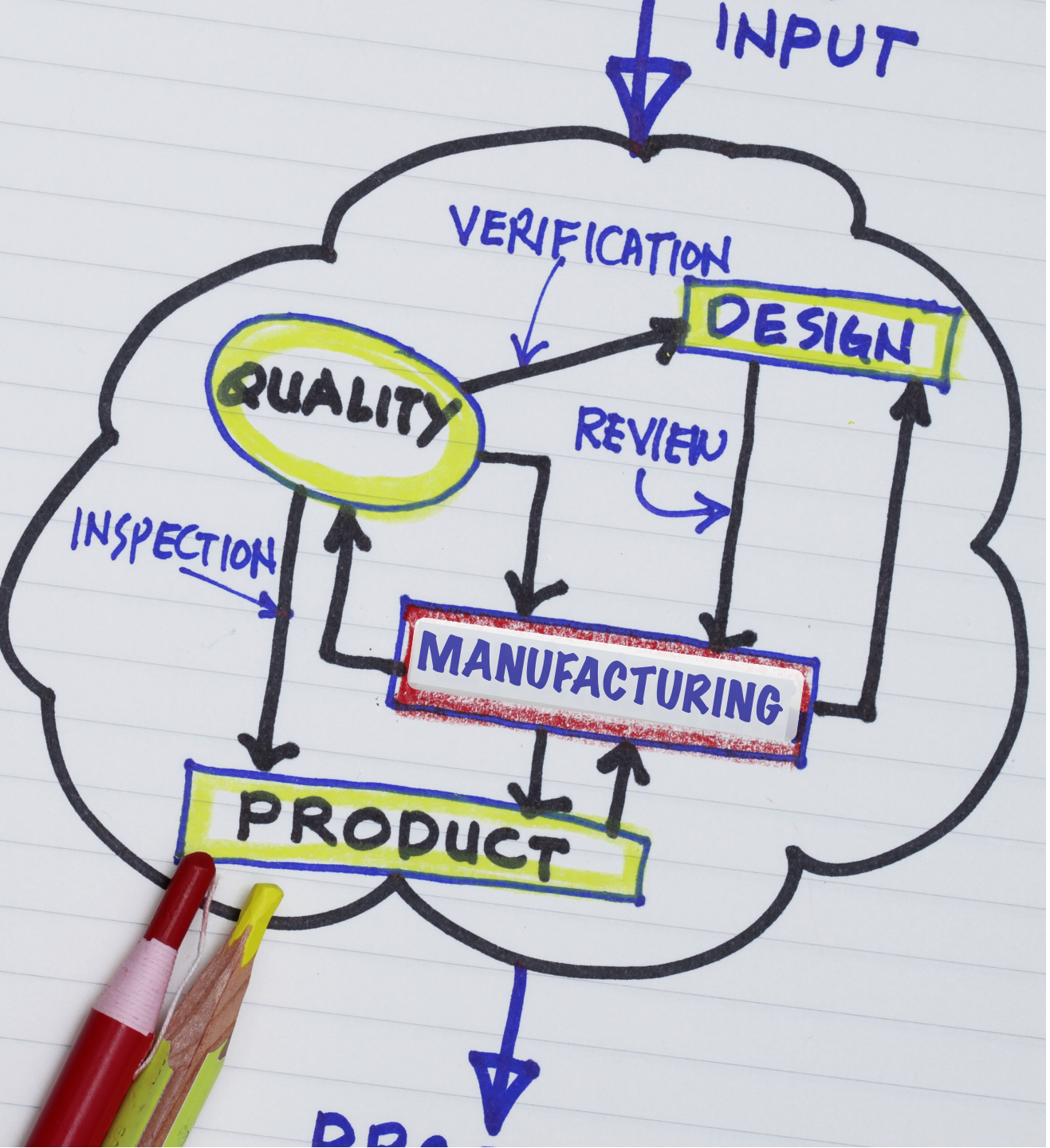
MPW TWO

- 11 x Open processor cores
- 11 x SoC's
- Crypto-router
- T2D Converter - LIDAR
- Multi-project harness for Caravel x 16
- 17 x Analog/RF
- 2 eFPGA's

See Projects Here: https://efabless.com/projects/shuttle_name/MPW-2

efabless.com


MPW ONE
FOUNDRY TEST STRUCTURES
FOUNDRY TEST STRUCTURES
SO FA
FOUNDRY TEST STRUCTURES
QLSO FA
FOUNDRY TEST STRUCTURES
FOUNDRY TEST STRUCTURES
MPW TWO

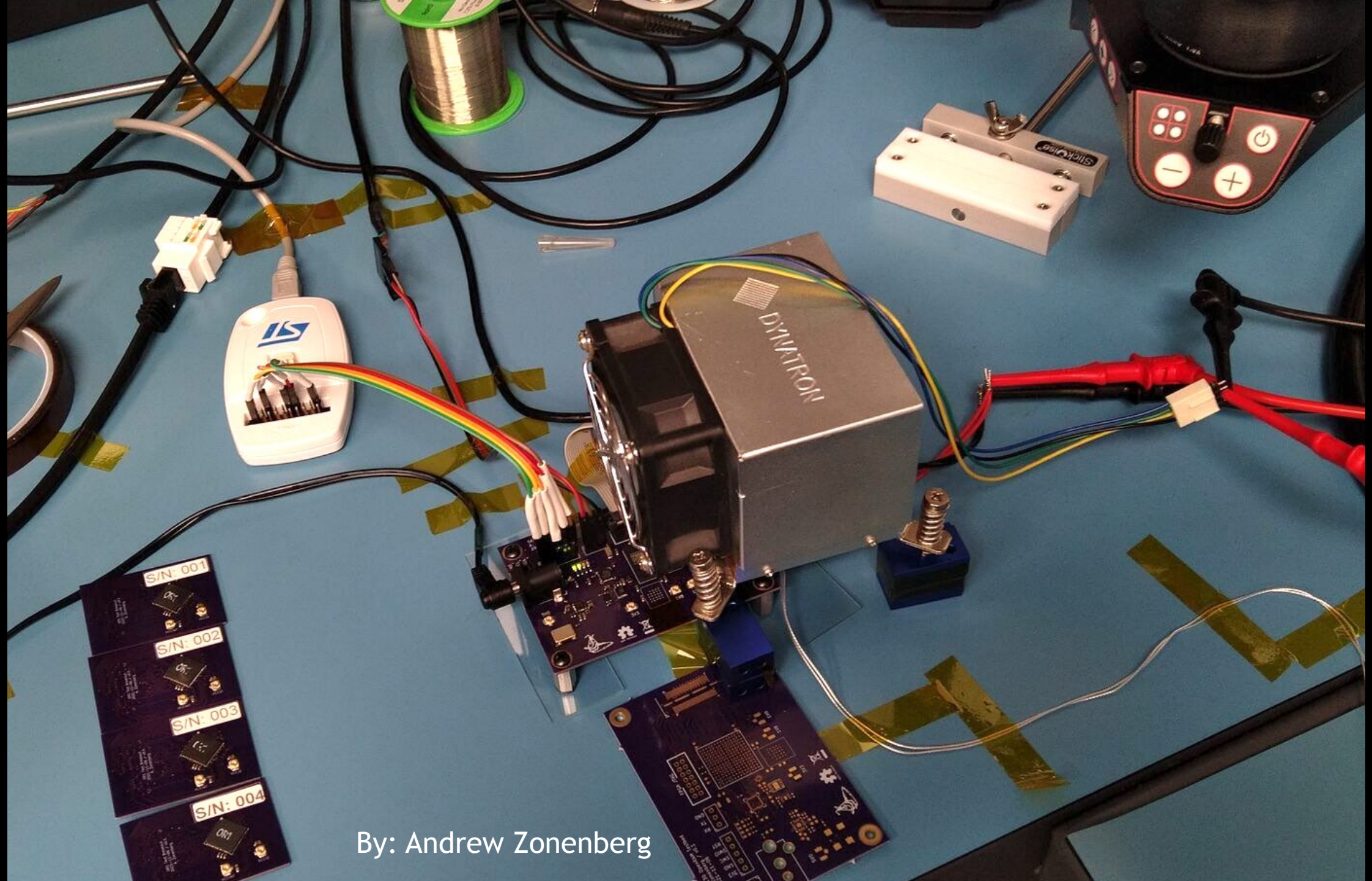

INPUT
VERIFICATION
DESIGN
QUALITY
REVIEW
INSPECTION
MANUFACTURING
PRODUCT


The Community
At Work

SRAM Characterization

By: Andrew Zonenberg

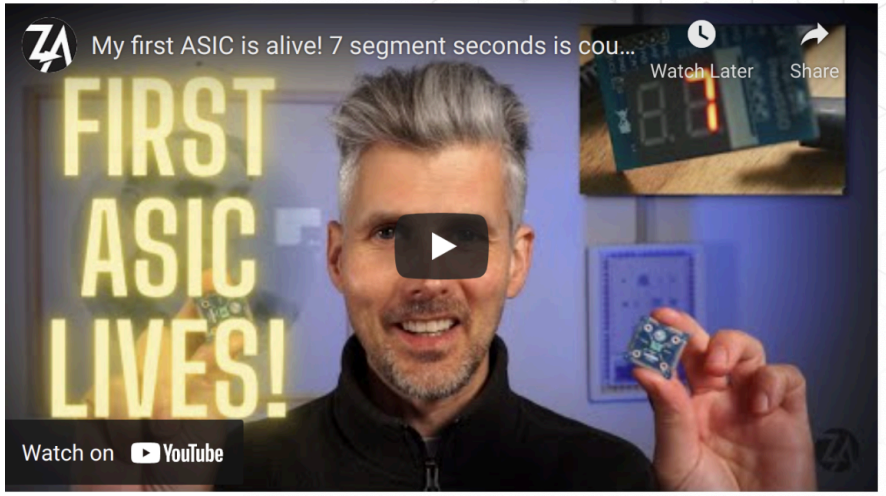
SRAM Characterization



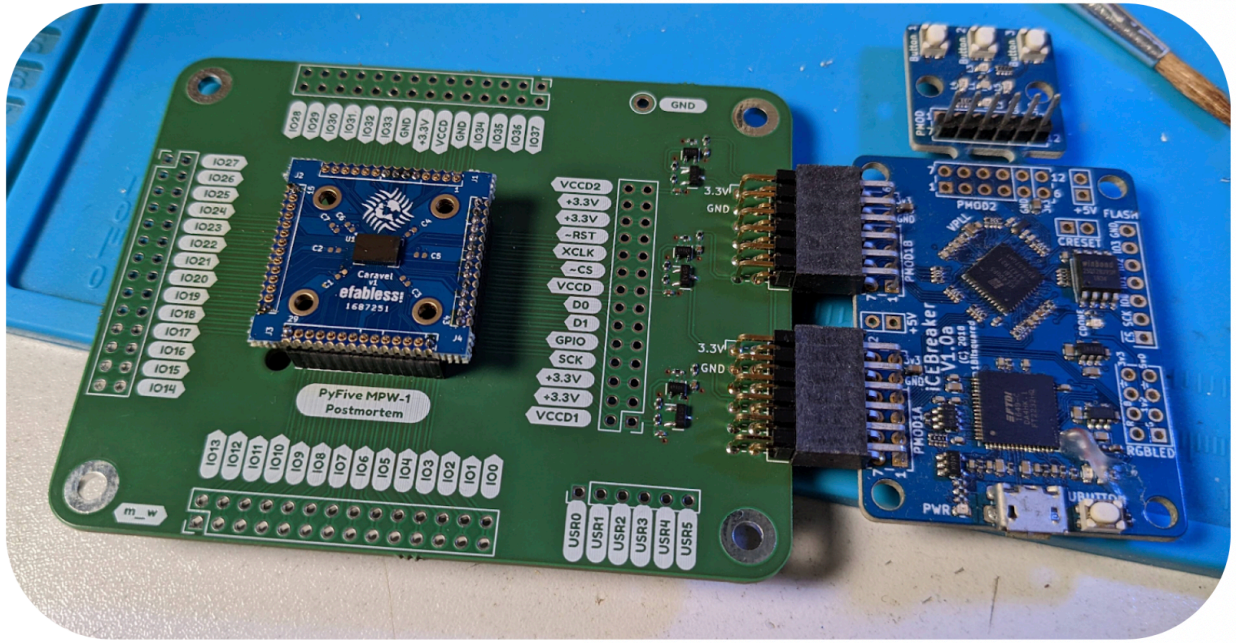
By: Andrew Zonenberg

Temperature Control

- www.zerotoasiccourse.com/post/mpw1-bringup/
- github.com/mattvenn/mpw1-bringup



youtu.be/f_G5ad8SbHo



bit.ly/cicc22-edu-goog

Slide provided by: Tim Ansell

efabless.com

All my ASIC designs for Google MPW1 are working!

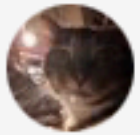
Zero To ASIC Course

1.1K views • 1 month ago

youtu.be/IdOvywOSSmI

bit.ly/cicc22-edu-goog

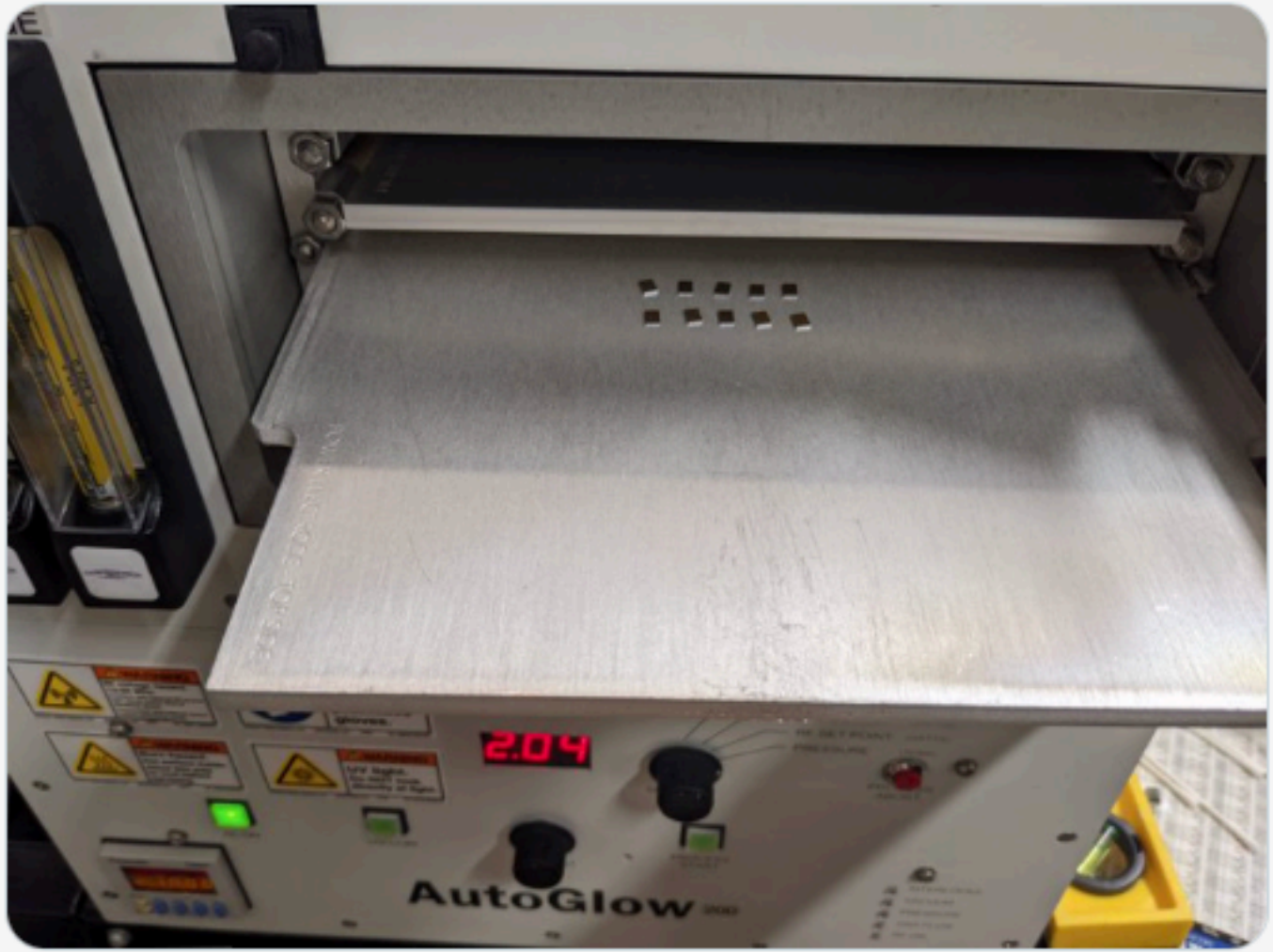
Slide provided by: Tim Ansell



Volodymyr Pikhur @vpikhur · 15h

Replying to @vpikhur

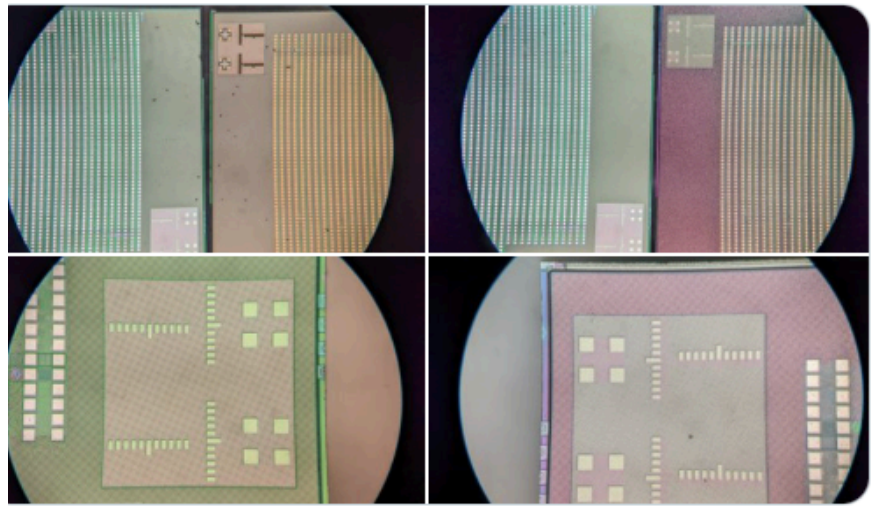
Loaded 10 samples into RIE (reactive ion etcher) to strip top polyimide layer. I use a 50/50 mix of oxygen and argon gas.



1 1 4

olodymyr Pikhur @vpikhur · 15h

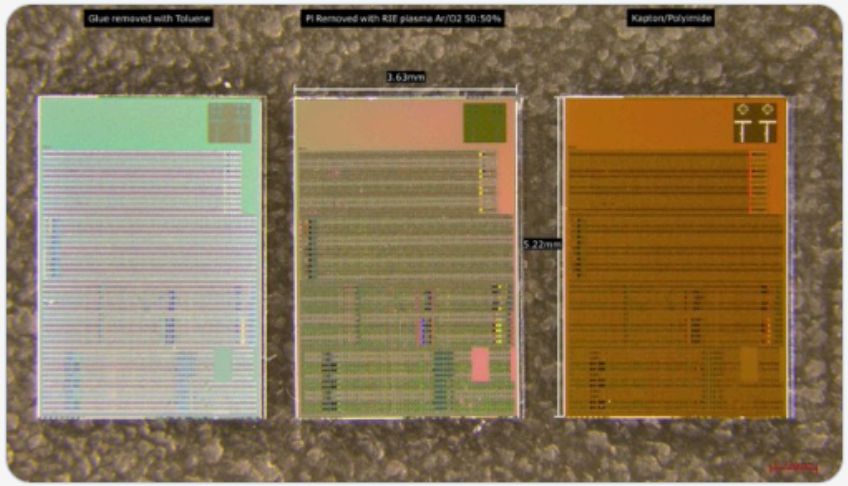
ter polyimide is removed there is adhesive/glue layer remaining. Adhesive n't be removed with normal solvents such as acetone or isopropyl cohol. Very non-polar solvent such as Toluene does the job.



1 4 8

Volodymyr Pikhur @vpikhur · 15h

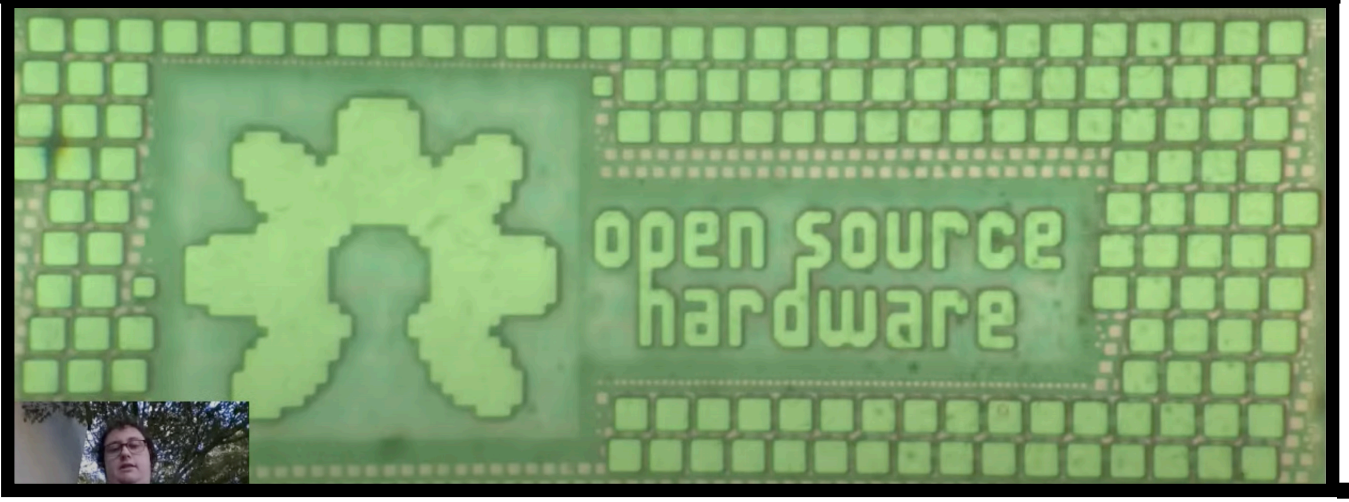
All 3 under stereo microscope with approximate die dimensions.



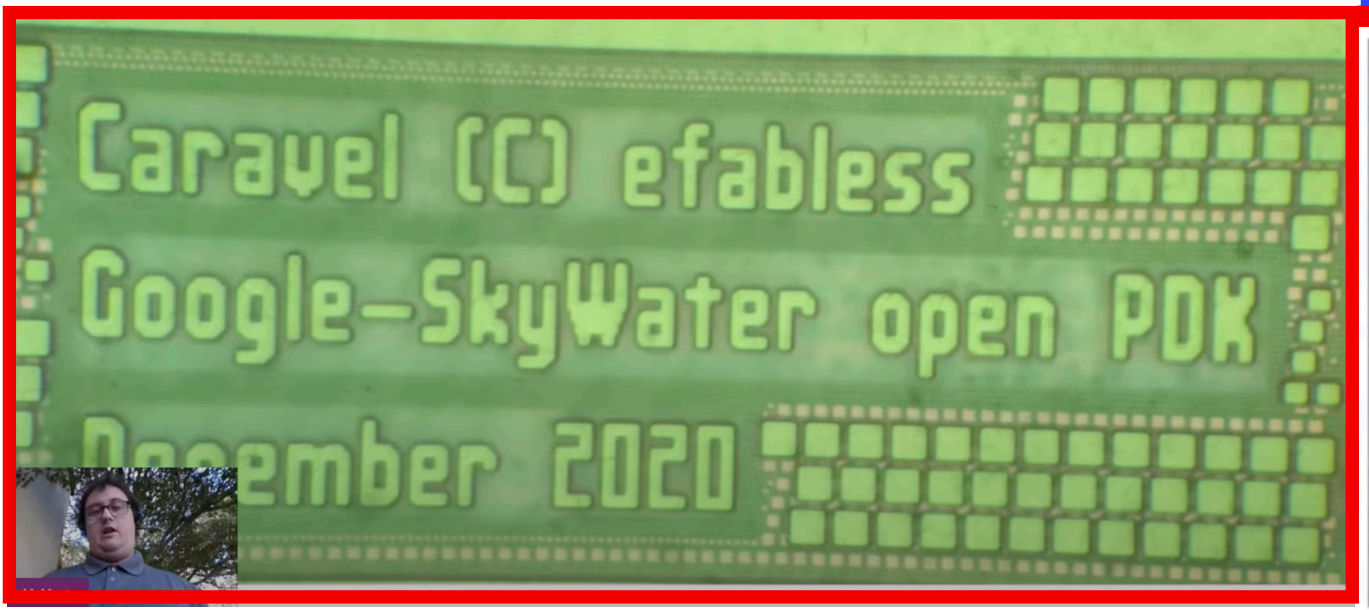
1 1 6

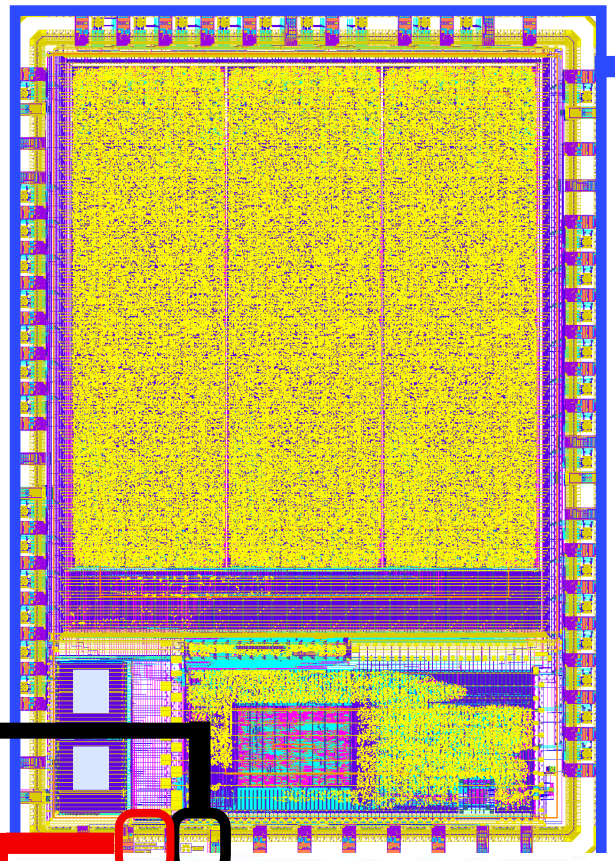
History in The Making

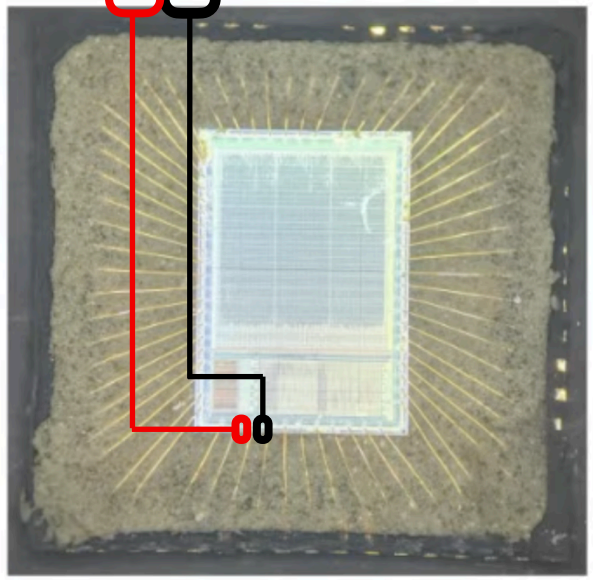
Open Source HW Logo Carved in the IO Ring of the First 100% Open Source SoC

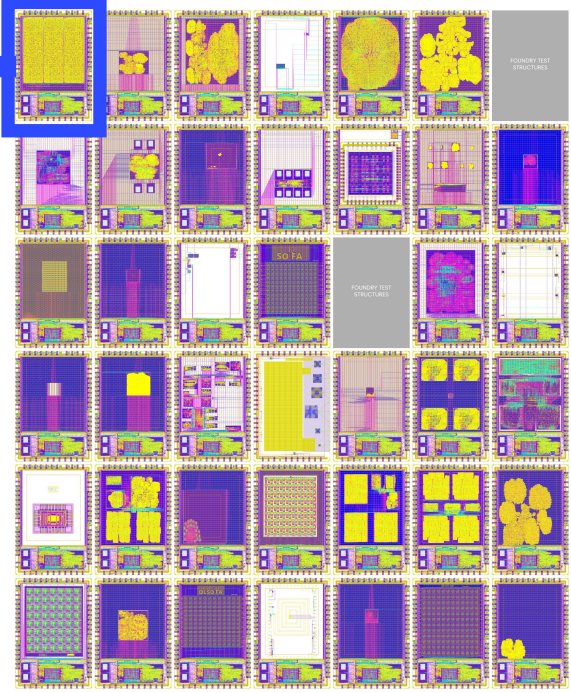


Images by John McMaster - presented at OpenTapeOut.dev









MPW-ONE

Google

skywater
TECHNOLOGY FOUNDRY

efabless.com

OpenMPW Project Submissions

Always Overbooked

45	56	52	55	77
MPW-1	MPW-2	MPW-3	MPW-4	MPW-5

https://efabless.com/projects

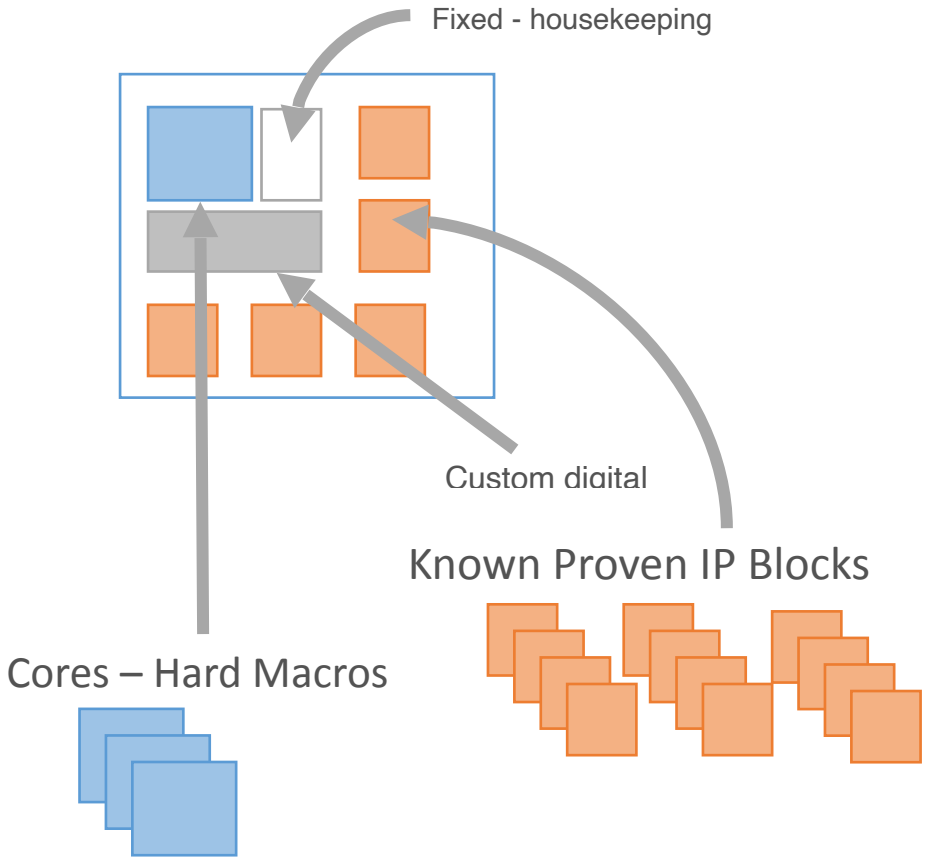
efabless.com

How do we simplify chip design? **#5**

Build an open library of “blocks”

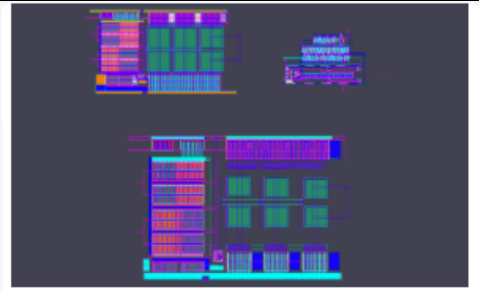
Make them like LEGO

- Silicon proven / verified functions
- Highly leveraged known proven IP blocks
- Scope of customization is constrained
- Quality processes enable extending the known proven IP blocks by community



efabless.com

Select Open Source Designs

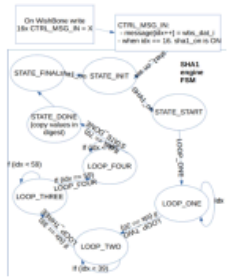


caravel_fulgor_opamp public

Diego Hernando | http://www.fundacionfulgor.org.ar/sitio/index.php

Operational amplifier (opamp) based on the Miller topology designed in Skywater SKY130 CMOS process.

MPW-1 SKY130 2.2k

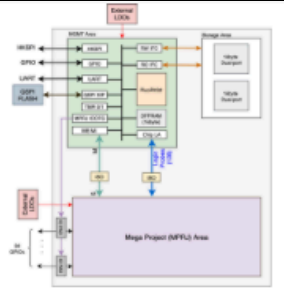


SHA1 engine public

Konrad Rzeszutek Wilk

The SHA1 engine, while not the most secure nowadays is still used by git commits and TPM PCR...

MPW-2 SKY130A 731

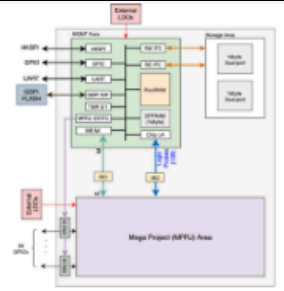


Caravel-SOFA-HD public

Xifan Tang | https://sites.google.com/site/pegaillardon/home

SOFA-HD (Skywater Opensource FPGAs)

MPW-1 SKY130 1.6k

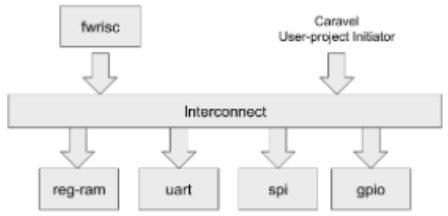


Caravel public

Sylvain Munaut | https://github.com/PyFive-RISC-V

Peripherals tests for future SoC targeting Micro/Circuit Python

MPW-1 SKY130 1.8k

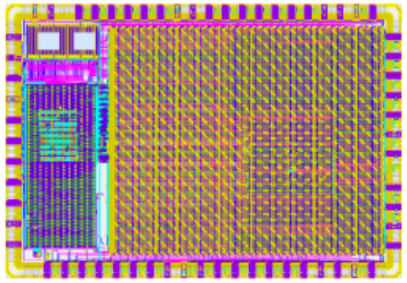


FWPayload public

Matthew Ballance | http://github.com/mballance

A simple RISC-V core+peripherals subsystem for the Google-sponsored Open MPW shuttles for SKY130.

MPW-1 SKY130 962

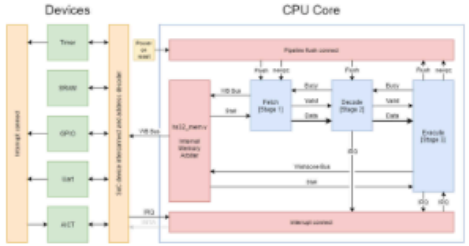


Caravel_Astria_Testchip public

Astria Nur Irfansyah | http://www.its.ac.id

Test circuits consisting of synthesizable comparators for a stochastic ADC, to be submitted for...

MPW-1 SKY130 1.2k

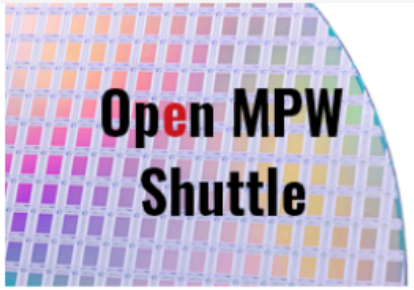


HS32Core public

Kevin Mack Baragona | https://github.com/hsc-latte

Open Source Hardware Processor

MPW-1 SKY130 1.4k



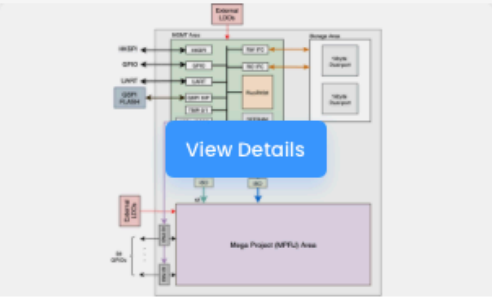
10_bit_potentiometric_DAC public

Sameer S Durgoji | https://www.vlsisystemdesign.com/

Design of a 10 Bit Potentiometric Digital to Analog Converter with 3.3V analog voltage, 1.8V...

MPW-2 SKY130A 1.1k

Select Open Source Designs

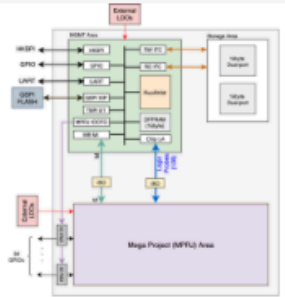


Caravel-SOFA-CHD public

Xifan Tang | https://sites.google.com/site/pegaillardon/home

SOFA-CHD (Skywater Opensource FPGAs)

MPW-1 SKY130 917

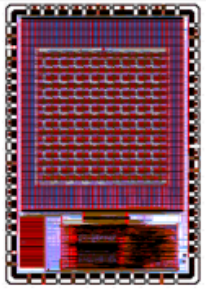


Caravel_RISCV_OSU public

James Stine | https://vlsiarch.ecen.okstate.edu/

Caravel_RISCV_OSU is an implementation of a single-cycle RISC-V processor inside of the Caravel...

MPW-1 SKY130 1.2k

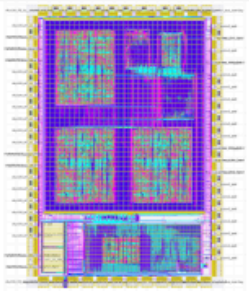


Caravel-OpenFPGA-EF public

Manar Abdelatty | http://efabless.com

A template SoC for Google sponsored Open MPW shuttles for SKY130.

MPW-1 SKY130A 1.8k

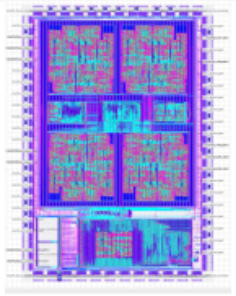


Chameleon SoC public

M. Shalan | http://efabless.com

AHB-Lite based SoC for IBEX

MPW-1 1.2k

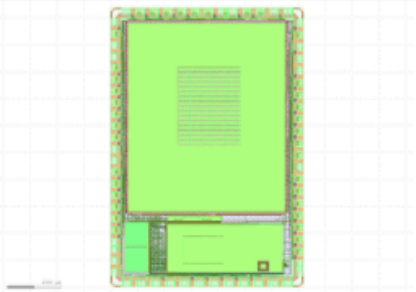


Caravel public

M. Shalan | http://efabless.com

NFive32-Based SoC to validate several open-source projects and IPs.

MPW-1 1.4k

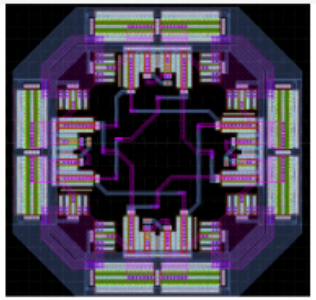


MorphleLogic public

Merik Voswinkel | http://www.fiberhood.org

A test-wafer for testing Mophle Logic reconfigurable hardware for SKY130.

MPW-1 SKY130 1.4k

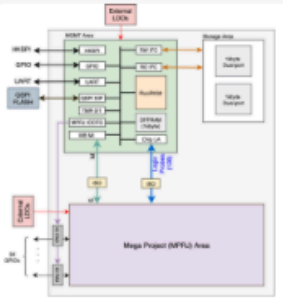


caravel_amsat_txrx_ic public

Thomas Parry | N/A

Amatuer Radio Satellite Transceiver (SKY130) - Caravel Submission

MPW-1 SKY130 2.8k



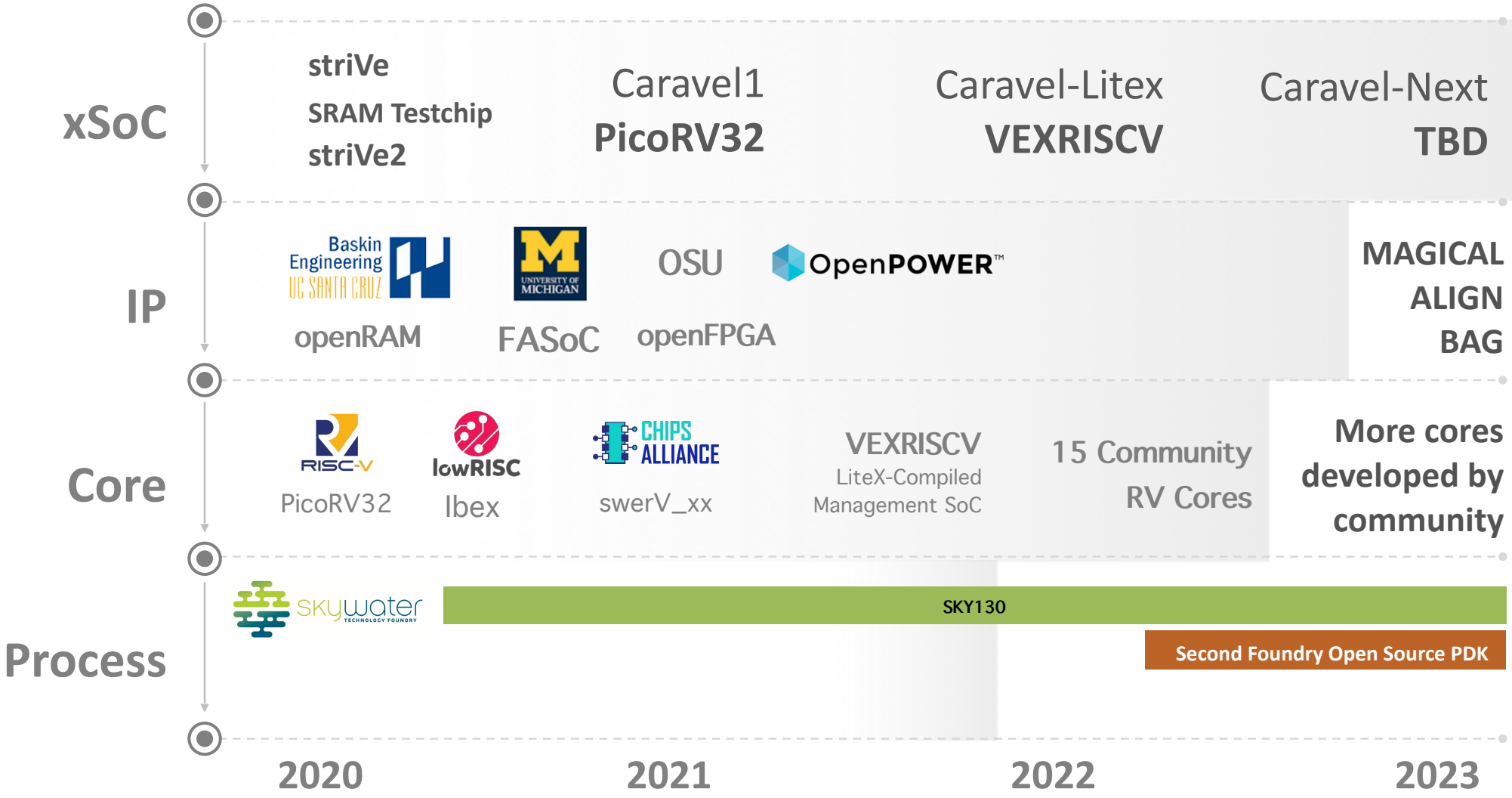
Softshell public

Harrison Pham | https://harrisonpham.com/

Multicore MCU for implementing software defined peripherals.

MPW-1 SKY130 1.7k

OPEN SOURCE ECOSYSTEM AT WORK



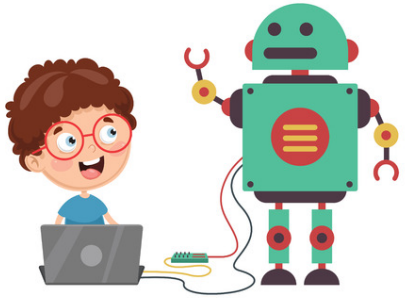
efabless.com

But there is always more…

Jump in and contribute

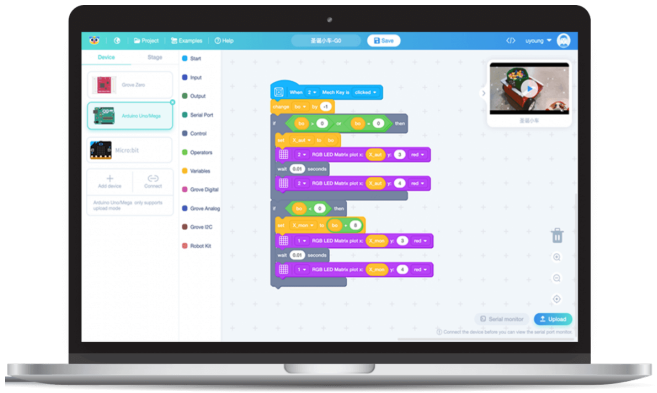
How do we simplify chip design? **#6**

Abstract access to knowledge

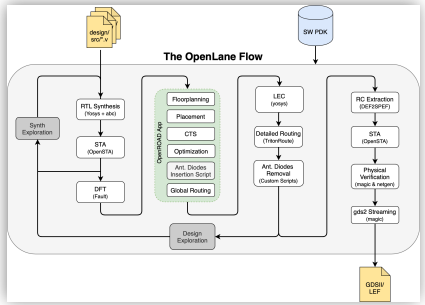


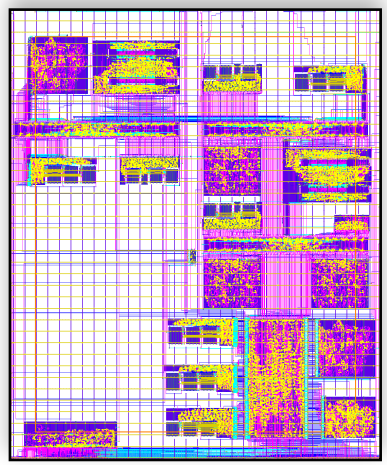
Think **Arduino** Raspberry Pi

*Use graphical tools or **code***



```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.scre
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ltdate = response.data[0]['created_at']
        ltdate2 = datetime.strptime(ltdate,'%a %b %d %H:%M:%S +0000 %Y'
        today = datetime.now()
        howlong = (today-ltdate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tweeted in the past' , daywindow,
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
                    for k in j.entities.urls:
                        newurl = k['expanded_url']
                        urlset.add((newurl, j.user.screen_name))
        else:
            print i.screen_name, 'has not tweeted in the past', daywind
```

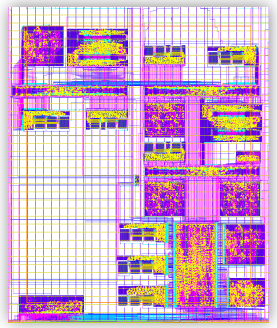




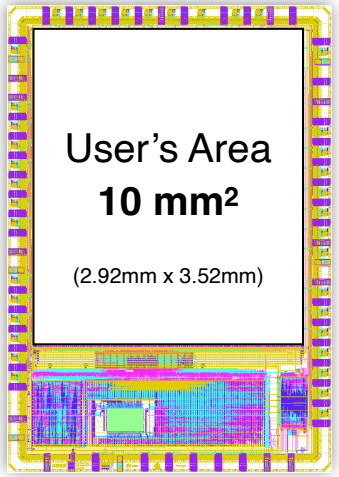
Scratch → Code → Make → Done

efabless.com

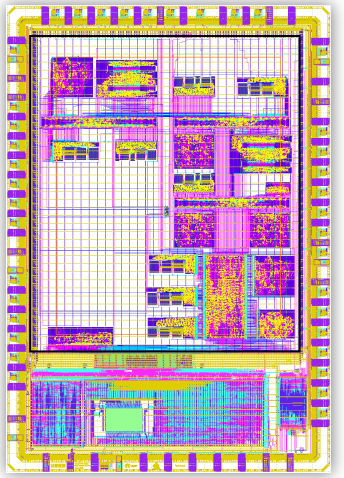
Arduino Scratch



My Design



CARAVEL

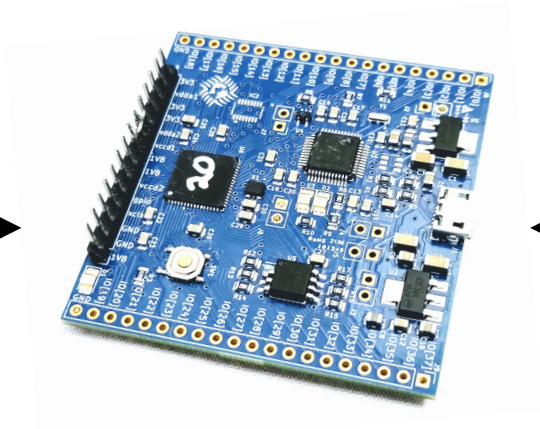


CARAVEL + My Design

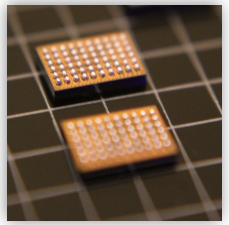
MANUFACTURING

Software (FOSS)

Programming



Plug and Play Dev Board

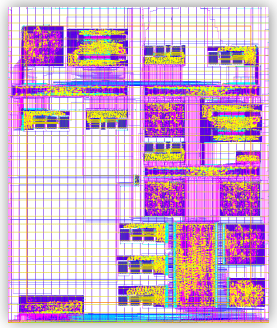


Parts

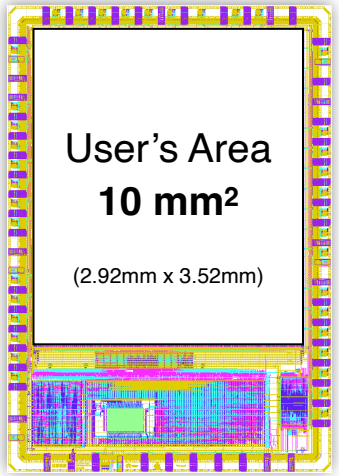
efabless.com

Python HDL Code

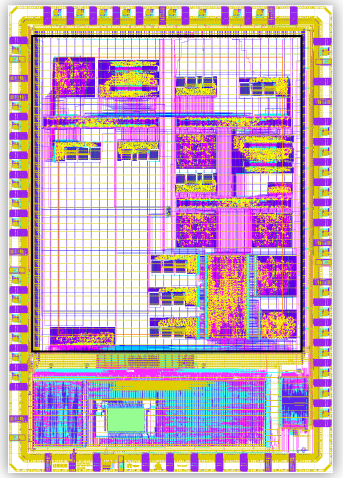
```
for i in people.data.users:
    response = client.api.statuses.user_timeline.get(screen_name=i.scre
    print 'Got', len(response.data), 'tweets from', i.screen_name
    if len(response.data) != 0:
        ltdate = response.data[0]['created_at']
        ltdate2 = datetime.strptime(ltdate,'%a %b %d %H:%M:%S +0000 %Y'
        today = datetime.now()
        howlong = (today-ltdate2).days
        if howlong < daywindow:
            print i.screen_name, 'has tweeted in the past' , daywindow,
            totaltweets += len(response.data)
            for j in response.data:
                if j.entities.urls:
                    for k in j.entities.urls:
                        newurl = k['expanded_url']
                        urlset.add((newurl, j.user.screen_name))
        else:
            print i.screen_name, 'has not tweeted in the past', daywind
```



My Design



CARAVEL

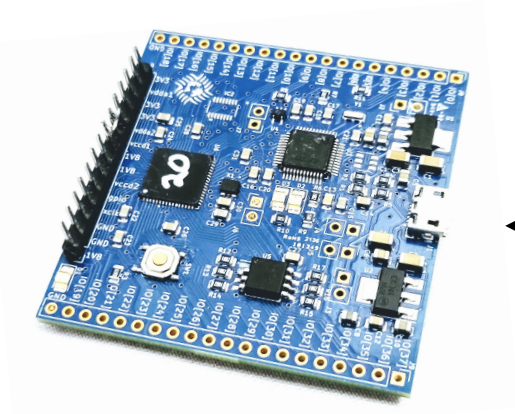


CARAVEL + My Design

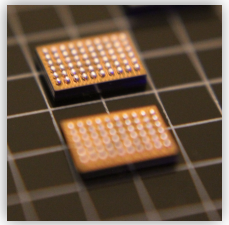
MANUFACTURING

Software (FOSS)

Programming



Plug and Play Dev Board

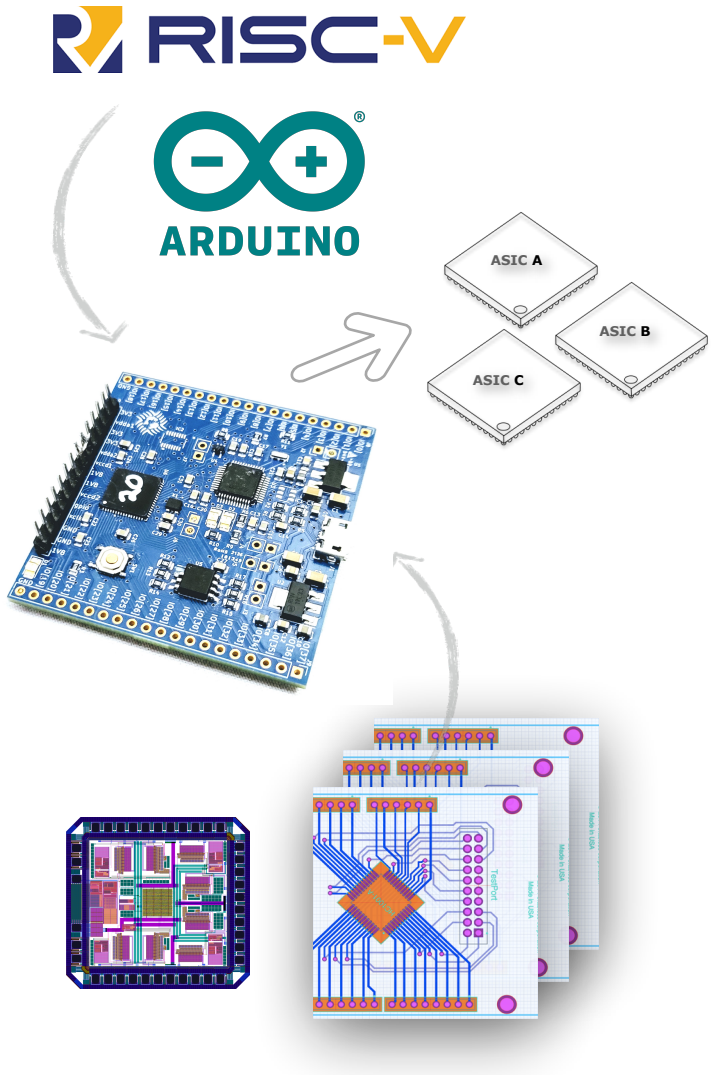


Parts

efabless.com

ARDUINO PROTOTYPING SHIELD

- **Simplified ASIC platform for prototyping - Arduino Shield**
- **Features**
  - Based on a standardized pad frame - used for multiple core chip.
  - Designed along with its companion test PCB & test SW.
  - Interfaces to a personal computer via USB.
  - Top-level is available to community members to modify & re-shuttle.
  - Intended as a design & test learning vehicle for analog/ms blocks
- **Included IP modules**
  - 10-bit SAR ADC - aadcc02
  - 10-bit voltage-scaling DAC - adacc02
  - GP Low Power BGAP - abgpc01
  - GP AMP - aopac01
  - GP Bias Cell - abiac01
  - SPI Controller - efdspi001
  - Dynamic Power-on-Reset - aporc01
  - LP CMOS Comparator - acmpc01
  - 14KHz RC Oscillator - arcoc01
  - 1-6MHz Crystal Oscillator - axtoc01



efabless.com

BUT ....

What if I do not want to open source my design?

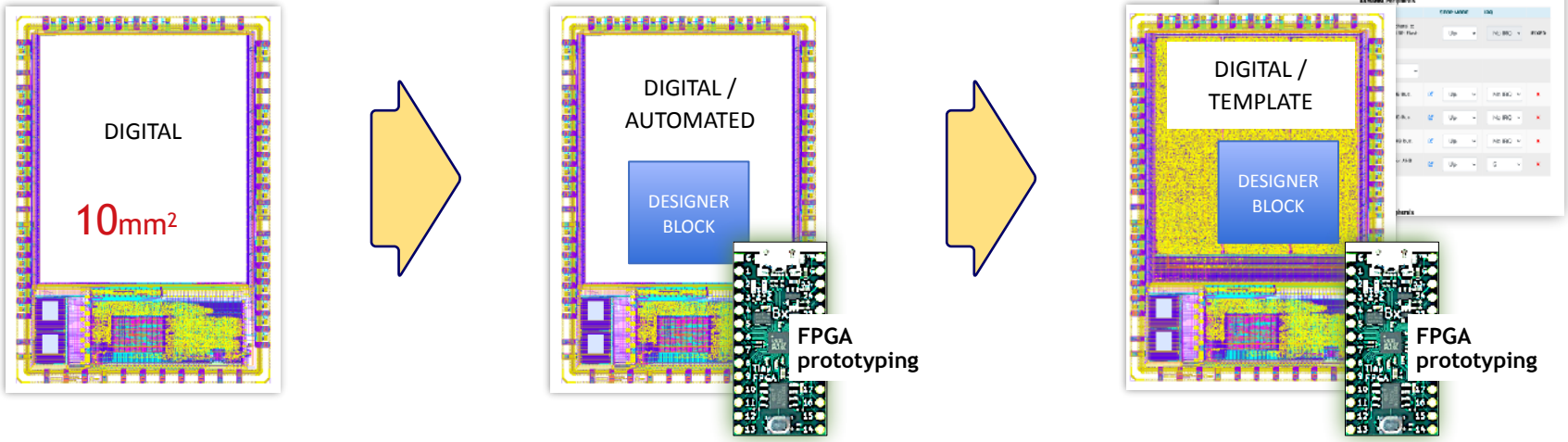
I would like to guarantee my spot on the run.

I need it on my own schedule.

chipIgnite OPTIONS

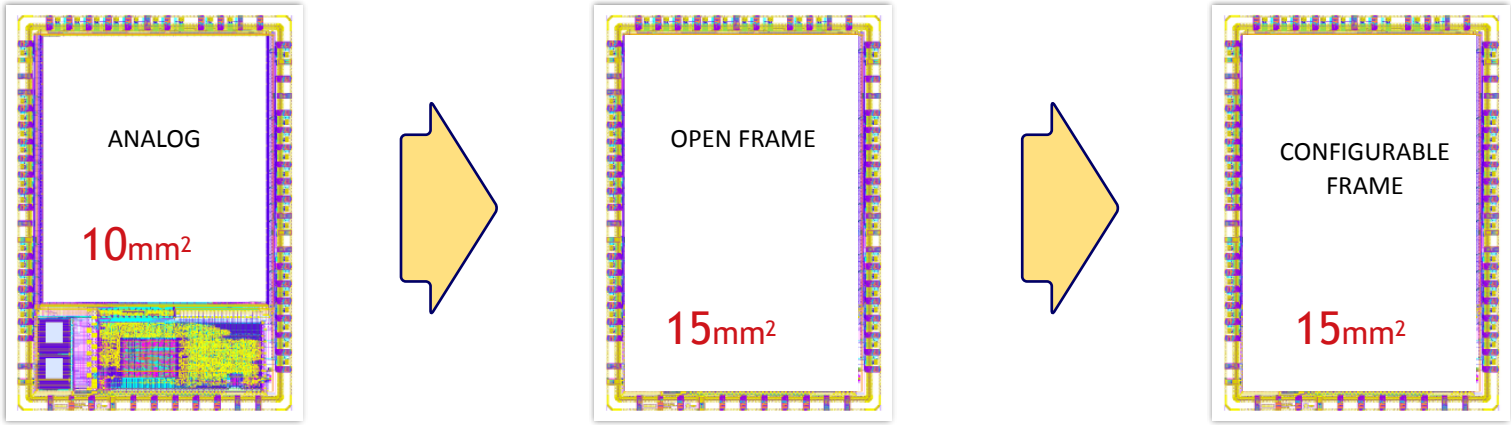
LOW COMPLEXITY

IP Development
Digital & low frequency analog
Enabling larger designer base



HIGH COMPLEXITY

IP Development
Complete Custom ASIC
Analog & Digital
Expert designer base



efabless.com

chipIgnite USERS

POWERED BY chipIgnite Rapid IC Creation

IEEE SOLID-STATE CIRCUITS SOCIETY™ IC Innovation

1

Stanford **EE272** Design Course

Three more universities …

High Schools coming in June 2022

2

SSCS "**PICO**" Open Source Design Contest:

56+ submitted designs

10 will tape out at Efabless

https://sscs.ieee.org/about/solid-state-circuits-directions/sscs-pico-program
https://efabless.com/projects/project_definition/SSCS-21

3

Open Source FPGA Foundation global program innovation among university students

2 **ChipIgnite** Slots per University

Turkey, Pakistan, India, Australia

4

Startups

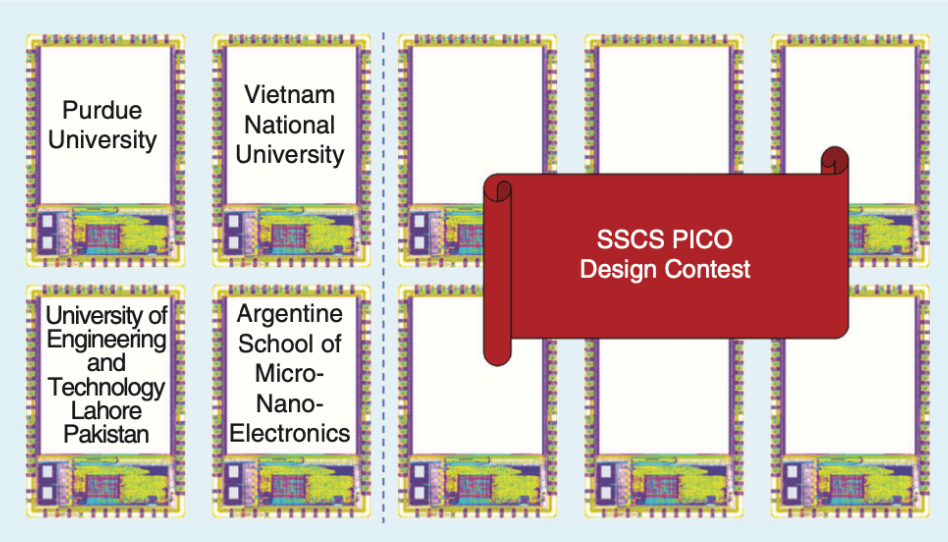
Using chipIgnite as their proof-of-concept or low volume production

efabless.com

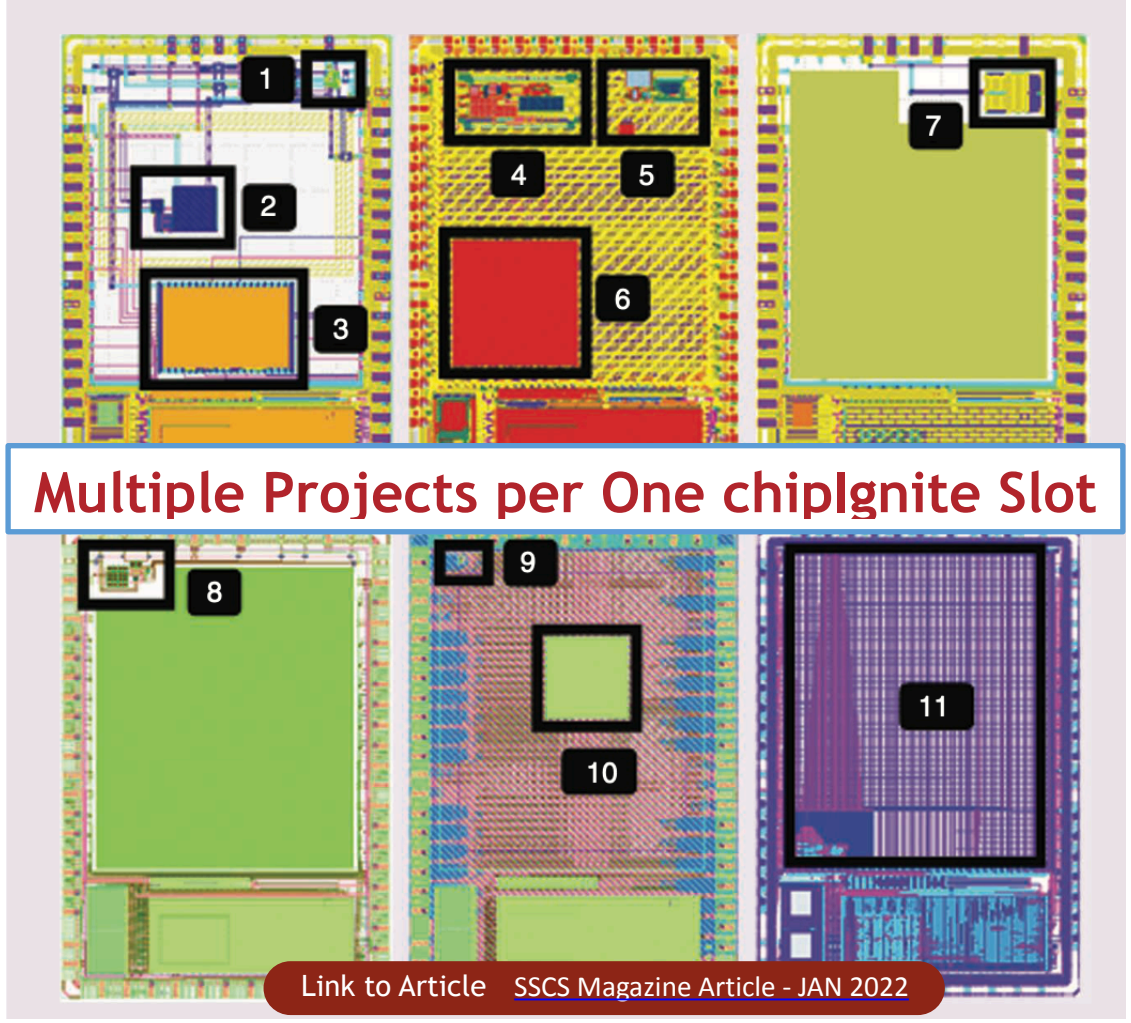
IEEE - PICO PROGRAM

IEEE SOLID-STATE CIRCUITS SOCIETY™ IC Innovation

POWERED BY chipIgnite Rapid IC Creation



Link to Article SSCS Magazine Article - NOV 2021



Multiple Projects per One chipIgnite Slot

Link to Article SSCS Magazine Article - JAN 2022

efabless.com

*~ $10K … still **not** a small number …*

Crowdfund it …

… or Community-fund it

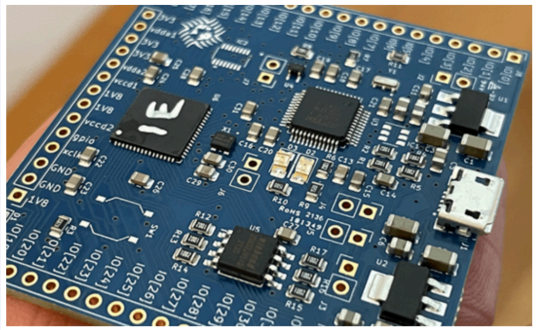
Write up your Idea

Go to GroupGets.com or CrowdSupply.com

Start a campaign at no risk - see example CLEAR

If your design gets funded (supported by the community)

then you got your own board with your custom ASIC



CLEAR - The Open Source FPGA ASIC - by chipIgnite

Single-unit price: **$74.99** + shipping

108% Funded

Not available / campaign end date: Mon, 28 Mar 2022 18:10:00 PDT

CLEAR

Backers: 142

Target: 200

Funded: 215

Buy Disabled

Details | Discuss | Updates

Product Details

Name

CLEAR - The Open Source FPGA ASIC - by chipIgnite

Brand / Manufacturer

Efabless Corporation

Details | Discuss | Updates

Product Details

Name

CLEAR - The Open Source FPGA ASIC - by chipIgnite

Brand / Manufacturer

Efabless Corporation

Manufacturer Part

CI00025

Product Description

CLEAR - The Open Source FPGA ASIC

POWERED BY

chipIgnite
Rapid IC Creation

CLEAR is an Open Source FPGA ASIC delivered to you on its development board and its open source software development tools and all the ASIC design tools used to create it. That's for you to create your own - yes that's right - ASIC.

CLEAR is utilizing *Caravel* - the open source ASIC platform provided as a base System on Chip (SoC) provided by Efabless' *chipIgnite* offering.

What we will do

1. Generate an embedded FPGA macro based on the famous *OpenFPGA* generator framework
2. Integrate the eFPGA macro into *Caravel* which makes it an FPGA ASIC - Clear
3. Manufacture the ASIC through chipIgnite program through *SkyWater Technology*
4. Package and mount the ASIC on its development board
5. Test the boards before you get them along with the open source FPGA programing software
6. You receive your board and have fun!

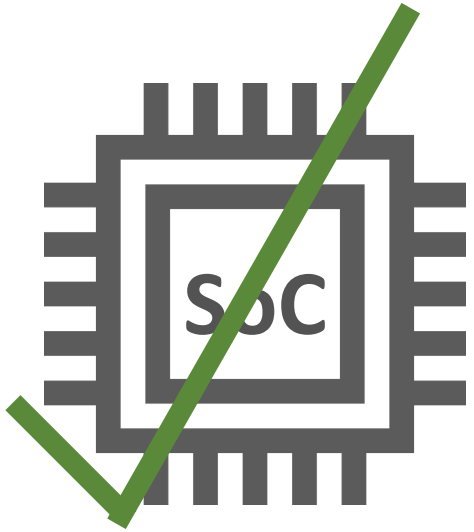
As part of the campaign we will show you ***everything*** we do including how to design ***your own ASIC*** with open source ASIC design software and how you can create a campaign just like this one for your own custom ASIC. All that without having to make a giant hole in your pocket for ASIC design and manufacturing.

efabless.com

….. that's how its done …

1. Market Access
2. Expertise
3. IP
4. EDA Tools
5. PDK
6. AFFORDABLE MFG/ASSY/TEST

At least these **6** things must exist to create useful chips



CAN BUILD A USEFUL **CHIP**

efabless.com

Creating a World where a 14-year-old Designs a Chip

The time is now

Get Involved - Useful Links / Repos

Information Hub

github.com/efabless/skywater-pdk-central

- Join SkyWater PDK Slack Space - https://join.skywater.tools
- The OpenLane flow for digital PnR can be found at https://openlane.io
- The OpenROAD Project https://theopenroadproject.org/
- The documentation is at https://docs.skywater.tools
- MPW-ONE https://efabless.com/open_mpw_shuttle_project_mpw_one
- Caravel documentation https://caravel-harness.readthedocs.io/en/develop/index.html
- WOSET/ICCAD Workshop on EDA https://woset-workshop.github.io/WOSET2020.html

Supporting Slides

chipIgnite

FOR UNIVERSITIES

Undergraduate courses

Capstone projects

Graduate **Research**

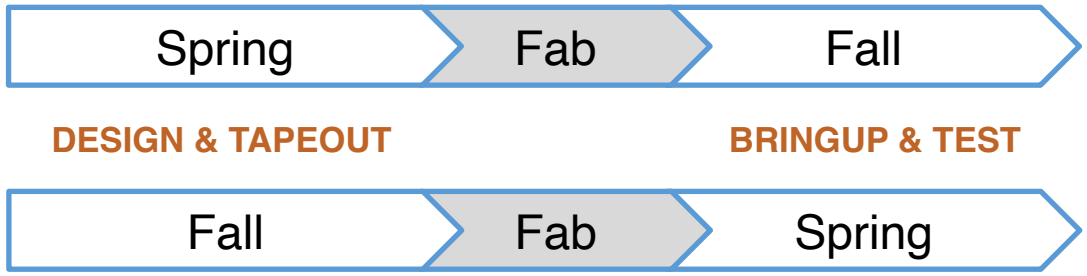
skywater

efabless.com

TAPE-OUT COURSES

Analog and / or Digital Design Courses

- **Two-session courses:**



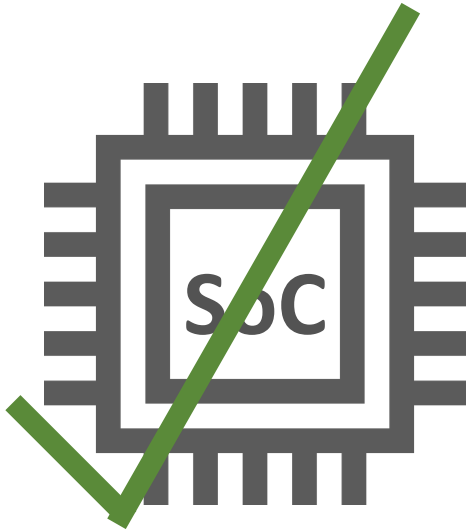
- **Single session courses:**
  - Design + test via FPGA
  - Design + test previous student project
  - Test broken design + fix and tapeout
  - Future: one course period design/tape-out/test

efabless.com

..... what we need ...

1. **Market Access** — Open Online Marketplace for Chips Like the App Stores
2. **Expertise** — Open Community Collaboration Clusters of global teams
3. **IP** — No NDA or Access Fee Open Source or Obfuscated
4. **EDA Tools** — No Licensing Cost = Open Source Low Licensing Cost = Cloud-based
5. **PDK** — No NDA = Open Source or Obfuscated to protect IP
6. **AFFORDABLE** MFG/ASSY/TEST — No NDA = Open Source or Obfuscated to protect IP while enabling designers to use it

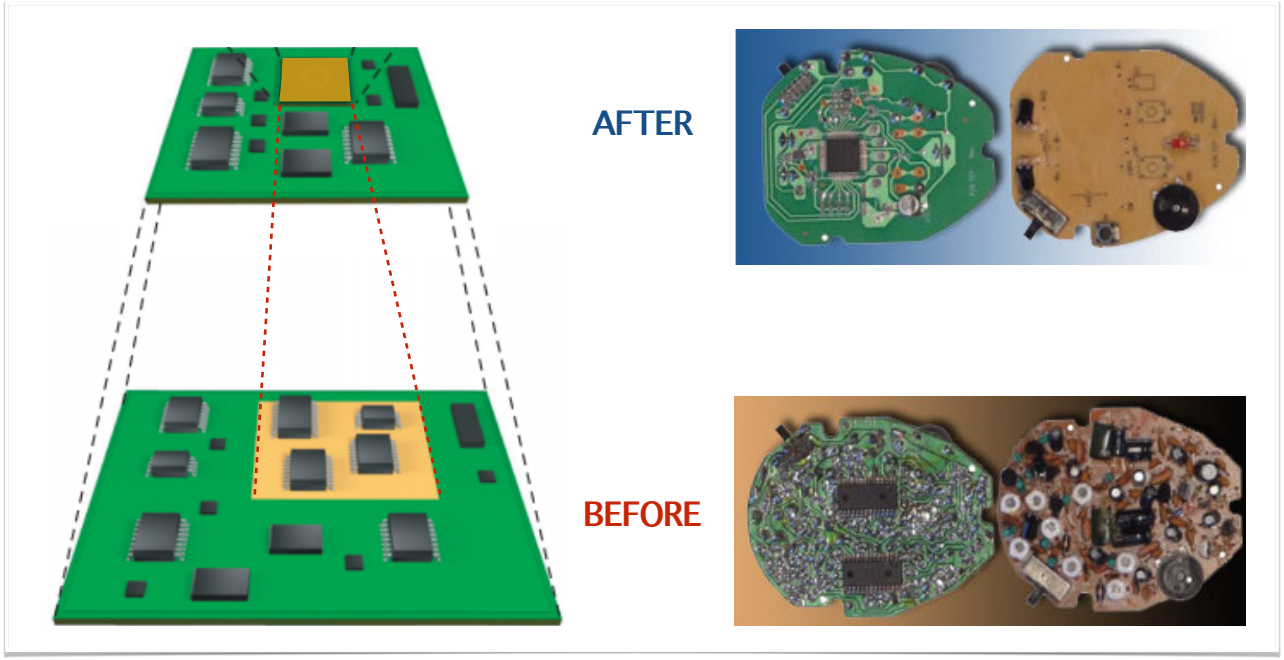
Solve **1** at time



CAN BUILD USEFUL **CHIPS**

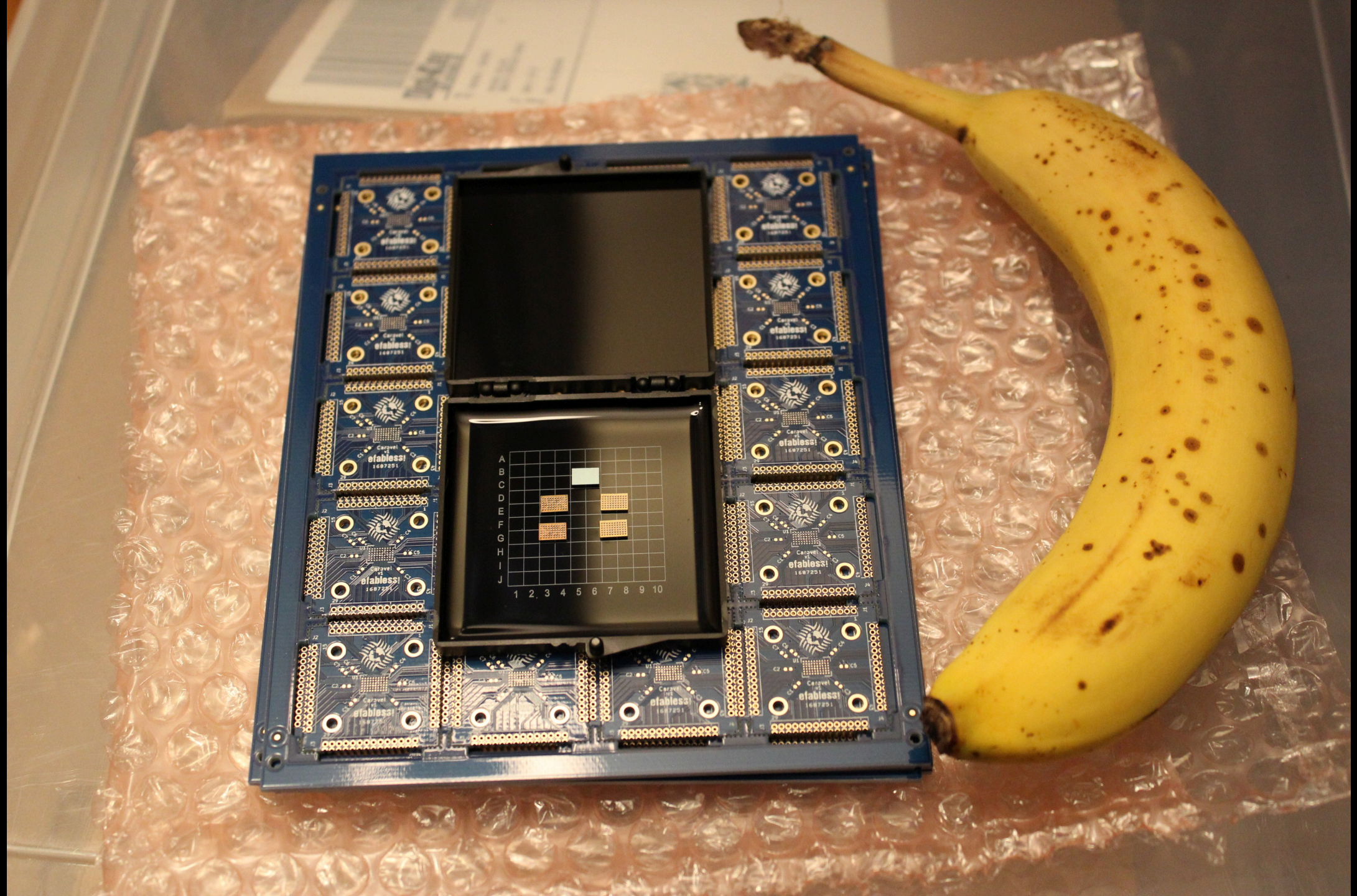
efabless.com

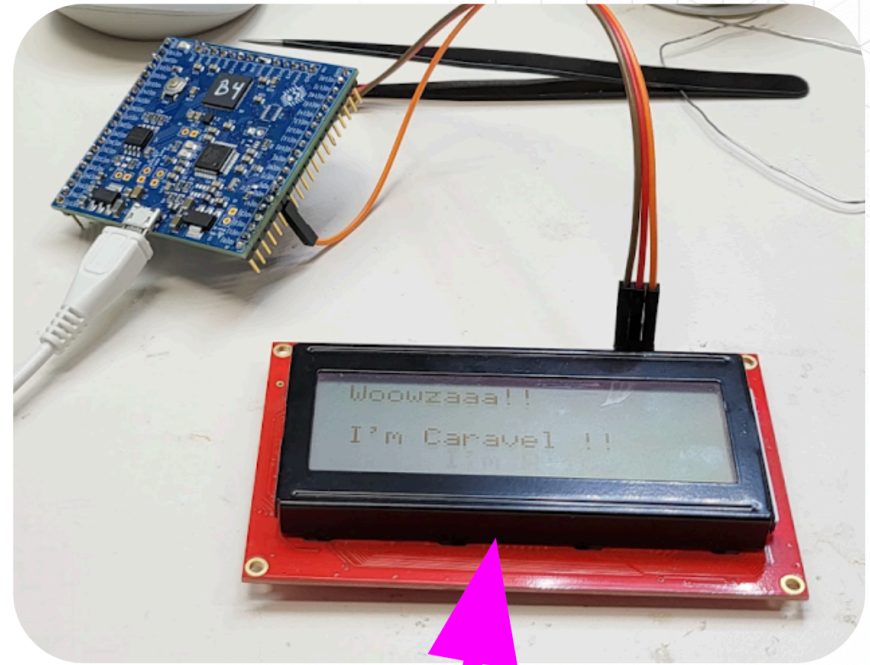
Why Care to have your own ASIC for your product?



- Reduce cost in your overall application
- Optimize performance and power efficiency via integration
- Reduce the size and enable superb Product Design
- Protect Intellectual Property through hardware embedding

efabless.com

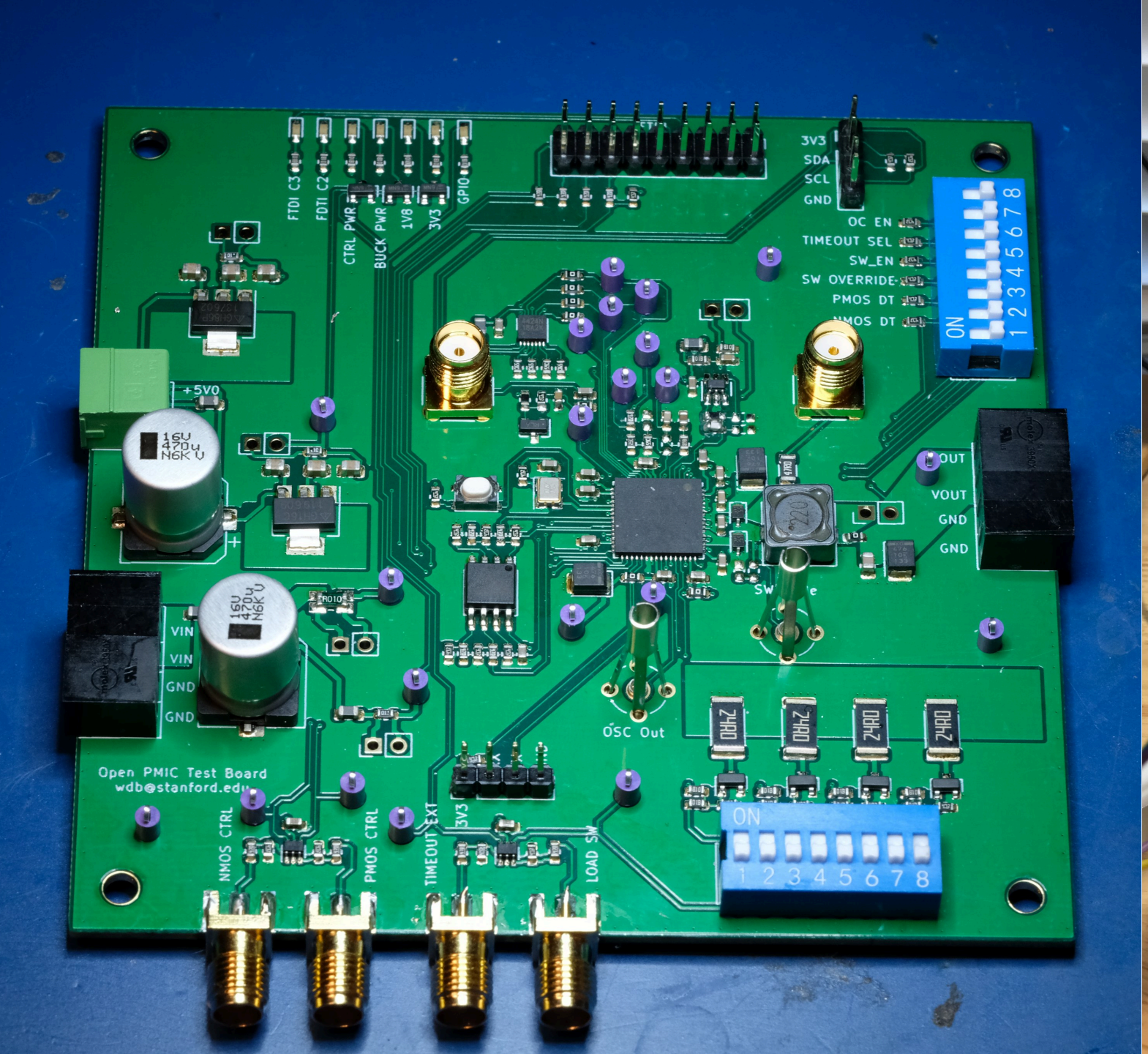

A
B
C
D
E
F
G
H
I
J
1 2 3 4 5 6 7 8 9 10



Woowzaaa!!
I'm Caravel !!

bit.ly/cicc22-edu-goog

efabless.com


FTDI C3
FDTI C2
CTRL PWR
BUCK PWR
1V8
3V3
GPIO
3V3
SDA
SCL
GND
OC EN
TIMEOUT SEL
SW_EN
SW OVERRIDE
PMOS DT
NMOS DT
ON
1 2 3 4 5 6 7 8
+5V0
16V
470u
N6K V
VIN
VIN
GND
GND
OUT
VOUT
GND
GND
OSC Out
74HC
Open PMIC Test Board
wdb@stanford.edu
NMOS CTRL
PMOS CTRL
TIMEOUT EXT
3V3
LOAD SW
ON
1 2 3 4 5 6 7 8

WaveForms (new workspace)

Welcome | Help | Scope 1 | Logic 1 | Wavegen 1

View Window

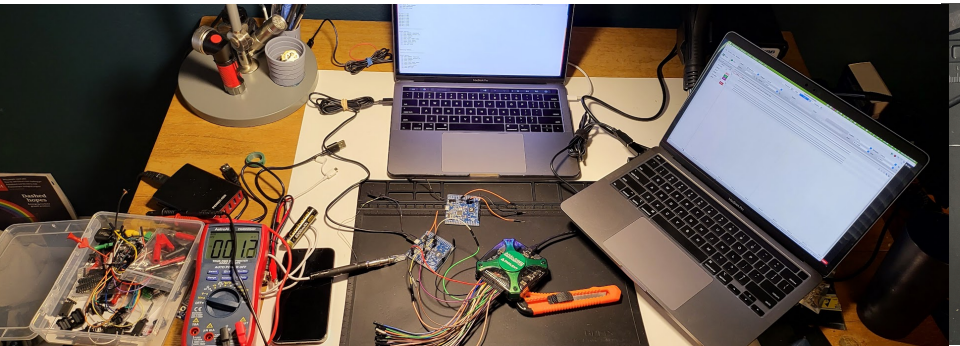
Stop

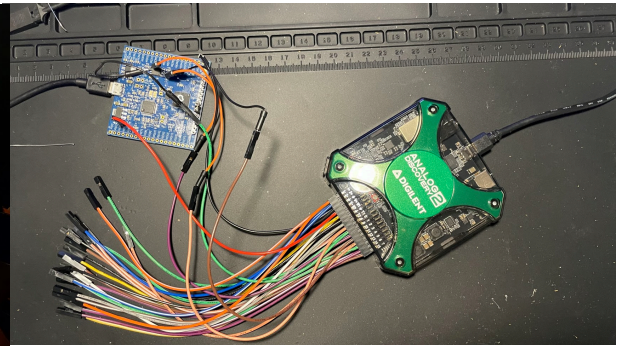
Mode: Repeated Trigger: Normal Simple Pulse Protocol Position: 9.002 us

Buffer: 10 Source: Digital Inputs: 100MHz DIO 0..15 Base: 2 us/div

Pin T Armed Name: FLASH IN Position: 0.00000404s Value: h03

	hFF	hFF	hFF	hFF	hFF	hFF	h93	h00	h00	h00	? h80
*DIO 0											
*DIO 1											
DIO 3											
	hFF	hAB	h03	h00	h00	h00	h00	h00	h00	h00	? h0
*DIO 0											
*DIO 1											
DIO 2											
DIO 4											





ANALOG TRANSISTOR-LEVEL

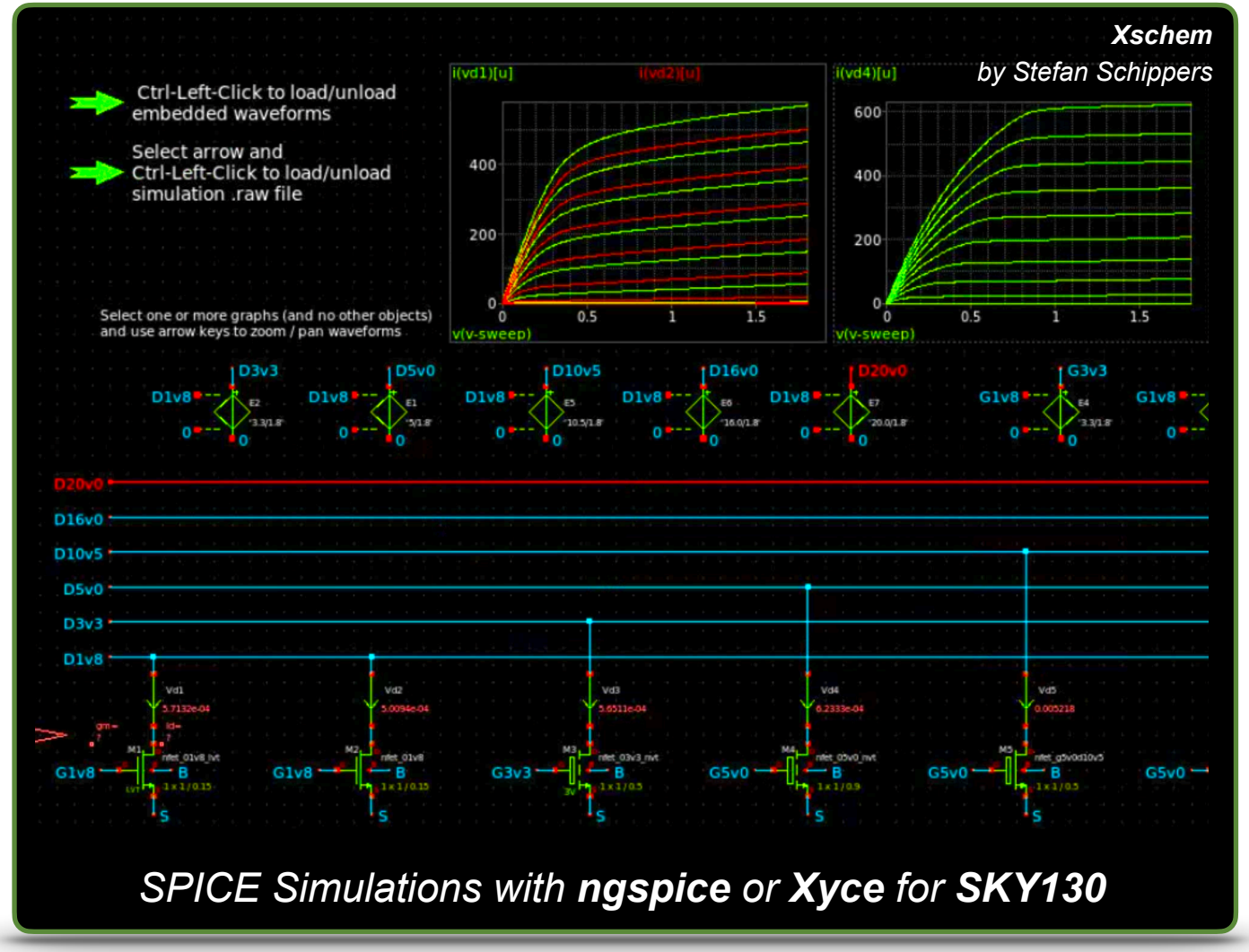
Analog transistor-level design, schematic and layout

Schematics: Xschem

Simulators: ngspice, Xyce

Support: Process Corners and statistical simulations

Packed with out-of-the-box examples - just click, copy and run yours



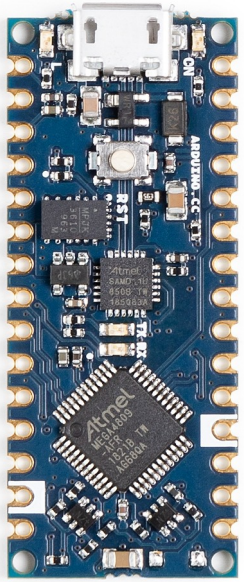
*SPICE Simulations with **ngspice** or **Xyce** for **SKY130***

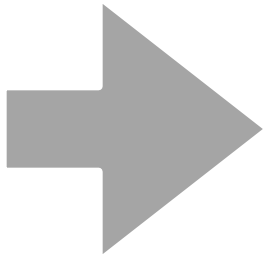
efabless.com

ARDUINO ASIC/SoC - WHAT - just thoughts

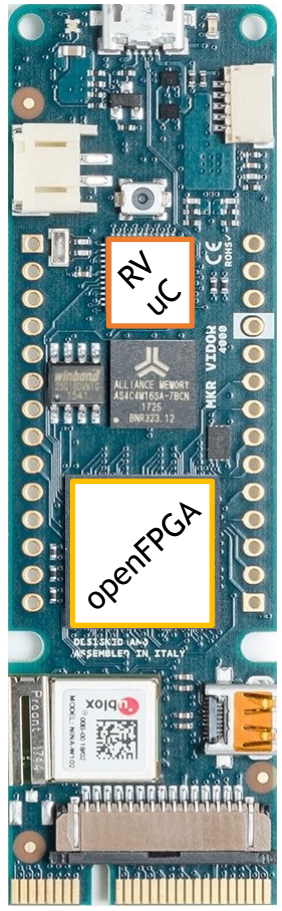
Arduino MKR VIDOR 4000

Arduino Nano Every

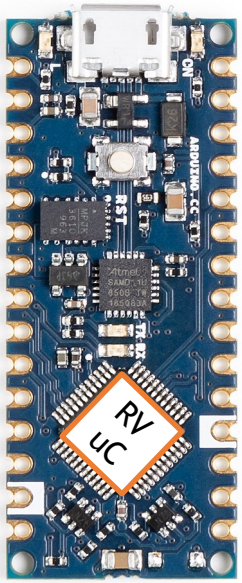




Arduino MKR OpenASIC



Arduino Every OpenASIC



efabless.com