

半導体（ASIC/LSI/IC）の
民主化の始まり
～自作半導体の作り方～
2024年バージョン

ODC2024

今村謙之 (Noritsuna Imamura) @ ISHI会

noritsuna@ishi-kai.org

本講演のターゲット層

省電力、省スペースに悩んでいる方

- 市販のICを用いた基板設計に限界を感じている
- 自作半導体（ASIC/LSI/IC）が作りたい！

処理速度不足に悩んでいる方

- FPGAでの処理に限界を感じている
- 自作半導体（ASIC/LSI/IC）が作りたい！


半導体（ASIC/LSI/IC）の製造に興味のある初心者

- どんなものでもよいのでオレオレ半導体を作ってみたい方


アジェンダ

- ISHI会とは？
- 知りたいことは？
- どうやって作るの？
- 初心者向け教材
- 国内コミュニティー紹介
- 今後のイベント紹介





ISHI会とは？



ISHI会とは？



- オープン化（民主化）されたISHI=石=Silicon=半導体（ASIC/LSI/IC）を扱い、いろいろな分野を繋げていくソサエティー・コミュニティ（会）から発想されたネーミングです。
- その先駆けとして登場したOpenMPW(Open Multi Project Wafer)は、Google社がEfabless社に出資して生まれたシャトルプログラムであり、半導体（ASIC/LSI/IC）を作るうえで必要なツ

半導体の設計・製造したことない人たちを
支えるコミュニティ



起こすという方針で、他分野向けの超初心者向けハンズオンセミナーや専門家向けの濃い内容の勉強会などのイベントを開催したり、チームを作ってOpenMPWシャトルや世界のChipathonに挑戦したり、Maker Faireなどのイベントへの参加をしていきたいと思っておりますので、よろしくお願いいたします。

ISHI会グラウンドデザイン

新規分野を開拓したいけどどうすればよいのかわからない

「みんなの経験をチップに！」

ASIC(LSI)化したいけど情報がない

ASIC(LSI)業界の現状（閉塞感）

- NDAでなにもしゃべれない
- 最先端は札束の応酬
 - 若者が入ってこない

他業界の現状（限界感）

- 高速・小型・省電力の要求
 - 汎用チップ+ソフトでは限

すべてがオープン！

OpenMPWの登場！

コミュニティの意義

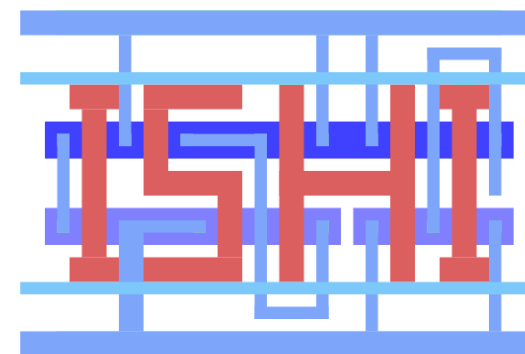
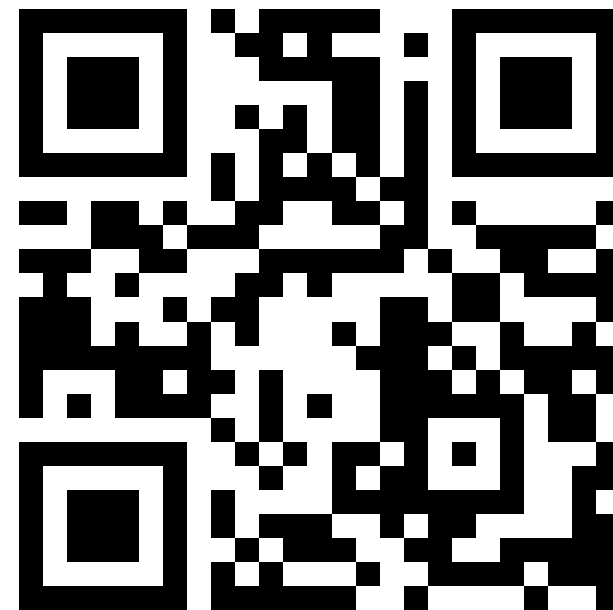
- 成果の再利用が可能。Do It With Others(それ、みんなで作ってみよう)の精神
- 日本の利点：地理的に物理的に集まりやすく、勉強会や合宿をやりやすい

ISHI会の意義

- 他（多）分野の知識の統合により、今までになかった研究・開発への期待

活動場所

- ホームページ
 - Github pages
 - <https://ishi-kai.org/>
- Discord上で活動中
 - <https://discord.gg/RwAWF5mZSR>
- イベント告知（勉強会など）
 - connpass
 - <https://ishikai.connpass.com/>

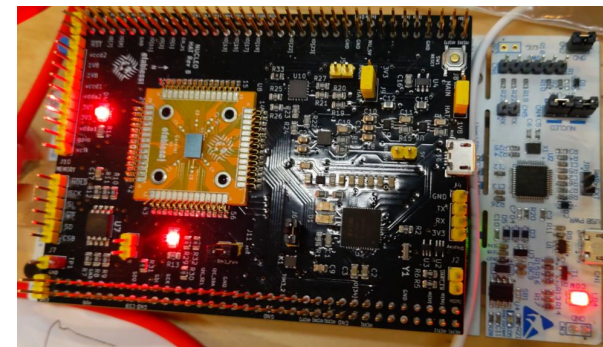
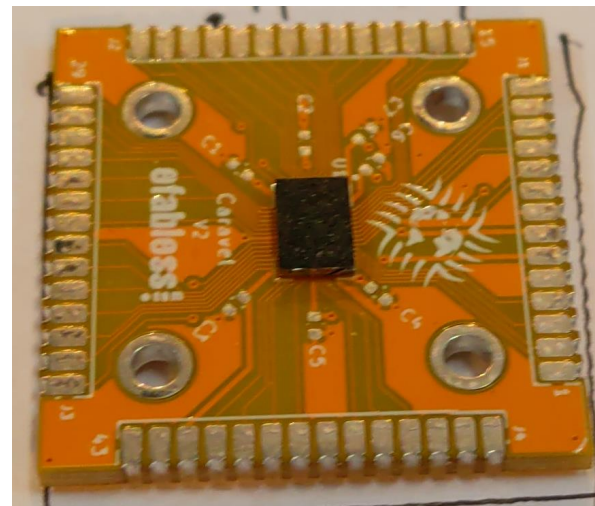
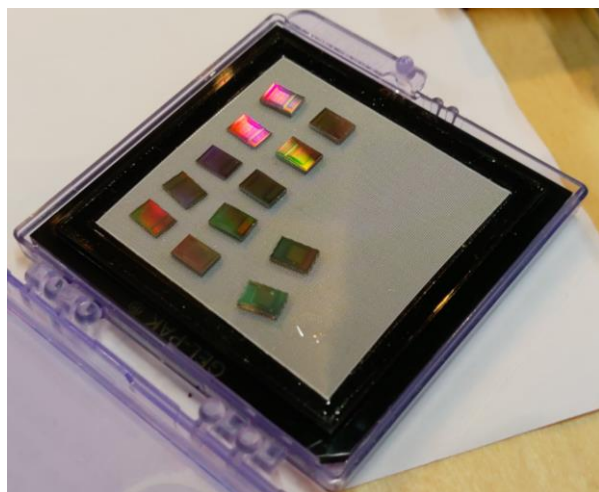
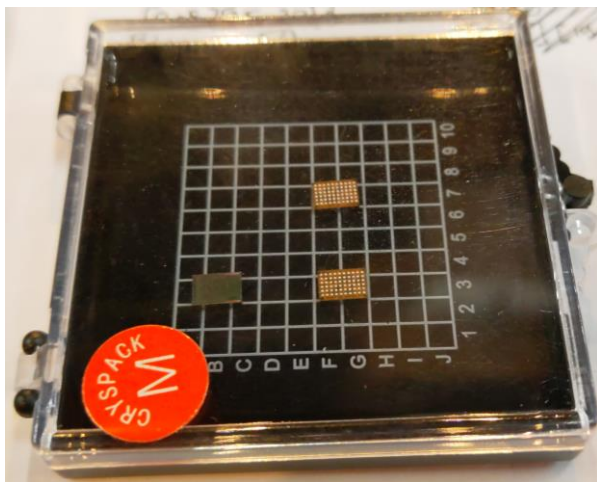


で、何しているの？




その前に . . .






作りたいたいのはこちらですね？



知りたいことは？

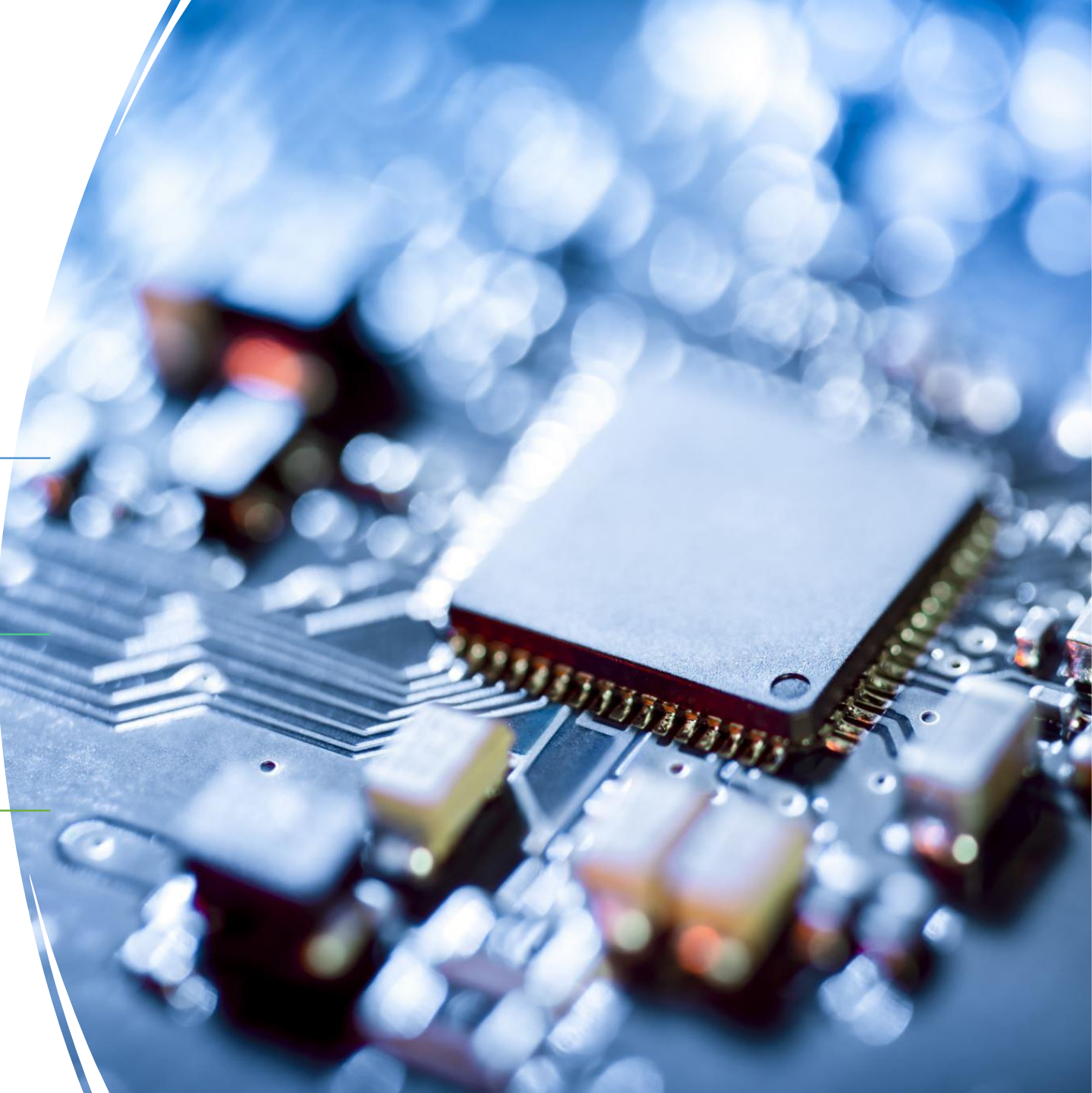


知りたいのは
これらですよね？

どんな機能の半導体が作れるのか？

どうやったら半導体が作れるのか？

それが自分でもできるのか？





RISC-V CPU (デジタル)

- オープンソースで提供されている命令セットアーキテクチャ (ISA)
- オープンソースであり、誰もが使え、ライセンス料が無料、契約の必要なし
- 拡張性が高く、必要な命令だけを選択して実装できる
- 多様性があり、様々な用途やアプリケーションに対応できる

AnalogDiscovery2 (アナログ)

- 2chオシロスコープ (14bit, 100 MSa/s)
- 2ch 任意波形発生器 (14bit, 100 MSa/s)
- 16chパターンジェネレータ (100 MSa/s)
- 16ch 仮想デジタルIO
- 16chロジックアナライザ
- 2入力/出力デジタルトリガ
- 2出力プログラマブル電源 (5 V, 2.1 W)
- 電圧計 (AC/DC)
- ネットワークアナライザ (10 MHz)
- スペクトラムアナライザ
- バスアナライザ (SPI, I2C, UART, パラレル)



どうやって作るの？

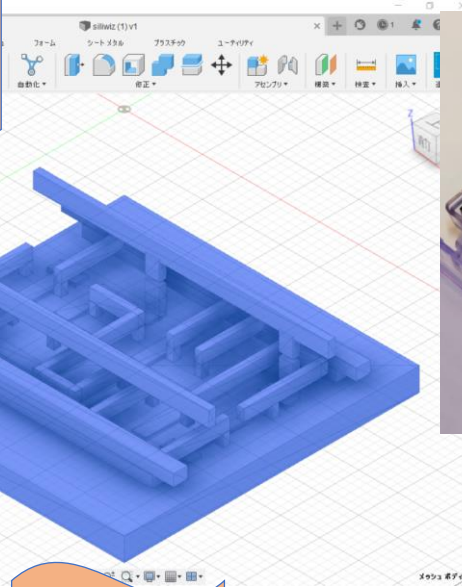
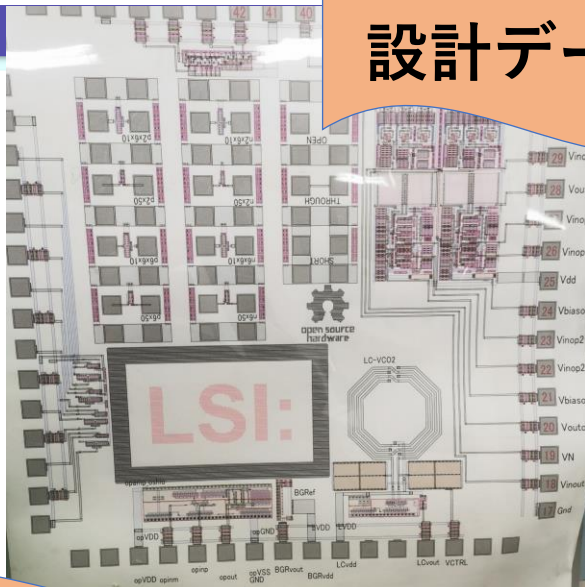


半導体の設計～製造～動作確認まで

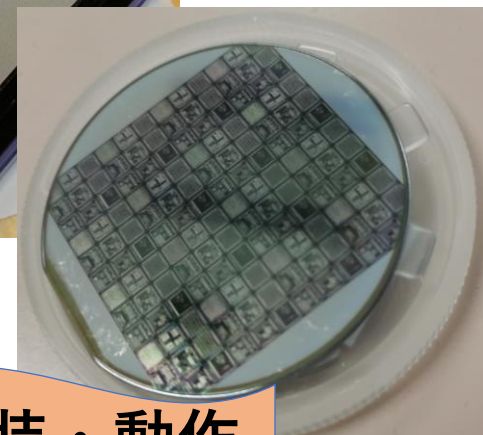
設計



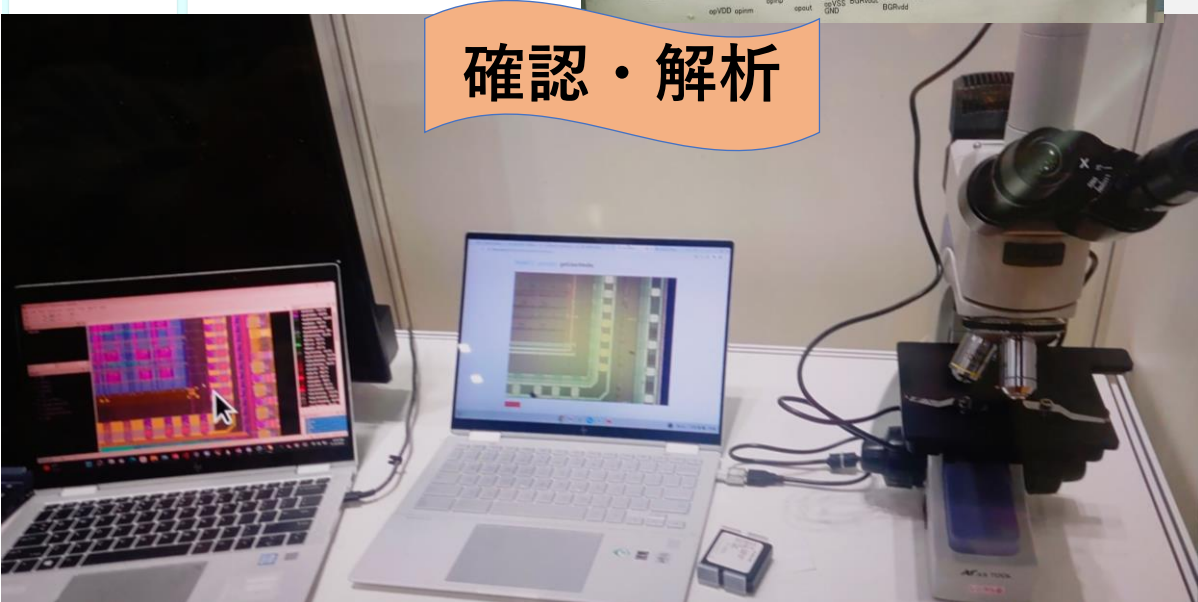
設計データ



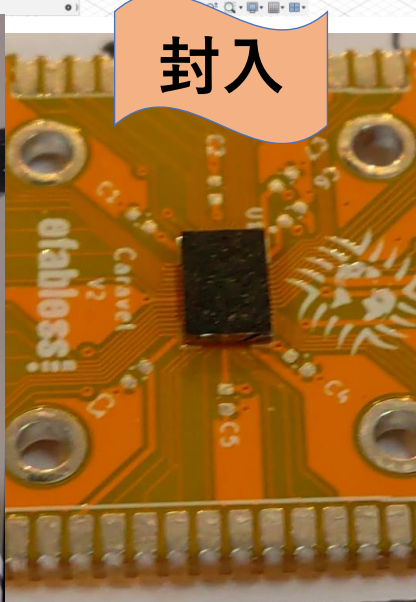
製造



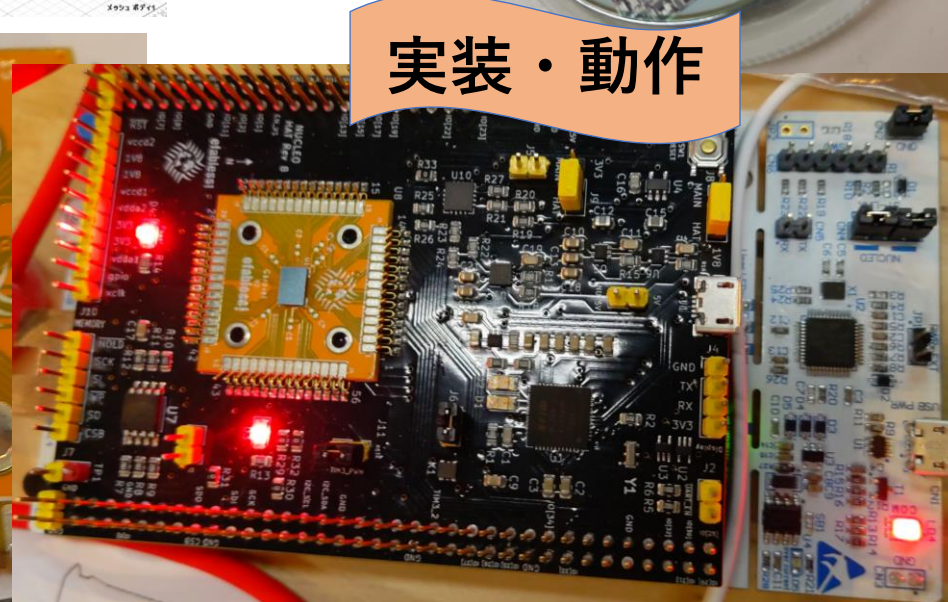
確認・解析



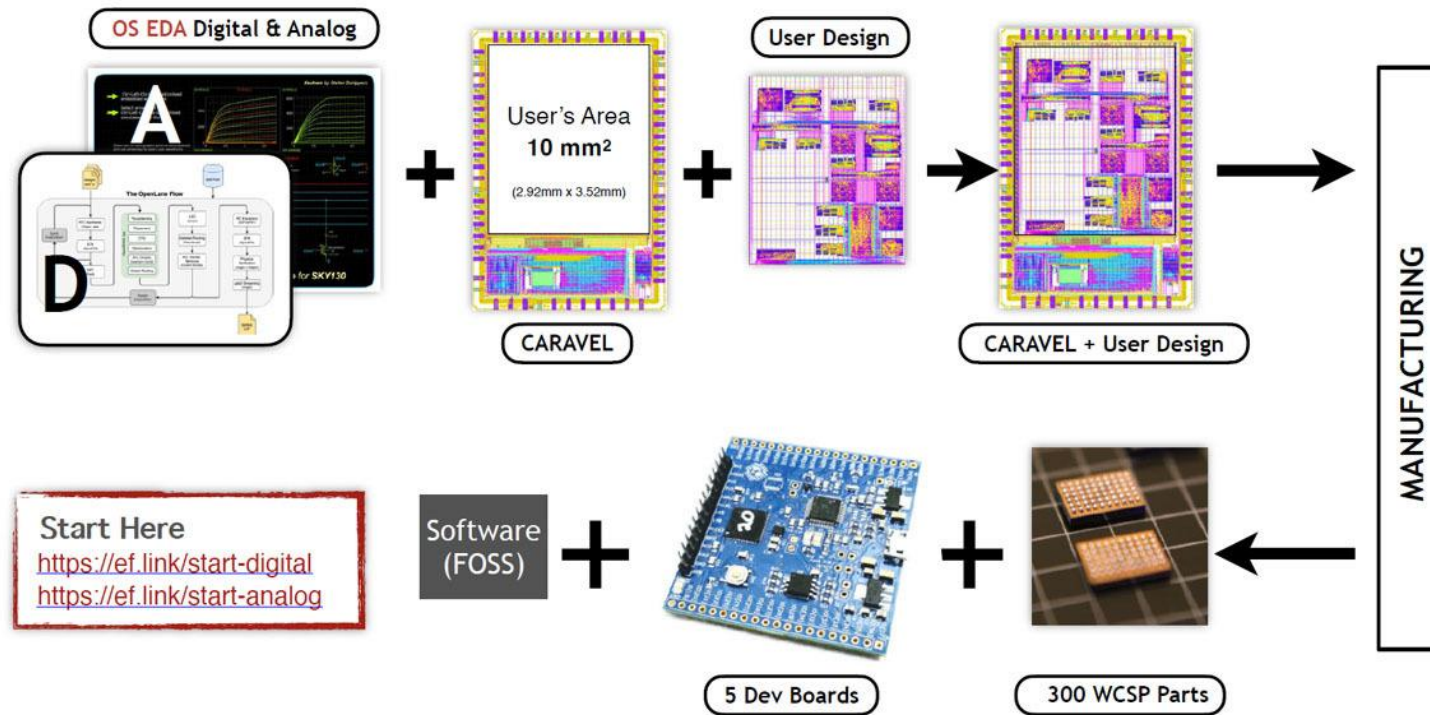
封入



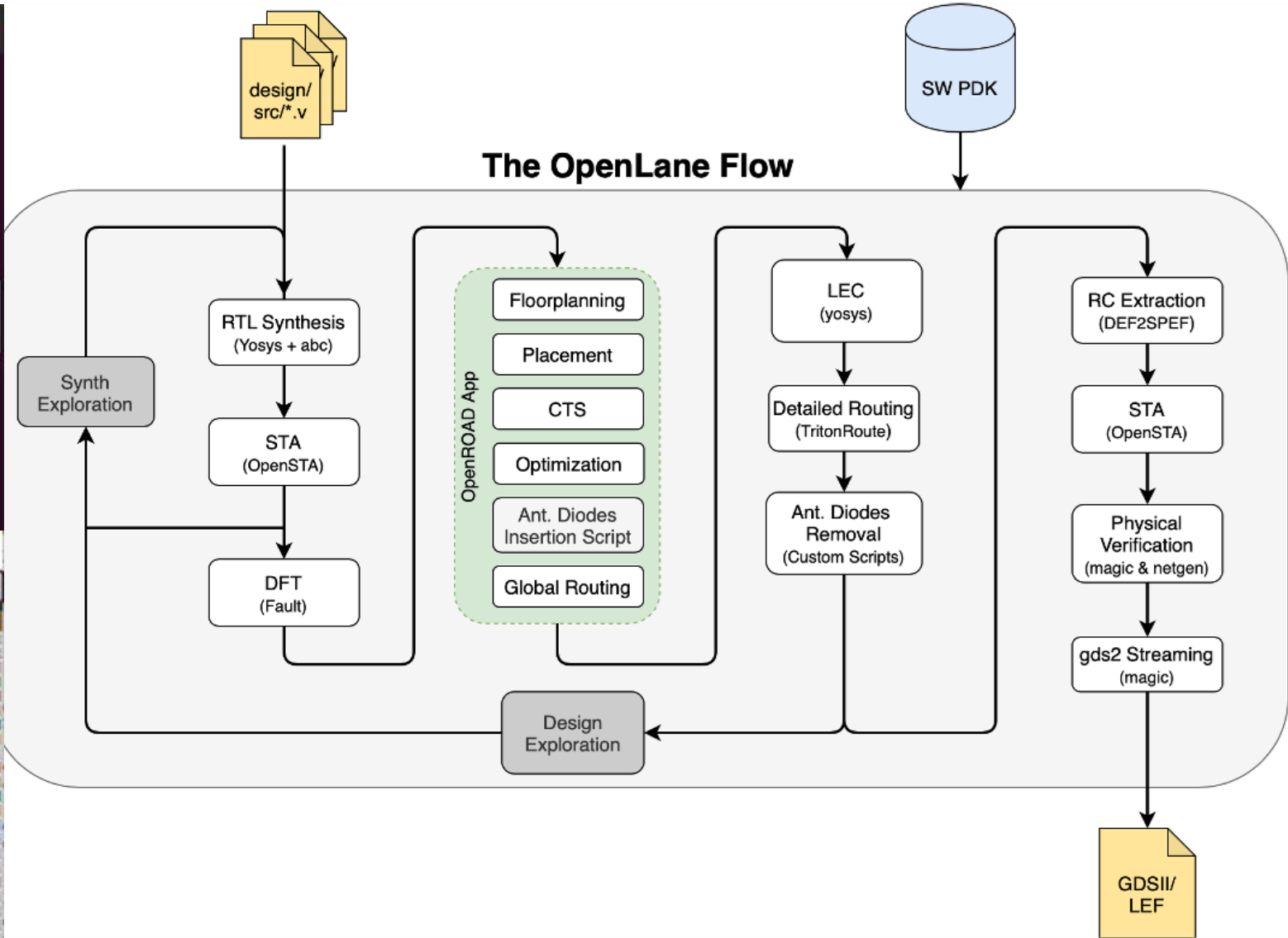
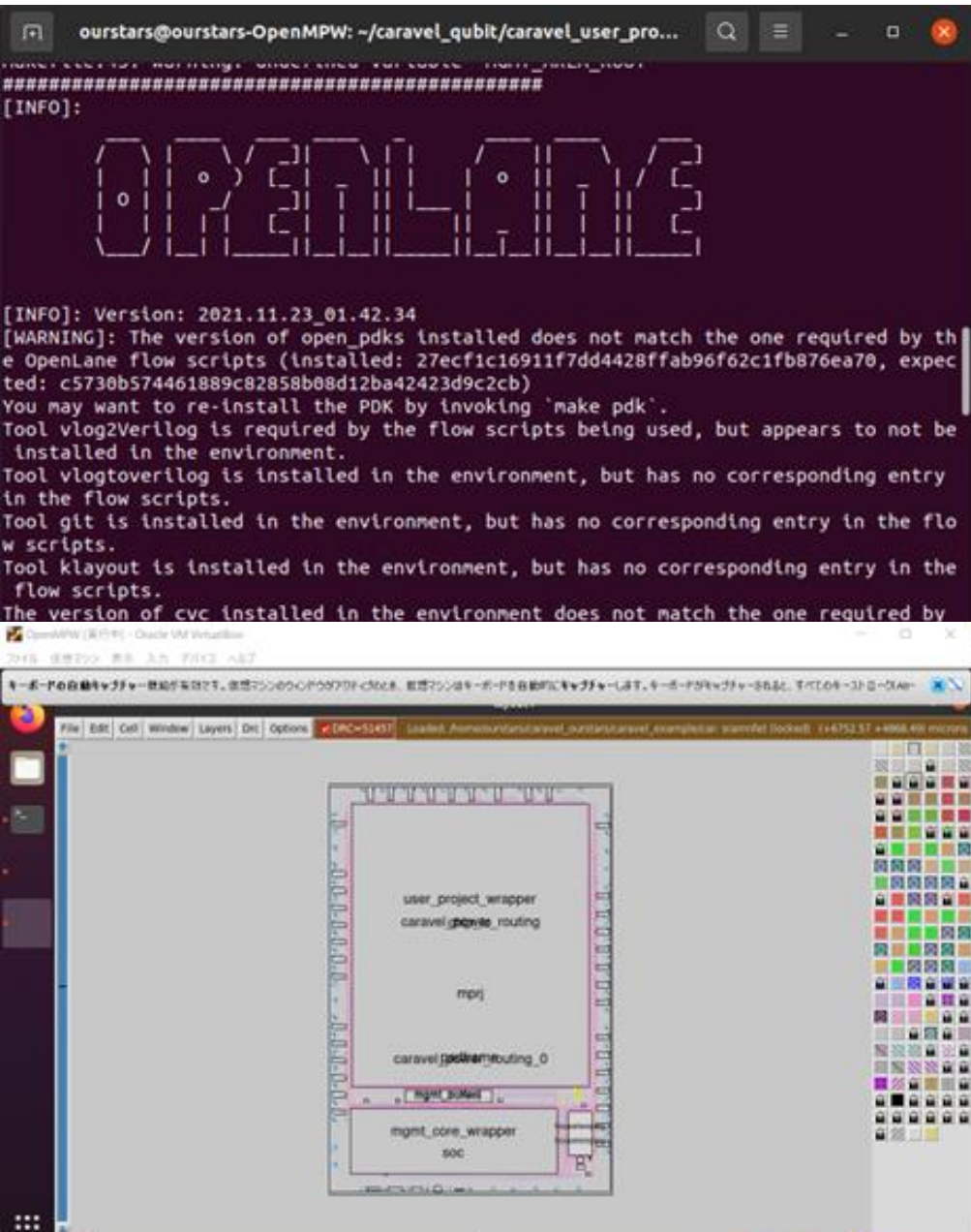
実装・動作



半導体の設計～製造までのフロー



設計ツール：デジタル



設計ツール：アナログ

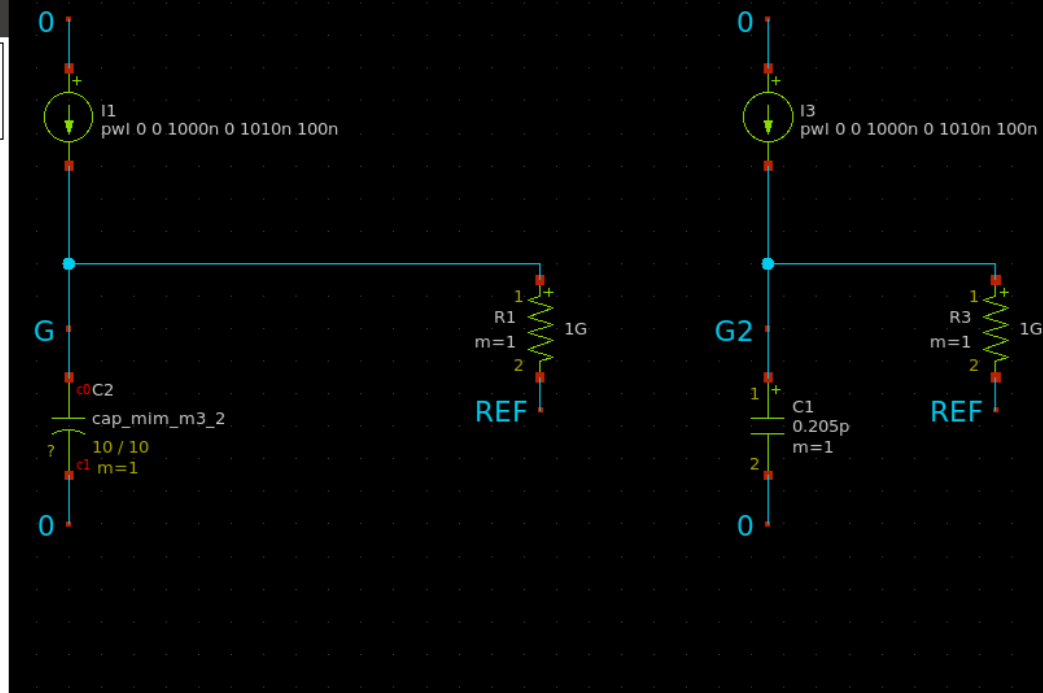
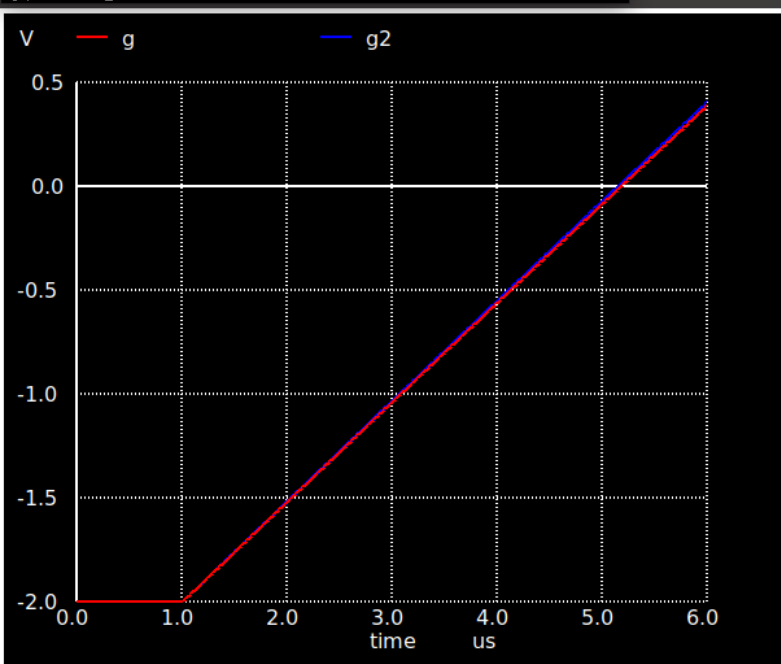
test_mim_cap.spice" -a || sh xschem - test_mim_cap.sch

Initial Transient Solution

| Node | Voltage |
|-------------------------------------|-------------|
| g | -2 |
| ref | -2 |
| g2 | -2 |
| xc2.a | 0 |
| xc2.b1 | 0 |
| sky130_fd_pr__esd_nfet_05v0_nvt.pm3 | 0 |
| v1#branch | 8.27181e-25 |

Reference value : 0.00000e+00
No. of Data Rows : 614
binary raw file "test_mim_cap.raw"
Doing analysis at TEMP = 27.000000 and TNOM = 27.000000

Warning: include: has no value, DC 0 assumed
Warning: i3: no DC value, transient time 0 value used
Warning: i1: no DC value, transient time 0 value used
ngspice 1 ->



- ★ STROKED_BOX
- ★ STROKED_POLYGON
- ★ TEXT

sky130_fd_pr.g

Physical layout of the circuit components, showing the placement of the capacitor and resistor on a grid. The layout includes a central capacitor area and a resistor area, both surrounded by a grid pattern. The layout is displayed in a window titled 'sky130_fd_pr.g'.

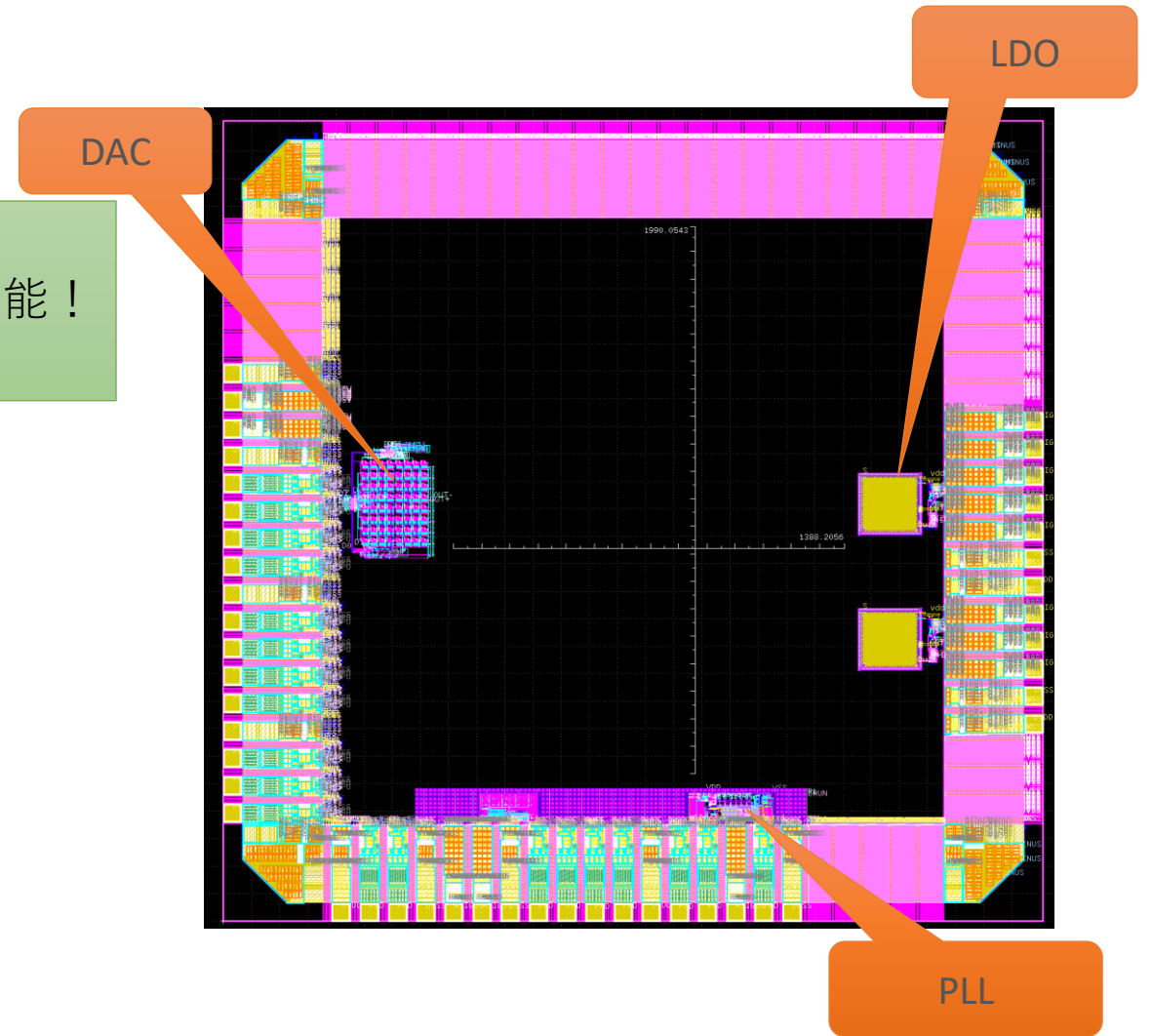
オープンソースPDK対応ファブ

| シャトル | Chiplgnite | iHP | 東海理化 | ミニマルファブ |
|--------------|--------------------------------------|-------------------------------------|--------------------------------------|-------------------------------|
| プロセス | 130nm | 130nm | 600nm | 2000nm |
| サイズ | 10mm ² (3.1mm x 3.2mm) | 2mm ² (1.4mm x 1.4mm) | 10mm ² (3.1mm x 3.2mm) | 直径12.7mm (ウェハとして 提供される) |
| 価格 | \$9750 | 無料 | 約50万円 | 約50万円 |
| 定期シャトル 回数 | 年4回 | 年5回 | 年2回 (試験運用 段階) | 装置が空いていれば いつでも可能 |

シャトル例：
フェニテック社シャトル便
(旧：MakeLSI:シャトル便)

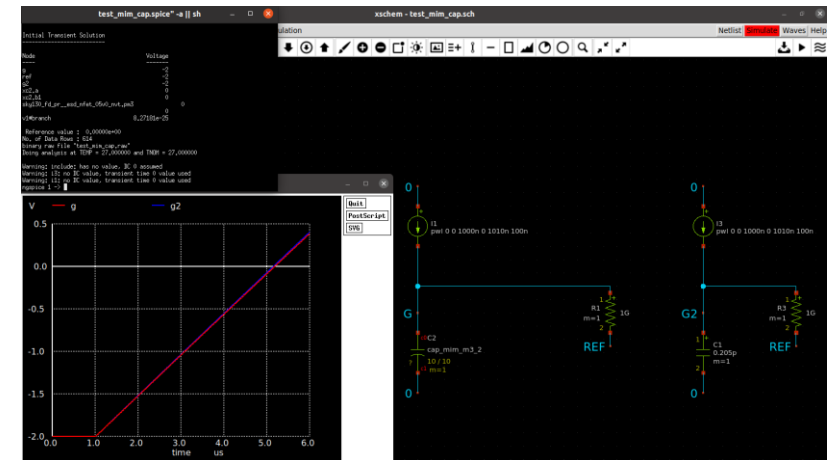
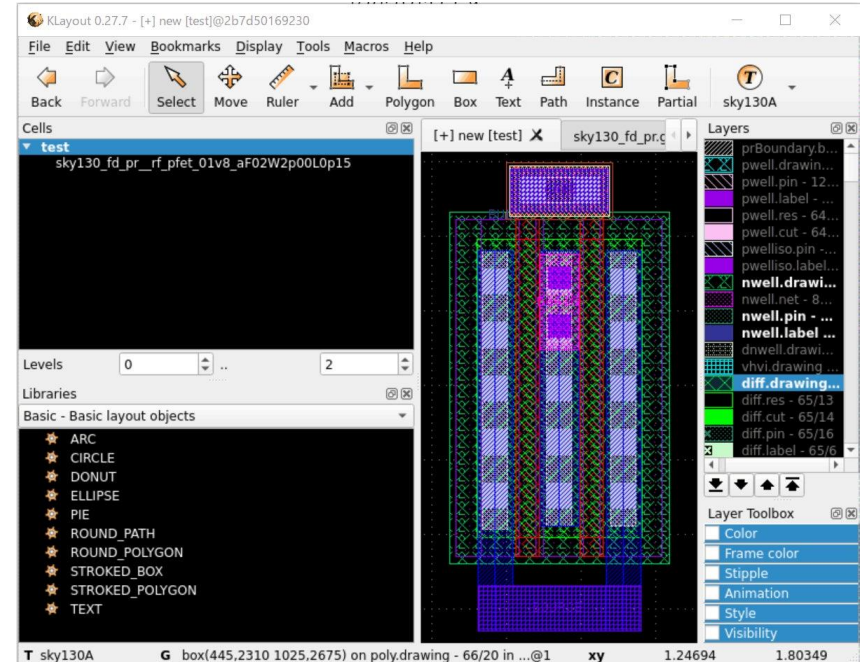
無料で利用可能！

- 1つのダイ (コア) をシェアする
 - 一人だとよほど頑張らないとスカスカ
 - みんなシェアしよう！
- シェア (シャトル) 内容
 - プロセス：600nm
 - エリア：1.8mm x 1.8mm
 - チップ数：20個
 - パッケージ入りとベアダイが選択可能
 - ピン：80ピン
 - オープンソースのEDA&PDK利用
 - ツール類もOSSで完全無料



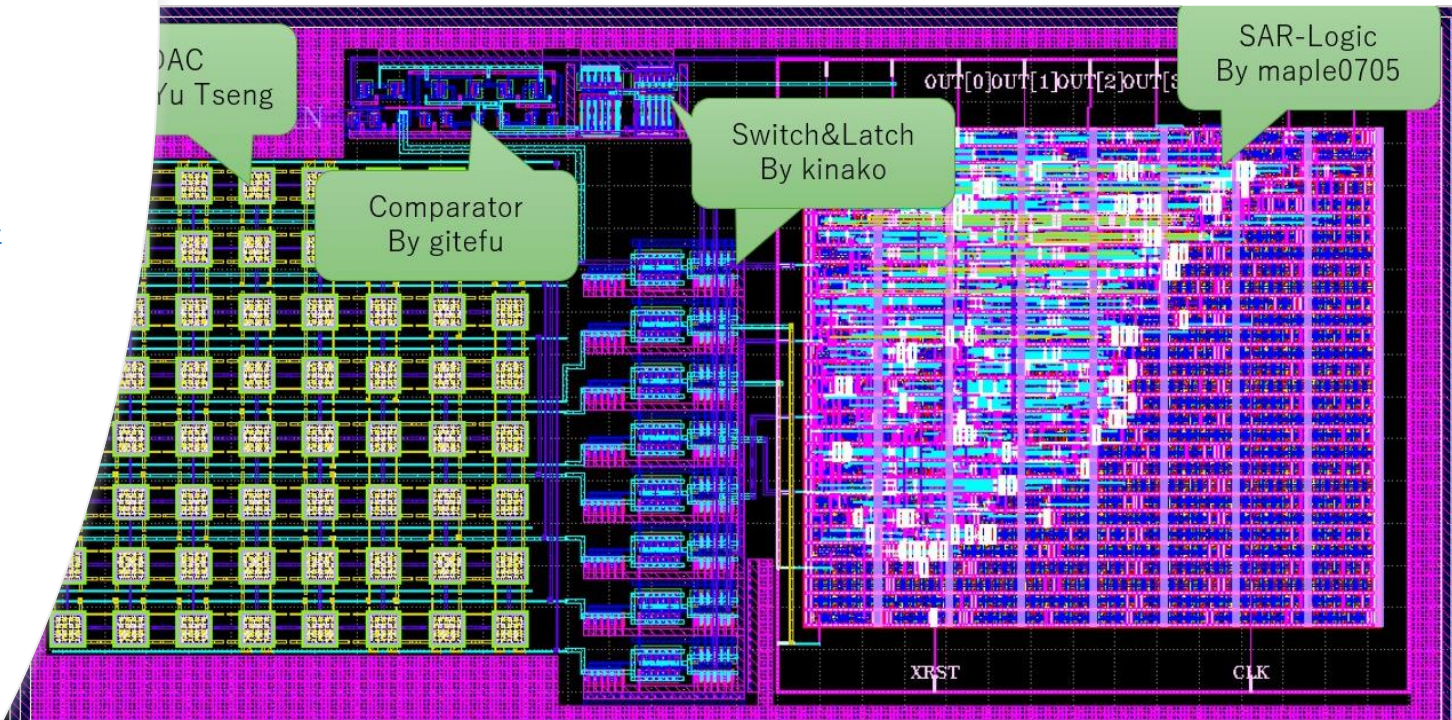
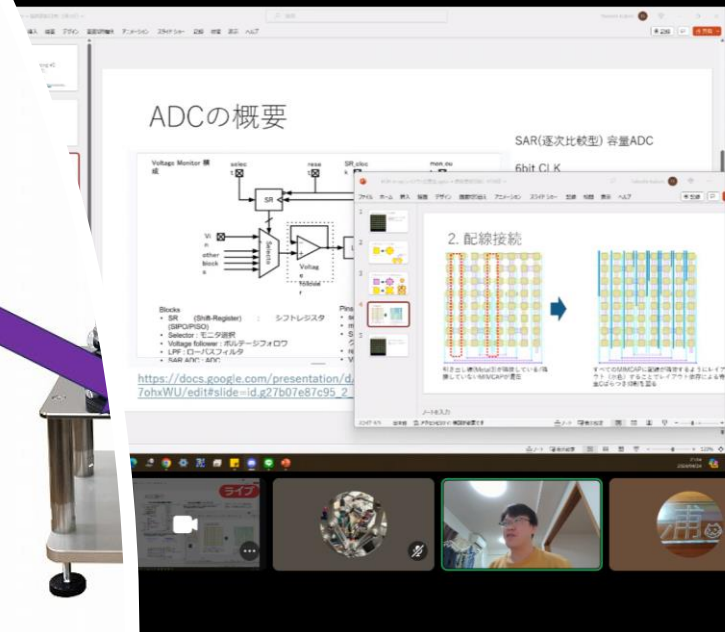
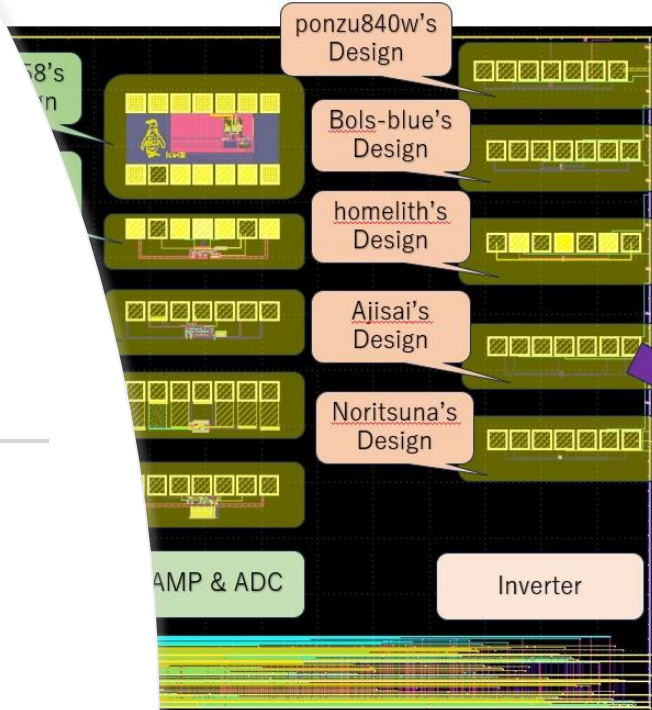
ハンズオンセミナー


- 知識ゼロから半導体設計の基礎がすべて学べる！
一番簡単なインバーター回路のハンズオンセミナー
 - Xschemによる回路設計
 - トランジスタの組み合わせで機能を実現する作業
 - ngspiceによる回路特性シミュレーション
 - 上記の回路が正しく動作するかを検証する作業
 - klayoutによる回路デザイン
 - トランジスタを実際の半導体の上に配置する作業
- 丸々1日の講習会となります
 - 講習会実施実績
 - <https://ishikai.connpass.com/event/303102/>
 - https://www.noritsuna.jp/download/ishi_20231110_3zki_ver2.pdf
- 参加者の声（半導体設計未経験者。電子工作をしたことがあるレベル）
 - チップの設計体験によりすごく技術的な刺激を受けた
 - 半導体は全く未知のものだったが理解できたことで新しい知見を得ることが出来た




シャトル： ISHI会シェアの実績

- 2023/12
 - OpenMPW GF-1シャトル
 - https://github.com/ishi-kai/ISHI-KAI_Multiple_Projects_OpenGFMPW-1/
- 2024/05
 - Chipathon2023
 - https://github.com/ishi-kai/Chipathon2023_ADC/tree/main/submit_version
 - <https://github.com/atuchiya/DC23-LTC2/tree/japan-test/TOP>
- 2024/08
 - ISHI会版OpenMPW PTC06-1シャトル
 - https://github.com/ishi-kai/ISHI-KAI_Multiple_Projects_OpenMPW_PTC06-1/





どうしてオープンソース
半導体という流れなのか？



オープンソース半導体 ～OpenMPWとは？～

1. オープンソースの設計ツール(OpenEDA)にて設計。設計環境やスクリプトを公開することが可能であること、第三者による検証・改良・複製により、コミュニティにて共有できること。
2. オープンソースのプロセス情報 (OpenPDK)にて設計。設計資産(回路図・GDSII)やソースファイルを公開することが可能であること、第三者による検証・改良・複製により、コミュニティにて共有できること。
3. 上記1、2で設計したオープンソース設計チップを製造するファブ・サービスが存在し、設計したハードウェアの動作を検証できること。



なぜ今オープンソース半導体なのか？

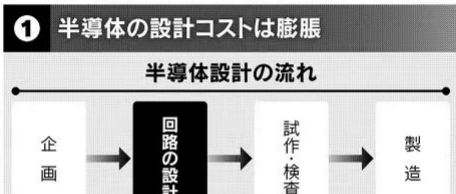
- 半導体設計教育の危機
 - 電気電子課程への進学者の減少。教科書売上減少。VLSI教育にかかるソフトウェアのコスト高。EDAサーバーの保守・更新にかかる経費増等、教育側のコスト負荷が大きすぎる。
- 半導体産業の衰退 = 経済と安全保障において国家的な脅威
 - 偽造電子機器は、数十億ドル規模の闇市場が存在し、米国国防総省が購入する予備電子部品の推定 **15%** が偽造品であり、信頼性と安全性の両方を脅かしていると報告されている
 - →半導体人材育成が喫緊の課題。
- チップ設計者の作業効率向上
 - オープンソース化より、ハードウェア設計は多くの恩恵をオープンソースソフトウェアと同様に受けることができる。

オープンソース半導体・タイムライン

- 2018 : DARPA (国防高等研究計画局) OpenIDEA プログラム
 - \$11.3M grant to UC San Diego for “OpenROAD” project
- 2020 : Google/efabless/SkyWater OpenMPW プログラムスタート
- 2022 : Global Foundries が OpenMPW プログラムに参加
- 2023 : 独) iHP (130nm/SiGe) が PDK のオープン化を宣言
 - Free Silicon Foundation (FSI) が、欧州の半導体産業の競争力、革新性、教育、
独立性、サイバー耐性、環境持続可能性などに貢献できると主張
- 2023 : Open PDKの管理を Chips Alliance がサポート
- 2024 : 各ファブがOpenPDKよるシャトルを開始

半導体設計、オープン化の波

半導体の設計に、無償で一般公開された「オープンソース」が活用され始めた。高度化によるコスト増や技術者不足などの構造問題の解決に向けて、誰でもアクセスできるツール群を使おうという試みだ。産業技術総合研究所(産総研)や米グーグルは利用環境の整備に動く。オープン規格を採用する企業も増えている。



産総研や

4月、半導体産業のオープン化を掲げる団体が日本を充足した。産総研が主導するAIST Solutions(アイストソリューション)が設立した「OpenSUS」に、設計に必要なツール群を利用できる環境を整備し、国内の半導体工場に製造を委託するサービスに引き上げる。半導体の設計にはこれまでおろそかであった「試作・検査」も、半導体工場に製造を委託するサービスに引き上げる。半導体の設計にはこれまでおろそかであった「試作・検査」も、半導体工場に製造を委託するサービスに引き上げる。



• 日本での動き

- 2023年：ロジックリサーチ社の主催でオープンソースEDAフォーラムが開催される
 - 2023年6月：第三回よりハイブリット開催
 - 2024年7月：オープンソースEDA研究会として始動
- 2023年5月：滋賀県立大学の土谷先生主導でコミュニティ：ISHI会が発足
 - 2023年12月：ISHI会主導でOpenMPW GF-1にグループ投稿
 - 2023-2024年：IEEE SSCSのChipathon2023に土谷先生や熊本大学の久保木先生がリーダーとなって、日本チームを結成して投稿
 - 2024年5月：2014年より金沢大学の秋田先生が主導してたMakeLSI:がISHI会に合流して、統合された
- 2024年5月：産総研主導で産業界団体：OpenSUSIが発足
 - <https://www.nikkei.com/article/DGXZQOUC228690S4A420C2000000/>



どう動作するかを定義するチップを二つ。半導体チップを定義する「命令セット」がある。これに沿って、区画ごとに「IP」が作られて、半導体工場に渡り、IPが市場を握る。設計の自由度を高め、コストを削減し、競争力を高める。命令セットに注目。AIST Solutions(アイストソリューション)が設立した「OpenSUS」に、設計に必要なツール群を利用できる環境を整備し、国内の半導体工場に製造を委託するサービスに引き上げる。半導体の設計にはこれまでおろそかであった「試作・検査」も、半導体工場に製造を委託するサービスに引き上げる。

5 ソフトの世界では成功

オープンソースが勝つ

GitHub
技術者同士がソフトのソースコードを共有

リナックス
自由に改良できる基本ソフト(OS)。スマホのOS、アンドロイド開発などの基礎に

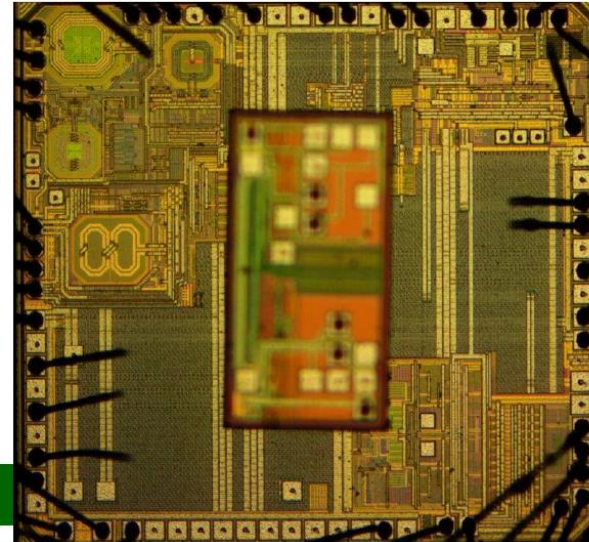
グラフィックス 池田奈美

中国がアツい！

- 中国には3000社以上の半導体メーカー（大半はファブレス＝設計専業）最先端SoCから安価な互換IC、ディスクリット半導体まで、企業の幅は非常に広い
- 世界中から中古の半導体製造装置を買い漁っている！

PCB基板製造で起きた流れが来ている！

☑少量多品種のSoC/SiP市場が成立している



一般に半導体は初期コストが非常に高いので少量多品種に向かない産業

Bluetoothイヤホン専用SoC
空間オーディオDSP、LiPo充電、
タッチ検出など必要機能が一式
(65nmプロセス、別フラッシュのSiP)

写真提供: 高須正和氏

zawa University <http://ifdl.jp/>

180 Attendees!! Record attendance among all workshops at VLSI Symposium

Open Source PDKs and EDAs, Community Experiences toward Democratization of Chip Design

Organizer : Makoto Ikeda (The University of Tokyo)

Organizer : Mehdi Saligane (University of Michigan)

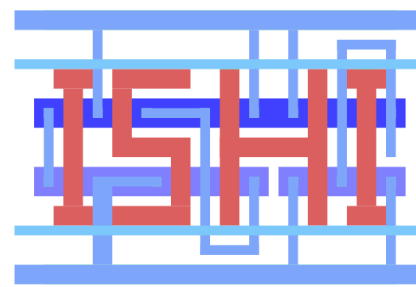
Since its launch in 2020, the Open MPW shuttle program has received over 500 designers' experiences, including measured results, foundry perspectives, and

About Makoto Ikeda

Makoto Ikeda received his BE, ME, and Ph.D. degrees all in EE department of d.lab, the University of Tokyo. This workshop is co-organized with Dr. Mehdi S

- | 1. Design experience: "The Journey of Two Novice LSI Enthusiasts: T Communications and Yuki Azuma, University of Tsukuba
- | 2. From Zero to 1000 Open Source Custom Designs in Two Years, Mo
- | 3. The SKY130 Open Source PDK: Building an Open Source Innovati
- | 4. Open Source Chip Design on GF180MCU – A foundry perspective, Karthik Chandrasekaran, Global





ここまで、理解できましたか？

- 初心者でもこれらを理解して、半導体を作れるようにサポートするのがISHI会です。
- 初心者向け単語集
- ワンタッチビルド環境
- サポート勉強会
 - ホームページ
 - <https://ishi-kai.org/>
 - Discord上で活動中
 - <https://discord.gg/RwAWF5mZSR>



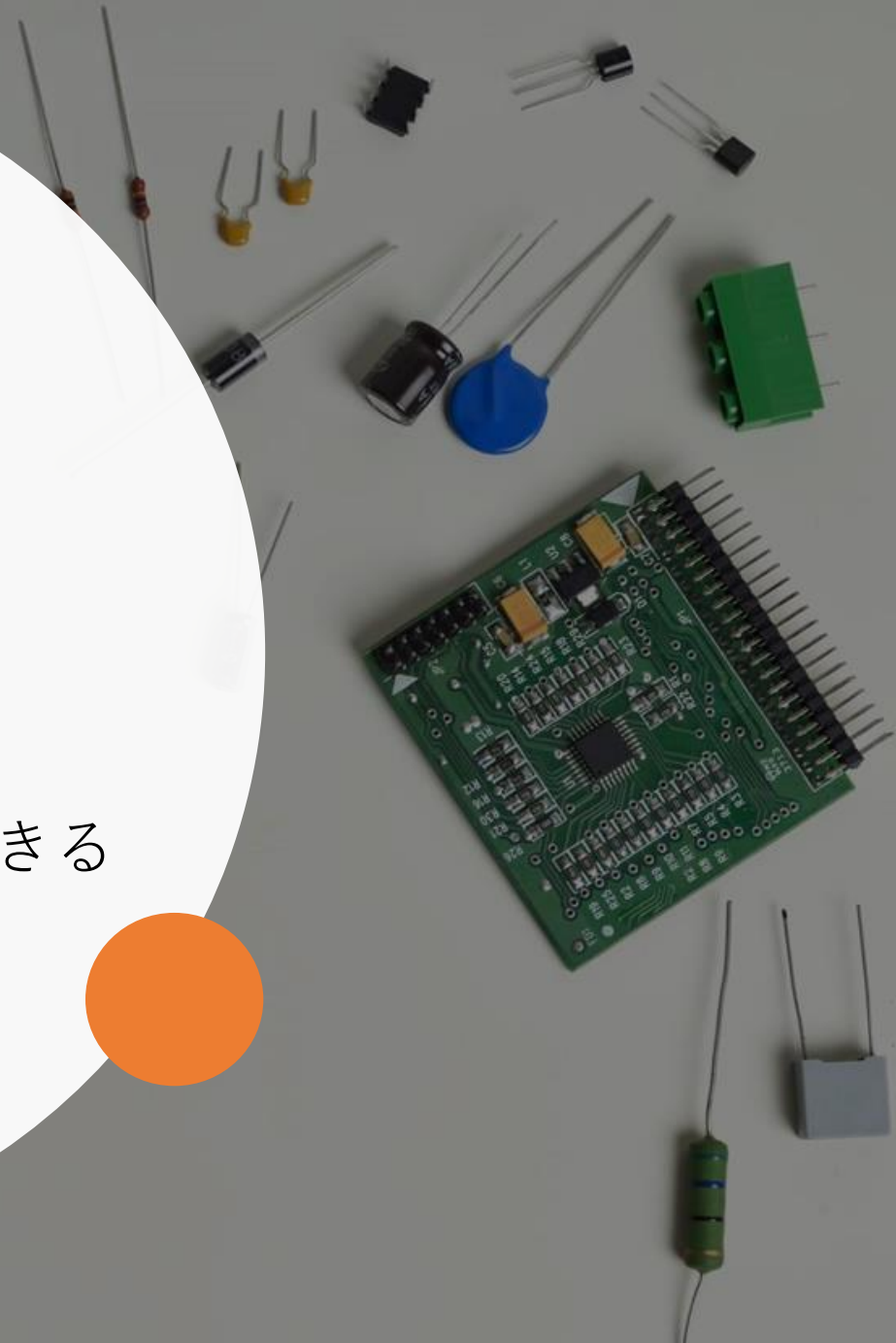


初心者向け教材：デジタル



デジタル とは？

VerilogやHDLで設計・開発できる
回路



必要な知識は？

- コンピュータサイエンス
 - デジタル回路
 - VerilogやHDLの言語知識
 - コンピュータアーキテクチャ



VLSI.JP

[To Article Index](#)

- [無から始める自作CPU](#)
 - [必要な物](#)
 - [デジタル回路とVerilog入門](#)
 - [コンピュータアーキテクチャ入門](#)
 - [この次へ](#)
 - [RISC-V CPUを作る](#)
 - [半導体を作る](#)
 - [コンパイラを作る](#)
 - [OSを作る](#)
- [謝辞](#)



[クレイジーピエロ](#) 著

無から始める自作CPU

CPUは作れる！！！！！！！！ご存知でしたか！！！！！！！！

CPU、それは我々が暮らす情報社会の基盤となる魔法の石です。

世に存在する全てのソフトウェア、例えばゲーム、AI、Webサーバ、OS、これらは全てCPUが無ければ動きませんし、今や車や飛行機、家電にも全てCPUが入っている時代です。

そんな誰もがCPUに依存している時代にも関わらず、CPUについて理解を持っている人間は余りにも僅か、というのが現状です。

そんな今こそCPUを作りましょう。

CPUを作り、完全に理解する事で、CPUによって成り立つ技術を学ぶ上での、揺るぎない自信と確証を身につける事が出来るでしょう。

本記事ではCPUという究極のブラックボックスに光を当て、半導体やプログラミングの知識が無の状態から、CPUを作る事を目標としています。

- [デジタル回路とVerilog入門](#)
 - [基礎知識](#)
 - [CPU](#)
 - [二進数と16進数](#)
 - [二進数](#)
 - [16進数](#)
 - [二進数の負の数](#)
 - [デジタル回路](#)
 - [デジタル回路が扱う値](#)
 - [NOT](#)
 - [OR](#)
 - [AND](#)
 - [NAND](#)
 - [XOR](#)
 - [MUX](#)
 - [HalfAdder](#)
 - [FullAdder](#)
 - [D-FF](#)
 - [MUXによるD-FFの改良](#)
 - [FPGA](#)
 - [Verilog_HDL入門](#)
 - [開発の流れ](#)
 - [開発環境構築](#)
 - [テキストエディタのインストール](#)
 - [Verilog_HDLシミュレータのインストール](#)
 - [開発環境に慣れる](#)



デジタル回路とVerilog入門

デジタル回路とVerilog入門では、CPUを作る前に必要な基礎知識、そして作るために必要な道具の使い方を学んでいきます。

基礎知識

ここではCPUを作るのに必要な知識を説明します。覚える必要はありません。

CPU

CPU、我々が作る対象です。CPUとは一体なんでしょうか？概要すら知らないのに作ろうとするのは流石に無謀と言えます。ちょっとだけ先に知っておきましょう。

プログラミングという単語は皆さん人生のどこかで聞いたことがあるでしょう。最近の中高生はプログラミングの授業があるんですかね、気の毒ですね。プログラミング、プログラムを書いてゲームを作ったりモーターを動かしたりするアレですね。あなたがこの記事を読んでいるSafariやChromeもプログラムですし、YoutubeもTwitterもInstagramもプログラムです。ああ素晴らしきかなプログラム。プログラムが無ければお前は生きてはいけません。

```
#include <stdio.h>
int main(){
    printf("Hello World!\n");
    return 0;
}
```

- [コンピュータアーキテクチャ入門](#)
 - [プログラムが動く流れ](#)
 - [命令セットアーキテクチャ](#)
 - [Z16の概要](#)
 - [Z16のレジスタ](#)
 - [Z16の命令](#)
 - [演算命令](#)
 - [ADD](#)
 - [SUB](#)
 - [MUL](#)
 - [DIV](#)
 - [OR](#)
 - [AND](#)
 - [XOR](#)
 - [SLL](#)
 - [SRL](#)
 - [演算命令まとめ](#)
 - [即値命令](#)
 - [ADDI](#)
 - [即値命令まとめ](#)
 - [メモリ命令](#)
 - [LOAD](#)
 - [STORE](#)
 - [メモリ命令まとめ](#)
 - [ジャンプ命令](#)
 - [JAL](#)
 - [JRL](#)
 - [ジャンプ命令まとめ](#)
 - [分岐命令](#)
 - [BEQ](#)

コンピュータアーキテクチャ入門

コンピュータアーキテクチャ入門

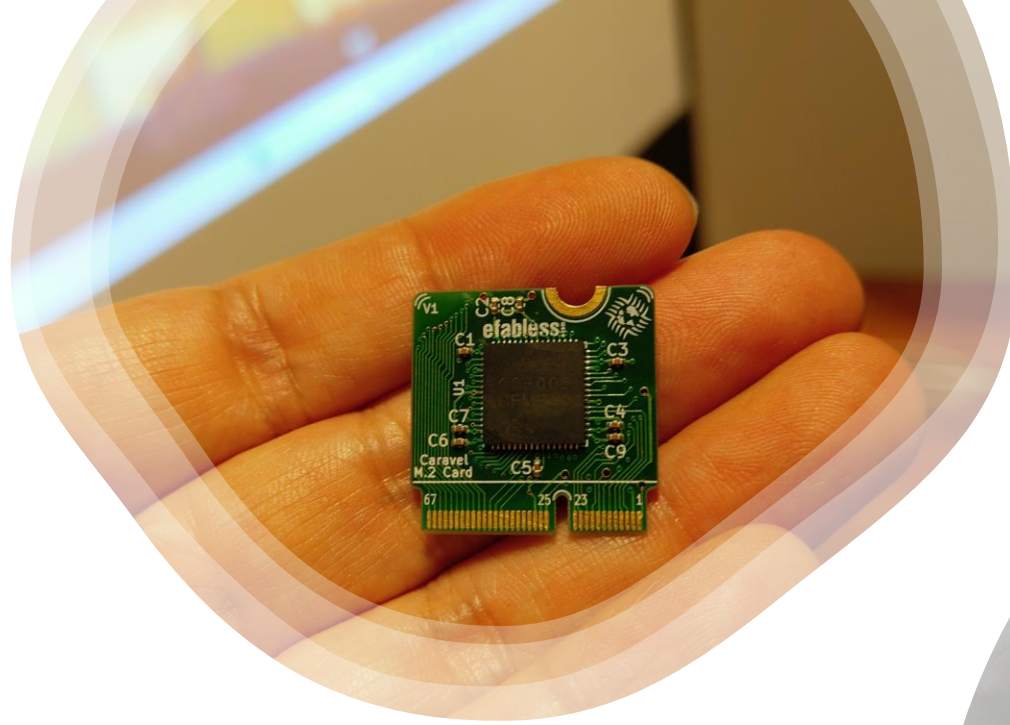
さあ始めましょう

プログラムが動く流れ

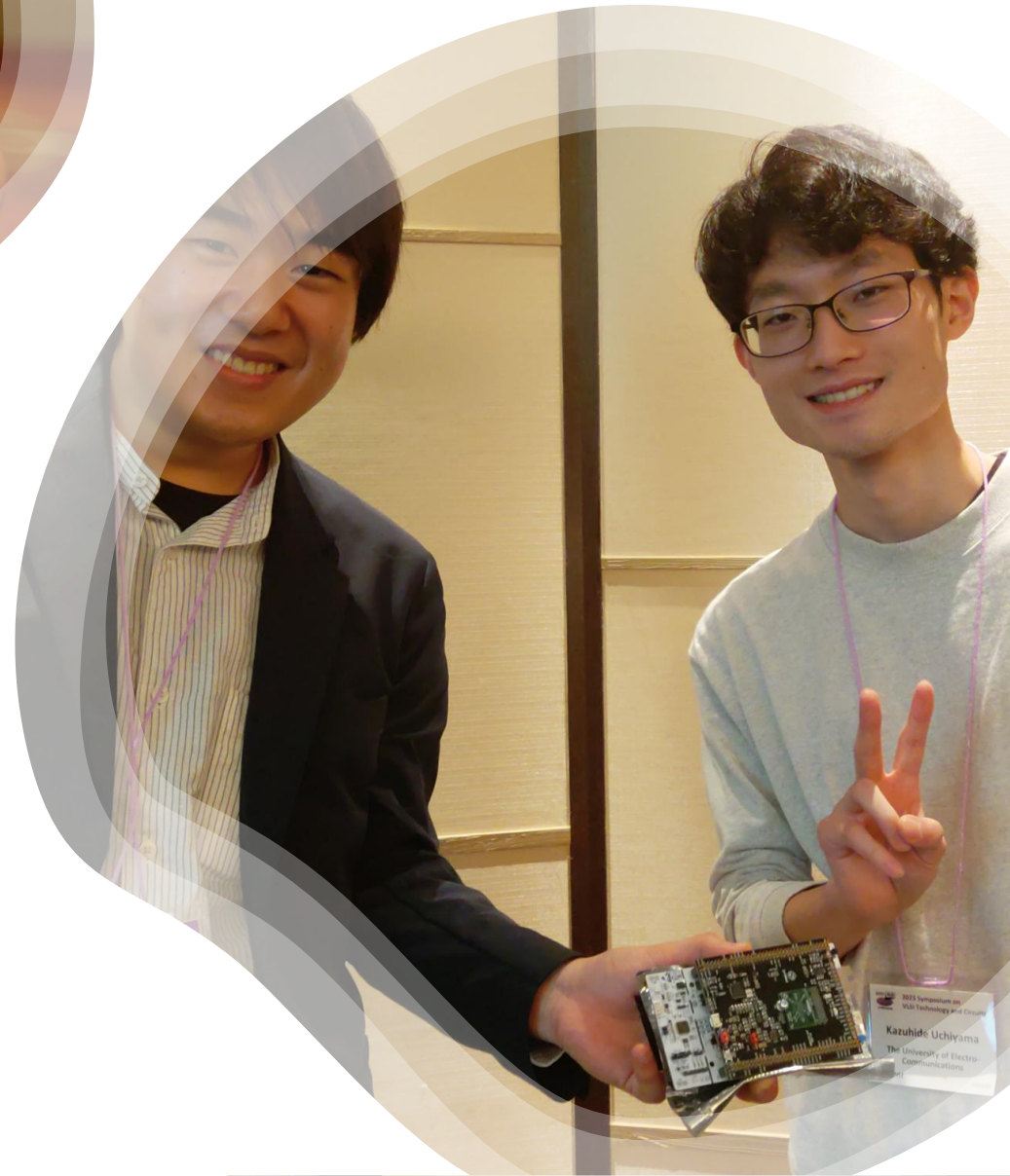
CPUはプログラムをどのように実行しているのでしょうか？CPUはプログラムを動かす物体ですので、一度ここで学んでおきましょう。



CPUはプログラムをそのまま実行している訳ではありません。



8bit CPU





初心者向け教材：アナログ

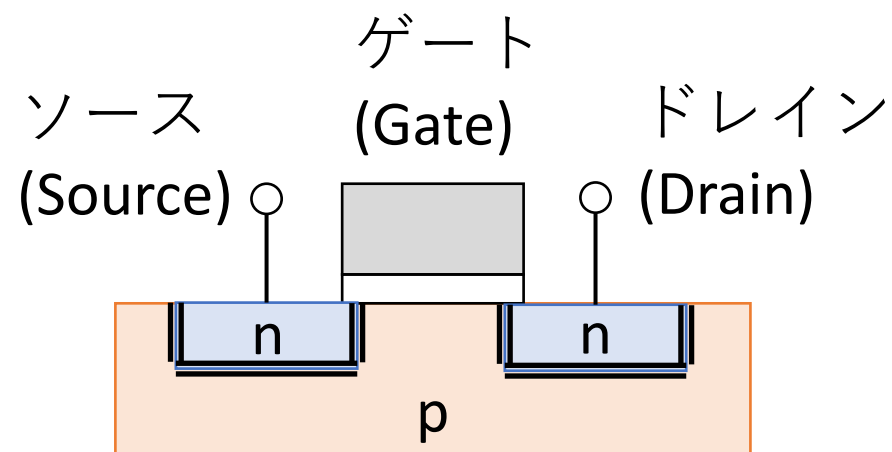


アナログ とは？

RとCとLとトランジスタを
組み合わせて作る回路

必要な知識は？

- トランジスタ設計



画面構成 (Simulation)

SILIWIZ

右ペインの
切り替えボタン

レイヤー操作

Layers

active

- p substrate
- n well
- n diffusion
- p diffusion
- p tap
- n tap

passive

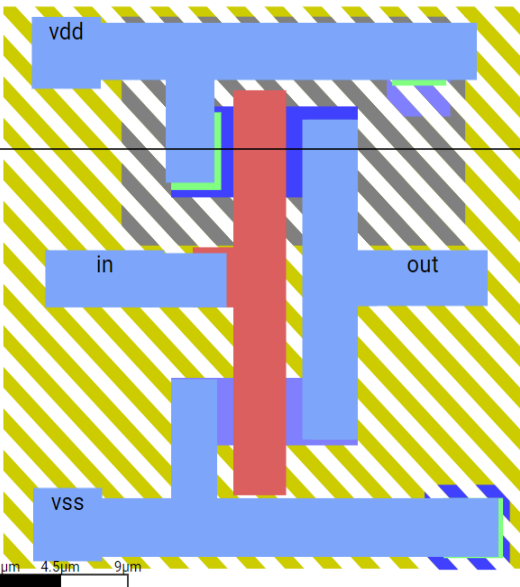
- polysilicon
- polyres
- metal1
- mim capacitor
- metal2

via

- metal1 via
- metal2 via

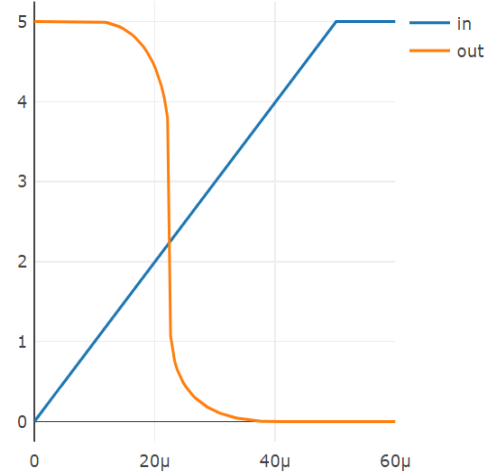
Preset

LOAD SAVE CLEAR STL



描画

CROSS SECTION & DRC ✓ SIMULATION



Plot signals:

in × out × +

Input voltage:

Min: 0V

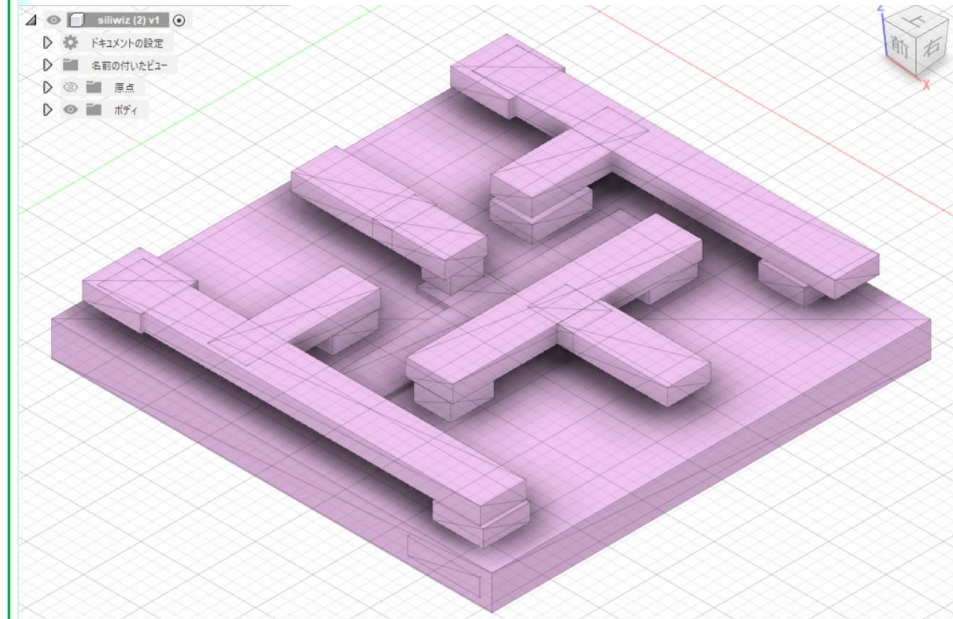
Max: 5V

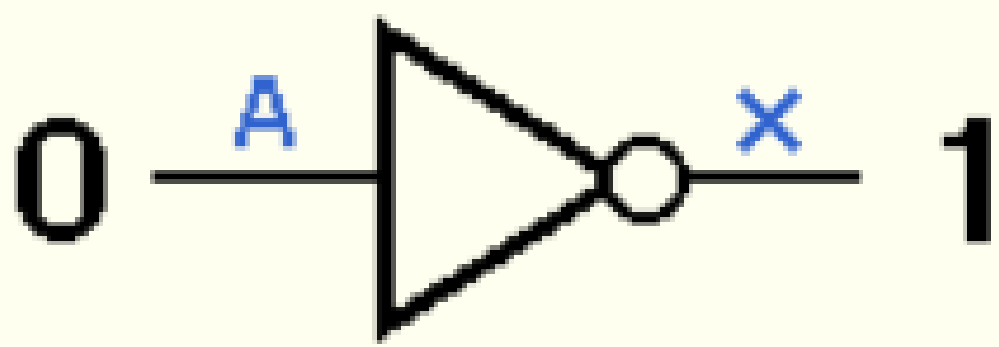
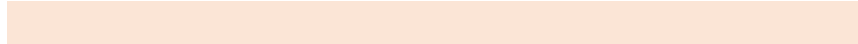
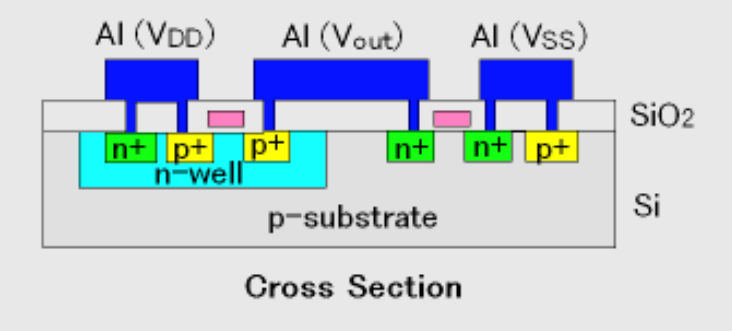
Pulse delay: 0μs

Rise time: 50μs

Time scale: 60μs

Show SPICE (advanced)

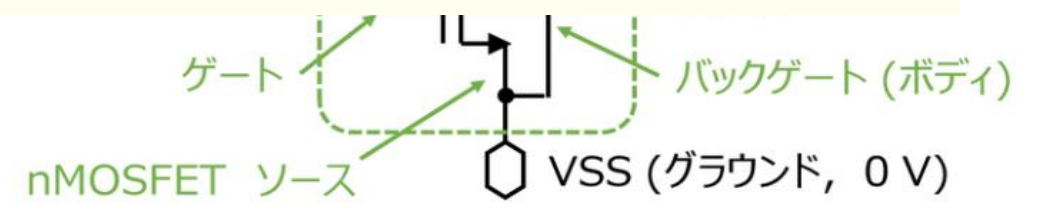




真理値表

| | |
|----------|----------|
| A | X |
| 0 | 1 |
| 1 | 0 |

イ)
ロ)



CMOSインバータの回路図

画面構成 (Simulation)

SILIWIZ

右ペインの
切り替えボタン

Layers

active

- p substrate
- n well
- n diffusion
- p diffusion
- p tap
- n tap

passive

- polysilicon
- polyres
- metal1
- mim capacitor
- metal2

via

- metal1 via
- metal2 via

Preset

LOAD SAVE CLEAR STL

CROSS SECTION & DRC ✓ SIMULATION



描画

Input voltage:

Min: 0V

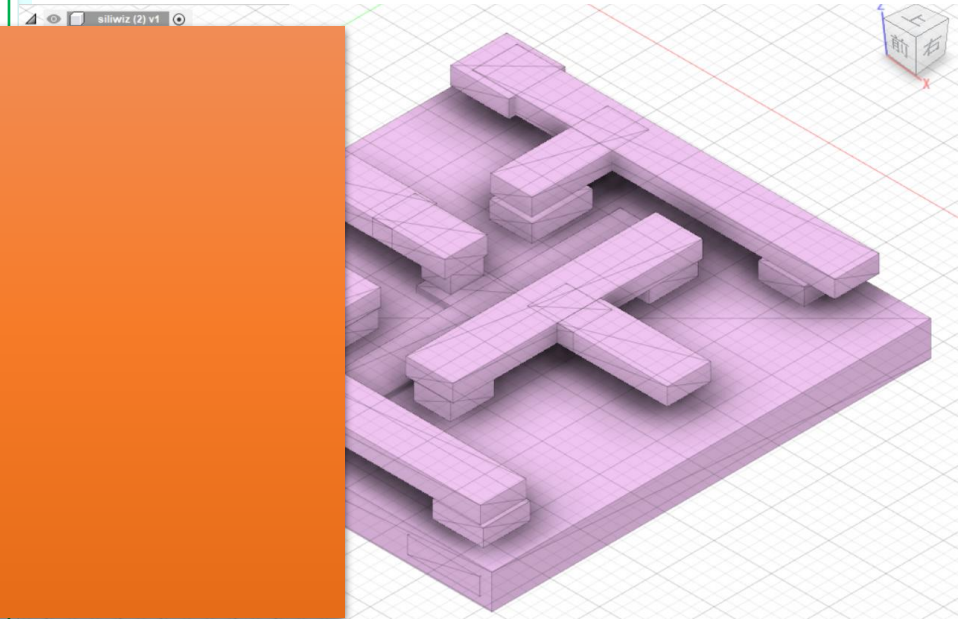
Max: 5V

Pulse delay: 0μs

Rise time: 50μs

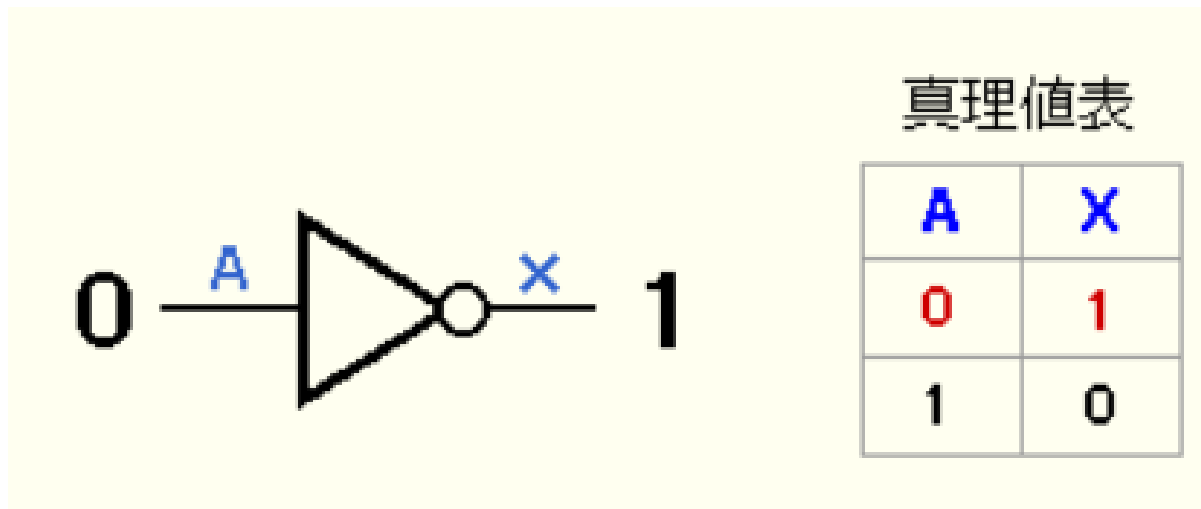
Time scale: 60μs

Show SPICE (advanced)

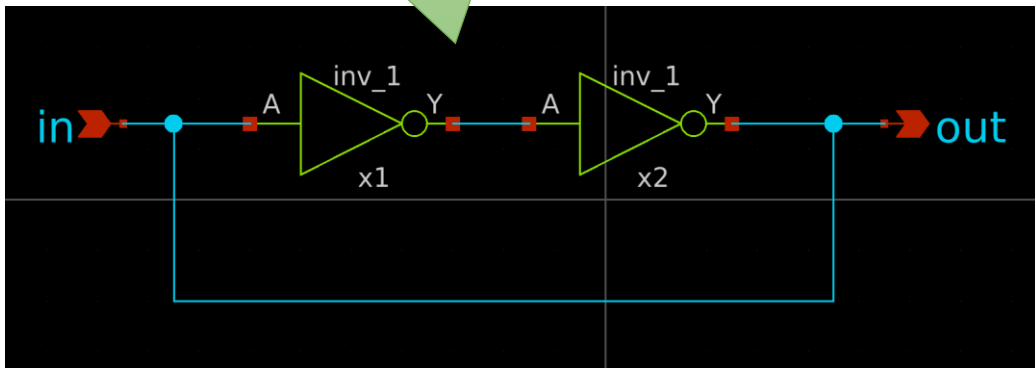


レイヤー操作

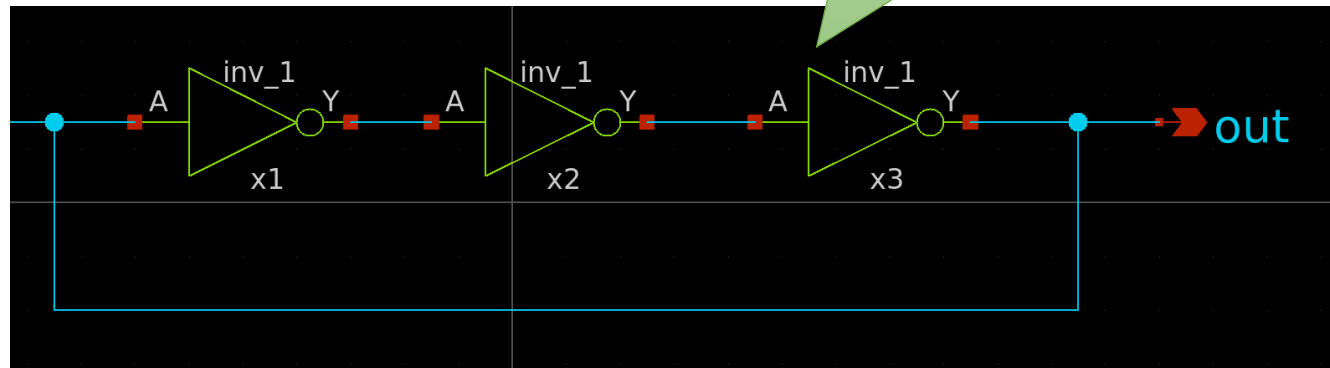
でも・・・基本なんです！
～デジタル回路として～




SRAM 1bitの回路




クロック生成回路



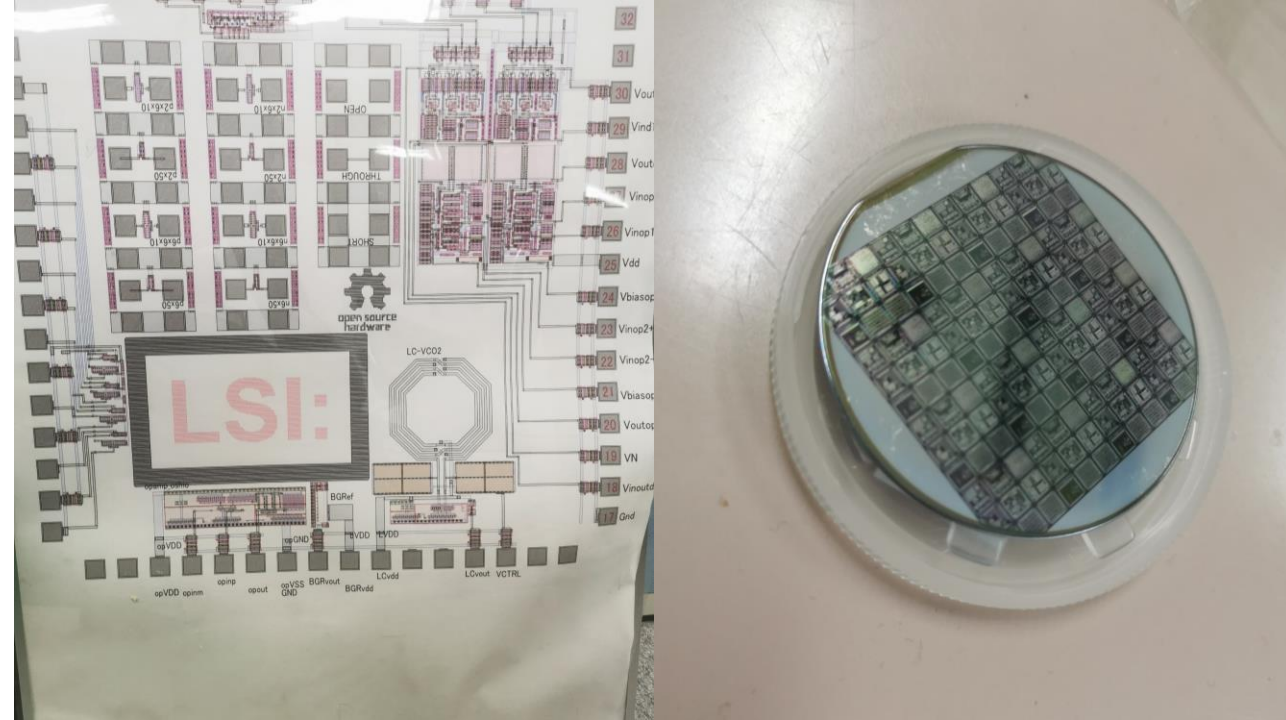


国内コミュニティの紹介



Make LSI:

- 最古のオープンソースPDKプロジェクトの一つ
 - <https://scrapbox.io/makelsi/>
 - OpenRule1umPDK
 - <https://github.com/MakeLSI/OpenRule1um>



OpenRule1um

これは何

- NDA（秘密保持契約）が不要の1um設計ルール(3層メタル)と、それに関連するファイルです。
- フェニテックセミコンダクタ(PTS)のCMOS 0.6umプロセスで製造可能とする予定です。
- グリッド単位を0.5umとし、その整数倍で設計ルールを定めています。（そのためにPTSで製造可能な設計ルールぎりぎりよりは、多少大きくなりますが、その代わりに、NDAが不要となります）
- コンタクトとVIAは、接続対象のML1等のレイヤの図形を含む以下のような指定したセルのみを使用可能です。単体でコンタクトやVIAの図形を描画するのは不可です。なおOpenRule1um.xlsxでは、それぞれのサイズが0.5umの整数倍ではないため、0.5umの整数倍となるように少し拡大しています。また各セル内に描画してあるコンタクトやVIAのサイズは実寸ではなく製造時には変更されます（なのでこのセルの中身をいじったり、単体でコンタクトやVIAを描画してはいけません）。またDRC（や回路抽出）は、これらのセルの外形を示すダミーレイヤを用いて行われます。※ダミーセルを実体セルに置き換えるとき、外形のメタル配線のサイズが保持されます。
- これらのダミーセルを含めた配線は、基本的にPath（幅を1um以上に設定）を使ってください。（長方形で描かない）Pathの中心線の端が、各ダミーセルの中央まで至るように配線をしてください。

半導体初学者むけ「半導体設計で使われる用語集」

※ ぼちぼち更新しています。追加して欲しい用語・略語等あれば連絡ください。

【おすすめ動画】

【京都賞記念講演】カーヴァー・ミード「情報革命の時代を生きて」

Carver Andress Mead教授の講演。EDA（集積回路設計ソフト）の生みの親です。チップ設計の歴史を俯瞰して理解することができます。

【A - F】

Analog Hard IP (Design Knowledge)

- **AFE** : Analog Front-End の略。センサー等のアナログ出力信号をデジタル信号へ変換するアナログ回路を指す。一般には、アンプやA/Dコンバータ、フィルタなどのアナログ回路を指すことが多いが、高速シリアル通信向けのクロックが重畳された小振幅信号やPAM(Pulse Amplitude Modulation)信号に変調された信号をデジタル信号に復調するミックスドシグナル回路もAFEと呼ぶことがある。SoC内部にAFEを搭載する場合と、SoCとは別チップでAFEを構成する場合がある。

VLSI.JP

[To Article Index](#)

- [OpenMPW入門 改訂版](#)
 - [OpenMPWとは](#)
 - [宣伝](#)
 - [環境構築](#)
 - [OpenLANEの概要](#)
 - [OpenLANEのインストール](#)
 - [OpenLANEの動作確認](#)
 - [OpenLANEの設定](#)
 - [Caravelの概要](#)
 - [Caravelのインストール](#)
 - [Caravelの動作確認](#)
 - [Caravelのファイル構造](#)
 - [Caravelにおけるデザインのビルド](#)
 - [Caravelのドキュメント](#)
 - [OpenMPWで自分のデザインを焼こう！](#)
 - [1. 焼きたいデザインを用意する](#)
 - [2. 自分のデザインにCaravel用のインターフェースを生やす](#)
 - [WishboneとLogic Analyzerの違い](#)
 - [Wishbone interfaceを作る](#)
 - [Logic Analyzerを扱う](#)
 - [mprij_ioを扱う](#)
 - [割り込みの扱い方](#)
 - [3. 自分のデザインをGDSIIにする](#)
 - [設定ファイルを書く](#)

OpenMPW入門 改訂版

質問、修正案、その他連絡は@Cra2yPierr0tマデ

久々に以前書いた入門記事を読んだら日本語は雑だし内容は古いしこの世の終わりみたいな出来だったので90nmに備えて書きます。

オレオレLSIを焼きたいですよね？ 焼きましょう。Skywater社がPDKを公開し、OSSなGDSIIコンパイラであるOpenLANEも生まれました。そしてGoogleの出資によってEfablessが無料でLSIを作らせてくれるプログラム、Open MPW Shuttle Programをスタートしました。今こそオレオレLSIを焼くチャンスです。あなたの作りたいチップをGoogleの金で作りましょう！

OpenMPWとは

OpenMPW(Open Multi Project Wafer)はEfablessにGoogleが出資して生まれたシャトルプログラムであり、ホームページには次の文言が書かれています。

The shuttle provides opportunities for designers to experiment and push the state-of-the-art without having to reconcile the risk associated with the cost of fabrication. The shuttle program is open to anyone, provided that their project is fully open source and meets the other program requirements. Costs for fabrication, packaging, evaluation boards and shipping are covered by Google for this program.

出典：https://efabless.com/open_shuttle_program

このシャトルはデザイナーに製造コストに纏わるリスクを負うことなく、実験し、最先端を追求する機会を提供します。シャトルプログラムは、プロジェクトが完全にオープンソースであり、一定の要件を満たしていれば、誰でも参加することができます。製造、パッケージング、評価ボード、そして送料は全てGoogleが負担します。

つまりデザインをオープンソースにすれば**完全無料**で自分の半導体を作れるプログラムという事ですね。これは熱い、参加するしかない、Google最高一生ついていきます。

と言っても提出された全てのデザインを焼いてくれるという訳ではなく、いくつかの条件が存在しています。

VLSI.jp

日本で最初にOpenMPWのシャトルに乗った人物によるOpenMPWの解説サイト & コミュニティー

ISHI会グラウンドデザイン

新規分野を開拓したいけどどうすればよいのかわからない

「みんなの経験をチップに！」

ASIC(LSI)化したいけど情報がない

ASIC(LSI)業界の現状（閉塞感）

- NDAでなにもしゃべれない
- 最先端は札束の応酬
 - 若者が入ってこない

他業界の現状（限界感）

- 高速・小型・省電力の要求
 - 汎用チップ+ソフトでは限

すべてがオープン！

OpenMPWの登場！

コミュニティの意義

- 成果の再利用が可能。Do It With Others(それ、みんなで作ってみよう)の精神
- 日本の利点：地理的に物理的に集まりやすく、勉強会や合宿をやりやすい

ISHI会の意義

- 他（多）分野の知識の統合により、今までになかった研究・開発への期待



今後のイベント紹介



現在進行中

- Chipathon 2024日本チーム
 - 開発中
 - 定例もくもく会（木曜日21時～）
- Discord上
 - TinyTapeout投稿
 - 11月に出す
 - iHP投稿
 - 3月に出す
 - 初心者向け勉強会
 - 9月21日
 - 10月予定

<https://ishi-kai.org/links/>



ご清聴ありがとうございました



- ホームページ
 - Github pages
 - <https://ishi-kai.org/>
- Discord上で活動中
 - <https://discord.gg/RwAWF5mZSR>
- イベント告知（勉強会など）
 - connpass
 - <https://ishikai.connpass.com/>

