

# 오픈 - 소스 도구 활용 “내 칩 MPW” 지원하기

국일호 , 경희대학교

ETRI 0.5u CMOS 표준셀 디자인 키트의 제작과 활용

대한전자공학회 하계종합학술대회  
“내 칩 MPW” 특별세션  
2024년 6월 26일

# 약력

성명 국일호

소개 양평에 살며 텃밭과 꽃밭을 가꾸는 반도체 설계 취미가

연락처 [goodkook@gmail.com](mailto:goodkook@gmail.com)

학력

경희대학교 문리대 물리학과 학사  
경희대학교 대학원 공학석사 / 박사

경력

반도체 IP 설계회사 운영

8- 비트 CPU

16- 비트 DSP( 오디오 압축 디코딩 F/W),

MP3 디코더 , 기타 컨트롤러 IP 설계 판매

칩 설계 / 제작 / 검사 대행 /FPGA 프로토타이핑 /PSCE-MI 개발

항공기 시뮬레이터 / 시현 계기 (ARM RTOS)

강의경력

경희대학교 대학원 SoC 설계 방법론 강의  
( 현 ) 경희대학교 산업중점교원 / 반도체 전공트랙



Home  
News  
Products  
Services  
Download  
-----  
About us  
Links  
-----  
SystemC  
-----  
Contact E-mail:  
[goodkook@codecell.com](mailto:goodkook@codecell.com)

Korean Service Only  
SoC & IP News Letter  
Bigcloud's Q&A  
CODECSIL  
<http://www.codecell.com>

## PSCE IP Design & Verification System

PSCE (Poorman's Standard Hardware/Software Co-Emulation) Design System is complete IP Design and verification system, made up of multi-million gates level FPGA board and high-level system modeling APIs for hardware and software co-emulation. The system supports interfacing with HDL (VHDL and/or VerilogHDL), C/C++ model and hardware emulation FPGA board for synthesized IP is running. Verify your IP at higher abstracted system-level environment with accelerated simulation speed.

### PSCE Presentation

AnsLab provides PSCE Design System for **FREE** of its use. [Download](#) PSCE System Software (Sorry, download is not available now: comming soon...).

### PSCE Design System

PSCE Design System is complete IP Design and verification system, made up of:

- **PSCE\_BRIDGE**  
Modeling-Interface for Co-Simulation
- **PSCE API**  
Hardware/Software Co-Emulation API
- **PSCE FPGA Hardware**  
Hardware Protocol to interact Vendor independent Multi-million gates level FPGA

# 내 칩 MPW 서비스 공고문의 변화

- 2023년 공고문  
“0.5u Analog Si CMOS ...”  
디지털은 안되고 ?



- 2024년 공고문  
“0.5u Si CMOS ...”
- ⇒ 풀 – 커스텀 (Full-Custom) 레이아웃 GDS 제출

# “내 칩 MPW” 지원서 작성

## 1. 사용 가능한 틀

2. 연계 교육 프로그램
3. 회로 설계 내용

## 4. 평가 / 검증 계획

# “내 칩 MPW” 지원서

## 1. 사용 가능한 툴

### 상용 또는 오픈 - 소스 설계 툴

표준 - 셀 기반 디지털 회로 설계의 **디자인 - 키트** [\*]

- 직접 제작
- 경희대학교 디자인 - 키트 활용

<https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK>

\* 디자인 - 키트 (Design-Kit): 표준 셀 라이브러리 및 설계 자동화 도구 용 테크 파일 (Tech, LEF)

## 4. 검증 / 평가 계획

“검증” 과 “평가” 없는 칩 제작은 **자원 낭비**다.

디지털 회로의 **테스트**의 과제는 **다수**의 입출력 핀

**검증** 및 칩 **테스트** 장비 **자작** [\*] (FPGA 에뮬레이터)

- 칩 테스트 장비 = 논리 분석기 + **신호 생성기**
- 오픈 – 소스 하드웨어 활용 (아두이노와 FPGA 활용)

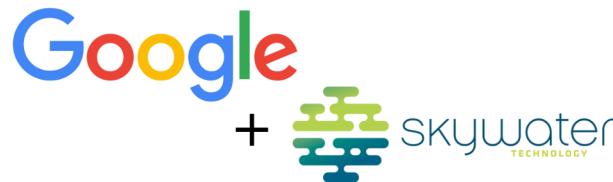
[\*] 목표비용 10 만원 이내 , 25K LUT 급 Gowin FPGA + Arduino DUE

# 반도체 오픈 - 소스 운동

, 갑자기?



오픈 - 소스 운동 : 소프트웨어 > 하드웨어 > 실리콘



**FOSS 130nm Production PDK**  
[github.com/google/skywater-pdk](https://github.com/google/skywater-pdk)

**efabless**.com



Free Silicon Conference 2024

**FSiC2024**  
Free Silicon Conference

**OpenROAD**

Democratizing Hardware Design

The OpenROAD™ project attacks the barriers of Cost, Expertise and Uncertainty (i.e., Risk) that block the feasibility of hardware design in advanced technologies.

[Read More](#)



Custodian of the Free and Open Source Silicon movement.

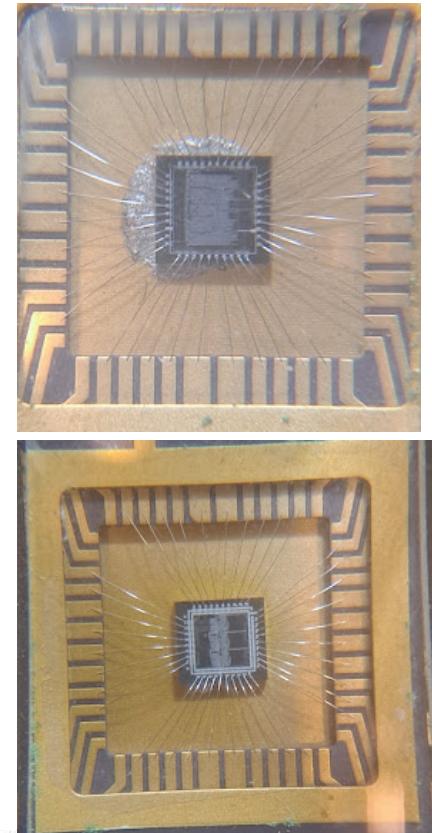
# 30년 전 MPC의 추억

## 반도체공동연구소 (ISRC)

- 1.5um CMOS 1-Poly 2-Metal 풀 – 커스텀 레이아웃
- 40 핀 DIP 패키지
- 사용 툴 : 썬 워크스테이션 , *Construct(?)*, DRC 를 볶

## 지원 당시 ( '92?) 설계환경

- 4000 Tr. 급 설계 , Std-Cell 라이브러리 자작 , 수동 P&R
- 썬 워크스테이션 , **Magic**, SPICE, iRSIM, Altera Max+II
- PC 프린터 포트 활용 칩 테스트 보고서 /100 만원 상금 (!)



# 오픈 - 소스 반도체 설계 도구들

## 알고리즘 / RTL에서 GDS 까지

“democratization of hardware design”

-- OpenROAD: Toward a Self-Driving, Open-Source Digital Layout Implementation Tool Chain

알고리즘 개발 (Un-Timed/Timed)



RTL 시뮬레이션 (Pre/Post-Sim)

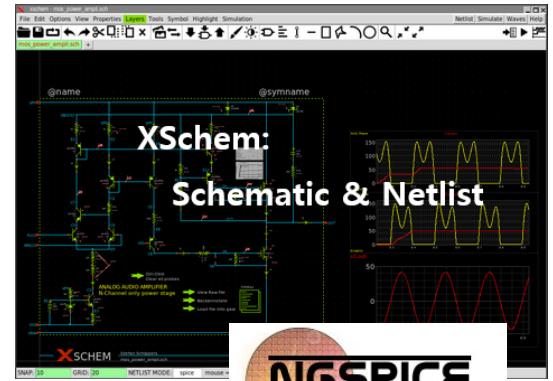


\* Questa 는 오픈 - 소스 아님.  
인텔에서 스타터 버전 무료 라이센스 발급

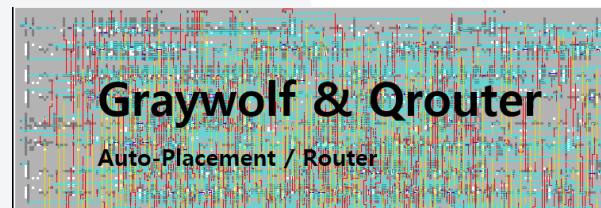
RTL 합성 (Synthesis)



표준 셀 제작



표준 - 셀 자동 배치 및 배선 (P&R)



LVS (SPICE Netlist)

Netgen 1.5

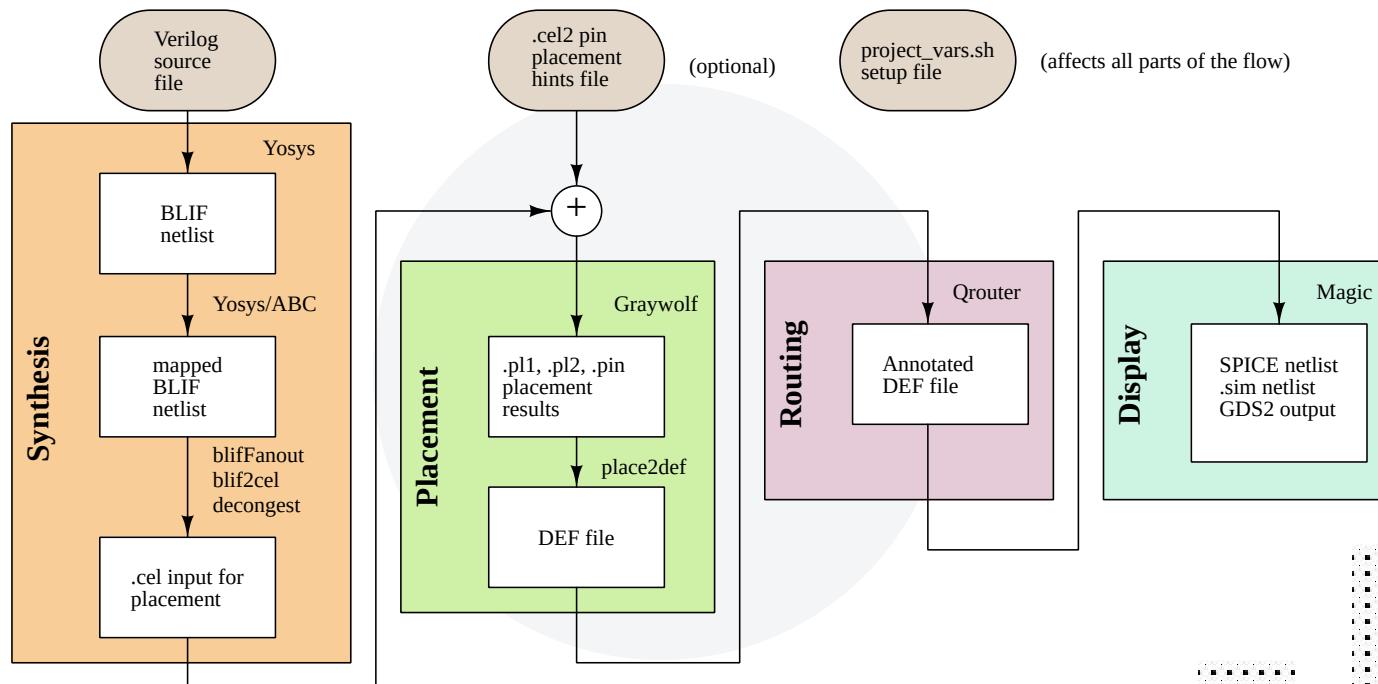
Netgen version 1.5 netlist comparison (LVS)

레이아웃 편집, DRC, GDS 생성



# Qflow: 설계 플로우 관리 도구

- <http://opencircuitdesign.com/qflow>
- RTL 합성, 배치배선, LVS, 레이아웃 생성 툴 통합
- 쉘, 파이썬, Tcl 스크립트로 구성



# ETRI 0.5u CMOS Std-Cell DK

Qflow: ETRI 0.5um CMOS 공정 추가에 필요한 파일들 [참조]

- 표준 셀 레이아웃 /NSPL 0.5um CMOS 디자인 툴 (VIA1, VIA2 주의 )
- 합성기용 리버티 라이브러리 (지연 특성 무효)
- P&R 용 LEF 및 파라메터 파일 (Magic 테크 파일 포함)
- 네트 시뮬레이션 용 베릴로그 모델 (지연 특성 무효)
- LVS 용 표준 셀 SPICE 네트리스트 (트랜지스터 수준 LVS)

## 공정 특이 사항 점검 및 툴 플로우 빈 곳 채우기

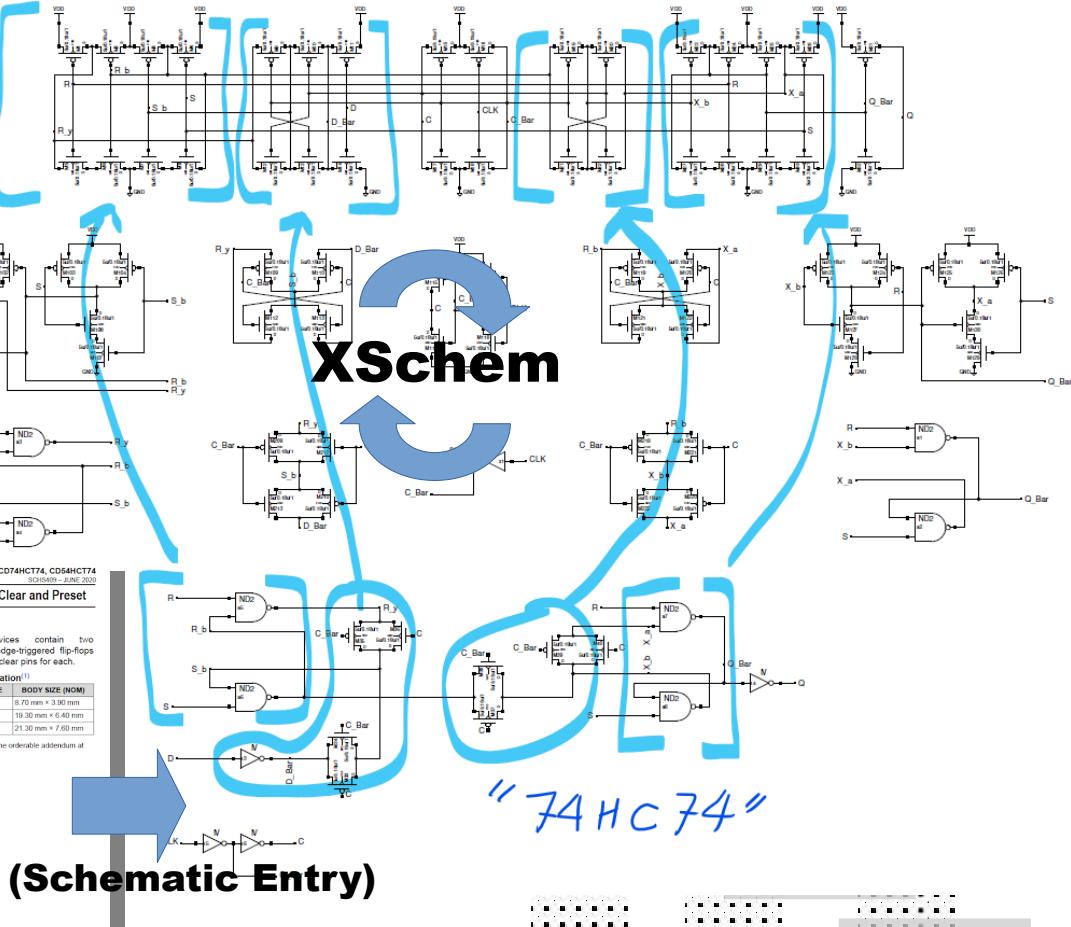
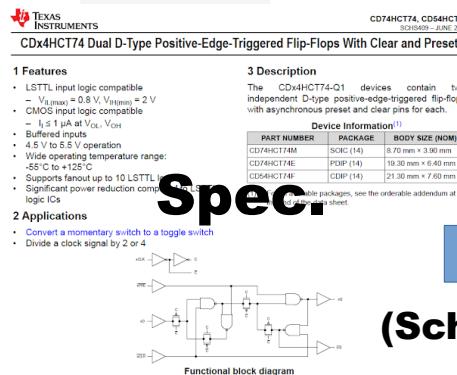
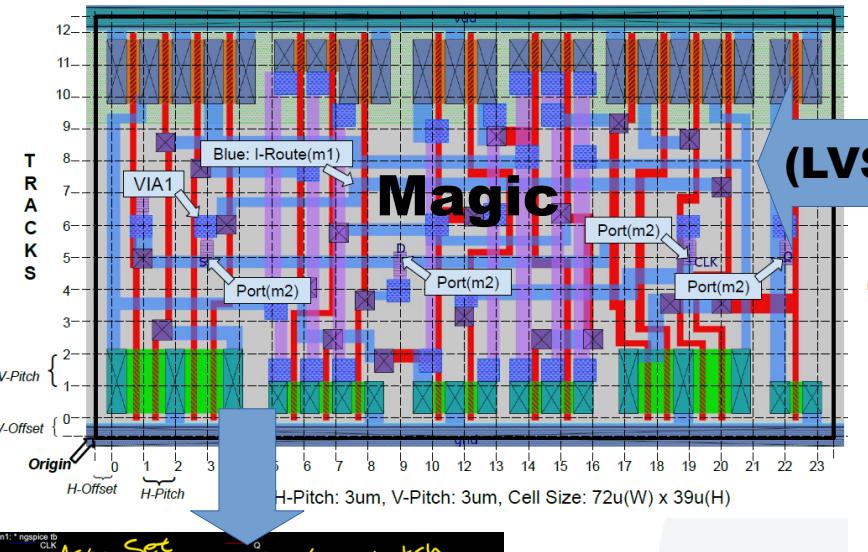
- VIA1 과 VIA2 의 크기 확대 (DRC), 금지된 겹침 VIA 검사
- 레이아웃 수동 수정 후 LVS 재검사
- 입출력 패드 라이브러리와 코어의 디자인 툴 검사 분리 (주의 :Poly2)

## 검증과 평가 방법 포함

- 오픈 – 소스 하드웨어 활용 (아두이노, 라즈베리 파이 ...)
- FPGA 에뮬레이터 (PSCE/Poorman's Standard Co-Emulator)

[참조]

# 표준 셀 제작 예 : D-F/F



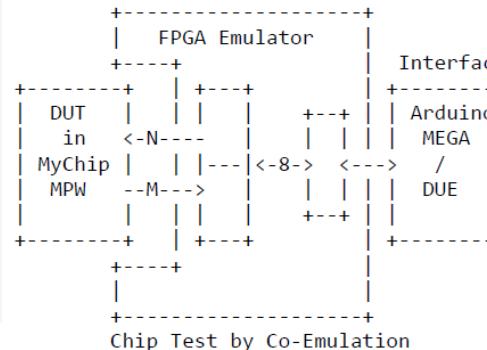
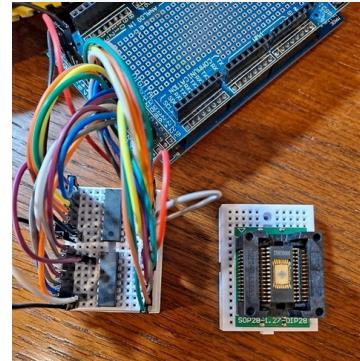
# 검증과 평가

## 칩 제작전 “검증”

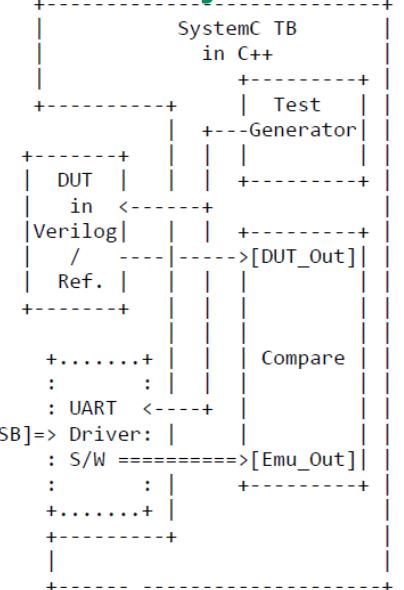
- 알고리즘 시뮬레이션 (Un-Timed) : **SystemC TB**(Timed)
- 합성전 RTL 시뮬레이션 : Verilator+**SystemC TB**
- 합성 후 네트 시뮬레이션 : Questa+**SystemC TB**
- FPGA 에뮬레이션 : FPGA/Arduino+**SystemC TB**
- LVS, DRC, 겹침 VIA 검사

## 칩 출고 후 기능성 “평가”

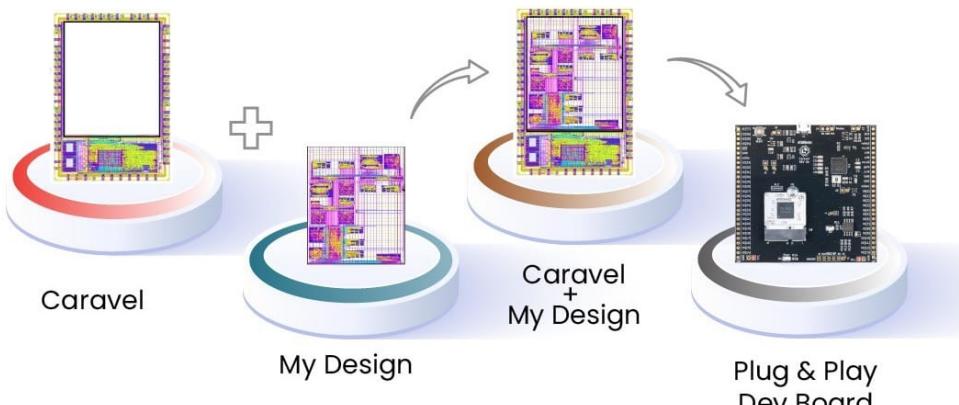
- 테스트 벤치 재사용 : FPGA/Arduino+**SystemC TB**
- 칩 테스트 장비 제작



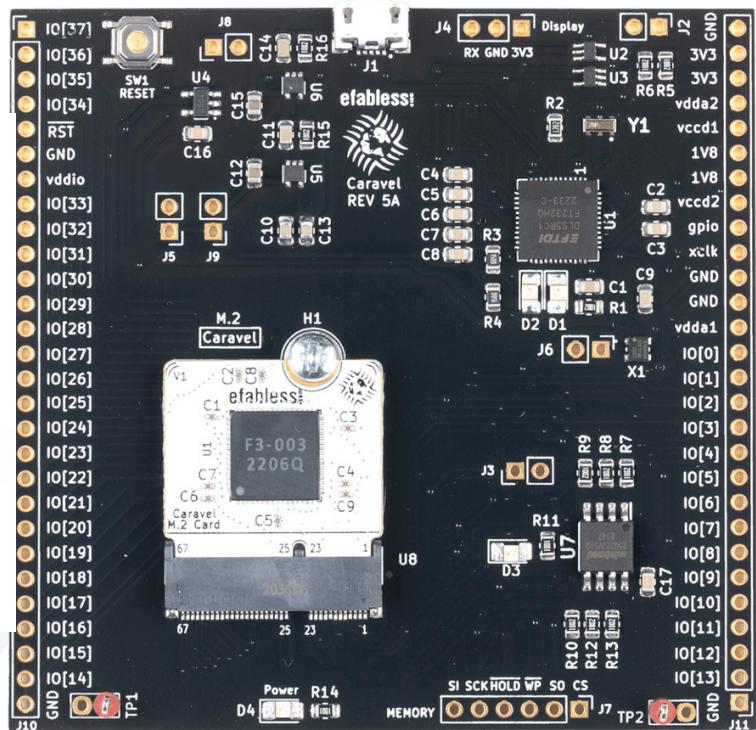
GoodKook's  
**PSCE System**



# eFabless 평가 보드



<https://efabless.com/chipignite>

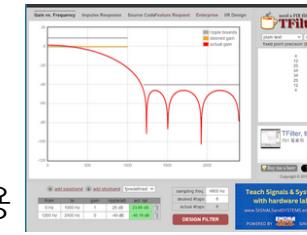


# 디자인 키 활용 예제 (1)

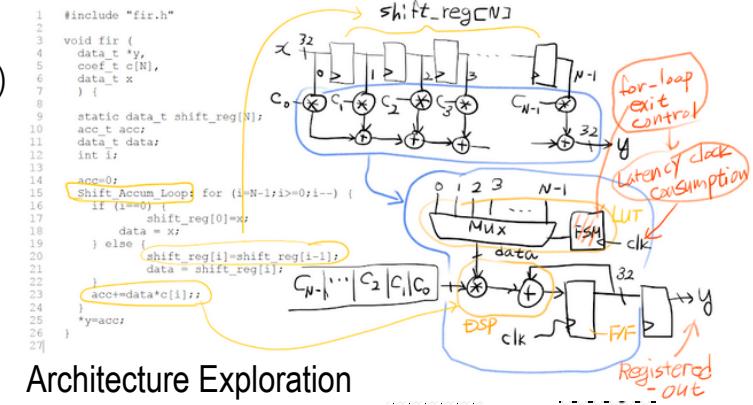
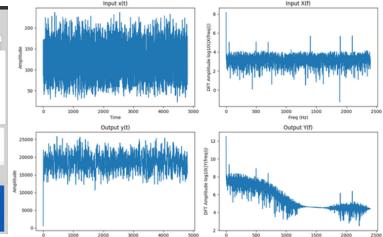
## 8-Tab FIR Filter

<https://github.com/GoodKook/ETRI-0.5u-CMOS-MPW-DK-Example--FIR8>

- 알고리즘 개발 : FIR 계수 (<http://t-filter.engineerjs.com>) , 파이썬 활용
- 구조탐색 : 파이프라인 병렬처리 구조 (Array Processing Elements)
- Un-Timed C – 모형에서 RTL로 변환하는 과정
- “내 칩 MPW” 요구에 맞추기 (칩 면적, 입출력 핀 수 제한)
- 아두이노 + FPGA 에뮬레이션 / 칩 테스트



Test vector visualization



# 디자인 키 활용 예제 (2)

## CPU 6502

<https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-DK-Example--6502-CPU>

- Apple-1 컴퓨터 재현 (워즈니악 모니터)
- 발매 당시 응용 프로그램 루미 이미지 실행 (AppleSoft BASIC, Jmon 등)
- 개발 도구 (CC65 컴파일러, A65 어셈블러) 활용 응용프로그램 제작
- SystemC TB / 싸이클 상세 동작
- C++ TB/ 아두이노 +FPGA 에뮬레이터 동작 (300Khz 이상)
- 레트로 컴퓨팅 (Retro-Computing) 취미가 들

[참조] Apple-1 복원 / 복제

[https://en.wikipedia.org/wiki/Replica\\_1](https://en.wikipedia.org/wiki/Replica_1)

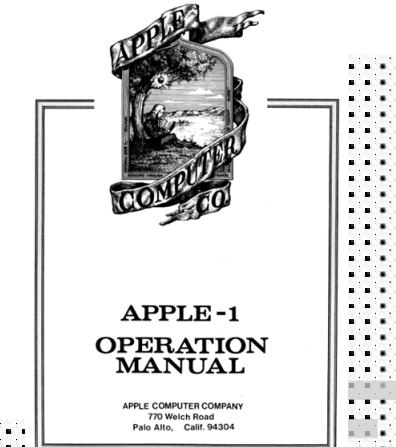
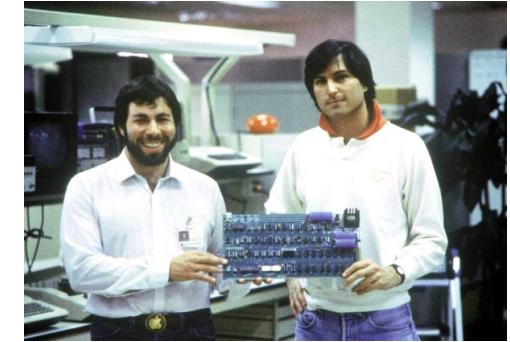
<https://www.apple-1-replica.com/>

<http://www.visual6502.org/>

<https://eater.net/6502>

<https://smartykit.io/products/smartykit-computer-construction-kit-late-2022>

<https://monster6502.com/>



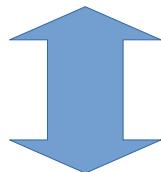
# 질문과 답

- “내 칩 MPW” 의 발전을 위한 과제  
날로 넓어지는 “추상성 격차” 의 극복



(최신 기술에 노출된 학생들)

AI 반도체 ?



(내 칩 MPW)

“교과서의 궁금증 풀어주기”

“재미있는 것 보여주기”

트랜지스터 !

(기초 기술의 풀 - 커스텀 공정)



- 반도체 산업 생태계 [\*] 이해 → 진출 분야를 넓히는 기회

[\*] 물질 / 공정 / 설계 / 자동화 도구 / 컴퓨팅 / 검증 / 평가 / 물류

# SCE-MI

Standard Co-Emulation / Modeling Interface, <http://www.accelera.org>

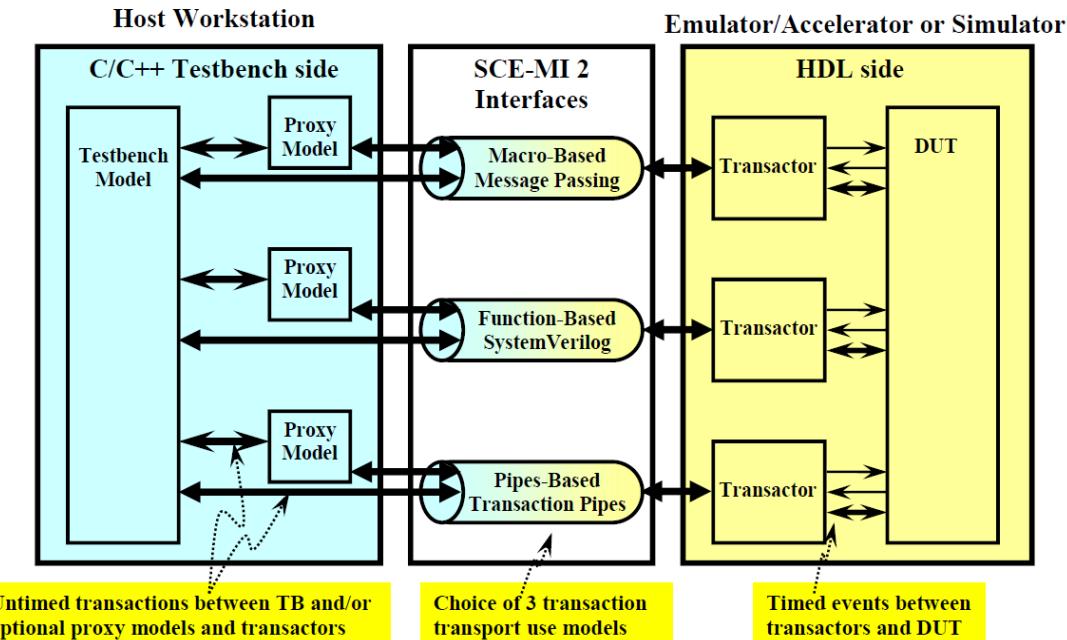


Figure 4.1

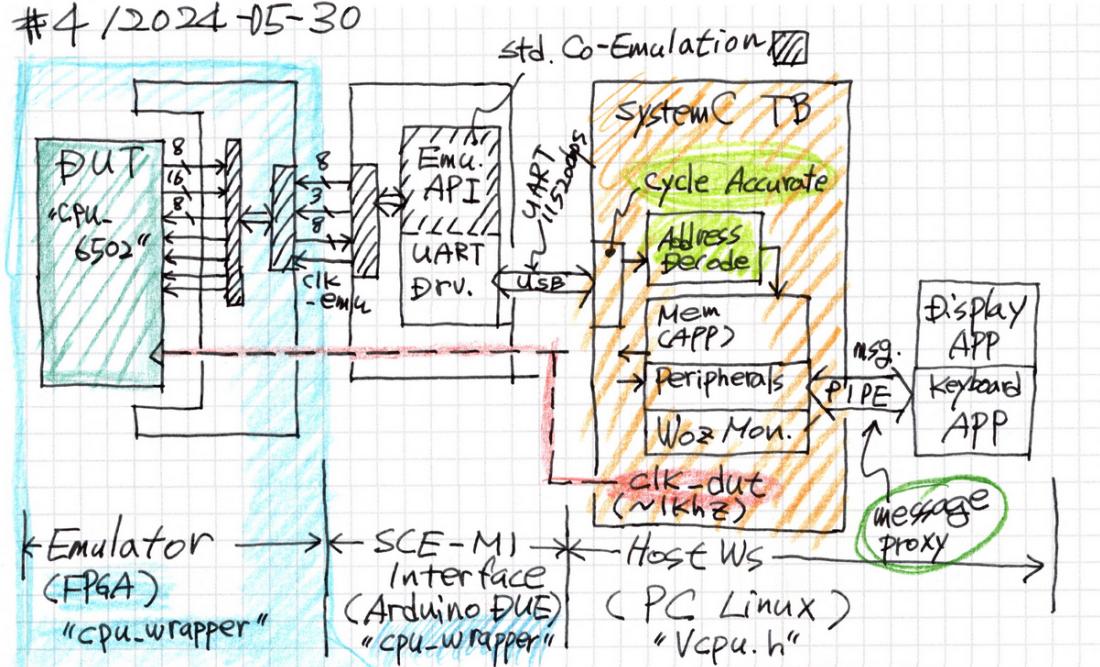
Three Interfaces

# Poorman's SCE-MI(CA)

[https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE\\_API](https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE_API)

Emulation MI/API: Cycle-Accurate

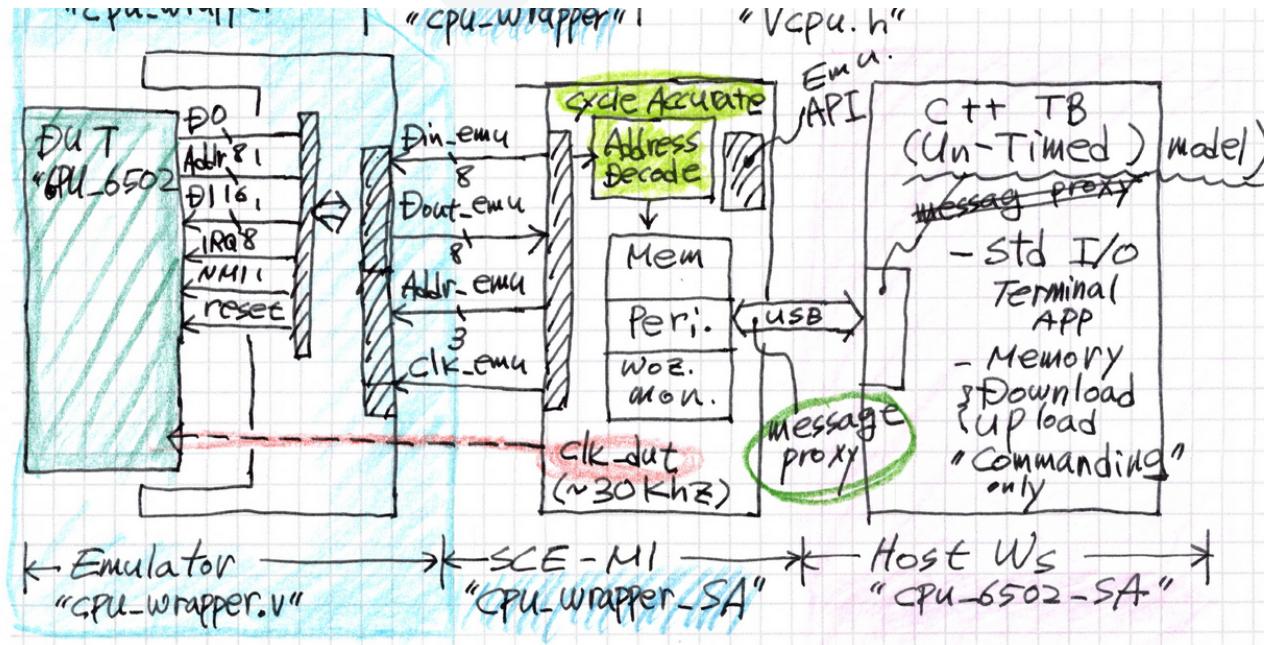
#4 / 2024-05-30



# Poorman's SCE-MI(SA)

[https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE\\_API](https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE_API)

## Emulation MI/API: Stand-Alone



# Poorman's SCE-MI(RT)

[https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE\\_API](https://github.com/GoodKook/ETRI-0.5um-CMOS-MPW-Std-Cell-DK/tree/main/PSCE_API)

Emulation MI/API: Real-Time

