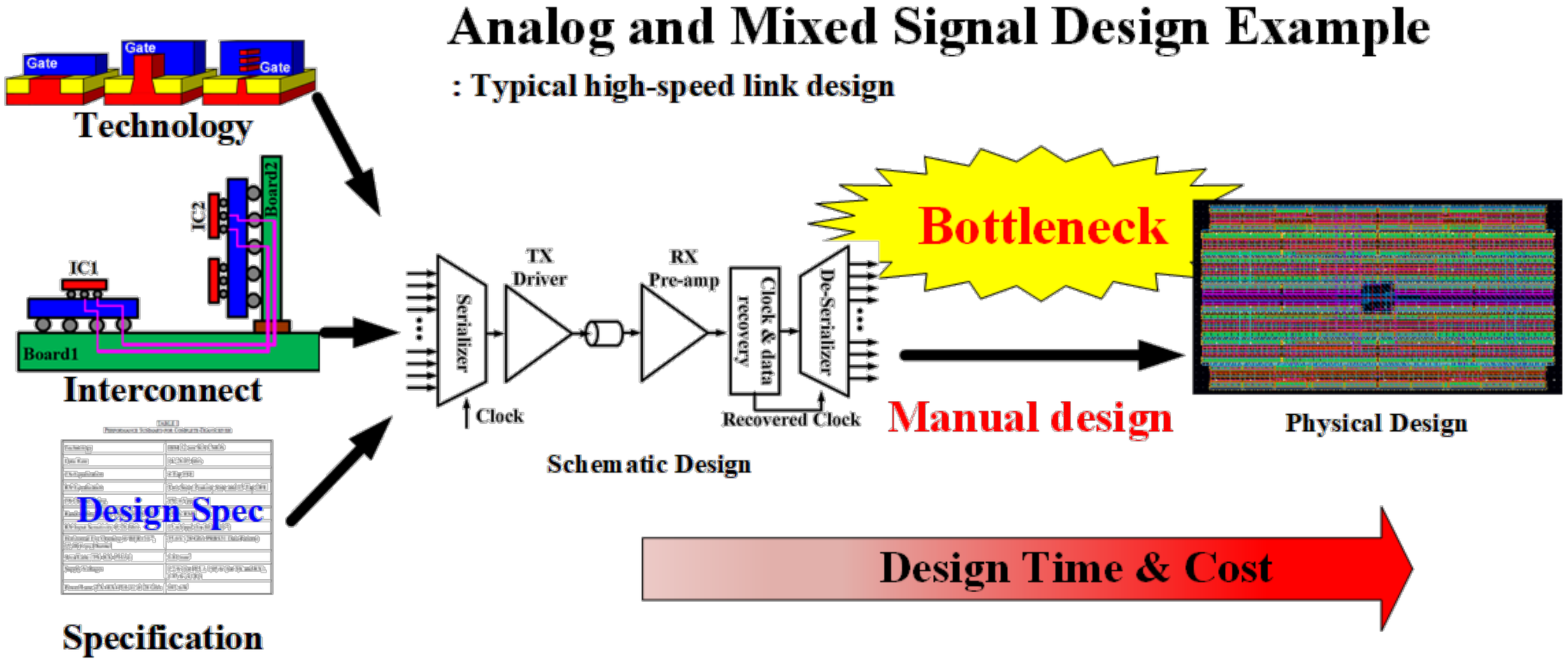
Methods for
Analog Layout Design Automation

Sungyu Jeong
Electrical Engineering Dept.
Pohang University of Science and Technology (POSTECH), South Korea
Advisor: Prof. Byungsub Kim (BEVIL Lab)

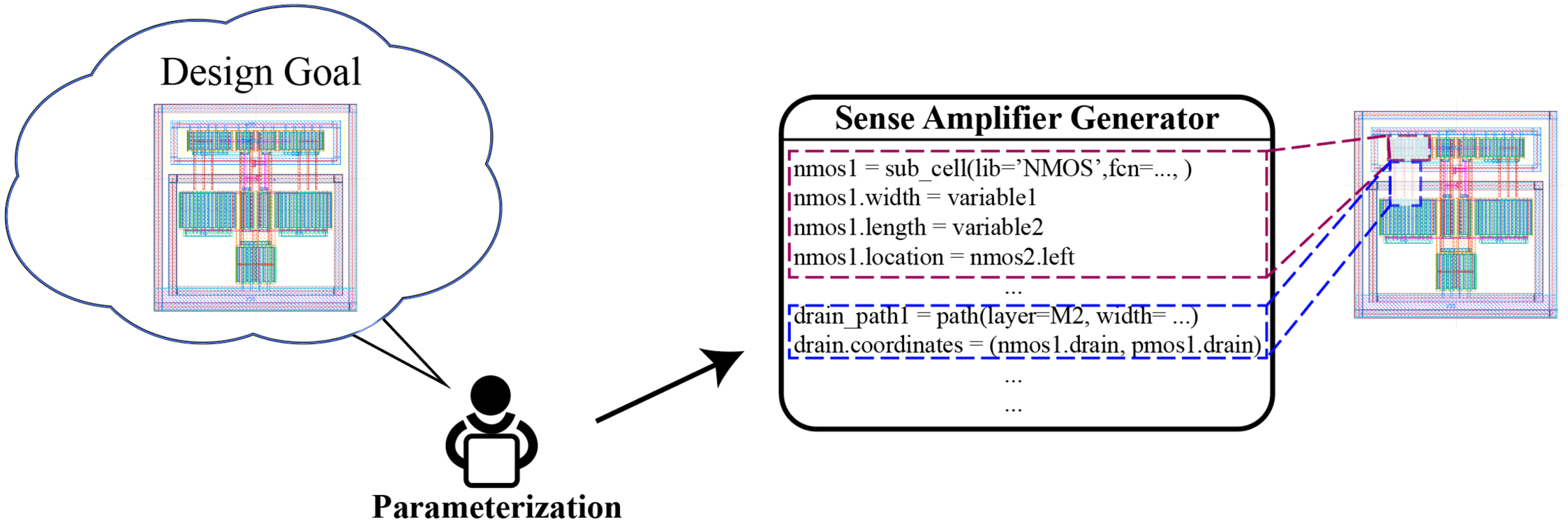
Contact:
sgjeong@postech.ac.kr
https://sungyu.dev

The problems of analog layout automation



■Full-custom layout is the major bottleneck in circuit design.

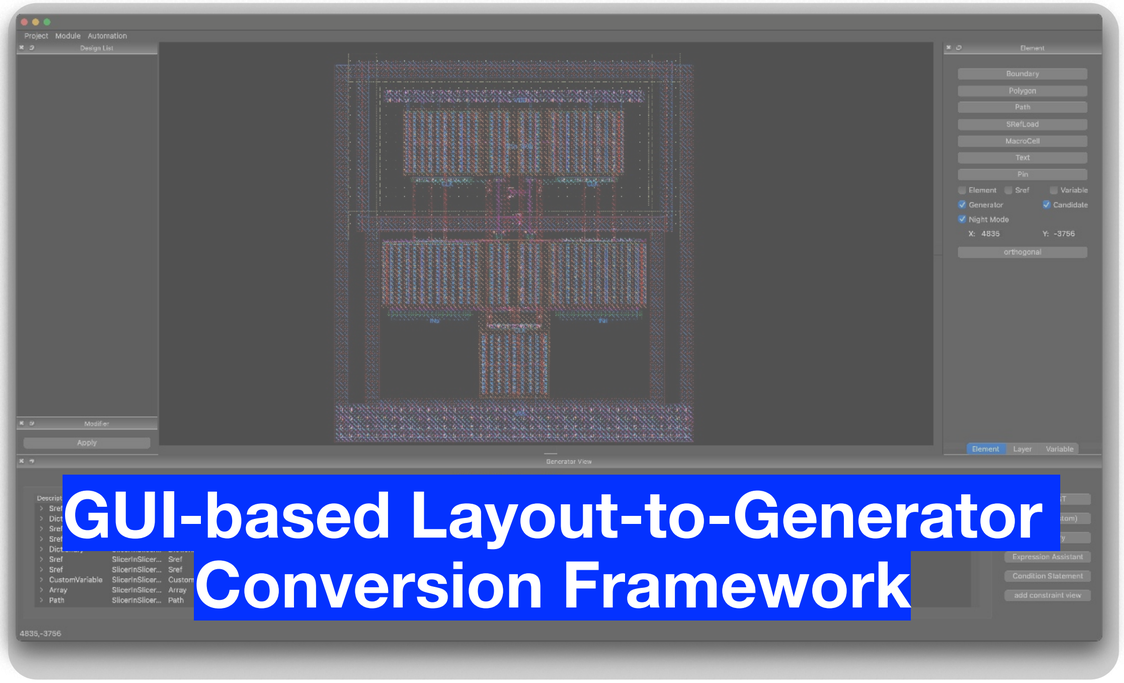
Previous Art: Procedural Layout Generator



- ☺ **Fast generation time**
- ☺ **Produces predictable output quality.**
- ☹ **Requires expertise in both analog circuits and programming.**
- ☹ **Requires significant work effort to develop generator.**

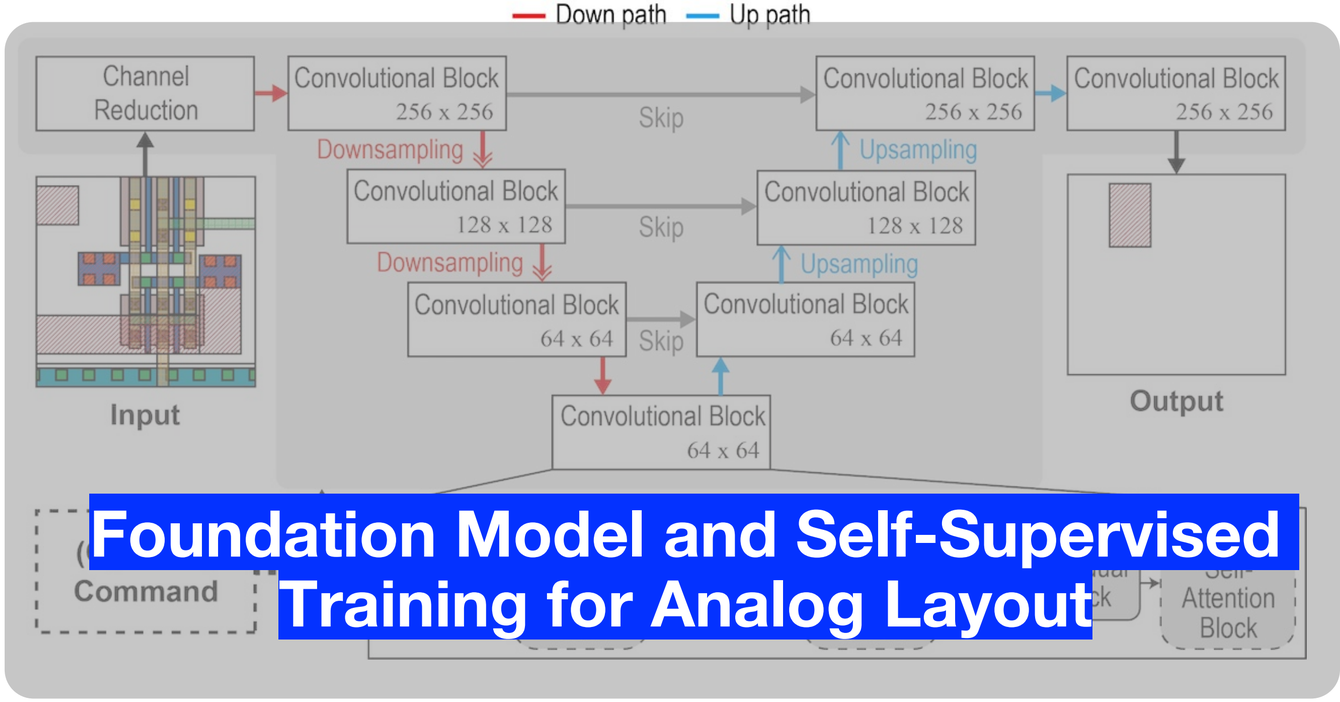
Suggested Solution: Reuse of Reference Designs

1) Explicit Reuse



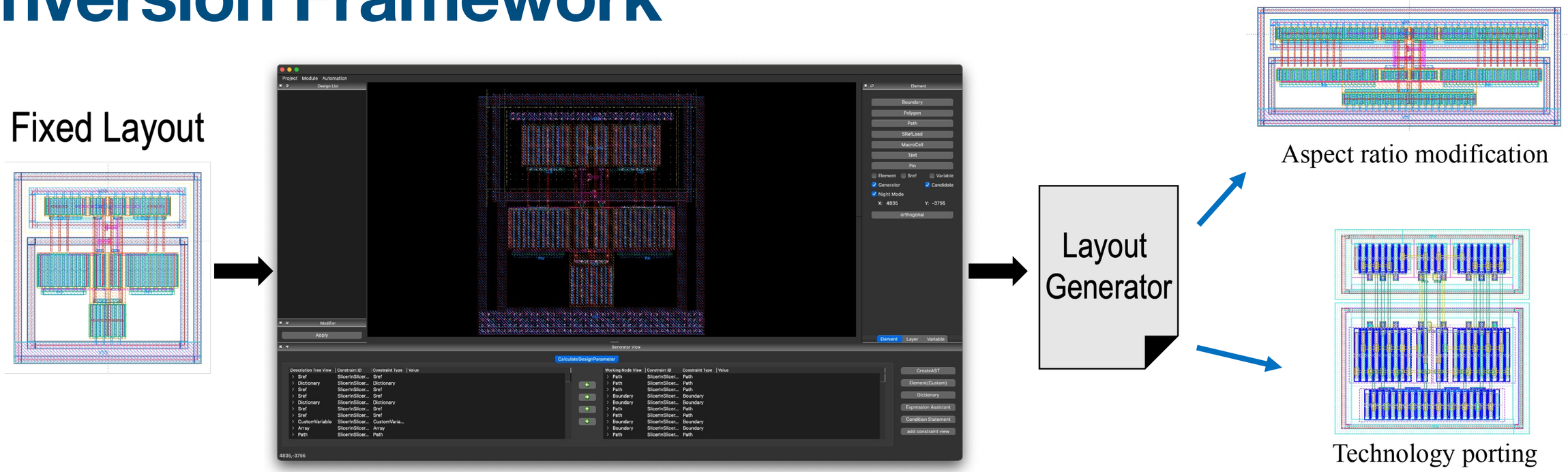
- Layout-to-generator conversion
- Deterministic approach
- Explicitly encodes design intent.

2) Implicit Reuse



- Self-supervised pre-training
- Probabilistic approach
- Implicitly learns design knowledge.

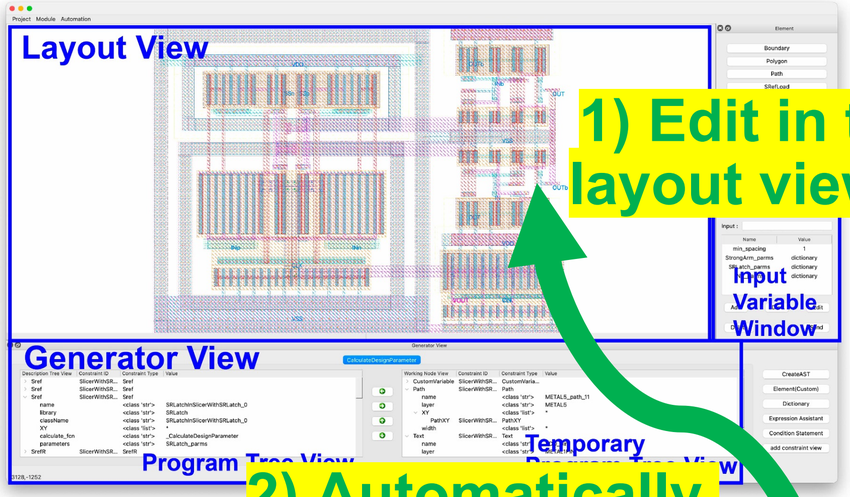
Research 1) GUI-based Layout-to-Generator Conversion Framework



Proposed workflow: Load reference layout, build hierarchy, and describe parameterization expression **(whole process in GUI)**

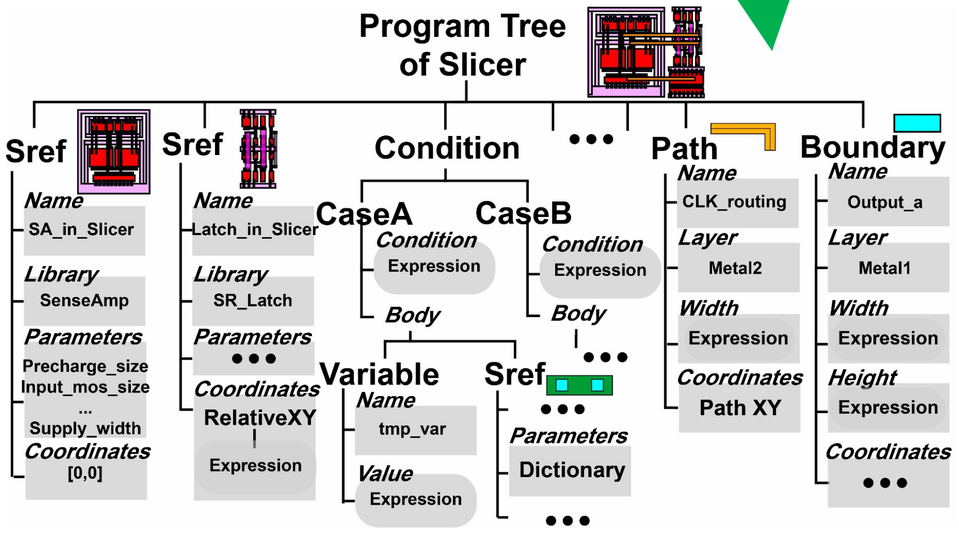
- ☺ **Requiring less knowledge of programming**
- ☺ **Leveraging well-made designs**
- ☺ **Reduced work effort and time**

Research 1) GUI-based Layout-to-Generator Conversion Framework



1) Edit in the layout viewer

2) Automatically updated!



■Users can edit directly in the layout viewer, similar to a conventional editor.

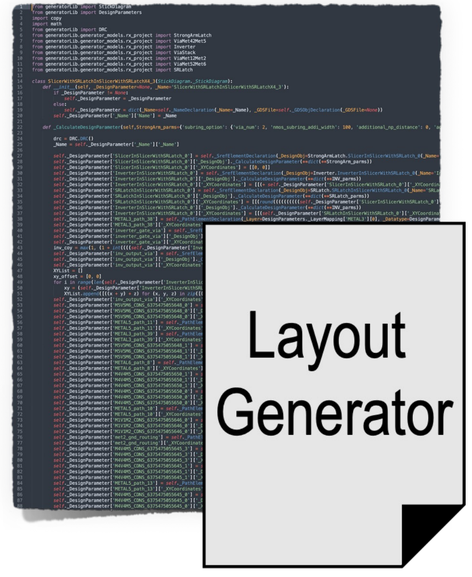
➔☺ Reduced workload

■Visualized program block

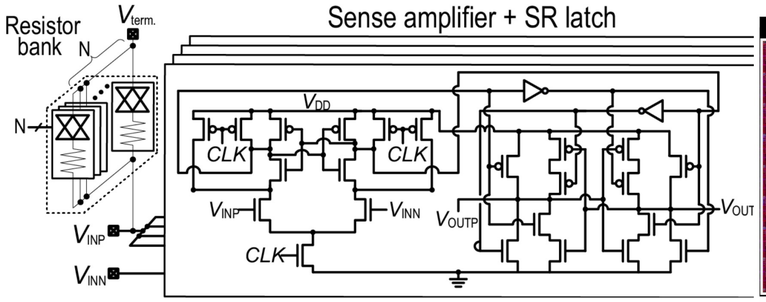
■Automatically compiled into Python scripts

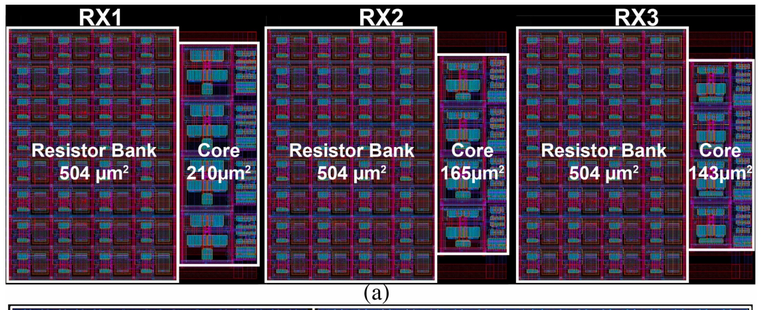
➔ ☺ Programming becomes as easy as building Lego blocks.

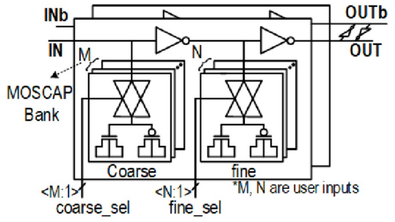
3) Compilation

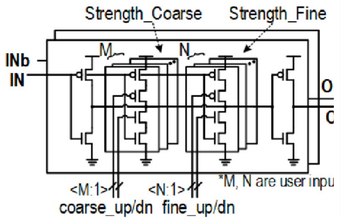


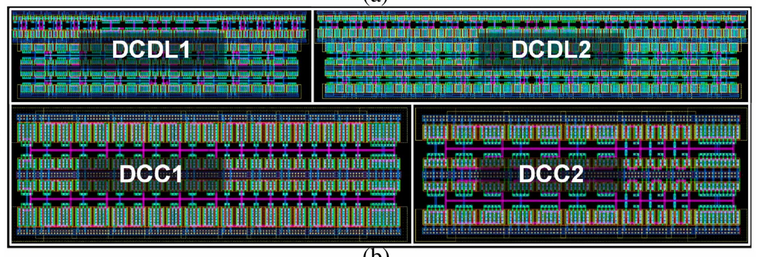
Research 1) Experimental Results











		RX1	RX2	RX3
Maximum Data Rate (Gb/s)		20	16	12
Active Region	Power (mW)	1.611	0.912	0.626
	Area (μm^2)	210	165	143
Resistor Bank	Power (mW)	8.487	8.548	8.601
	Area (μm^2)	504	504	504
Eye height (mV)		191.28	208.03	195.23
Eye width (UI)		0.6324	0.6749	0.7513
Generation time (sec)		0.289	0.289	0.284

- **Converted three reference circuits into generators**
  (High-speed wireline receiver, digitally controlled delay line, and duty cycle corrector)
- **Generated more than hundreds instances**
- **Demo video link (3:47)**

Generates in < 0.3 sec

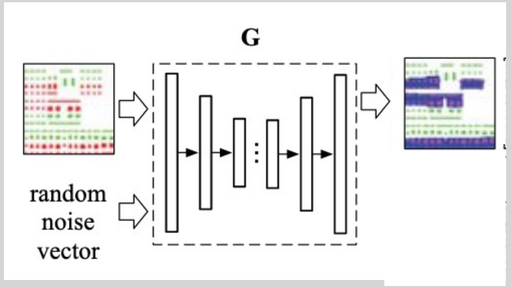
Cell	GUI (hour)	Script(hour)	Time Improvement
Receiver_top[1]	5	10.5	2.1x
├Sense amplifier	5.5	13.5	2.5x
├SR latch	4	11.5	2.9x
├Resistor bank	5	7	1.4x
└Inverter	0.5	2	4.0x
Receiver total	20	44.5	2.2x
DCDL_top[2]	7.5	17	2.3x
├MOSCAP bank unit	1.5	19.5	13.0x
└Inverter	0.5	10	20.0x
DCDL total	9.5	46.5	4.9x
DCC_top[2]	2.5	21	8.4x
├Inverter	0.5	12	24.0x
└Tri-state inverter	1.5	18	12.0x
DCC total	4.5	51	11.3x

Note
[1]:Expert designers each GUI, Script
[2]:Novice designers each GUI, Script
Reported time includes DRC+LVS runtime and code correction time.
Time measurement number is rounded by half hour.

- Case study shows development time of procedural layout generator decreases dramatically.
  (2.2X-11.3X faster)

Research 2) Motivation

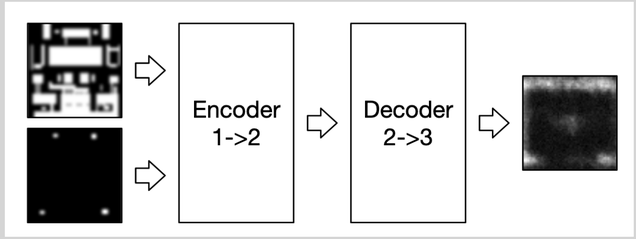
Task A



N-well generation model

B. Xu *et al.*, "WellGAN: Generative-Adversarial-Network-Guided Well Generation for Analog/Mixed-Signal Circuit Layout," in *2019 56th ACM/IEEE Design Automation Conference (DAC)*, June 2019, pp. 1–6.

Task B



Routing guidance model

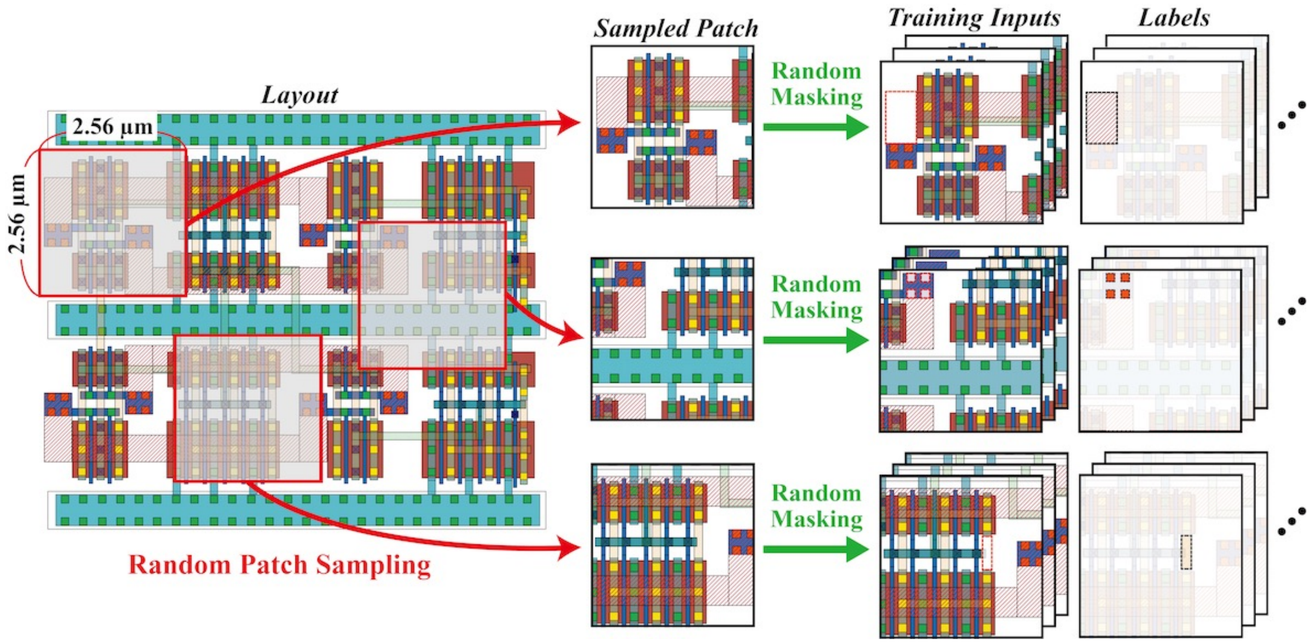
K. Zhu *et al.*, "GeniusRoute: A New Analog Routing Paradigm Using Generative Neural Network Guidance," in *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, Jan. 2019, pp. 1–8.

Task Z

Deep learning in analog layout is promising, but task-specific approaches face high labeling costs and development cost.

Research 2) Foundation Model & Self-Supervised Learning Method for Analog Layout Task

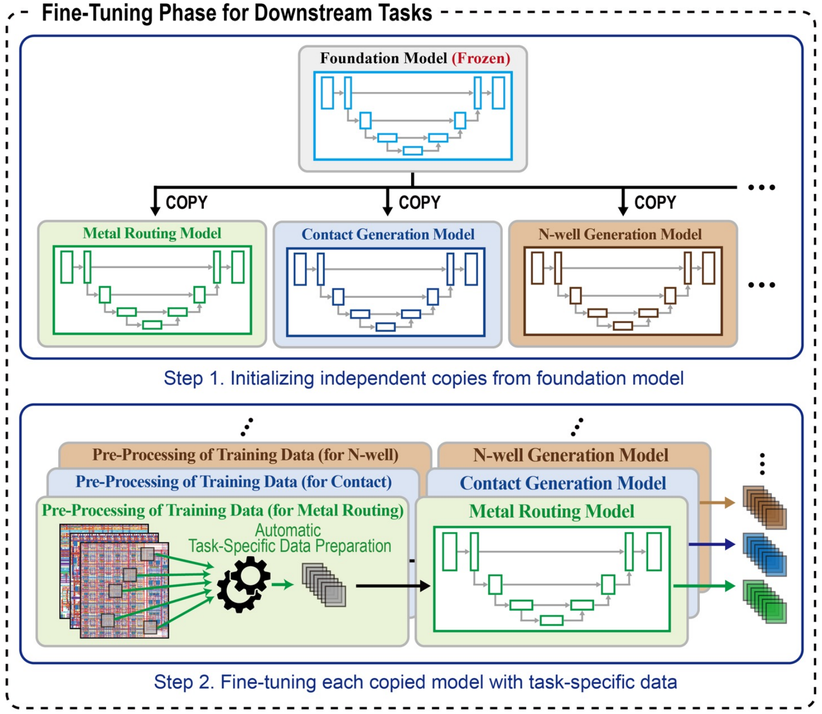
Self-supervised learning



Input: Unlabeled data

Output: Trainable input/output pair

Foundation model



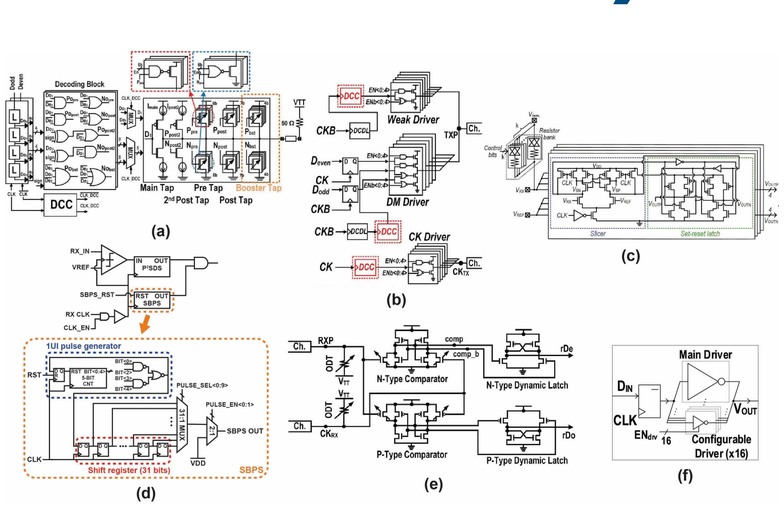
Task-agnostic foundation model

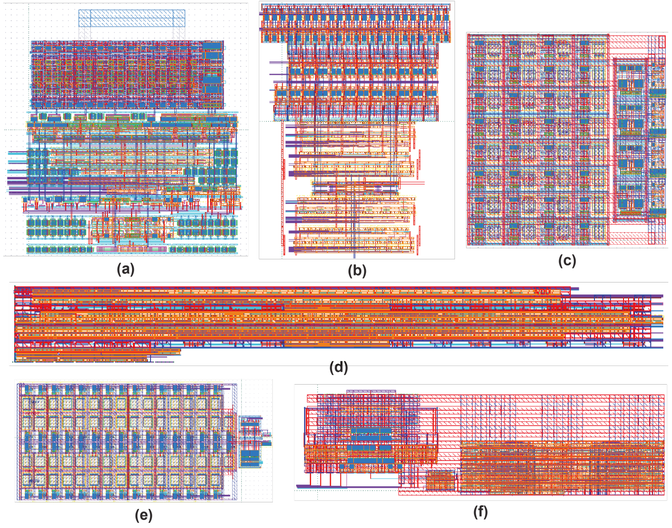
➔Fine-tuning for diverse downstream tasks

☺ **Leverages abundant unlabeled data → Low annotation cost**

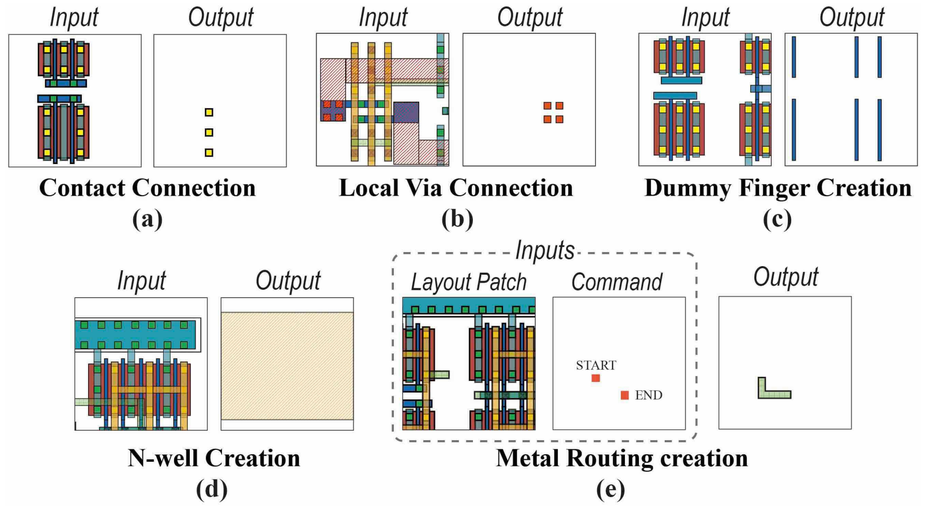
☺ **Single foundation model for all tasks → High scalability**

Research 2) Experimental Results



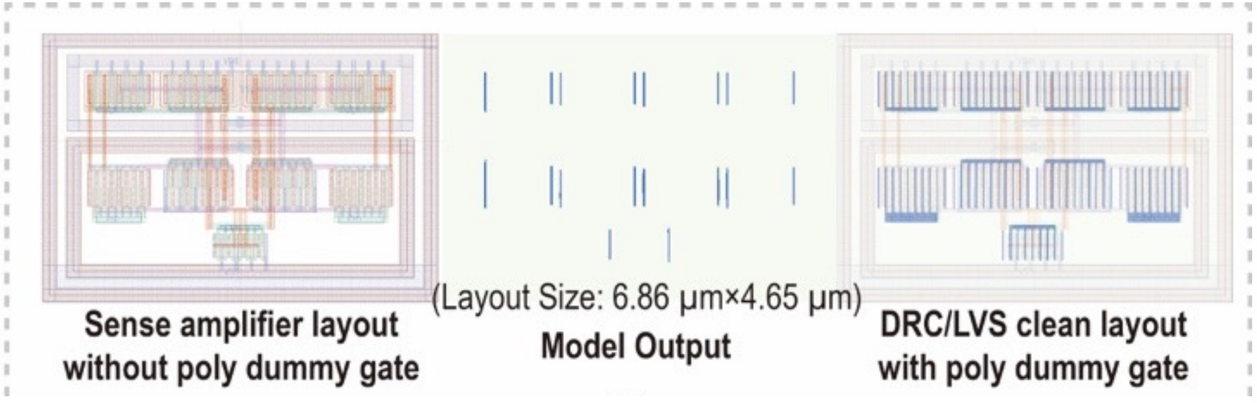


- Self-supervised pre-training on 6 different system-level AMS circuits
- 6 circuits → 324,000 training samples

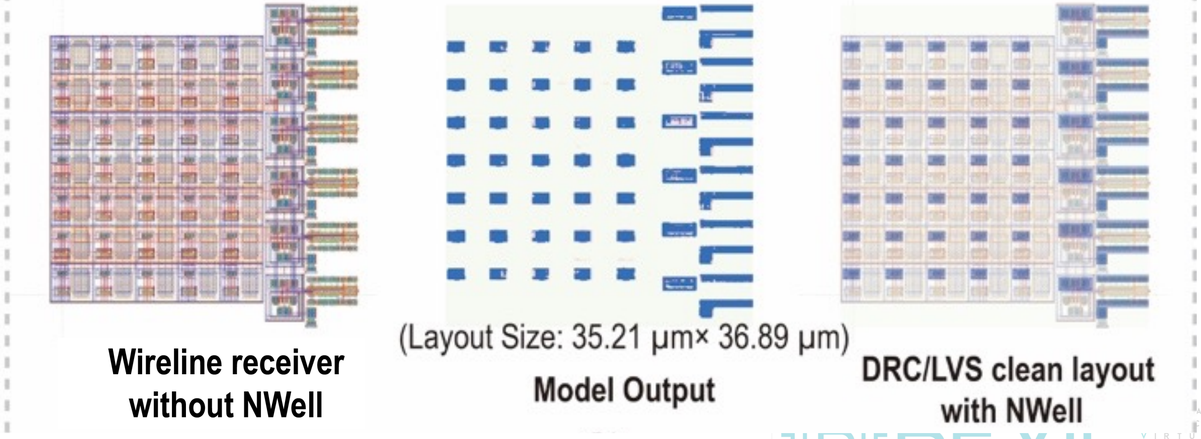


- Fine-tuning for five different downstream tasks

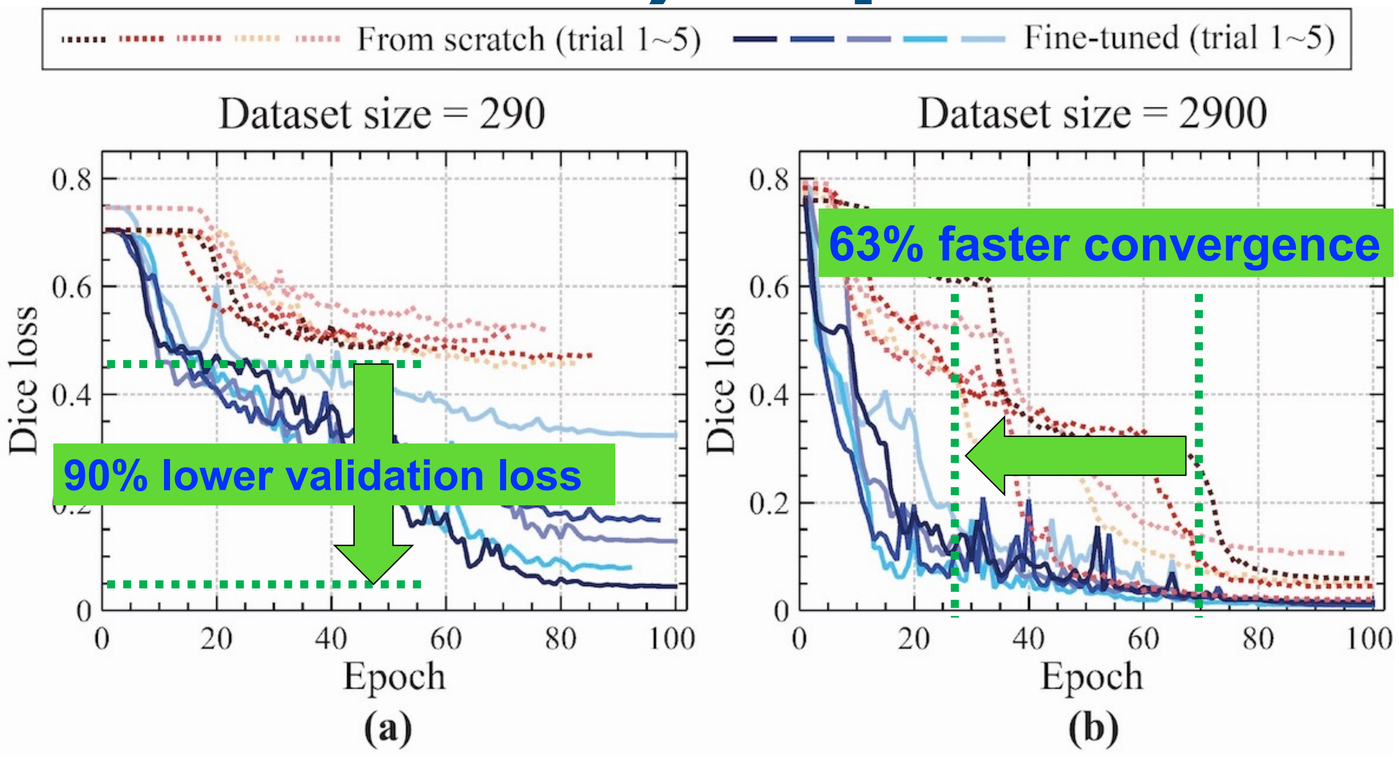
Dummy finger generation example



N-well generation example



Research 2) Experimental Results



☺ **Fine-tuning: Fast convergence & lower validation loss**

☹ **Train-from-scratch: Unstable convergence & overfitting**

Models achieved 96.6% DRC/LVS-clean success on 1,824 benchmark samples across five tasks.

Comparison with prior works