

WeChat
Official
Account

Website: ysyx.org
Email: ysyx@bosc.ac.cn

“One Student One Chip” Initiative

Let Students Design Their Own Open Source Processor Chips

2024.06



北京开源芯片研究院
BEIJING INSTITUTE OF OPEN SOURCE CHIP

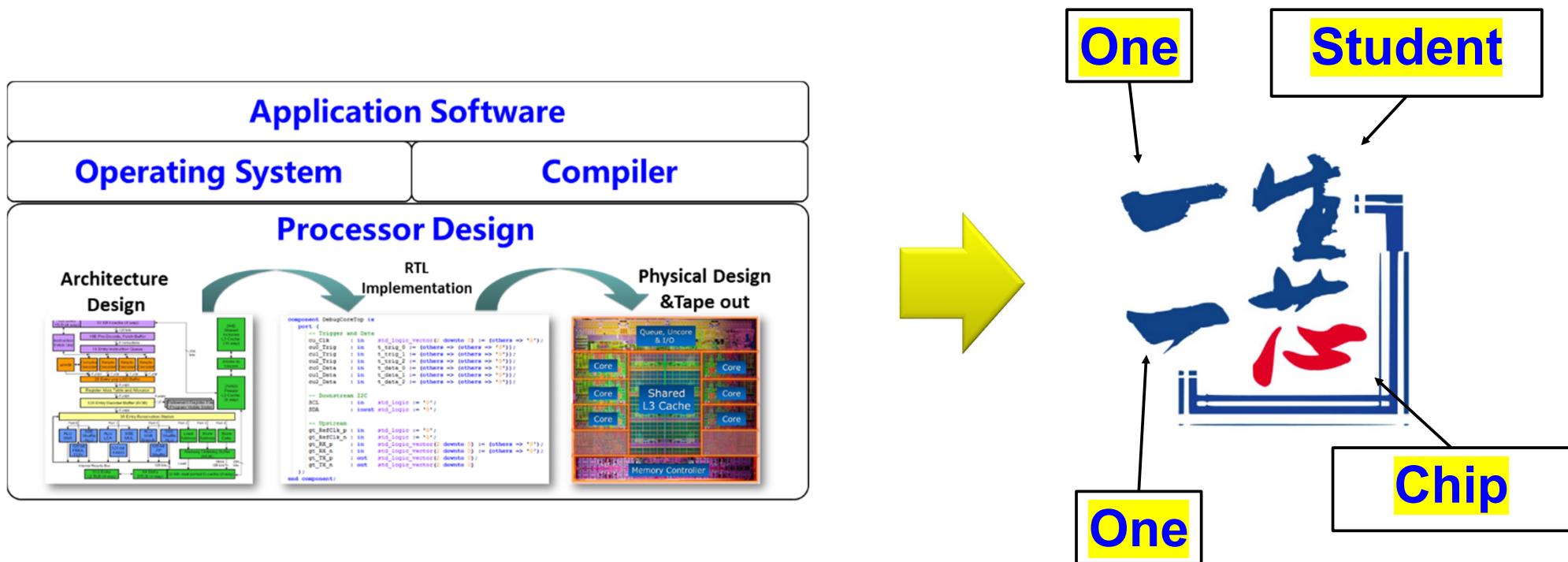


中国科学院大学
University of Chinese Academy of Sciences

1. Background

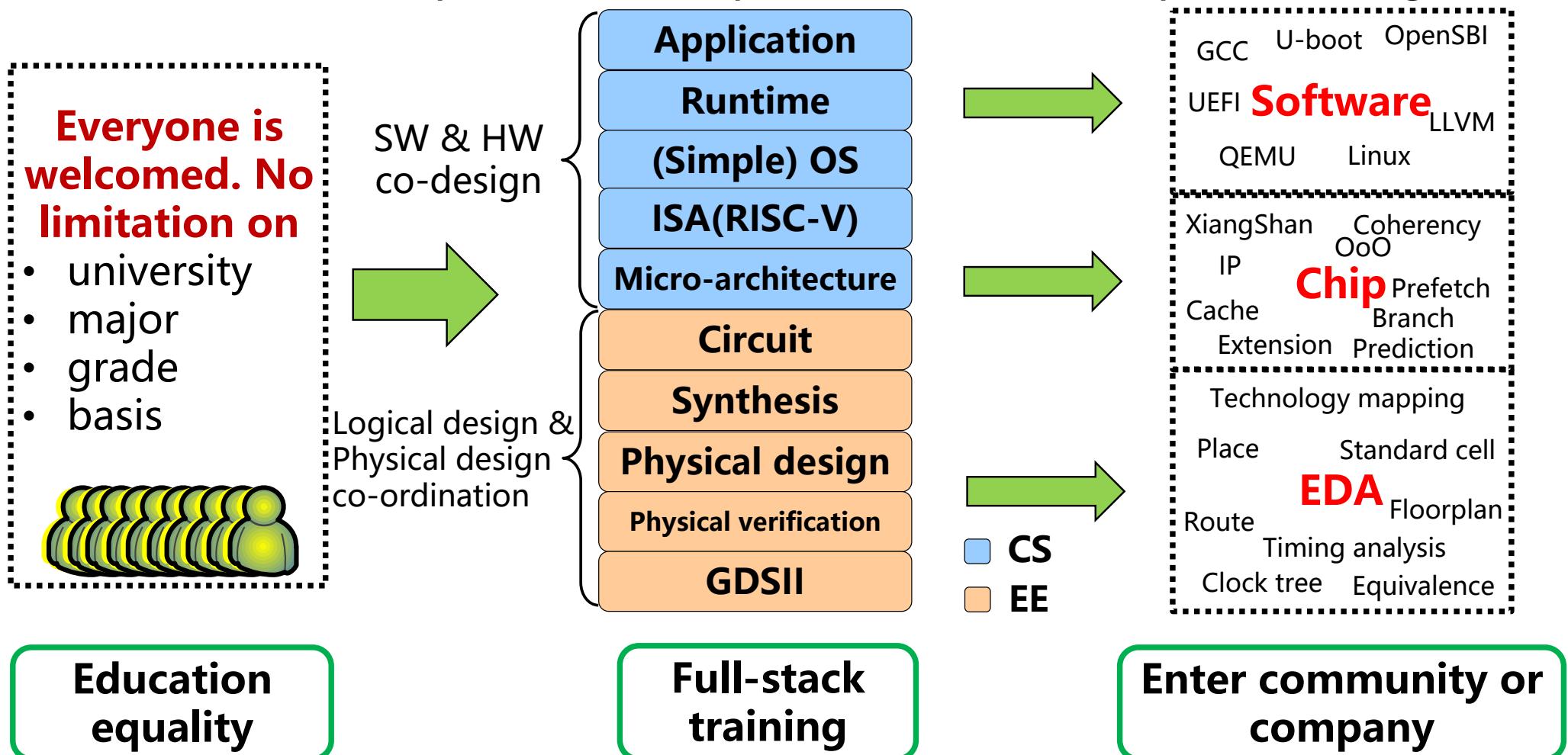
The One Student One Chip (OSOC) Initiative

- Learning-by-Doing: Teach students to build real chips
- Launched the OSOC Initiative in 2019



OSOC Initiative

Based on open-sourced, practice-oriented, open learning



> 6000 students participated in the OSOC Initiative

No.	Start Date	End Date	# of Applicants	# of Schools	#Stu. Learning	#Stu. Tapeout
1 st	Aug, 2019	-	5	1	5	5
2 nd	Aug, 2020	-	11	5	11	11
3 rd	Jul, 2021	Sep, 2021	760	168	215	51
4 th	Feb, 2022	Aug, 2022	1753	328	215	16
5 th	Aug, 2022	Jul, 2023	1881	379	155	13
6 th	July, 2023	In progress	2208	383	176	-

Updated: Jan 13, 2024

**2. 1st OSOC – Let students graduate along with their own chips
(Aug to Dec, 2019)**

The 1st OSOC (2019)

- Five senior undergraduates participated
- Completed the design of a **64-bit RISC-V** processor in four months
- The chip was **taped out** with 110nm and ran Linux and a self-built UCAS-core OS



Yue Jin



Huangqiang Wang



Kaifan Wang

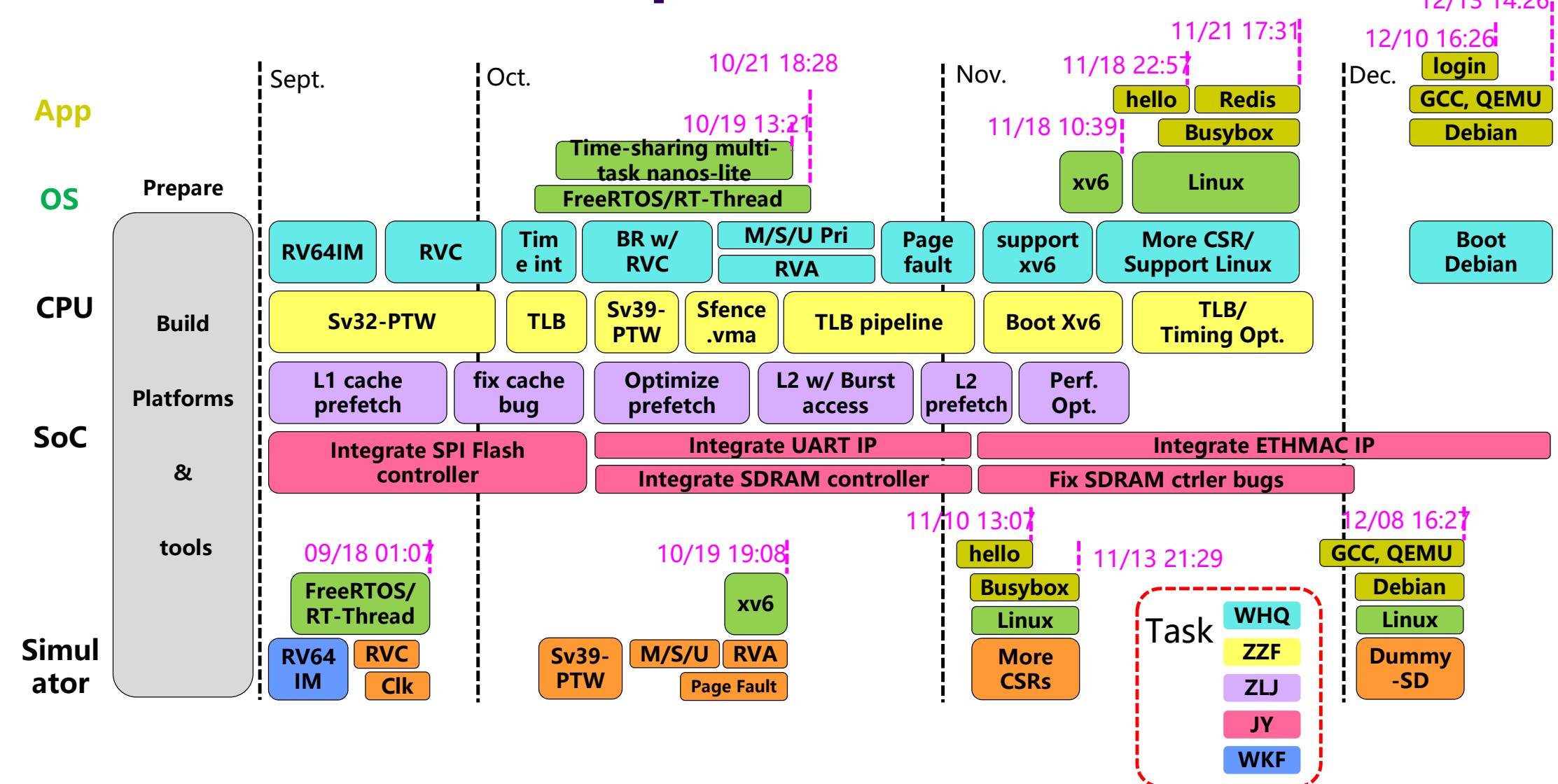


Linjuan Zhang



Zifei Zhang

Development Timeline

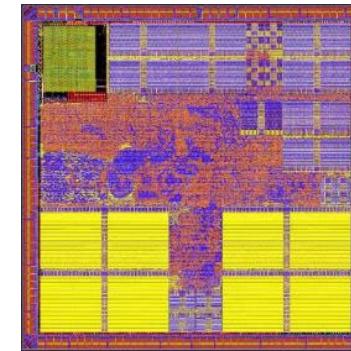


NutShell: A Linux-Compatible RISC-V Processor Designed by Undergraduates

A 64-bit RISC-V Processor

- Single-issue, 9-stage, in-order
- RV64IMAC, support M/S/U
- BPU with PHT, 512-entry BTB, 16-entry RAS
- Sv39, hardware TLB refill
- 32K L1I & L1D
- Read consistency for L1I & L1D
- 128K L2 cache, next line prefetch
- Develop with Chisel
- SDRAM, SPI flash, UART
- Support Linux 4.18.0 kernel
- Support Busybox
- Can boot Debian 11 on Emulator & FPGA

Tape-out w/ 110nm process



- 110nm process
- 10mm²
- 200mw@350MHz Typical
- TQFP100 package

Feedback from students

Self-exploring

与之前实验最大的不同.....就是**没有先行者一步一步的详细指导**，而是要**自己寻找方法，独立实现**，然后进行验证甚至推倒重来。

From user to creator

胡伟武老师曾经说过，我们计算机系的同学应该学会怎么造计算机而不是怎么用计算机。我以前对这句话并不太有感触，相反曾经质疑国科大计算机系的课程设置这么多硬件的内容是否合理。但**真正参与到项目中才发现在大学里所学的知识和技能是真的有用**。

大部分知识在体系结构课程中...**工作原理也很简单**，只有短短的几行，但是**真正在代码中实现却比自己所想象的要困难得多**。

More confidence, more patience

和4个月之前的自己相比.....**最重要的就是这种观念上的转变**。遇到bug不再在一个地方上死磕，而是从心理上告诉自己bug都是人写出来的，**只要有耐心，只要挖得足够深就一定能找到问题所在**。

Fulfillment

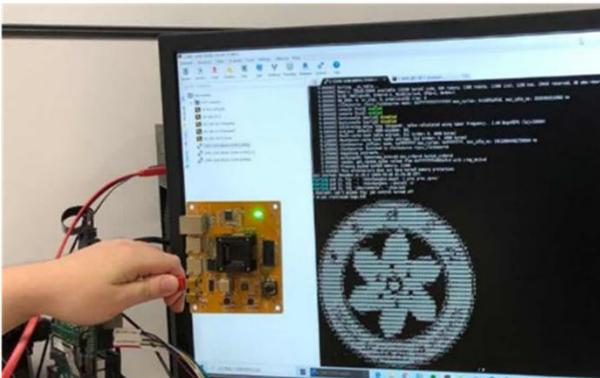
真正参与到项目中才知道课程作业就像直接给人采摘的果园一样，但项目却是**给一片荒地和几颗果树苗，从开垦种植和施肥都要自己动手**，并且还不知道这样能不能结出果实。不知为何，总觉得**从0开始种出的果实要更甜一些**。

Make students stronger!

Chips and Demos



Chips



Run Linux
Display CAS logo

配置开关	倍率	50MHz晶振	100MHz晶振
000	1	50MHz	100MHz
001	1.5	75MHz	150MHz
010	2	100MHz	200MHz
011	2.5	125MHz	250MHz
100	2.75	137.5MHz	275MHz
101	3	150MHz	300MHz
110	3.5	175MHz	350MHz
111	4	200MHz	400MHz

Demo



Thesis Defense 2020.6.2

1st OSOC finished successfully

- Students graduated with their own processor chips.



Report Accepted by RISC-V Global Forum 2020

NutShell: A Linux-Compatible RISC-V Processor Designed by Undergraduates

Your submission was accepted for
RISC-V Global Forum 2020

The RISC-V Event Team

★ 详情

Dear Huaqiang,

It is our pleasure to inform you that we
have accepted your submission.

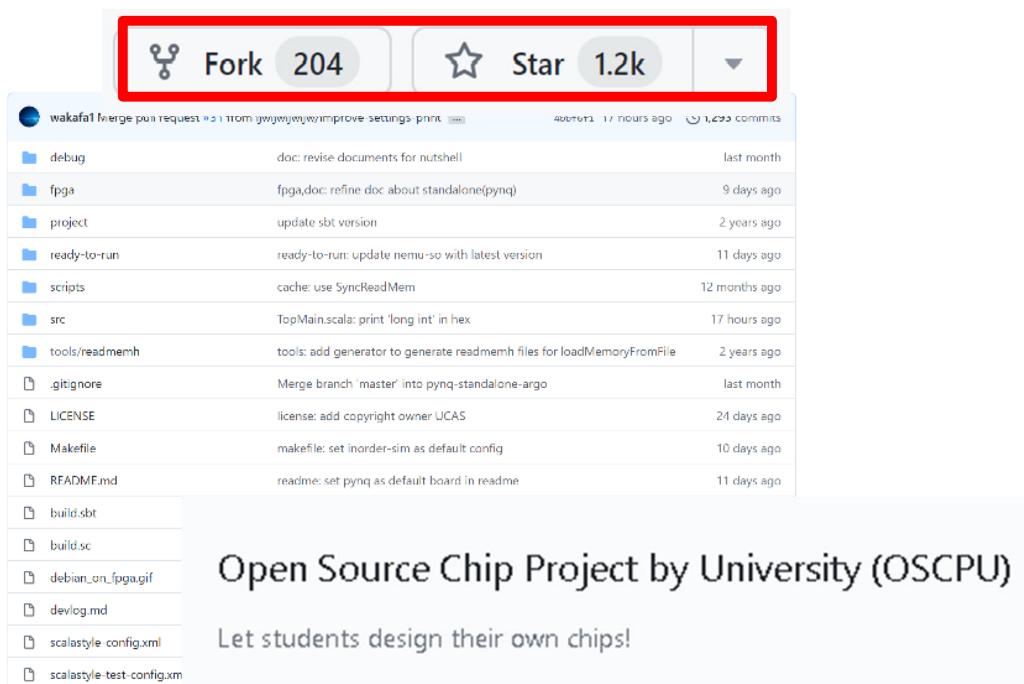
**Nutshell: A Linux-Compatible RISC-V
Processor Designed by Undergraduates**,
as a lightning talk, and would like to
welcome you as a speaker to the [RISC-V
Global Forum 2020](#), happening virtually
Thursday, September 3. Sessions will
overlap during US, Europe, and Asia
working hours. We ask that you please
review the information below and on the
[Speaker Guide](#) and complete all the
required items to confirm your speaking
engagement.

The event will take place on the virtual
event platform. MeetingPlav. The



Open-sourcing

- Open-sourced on **GitHub**
 - #Fork > 200

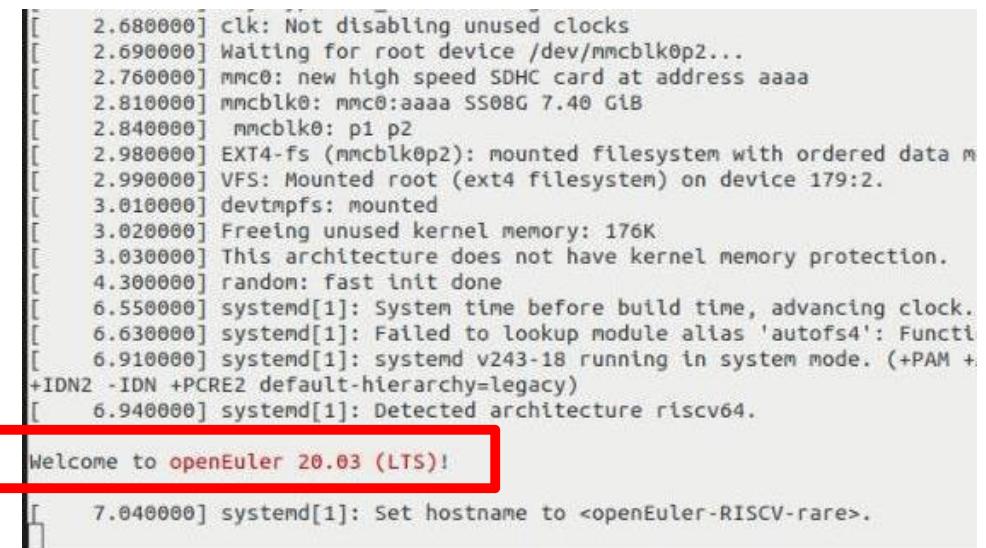


wakafat Merge pull request #5 from jwjjwjjw/improve-settings-print
4066761 17 hours ago 1,295 commits

File	Description	Time Ago
debug	doc: revise documents for nutshell	last month
fpga	fpga/doc: refine doc about standalone(pynq)	9 days ago
project	update sbt version	2 years ago
ready-to-run	ready-to-run: update nemu-so with latest version	11 days ago
scripts	cacher: use SyncReedMem	12 months ago
src	TopMain.scala: print 'long int' in hex	17 hours ago
tools/readmemh	tools: add generator to generate readmemh files for loadMemoryFromFile	2 years ago
.gitignore	Merge branch 'master' into pynq-standalone-argo	last month
LICENSE	license: add copyright owner UCAS	24 days ago
Makefile	makefile: set inorder-sim as default config	10 days ago
RFADME.md	readme: set pynq as default board in readme	11 days ago
build.sbt		
build.sc		
debian_on_fpga.gif		
devlog.md		
scalastyle config.xml		
scalastyle-test-config.xml		

Open Source Chip Project by University (OSCPU)
Let students design their own chips!

- **OpenEuler, an OS developed by Huawei, is booted successfully on NutShell**



```
[ 2.680000] clk: Not disabling unused clocks
[ 2.690000] Waiting for root device /dev/mmcblk0p2...
[ 2.760000] mmc0: new high speed SDHC card at address aaaa
[ 2.810000] mmcblk0: mmc0:aaaa SS08G 7.40 G18
[ 2.840000] mmcblk0: p1 p2
[ 2.980000] EXT4-fs (mmcblk0p2): mounted filesystem with ordered data mode
[ 2.990000] VFS: Mounted root (ext4 filesystem) on device 179:2.
[ 3.010000] devtmpfs: mounted
[ 3.020000] Freeing unused kernel memory: 176K
[ 3.030000] This architecture does not have kernel memory protection.
[ 4.300000] random: fast init done
[ 6.550000] systemd[1]: System time before build time, advancing clock.
[ 6.630000] systemd[1]: Failed to lookup module alias 'autofs4': Function not found
[ 6.910000] systemd[1]: systemd v243-18 running in system mode. (+PAM +IDN +IDN +PCRE2 default-hierarchy=legacy)
[ 6.940000] systemd[1]: Detected architecture riscv64.

Welcome to openEuler 20.03 (LTS)!

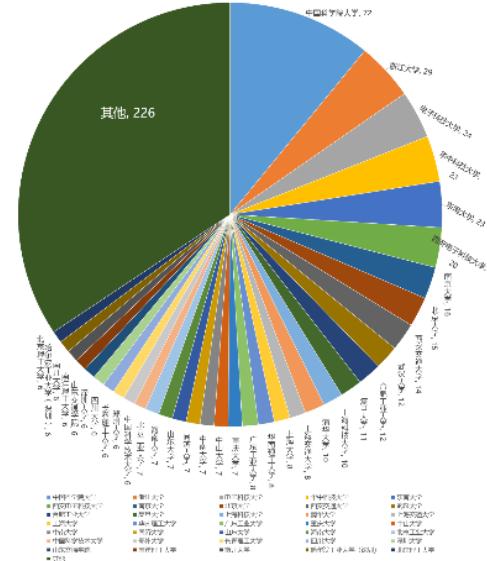
[ 7.040000] systemd[1]: Set hostname to <openEuler-RISCV-rare>.
```

<https://github.com/OSCPU/NutShell>

3. 3rd OSOC – Exploring large-scale learning (Jul to Dec, 2021)

3rd One Student One Chip

- Bring chip talent cultivation and open-source chip community together, while paying special attention to whole-flow chip design.
 - **760 students from 168 colleges (30 oversea)**
 - Undergraduate students: 50%
 - Students-in-school: 82%



中国开放指令生态(RISC-V)联盟
China RISC-V Alliance

Breakdown of colleges

■ Total : 760 students from 168 colleges (including 30 oversea)

■ Students in school: 625 (82%)

■ China

- 72 : University of Chinese Academy of Sciences
- 29 : Zhejiang University
- 24 : University of Electronic Science and Technology of China
- 23 : Huazhong University of Science and Technology
- . . .

■ Oversea

- Many students are from the USA and Europe
 - Most of them are Chinese students
 - We are translating the Chinese materials into English version.

Georgia Institute of Technology
University of Toronto
The University of Edinburgh
KTH Royal Institute of Technology
Nanyang Technological University
The University of Melbourne
University of Michigan
Monash University
University College London
Clemson University
University of California, Los Angeles
University of California, Riverside
University of California, Davis
University of Waterloo
University of Virginia
Technische Universiteit Delft
Dartmouth College
Dalhousie University
Paris-Saclay University
Royal Melbourne Institute of Technology

Teaching Assistants

- Selected from students

- Inspire others to think, instead of giving the answer
- Leave the training to students

List of TA (Jun, 2023)

	Name	University	Grade	
1	Jinbiao Miao	University of Science and Technology of China	Grade 2 master	
2	Zhenwei Duan	University of Science and Technology of China	Grade 3 master	
3	Hanzhang Liu	Taiyuan University of Technology	Junior	
4	Xun Cao	University of Science and Technology of China	Grade 2 master	
5	Haifan Yang	Zhejiang Gongshang University	Senior	
6	Shiyang Cao	University of Science and Technology of China	Grade 2 master	
7	Rentao Ni	Northeasten University	Grade 1 master	
8	Ren Wei	Lanzhou University	Senior	
9	Jiabin Wu	Qingdao University	Senior	
10	Lu Chen	University of Chinese Academy of Sciences	Ph.d Candidate	
11	Jinlun Su	Taiyuan University of Technology	Senior	

Students' learning note and meeting attendance

学号	姓名	学校	专业	年级	进度记录链接	"提问的智慧"读后感	实验报告	是否两周未更新进度记录	3.2	3.2	3.2	4.1	4.1	4.1	5.0	5.1	5.15	5.20	6.5.10	7.3	7.10	7.17	7.24	8.7	8.14	
ysyx_22040020	毛英勋	天津大学	电子信息工程	大四	ysyx_220020-毛英勋-天津大学-学习记录	ysyx_220020-毛英勋-天津大学-读后感	ysyx_220020-毛英勋-天津大学-实验报告	两周未更新	3.20	2022.4.15-待办	PA1	2022.2.2022.npc-c	2022.2.2022.2022.7.10请假	2022.8.7请假												
ysyx_22040047	吴浩宇	东北大学	计算机科学与技术	大二	ysyx_220017-吴浩宇-东北大学-学习记录	ysyx_220017-吴浩宇-东北大学-读后感	ysyx_220017-吴浩宇-东北大学-实验报告		2022.3.20	2022.4.10请假	PA2-3	2022.6.1	2022.2022.7.17请假													
ysyx_22040091	冯洁源	中国科学院大学	计算机科学与技术	大三	ysyx_220091-冯洁源-中国科学院大学-学习记录	ysyx_220091-冯洁源-中国科学院大学-读后感	ysyx_220091-冯洁源-中国科学院大学-实验报告		2022.3.20	2022.4.10请假	PA4-1															
ysyx_22040030	刘刚平	东北大学秦皇岛分校	计算机类	大一	ysyx_20030-刘刚平-东北大学秦皇岛分校-学习记录	ysyx_20030-刘刚平-东北大学秦皇岛分校-读后感	ysyx_20030-刘刚平-东北大学秦皇岛分校-实验报告		2022.3.20	2022.3.2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040063	包子姐	河南理工大学	计算机科学与技术	大二	ysyx_220063-包子姐-河南理工大学-学习记录	ysyx_220063-包子姐-河南理工大学-读后感	ysyx_220063-包子姐-河南理工大学-实验报告		2022.3.20	2022.4.10请假	PA2-17	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假										
ysyx_22040031	尚伟	青海科技大学	电子信息	研一	ysyx_220031-尚伟-青海科技大学-学习记录	ysyx_220031-尚伟-青海科技大学-读后感	ysyx_220031-尚伟-青海科技大学-实验报告		2022.3.20	2022.4.10请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040012	李睿	海南大学	电子信息工程	大二	ysyx_220012-李睿-海南大学-学习记录	ysyx_220012-李睿-海南大学-读后感	ysyx_220012-李睿-海南大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040099	樊海荣	太原理工大学	通信工程	大二	ysyx_220099-樊海荣-太原理工大学-学习记录	ysyx_220099-樊海荣-太原理工大学-读后感	ysyx_220099-樊海荣-太原理工大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040025	顾立强	东北大学	物联网工程	大二	ysyx_220025-顾立强-东北大学-学习记录	ysyx_220025-顾立强-东北大学-读后感	ysyx_220025-顾立强-东北大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040034	杜亮明	太原理工大学	软件工程	大四	ysyx_220034-杜亮明-太原理工大学-学习记录	ysyx_220034-杜亮明-太原理工大学-读后感	ysyx_220034-杜亮明-太原理工大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040064	黎伟强	国防科技大学	软件工程	大二	ysyx_220064-黎伟强-国防科技大学-学习记录	ysyx_220064-黎伟强-国防科技大学-读后感	ysyx_220064-黎伟强-国防科技大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040046	王凯	中国科学技术大学	计算机科学与技术	大三	ysyx_220046-王凯-中国科学技术大学-学习记录	ysyx_220046-王凯-中国科学技术大学-读后感	ysyx_220046-王凯-中国科学技术大学-实验报告		2022.3.20	2022.4.24请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040073	关嵩润	华南理工大学	集成电路工程	研一	ysyx_2204073-关嵩润-华南理工大学-学习记录	ysyx_2204073-关嵩润-华南理工大学-读后感	ysyx_2204073-关嵩润-华南理工大学-实验报告		2022.3.20	2022.4.24请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040096	叶晓豪	杭州电子科技大学	软件工程	大二	ysyx_220096-叶晓豪-杭州电子科技大学-学习记录	ysyx_220096-叶晓豪-杭州电子科技大学-读后感	ysyx_220096-叶晓豪-杭州电子科技大学-实验报告		2022.3.20	2022.4.10请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040080	孙伟华	电信科学技术研究院	电子科学与技术	研三	ysyx_220080-孙伟华-电信科学技术研究院-学习记录	ysyx_220080-孙伟华-电信科学技术研究院-读后感	ysyx_220080-孙伟华-电信科学技术研究院-实验报告		2022.3.20	2022.4.24请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040040	祝静	天津理工大学	集成电路工程	研一	ysyx_220040-祝静-天津理工大学-学习记录	ysyx_220040-祝静-天津理工大学-读后感	ysyx_220040-祝静-天津理工大学-实验报告		2022.3.20	2022.4.24请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040070	杨利民	北京工商大学	软件工程	大二	ysyx_220070-杨利民-北京工商大学-学习记录	ysyx_220070-杨利民-北京工商大学-读后感	ysyx_220070-杨利民-北京工商大学-实验报告		2022.3.20	2022.4.24请假	PA2	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040163	王宇宇	南开大学	计算机科学与技术	大四	ysyx_2204163-王宇宇-南开大学-学习记录	ysyx_2204163-王宇宇-南开大学-读后感	ysyx_2204163-王宇宇-南开大学-实验报告		2022.3.20	2022.4.16-王宇宇-南开大学-读后感	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040068	王俊	南开大学	保研至微电子学,研究方向:	大四	ysyx_220068-王俊-南开大学-学习记录	ysyx_220068-王俊-南开大学-读后感	ysyx_220068-王俊-南开大学-实验报告		2022.3.20	2022.4.16-王俊-南开大学-读后感	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040188	徐伟伟	西安电子科技大学	信息与通信工程	研一	ysyx_2204188-徐伟伟-西安电子科技大学-学习记录	ysyx_2204188-徐伟伟-西安电子科技大学-读后感	ysyx_2204188-徐伟伟-西安电子科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22042141	朱吉春	哈尔滨工业大学	电子信息	研一	ysyx_22042141-朱吉春-哈尔滨工业大学-学习记录	ysyx_22042141-朱吉春-哈尔滨工业大学-读后感	ysyx_22042141-朱吉春-哈尔滨工业大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040539	周家军	清华大学	电子科学与技术	大四	ysyx_22040539-周家军-清华大学-学习记录	ysyx_22040539-周家军-清华大学-读后感	ysyx_22040539-周家军-清华大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040017	智博元	东北大学秦皇岛分校	计算机科学与技术	大二	ysyx_220017-智博元-东北大学秦皇岛分校-学习记录	ysyx_220017-智博元-东北大学秦皇岛分校-读后感	ysyx_220017-智博元-东北大学秦皇岛分校-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040067	李泽琪	国防科技大学	计算机科学与技术	大二	ysyx_220067-李泽琪-国防科技大学-学习记录	ysyx_220067-李泽琪-国防科技大学-读后感	ysyx_220067-李泽琪-国防科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040161	蒋海龙	深圳大学	计算机科学与技术	大四	ysyx_220001-蒋海龙-深圳大学-学习记录	ysyx_220001-蒋海龙-深圳大学-读后感	ysyx_220001-蒋海龙-深圳大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040178	王九龙	北京邮电大学	电子科学与技术	大三	ysyx_22040178-王九龙-北京邮电大学-学习记录	ysyx_22040178-王九龙-北京邮电大学-读后感	ysyx_22040178-王九龙-北京邮电大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040374	崔洪波	南京大学	软件工程	大二	ysyx_22040374-崔洪波-南京大学-学习记录	ysyx_22040374-崔洪波-南京大学-读后感	ysyx_22040374-崔洪波-南京大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_2204154	徐伟华	东南大学	电子科学与技术	大四	ysyx_2204154-徐伟华-东南大学-学习记录	ysyx_2204154-徐伟华-东南大学-读后感	ysyx_2204154-徐伟华-东南大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040193	苑东强	中国科技大学	电子信息	研二	ysyx_22040193-苑东强-中国科技大学-学习记录	ysyx_22040193-苑东强-中国科技大学-读后感	ysyx_22040193-苑东强-中国科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040142	刘峰	上海海洋大学	计算机科学与技术	大三	ysyx_22040142-刘峰-上海海洋大学-学习记录	ysyx_22040142-刘峰-上海海洋大学-读后感	ysyx_22040142-刘峰-上海海洋大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040200	乌云其木格	北京协和医学院	计算机科学与技术	大三	ysyx_22040200-乌云其木格-北京协和医学院-学习记录	ysyx_22040200-乌云其木格-北京协和医学院-读后感	ysyx_22040200-乌云其木格-北京协和医学院-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040414	林润闻	广东工业大学	集成电路设计与集成系统	大三	ysyx_22040414-林润闻-广东工业大学-学习记录	ysyx_22040414-林润闻-广东工业大学-读后感	ysyx_22040414-林润闻-广东工业大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040413	顾振华	电子科技大学	电子信息工程	研二	ysyx_22040413-顾振华-电子科技大学-学习记录	ysyx_22040413-顾振华-电子科技大学-读后感	ysyx_22040413-顾振华-电子科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040595	邓继伟	华中科技大学	电子科学与技术	大四	ysyx_22040595-邓继伟-华中科技大学-学习记录	ysyx_22040595-邓继伟-华中科技大学-读后感	ysyx_22040595-邓继伟-华中科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040301	于皓哲	沈阳工业大学	电子科学与技术	研二	ysyx_22040301-于皓哲-沈阳工业大学-学习记录	ysyx_22040301-于皓哲-沈阳工业大学-读后感	ysyx_22040301-于皓哲-沈阳工业大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040562	李华助	华中科技大学	集成电路设计与集成系统	大三	ysyx_22040562-李华助-华中科技大学-学习记录	ysyx_22040562-李华助-华中科技大学-读后感	ysyx_22040562-李华助-华中科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040228	李锐琪	齐鲁理工学院	电子信息工程	大二	ysyx_22040228-李锐琪-齐鲁理工学院-学习记录	ysyx_22040228-李锐琪-齐鲁理工学院-读后感	ysyx_22040228-李锐琪-齐鲁理工学院-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040221	郭明星	山东科技大学	计算机技术	研一	ysyx_22040221-郭明星-山东科技大学-学习记录	ysyx_22040221-郭明星-山东科技大学-读后感	ysyx_22040221-郭明星-山东科技大学-实验报告		2022.3.20	2022.4.20请假	PA1	202.2022.2022.7.3*	2022.7.17请假	2022.8.7请假	2022.8.14请假											
ysyx_22040140	周治杰	浙江工业大学	计算机科学与技术	大四	ysyx_22040140-周治杰-浙江工业大学-学习记录	ysyx_22																				

Learning note of a student

日期	计划任务	总时长	任务完成情况		卡了一段时间的bug
			开始时间	结束时间	
第一周 (5天)	安装Verilator	2月25日	尝试完成MIPS的模拟	2月28日	基础部分MIPS的逻辑不大，实现起来的思路也大同小异
	调试	2月26日	完成MIPS的浮点计算	2月29日	完成浮点寄存器，可以将32位和64位浮点数在寄存器中
	完成实验报告	2月27日	完成MIPS的浮点	3月1日	貌似MIPS的JISA没有更新，直接rate合作错了。重看了报错，是自己没有看懂注释
	完成	3月1日	完成MIPS的浮点	3月3日	
	准备	3月4日	完成C3R指令	4月1日	
	完成C3R	2月1日	完成C3R指令	4月2日	
	修正bug	2月8日	修正bug	4月3日	
	完成	2月9日	清明放假	4月4日	
	引入PA	3月10日	FFMA造型+拉	4月5日	
	完成	3月11日	完成CORAL内存	4月6日	
第二周 (7天)	进一步	2月12日	完成Flash	4月7日	
	完成	2月14日	修正上电时序	4月8日	
	完成	2月15日	修正上电时序	4月9日	
	完成	2月16日	完成WIFI	4月10日	
	完成	3月17日	完成	4月11日	
	完成	3月18日	完成	4月12日	
	完成	3月19日	完成PCI	4月13日	
	完成	3月20日	完成strace, 完成	4月14日	
	完成	3月21日	完成中	4月15日	
	完成	3月22日	完成取指&译码	4月16日	
第三周 (7天)	测试中断	2月22日	完成取指&译码	4月17日	
	完成	3月23日	完成取指&译码	4月18日	
	完成	3月24日	完成取指&译码	4月19日	
	完成	3月25日	完成取指&译码	4月20日	
	完成	3月26日	尝试引脚	4月21日	
	完成累加	3月27日	完成累加	4月22日	
	完成	4月1日	完成Teardown	4月23日	
	完成	4月2日	完成Teardown	4月24日	
	完成	4月3日	完成Teardown	4月25日	
	完成	4月4日	完成Teardown	4月26日	
第四周 (7天)	五一放假	4月27日	编写计分器	4月27日	
	完成	4月28日	学习香山代码	4月28日	
	完成	4月29日	学习香山代码	4月29日	
	完成	4月30日	学习香山代码	4月30日	
	完成	5月1日	学习香山代码	5月1日	
	完成	5月2日	学习香山代码	5月2日	
	完成	5月3日	学习香山代码	5月3日	
	完成	5月4日	学习香山代码	5月4日	
	五一放假	5月4日	学习香山代码	5月4日	
	五一放假	5月4日	学习香山代码	5月4日	
第五周 (6天)	第五周 (6天)	2月28日	尝试完成MIPS的模拟	3月1日	基础部分MIPS的逻辑不大，实现起来的思路也大同小异
	第六周 (6天)	2月29日	完成MIPS的浮点计算	3月1日	完成浮点寄存器，可以将32位和64位浮点数在寄存器中
	第七周 (7天)	3月1日	完成MIPS的浮点	3月3日	貌似MIPS的JISA没有更新，直接rate合作错了。重看了报错，是自己没有看懂注释
	第八周 (6天)	3月3日	完成MIPS的浮点	3月4日	
	第九周 (4天)	3月4日	完成C3R指令	4月1日	
	第十周 (7天)	4月1日	完成C3R指令	4月2日	
	第十一周 (7天)	4月2日	完成C3R指令	4月3日	
	第十二周 (7天)	4月3日	完成C3R指令	4月4日	
	第十三周 (4天)	4月4日	完成C3R指令	4月5日	
	第十四周 (7天)	4月5日	修改SpinalHDL	5月5日	貌似MIPS的JISA没有更新，直接rate合作错了。重看了报错，是自己没有看懂注释
第十五周 (7天)	第十五周 (7天)	5月5日	修改SpinalHDL	5月6日	学习
	第十六周 (7天)	5月6日	修改SpinalHDL	5月7日	学习
	第十七周 (7天)	5月7日	修改SpinalHDL	5月8日	学习《超时》
	第十八周 (7天)	5月8日	修改SpinalHDL	5月9日	调试
	第十九周 (7天)	5月9日	修改SpinalHDL	5月10日	调试AXI总线
	第二十周 (7天)	5月10日	修改AXI总线	5月11日	调试AXI总线
	第二十一周 (2天)	5月11日	修改AXI总线	5月12日	调试AXI总线
	第二十二周 (7天)	5月12日	修改AXI总线	5月13日	学习《超时》
	第二十三周 (7天)	5月13日	修改AXI总线	5月14日	学习《超时》
	第二十四周 (7天)	5月14日	修改AXI总线	5月15日	编写fix
第二十五周 (7天)	第二十五周 (7天)	5月15日	修改AXI总线	5月16日	切换到Lattice芯片
	第二十六周 (7天)	5月16日	修改AXI总线	5月17日	学习《超时》
	第二十七周 (7天)	5月17日	修改AXI总线	5月18日	学习Reindl
	第二十八周 (7天)	5月18日	修改AXI总线	5月19日	学习Reindl
	第二十九周 (7天)	5月19日	修改AXI总线	5月20日	学习Reindl
	第三十周 (7天)	5月20日	修改AXI总线	5月21日	学习Reindl
	第三十一周 (2天)	5月21日	修改AXI总线	5月22日	学习《超时》
	第三十二周 (7天)	5月22日	修改AXI总线	5月23日	学习《超时》
	第三十三周 (7天)	5月23日	修改AXI总线	5月24日	修改串行器
	第三十四周 (7天)	5月24日	修改AXI总线	5月25日	使用IP核模块
第三十五周 (7天)	第三十五周 (7天)	5月25日	修改AXI总线	5月26日	提升CPU频率
	第三十六周 (7天)	5月26日	修改AXI总线	5月27日	
	第三十七周 (7天)	5月27日	修改AXI总线	5月28日	
	第三十八周 (7天)	5月28日	修改AXI总线	5月29日	电源休息
	第三十九周 (7天)	5月29日	修改AXI总线	5月30日	
	第四十周 (7天)	5月30日	修改AXI总线	5月31日	
	第四十一周 (2天)	5月31日	修改AXI总线	6月1日	
	第四十二周 (7天)	6月1日	分离AFB总线时钟	6月2日	
	第四十三周 (7天)	6月2日	分离AFB总线时钟	6月3日	
	第四十四周 (7天)	6月3日	分离AFB总线时钟	6月4日	
第四十五周 (7天)	第四十五周 (7天)	6月4日	分离AFB总线时钟	6月5日	
	第四十六周 (7天)	6月5日	分离AFB总线时钟	6月6日	
	第四十七周 (7天)	6月6日	分离AFB总线时钟	6月7日	
	第四十八周 (7天)	6月7日	分离AFB总线时钟	6月8日	
	第四十九周 (7天)	6月8日	分离AFB总线时钟	6月9日	
	第五十周 (7天)	6月9日	分离AFB总线时钟	6月10日	
	第五十一周 (7天)	6月10日	分离AFB总线时钟	6月11日	
	第五十二周 (7天)	6月11日	分离AFB总线时钟	6月12日	
	第五十三周 (7天)	6月12日	分离AFB总线时钟	6月13日	
	第五十四周 (7天)	6月13日	分离AFB总线时钟	6月14日	
第五十五周 (7天)	第五十五周 (7天)	6月14日	分离AFB总线时钟	6月15日	
	第五十六周 (7天)	6月15日	分离AFB总线时钟	6月16日	
	第五十七周 (7天)	6月16日	分离AFB总线时钟	6月17日	
	第五十八周 (7天)	6月17日	分离AFB总线时钟	6月18日	
	第五十九周 (7天)	6月18日	分离AFB总线时钟	6月19日	
	第六十周 (7天)	6月19日	分离AFB总线时钟	6月20日	
	第六十一周 (7天)	6月20日	分离AFB总线时钟	6月21日	
	第六十二周 (7天)	6月21日	分离AFB总线时钟	6月22日	
	第六十三周 (7天)	6月22日	分离AFB总线时钟	6月23日	
	第六十四周 (7天)	6月23日	分离AFB总线时钟	6月24日	
第六十五周 (7天)	第六十五周 (7天)	6月24日	分离AFB总线时钟	6月25日	
	第六十六周 (7天)	6月25日	分离AFB总线时钟	6月26日	
	第六十七周 (7天)	6月26日	分离AFB总线时钟	6月27日	
	第六十八周 (7天)	6月27日	分离AFB总线时钟	6月28日	
	第六十九周 (7天)	6月28日	分离AFB总线时钟	6月29日	
	第七十周 (7天)	6月29日	分离AFB总线时钟	6月30日	
	第七十一周 (7天)	6月30日	分离AFB总线时钟	6月31日	
	第七十二周 (7天)	6月31日	分离AFB总线时钟	7月1日	
	第七十三周 (7天)	7月1日	分离AFB总线时钟	7月2日	
	第七十四周 (7天)	7月2日	分离AFB总线时钟	7月3日	
第七十五周 (7天)	第七十五周 (7天)	7月3日	分离AFB总线时钟	7月4日	
	第七十六周 (7天)	7月4日	分离AFB总线时钟	7月5日	
	第七十七周 (7天)	7月5日	分离AFB总线时钟	7月6日	
	第七十八周 (7天)	7月6日	分离AFB总线时钟	7月7日	
	第七十九周 (7天)	7月7日	分离AFB总线时钟	7月8日	
	第八十周 (7天)	7月8日	分离AFB总线时钟	7月9日	
	第八十一周 (7天)	7月9日	分离AFB总线时钟	7月10日	
	第八十二周 (7天)	7月10日	分离AFB总线时钟	7月11日	
	第八十三周 (7天)	7月11日	分离AFB总线时钟	7月12日	
	第八十四周 (7天)	7月12日	分离AFB总线时钟	7月13日	
第八十五周 (7天)	第八十五周 (7天)	7月13日	分离AFB总线时钟	7月14日	
	第八十六周 (7天)	7月14日	分离AFB总线时钟	7月15日	
	第八十七周 (7天)	7月15日	分离AFB总线时钟	7月16日	
	第八十八周 (7天)	7月16日	分离AFB总线时钟	7月17日	
	第八十九周 (7天)	7月17日	分离AFB总线时钟	7月18日	
	第九十周 (7天)	7月18日	分离AFB总线时钟	7月19日	
	第九十一周 (7天)	7月19日	分离AFB总线时钟	7月20日	
	第九十二周 (7天)	7月20日	分离AFB总线时钟	7月21日	
	第九十三周 (7天)	7月21日	分离AFB总线时钟	7月22日	
	第九十四周 (7天)	7月22日	分离AFB总线时钟	7月23日	
第九十五周 (7天)	第九十五周 (7天)	7月23日	分离AFB总线时钟	7月24日	
	第九十六周 (7天)	7月24日	分离AFB总线时钟	7月25日	
	第九十七周 (7天)	7月25日	分离AFB总线时钟	7月26日	
	第九十八周 (7天)	7月26日	分离AFB总线时钟	7月27日	
	第九十九周 (7天)	7月27日	分离AFB总线时钟	7月28日	
	第一百周 (7天)	7月28日	分离AFB总线时钟	7月29日	
	第一百零一周 (7天)	7月29日	分离AFB总线时钟	7月30日	
	第一百零二周 (7天)	7月30日	分离AFB总线时钟	7月31日	
	第一百零三周 (7天)	7月31日	分离AFB总线时钟	8月1日	
	第一百零四周 (7天)	8月1日	分离AFB总线时钟	8月2日	
第一百零五周 (7天)	第一百零五周 (7天)	8月2日	分离AFB总线时钟	8月3日	
	第一百零六周 (7天)	8月3日	分离AFB总线时钟	8月4日	
	第一百零七周 (7天)	8月4日	分离AFB总线时钟	8月5日	
	第一百零八周 (7天)	8月5日	分离AFB总线时钟	8月6日	
	第一百零九周 (7天)	8月6日	分离AFB总线时钟	8月7日	
	第一百一十周 (7天)	8月7日	分离AFB总线时钟	8月8日	
	第一百一一周 (7天)	8月8日	分离AFB总线时钟	8月9日	
	第一百一两周 (7天)	8月9日	分离AFB总线时钟	8月10日	
	第一百一三周 (7天)	8月10日	分离AFB总线时钟	8月11日	
	第一百一四周 (7天)	8月11日	分离AFB总线时钟	8月12日	
第一百一十五周 (7天)	第一百一十五周 (7天)	8月12日	分离AFB总线时钟	8月13日	
	第一百一十六周 (7天)	8月13日	分离AFB总线时钟	8月14日	
	第一百一十七周 (7天)	8月14日	分离AFB总线时钟	8月15日	
	第一百一十八周 (7天)	8月15日	分离AFB总线时钟	8月16日	
	第一百一十九周 (7天)	8月16日	分离AFB总线时钟	8月17日	
	第一百二十周 (7天)	8月17日	分离AFB总线时钟	8月18日	
	第一百二十一周 (7天)	8月18日	分离AFB总线时钟	8月19日	
	第一百二十二周 (7天)	8月19日	分离AFB总线时钟	8月20日	
	第一百二十三周 (7天)	8月20日	分离AFB总线时钟	8月21日	
	第一百二十四周 (7天)	8月21日	分离AFB总线时钟	8月22日	
第一百二十五周 (7天)	第一百二十五周 (7天)	8月22日	分离AFB总线时钟	8月23日	
	第一百二十六周 (7天)	8月23日	分离AFB总线时钟	8月24日	
	第一百二十七周 (7天)	8月24日	分离AFB总线时钟	8月25日	
	第一百二十八周 (7天)	8月25日	分离AFB总线时钟	8月26日	
	第一百二十九周 (7天)	8月26日	分离AFB总线时钟	8月27日	
	第一百三十周 (7天)	8月27日	分离AFB总线时钟	8月28日	
	第一百三十一周 (7天)	8月28日	分离AFB总线时钟	8月29日	
	第一百三十二周 (7天)	8月29日	分离AFB总线时钟	8月30日	
	第一百三十三周 (7天)	8月30日	分离AFB总线时钟	8月31日	
	第一百三十四周 (7天)	8月31日	分离AFB总线时钟	8月32日	
第一百三十五周 (7天)	第一百三十五周 (7天)	8月32日	分离AFB总线时钟	8月33日	
	第一百三十六周 (7天)	8月33日	分离AFB总线时钟	8月34日	
	第一百三十七周 (7天)	8月34日	分离AFB总线时钟	8月35日	
	第一百三十八周 (7天)	8月35日	分离AFB总线时钟	8月36日	
	第一百三十九周 (7天)	8月36日	分离AFB总线时钟	8月37日	

3rd One Student One Chip

■ Tape out

- 12/2021: 39 cores
- 02/2022: 9 cores

■ Breakdown

- EE/IC: 25; CS/SE: 13; Others: 4
- Freshman (2) , Sophomore (3) , Junior (11) , Senior (3)
- Grade 1 master (8), Grade 2 master (11), Grade 3 master (1)
- Ph.d Candidate(4)

■ Micro-architecture Design

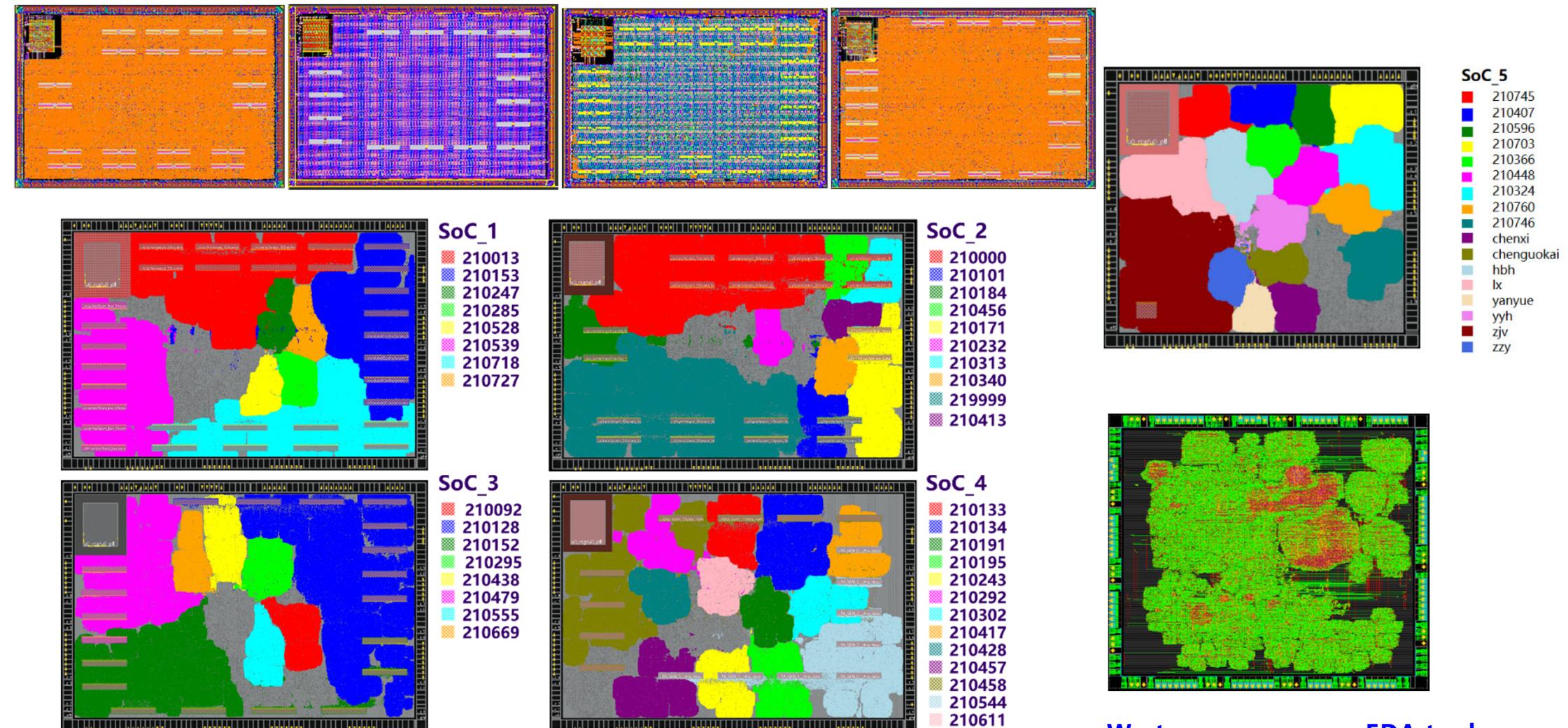
- Branch Prediction: 12;
- OoO: 3;
- Chisel: 16; Verilog/sv: 31
- 5-stage pipeline with Cache: 12

Water Conservancy

University	Grade	Major
ysyx_210092 西安电子科技大学	大三 (即将升计算机科学与技术专业硕)	
ysyx_210456 电子科技大学	研一	电子信息科学与技术
ysyx_210247 南京理工大学	研一	电子信息
ysyx_210243 华中科技大学	大三	电子信息与通信工程
ysyx_210544 南京航空航天大学	博士一年级	软件工程
ysyx_210232 青岛科技大学	大三	集成电路设计与集成系统
ysyx_210295 华东师范大学	研一	集成电路设计与集成系统
ysyx_210457 山东交通学院	大一	电子信息工程
ysyx_210458 太原理工大学	大二	水利
ysyx_210611 南京大学	大一	计算机科学与技术
ysyx_210285 南京大学	大二 (准大三)	计算机科学与技术
ysyx_210128 上海交通大学	大四	电子与计算机工程
ysyx_210727 华中科技大学	研二	计算机科学与技术
ysyx_210718 深圳大学	研二	电子信息
ysyx_210133 电子科技大学	研二	电子科学与工程学院

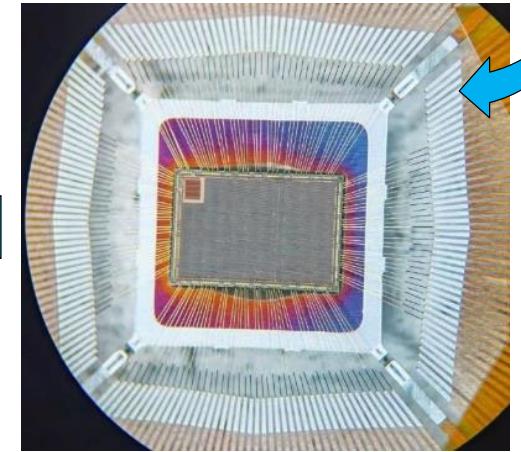
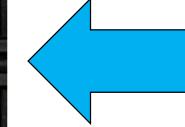
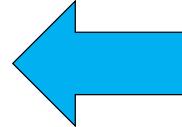
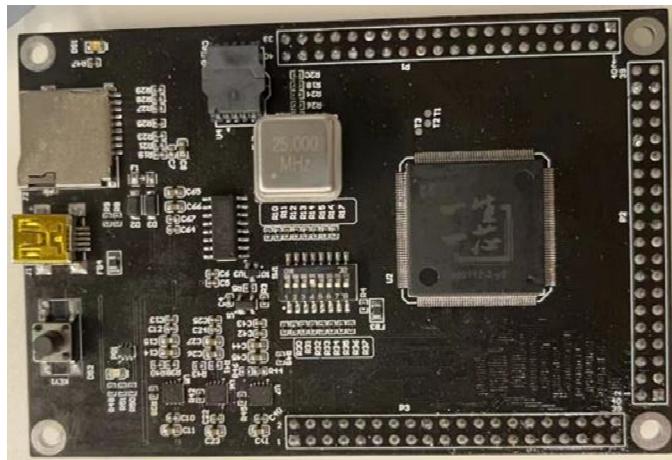
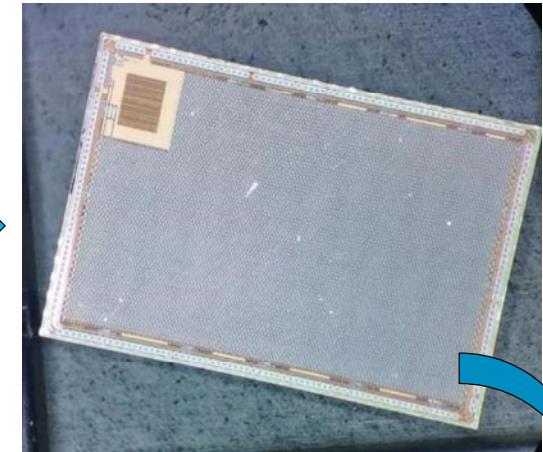
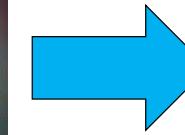
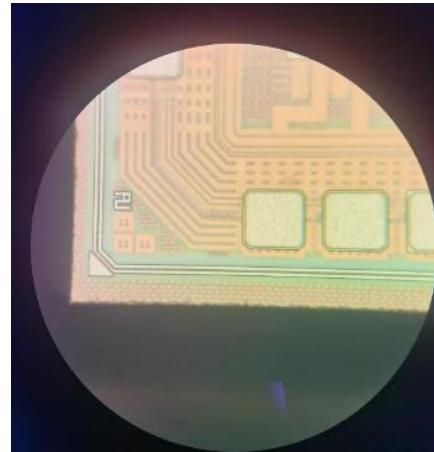
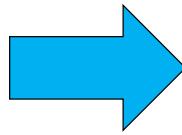
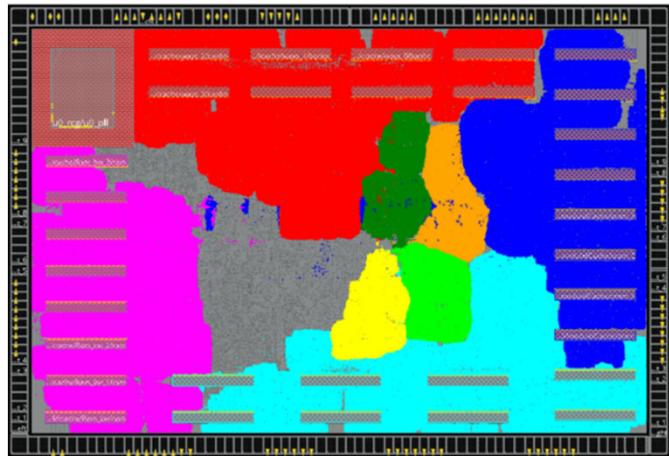
ysyx_210292 集美大学	大四	计算机科学与技术
ysyx_210191 南京理工大学	大三	计算机科学与技术
ysyx_210195 西安电子科技大学	硕士二年级	电子科学与技术
ysyx_210413 大连理工大学	研一	软件工程
ysyx_210428 沈阳工业大学	研二	电子科学与技术
ysyx_210313 电子科技大学	大三	微电子
ysyx_210302 复旦大学	研一	微电子学与固体电子学
ysyx_210184 清华大学	研二	集成电路工程
ysyx_210479 太原理工大学	大三	计算机科学与技术
ysyx_210013 西安交通大学	研二	微电子学与固体电子学
ysyx_210438 南京大学	直博二年级	电子信息技术
ysyx_210555 南京大学	研一	集成电路工程
ysyx_210528 中国农业大学	大三	电子信息工程
ysyx_210669 北京工业大学	研二	计算机技术
ysyx_210417 中国科学技术大学	研一	集成电路工程
ysyx_210134 浙江大学	大三	计算机科学与技术
ysyx_210152 重庆邮电大学	大三	电子信息工程

Taping out of 3rd OSOC



We try open-source EDA tools

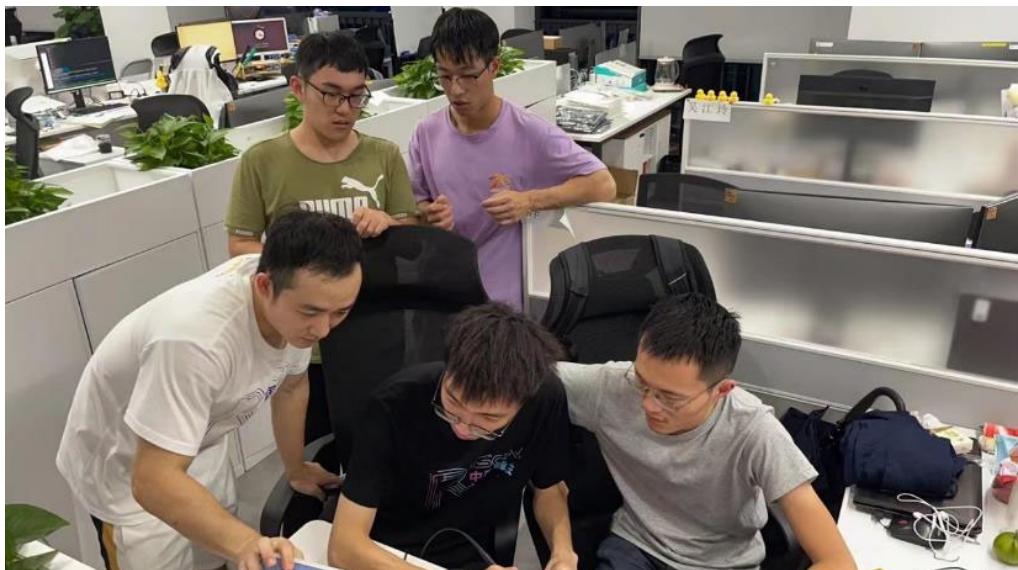
Chip & Board



PCB Testing

Testing team

- 黄健明(海南大学)
- 卢非凡(西安财经大学)
- 马壮(中国科学技术大学)
- 缪宇驰(鹏城实验室)
- 许立达(中科院微电子所)



Software Testing

	hello world	memtest
flash	[10:59:26.279]收←◆Hello World! [11:04:04.430]收←◆Hello World! [11:04:06.275]收←◆Hello World!	[14:51:55.890]收←◆start test... mem tests prepared mem tests passed!!
mem	[14:19:33.762]收←◆Loading program of size: 208 bytes, expect 128 '# Loading.... ##### Load finished Exec app... Hello World!	[15:06:04.482]收←◆Loading program of size: 3840 bytes, expect 128 '# Loading.... ##### [15:06:04.745]收←◆##### Load finished Exec app... start test... mem tests prepared mem tests passed!!

Memory access

```
[10:31:49.356]收←◆[mem data] cnt: 65929216(3ee0000), addr: 0x9f900000
[10:31:49.416]收←◆[mem data] cnt: 65994752(3ef0000), addr: 0x9f980000
[10:31:49.476]收←◆[mem data] cnt: 66080288(3f00000), addr: 0x9fa00000
[10:31:49.535]收←◆[mem data] cnt: 66125824(3f10000), addr: 0x9fa80000
[10:31:49.594]收←◆[mem data] cnt: 66191360(3f20000), addr: 0x9fb00000
[10:31:49.653]收←◆[mem data] cnt: 66256896(3f30000), addr: 0x9fb80000
[10:31:49.712]收←◆[mem data] cnt: 66322432(3f40000), addr: 0x9fc00000
[10:31:49.772]收←◆[mem data] cnt: 66387968(3f50000), addr: 0x9fc80000
[10:31:49.830]收←◆[mem data] cnt: 66453504(3f60000), addr: 0x9fd00000
[10:31:49.889]收←◆[mem data] cnt: 66519040(3f70000), addr: 0x9fd80000
[10:31:49.949]收←◆[mem data] cnt: 66584576(3f80000), addr: 0x9fe00000
[10:31:50.007]收←◆[mem data] cnt: 66650112(3f90000), addr: 0x9fe80000
[10:31:50.067]收←◆[mem data] cnt: 66715648(3fa0000), addr: 0x9ff00000
[10:31:50.126]收←◆[mem data] cnt: 66781184(3fb0000), addr: 0x9ff80000
[10:31:50.185]收←◆mem tests passed!!
[10:32:25.838]收←◆\0\0
```

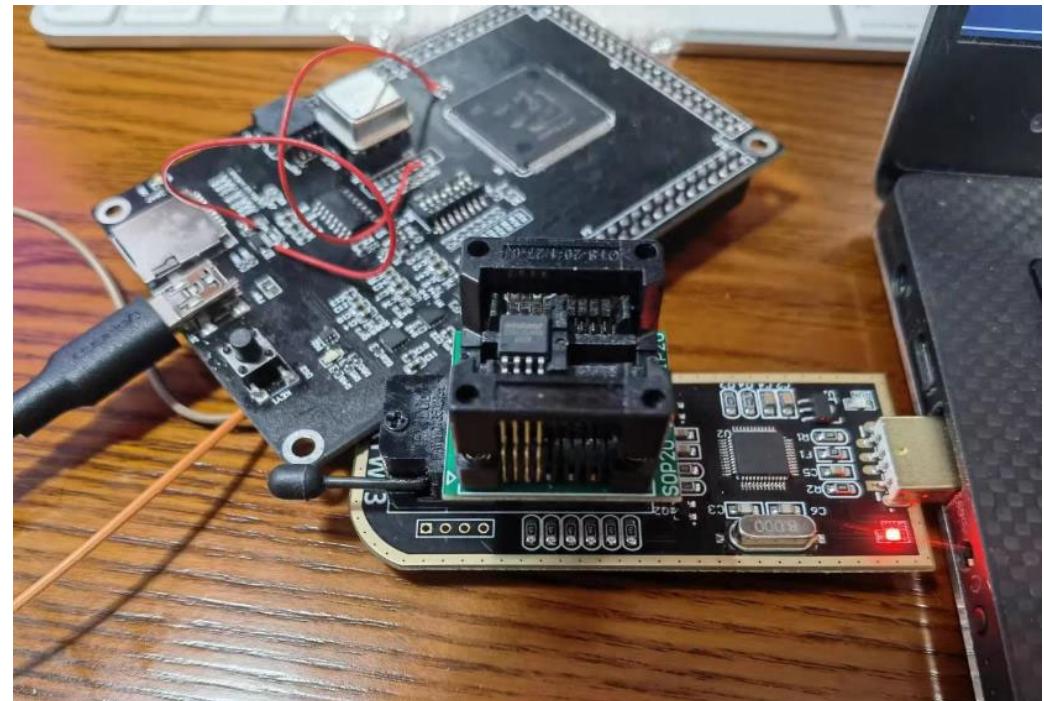
Booting
RT-Thread

```
Load finished
Exec app...
heap: [0x80022590 - 0x86422590]
\ | /
- RT - Thread Operating System
/ | \ 4.0.4 build Nov 29 2022
2006 - 2021 Copyright by rt-thread team
Hello RISC-V!
thread1 count: 0
thread2 count: 0
msh />
[10:41:52.743]收←◆thread1 count: 1
thread2 count: 1
[10:41:53.385]收←◆thread1 count: 2
thread2 count: 2
```

Student tries at home



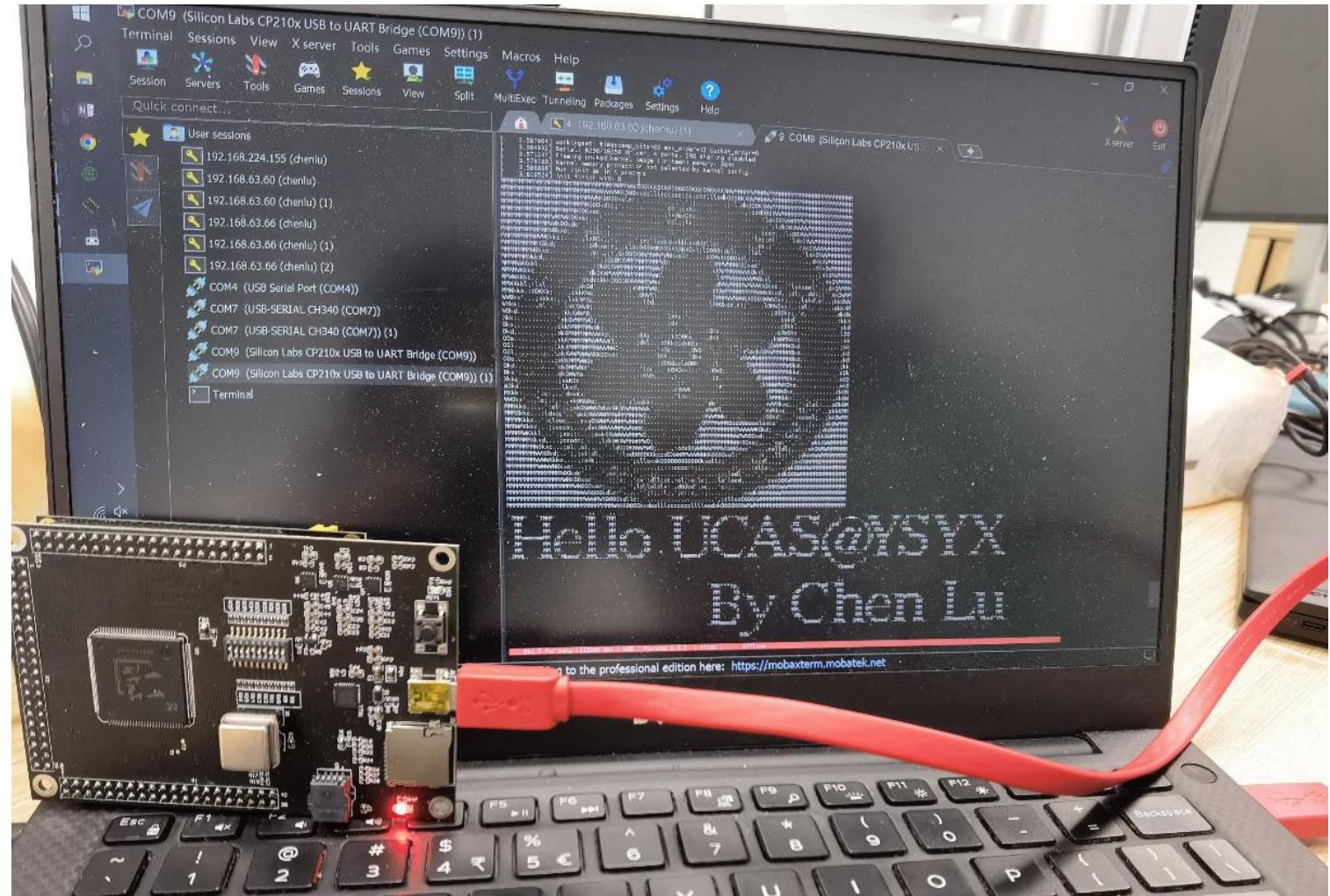
A screenshot of a software interface for memory dump analysis. The window title is '中区(B) 编程操作(O) 自动选项(A) 硬件(H) 语言(L) 关于(I)' (Central Area(B) Programming Operations(O) Automatic Options(A) Hardware(H) Language(L) About(I)). The main area displays memory dump data in a grid format. The columns are labeled 'ADDR' (Address), '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F', and 'ASCII'. The data shows a sequence of bytes followed by their ASCII representation. The bottom left of the interface shows settings: '容量: 16MB', '页大小: 256B', and '芯片电压: 3.3V'. The bottom right shows file statistics: '22 KB', '8 KB', '1 KB', '11 KB', '800 KB', '0 KB', '8 KB', '4,652 KB', and '22 KB'.



Hello, YSYX!
Have a good luck!
Have a good luck!

Demo by single student

- Lu Chen@NJU,
computer science,
learning when
junior
 - Load Linux from
flash and boot
successfully,
showing the CAS
logo



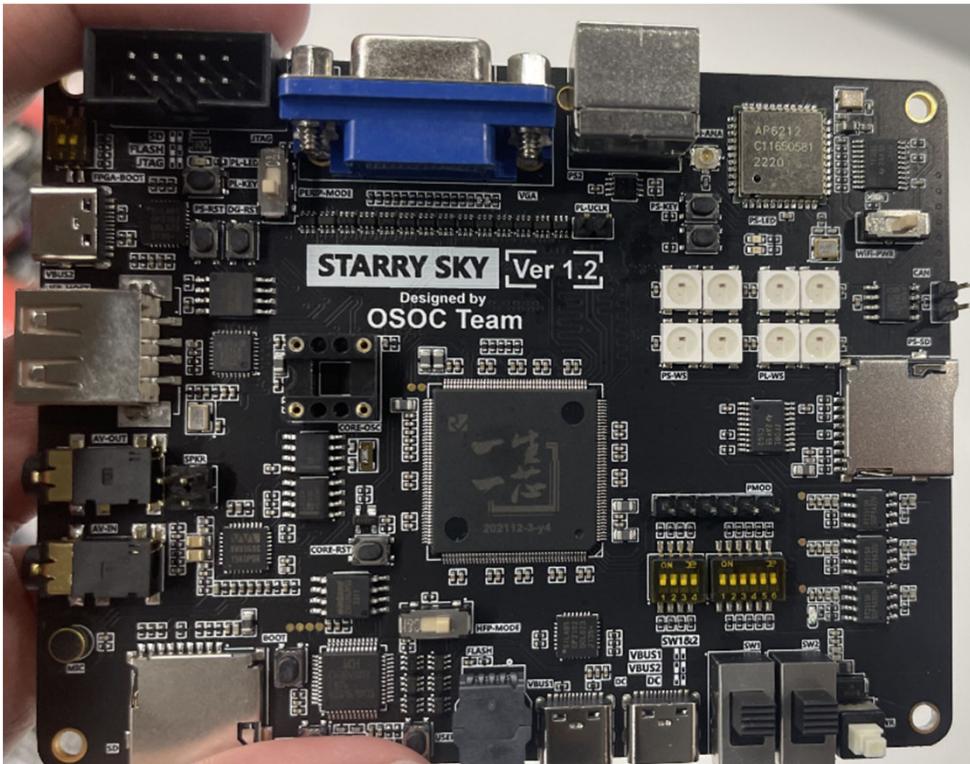
Demo by another single student

- Haojin Tang@UCAS,
Electronic engineering,
learning when junior
 - Boot Debian and run
games



<https://www.bilibili.com/video/BV1CL411X7wV>

Delivering PCB to students

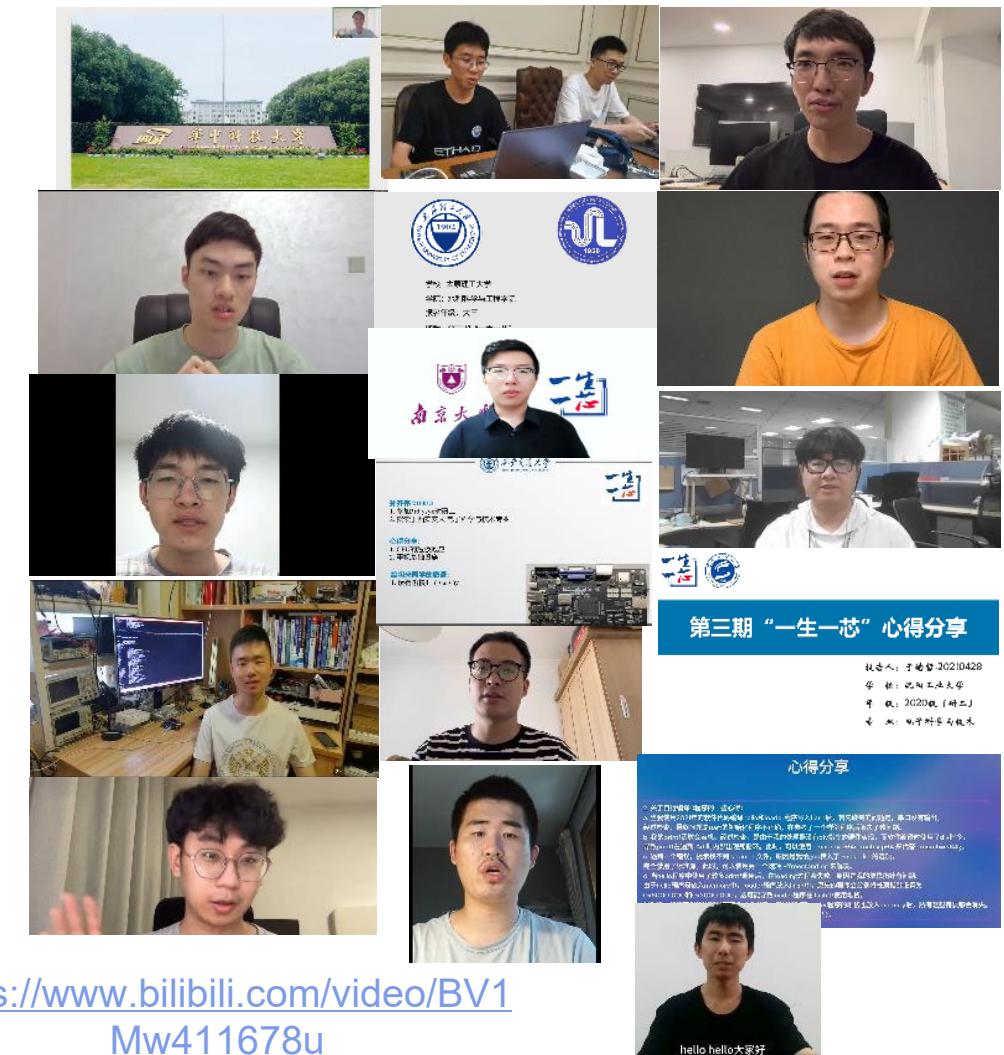
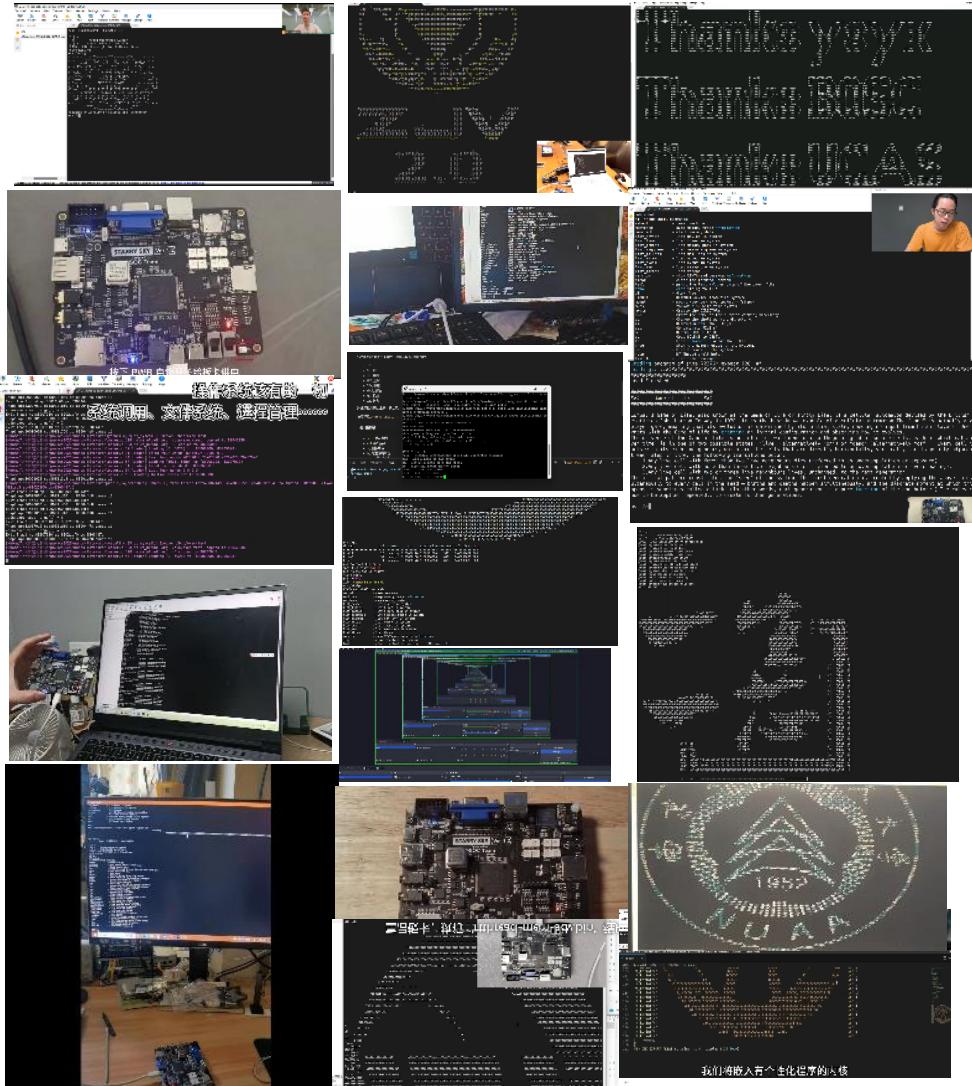


PCB designed by OSOC



Before delivering

Brought up Chips, Ran OS & Applications



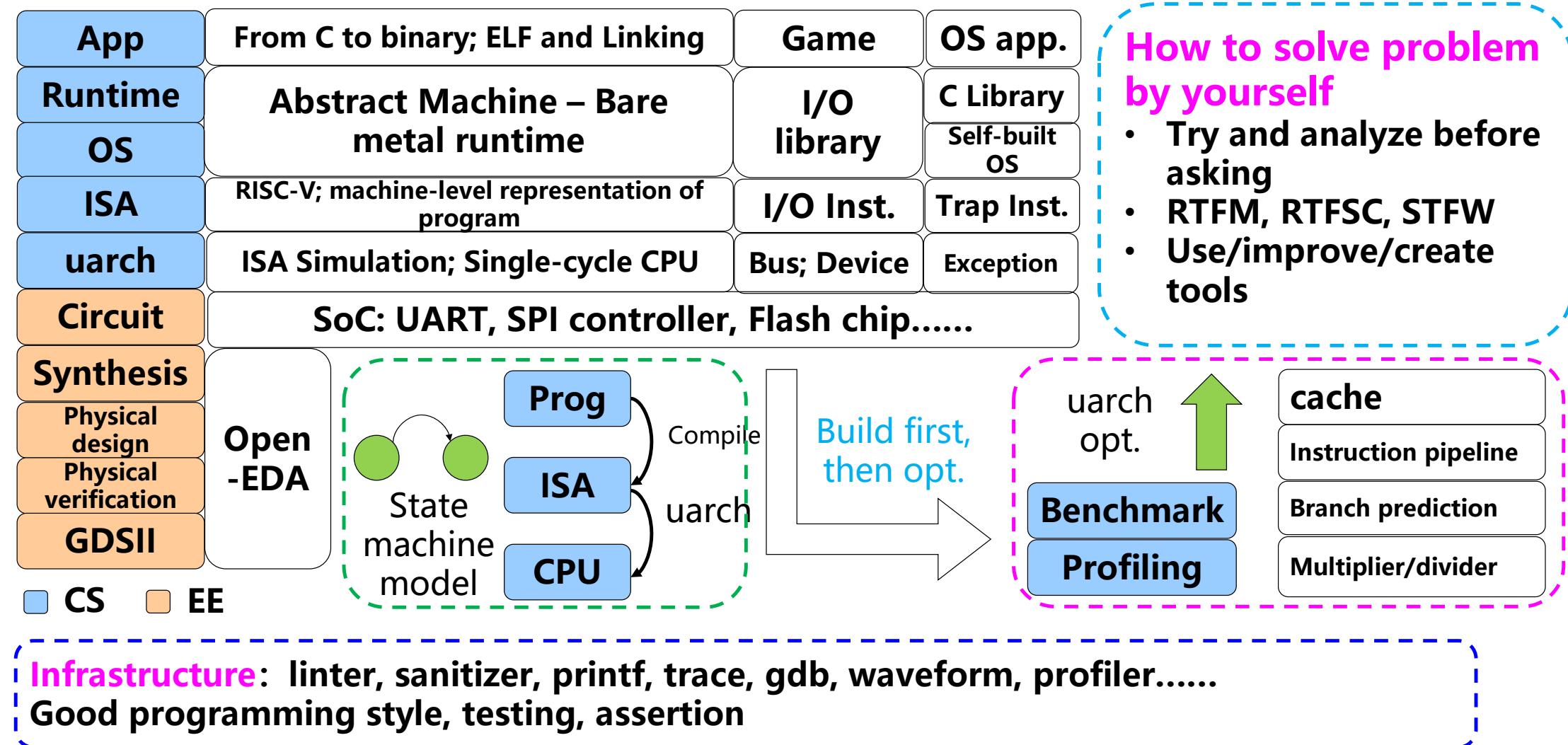
<https://www.bilibili.com/video/BV1Mw411678u>

4. 4th OSOC and after – Learning more (Feb 2022 to now)

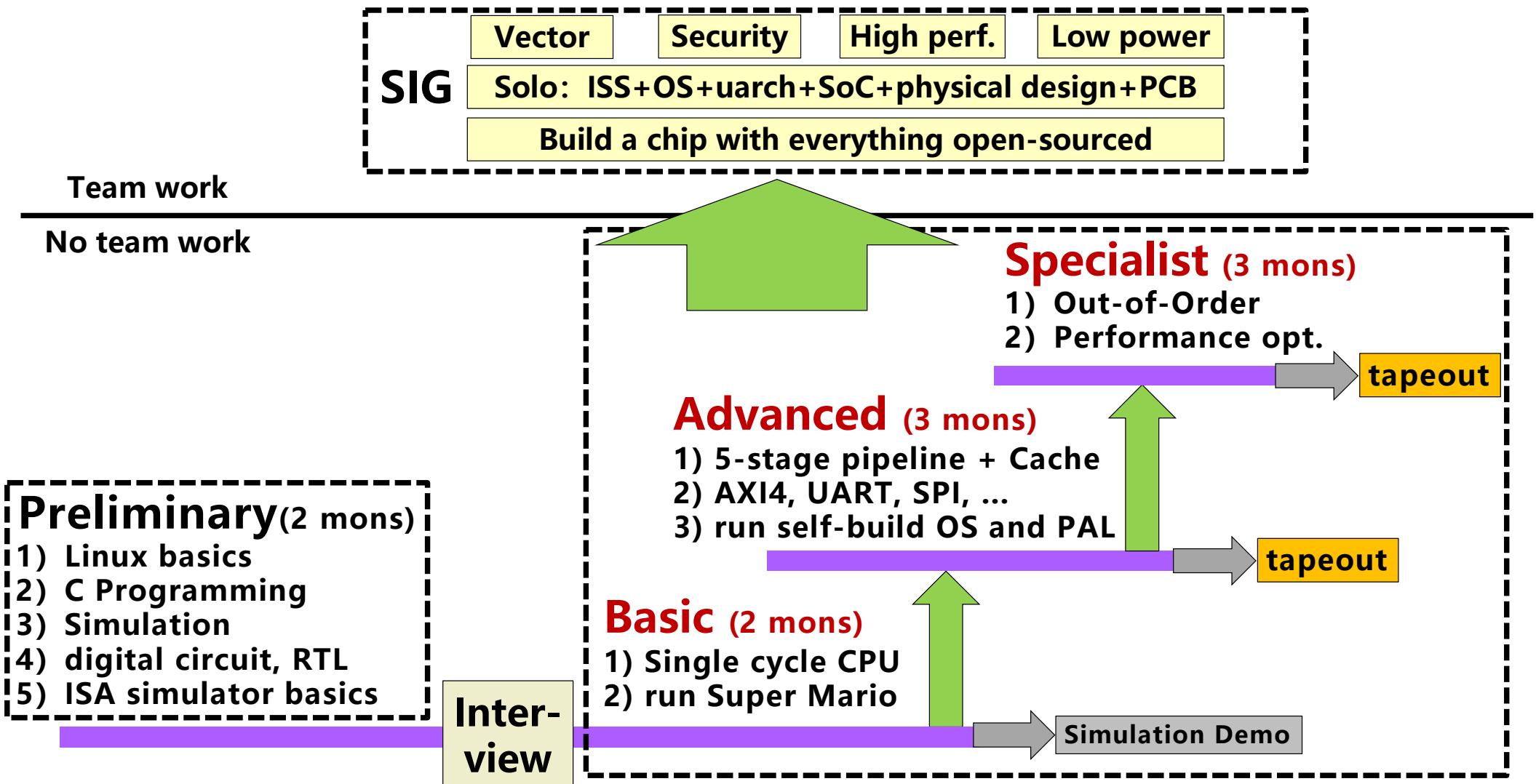
4th and 5th OSOC

- Higher goal: RV64IM, self-build OS, run non-trivial games
- Add self-build simulator and system software **practicing software-hardware co-design**
 - Build an ISA simulator, compile programs, develop a small OS and runtime
- **Better lecture note**
 - Preliminary stage/Basic stage/Advanced stage
 - Building a RISC-V system from scratch
- No team work any more
- **Videos and slides** in 5th OSOC

Knowledge Diagram



Learning stages



Learning Schedule

C = C language(sw)

R = RISC-V(ISA)

P = Processor(hw)

T = Tools(infrastructure)

RV32E
system

Intro
Simple system

阶段	序号	任务	讲义	课件	视频	C	R	P	T
预学习阶段	1	如何科学地提问	书	文	影				
	2	Linux系统安装和基本使用	书	文	影				✓
	3	计算机系统的状态机模型	-	文	影	✓	✓	✓	
	4	复习C语言	书	文	影	✓			✓
	5	程序的执行和模拟器	-	文	影	✓	✓		
	6	搭建verilator仿真环境	书	-	-	✓		✓	
	7	数字电路基础实验	书	文	影			✓	
	8	完成PA1	书	文	影	✓			✓

■申请入学答辩

9	支持RV32IM的NEMU	书	文	影	✓	✓	✓		
10	程序的机器级表示(上)	-	文	影		✓	✓		
11	程序的机器级表示(下)	-	文	影		✓	✓		
12	用RTL实现最简单的处理器	书	-	-					✓
13	AM运行时环境	书	文	影			✓		
14	工具和基础设施	-	文	影					✓
15	支持RV32E的单周期NPC	书	文	影				✓	✓
16	ELF文件和链接	-	文	影		✓	✓		
17	设备和输入输出	书	文	影		✓	✓	✓	
18	调试技巧	-	文	影					✓
19	异常处理和RT-Thread	书	文	影		✓	✓	✓	
20	总线	书							✓
21	SoC计算机系统	书							✓
22	性能和简易缓存	书							✓
23	流水线	书							✓

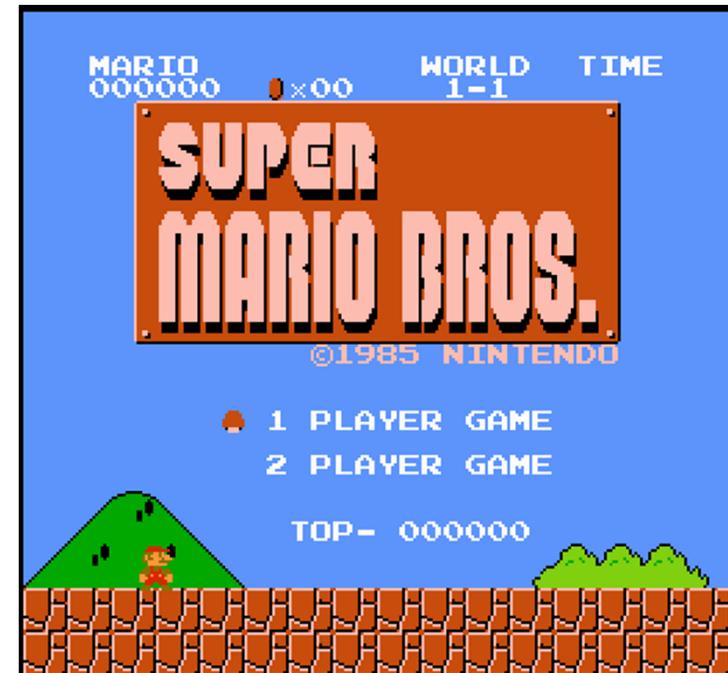
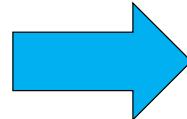
■达成B阶段流片指标

Advanced stage: M extension, S mode & U mode, A extension, xv6, Linux, uarch opt.

Demo in 4th OSOC

- Adding VGA controller

Display by UART in 3rd OSOC



Basic stage: run Super Mario (released in 1980s)

Demo in 4th OSOC

- Adding VGA controller



Advanced stage: run PAL
(a famous Chinese RPG,
released in 1990s)



Challenge task: run CLANNAD
(a Japanese game,
released in 2000s)

Learning resources are opened

第五期“一生一芯”课程主页

- 课时:每周六19:00~21:00
 - B站直播 | 录播链接
- 答疑:每周日19:00~20:00 (通过预学习答辩后由助教通知)
- 报名流程 | 报名常见问题

课件和讲义

0. C = C语言(程序/微处理器/系统软件) | R = RISC-V指令集 | P = 处理器设计 | T = 工具

预学习阶段

- “一生一芯”概述 | 如何科学地提问
- 工具是第一生产力——Linux入门教程 | Linux系统安装和基本使用
- 计算机系统的状态机模型 | 复习C语言
- 从C语言到二进制程序 | 搭建Verilator仿真环境 | 数字电路基础实验
- NEMU汇编器 | 完成PA1

A阶段

- RISC-V指令集 | 支持RV64IM的NEMU
- 程序的机器码表示
- RISC-V单周期处理器设计 | 用RTL实现最简单的处理器
- Abstract Machine裸机运行时环境 | 运行时环境和基础设施
- ELF文件和链接
- 工具和基础设施
- 设备和输入输出 | 设备和输入输出
- 调试技术进阶

A阶段

- 异常处理 | 简单的异常处理机制
- 计算机系统软件 | 用户程序与系统调用 | 精彩纷呈的用户程序



计算机系统的状态机模型

余子濠



中国科学院

计算技术研究所



计算机系统与处理器
芯片课程虚拟教研室

- Course Website**
- Lecture note(260,000 words)**
- Slides(>800 pages, 85,000 words)**
- Videos(> 40 hours)**

一生一芯 官方文档

关于项目 · 常见问题 · 隐私政策 · 其它说明 · 登录 · 新闻

预学习阶段

如何科学地提问

Linux系统安装和基本使用

安装一个Linux操作系统

我们尊重PA的讲义, 并进行更丰富PA样例代码的部分, 将会有进阶样例回到此处的讲义内容

首先运行在Bilibili上执行一个`key`脚本, 具体操作请STW, 然后通过以下命令安装一生一芯的样例代码

```
git clone -b psyx2022 git@github.com:oscc/psyx-workbench.git
```

获取后, 你可以回到PA讲义的位置, 继续阅读了! 不过你需要注意:

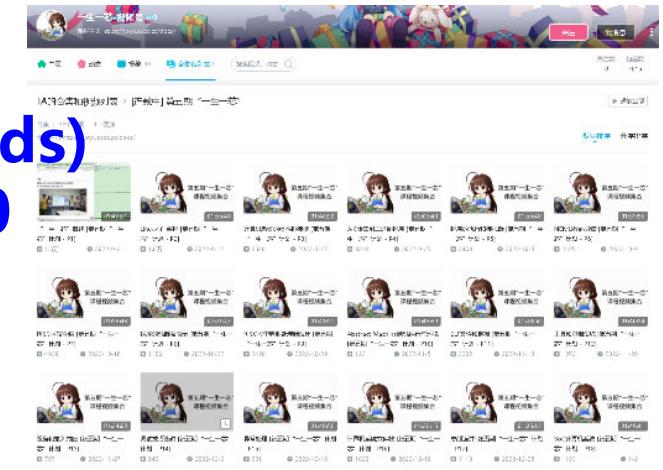
- 访问 `psyx-workbench` 为了解讲义中的命令, 切勿对文件的 `key` 做修改
- 修改 `psyx-workbench` 中的字符和字符串, 请尊重“一生一芯”的学生和真实姓名

这种来回跳转的做法可能会给你带来一些麻烦, 但我们之所以这样做, 是希望把文档和代码分离, 我们希望做到类似“一生一芯”使用内核的风格, 因此我们在PA讲义中尽可能少地提到“一生一芯”, 而把“一生一芯”的开发者放到“一生一芯”的学生和真实姓名

! 安装系统是独立解决问题的最简单的训练

如果你第一次安装并使用Linux, 你可能会遇到非常多的bug, 不用担心, 因为全世界都在用Linux, 这样你遇到的问题, 很大概率别人已经遇到, 在互联网上搜索关键词, 很大概率能解决你的问题

树立正确的价值观 接受量大程度的训练



Bilibili Account: 一生一芯-视频号

第五期一生一芯 | 周六 19:00-21:00 | Sat 08 Oct 2022 08:37:31

课程主页 <https://ysyx.oscc.cc/docs/>

```
#include <cs52.h>
int main() {
    char *s = "Hello, world!";
    int n = strlen(s);
    printf("%d\n", n);
}
```

解密黑科技 - 现代方法

使用工具查看宏展开结果

回顾: 使用gcc的-E参数可以输出预处理结果

- 但直接编译会报错: 找不到头文件

解决方案: 在Makefile文件的编译规则中添加命令

```
# Compilation patterns
$(OBJ DIR)/%: %.c
    $(echo) + CC $@
    $(mkdir) -p $(OBJ DIR)
    $(CC) $(CFLAGS) -c -o $@ $-
$(OBJ DIR)/%.s: %.c
    $(echo) + CC $@
    $(rm) -f $@ $(OBJ DIR)
    $(CC) $(CFLAGS) -c -O3 $-
    $(call call_fixdep, $@, $(.S))

```

展开的结果不好阅读

- 使用代码格式化工具

Teaching with slides, codes and demos

English version of learning resources

The 6th "One Student One Chip" Program Home Page

Learning Objectives

Learning Resource

Past Courses Home

Other resources

Record of events

The 6th "One Student One Chip" Program Home Page

Time: Every Saturday 15:00~17:00 China Standard Time

Bilibil Live | recording

Learning Objectives

"One Student One Chip" will develop your general skills. At the end of the course, you will have a better understanding of the following questions:

1. how processors are designed?
2. how programs run on computers?
3. how to optimize the performance of a processor?
4. how to use/design the right tools for efficient debugging?
5. how to write your own test cases for unit testing?
6. how does an RTL design generate a flowable layout?

We will guide you to design a RISC-V pipeline processor. Run an operating system on your processor. Run a real game on the OS. The processor that achieves the target will be connected to the SoC and will be given the opportunity to tapeout.

Learning Resource

Icons can be clicked to jump to the appropriate resource

Complete handouts can be viewed via the navigation bar at the top right of the page

The content of the Stage 5 handout is still available

C = C language (program/simulator/system software) | R = RISC-V instruction set | P = processor design | T = tools

Stage	ID	Task	Handouts	Slides	Recording	C	R	P	T
-------	----	------	----------	--------	-----------	---	---	---	---

The 6th "One Student One Chip" Program Home Page

Prestudy Stage

Prestudy overview

How to ask smart questions

Linux system installation and basic usage

Reviewing the C language

Build a verilator simulation environment

Basic Digital Circuit Lab

Complete PA1

Submission of pre-study defense application

How to ask smart questions

Fill in the general education questionnaire

Before starting the first task of pre learning, please carefully read the content in sections "Signup" and "FAQ" on the official website and fill in "The general education questionnaire of 'One Student One Chip'". Note: The general education questionnaire can be repeated many times, and only those who score 100 can apply for admission defense.

Read "How To Ask Questions The Smart Way" and "Stop Ask Questions The Stupid Ways", and write an essay of your thoughts on them

Your first task in the prestudy is to read the articles "How To Ask Questions The Smart Way" and "Stop Ask Questions The Stupid Ways" and "Don't Ask Like a Retard", and write an 800-word essay about your experience of asking and being asked questions, and what you think about "good questioning" and "Independent problem solving through STFW and RTFM".

This task is not intended to be a waste of time, nor is it intended to prohibit you from asking any questions, but it is intended to show you "what is the right thing to do". When you are willing to work on these "right things" and try to ask questions in a professional way, you have already taken the first step to become a "professional".

STFW, RTFM, RTFSC

Try to find and understand the meanings of the three acronyms in the above article.

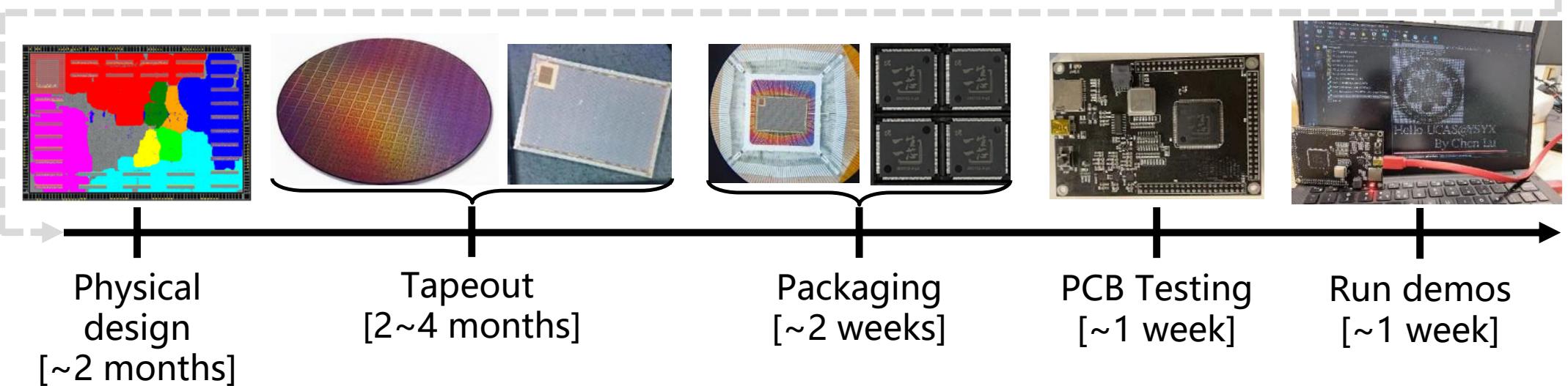
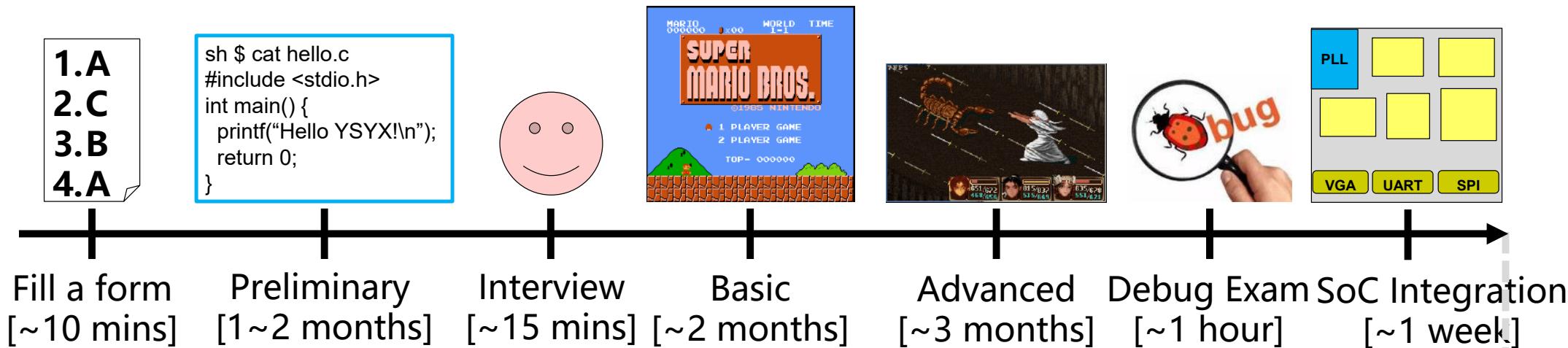
You may feel offended by the F word, but in fact the meaning of the F word is never the point, it just reflects the legend behind the three acronyms and makes them easier to remember. For example, RTFSC originated with the first words of Linus Torvalds, the father of Linux, in a reply to an email dated April 1, 1991, which is still available on the Internet mailing list. Interestingly,

The translation work is still on-going.

Online-debug exam, instead of paper exam

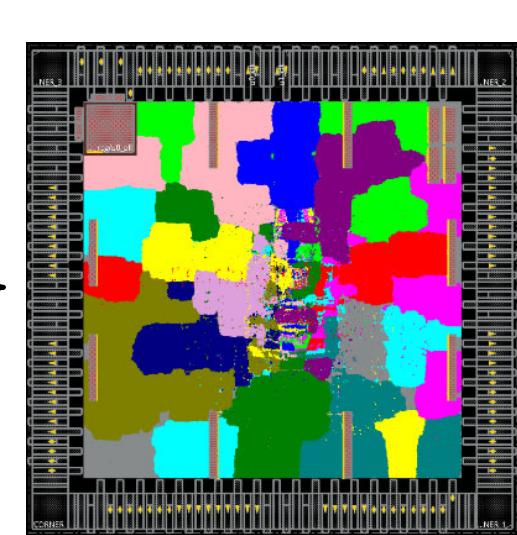
- When students apply for tapeout, TA will inject some bugs randomly in students' project
 - Including hardware, software, building and simulation system
- Students are required to debug within 1 hour in an online meeting
 - Students should share the screen
- According to the debugging process, TA will evaluate whether
 - Student knows enough details in the whole project
 - Student can analyze problem from the respect of sw-hw co-operation
 - Student knows when to use the right tools during debugging
 - Student can solve new problems by himself

Learning Roadmap



Students finish in 4th OSOC

	Student #	University	Major	Grade
1	ysyx_22040053	南京大学	CS	Sophomore
2	ysyx_22040066	南京大学	CS	Sophomore
3	ysyx_22040656	中国计量大学	CS	Sophomore
4	ysyx_22040163	南通大学	CS	Sophomore
5	ysyx_22040091	中国科学院大学	CS	Junior
6	ysyx_22040596	华南理工大学	EE	Senior
7	ysyx_22041812	南方科技大学	EE	Grade 1 master
8	ysyx_22040127	东南大学	AI	Grade 1 master
9	ysyx_22040654	福州大学	EE	Grade 1 master
10	ysyx_22040978	中国科学院大学	EE	Grade 1 master
11	ysyx_22041514	杭州电子科技大学	CS	Grade 1 master
12	ysyx_22040213	中国科学院大学	EE	Grade 2 master
13	ysyx_22040561	北京大学	EE	Grade 2 master
14	ysyx_22041461	四川大学	EE	Sophomore
15	ysyx_22040886	北京理工大学	EE	Junior
16	ysyx_22050228	东北大学	EE	Junior
17	ysyx_22050920	杭州电子科技大学	EE	Senior
18	ysyx_22040501	哈尔滨工业大学	Information and Communication Engineering	Grade 1 master
19	ysyx_22050133	北京大学	Machinery	Grade 2 master



- ysyx_040053
- ysyx_040066
- ysyx_040091
- ysyx_040127
- ysyx_040163
- ysyx_040213
- ysyx_040561
- ysyx_040596
- ysyx_040654
- ysyx_040656
- ysyx_040978
- ysyx_041514
- ysyx_041812

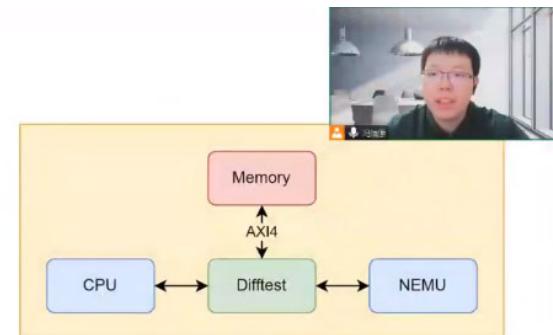
Learning Experiment Shared by Students

- Haoyuan Feng@UCAS, computer science, learning when junior
 - Later became one of the key member of the "XiangShan" team
 - <https://www.bilibili.com/video/BV1C54y1T7hw>



心得分享①：理解系统

- “一生一芯” 仿真系统
 - 逐步建立了对系统的认识，增加调试的信心与效率
 - 仔细阅读源码和手册说明
 - 通过阅读源码，了解 NEMU 和 Spike 的对比机制
 - 接入 ysyxSoC 后，为什么无法从 SRAM 中读出数据
- “香山” 团队中的学习、工作
 - 处理器架构和验证框架复杂很多
 - 主要负责 MMU 模块的开发
 - 涉及到和前端取指、后端访存、内存访问的交互
 - 包含 L1 TLB、L2 TLB、PMP、PMA 等子模块
 - 实践一生一芯的系统观念
 - 从整体入手，关注前端取指和后端访存对 MMU 的查询行为
 - 关注细节，理清 MMU 对于页表查询请求的处理过程
 - 仔细阅读源码，不放过每一处内容，对模块充分掌握



“一生一芯” 仿真系统示意图

端口	说明
output [127:0] Q	读数据
input CLK	时钟
input CEN	使能信号, 低电平有效
input WEN	写使能信号, 低电平有效
input [5:0] A	读写地址
input [127:0] D	写数据

YsyxSoC 中对 SRAM 行为的描述

OSOC Forum@2nd RISC-V China Summit



High school student shares
learning experience

The FIRST high school student who
passes the interview in OSOC!

RVFA certification exam translation

consider some words



Chapter 1: RISC-V Overview

Conventions used in this file:

- Heading 1 is used to mark the beginning of a course page.
- Heading 2 is used for subtitles within a page.
- Bold for references to buttons or menu options, and first sentences in bullet points.
- Bold *Italics*** for introducing new terms.

According to LF Author's Guide:

- Bold Consolas Font (black)** for source code.
- Bold Consolas Font (Dark Blue)** for file names and folder names.
- Bold Consolas Font (Dark Blue)** Text typed at the command line.
- Bold Consolas Font (Green)**: Output.
- Hyperlinks** are left in GoogleDocs' format.

Learning Objectives

- By the end of this chapter, you should be able to:
- Understand what RISC-V is and what it is not.
 - Identify the characteristics of an ISA to decide whether it is a CISC or a RISC type ISA.
 - Describe the history of RISC-V.
 - Identify the most notable differences between RISC-V and the leading commercial ISAs, like x86 and ARM.
 - Understand the structure and operation of RISC-V International.
 - Analyze the documentation of RISC-V specifications.
 - Explore ways to contribute to the RISC-V effort.

Chapter Introduction

In this chapter, you will get acquainted with RISC-V. Please note that you are expected to already be familiar with computer architecture and to have some exposure to some specific ISA. We will provide some refreshers on basic terms, but this is certainly not an introduction to computer architecture.

Among the topics we will cover, you will learn about the history of RISC-V and how it differs from today's dominating ISAs. You will also get the chance to browse through the vast documentation of RISC-V, and you will learn how the documents are organized.

You will be introduced to RISC-V International and the RISC-V ecosystem, and you will learn about quite a few different ways you may contribute to the RISC-V community.

Let us get started!

History of RISC-V: The Free and Open ISA

RISC-V (pronounced "risk-five") is an open standard instruction set architecture principle, enabling a new era of processor innovation through open standard co

RISC stands for Reduced Instruction Set Computer, a computer architecture principle based on simplicity, as opposed to current microprocessors at the time, dubbed Computers, or CISC. The RISC architecture was born in an academic environment for simplicity and efficiency, proposing a series of features that dramatically improved what was motivated by commercial interests at the time. RISC is the opposite of CISC in many ways:

original & translation

RISC-V翻译工作组 (18)
2023-6月7日 1

刘汉章
这种语序的调整是可以的，但是忽略了它原本的一些连词 (thus) 但是更符合中国人的语言好像

刘汉章
因此，一个I

刘汉章
针去正确地

刘汉章
好

我觉得没问题，在不丢失原本语义的情况下，可以按照中文习惯进行适当地修改。

中英文对照表已经整理好了
<https://docs.google.com/spreadsheets/d/1TgFLAT4hb3sC37cvUWBeoSU75iQLP9q73WcaWtdZB/edit#heading=h.32of465ab4d3>，但是由于缺乏具体的语境，有一些单词没办法识别是动词还是名词，所以中文只能作为参考，要具体问题具体分析（翻译有问题可直接在线修改）。

2. 昨天提到的每篇文档中都有的内容或者跟格式相关的标准我统一放到这个文档里面 (<https://docs.google.com/document/d/1TgFLAT4hb3sC37cvUWBeoSU75iQLP9q73WcaWtdZB/edit#heading=h.32of465ab4d3>)，如果大家没有异议的话，可以直接复制到自己的文档中（翻译有问题可直接在线修改）。

3. 像rs1, jar, opcode, qemu这种专有名词我觉得保持原样就好，像ISA, IPC这种由几个单词首字母拼接在一起的专有名词，可以像昨天讨论的那样翻译时保持原样。关于是否在第一次出现的地方添加括号解释，我觉得可以最后等到Review时由一个人统一添加一下会比较好。



缪宇飚



苗金标



段震伟



刘汉章



曹勋



杨海帆



曹世洋



倪仁涛



魏人



陈璐



粟金伦



吴佳宾

Name	University	Grade
缪宇飚	中科院计算所	- (组长)
苗金标	中国科学技术大学	研二
段震伟	中国科学技术大学	研三
刘汉章	太原理工大学	大三
曹勋	中国科学技术大学	研二
杨海帆	浙江工商大学	大四
曹世洋	中国科学技术大学	研二
倪仁涛	东北大学	研一
魏人	兰州大学	大四
吴佳宾	青岛大学	研一
陈璐	中国科学院大学	博一
粟金伦	太原理工大学	大四

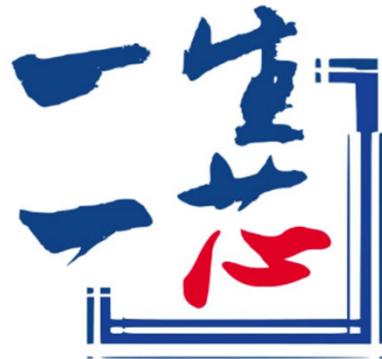
Campus lecture tour

- From March 2023 to now, OSOC has gone deep into the campus and carried out many lecture tours
- In 2024, the journey will continue!

Data	University	Contents
3月19日	北京工业大学	宣讲
4月13日	北京科技大学	宣讲+教学交流
5月17日	东北大学（秦皇岛分校）	宣讲
5月25日	青岛大学	宣讲+教学交流
6月10日	天津工业大学	宣讲
6月14日	天津理工大学	宣讲+教学交流
6月16日	太原理工大学	宣讲



Thank you!



Website
ysyx.org



WeChat Official
Account



北京开源芯片研究院
BEIJING INSTITUTE OF OPEN SOURCE CHIP



中国科学院大学
University of Chinese Academy of Sciences



计算机系统与处理器
芯片课程虚拟教研室



RvA
中国开放指令生态 (RISC-V) 联盟
China RISC-V Alliance

ByteDance
字节跳动

SHANGHAI INNOVATION CENTER FOR COMPUTING TECHNOLOGY

有道
youdao