

# YANG YANG

## PERSONAL INFORMATION

birth	Born in China, Sept. 2001
personal email	jluelioyang2001@gmail.com
official email	yangyang1519@mails.jlu.edu.cn
website	<a href="https://elio-yang.github.io/">https://elio-yang.github.io/</a>
github	<a href="https://github.com/Elio-yang/">https://github.com/Elio-yang/</a>
blog	<a href="https://www.cnblogs.com/oasisyang/">https://www.cnblogs.com/oasisyang/</a>
phone	(+86) 137 8668 9751
address	Jilin University, 2699 Qianjin Street, Changchun, Jilin

## EDUCATION

Undergraduate	<i>Jilin University, Changchun, China</i> GPA: 3.67/4.0 Rank: 10% Major: Computer Science and Technology Interests: Operating System, Computer Architecture and HPC.	Feb. 2019 – Present
---------------	--	---------------------

## AWARDS

Undergraduate	The First Prize Scholarship	Sept. 2020
Academic Year Scholarship	The Second Prize Scholarship	Sept. 2021

## RESEARCH EXPERIENCE

ETECA Lab	<i>Emerging Technology Enabled Computer Architecture, Jilin University</i> Lab Website: <a href="#">here</a> Advisor: Prof. Jingweijia TAN Research on: Computer architecture & High-Performance Computing	Feb. 2022 – Present
-----------	---	---------------------

In short, I am doing research on the **microarchitecture** of General-Purpose Graphics Processing Unit (**GPGPU**). Due to the **FinFET** and state-of-the-art **chiplet** (based on package-level integration), nanometer scale is much more reachable, as a consequence, **process variation** is more complex than before. Therefore I have also been researching on **hardware variability** related to Multi-Chip-Module (**MCM**)-GPUs. Simultaneously, developing a hybrid approach to model and predict the **energy consumption** of the GPGPU under various condition and optimizing it using methods like dynamic voltage/frequency scaling (**DVFS**) is what I am exploring now.

## SKILLS

Languages	C/C++ · Assembly (x86, RISC-V) CUDA · Python Go
Hardware	HDLs: Verilog Modelsim Basic analog circuit design
Software	LINUX/UNIX/Windows GIT L <sup>A</sup> T <sub>E</sub> X · Markdown GNU compiler (gcc, etc.)

# PROJECTS

---

EOS	<p>EOS is a 32bit *nix operating system developed in C language. <span style="float: right;">Sept. 2021</span></p> <p>Till now EOS contains a basic <b>bootloader</b>, 2-level <b>paging</b>, 4GB <b>memory management</b> and <b>kernel multithreads</b>. For user environment, it provide a set of traditional shell programs and <b>multi-process</b> mechaism. It follows the x86 ABI, so it's easy to port thoses x86 applications. This project is still <i>active</i> and it will provide a <i>GNU C Project</i> like library and compiler support in the future. You can find the codes <a href="#">here</a>.</p>
MapReduce Engine	<p><i>MapReduce Engine</i> is a Go language implementation of the paper.<sup>1</sup> <span style="float: right;">Apr. 2022</span></p> <p>This engine consists of a <b>fault tolerance</b> (failures like crash and communication-lose of workers) master and a worker cluster. Users can specify their cluster size and working functions (mapf &amp; reducef). With a simulated distributed file system, the workers can communicate with the master through <b>Remote Procedure Call</b>. This MapReduce Engine is a basic component for building a distributed system used for operations over large-scale datasets. You can find the codes <a href="#">here</a>.</p>
WYZ-BAR	<p><i>WYZ-BAR</i> is a bar management system developed in C language. <span style="float: right;">Mar. 2020</span></p> <p>WYZ-BAR is a <i>collaborative project</i> (WYZ stands for 3 members and Y is for me) and I am the leader. With the <b>multi-process</b> organization and a simple builtin <b>sqlite style database</b>, WYZ-BAR is my <i>first</i> course project in the university and it made me a minor celebrity. The development flow follows the modern <b>open source</b> software's way. A lot of <b>parsing</b> techniques are used to deal with all kinds of data input, this system is purposely optimized for unqualified input like the real world. You can find the codes <a href="#">here</a>.</p>
CUDA-FFT	<p><i>CUDA-FFT</i> is a CUDA implementation of the <b>Fast Fourier Transform</b> algorithm. <span style="float: right;">Dec. 2021</span></p> <p>This project implemented 3 algorithms to do the <i>polynomials multiplication</i>, including ordinary multiplication, <b>recursive-FFT</b> and <b>gpu-FFT</b>. The performance was well tested and the contrast was shown in the report. This is my first time doing heterogeneous computing and this project leads me to the research of <b>HPC &amp; GPGPU</b>. You can find the codes, slide, and report <a href="#">here</a>.</p>
Others	<p>You can find more projects including course labs (like MIT 6.828), Android application (SmogDetector), <i>etc.</i> in <a href="#">GitHub</a></p>

# OTHER INFORMATION

---

Languages	CHINESE · Mother tongue
	ENGLISH · Intermediate (conversationally fluent)
Interests	Literature (Latin-American, magic realism) · Physics · NBA (Golden State Warriors) · Classic (Chopin)
Characteristic	Strong patience · Highly self-motivated · Creative · Communication and collaboration skilled.

---

<sup>1</sup> J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008, doi:[10.1145/1327452.1327492](#).