VLSI Technology page:

VANGUARD (VG) – The First Open Source RISC-V SoC Project in Vietnam

QUYNH DO-NGOC (QUYNHDO.ICD@GMAIL.COM)

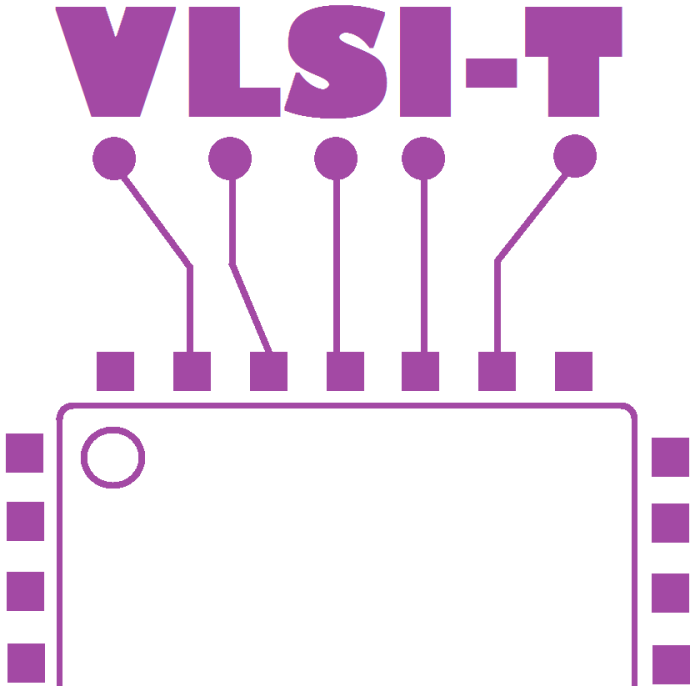
QUAN NGUYEN-HUNG (NGUYENQUAN.ICD@GMAIL.COM)

& VLSI TECHNOLOGY PAGE'S COWORKERS

Agenda

- Introduction to VLSI Technology page
- Vanguard (VG) open source project
  - VG structure
  - Roadmap and Status
  - Purpose
  - Resource

Introduction to VLSI Technology page



Introduction to VLSI Technology page

Engineers

Technical Students

More technical students

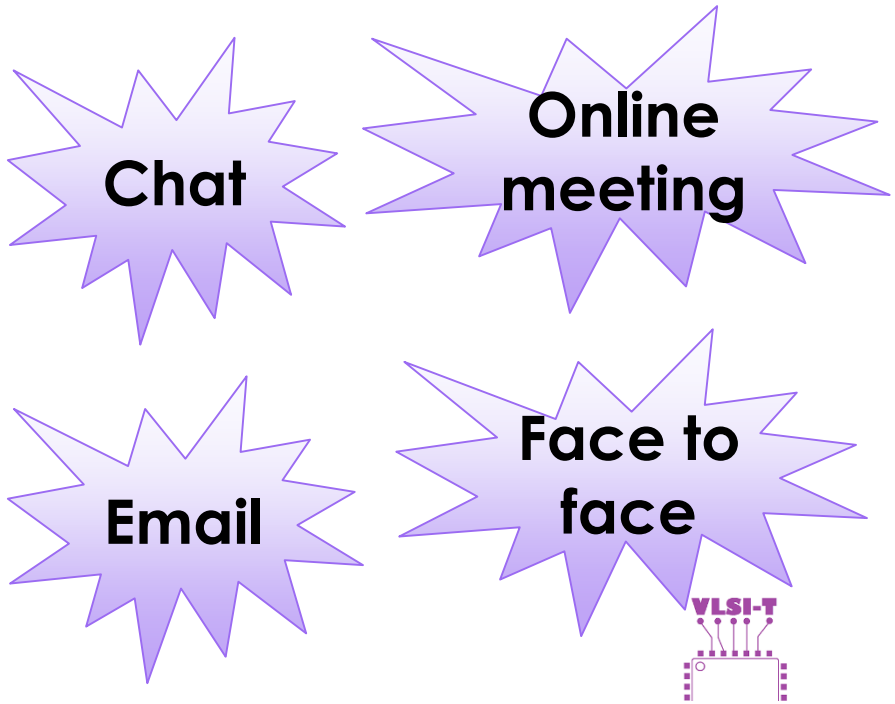
More sharing

More the public discussion

Activities

- Discuss and select topics
- Establish a team for each topic
- Make a plan and investigate the selected topics
- Create examples and tutorials
- Publish results as documents or videos

Flexible communication



The current content is focused on the design and verification but we are investigating the back-end phase.

First Open Sources

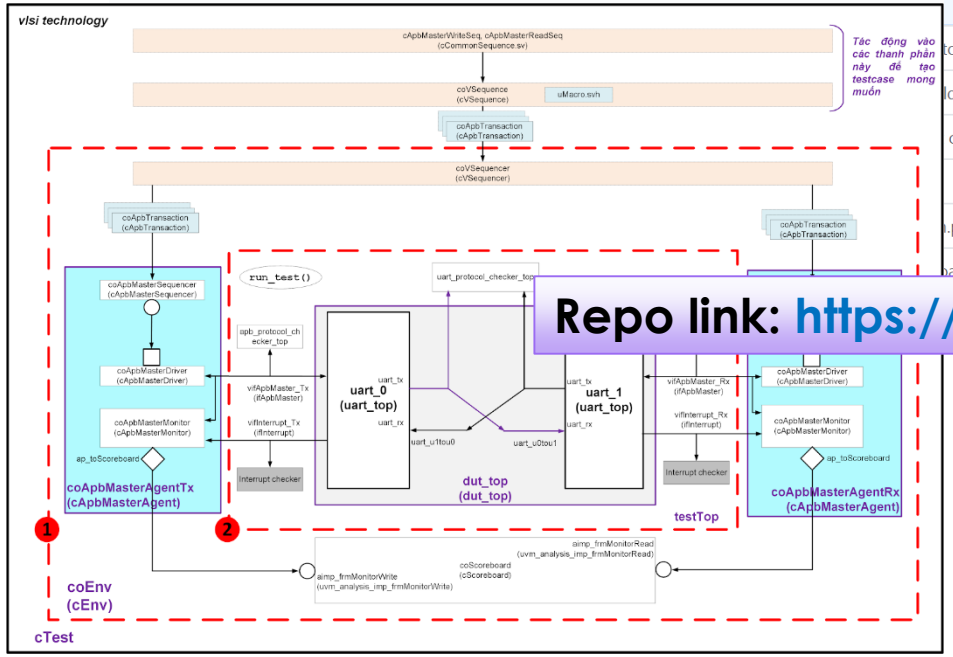
SystemVerilog UVM environment

nguyenquanicd / UvmEnvUartApb Template

SystemC UVM environment

nguyenquanicd / apbUartUvmSystemC

Code Issues Pull requests Actions Projects Wiki



RTL code generation tool

nguyenqu

Code

IP cores

nguyenquanicd / Aes128_ECB_CBC_CFB_OFB_CTR

Code Issues Pull requests Actions Projects

master

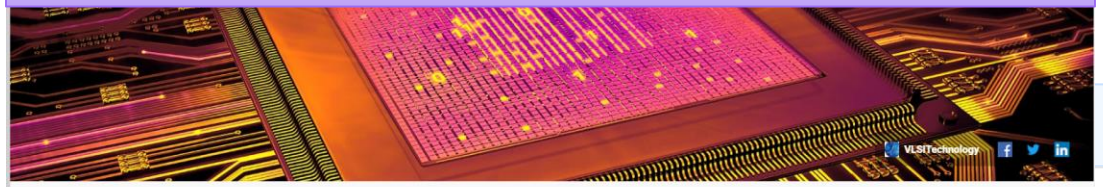
nguyenquanicd / FirstX2P

generated from nguyenquanicd/UvmEnvUartApb

Code Issues Pull requests Actions Projects

master

Online courses



CUSTOMIZE CHANNEL YOUTUBE STUDIO

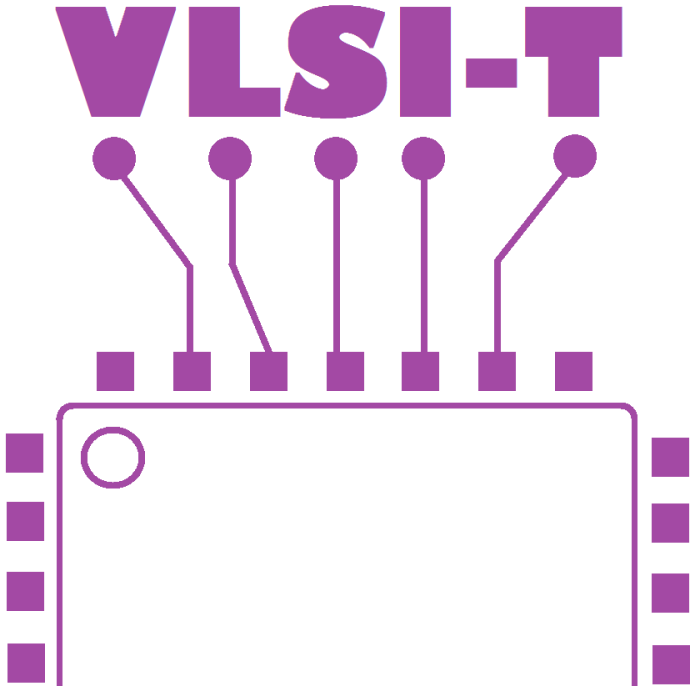
HOME VIDEOS PLAYLISTS CHANNELS DISCUSSION ABOUT

Uploads PLAY ALL SORT BY

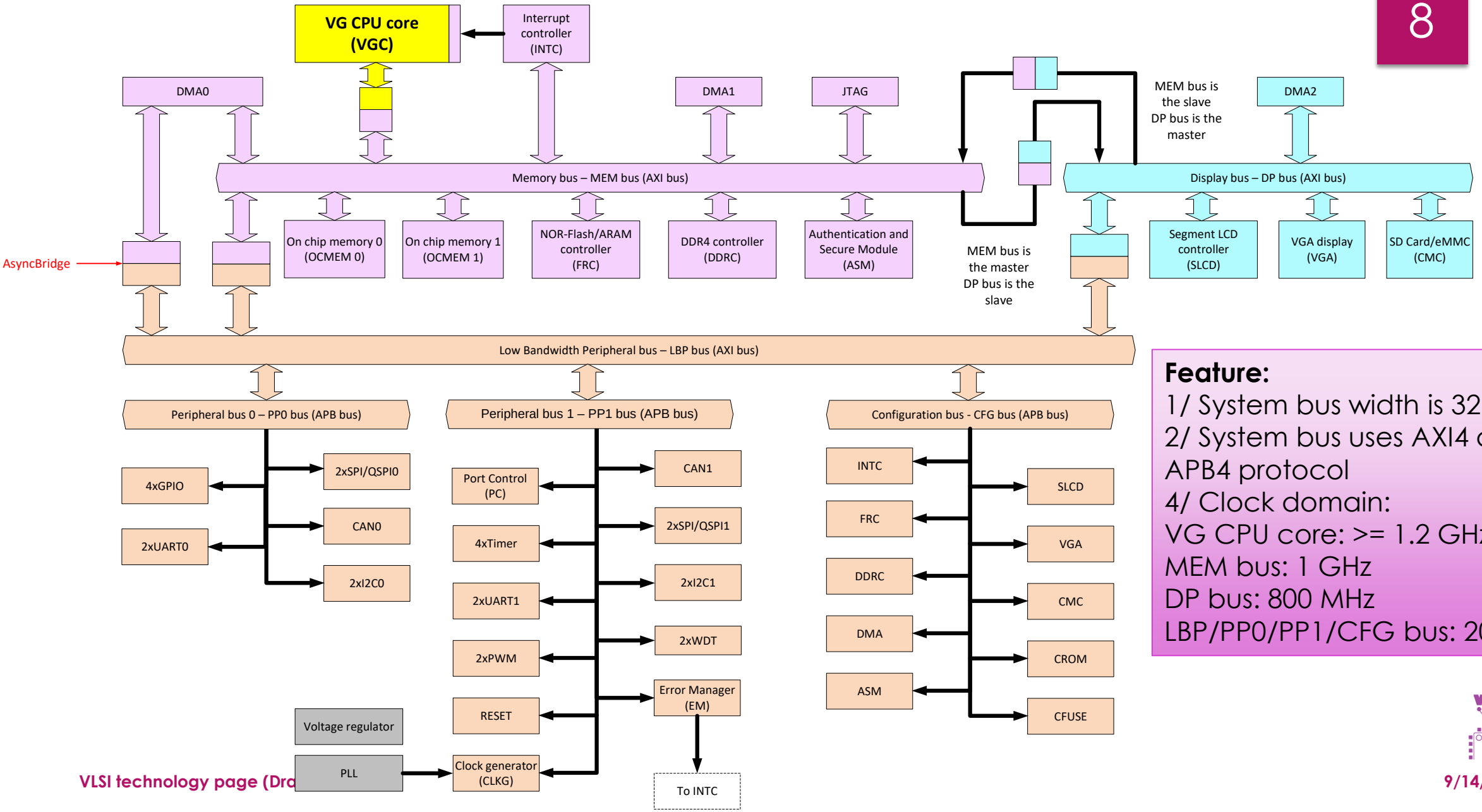
#VLSITechnology [VLSIE002] Bài 3 - Nguyên tắc chèn và... | #VLSITechnology [VLSIE002] Bài 2 - Biểu diễn giá trị logic... | #VLSITechnology - Hướng dẫn biên dịch và mô phỏng... | #VLSITechnology [VLSIE002] Bài 1 - Giới thiệu về... | #VLSITechnology [VLSIE002] Giới thiệu khóa...

Repo link: https://github.com/nguyenquanicd/VG_RTL

Vanguard (VG) Open Source Project



VG Diagram



Feature:

1/ System bus width is 32 bits

2/ System bus uses AXI4 and APB4 protocol

4/ Clock domain:

VG CPU core: >= 1.2 GHz

MEM bus: 1 GHz

DP bus: 800 MHz

LBP/PP0/PP1/CFG bus: 200 MHz

Detail Function Block

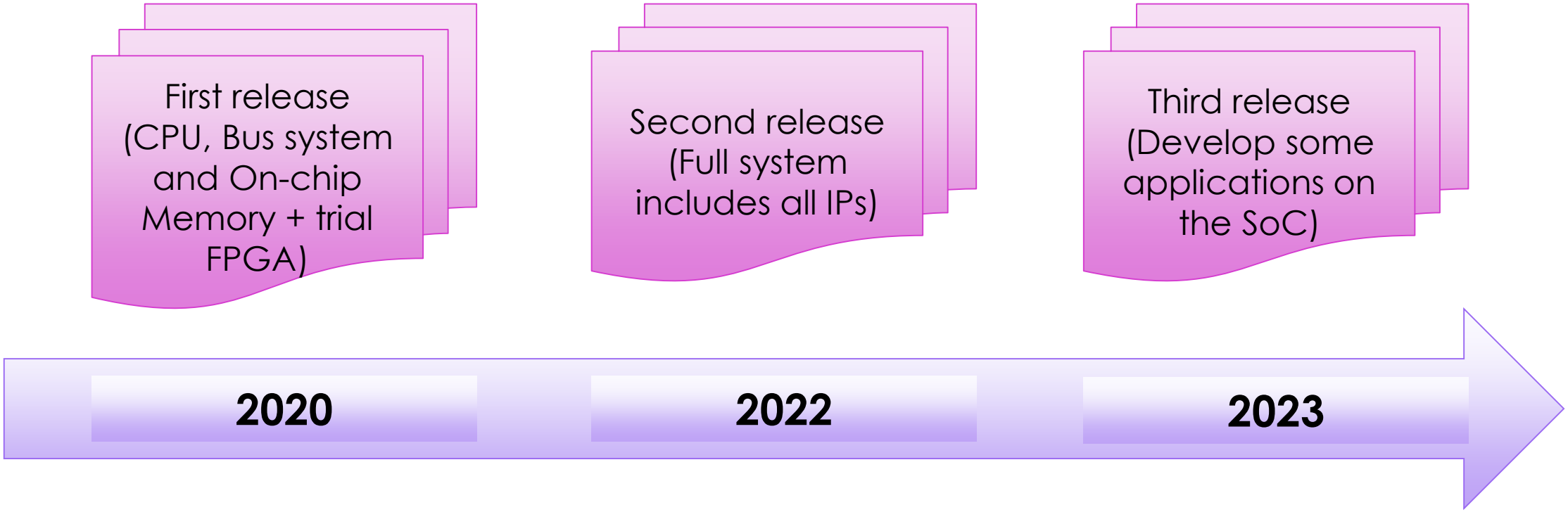
No.	Sub-System	Description
1	VGC	RISC-V CPU core executes all programs
2	Memory (MEM) bus	Interrupt controller (INTC), JTAG interface (JTAG), Watchdog timer (WDT), Direct Memory Access (DMA), On-chip memory (OCMEM), External Asynchronous NOR-Flash/RAM controller (FRC), DDR4 SDRAM controller (DDRC), Authentication and Security Management (ASM)
3	Display (DP) bus	Direct Memory Access between two sub-system (DMA), Segment Liquid Crystal Displays controller (SLCD), Video Graphics Array controller (VGA), SD Card/eMMC memory controller (CMC)
4	Low Bandwidth Peripheral (LPB) Bus	Timer controller (Timer), Universal Asynchronous Receiver/Transmitter (UART), Pulse-width modulation (PWM), General purpose input/output (GPIO), Reset controller (RESET), Clock controller and generator (CLKG), Serial Peripheral Interface (SPI/QSPI), Controller Area Network (CAN), Inter-Integrated Circuit (I2C), Error management (EM)
5	Configuration (CFG) bus	Read/Write to all configuration registers of peripherals in AXI bus

Why we choose RISC-V for SoC

- First thing: RISC-V is Royalty Free Industry standard Instruction Set Architecture (ISA)
  - It is easy for everyone to access and use ISA in their open projects
  - The open source community will grow strongly like the ISA becomes the more and more popular. In the hardware world, RISC-V role will be same as Linux OS in the software world.
- RISC-V is not only built on FPGA but also proven by the silicon.
  - Some chips are commercial such as products from SiFive*1, Syntacore*2, etc.
- According to SiFive website, the performance of RISC-V chips are comparable to ARM chips. Concretely, SiFive U8-Series RISC-V chip designs should be competitive with ARM Cortex-A72.
- This ISA supports both 32-bit instructions and 16-bit compressed instructions: they can be used in many chips from a small embedded to a high performance processor.

*1: https://www.sifive.com
*2: https://syntacore.com

Roadmap


First release (CPU, Bus system and On-chip Memory + trial FPGA)
Second release (Full system includes all IPs)
Third release (Develop some applications on the SoC)
2020
2022
2023

RISC-V Core Development Plan

- First and second version: RISC-V CPU is open source core from Roa logic*1 with following modifications:
  - Bus interface unit has been changed from AMBA AHB interface to AMBA AXI interface to improve the system performance
  - Besides, the unit test environment of CPU is also updated. Currently, the modification has been completed and verified with these sets of instructions: rv32ui and rv32si from riscv-tests*2
- Third version: RISC-V core is improved
  - Architecture: upgrade this core to a superscalar processor

*1: https://roalogic.com
*2: https://github.com/riscv/riscv-tests

Current Status: Focus on the phase 1

- RISC-V CPU Core modification and unit test are completed
- Bus systems
  - The design and the basic verification of APB bus is completed. The full simulation environment is being built
  - AXI bus is on-going the design phase
- On-chip Memory is completed
- Others: SPI, UART and CAN are completed

Purpose

- Supplying an open source SoC to study, research, and use in Vietnam
- Developing a RISC-V SoC on FPGA
- Building a RISC-V community in Vietnam

Resource

- Core team: 10 engineers and 2 students
- Short-term supporters (3 months): 5 to 10 members
- Advantage:
  - Many students and engineers have a need for academic research
  - Knowledge of RISC-V started to be popular in Vietnam beginning from the universities and institutes
  - Participants will have access right to project knowledge and data soon
- Disadvantage:
  - The activities of group are not regular
  - The working time cannot be strictly because membership is voluntary and cannot join full-time
  - No member has a lot of experience in back-end phase

THANK FOR YOUR ATTENTION!

VLSI TECHNOLOGY PAGE – VIETNAM