

# All Open Source Toolchain for ZYNQ 7000 SoCs

Yimin Gu  
Embedded, Mobile and Automotive devroom,  
FOSDEM 2025

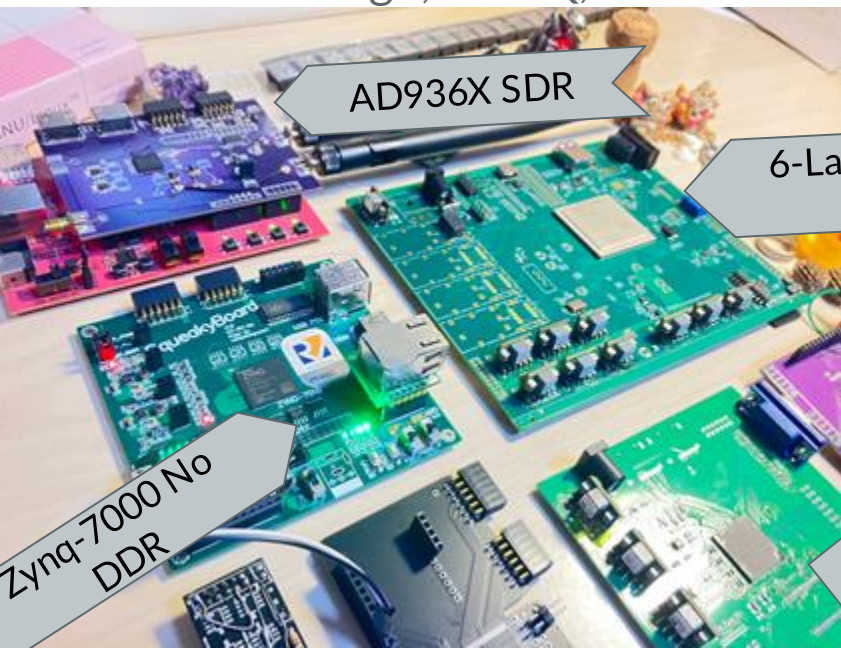
# About Me

- PhD candidate
- Amateur FPGA programming, RISC-V, PCB design, ZYNQ, Maker Faire, etc

Hackers & Painters



I'm neither



AD936X SDR

6-Layer ZU15EG  
DDR4

Zynq-7000 No  
DDR

2-Layer Zynq-  
7000

MMU/No-MMU 32-bit RISC-V SoC

```
T1) devtmpfs: initialized
T1) clocksource: jiffies: mask: 0xffffffff max_cycles: 0xffffffff,
T1) futex hash table entries: 256 (order: 0, 7168 bytes, linear)
C0) hrtimer: interrupt took 10999999 ns
T1) workingset: timestamp_bits=30 max_order=14 bucket_order=0
T1) io scheduler mq-deadline registered
T1) io scheduler kyber registered
T1) 930000000-quasi6c: quasi6c registered
T1) printk: console [ttyL0] enabled
T1) printk: bootconsole [sb10] enabled
T1) printk: bootconsole [sb10] disabled
T1) start plist test
40.972442
43.275265
T1) debug_vm_pgtable: [debug_vm_pgtable
T1) clk: disabling unused clocks
T1) Validating architecture
209.319761
209.346633
T1) TERM=linux
```



# Table of Contents

- Open-source FPGA toolchains
- Now Zynq (demo)
- What does this mean
- Future

# Open-source FPGA Toolchains

# From iCE40 to Kintex 7

- Icestorm for **Lattice iCE40**, in 2015
- PrjTrellis for **Lattice ECP5**
- Project Apicula 🐝 for **Gowin**
- F4PGA for **Xilinx 7 Series**
- **OpenXC7** supports **Kintex 7**
- ice40hx8k, 8K LE
- lfe5u-85f, 85K LE
- gw2ar, 20K LE
- xc7a200t, 200K LE
- xc7k480t, 480K LE



# Possibilities

- PicoRV32, KianRiscV, Quasi SoC (No-MMU), UberDDR3, SD Card, HDMI, MMCM, IOSERDES
- Spartan 7/Artix 7/Kintex 7, **Zynq 7000 PL (Artix and Kintex fabric)**
- **SATA controller, 10G ETH, GTP, GTX, PCIe**

Runs on ARM Machines (Apple M1 included)

Runs in Docker

`docker pull regymm/openxc7 (openxc7-arm)`

Runs for online services

`docker pull regymm/gowin`

`docker pull regymm/oss-cad-suite`

Now Zynq!

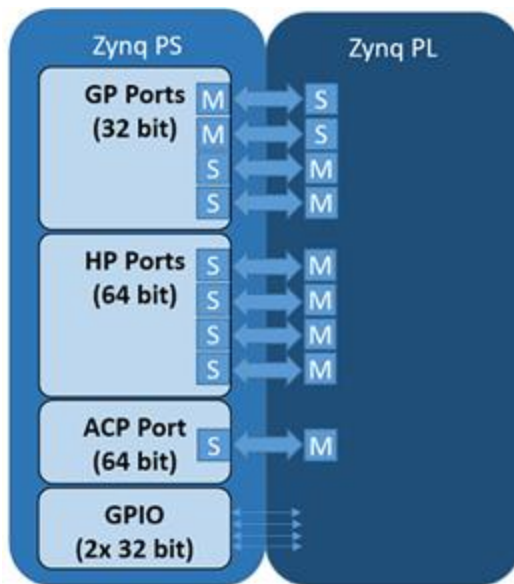
# Double trouble?

- “Unique” PS + PL structure

2x ARM A9, DDR3  
ETH/USB/...  
Flexible pins

Linux / RTOS

Boot



FPGA fabric,  
HDMI/ETH/LED/...

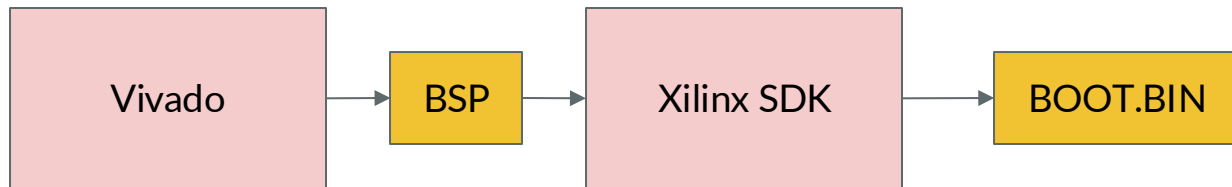
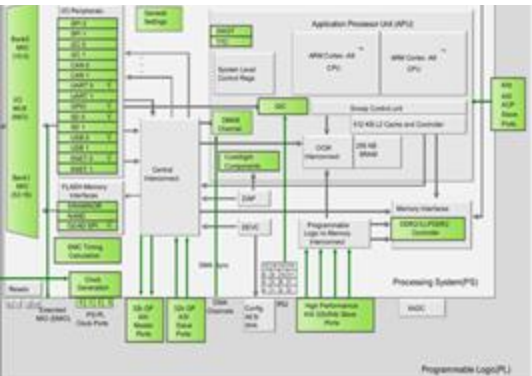
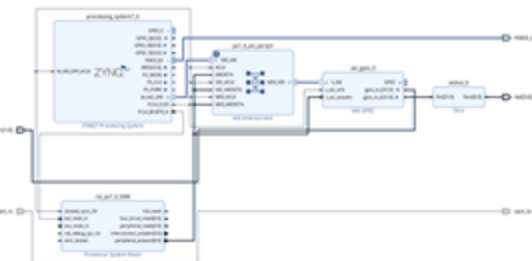
Custom Logic  
bitstream

[https://pynq.readthedocs.io/en/latest/overlay\\_design\\_methodology/pspl\\_interface.html](https://pynq.readthedocs.io/en/latest/overlay_design_methodology/pspl_interface.html)



# Are you confident...

- Building a Zynq BOOT.BIN firmware by yourself?
- How long does it take?



PL Bitstream  
PS Configuration

.bit

ps7\_init.c  
xparameters.h

FSBL  
User App

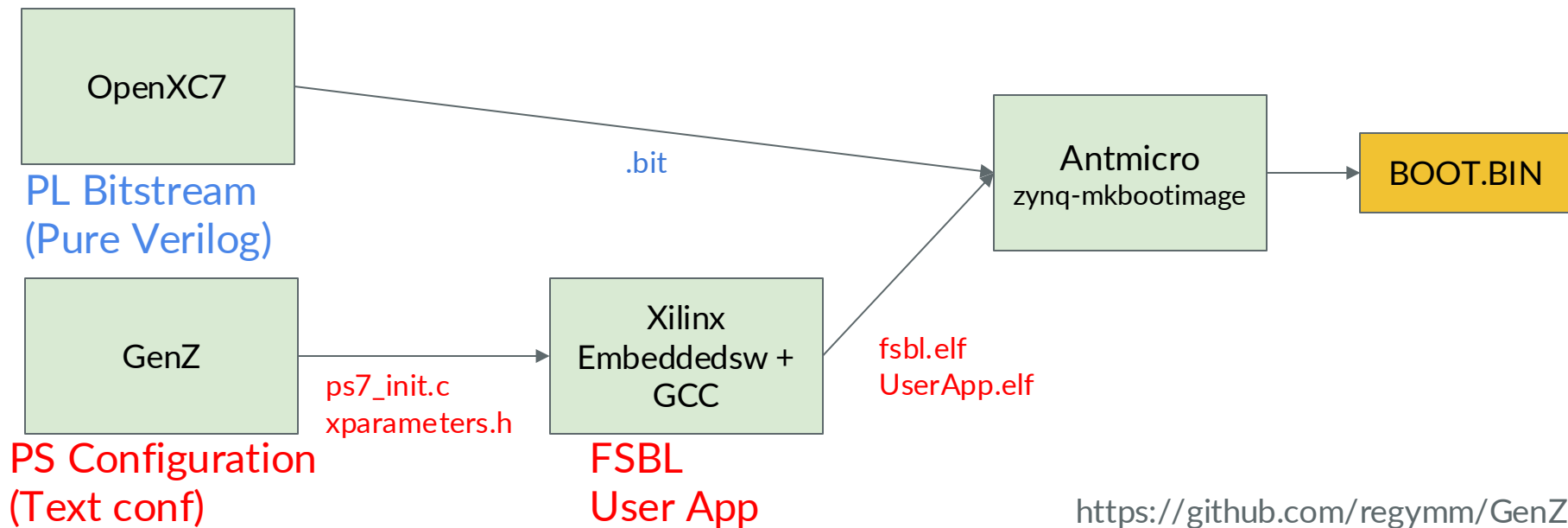
.bit

fsbl.elf  
UserApp.elf

Hundreds of mouse clicks!

# OpenXC7 + GenZ

- BOOT.BIN in 5 minutes
- Awesome through 2035





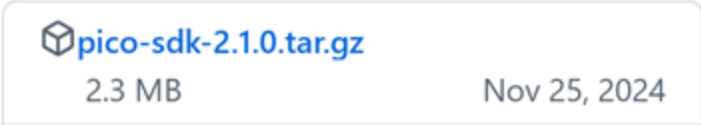
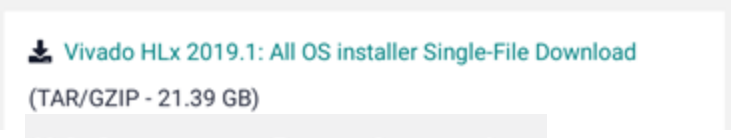
# Decoding the ARM registers

- 100+ reg writes
- Zynq 7000 SoC Technical Reference Manual (UG585)
  - PLLs (APU, DDR, IO)
  - Clock Freqs (APU, DDR, every peripheral)
  - DDR (TODO)
  - MIO (MUX selection)
  - PERIPH (every peripheral's register space)
- Debug (???)



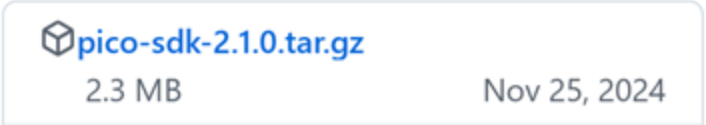
```
// START: top
// .. START: SLCR SETTINGS
// .. UNLOCK_KEY = 0XF00
// .. ==> 0XF0000008[15:0] = 0x0000DF00
// .. ==> MASK : 0x0000FFFF VAL : 0x0000DF00
// ..
EMIT_WRITE(0XF0000088, 0x00000F00),
// .. FINISH: SLCR SETTINGS
// .. START: PLL SLCR REGISTERS
// .. .. START: ARM PLL INIT
// .. .. PLL_RES = 0x2
// .. .. ==> 0XF000110[7:4] = 0x00000020
// .. .. ==> MASK : 0x00000F00 VAL : 0x00000020
// .. .. PLL_CP = 0x2
// .. .. ==> 0XF000110[11:8] = 0x00000020
// .. .. ==> MASK : 0x00000F00 VAL : 0x00000200
// .. .. LOCK_CNT = 0xf0
// .. .. ==> 0XF000110[21:12] = 0x000000FA
// .. .. ==> MASK : 0x003FF000 VAL : 0x0000FA000
// .. ..
EMIT_MASKWRITE(0XF000110, 0x003FFF00, 0x000FA220),
// .. .. START: UPDATE FB_DIV
// .. .. PLL_FDIV = 0x28
// .. .. ==> 0XF000100[18:12] = 0x00000028
// .. .. ==> MASK : 0x0007F000 VAL : 0x00028000
// .. ..
EMIT_MASKWRITE(0XF000100, 0x0007F000, 0x00028000),
// .. .. FINISH: UPDATE FB_DIV
// .. .. START: BY PASS PLL
// .. .. PLL_BYPASS_FORCE = 1
// .. .. ==> 0XF000100[4:4] = 0x00000010
// .. .. ==> MASK : 0x00000010 VAL : 0x00000010
// .. ..
EMIT_MASKWRITE(0XF000100, 0x00000010, 0x00000010),
// .. .. FINISH: BY PASS PLL
// .. .. START: ASSERT RESET
// .. .. PLL_RESET = 1
// .. .. ==> 0XF000100[0:0] = 0x00000010
// .. .. ==> MASK : 0x00000010 VAL : 0x00000010
```

What does this mean?

# Enjoy RP2040? How about Zynq!

	<b>RP2040</b>	<b>Zynq 7010</b>
“PS”	2x Cortex M0+ @ 133 MHz 264 kB SRAM UART, SPI, etc USB 1.1 	2x Cortex A9 @ 667 MHz 256 kB OCM + DDR UART, SPI, SDCARD, etc USB 2.0, 1Gbps ETH 
“PL”	8x Programmable I/Os Modules	28K+ LE
I/O	30 GPIOs	80+ HR I/Os
Price	<€5	€10 - €200
Toolchain	<b>Cross platform, FOSS</b>  2.3 MB Nov 25, 2024	<b>x86 Win/Linux only, Proprietary</b>  (TAR/GZIP - 21.39 GB) <b>U.S. Government Export Approval</b>

# Enjoy RP2040? How about Zynq!

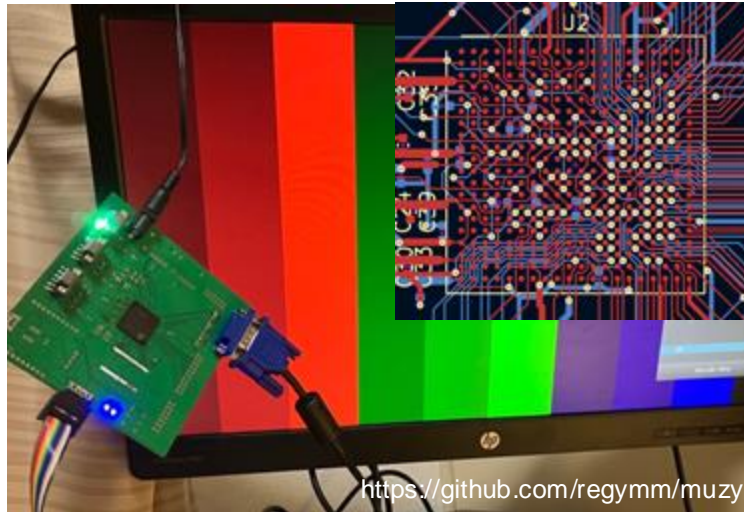
	<b>RP2040</b>	<b>Zynq 7010</b>
“PS”	2x Cortex M0+ @ 133 MHz 264 kB SRAM UART, SPI, etc USB 1.1 	2x Cortex A9 @ 667 MHz 256 kB OCM + DDR UART, SPI, SDCARD, etc USB 2.0, 1Gbps ETH 
“PL”	8x Programmable I/Os Modules	28K+ LE
I/O	30 GPIOs	80+ HR I/Os
Price	<€5	€10, €60 - €200
Toolchain	<b>Cross platform, FOSS</b>  2.3 MB Nov 25, 2024	<b>Cross platform, FOSS</b> <b>docker pull regymm/openxc7</b> <b>git clone <a href="https://.../GenZ">https://.../GenZ</a></b>

# Zynq-as-MCU

- Boards are available
- Upcycled boards set price at €10



EBAZ4205, ~€10



2-Layer  
640x480 VGA  
115 MHz SDRAM

Muzy-2, BoM ~€10



shop.trenz-electronic.de

Photo Shows  
Similar Product

ZynqBerryZero, ~€100  
Trenz electronic



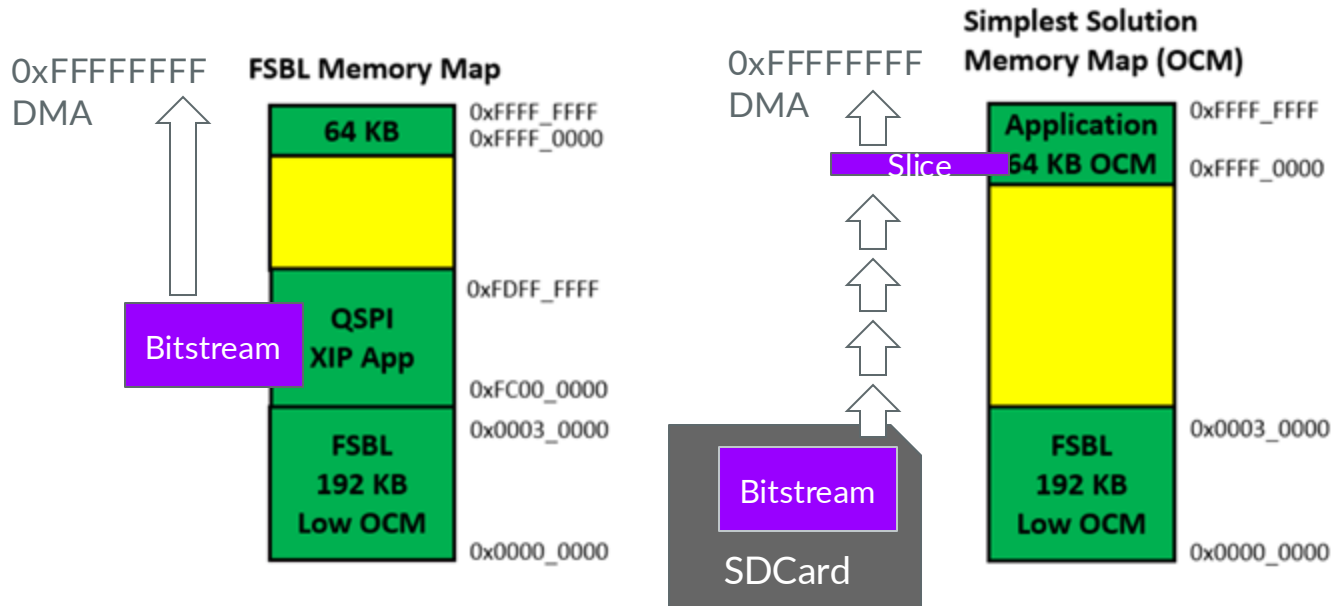
shop.trenz-electronic.de

Photo Shows  
Similar Product

DIPFORTy1, ~€60  
Trenz electronic

# Possibilities unlocked

- #1 No DDR SD Boot
  - Not supported officially, *was* not convenient





# Possibilities unlocked

- #2 Overclock 🔥

- [hz12opensource/libresdr](#)

- GenZ

```
+cpu_mult_template='EMIT_MASKWRITE\(\0XF8000100, 0x0007F000U ,0x000XX000U\),'  
+cpu_mult_find='\s' "${cpu_mult_template/XX/(..)}"  
+ddr_mult_template='EMIT_MASKWRITE\(\0XF8000104, 0x0007F000U ,0x000XX000U\),'  
+ddr_mult_find='\s' "${ddr_mult_template/XX/(..)}"
```

```
z7000_ps_param_900 = {  
    'freq' : { 'crystal' : 33.333333333,  
              'apu'      : 900.0  
            }  
}
```

# Flexibility meets more flexibility

- #3 PL bitstream built on PS itself
- It's an ARM anyways
- OpenXC7 runs on 1 GB mem



The terminal window displays the following output:

```
Info: SA placement time 1.98s
Info: Max frequency for clock 'FCLK_CLK_buffered[0]': 145.50 MHz (PASS at 12.00 MHz)
Info: Slack histogram:
Info: legend: * represents 1 endpoint(s)
Info:          + represents [1,1] endpoint(s)
Info: [ 76460, 76788) |****
Info: [ 76788, 77116) |*
Info: [ 77116, 77444) |**
Info: [ 77444, 77772) |**
Info: [ 77772, 78100) |**
Info: [ 78100, 78428) |**
Info: [ 78428, 78756) |**
Info: [ 78756, 79084) |*
Info: [ 79084, 79412) |***
Info: [ 79412, 79740) |**
Info: [ 79740, 80068) |**
Info: [ 80068, 80396) |**
Info: [ 80396, 80724) |**
Info: [ 80724, 81052) |**
Info: [ 81052, 81380) |***
Info: [ 81380, 81708) |**
Info: [ 81708, 82036) |**
Info: [ 82036, 82364) |*
Info: [ 82364, 82692) |**
Info: [ 82692, 83020) |**
Info: Checksum: 0x05ea868f
Info: Running post-placement legalisation...
Info: Tying unused PS7 inputs to constants...
Info: Routing global clocks...
Info:   routing clock 'FCLK_CLK_buffered[0]'
Info: Running router2...
Info: Setting up routing resources...
```

The photograph shows a PYNQ-Z1 board with a cat sticker. The board is populated with various components, including a Zynq-7010 SoC, DDR3 memory, and a USB-to-UART bridge. The board is connected to a monitor displaying a Linux desktop environment.

The terminal window also shows a task list and a resource usage table:

```
Tasks: 31, 7 thr; 2 running
Load average: 1.40 0.44 0.1
Mem[|||||421K/497M]
Uptime: 00:20:51
Swp[ ]
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPUS	MEM%
2137	xilinx	20	0	1119M	439M	464	R	10.3	88.6
2120	xilinx	20	0	5832	1040	784	R	0.3	0.2
1706		20	0	25952	1148	1000	S	0.2	0.2
2106	xilinx	20	0	10140	92		S	0.1	0.0
814		19	-1	40492	168	4	S	0.0	0.0
1		20	0	27544	40		S	0.0	0.0
1341		20	0	13584	12		S	0.0	0.0
1445		20	0	17088	0		S	0.0	0.0

The terminal window also shows a file manager window with the following contents:

```
F1: /tmp F2: /setup F3: /src F4: /lib F5: /tree F6: /src F7: /src
```

# Larger devices

- Zynq 7030, 7035, 7045, 7100 now supported by OpenXC7
- Would have transceivers and PCIe blocks, DDR3 on PL, ...



AX7Z100B, [alinx.com](http://alinx.com)

Future

# Freedom is the future

- PetaLinux?
- SDRs and more



 OpenWiFi



# Acknowledgement



OCP-TAP



LUG@USTC



VLAB@USTC

# Thank you!

- [github.com/regymm](https://github.com/regymm)
- [ustcymgu @ gmail.com](mailto:ustcymgu@gmail.com)







# Quasi SoC IRL

(Two cores, one RV32IMA\_Zicsr main CPU, one RV32I VT100)



# RV32 No-MMU

