FORCE-RISCV Deployment on XuanTie CPU Verification Project

Binguang.Zhao/Ethan

Force-riscv Acting Maintainer

CONTENTS

1 What’ s force-riscv?

2 XuanTie Deploy story

3 Force-riscv stage2 plan

What’ s force-riscv?

Dynamic Instruction Sequencer

Spike/Simulator Integrated

Server class cpu top verification

RISCV RV64/32 I,M,A,V…

Python Frontend/C++ backend
Python fine grained templates

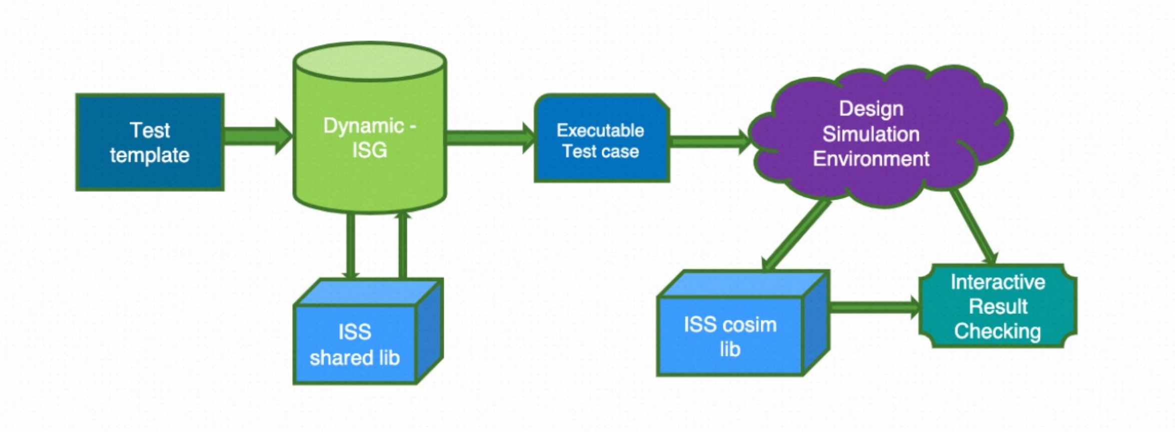
Dynamic Virtual Memory
MP scenario

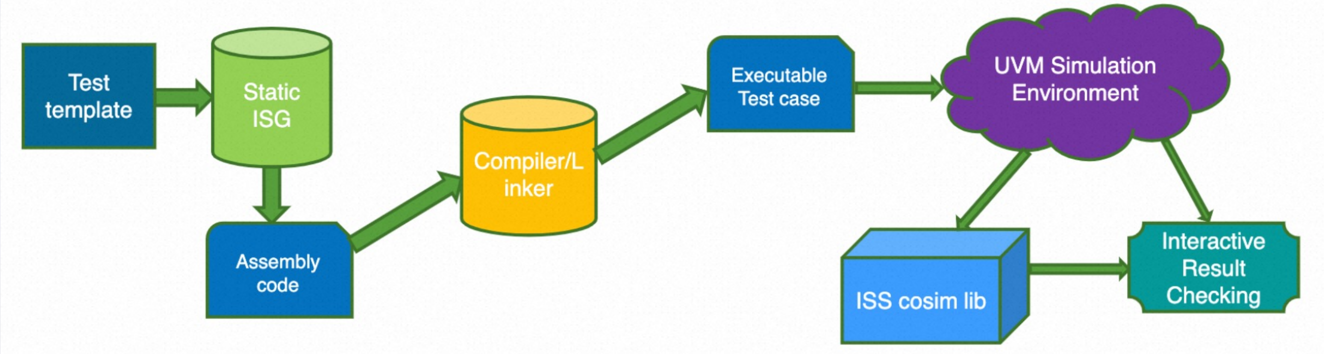
- 2020.6 Open source by FUTUREWEI. RV64G,F,D,C
- 2022.2 v0.9/v1.0 released with RV32, paging fault, memory trait, vector 0.9 …
- 2023.2 XuanTie starts stage 2 development: handcar/ spike upgrade, vector 1.0, version control, mp …
- 2023.6 cmake build system ready ……

What’ s force-riscv?

Dynamic ISG:

- Output is ELF file can be directly loaded by env
- Instruction record feedback for fine-grained control
- Good coverage possible
- Hard to implement





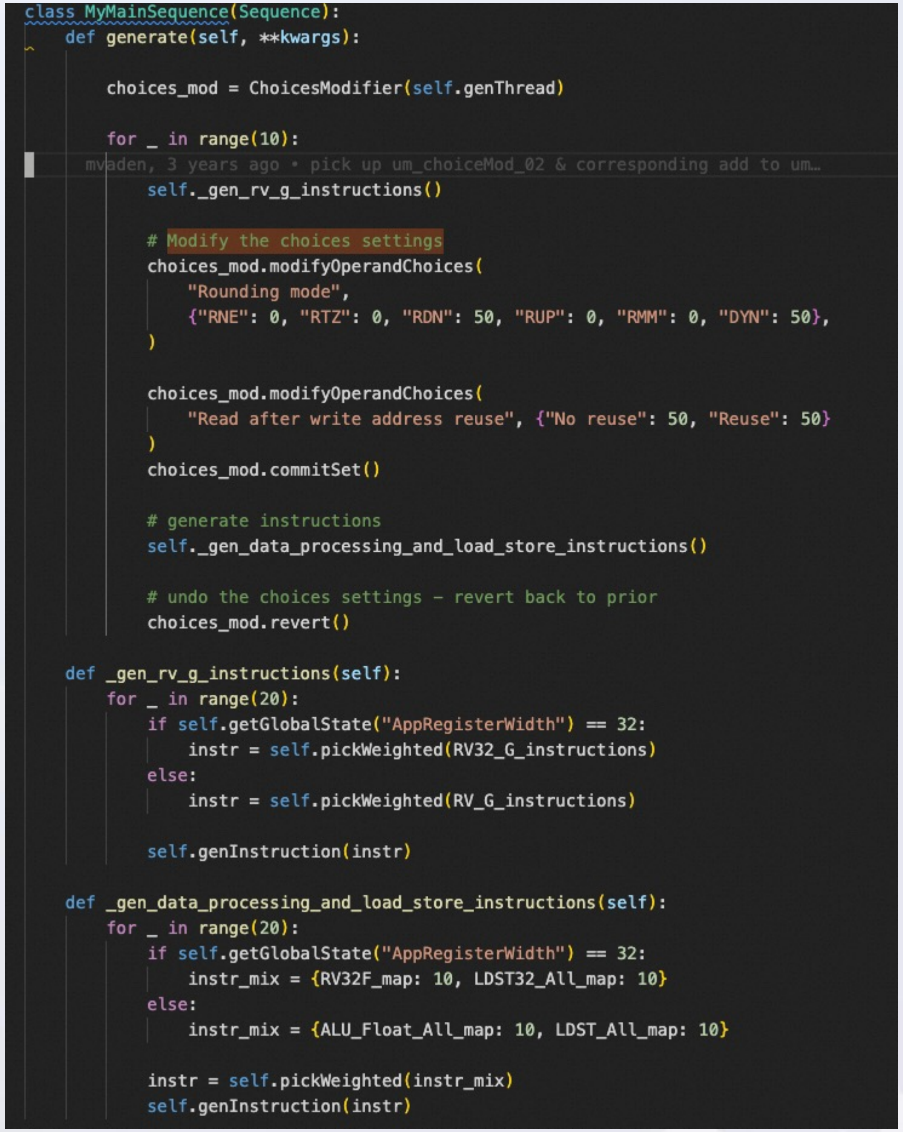
Static ISG:

- Output is .s/.S file. Need toolchain
- Coarser granule controls
- bad coverage
- Easy to implement

Provided by Futurewei: https://github.com/openhwgroup/programs/blob/master/TWG/MeetingPresentations/2020-09-07%20FORCE-RISCV%20ISG%20-%20status.pdf

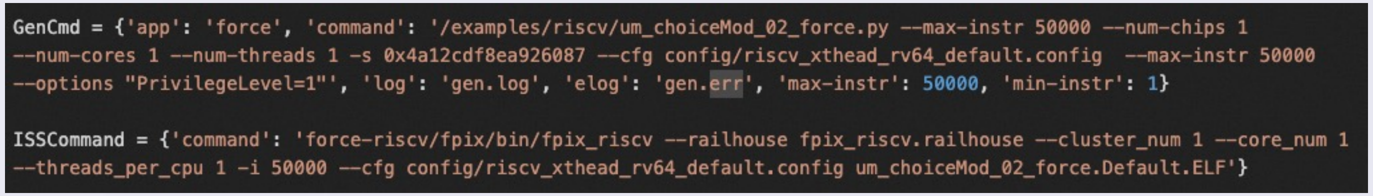
What’ s force-riscv?

Template:



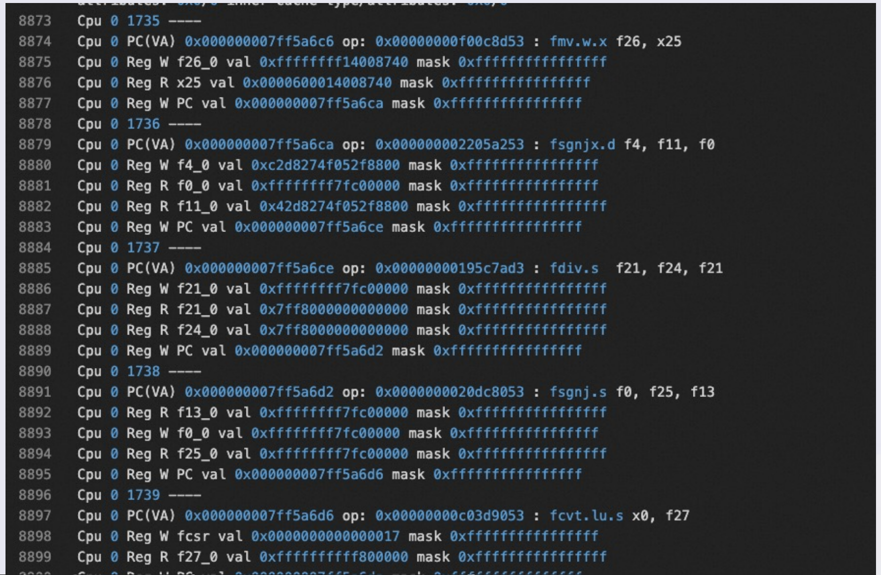
```
class MyMainSequence(Sequence):
    def generate(self, **kwargs):

        choices_mod = ChoicesModifier(self.genThread)

        for _ in range(10):
     mvaden, 3 years ago • pick up um_choiceMod_02 & corresponding add to um…
            self._gen_rv_g_instructions()

            # Modify the choices settings
            choices_mod.modifyOperandChoices(
                "Rounding mode",
                {"RNE": 0, "RTZ": 0, "RDN": 50, "RUP": 0, "RMM": 0, "DYN": 50},
            )

            choices_mod.modifyOperandChoices(
                "Read after write address reuse", {"No reuse": 50, "Reuse": 50}
            )
            choices_mod.commitSet()

            # generate instructions
            self._gen_data_processing_and_load_store_instructions()

            # undo the choices settings - revert back to prior
            choices_mod.revert()

    def _gen_rv_g_instructions(self):
        for _ in range(20):
            if self.getGlobalState("AppRegisterWidth") == 32:
                instr = self.pickWeighted(RV32_G_instructions)
            else:
                instr = self.pickWeighted(RV_G_instructions)

            self.genInstruction(instr)

    def _gen_data_processing_and_load_store_instructions(self):
        for _ in range(20):
            if self.getGlobalState("AppRegisterWidth") == 32:
                instr_mix = {RV32F_map: 10, LDST32_All_map: 10}
            else:
                instr_mix = {ALU_Float_All_map: 10, LDST_All_map: 10}

            instr = self.pickWeighted(instr_mix)
            self.genInstruction(instr)
```

Command:



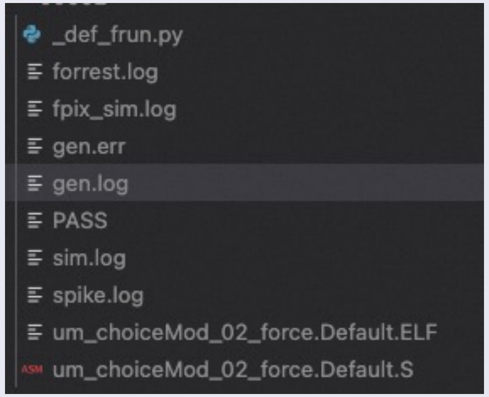
```
GenCmd = {'app': 'force', 'command': '/examples/riscv/um_choiceMod_02_force.py --max-instr 50000 --num-chips 1
--num-cores 1 --num-threads 1 -s 0x4a12cdf8ea926087 --cfg config/riscv_xthead_rv64_default.config  --max-instr 50000
--options "PrivilegeLevel=1"', 'log': 'gen.log', 'elog': 'gen.err', 'max-instr': 50000, 'min-instr': 1}

ISSCommand = {'command': 'force-riscv/fpix/bin/fpix_riscv --railhouse fpix_riscv.railhouse --cluster_num 1 --core_num 1
--threads_per_cpu 1 -i 50000 --cfg config/riscv_xthead_rv64_default.config um_choiceMod_02_force.Default.ELF'}
```

Fpix_sim.log:



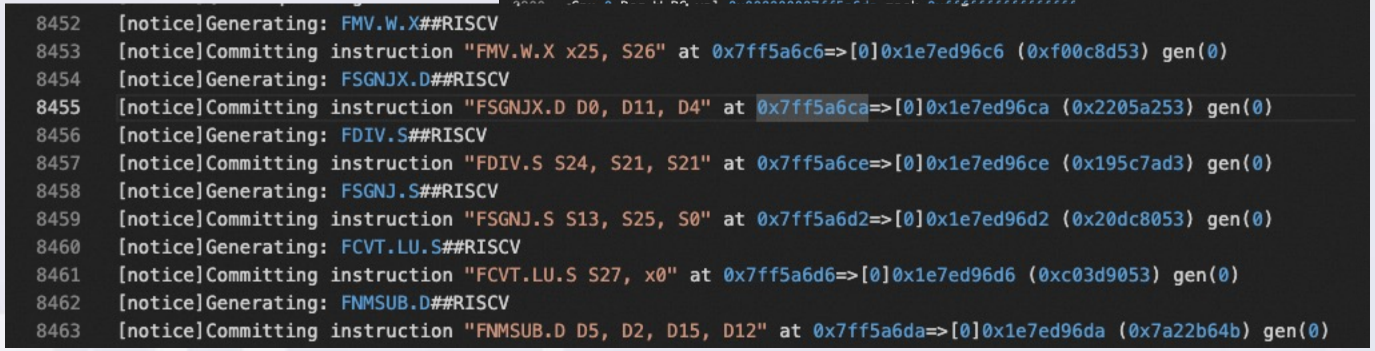
```
Cpu 0 1735 ----
Cpu 0 PC(VA) 0x000000007ff5a6c6 op: 0x00000000f00c8d53 : fmv.w.x f26, x25
Cpu 0 Reg W f26_0 val 0xffffffff14008740 mask 0xffffffffffffffff
Cpu 0 Reg R x25 val 0x0000600014008740 mask 0xffffffffffffffff
Cpu 0 Reg W PC val 0x000000007ff5a6ca mask 0xffffffffffffffff
Cpu 0 1736 ----
Cpu 0 PC(VA) 0x000000007ff5a6ca op: 0x000000002205a253 : fsgnjx.d f4, f11, f0
Cpu 0 Reg W f4_0 val 0xc2d8274f052f8800 mask 0xffffffffffffffff
Cpu 0 Reg R f0_0 val 0xffffffff7fc00000 mask 0xffffffffffffffff
Cpu 0 Reg R f11_0 val 0x42d8274f052f8800 mask 0xffffffffffffffff
Cpu 0 Reg W PC val 0x000000007ff5a6ce mask 0xffffffffffffffff
Cpu 0 1737 ----
Cpu 0 PC(VA) 0x000000007ff5a6ce op: 0x00000000195c7ad3 : fdiv.s  f21, f24, f21
Cpu 0 Reg W f21_0 val 0xffffffff7fc00000 mask 0xffffffffffffffff
Cpu 0 Reg R f21_0 val 0x7ff8000000000000 mask 0xffffffffffffffff
Cpu 0 Reg R f24_0 val 0x7ff8000000000000 mask 0xffffffffffffffff
Cpu 0 Reg W PC val 0x000000007ff5a6d2 mask 0xffffffffffffffff
Cpu 0 1738 ----
Cpu 0 PC(VA) 0x000000007ff5a6d2 op: 0x0000000020dc8053 : fsgnj.s f0, f25, f13
Cpu 0 Reg R f13_0 val 0xffffffff7fc00000 mask 0xffffffffffffffff
Cpu 0 Reg W f0_0 val 0xffffffff7fc00000 mask 0xffffffffffffffff
Cpu 0 Reg R f25_0 val 0xffffffff7fc00000 mask 0xffffffffffffffff
Cpu 0 Reg W PC val 0x000000007ff5a6d6 mask 0xffffffffffffffff
Cpu 0 1739 ----
Cpu 0 PC(VA) 0x000000007ff5a6d6 op: 0x00000000c03d9053 : fcvt.lu.s x0, f27
Cpu 0 Reg W fcsr val 0x0000000000000017 mask 0xffffffffffffffff
Cpu 0 Reg R f27_0 val 0xfffffffff800000 mask 0xffffffffffffffff
```

Generated files:



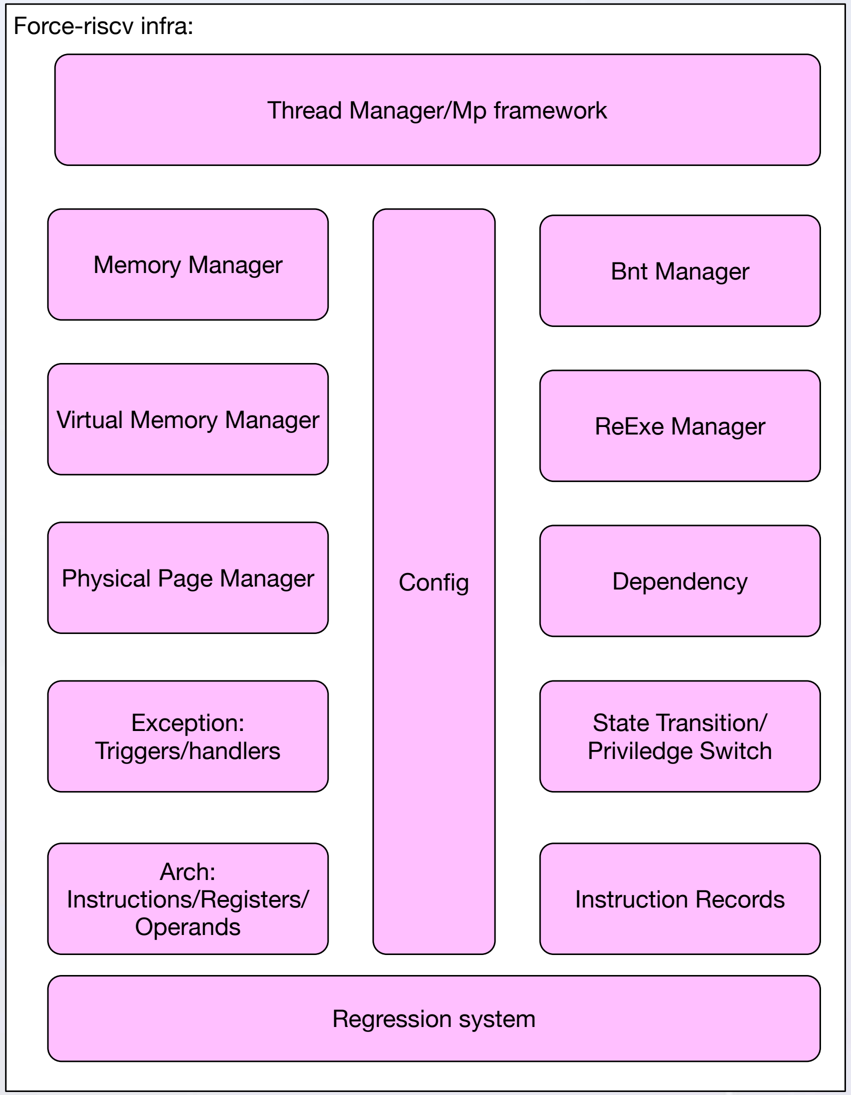
```
_def_frun.py
forrest.log
fpix_sim.log
gen.err
gen.log
PASS
sim.log
spike.log
um_choiceMod_02_force.Default.ELF
um_choiceMod_02_force.Default.S
```

Gen.log:



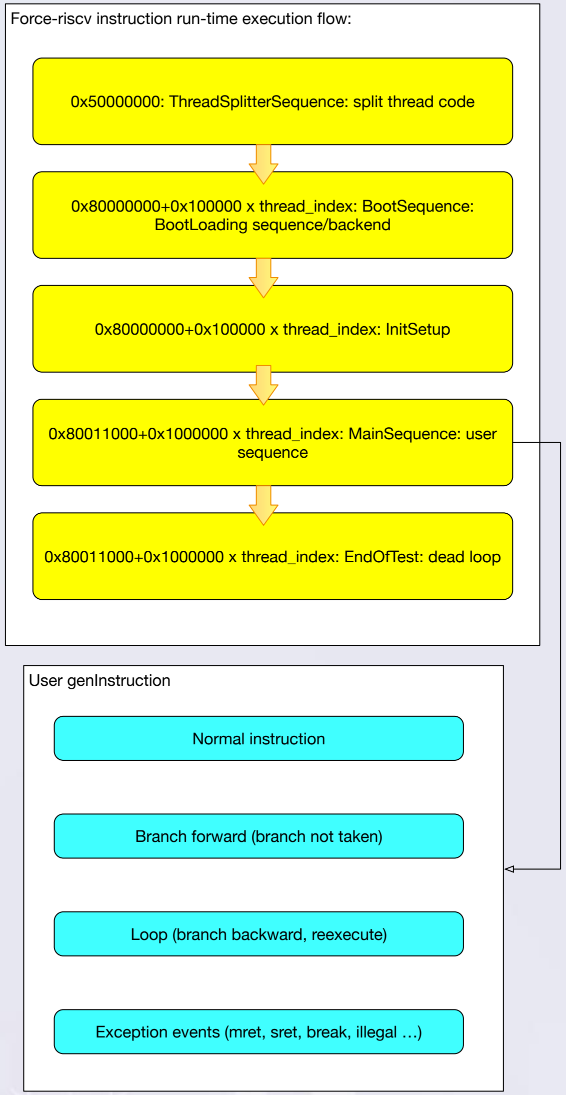
```
[notice]Generating: FMV.W.X##RISCV
[notice]Committing instruction "FMV.W.X x25, S26" at 0x7ff5a6c6=>[0]0x1e7ed96c6 (0xf00c8d53) gen(0)
[notice]Generating: FSGNJX.D##RISCV
[notice]Committing instruction "FSGNJX.D D0, D11, D4" at 0x7ff5a6ca=>[0]0x1e7ed96ca (0x2205a253) gen(0)
[notice]Generating: FDIV.S##RISCV
[notice]Committing instruction "FDIV.S S24, S21, S21" at 0x7ff5a6ce=>[0]0x1e7ed96ce (0x195c7ad3) gen(0)
[notice]Generating: FSGNJ.S##RISCV
[notice]Committing instruction "FSGNJ.S S13, S25, S0" at 0x7ff5a6d2=>[0]0x1e7ed96d2 (0x20dc8053) gen(0)
[notice]Generating: FCVT.LU.S##RISCV
[notice]Committing instruction "FCVT.LU.S S27, x0" at 0x7ff5a6d6=>[0]0x1e7ed96d6 (0xc03d9053) gen(0)
[notice]Generating: FNMSUB.D##RISCV
[notice]Committing instruction "FNMSUB.D D5, D2, D15, D12" at 0x7ff5a6da=>[0]0x1e7ed96da (0x7a22b64b) gen(0)
```

What’ s force-riscv?



Force-riscv infra:

- Thread Manager/Mp framework
- Memory Manager
- Virtual Memory Manager
- Physical Page Manager
- Exception: Triggers/handlers
- Arch: Instructions/Registers/ Operands
- Config
- Bnt Manager
- ReExe Manager
- Dependency
- State Transition/ Priviledge Switch
- Instruction Records
- Regression system

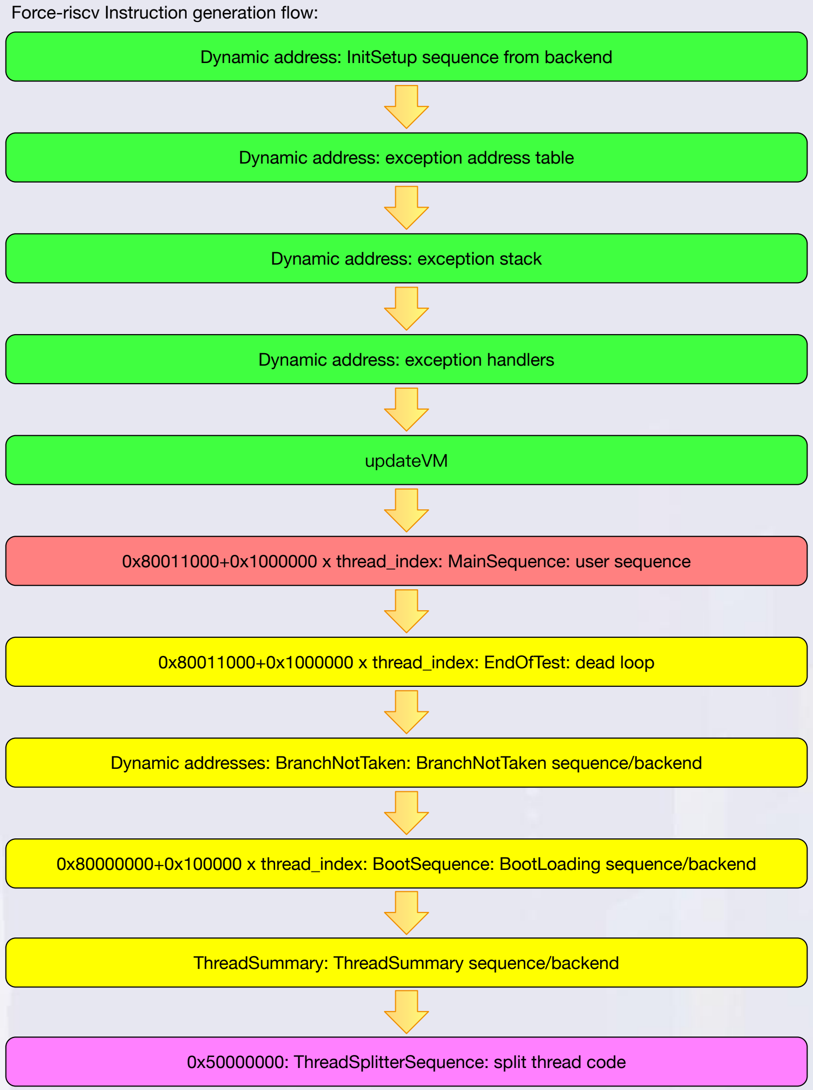


Force-riscv instruction run-time execution flow:

- 0x50000000: ThreadSplitterSequence: split thread code
- 0x80000000+0x100000 x thread_index: BootSequence: BootLoading sequence/backend
- 0x80000000+0x100000 x thread_index: InitSetup
- 0x80011000+0x1000000 x thread_index: MainSequence: user sequence
- 0x80011000+0x1000000 x thread_index: EndOfTest: dead loop

User genInstruction

- Normal instruction
- Branch forward (branch not taken)
- Loop (branch backward, reexecute)
- Exception events (mret, sret, break, illegal …)

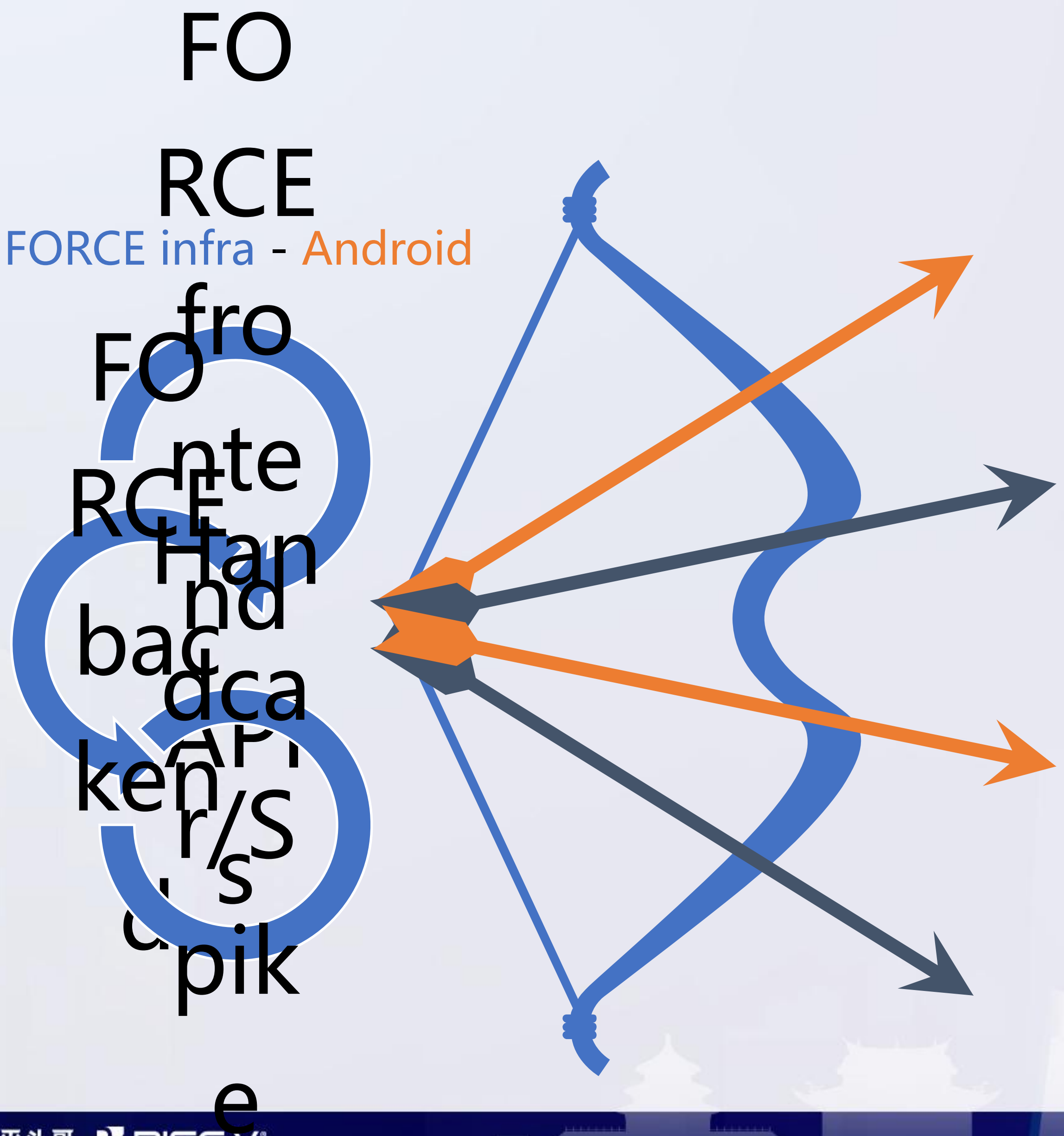


Force-riscv Instruction generation flow:

- Dynamic address: InitSetup sequence from backend
- Dynamic address: exception address table
- Dynamic address: exception stack
- Dynamic address: exception handlers
- updateVM
- 0x80011000+0x1000000 x thread_index: MainSequence: user sequence
- 0x80011000+0x1000000 x thread_index: EndOfTest: dead loop
- Dynamic addresses: BranchNotTaken: BranchNotTaken sequence/backend
- 0x80000000+0x100000 x thread_index: BootSequence: BootLoading sequence/backend
- ThreadSummary: ThreadSummary sequence/backend
- 0x50000000: ThreadSplitterSequence: split thread code

Deploy story – infra+templstes

FORCE infra - Android



FORCE templates - Applications

1 **Architecture:**

- Auto-generated ISA templates
- Vector/FP: knowledge based data pattern
- Paging: dynamic page table, dynamic context switch

2 **Micro-arch:**

- Branch: branch shadow, loop
- Dependency: register, address
- Memory: hw aware templates

3 **MP: cache coherency killer**

- Assembly compiler, litmus porting
- MP zone based flow + golden memory
- Random sequence library

4 **ML:**

- AI based coverage flow

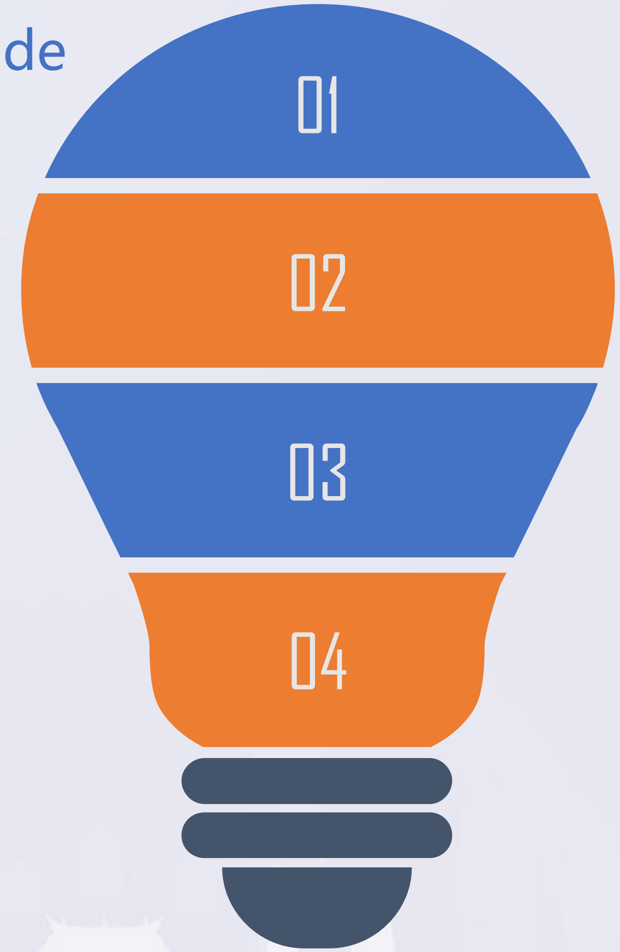
Deploy story – issues of current version

01 Handcar/Spike upgrade

- Version control: patch scheme is hard to control, not friendly
- Latest version merge is hard
- Configuration choice scheme lack for implementation defined features
- …

Build system 02

- Makefile based. compiler options are complex for cross-platform/os support
- Modern IDE need to be supported



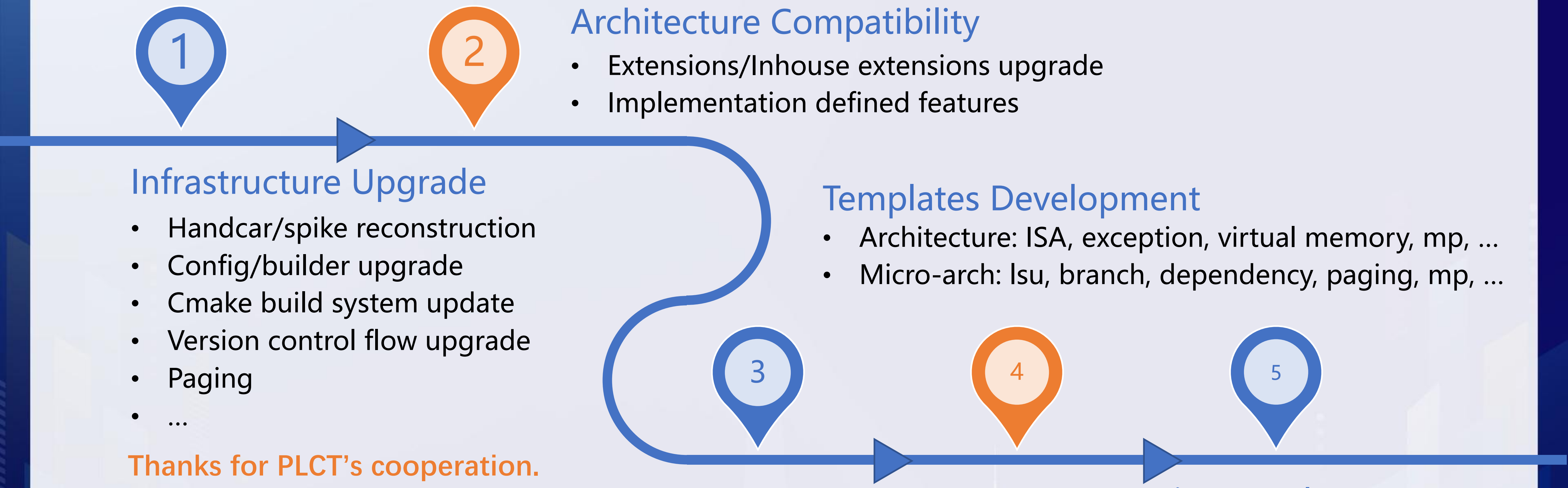
03 Extensions upgrade

- Existing extensions upgrades, vector1.0 …
- New extensions support, RVA22?
- …

Framework upgrade 04

- Multi configurations is hard to support
- MP framework
- …

Deploy story – XuanTie cpu project



Architecture Compatibility

- Extensions/Inhouse extensions upgrade
- Implementation defined features

Infrastructure Upgrade

- Handcar/spike reconstruction
- Config/builder upgrade
- Cmake build system update
- Version control flow upgrade
- Paging
- …

Thanks for PLCT’s cooperation.

Templates Development

- Architecture: ISA, exception, virtual memory, mp, …
- Micro-arch: lsu, branch, dependency, paging, mp, …

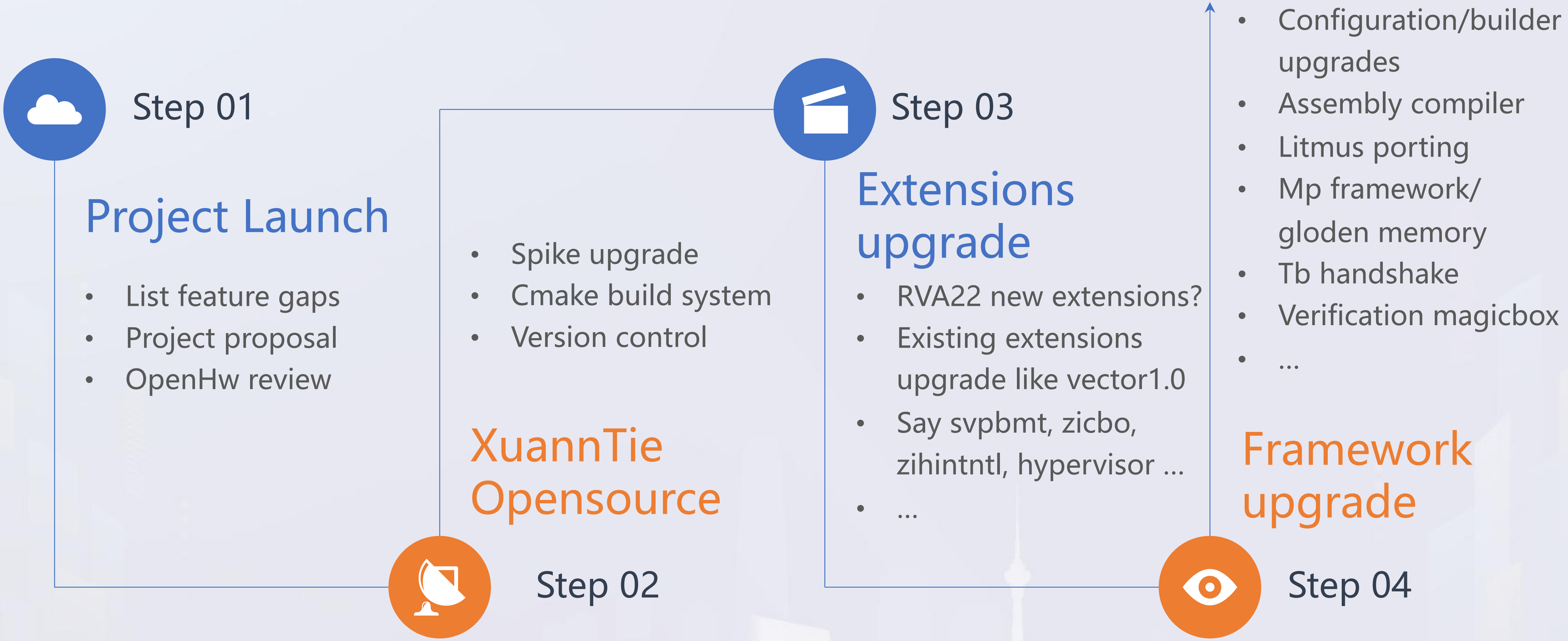
Main Apps Framework

- Custom instruction groups, Inhouse sequence libraries
- Vector infra, mp infra, tb handshake infra, …

Project Deployment

- Xxx, xxx cpu project deploy

Force-riscv stage2 plan with OpenHW



Step 01

Project Launch

- List feature gaps
- Project proposal
- OpenHw review

Step 02

XuannTie Opensource

- Spike upgrade
- Cmake build system
- Version control

Step 03

Extensions upgrade

- RVA22 new extensions?
- Existing extensions upgrade like vector1.0
- Say svpbmt, zicbo, zihintntl, hypervisor …
- …

Step 04

Framework upgrade

- Configuration/builder upgrades
- Assembly compiler
- Litmus porting
- Mp framework/ gloden memory
- Tb handshake
- Verification magicbox
- …

References

- T-HEAD FORCE-RISCV forked repos (early version will be released here):
  - https://github.com/T-head-Semi/force-riscv
  - https://github.com/T-head-Semi/riscv-isa-sim
- FORCE-RISCV repo:
  - https://github.com/openhwgroup/force-riscv
- Former introduction materials:
  - https://github.com/openhwgroup/programs/blob/master/TWG/MeetingPresentations/2020-09-07%20FORCE-RISCV%20ISG%20-%20status.pdf
  - https://www.youtube.com/watch?v=vQecvZm_dmk&t=95s
- Stage2 proposal:
  - https://github.com/openhwgroup/programs/blob/master/Project-Descriptions-and-Plans/FORCE-RISCV/FORCE-RISCV-Project-Launch-revision2023.md
- FORCE-RISCV instance message system:
  - url: https://mattermost.openhwgroup.org/all-users/channels/town-square
  - Create openhw/eclipse account guide: https://www.openhwgroup.org/register/
- RISCV spec:
  - Specifications: https://riscv.org/technical/specifications/
  - RVA22: https://github.com/riscv/riscv-profiles/blob/main/profiles.adoc#rva22-profiles
  - RVA23: https://github.com/riscv/riscv-profiles/blob/main/rva23-profile.adoc

DingTalk ★

Find More

Xuantie @ GitHub

FORCE-RISCV TG