RVCoreP-32IC: An optimized RISC-V soft processor supporting the compressed instructions

International Symposium on Embedded Multicore Socs (MCSoC),2021

Takuto Kanamori
Tokyo Institute of Technology

Contribution

- We propose the instruction fetch unit that supports the compressed instructions of RISC-V and achieves high operating frequency.
- We propose RVCoreP-32IC soft processor that using this unit and implement it on Artix-7 FPGA.
- RVCoreP-32IC achieves 42.5% performance improvement compared with VexRiscv, which is a high-performance soft processor.

What is RISC-V ?

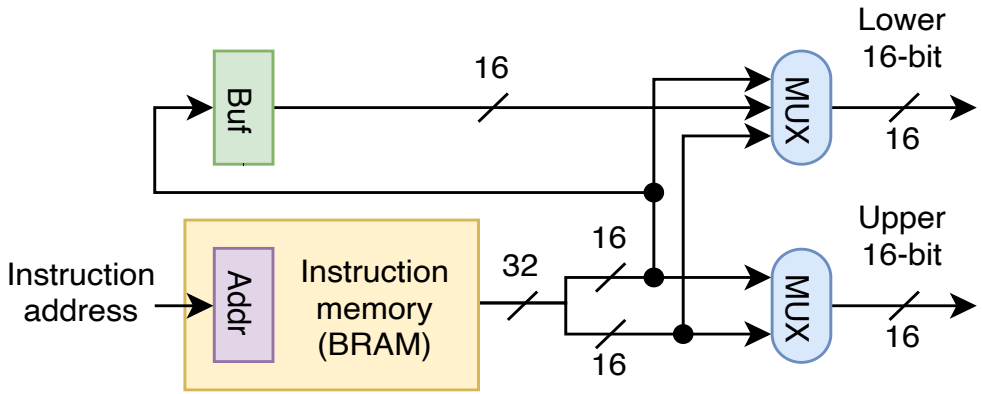
- Open Instruction Set Architecture(ISA)
- RISC-V ecosystems provide software toolchain
- no license fee

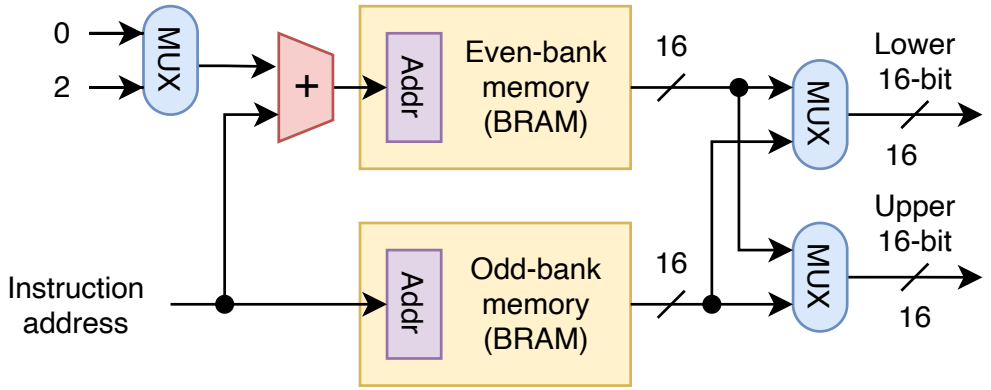
Compressed instructions of RISC-V

- The lower 2-bit indicates the size of an instruction.
  - 11 -> 32bit, other -> 16bit
- This 2-bit is required for calculation of program counter(PC).
  - PC + 4 or PC + 2
  - Align logics and buffers are needed.
  - It may be a critical path and cause of lower operating frequency.

Typical instruction fetch unit for compressed instructions

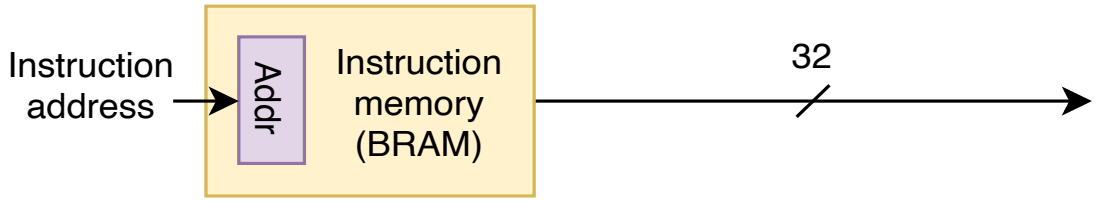
- Buffering method
  - some multiplexers and a buffer are needed
- Banked memory
  - swap logic and an adder are needed





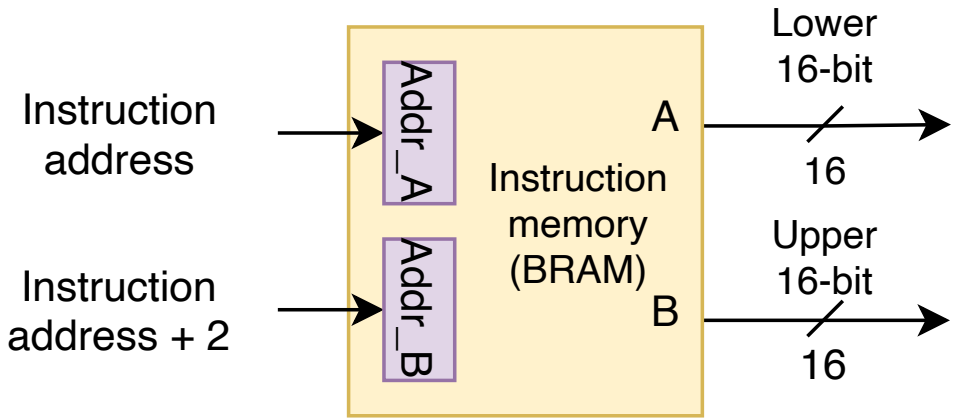
Problem of typical methods

- instructions pass logics after read from memory.
- And this is cause of decreasing operating frequency compared with no compressed instruction.
- We eliminate the logics as below image.



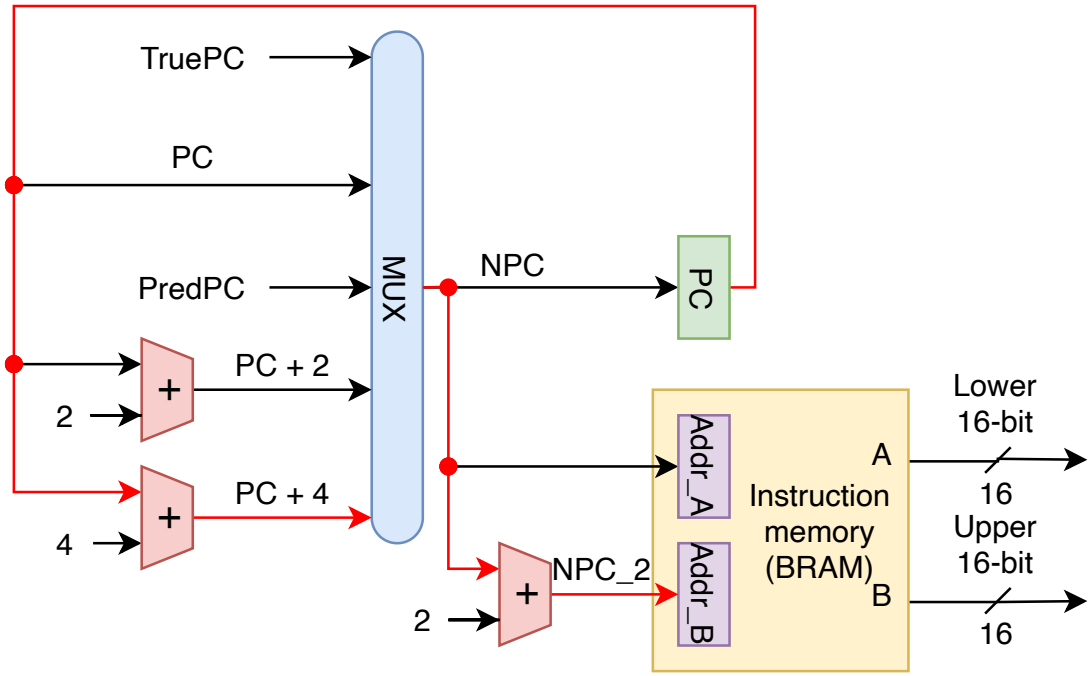
Our approach

- Two port instruction fetch(TIF)
  - BRAM has two port.
  - We change BRAM width 32-bit to 16-bit.
- Logics are no needed for instructions.



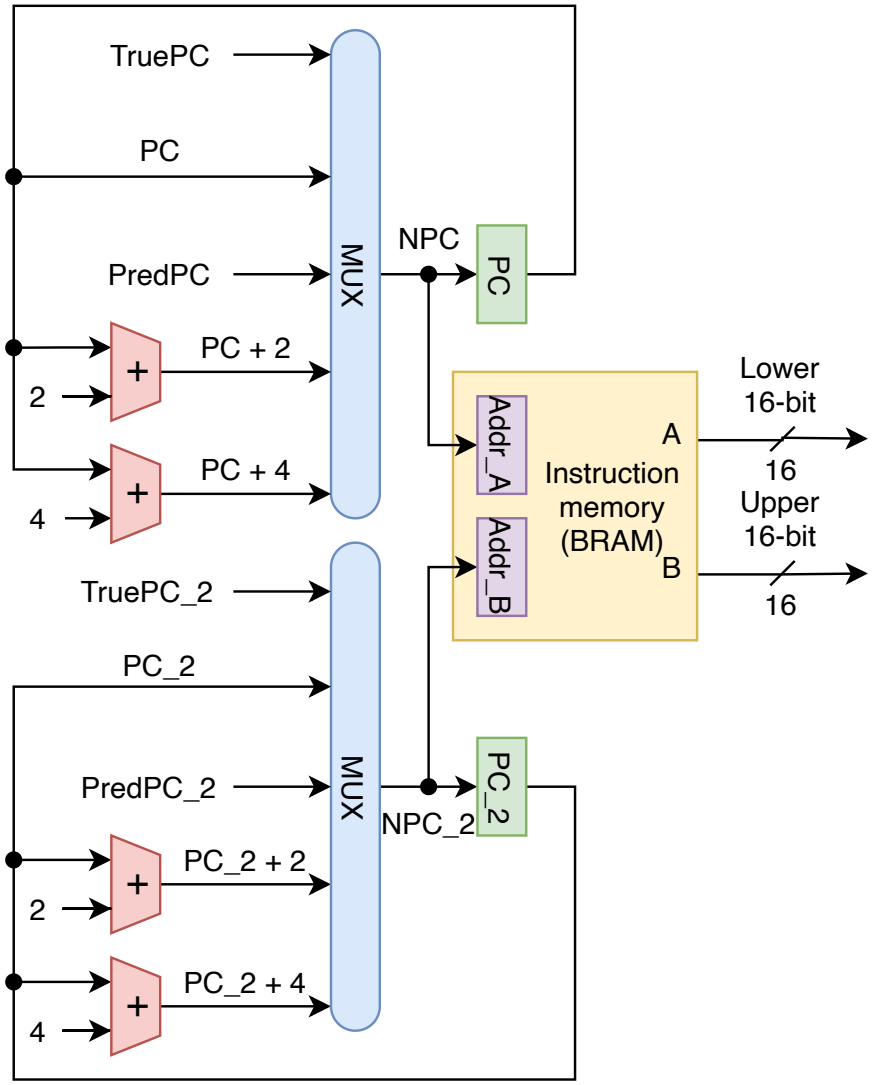
Naive implementation of TIF

- The adder is needed for the next address after calculate PC.
- The red path may become a critical path.



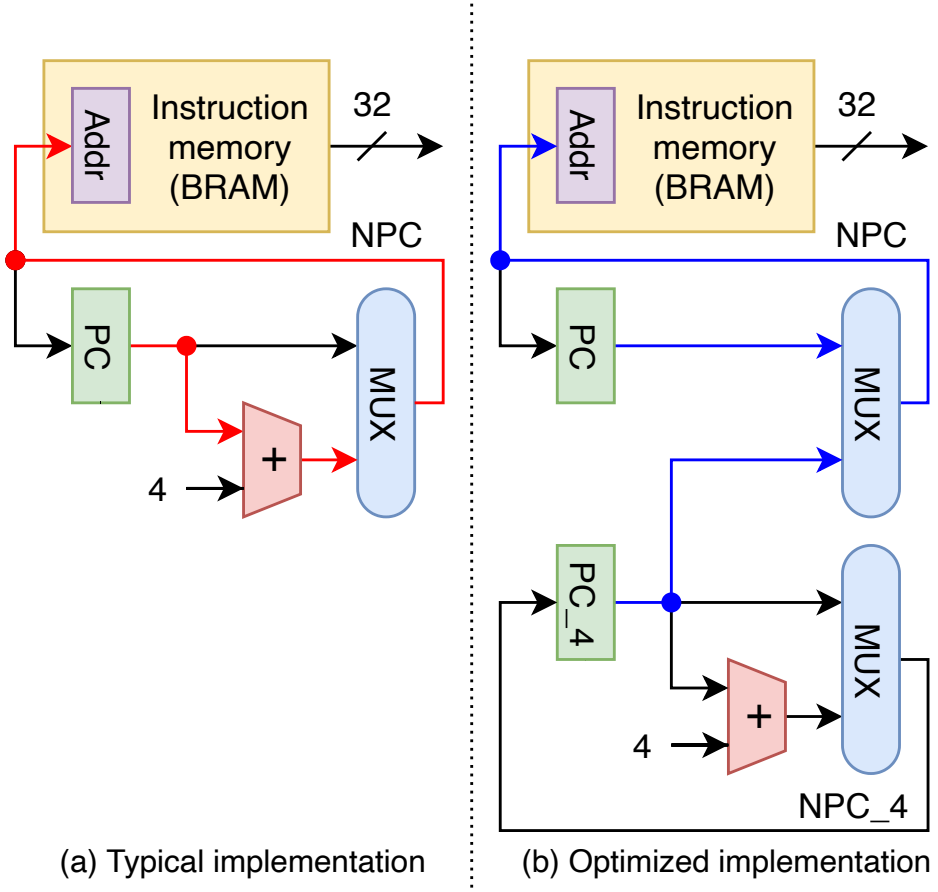
Improved implementation of TIF

- All candidates of PC are calculated in pipeline.
- Each candidates + 2 can be calculated in same time.



More optimization

- Eliminating logics to go into register for BRAM address improves operating frequency.

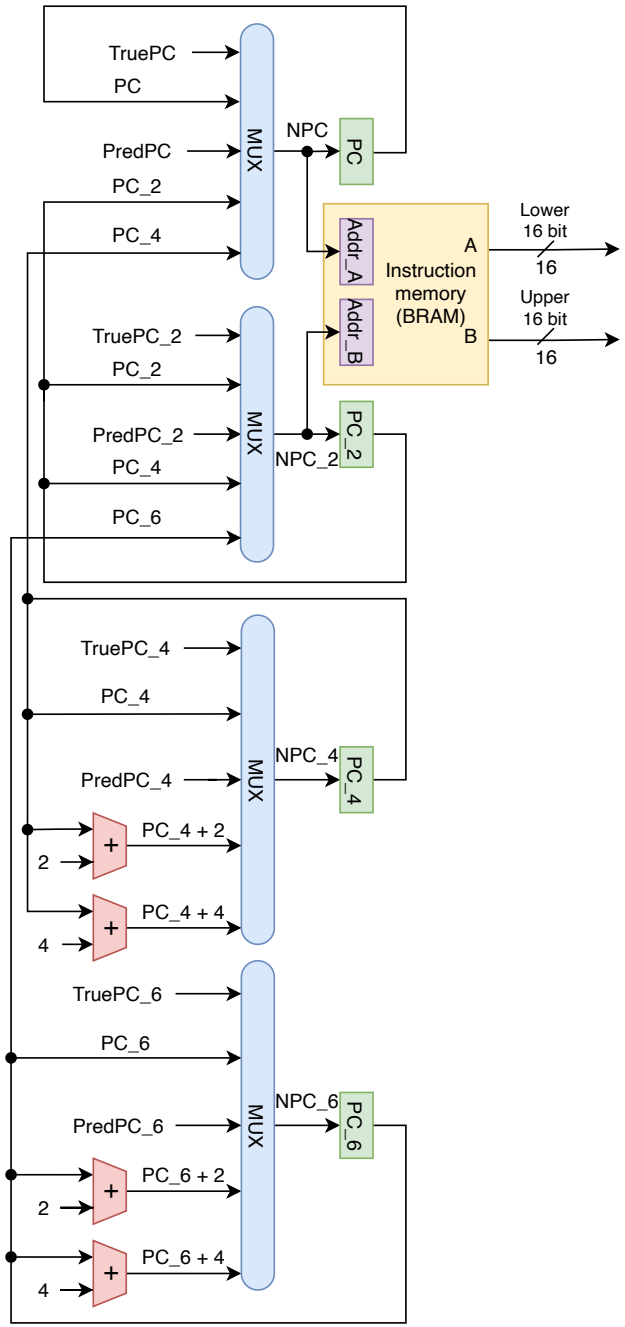


(a) Typical implementation

(b) Optimized implementation

Optimized implementation of TIF

- Eliminate the adders for + 2, +4 in improved implementation.
- And similar logics are added to calculate PC+4, PC+6.

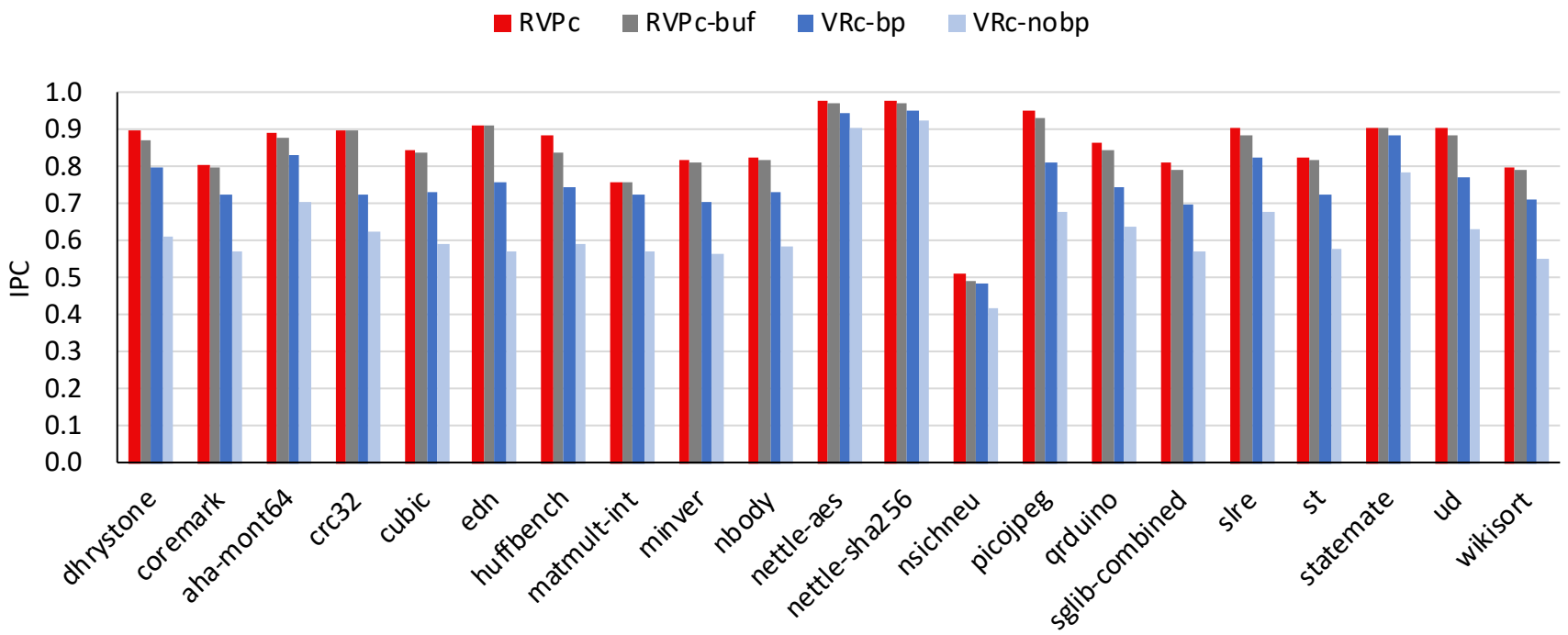


Evaluation

- We modify RVCoreP(RVP) to support the compressed instruction using our proposal fetch unit and name it RVCoreP-32IC.
- We prepare 3 version of RVCoreP-32IC(RVPc).
  - RVPc-opt : using optimized fetch unit
  - RVPc : using improved fetch unit
  - RVPc-buf : using buffering fetch unit (typical)
- We compare the 3 versions and VexRiscv.
  - VexRiscv is high performance in-order & scala soft processor and has won RISC-V processor design contest 2018.

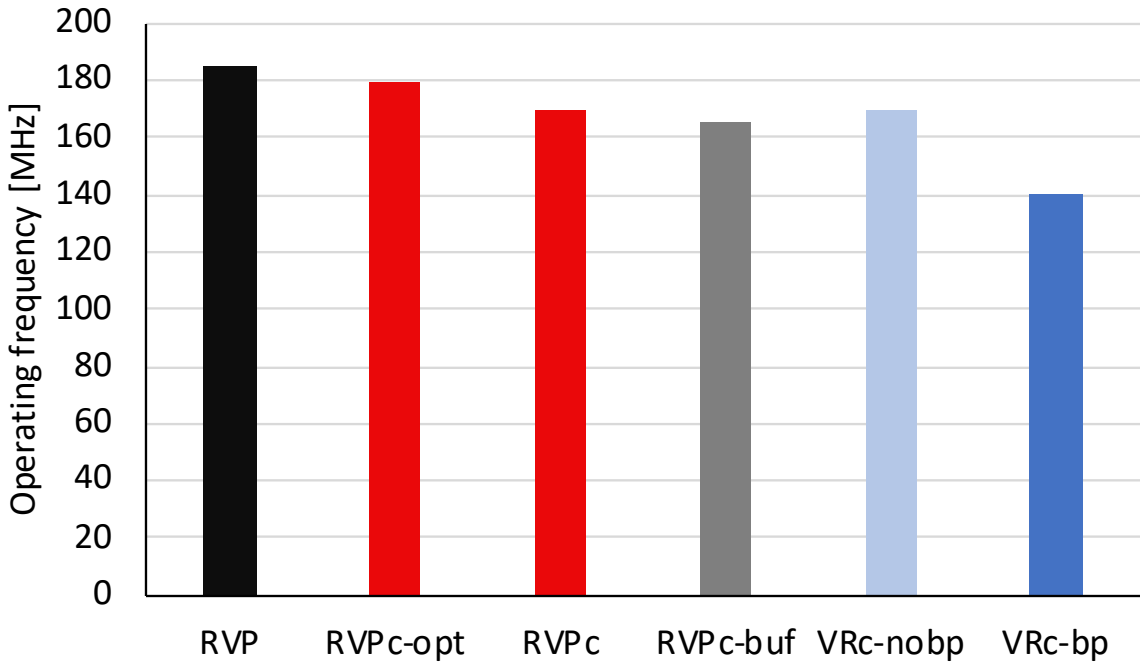
Instruction per cycle (IPC)

- RVPc achieves higher IPCs than RVPc-buf because our fetch unit can fetch even unaligned 32-bits instruction in every cycle.
  - RVPc-opt and RVPc have same IPCs.



Operating frequency

- RVPc-opt minimizes overhead supporting the compressed instruction.
- And achieves the highest operating frequency among versions that support the compressed instruction.



Performance (IPC ✕ Freq)

- RVPc-opt and RVPc achieves 4.6% and 10.8% performance improvement compared with RVPc-buf.
- RVPc-opt achieves 42.5% performance improvement compared with VexRiscv.