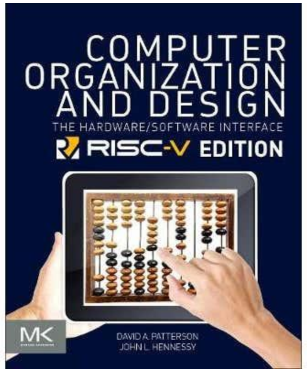
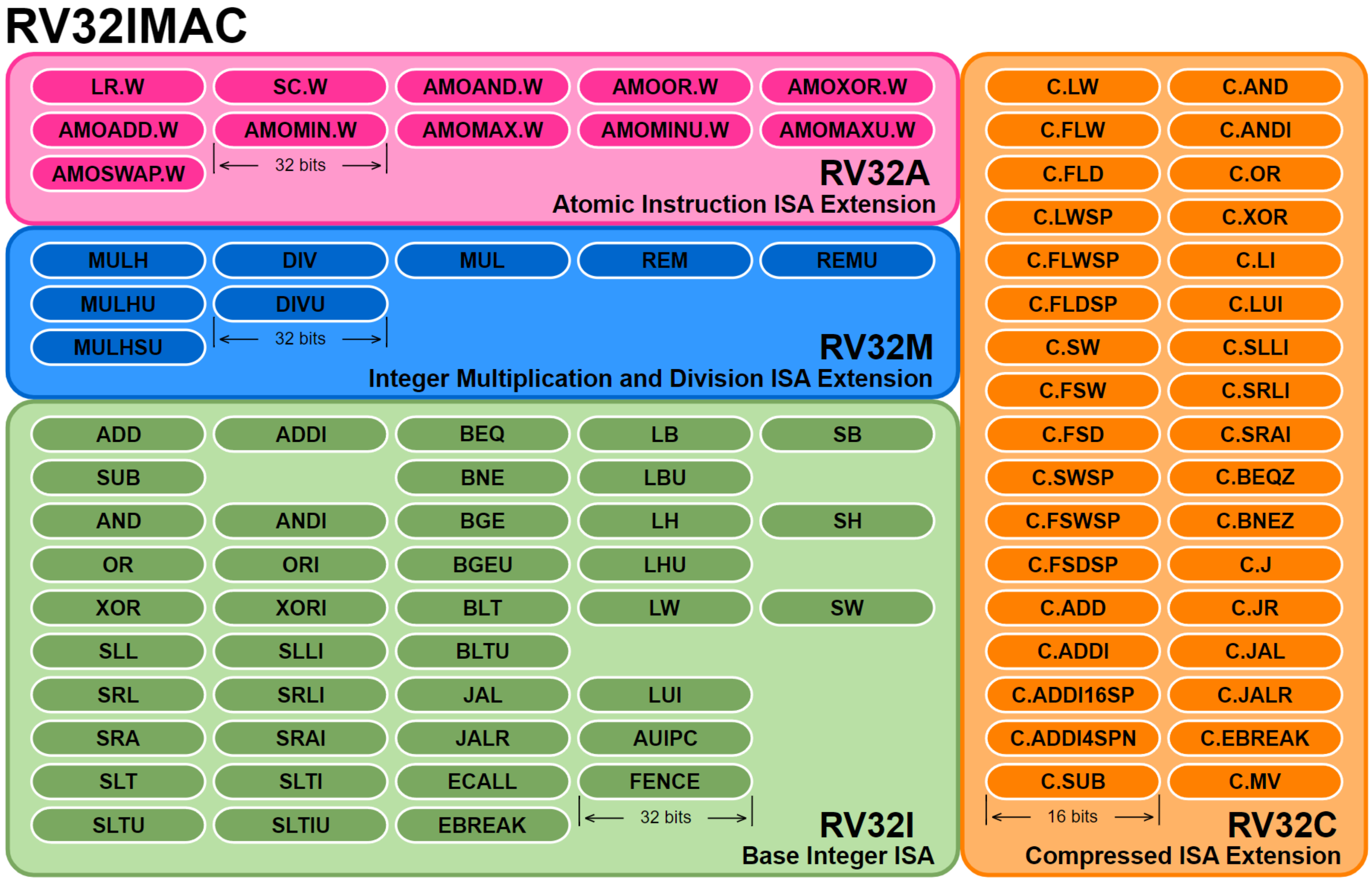
# Build Your Own RISC-V®

- Reduced Instruction Set Computer
- Open instruction architecture
- Modular design
- Your job
  - 32-bit RISC-V with a subset of instructions
- What you’re given:
  - The 5-stage pipeline in VHDL (and SystemVerilog) with a few basic instructions
    - » https://github.com/masoud-ata/PH-RISC-V
    - » https://github.com/masoud-ata/riscv_sv
  - Assembler (and simulator) in Python
    - » https://github.com/masoud-ata/Masimulator



# Modular ISA



## RV32IMAC

|  |  |  |  |  |
|---|---|---|---|---|
| LR.W | SC.W | AMOAND.W | AMOOR.W | AMOXOR.W |
| AMOADD.W | AMOMIN.W | AMOMAX.W | AMOMINU.W | AMOMAXU.W |
| AMOSWAP.W |  |  |  |  |

32 bits

**RV32A**
Atomic Instruction ISA Extension

|  |  |  |  |  |
|---|---|---|---|---|
| MULH | DIV | MUL | REM | REMU |
| MULHU | DIVU |  |  |  |
| MULHSU |  |  |  |  |

32 bits

**RV32M**
Integer Multiplication and Division ISA Extension

|  |  |  |  |  |
|---|---|---|---|---|
| ADD | ADDI | BEQ | LB | SB |
| SUB |  | BNE | LBU |  |
| AND | ANDI | BGE | LH | SH |
| OR | ORI | BGEU | LHU |  |
| XOR | XORI | BLT | LW | SW |
| SLL | SLLI | BLTU |  |  |
| SRL | SRLI | JAL | LUI |  |
| SRA | SRAI | JALR | AUIPC |  |
| SLT | SLTI | ECALL | FENCE |  |
| SLTU | SLTIU | EBREAK |  |  |

32 bits

**RV32I**
Base Integer ISA

|  |  |
|---|---|
| C.LW | C.AND |
| C.FLW | C.ANDI |
| C.FLD | C.OR |
| C.LWSP | C.XOR |
| C.FLWSP | C.LI |
| C.FLDSP | C.LUI |
| C.SW | C.SLLI |
| C.FSW | C.SRLI |
| C.FSD | C.SRAI |
| C.SWSP | C.BEQZ |
| C.FSWSP | C.BNEZ |
| C.FSDSP | C.J |
| C.ADD | C.JR |
| C.ADDI | C.JAL |
| C.ADDI16SP | C.JALR |
| C.ADDI4SPN | C.EBREAK |
| C.SUB | C.MV |

16 bits

**RV32C**
Compressed ISA Extension

# All-FPGA Track

- What is expected:
  - Grade 3: Implement & verify on FPGA
    - Some of RV32I + some of RV32C
    - The necessary assembler modifications
    - Hazard detection (adds pipeline bubble)
    - Serial interface (to send program to the FPGA and read from it)
  - Grade 4: Implement & verify on FPGA
    - Most of RV32I + some of RV32M
    - The necessary assembler modifications
  - Grade 5: Implement & verify on FPGA
    - Some of RV32F
    - The necessary assembler modifications

# FPGA + ASIC Track

- What is expected:
  - Grade 3: Implement & verify on FPGA
    - Some of RV32I + some of RV32C
    - The necessary assembler modifications
    - Hazard detection (adds pipeline bubble)
    - Serial interface (to send program to the FPGA and read from it)
  - Grade 4: Implement & verify on FPGA
    - Most of RV32I + some of RV32M
    - The necessary assembler modifications
  - Grade 5: Implement & verify for ASIC
    - Synthesis & PNR
    - Power analysis
    - Testbench with serial interface support