

# RISC-V CPUコアの Chisel6対応ハマリポイント

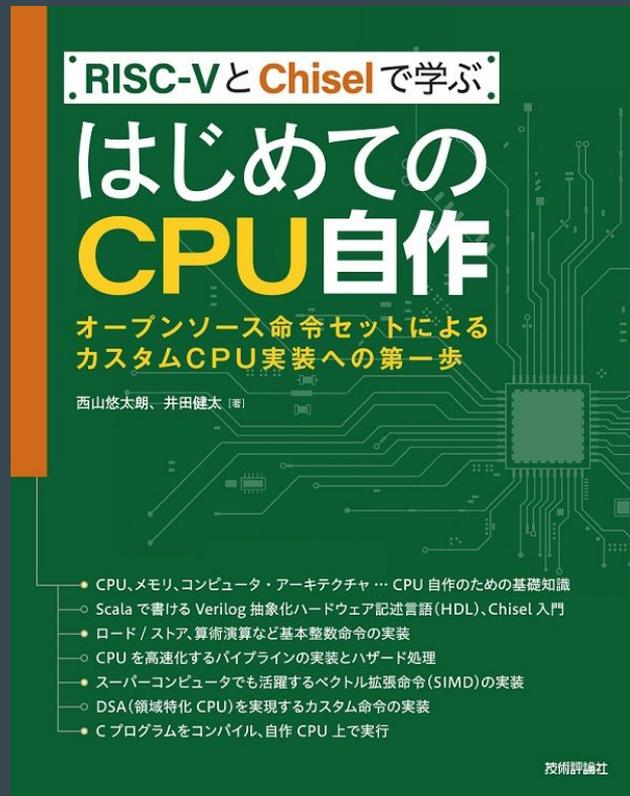
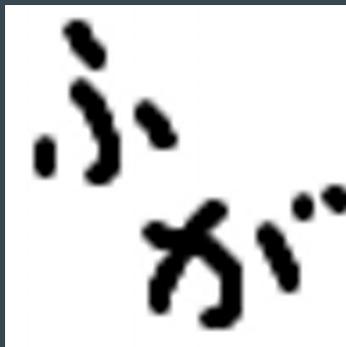
...

2024-06-02

第3回 自作CPUを語る会

# 自己紹介

- 井田 健太
- 元FPGA屋 (Vivadoぽちぽちマン)
- ゆるふわCPU自作勢
- RISC-V CPU自作本の共著者の一人→
  - (私の作業はおまけみたいな感じだけ)
- X: @ciniml



# Chiselとは

- Verilog HDLやVHDLと同じハードウェア記述言語
- **Scala** (JVM向けの言語) の内部DSLとして作られている
- Scalaで表現した論理回路から最終的にVerilog HDLを出力

# Chiselのバージョン

- 最新: Chisel 6.4
- Chisel 3 → 5 → 6 でメジャーバージョン変更
  - 4は欠番
- Chisel 3までは中間表現 (FIRRTL) からVerilog HDLへの変換にScala実装の変換器を使用
- Chisel 5以降はLLVMベースの変換器 CIRCT を使用
  - ちょっとインストール面倒になった
- →出力されるVerilog HDLがかなり変わった

# Chiselのバージョン

- 最新: Chisel 6.4
- Chisel 3 → 5 → 6 でメジャーバージョン変更
  - 4は欠番
- Chisel 3までは中間表現 (FIRRTL) からVerilog HDLへの変換にScala実装の変換器を使用
- Chisel 5以降はLLVMベースの変換器 CIRCT を使用
  - ちょっとインストール面倒になった
- →出力されるVerilog HDLがかなり変わった

# 問題 – FPGA実装がうごかなくなりました！

- RISC-V CPU自作本のコアをFPGA実装
  - Tang Nano 9K (GOWIN FPGA) に実装していた
- Chisel 3では問題なし
- Chisel 6では全くうごかない
- XilinxのFPGAでは動くという報告あり
- →どうもGOWINとの相性ぽい？



## 謎の現象 – デバッグしたら動いた

- CPU内部信号を外部に出力して観測を繰り返し
- どうも0番地からすすんでいないっぽい？
- **id\_op2\_data** (命令デコードステージの第2オペランドの信号) をデバッグ出力したら動くようになった！
  - 「このprint文を消したらうごかない」がHDLで起きるとは..
- どういうこと...？

# 変更前後での出力Verilog HDLの比較

- デバッグ出力前後で比較
- どうもデバッグ出力が無いときは、  
**automatic logic xxx** という記述が生成されている
  - 動くときは同じところの信号は**wire**で出ている

```
798 automatic logic [3:0][31:0] _GEN_16; // cpu_design/src/main/scala/cpu/Core.s
799 automatic logic [7:0][31:0] _GEN_17 =
800 wire [7:0][31:0] _GEN_6 =
801     {{32'h0}},
802     {32'h0},
803     {{id_inst[31:12], 12'h0}},
804     {{{12{id_inst[31]}}, id_inst[19:12], id_inst[20], id_inst[30:21], 1'h0}},
805     {{{20{id_inst[31]}}, id_inst[31:25], id_inst[11:7]}},
806     {{{20{id_inst[31]}}, id_inst[31:20]}},
807     {id_rs2_data},
808     {32'h0}}; // cpu_design/src/main/scala/cpu/Core.scala:8:7, :32:29, :74:38, :167:20,
wire [31:0] id_op2_data =
_GEN_6[_GEN_5
```

## automatic logic?

- ブロック内のローカル変数定義
- 観測用に外とつないだのでwireに変わったぽい
- GOWINの合成系が対応していない模様 -> 常に0になってる
  - 対応していないのはいいがエラーになってくれ! (重要)

```
798 automatic logic [3:0][31:0] _GEN_16; // cpu_design/src/main/scala/cpu/Core.s
799 automatic logic [7:0][31:0] _GEN_17 =
800 wire [7:0][31:0] _GEN_6 =
801     {{32'h0}},
802     {32'h0},
803     {{id_inst[31:12], 12'h0}},
804     {{{12{id_inst[31]}}, id_inst[19:12], id_inst[20], id_inst[30:21], 1'h0}},
805     {{{20{id_inst[31]}}, id_inst[31:25], id_inst[11:7]}},
806     {{{20{id_inst[31]}}, id_inst[31:20]}},
807     {id_rs2_data},
808     {32'h0}}; // cpu_design/src/main/scala/cpu/Core.scala:8:7, :32:29, :74:38, :167:20,
wire [31:0] id_op2_data =
_GEN_6[_GEN_5
```

## 対策 – automatic logicを使わない指定

- Verilog HDLを出力する関数 `emitSystemVerilogFile` にオプション追加
- 第3引数でCIRCTでの変換時のオプションを指定可能
- `disallowLocalVariables`を指定するとよい

```
ChiselStage. emitSystemVerilogFile(  
  new TopWithStepper(memoryPathGen = i =>  
f"${bootromDir}/bootrom_${i}.hex", memorySize = memorySize,  
forSimulation = false, enableProbe = false, useTargetPrimitive =  
true),  
  Array("--target-dir", directory),  
  Array("--lowering-options=disallowLocalVariables")  
)  
  issueあがってた -> https://github.com/llvm/circt/issues/1633
```

## まとめ

- GOWIN FPGAでChisel5以降を使う場合は、必ず **disallowLocalVariables** を指定しよう！