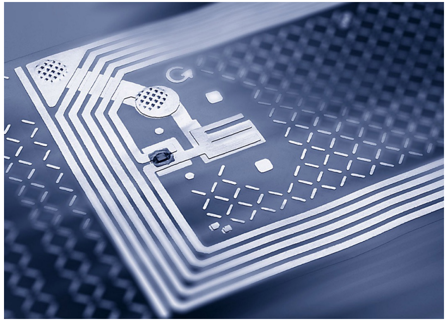
NTNU | Norwegian University of Science and Technology

Minimizing the Energy Usage of Tiny RISC-V Cores

Asbjørn Djupdal, Magnus Själander, Magnus Jahre, Snorre Aunet

Introduction

- Minimizing energy is important for many applications
  - Biomedical devices
  - RFID tags
  - Devices using batteries or energy harvesters



- We evaluate two approaches for low-energy CPU cores
  - Reduce the supply voltage (V_{DD})
  - Reduce the complexity of the core

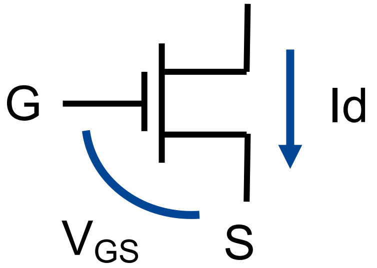
Low Power Electronics

- $Power = \frac{1}{2} \times \alpha \times C \times f \times V^2$

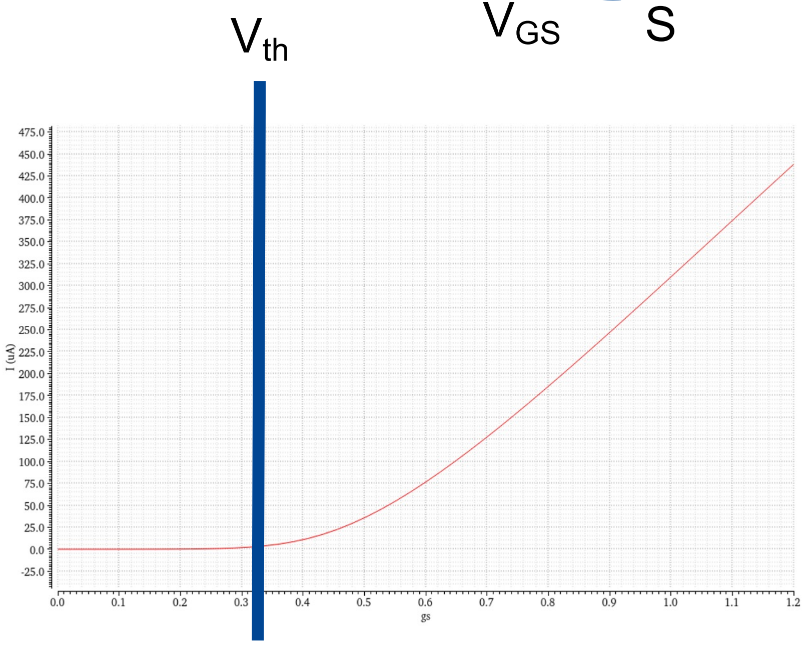
- Reducing supply voltage (V_{DD}) is the most effective way of reducing power

- DVFS is commonly used
  - But normally not reduced below transistor threshold voltage (V_{th})

Subthreshold Electronics



- Transistor threshold voltage (V_{th}):
  - Gate-to-Source voltage needed for transistor to “turn on”
- $V_{DD} < V_{th}$
  - Transistors will never fully turn on
  - Still small currents that depend on the gate-source voltage
  - Can be used for signaling
- Power decreases by orders of magnitude compared to nominal supply voltage



Current vs gate-to-source voltage

Subthreshold Electronics

- First used in Swiss watches in the 1970s
- Extensively researched in the 1990s and 2000s
- Still not commonly used in industry
  - Some commercial implementations exist (Ambiq SPOT)
- Challenges
  - Very slow due to small currents (kHz to a few MHz)
  - Exponential dependencies on the threshold voltage (V_{th})
    - Threshold voltage affected by process variations, temperature, supply voltage variations, …
    - Performance will vary significantly between samples and different environments

Custom Subthreshold Library

- We designed a cell library for a commercial 130nm process
  - V_{th} = 350mV
- Characterized for a range of voltages
  - 250mV — 600mV
- Transistors sized for subthreshold operation
  - Limit V_{th} and frequency variability
- 15 gate types of different sizes
  - 36 gates in total

Energy Efficient Architectures

- What is the most energy efficient CPU architecture?
  - Power is roughly proportional to core complexity (area)
- We decided to investigate small cores of low complexity
- Two tiny cores are compared
  - PicoRV32 (conventional multicycle architecture)
  - SERV (multicycle bit-serial architecture)
- Both configured as RV32E with equivalent feature set
- Both cores use our custom latch-based register file

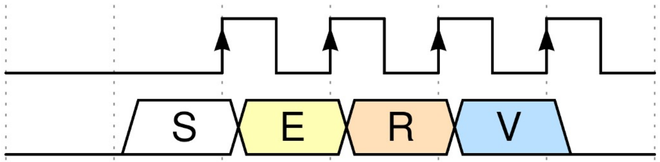
PicoRV32

- Designed to be tiny in area
- Multicycle
  - Pipelined but executes a single instruction at a time
  - Average CPI around 4
- 32-bit parallel datapath
- `https://github.com/YosysHQ/picorv32`

SERV

- Claims to be the world's smallest RISC-V CPU

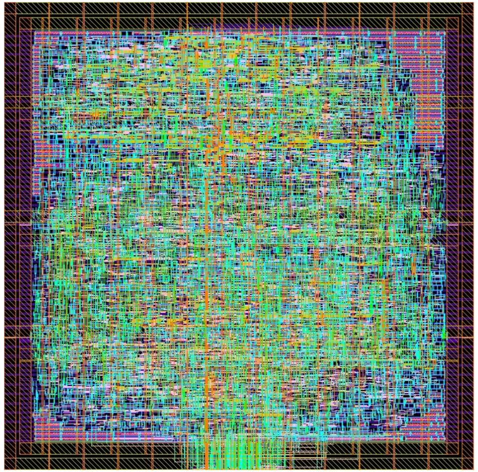
- Bit-serial datapath
  - ALU works on one bit at a time
  - Tiny in area
  - CPI between 35 and 76



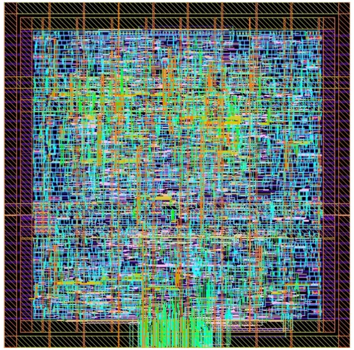
- `https://github.com/olofk/serv`

Physical Layout

- Full physical flow
  - Cadence tools
- Simulated with Spectre
  - SPICE; transistor level simulations
  - Extracted parasitic
- PicoRV32
  - $0.23mm^2$
  - 54 632 transistors
- SERV
  - $0.13mm^2$
  - 27 008 transistors



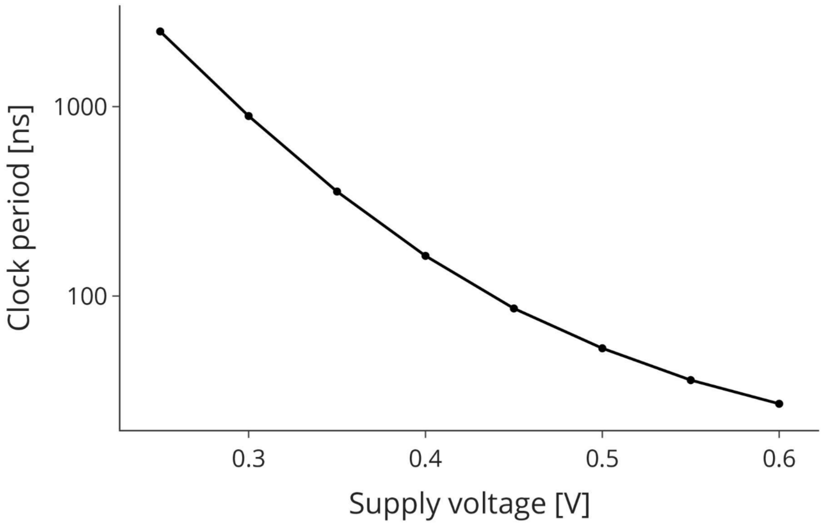
PicoRV32



SERV

Clock vs Supply Voltage

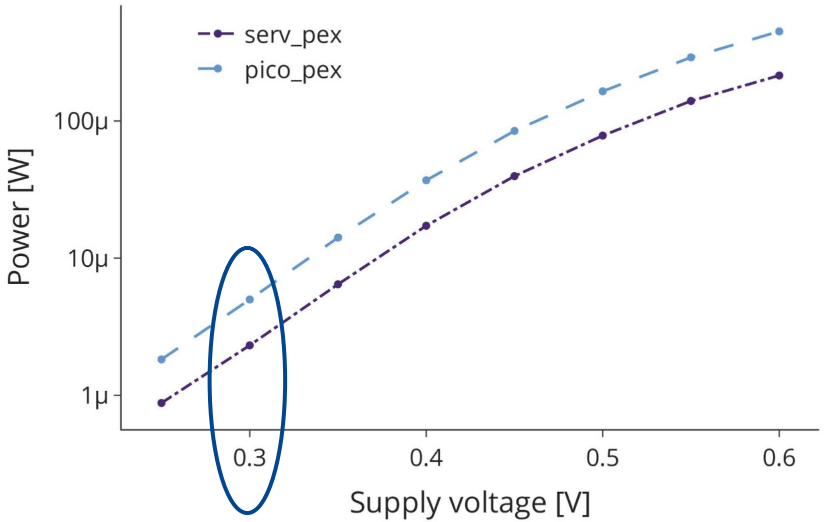
log scale



- Clock period determined by static timing analysis
- Exponential increase in clock period with decreasing supply voltage

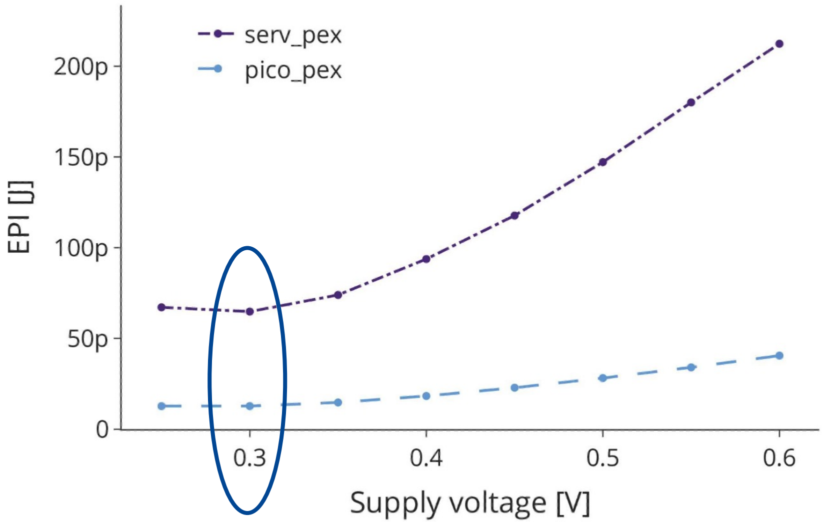
Power vs Supply Voltage

log scale



- Power decreases by three orders of magnitude, going from 1.2V to 300mV
- At 300mV
  - SERV 2.3 μW
  - PicoRV32 5.0 μW
- PicoRV32 dissipates twice as much power as SERV

Energy per Instruction (EPI)



- Energy minimum at 300mV
  - Below threshold voltage
- PicoRV32 one fifth the EPI of SERV
  - SERV 64.8 pJ
  - PicoRV32 12.7 pJ

Subtreshold vs Nominal Supply Voltage

	PicoRV32 @ 300mV	PicoRV32 @ 1.2V	
Power (µW)	2.99	5147.05	1700×
Frequency (MHz)	1.77	250.00	140×
EPI (pJ)	7.63	92.65	12×

Note: Simulated with standard library without extracted parasitic,
i.e., EPI is 7.63 pJ instead of 12.7 pJ at 300 mV

Conclusion

- Energy optimal supply voltage lies below threshold voltage
- The simplest core (SERV) dissipates less power
- The more complex core (PicoRV32) is more energy efficient
  - The increased performance more than makes up for the increased power dissipation