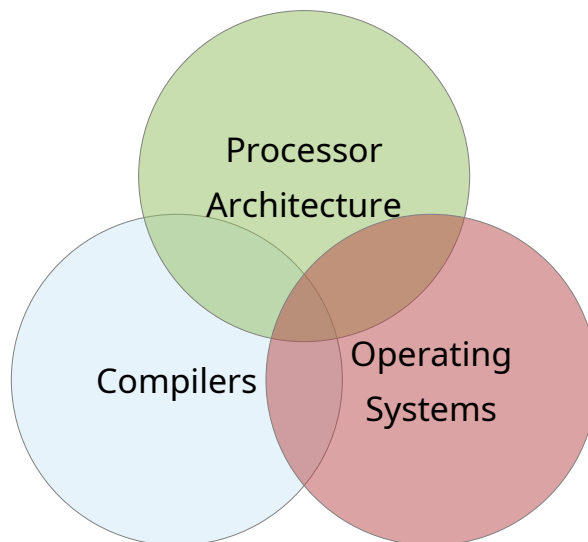


Project suggestions

Sven Karlsson

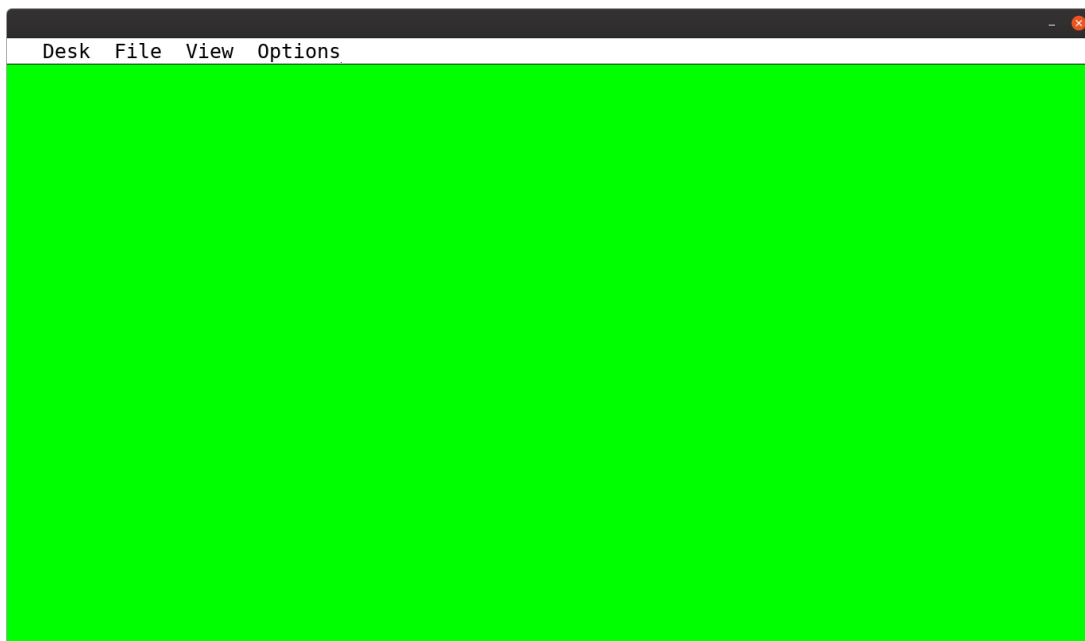
svea@dtu.dk

Research area



FenixOS

- Ambitious research operation system project
- Multiple projects possible
 - Applications
 - Development environments
 - GUI
 - System libraries
 - Kernel subsystems
- Especially interesting if you enjoy retrocomputing

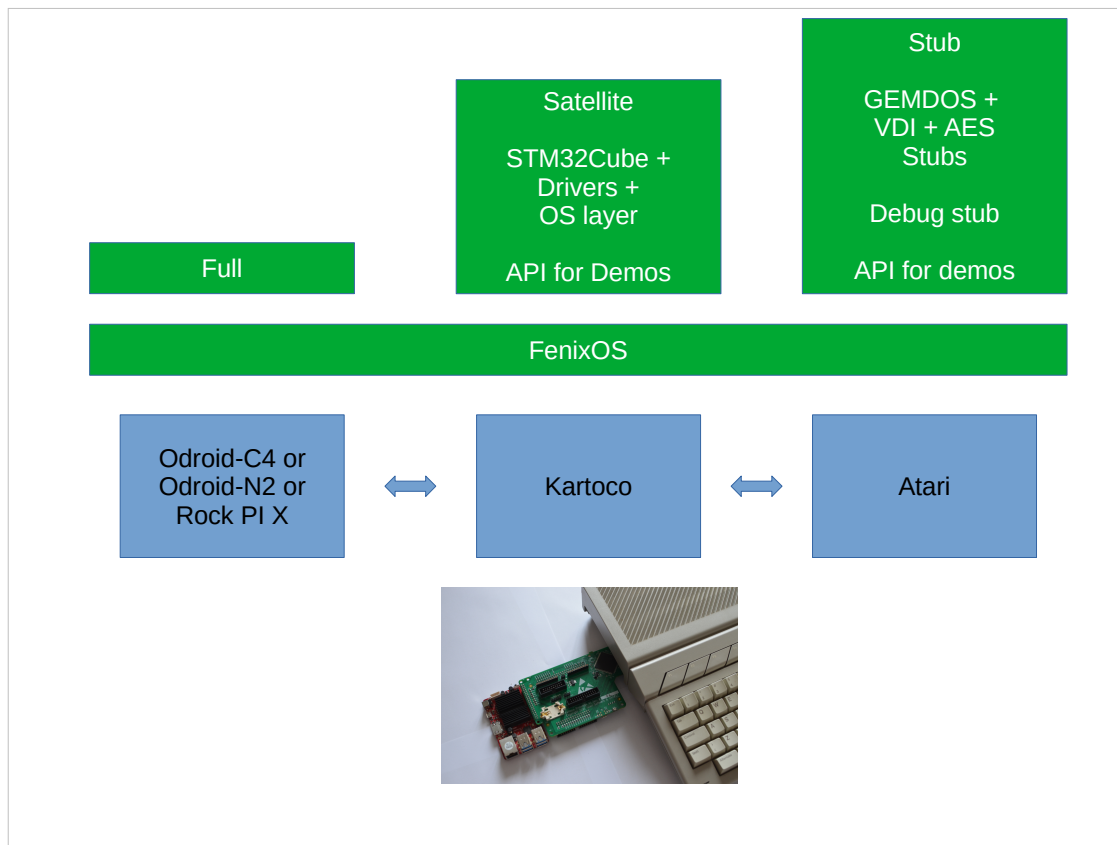


Execution modes

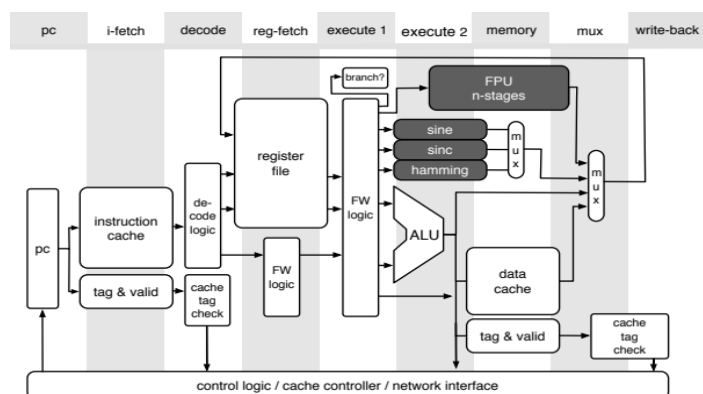
- FenixOS
 - Clean applications
 - 4000x ST, 40x fastest 060 accelerator
- Legacy
 - Sandboxed TOS and hybrid applications with TOS emulation
 - Optional 680x0 emulation
- TOS
 - Sandboxed TOS applications with TOS emulation
 - 680x0 emulation
- Hypervisor
 - Actual TOS or MagiC running
 - 680x0 emulation
- Emulation
 - Full emulation for legacy demos and games

Application environment

- Similar to Snap / Docker
- Applications carry their environment
 - Mapped to a TOS drive
 - Think the U: system from MiNT but expanded
- Home directory and U: is retained from MiNT
 - Drives mapped under U: for a clean file system
- Processes can be moved transparently between machines

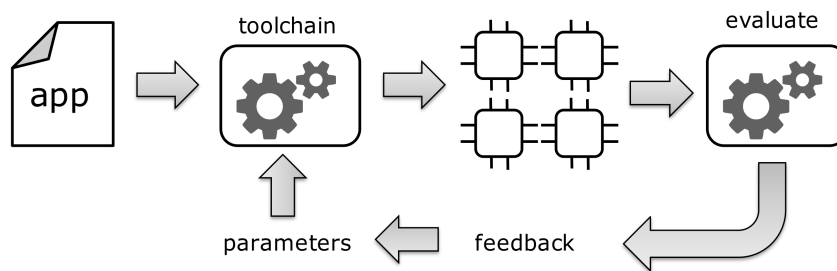


Tinuso : The worlds fastest processor in its class



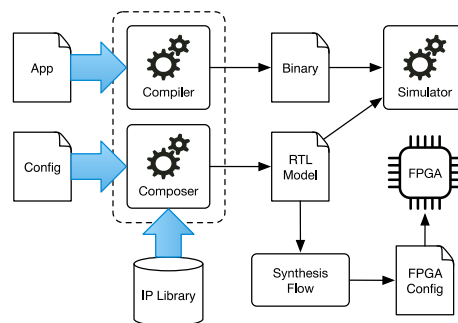
- Designed for high-performance multi-core systems
- Double the clock frequency / half the resources
- Next version 16x faster.

Accelerator composer and design space explorer



- Tool which automatically can compose custom hardware based on application and configuration
- Iterative feedback to designer → agile development
- Can quickly evaluate different strategies

Accelerator composer and design space explorer



- Tool which automatically can compose custom hardware based on application and configuration
- Iterative feedback to designer → agile development
- Can quickly evaluate different strategies



Automatic Optimization

Limitations

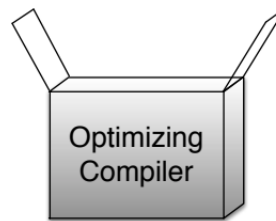
- Not all code can be automatically optimized
- Alternative: vectorize and parallelize by hand



Courtesy of Flickr user Martin Maciaszek

Our Approach

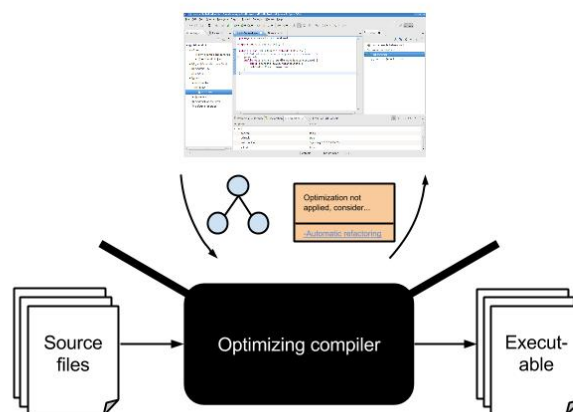
- Let the compiler do the heavy lifting
- Modify code to allow automatic optimization
 - Help the programmer understand why the compiler has not optimized



18

Approach

- Interactive compilation framework
 - Code Commentary
 - Automatic refactorings



19