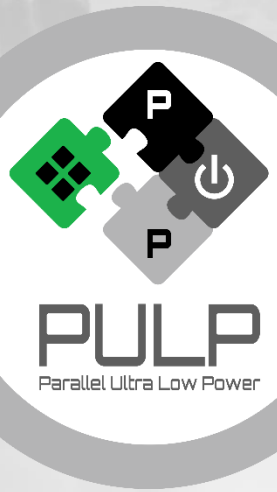


PULP Software Development Kit and Tools

Compiler, Virtual Platform, PulpOS

21.01.2019

Germain Haugou

Andreas Kurth

and the PULP team led by Prof. Luca Benini

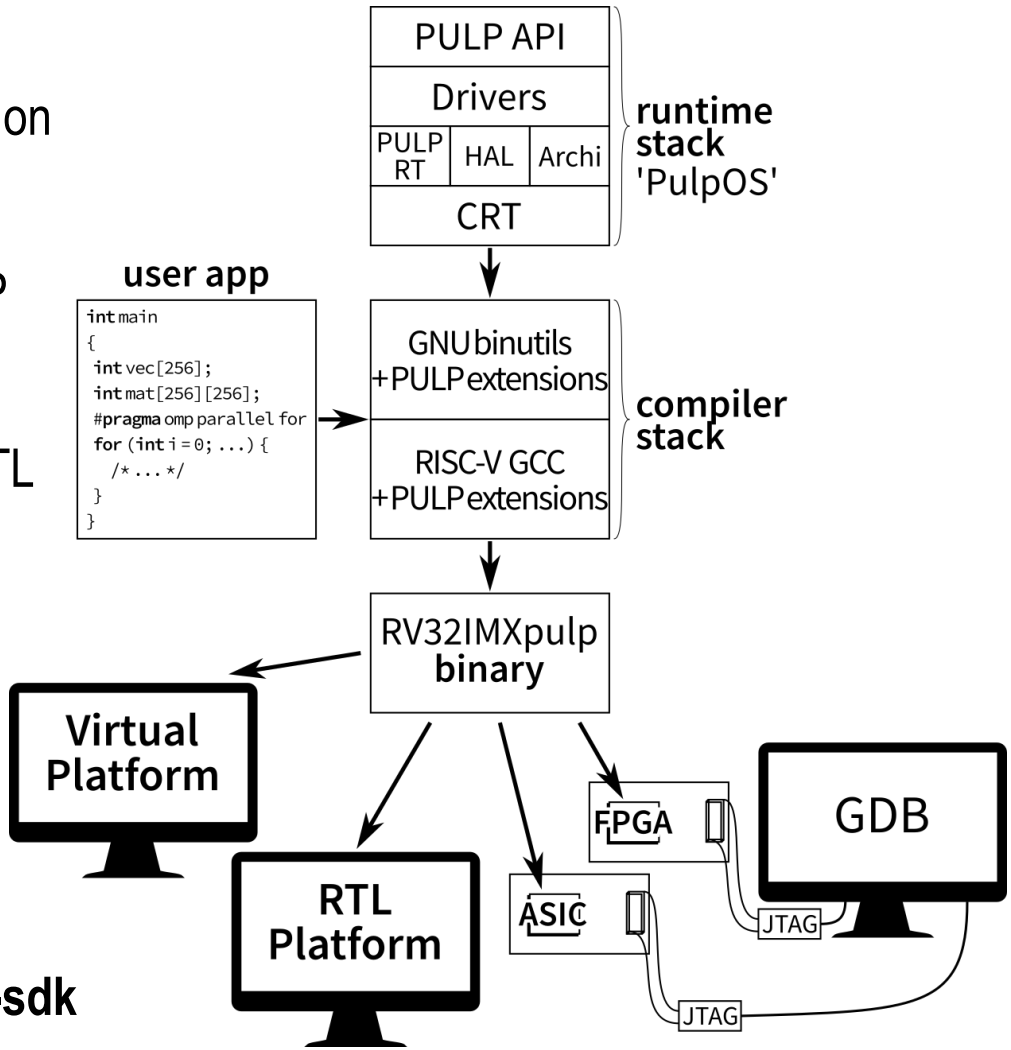
[1]Department of Electrical, Electronic and Information Engineering

ETH zürich

[2]Integrated Systems Laboratory

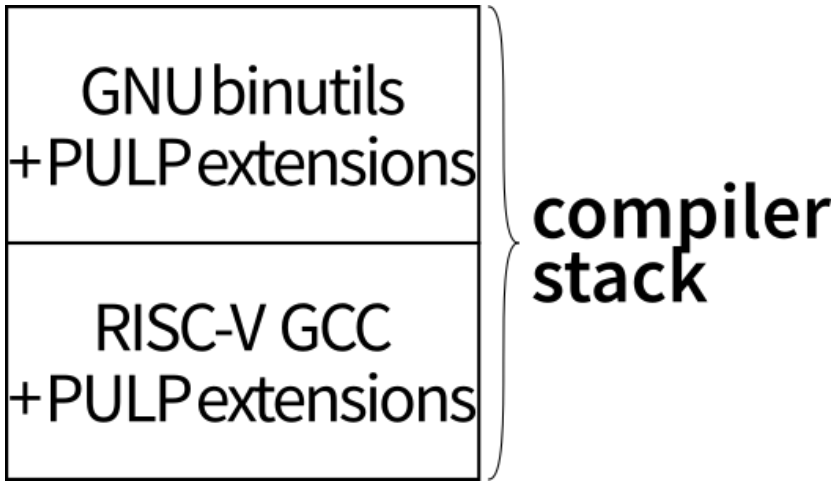
PULP Software Development Kit (SDK)

- Package for **compiling**, **running**, **debugging** and **profiling applications** on **PULP** platforms
- Supports all recent and upcoming PULP chips: Mr.Wolf, GAP, Vega, ...
- Supports all targets: Virtual Platform, RTL platform, FPGA, dev boards
- RISC-V GCC with support for PULP extensions
- Basic OpenMP support
- **Open-source**, available at https://**github**.com/**pulp-platform**/**pulp-sdk**



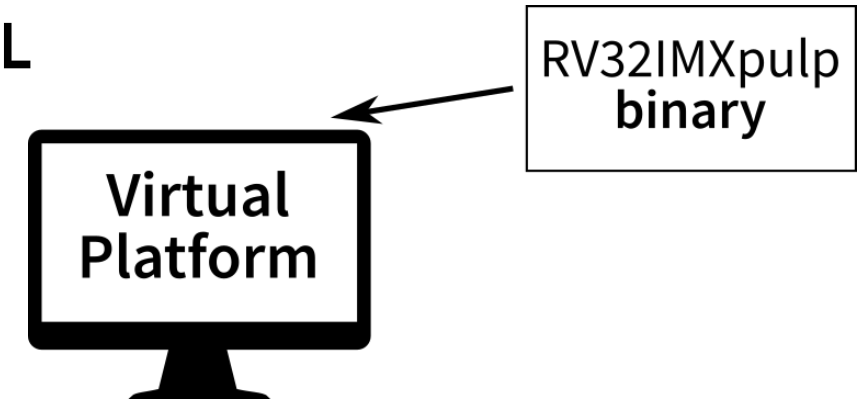
Compiler

- Forked **GCC 7.1**
- **Extended** with **all PULP custom instructions**
- Some **custom instructions instantiated by GCC** (e.g. bit manipulation instructions, auto-vectorization), others available through builtins
- **CoreMark 3.1** with RI5CY v2
- Extended binutils for **full GDB support** of **all custom instructions**



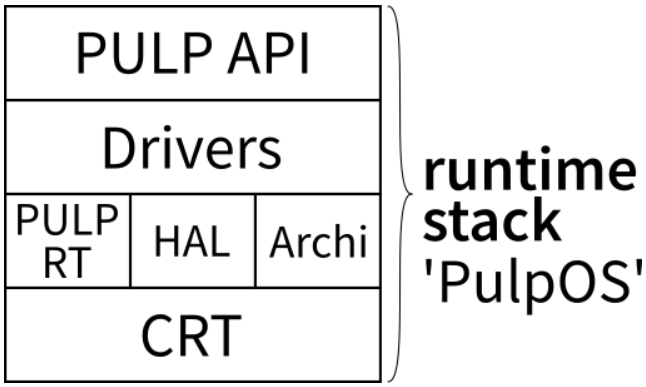
Virtual Platform: Features

- **100% functional equivalence to RTL** (or supposed to)
- **Performance estimation** (20% error margin)
- **Frequency scaling**
- Power on/off
- **Power consumption estimation**
- **Architecture traces**
- **VCD** traces
- **Peripheral** models (camera, ADC, microphone, etc)
- **GDB** connection



Runtime / OS

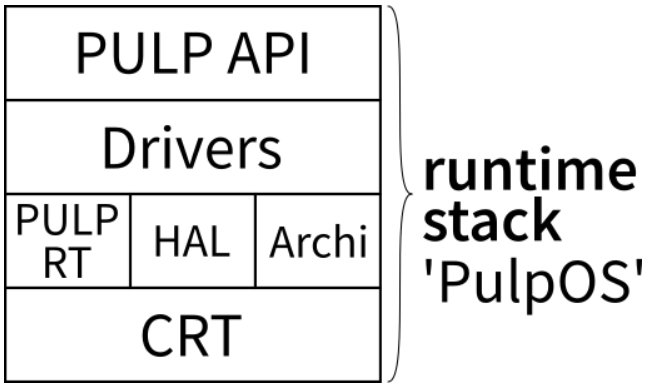
- **PulpOS**
  - Provides a simple OS for quick prototyping
  - Supports all PULP variants, with or without fabric controller (FC) and multiple clusters
  - Used for full applications including drivers, as well as basic tests
  - All APIs are asynchronous to support small reactive applications



- **Zephyr**
  - Just starting now
  - Plan is to port the kernel to PULP platforms, create new API for managing the cluster and port Zephyr drivers (SPI, etc)

PulpOS

- Features
  - **Multi-threading**: to get different priorities
  - **Event scheduler:** one per thread, to schedule run-to-completion tasks (all APIs are asynchronous)
  - **Memory allocators**: for all PULP memory levels (L2, L1)
  - **Cluster management**: cluster mount/unmount, remote cluster call, FC remote services for cluster
  - **Power management**: frequency scaling, deep sleep, voltage scaling
  - **Drivers**: SPI, I2S, I2C, CPI, etc.
  - **Cluster execution**: team fork / barriers / critical sections, DMA transfers



PULP SDK: Getting Started

```
git clone \
 https://github.com/pulp-platform/pulp-sdk
```

Check `README.md` for prerequisites and install them.

`Source` the configuration file of your target platform.

```
make all
```

Questions?

www.pulp-platform.org

@pulp_platform

PULP
Parallel Ultra Low Power

Florian Zaruba[2], Davide Rossi[1], Antonio Pullini[2], Francesco Conti[1], Michael Gautschi[2], Frank K. Gürkaynak[2], Florian Glaser[2], Stefan Mach[2], Giovanni Rovere[2], Igor Loi[1] Davide Schiavone[2], Germain Haugou[2], Manuele Rusci[1], Alessandro Capotondi[1], Giuseppe Tagliavini[1], Daniele Palossi[2], Andrea Marongiu[1,2], Fabio Montagna[1], Simone Benatti[1], Eric Flamand[2], Fabian Schuiki[2], Andreas Kurth[2], Luca Benini[1,2]

[1]Department of Electrical, Electronic and Information Engineering

ETH zürich
[2]Integrated Systems Laboratory