A Raspberry Pi image with ROS 2 + RT and a customizable image builder

Shuhao Wu

Oct 21st 2022

ROSCon '22 Kyoto

CACTUS
DYNAMICS

Who am I?

- Shuhao Wu
  - https://shuhaowu.com
- Masters in Mechanical Engineering
- Staff Production Engineer @ Shopify
  - Distributed systems, system performance
- Occasional robotics consultant @ Cactus Dynamics
  - Robotics software architecture, advanced technology development
- Also collaborating with the ROS real-time working group

CACTUS
DYNAMICS

Background

- ROS is widely used in the industry, academia, and hobbyist setting
  - ~100K users, 1000-2000 citations/year, 600M downloads from ~700k sources[1]
- Wide skill range
  - Beginner: tedious first installation
  - Expert: tricky n-th installation
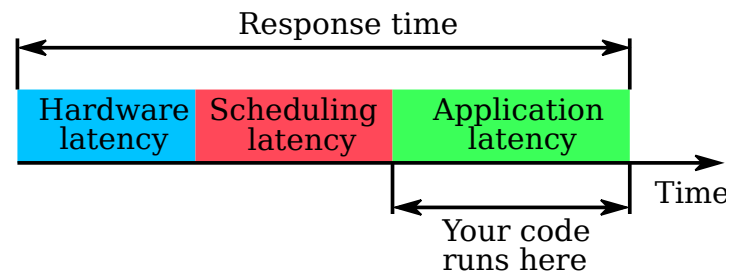- Real-time applications
  - Real-time Linux

[1] Interpreted from the 2021 ROS metrics report

CACTUS DYNAMICS

A downloadable image for ROS 2

- Raspberry Pi image with ROS 2 Humble preinstalled

- Minimal image: ~1GB download
  - No GUI, but easily installable

- Easier for newcomers to get started

- Also included: real-time kernel for real-time applications

CACTUS
DYNAMICS

Real-time and Linux

- Application that has strict deadlines for latency
  - Example: inverted pendulum controller
- Latency comes from different sources[2]
  - OS-level (scheduling) latency must be minimized
- Stock Linux kernel is not real-time
- Linux with PREEMPT_RT is "soft" real-time
  - Must compile your own real-time kernel!
- **This is difficult to do as a newcomer**

Response time

| Hardware latency | Scheduling latency | Application latency |

Time

Your code runs here

[2] More info: https://shuhaowu.com/blog/2022/01-linux-rt-appdev-part1.html

CACTUS
DYNAMICS

Raspberry Pi ROS + RT image

- Image preinstalled with custom-built Linux + PREEMPT_RT
  - https://github.com/ros-realtime/linux-real-time-kernel-builder
- RT configurations
  - CPU frequency, permissions, kernel options...
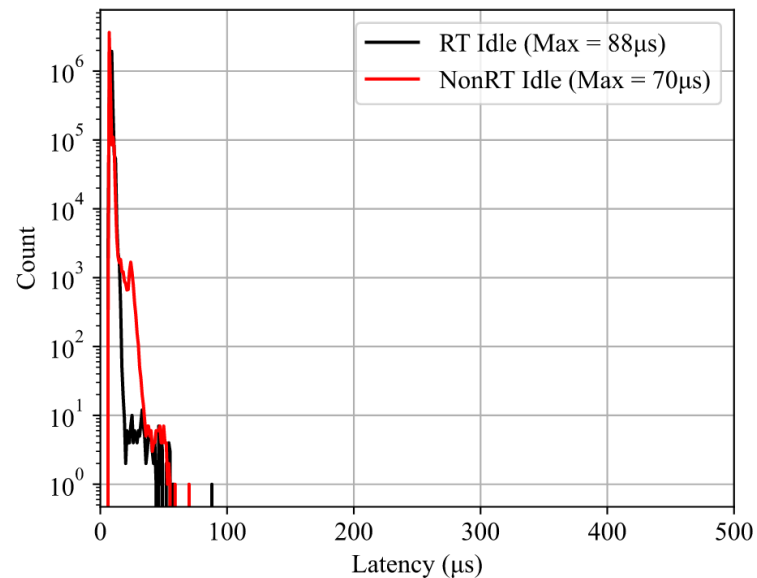- Worst-case scheduling latency measured with validation benchmarks

CACTUS
DYNAMICS

Validation of real-time
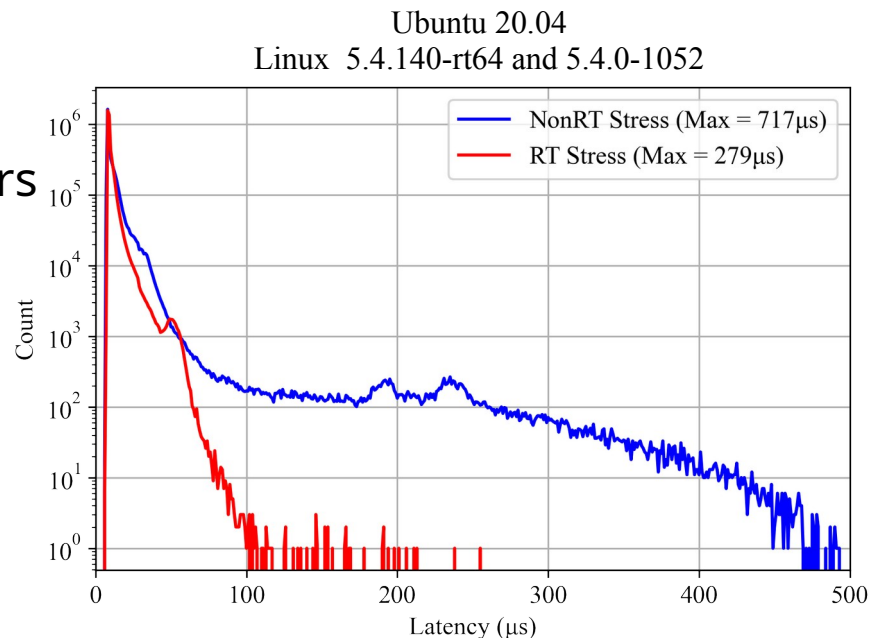
- cyclictest: detects worst-case hardware and scheduling latency
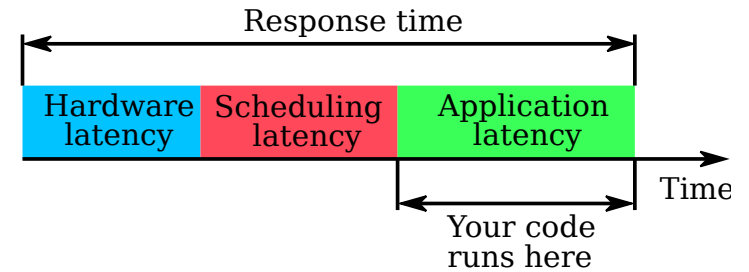
Validation of real-time

- cyclictest: detects worst-case hardware and scheduling latency

- Run cyclictest with CPU stress test for 2 hours

  - Without CPU stress, latency between RT and non-RT kernel is similar

  - CPU stress made the most impact on latency compared to other stress (network, disk, memory)
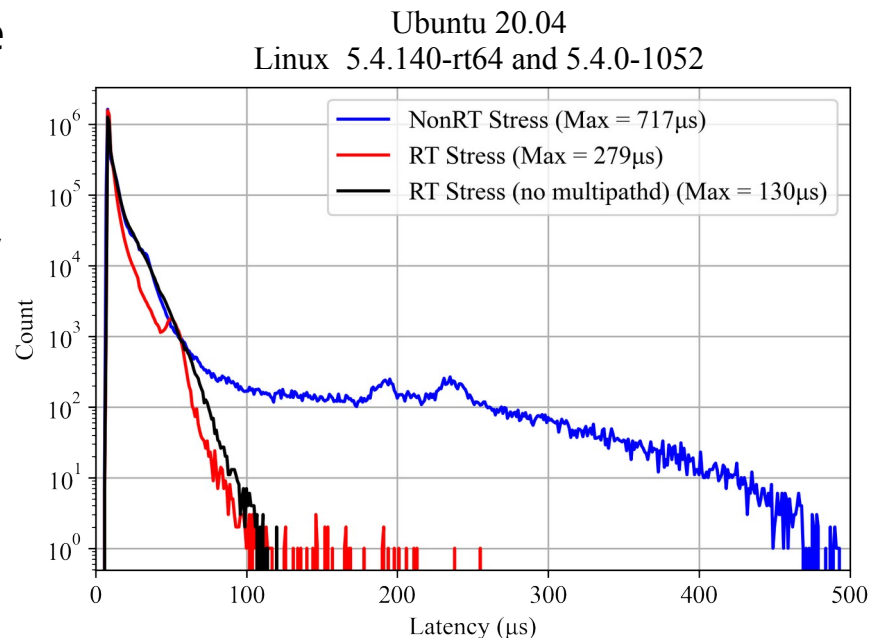
Validation of real-time

- cyclictest: detects worst-case hardware and scheduling latency

- Run cyclictest with CPU stress test for 2 hours
  - Without CPU stress, latency between RT and non-RT kernel is similar
  - CPU stress made the most impact on latency compared to other stress (network, disk, memory)

- Non-RT kernel latency is at least 700 μs
  - Not suitable for 1000 Hz control loop

- RT kernel latency is ~280 μs??

Ubuntu 20.04
Linux 5.4.140-rt64 and 5.4.0-1052

NonRT Stress (Max = 717μs)
RT Stress (Max = 279μs)

CACTUS
DYNAMICS

Validation of real-time

- Identified source of latency with ftrace to be **multipathd**
  - Is a RT process with priority of 99!
    - Interrupts cyclictest and introduces latency
  - A daemon used for configure multiple I/O paths for a single storage device
  - Not useful for the Raspberry Pi
- Latency with multipathd removed: ~130 µs
- multipathd present on all Ubuntu server images?

Ubuntu 20.04
Linux 5.4.140-rt64 and 5.4.0-1052

Legend:
- NonRT Stress (Max = 717µs)
- RT Stress (Max = 279µs)
- RT Stress (no multipathd) (Max = 130µs)

Count vs Latency (µs)

More info on this process: https://shuhaowu.com/blog/2022/02-linux-rt-appdev-part2.html

CACTUS
DYNAMICS

Validation of real-time

- Upgraded from Linux 5.4 → 5.15

- Scheduling latency: 130 µs → 50 µs

- Good baseline for a real-time OS image

Ubuntu 22.04
Linux 5.15.39-rt42-raspi

CACTUS
DYNAMICS
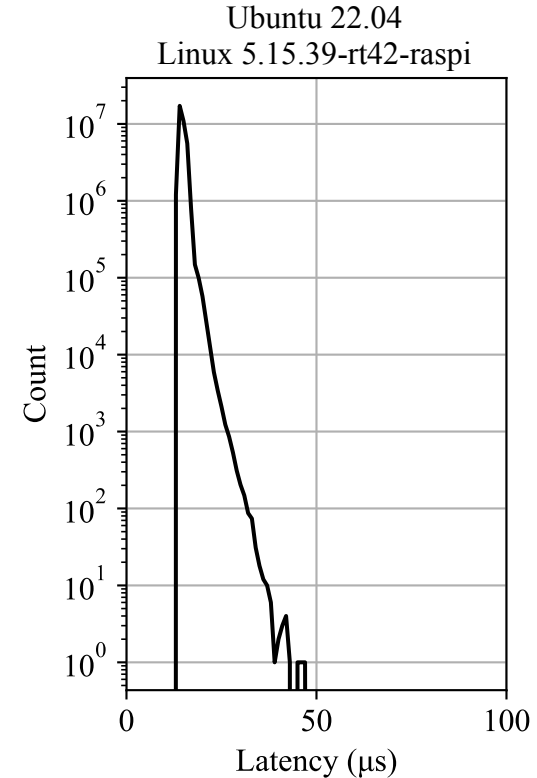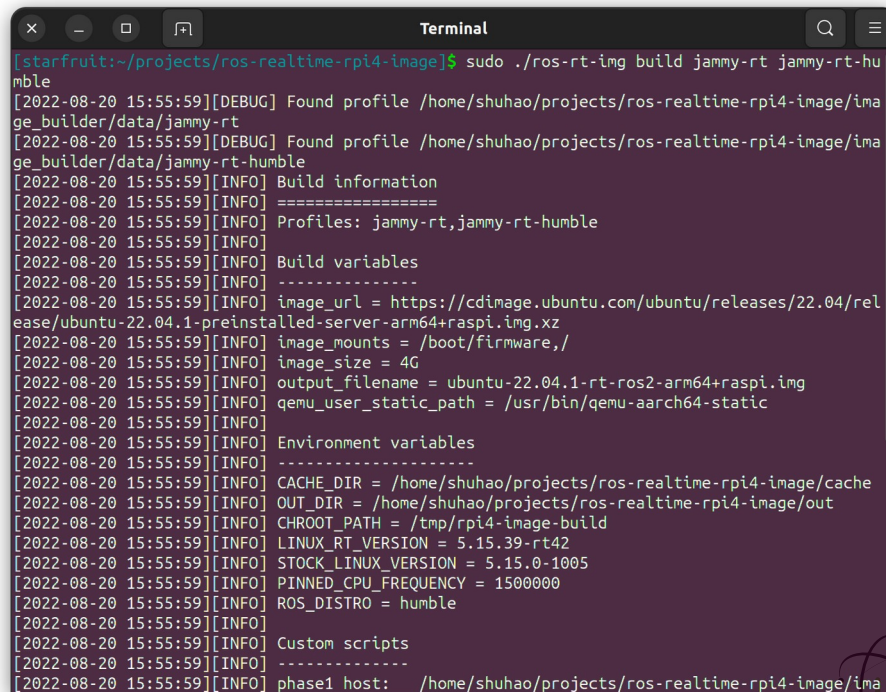
Raspberry Pi + ROS 2 + real-time

- Ready for ROS2
- Ready for 1000 Hz real-time applications
- A nice platform for beginners and experts

CACTUS
DYNAMICS
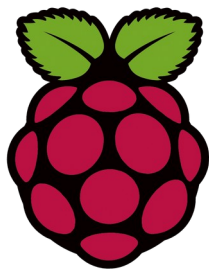
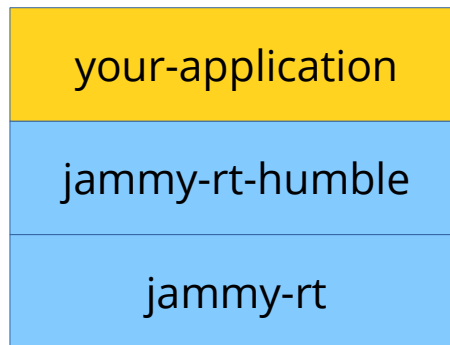# Extendable image builder

- Generic build process with a few files
  - Download vendor image
  - Extract and mount
  - Copy files
  - Run scripts via chroot + qemu
- Image builder is scalable to different single board computers with different base OSes and different apps

13

Advanced image builder features

- Layer your application on top of the ROS 2 + RT image
  - Example: Standard real-time benchmark image
  - Example: Real-time Raspberry Pi CI image
- Cross-compilation
- Extensive documentation

your-application
jammy-rt-humble
jammy-rt

CACTUS
DYNAMICS

Summary

- A downloadable ROS + RT image for the Raspberry Pi
    - Validated real-time performance
    - Useful for getting started
- An extendable image builder for deployment
    - Useful for reproducible builds

- Star us on Github: **https://github.com/ros-realtime/ros-realtime-rpi4-image**
- Special thanks to the ROS real-time working group
    - Andrei Kholodnyi, Carlos San Vicente, Christophe Bedard, Lander Usategui

CACTUS
DYNAMICS