A flexible test automation system for various Embedded Linux usecases

Khiem Nguyen / Engineer
Renesas

@KhiemNguyenT
WHO AM I?

- Name: Khiem Trong. Nguyen (KHIEM Nguyen)
- Company: Renesas Design Vietnam

- Career: 10 years experiences in embedded software development
  - Development and verification for Mobile and In-vehicle software platform
  - Development for test automation solutions of In-vehicle software platform

- Email: khiem.nguyen.xt@renesas.com
Renesas Design Vietnam Co., Ltd. (RVC) was founded in October 2004, as one of the main design centers in Renesas group.

Business line: Design of semiconductor for both hardware and software.
A FLEXIBLE TEST AUTOMATION SYSTEM FOR VARIOUS EMBEDDED LINUX USECASES

OPEN-SOURCE SUMMIT NORTH AMERICA 2018

08/29/2018, VANCOUVER
KHIEM NGUYEN
SENIOR STAFF ENGINEER
RENESAS DESIGN VIETNAM
RENESAS ELECTRONICS CORPORATION
AGENDA

- Motivation Page 00
- Models of test automation system Page 00
- Scale the system at will Page 00
- Automated Testing community Page 00
- Conclusion Page 00
MOTIVATION
“We have as many testers as we have developers. And testers spend all their time testing, and developers spend half their time testing. We’re more of a testing, a quality software organization than we’re a software organization.”

— Bill Gates
“The first rule of any technology used in a business is that automation applied to an efficient operation will magnify the efficiency.

The second is that automation applied to an inefficient operation will magnify the inefficiency.”

— Bill Gates
MOTIVATION (3/3)

- Applications
- Middleware and adaption layer
- Linux kernel
- BSP Drivers
- Hypervisor

SoC(s) and board

Linux kernel
BSP Drivers
SoC and board

Profiling & Benchmark
Multimedia I/O test

kernelci.org
labgrid

usbsdmux
muxpi
acme cape

Fuego

AAT/VAT tools

Orchestration tools
MODEL OF TEST AUTOMATION SYSTEM

Scalability
MODEL OF TEST AUTOMATION SYSTEM OVERVIEW

CIP’s Board@Desk model

The OSADL Test Rack
MODEL OF TEST AUTOMATION SYSTEM
INDIVIDUAL DEVELOPER DEVELOPMENT ENVIRONMENT

- The integrated development environment for each developer.
  - Include a test execution framework (e.g. kernelCI, LAVA) plus a board control service (e.g. LAVA).

**Pros**
- Easy to setup by each developer.
- Suitable for developing the software for a specific hardware.

**Cons**
- May not support multiple hardware targets.
- Cannot share the “working” environment with other developers.
- May not support extended testing scenarios.
The fully-integrated development environment with front-end, middle-end and back-end.

- **Front-end**: interface with developers.
- **Middle-end**: the test automation framework and related plug-ins or add-ons.
- **Back-end**: board control services and related hardware setup.

(*1) Enhance Fuego Test Efficiency by Applying Additional Software & Hardware Solutions

(*2) OSADL Test Rack
MODEL OF TEST AUTOMATION SYSTEM
SHARED DEVELOPMENT ENVIRONMENT WITH BOARD FARM(S) (2/2)

- **Pros**
  - Minimal effort on sharing/maintaining the development environment.
  - Easy to customize the development environment on-demand.
  - 24/7 development environment for different testing scenarios.

- **Cons**
  - Deploy whole system take time.

(*1) Enhance Fuego Test Efficiency by Applying Additional Software & Hardware Solutions
(*2) OSAI Test Rack
SCALE THE SYSTEM AT WILL
SCALE THE SYSTEM AT WILL

Make the system flexible enough for various demands

Fuego integrated test automation model

Fuego integrated system in our view point.

CIP’s Board@Desk model

The OSADL Test Rack
SCALE THE SYSTEM AT WILL – STEPS TO THE HEAVEN (1/4)

1. Start with specific requirements, i.e. short-term versus long-term
2. Fit the requirements into available solutions in the community.
   - Prioritize the portable solutions.
   - Give it a try.
     - To understand the Pros-Cons and the gaps with the defined requirements.
3. Do customization per demands
   - Add the portable software components
   - Develop the lacking pieces

Keep in mind
- Keep the interfaces across portable modules and module itself.
3. Do customization per demands
   - Add the portable software components.
   - Develop the lacking pieces.

Keep in mind
- Do ‘invention’ where it’s necessary.

(*1) Applying Video Test Automation to Automate Multimedia Verification with Embedded Linux
Will be presented in August 30 • 11:00am - 11:40am @ room 110
SCALE THE SYSTEM AT WILL – STEPS TO THE HEAVEN (4/4)

4 Contribute the changes to the community. Give requests for desired features in new releases.

- Receive feedback
- Minimize the maintenance cost
AUTOMATED TESTING COMMUNITY
AUTOMATED TESTING COMMUNITY

▪ Join mailing list for discussing all matters related to the community test automation solution.

▪ Join the first Automated Testing Summit to have discussion across test automation solutions.
  ▪ Being organized by Tim Bird, the Fuego maintainer, and Kevin Hilman, the KernelCI founder.
  ▪ Will be held along with ELC-E 2018 event (October 22 - 24, 2018)
  ▪ The detail is gathered at https://elinux.org/Automated_Testing_Summit.
CONCLUSION
CONCLUSION

- Testing becomes more efficient with right test automation solution.
- Utilize the open-source test automation framework and look into existing examples deployed in open-source software projects.
  - Prioritize the portable software components.
  - Keep the interface for upgrade path.
- Develop the lacking pieces where it’s necessary.

- Join the test automation community.
- Join Automated Testing Summit this October.
APPENDIX

- Some popular test automation frameworks
  - Fuego (http://fuegotest.org)
  - Lava (https://wiki.linaro.org/LAVA)
  - KernelCI (https://kernelci.org)
  - Labgrid (https://github.com/labgrid-project)

- Note about the architecture design (and idea) of test automation system.
  - https://elinux.org/Test_Stack_Layers