1A-04

Perspectives for Architecture of Sustainable Control Engine

Tadashi Ohashi†
iLICT†

1. Introduction

The proposed sustainable control engine (sUCE) is a system's core architecture comprising of main unit (eACE: embedded Architecture Control Engine) whose power supply is able to shut done[1)] and another subunit (sPMD: software Product Management in Device) is sustainable unit whose power source is kept. This computer architecture is able to change the computer system architecture for system upgrade or system downgrade by changing hardware configuration data and software configuration data of current computer's architecture, and by changing hardware functions in FPGA (Field Programmable Gate Allay) data or software products. Hardware software configuration data and hardware and software as object data are down-loaded from Cloud.

2. Problems and Themes on Research

An embedded system must be sustainable for energy conservation and user's satisfaction as well as utilities for multi core electronic devices. These three elements are topics of today's computer systems' architecture. This paper focuses on sustainable system and proposes "sUCE" abbreviated in Introduction. This sUCE has the following prerequisites.

(1) The sUCE must support upgradable or down-gradable hardware and software architecture.
(2) A sustainable engine must be implemented for hardware / software upgrade or downgrade.
(3) Hardware / Software Composition Deployment on web must be applied to meet the sUCE.
(4) Software Composition Table on web, must also be accommodated with the sUCE.

3. Architecture of sUCE

To solve the preceding problems, the sUCE has the following functions to perform the expected architecture.

(1) Software products are operating in multi core ARM processors in FPGA (or SoC).
(2) Partial Power Control Method must be applied to make sUCE be sustainable.
(3) A sustainable Operating System must be applied.
(4) Bridge must be developed between eACE and sPMD within sUCE.

† iLICT (http://www.ilict.jp/)

4. Use case diagram

The Unified Modeling Language (UML) [2, 3)] is not only applied to software development but also to hardware development. Especially sUCE is performed almost software products on web as well as in FPGAs. A use case diagram is described as follows.

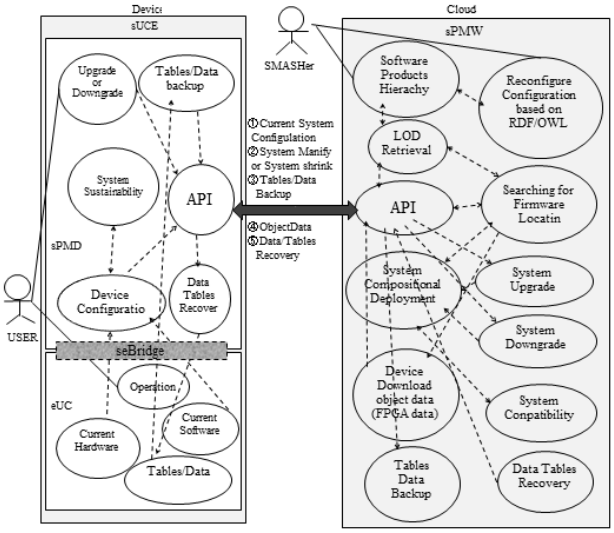

Fig.1 Diagram of Use Case for sUCE

The Embedded Architecture with Linked Data (eALD) [4)] has been proposed. An eALD has two parts. Device on the left hand, Cloud on the other hand are schemed in Fig.1. The device includes two units over seBridge. A user decides system upgrade or downgrade. This decision activates a device to get current Hardware / Software (H/S) Information and ask for upgrade and downgrade operations to the cloud based on current H/S Information. The cloud (sPMW: Software Product Management on the Web) has the compositional deployment of system upgrade and downgrade by searching for upper version or lower version's S/H Information based on RDF[5)] (Resource Data Framework) / OWL[6)] (Ontology Web Language) for hardware data (ex. FPGA data) or software products.

5. System Architecture

The Device consists of sPMD and eACE, whose processors are performed in FPGAs. sUCE's power supply is permanent power, on the other hand, eACE's power supply is shutdown when upgrade or downgrade are in progress. Two ARMs (multicore)

are implemented. DDR4s are interwoven over the same seBridge. There are two compatibility logics by web and in device. Configuration function also equipped. Embedded ICEs are implemented in both eACE and sUCE. sACE has functions of downgrading or upgrading of eACE. Upgrading or downloading data are made by sPMW. Through sUCE, eACE is upgraded or downgraded for the user's request.

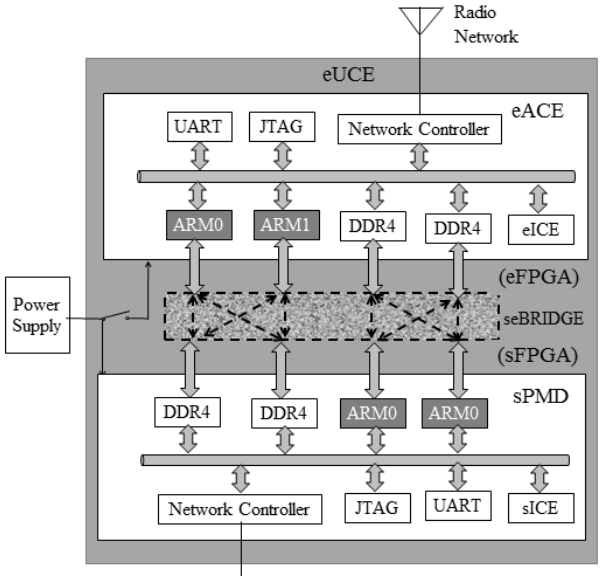

Fig.2 Computing Architecture

6. Device Performances with Cloud (sPMW)

Fig.3 shows software products, two major units, one of them is software Product Management on web (sPMW) as cloud. Another is software product management in device (sPMD) on the left side, uClinux is implemented, and on the other hand, Linux

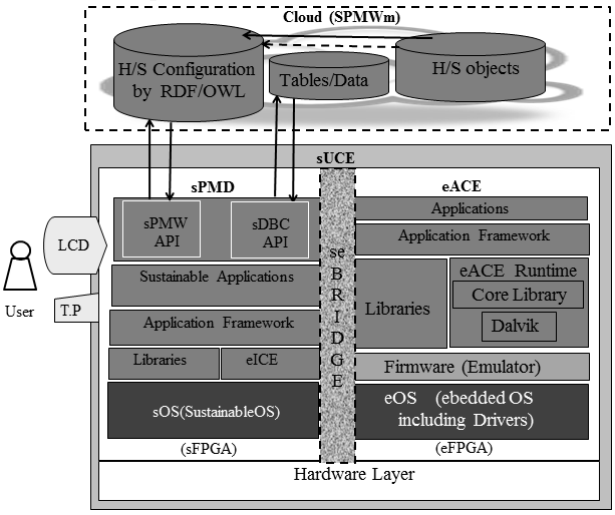

Fig.3Software Architecture

is implemented in eACE vice versa. uClinux[7] is useful for sustainable processor such as sUCE because MMU (Memory Management Unit) is unnecessary.

A given device hardware configuration and software configuration are uploaded to sPMW (Cloud) and system upgrade and downgrade are performed in sPMW. A given device hardware (FPGA data) and software products are compatible logically with upgrade or downgrade object data as beta images are deployed. Real configuration data is preliminary performed on sPMW (Cloud) and downloaded to device.

7. Conclusion

This research concludes the following 3 items.

(1)Architecture of Sustainable Control Engine (sUCE) is considered to be effective for changing or swapping hardware or software products in device.

(2) seBrige is one of very important functions for sustainable computing architecture.

(3) System's H/S upgrade or downgrade is deployed on sPMW (Cloud) and downloaded to a device.

8. Future themes

To brush up this research, further following research must be done as quickly as possible.

(1) seBridge must be researched and experimental survey must be done to improve efficacy.

(2) How to match H/S RDF/OWL Information with the proceeding architecture of conventional SHIM[8] (Software and hardware Information Management)

(3) uCLinux must be surveyed whether it supports Java Runtime Environment or not. At present, this environment was not fully discussed here.

(4) sPMS (Cloud) must be researched more deeply in order to upgrade and downgrade operation.

References

1) James G. Harris : Development of Sustainable Power for Electrical Resources – SuPER System https://courseware.ee.calpoly.edu/~jharris/research/super_project/super_grad_sem.ppt

2) Tim Schattkowisky, Achim Rettberg: UML for FPGA http://is.uni-paderborn.de/uploads/tx_sibibtex/Schattkowsky_UML_for_FPGA_Synthesis.pdf

3) A new Logic Circuit Design Methodology with UML. http://www.sato-zaidan.or.jp/h21/pic-h21/090605-r.pdf

4) Tadashi Ohashi: A Gradable Electric Equipment based on Open Source Platform with Linked Open Data: Forum for Information Technology 2015. C-004 (2015)

5) RDF: http://www.w3.org/RDF/

6) OWL: http://www.w3.org/2001/sw/wiki/OWL

7) Brian Estrada, Patrick Mariano: DEVELOPMENT OF UCLINUX PLATFORM FOR CAL POLY SUPER PROJECT

8)http://www.multicore-association.org/workgroup/shim.php