(IJNCAA)
ISSN 2220-9085(ONLINE)
ISSN 2412-3587(PRINT)

INTERNATIONAL JOURNAL OF NEW COMPUTER ARCHITECTURES AND THEIR APPLICATIONS

Volume 9, Issue 4, 2019

SDIWC
www.sdiwc.net

International Journal of
NEW COMPUTER ARCHITECTURES AND THEIR APPLICATIONS

Overview

The SDIWC International Journal of New Computer Architectures and Their Applications (IJNCAA) is a refereed online journal designed to address the following topics: new computer architectures, digital resources, and mobile devices, including cell phones. In our opinion, cell phones in their current state are really computers, and the gap between these devices and the capabilities of the computers will soon disappear. Original unpublished manuscripts are solicited in the areas such as computer architectures, parallel and distributed systems, microprocessors and microsystems, storage management, communications management, reliability, and VLSI.

One of the most important aims of this journal is to increase the usage and impact of knowledge as well as increasing the visibility and ease of use of scientific materials, IJNCAA does NOT CHARGE authors for any publication fee for online publishing of their materials in the journal and does NOT CHARGE readers or their institutions for accessing the published materials.

Publisher

The Society of Digital Information and Wireless Communications
20/F, Tower 5, China Hong Kong City, 33 Canton Road, Tsim Sha Tsui, Kowloon, Hong Kong

Further Information

Website: http://sdiwc.net/ijncaa, Email: ijncaa@sdiwc.net,
Tel.: (202)-657-4603 - Inside USA; 001(202)-657-4603 - Outside USA.

The issue date is Dec. 2019.

IJNCAA

ISSN 2220-9085 (Online)
ISSN 2412-3587 (Print)

Volume 9, Issue No. 4

2019

CONTENTS

ORIGINAL ARTICLES

International Journal of
NEW COMPUTER ARCHITECTURES AND THEIR APPLICATIONS

The *International Journal of New Computer Architectures and Their Applications* aims to provide a forum for scientists, engineers, and practitioners to present their latest research results, ideas, developments and applications in the field of computer architectures, information technology, and mobile technologies. The IJNCAA is published four times a year and accepts three types of papers as follows:

1. **Research papers**: that are presenting and discussing the latest, and the most profound research results in the scope of IJNCAA. Papers should describe new contributions in the scope of IJNCAA and support claims of novelty with citations to the relevant literature.
2. **Technical papers**: that are establishing meaningful forum between practitioners and researchers with useful solutions in various fields of digital security and forensics. It includes all kinds of practical applications, which covers principles, projects, missions, techniques, tools, methods, processes etc.
3. **Review papers**: that are critically analyzing past and current research trends in the field.

Manuscripts submitted to IJNCAA **should not be previously published or be under review** by any other publication. Plagiarism is a serious academic offense and will not be tolerated in any sort! Any case of plagiarism would lead to life-time abundance of all authors for publishing in any of our journals or conferences.

Original unpublished manuscripts are solicited in the following areas including but not limited to:

- Computer Architectures
- Parallel and Distributed Systems
- Storage Management
- Microprocessors and Microsystems
- Communications Management
- Reliability
- VLSI

International Journal of New Computer Architectures and their Applications (IJNCAA) 9(4): 77-82
The Society of Digital Information and Wireless Communications, 2019 ISSN 2220-9085 (Online); ISSN 2412-3587 (Print)

A Study on Model-Based Development of Embedded System using Open Source Computational Environment

Kengo MASUMI[1] Shinji TSUZUKI[2]
Masashi SUGIMOTO[2*] Hitoshi YOSHIMURA[3]
[1]Department of Electrical and Electronic Engineering,
Faculty of Engineering, Ehime University
3 Bunkyo, Matsuyama C., Ehime Pref. 790-8577 Japan
[2]Department of Electrical and Electronic Engineering,
Graduate School of Science and Engineering, Ehime University
3 Bunkyo, Matsuyama C., Ehime Pref. 790-8577 Japan
E-mail: sugimoto.masashi.du@ehime-u.ac.jp
($*$: *Corresponding author*)
[3]Division of Computer Science and Engineering, Department of Engineering for Innovation
National Institute of Technology, Tomakomai College
443 Nishikioka, Tomakomai C., Hokkaido 059-1275 Japan

ABSTRACT

In recently, the technique of the model-based development has come to be used more widely on the site of the embedded system development. In the present study, we propose the technique for model-based development of the embedded system using latest version of Scilab/Xcos. As former study, we have studied model-based development using Scilab 4/Scicos and LEGO Mindstorms NXT. However, these systems are obsolating and out of support. From this issue, as an instance of applying we tried the stabilizing control of the two-wheeled inverted pendulum, using Gyro Boy as known as LEGO Mindstorms EV3. The control procedure was generated from the designed feedback control system automatically of block diagram for stabilization and applied to TOPPERS/EV3RT of LEGO Mindstorms EV3. As a result of the experiment, we obtained stable execution results as simulation and actual EV3 Gyro Boy. Therefore, the utility of the proposed technique was confirmed.

KEYWORDS

Model-based Development, LEGO Mindstorms EV3, Scilab/Xcos, TOPPERS/EV3RT, Linear Quadratic Regulator

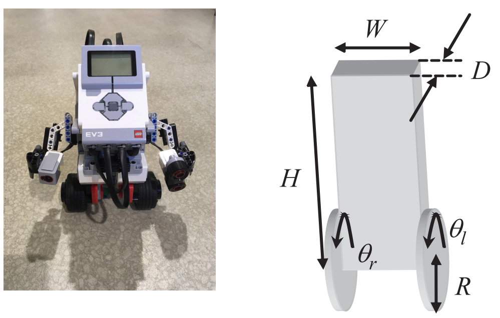
1 INTRODUCTION

In case of the development for embedded systems, non-object oriented languages, such as C language or assembler language will be used in situation of development. However, in recent, object-oriented development using UML has also been carried out, moreover, even in the hardware design phase has been used. UML is a modeling language that has focused on analyzing and designing for software development. Therefore, it is necessary to pay sufficient attention to whether the contents of the target block to analyze or designing software are provided by region or hardware region[1]. From this viewpoint, it is pointed out that UML cannot be applied in several aspects of the development of an embedded system[2]. For example, it is hard to description for a control system of a function, such as a describe for concurrency control, a control timing, and a feedback loop for continuous system. To solve these problems, a research have been provided a method that applying the RT (“Robot Technology” or “Robotics Technologies”) middleware technique. In this method, an RT middleware had been capturing Scilab/Scicos as a component. From this technique, this method generates some tasks from the Scicos block diagram, indirectly.

The authors had been studied the free model-based development environment for the embedded systems[3, 4, 5, 6]. However, the computational environments and hardware models are grown as years. From the viewpoint, the environment of the model-based development system must be enlarge and apply to the current issue. In addition, the RTOS (Real-time operating system) have much functions and approaching real-time processing.

In this study, the designation and simulation of control systems of the embedded systems have been focused on. Moreover, as shown in fig. 1, "Gyro Boy" has been applied as a target robot of control. We have previously reported that focused on development using TOPPERS/OSEK and TOPPERS/JSP[4, 5, 6]. In this paper, we will try to extend the model-based development system support to TOPPERS/EV3RT. From this standpoint, we will re-consider the block diagram of a system that will be connected to Auto-code generator.

This paper is organized as follows: In section 2, outline of model-based development, and previous study, will be stated. In section 3, a verification experiment configuration will be described. In section 4, the summary of this work is concluded.



(a) The Structure (b) 3D Model

Figure 1. The Structure of "EV3 Gyro Boy."

2 SYSTEM CONFIGURATION AND FLOW OF PROCEDURE

In this study, we aim to structure a system that can be directly generated a task or control API for TOPPERS/EV3RT from Xcos block diagrams. For this solution, we will be able to complement a missing function of UML. Further, Scilab/Xcos has simulation function, so we can obtain simulation results, directly. Therefore, the proposed system has an environment that provides portability and efficiency development.

In previous research by Yamamoto[3], simulation is carried out by MATLAB/Simulink as a numerical calculation system. That is a highly reliable system with many use records, however, it is very expensive. On the other hand, Scilab/Xcos (former name was Scilab/Scicos) used in this study is an open source numerical calculation system that has been developing by Scilab Enterprises Inc., that is open source software. Moreover, an expansion package is also distributed free of charge. From this standpoint, Scilab/Xcos will be applied in this study.

Moreover, nowadays, this technology will be needed what the demands on the robotics, emmbedded systems, and IoT industry are expected to increase. In recently, TOPPERS project has been developed or released some platforms. However, these works are conform μITRON or not is depends on each work. Moreover, some works are conform VDX specify. If target processor-oriented or target task-oriented application will be developed, it is necessary to support for development that roots in RTOS beyond the differences of specifications such as μITRON or VDX specify. As this result, our proposed system will be worthly we considered.

In our former study [4, 5, 6], we had been considered functions (fig. 2) as below:

(1) Simulator

(2) Auto-code Generator

(3) Builder / Uploader

However, we cannot apply the system that had been development, to TOPPERS/EV3RT, directly. Therefore, we focused on generated files by (2) auto-code generator. In detail, we considered the necessary files to generate

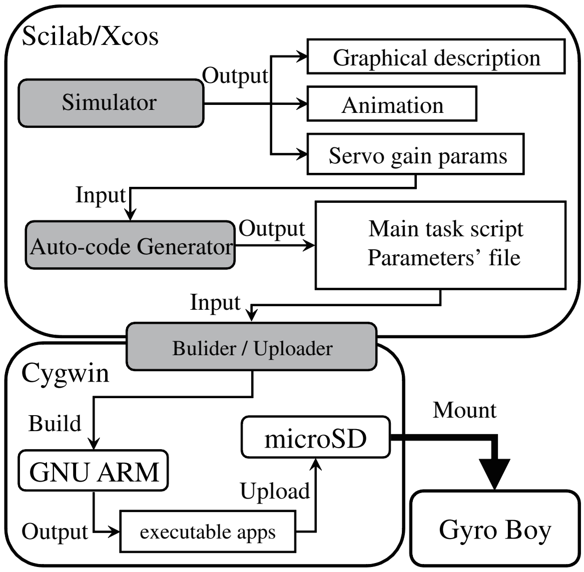


Figure 2. Outline for the Proposal of Model-based Development System.

a TOPPERS/EV3RT executable. In former study [4, 5, 6], we designed an C file for TOPPERS/OSEK or TOPPERS/JSP. However, task notation and variable definition are different as TOPPRES/OSEK or TOPPERS/JST. As mentioned above, in the future works, we have to re-consider these problems and rebuild the systems. Especially, in this study, as the first step for improve former study, we have tried to apply the parameters of feedback gain to Gyro Boy for stabilizing control.

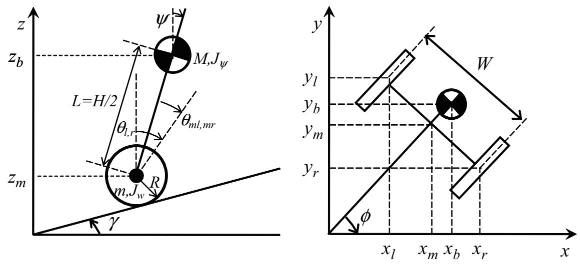
3 VERIFICATION EXPERIMENT

3.1 The Experimental Outline

In this verification experiment, the posture of a two-wheeled inverted pendulum “Gyro Boy’, as an application that was stabilized using a computer simulation as a verification experiment. Moreover, an actual robot “Gyro Boy”, has been also applied postural control. In this experiment, the simulational function of Xcos will be used between designing a control parameter of Gyro Boy based on LQR controller.

3.2 Configuration of Simulation 1 – Introduce Gyro Boy Model

As shown in figure 3, “Gyro Boy” can be described as a two-wheeled inverted pendulum model. The coordinate system used in section 3.3 is described in fig. 3. Moreover, in figure 3, ψ denotes the body pitch angle and $\theta_{l,r}$ denotes the wheel angle (l and r indicate *left* and *right*, respectively. Further, $\theta = 1/2 \cdot (\theta_l + \theta_r)$.), and $\theta_{ml,mr}$ denotes the DC motor angle (l and r indicate *left* and *right*, respectively). The Gyro Boy’s physical parameters are listed in table 1.



(a) Side Model View (b) Plane Model View

Figure 3. Model of Side and Plane View of Gyro Boy.

3.3 Configuration of Simulation 2 – for Gyro Boy Modeling

Gyro Boy’s motion equations can be derived, shown in fig. 3. If the direction of the model is the x-axis positive direction at $t = 0$, the equations of motion for each coordinate can be given as follows [4, 5, 6]:

$$\left[(2m + M) R^2 + 2J_w + 2n^2 J_m\right] \ddot{\theta} + \left(MLR - 2n^2 J_m\right) \ddot{\psi} - Rg (M + 2m) \sin\gamma = F_\theta \quad (1)$$

$$\left(MLR - 2n^2 J_m\right) \ddot{\theta} + \left(ML^2 + J_\psi + 2n^2 J_m\right) \ddot{\psi} - MgL\psi = F_\psi \quad (2)$$

$$\left[\frac{1}{2} m W^2 + J_\phi + \frac{W^2}{2R^2}\left(J_w + n^2 J_m\right)\right] \ddot{\phi} = F_\phi \quad (3)$$

Table 1. Physical Parameters of Gyro Boy

Symbol	Value	Unit	Physical property
g	9.81	[m/s^2]	Gravity acceleration
m	0.024	[kg]	Wheel weight
R	0.027	[m]	Wheel radius
J_w	$\frac{mR^2}{2}$	[kgm^2]	Wheel inertia moment
M	0.8	[kg]	Body weight
W	0.105	[m]	Body width
D	0.1	[m]	Body depth
H	0.21	[m]	Body height
L	$\frac{H}{2}$	[m]	Distance of center of mass from wheel axle
J_ψ	$\frac{ML^2}{3}$	[kgm^2]	Body pitch inertia moment
J_ϕ	$\frac{M(W^2+D^2)}{12}$	[kgm^2]	Body yaw inertia moment
J_m	1×10^{-5}	[kgm^2]	DC motor inertia moment
R_m	6.69	[Ω]	DC motor resistance [4]
K_b	0.468	[V·s/rad.]	DC motor back EMF constant
K_t	0.317	[N·m/A]	DC motor torque constant [4, 5, 6]
n	1	[1]	Gear ratio [4, 5, 6]
f_m	0.0022	[1]	Friction coefficient between body and DC motor [4, 5, 6]
f_W	0	[1]	Friction coefficient between wheel and floor [4, 5, 6]

Here, $\mathbf{x}_1$ and $\mathbf{x}_2$ represent state variables. In addition, $\mathbf{u}$ denotes input:

$$\mathbf{x}_1 = \begin{bmatrix}\theta & \psi & \dot{\theta} & \dot{\psi}\end{bmatrix}^\top \quad (4)$$

$$\mathbf{x}_2 = \begin{bmatrix}\phi & \dot{\phi}\end{bmatrix}^\top \quad (5)$$

$$\mathbf{u} = \begin{bmatrix}v_l & v_r\end{bmatrix}^\top \quad (6)$$

From above equations, state equations of Gyro Boy can be derived using eqs. (1), (2), and (3).

$$\frac{d}{dt}\mathbf{x}_1 = \mathbf{A}_1\mathbf{x}_1 + \mathbf{B}_1\mathbf{u} + \mathbf{S} \quad (7)$$

$$\frac{d}{dt}\mathbf{x}_2 = \mathbf{A}_2\mathbf{x}_2 + \mathbf{B}_2\mathbf{u} \quad (8)$$

In this verification experiment, only a state variable $\mathbf{x}_1$ will be used. Because $\mathbf{x}_1$ contains the body pitch angles as variables ψ and $\dot{\psi}$, which are important for self-balancing. That's why the plane motion ($\gamma_0 = 0, \mathbf{S} = \mathbf{0}$) will not be considered in this experiment:

$$\frac{d}{dt}\mathbf{x}_1 = \mathbf{A}_1\mathbf{x}_1 + \mathbf{B}_1\mathbf{u} \quad (9)$$

3.4 Configuration of Simulation 3 – Applying the LQR as A Controller

In this study, Linear-Quadratic Regulator (as known as LQR) is applied. The feedback gain $\mathbf{k}_f$ so as to minimize the quadratic cost function J will be calculated by this LQR given as eq. (10).

$$J = \int_0^\infty \left[\mathbf{x}^\top(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^\top(t)\mathbf{R}\mathbf{u}(t)\right] dt \quad (10)$$

In this verification experiment, matrices $\mathbf{Q}$ and $\mathbf{R}$ are as following:

$$\mathbf{Q} = \begin{bmatrix}1 & 0 & 0 & 0 & 0\\ 0 & 6\times10^5 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 4\times10^2\end{bmatrix} \quad (11)$$

$$\mathbf{R} = 1\times10^3 \cdot \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \quad (12)$$

Mentioned above equation, $\mathbf{k}_f$, is a gain of optimal feedback, will be obtained by minimizing J. However, in the equation, below equations that includes Riccati equation will be solved:

$$\mathbf{A}^\top\mathbf{S} + \mathbf{S}\mathbf{A} - \mathbf{S}\mathbf{A}\mathbf{R}^{-1}(\mathbf{B}^\top\mathbf{S}) + \mathbf{Q} = 0 \quad (13)$$

$$\mathbf{x}^\top = \begin{bmatrix}\mathbf{A}_1 & \mathbf{0}\\ \mathbf{C}_1(1,:) & 0\end{bmatrix} \quad (14)$$

In this equation, $\mathbf{C}_1(1,:)$ denotes the 1st row of the matrix $\mathbf{C}_1$. From these results, the feedback gain $\mathbf{k}_f$ of the state-feedback stabilizer will be applied. However, in the verification experiment, the plane movement of the two-wheeled inverted pendulum will be not considered. Hence, $\phi = 0, \theta_{ml} = \theta_{mr}$, and $\mathbf{u} = u, \mathbf{d}(t) = d(t)$ were considered.

International Journal of New Computer Architectures and their Applications (IJNCAA) 9(4): 77-82
The Society of Digital Information and Wireless Communications, 2019 ISSN 2220-9085 (Online); ISSN 2412-3587 (Print)

Table 2. Parameters for the Simulation of the Proposed Method

Symbol	Value	Unit	Physical property
ψ_0	5.6	[deg.]	Initial value of body pitch angle
γ_0	0.0	[rad.]	Slope angle of movement direction
t_s	0.05	[sec.]	Sampling rate

3.5 Simulation Conditions

In this verification simulation, the action signal with an unknown disturbance signal $\mathbf{d}(t)$ (fig. 4) will be mixed. In figure 4, we can describe the block diagram as fig. 5.

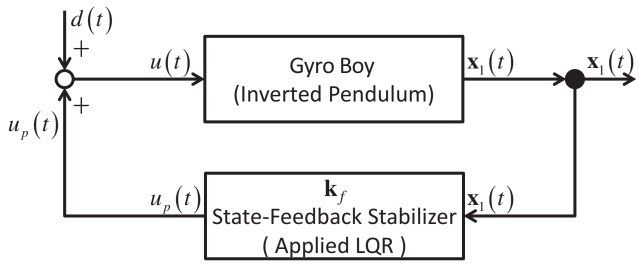


Figure 4. Control Input Obtained by Mixing the Action and Disturbance Inputs.

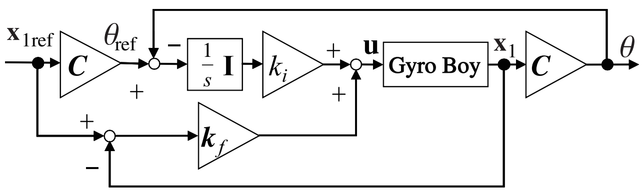


Figure 5. Gyro Boy Servo Controller Block Diagram.

In this verification experiment, Gyro Boy takes self-balancing. In addition, the properties of disturbance signal that we will provide as input and other conditions of a simulation are listed in table 2.

3.6 Results of Experiment and Discussion

In this experiment, firstly, we have done the simulation using the servo controller that had been applied to the Gyro Boy model. In addition, the Gyro Boy model performs stationary balancing while 60 [sec.]. Figure 6 shows a simulation result of stationary balancing at

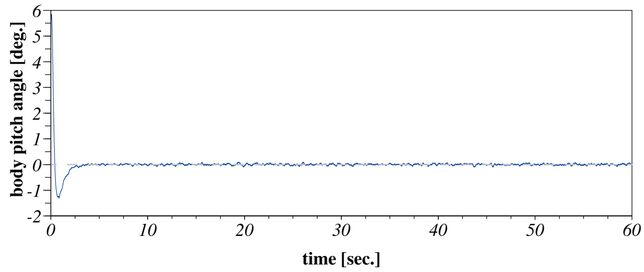


Figure 6. Control Response of Body Pitch Angle ψ. (Simulational Result)

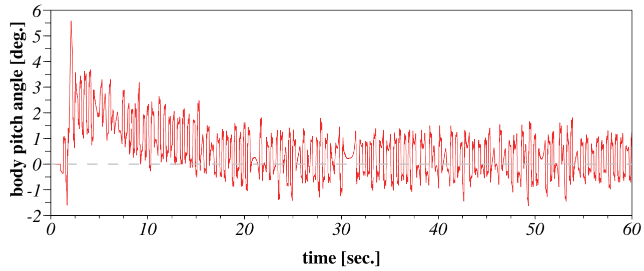


Figure 7. Control Response of Body Pitch Angle ψ. (Actual Robot operated by TOPPERS/EV3RT)

initial value of body pitch angle equals to 5.6 [deg.]. Moreover, figure 7 show the experimental results using "*actual*" Gyro Boy. In this experiment, the Gyro Boy is executed by TOPPERS/EV3RT executable that had been generated by simulated feedback gains. In addition, the Gyro Boy performs that keeps stationary balancing for 60 [sec.].

Now we are trying to focus on simulation result. According to figure 6, also, compensation result obtained using the proposed method (shown as the solid line) approach and oscillate to near zero, with time. Therefore, it can be said that the wheel is moving while trying to decrease the body pitch angle $\psi(t)$.

Next, we are trying to focus on experimental result as shown in fig 7. Here, focusing the control response point is shown after $t = 1$ [sec.]. Gyro Boy does not run until 1 [sec.] passes because of gyro calibration. Therefore, we will only argue and focus on the part of the graph pertaining to state prediction part shown in T. According to figure 7, compensation result obtained using the proposed method (shown as the solid line) approach and oscillate to near zero, with time. Therefore, it can be said that the wheel is moving while trying

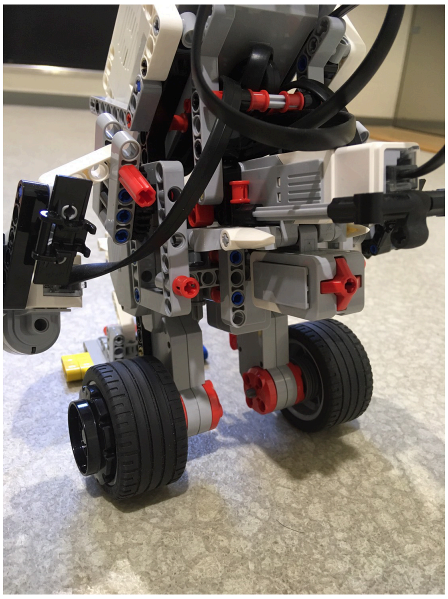


Figure 8. The scene of Stationary balancing control.

to decrease the body pitch angle $\psi(t)$.

4 CONCLUSION

In this paper, the designation and simulation of control systems of the embedded systems have been focused on. We have previously reported that focused on development using TOPPERS/OSEK and TOPPERS/JSP based on LEGO Mindstorms NXTway-GS. In this paper, we have been extending the model-based development system support to TOPPERS/EV3RT. As a result, the structure has been building a system that can be directly generated feedback gains for TOPPERS/EV3RT from Xcos block diagrams. Further, Scilab/Xcos has simulation function, so we can obtain simulation results, directly.

From each verification experiment, it can be concluded that the each result will be retaining stable state, continuously. Accordingly, the proposed method will have availability that is provided TOPPERS/EV3RT executable that includes simulation results and physical parameters can be controlled the actual EV3 Gyro Boy, stationary balancing as similar as simulational result. As a conclusion, as similar as former works that applied by TOPPERS/OSEK, will be concluded that abailavilyty for control the actual inverted pendulum as former result of the studies.

REFERENCES

[1] T. Bahill and J. Daniels, “Using objected-oriented and UML tools for hardware design: a case study,” Systems Engineering, Vol. 6 No. 3, pp.28-48, 2003.

[2] N. Ando, “Introduction to RT-middleware for beginners : OpenRTM-aist-1.0 and how to use it,” Journal of the Robotics Society of Japan, Vol.28 No. 5, pp.550-555, 2010.

[3] Y. Yamamoto, “NXTway-GS model-based design –control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT–,” Cybernet Systems Co., Ltd., 2009.

[4] M. Sugimoto, H. Yoshimura, T. Abe, and I. Ohmura, “A study on model-based development of embedded system using Scilab/Scicos,” In proc. of the Japan Society for Precision Engineering 2010 Spring Meeting, Saitama, D82, pp.343-344, 2010.

[5] M. Sugimoto, H. Yoshimura, T. Abe, and I. Ohmura, “A study on model-based development of embedded system using Scilab/Scicos : development of auto-code generator,” In proc. of the JSME annual Conference on Robotics and Mechatronics 2010 (Robomec2010), 2A2-C10, pp1-4, 2010.

[6] M. SUGIMOTO, T. SHIRAKAWA, and H. YOSHIMURA, “A Study on Model-Based Development of Embedded System using Scilab/Scicos with TOPPERS/JSP,” The 22nd Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES2018), pp.1-5, 2018.