

# Development of User Interface with Single Switch Scanning for Robot Arm to Help Disabled People Using RT-Middleware

Yujin Wakita, Natsuki Yamanobe, Kazuyuki Nagata, Noriaki Ando, Mathias Clerc  
Task Intelligence Research Group  
Intelligent Systems Research Institute  
National Institute of Advanced Industrial Science and Technology (AIST)  
Tsukuba-shi, Ibaraki-ken, Japan  
wakita.y@aist.go.jp

**Abstract**— We are developing a manipulator system in order to support disabled people with less muscle strength such as muscular dystrophy patients. Such a manipulator should have an easy interface for the users to control it. In this paper, we report on the construction of the user interface of the system using RT-Middleware. RT-Middleware is an open software platform for robot systems. Therefore other interface components or robot components which are adapted to other symptoms can be replaced with the interface without any change of the contents. A single switch is introduced as the input device for the manual control of the robot arm in our interface. The scanning menu panel is designed to perform various actions of the robot arm with the single switch. A manipulator simulation system was constructed to evaluate the input performance. Two muscular dystrophy patients tried of our interface to control the robot simulator and made comments.

**Keywords**—Service Robot, Support Manipulator, Single Switch, Muscle Dystrophy, RT-Middleware

## I. INTRODUCTION

Service robots which can support disabled people in their daily life are expected to appear in the real market. The development of robotic arms which can support disabled people with muscular dystrophy, spinal cord injuries, ALS and cerebral paralysis are under development [1, 2]. The manipulator should be portable for use on a bed or wheelchair and be easy for the user to control. The symptoms of the patients are varied and the input methods to control the robot should be adapted to them individually.

For example, the iARM (Assistive Robotic Manipulator) [1] was developed to be attached to the side of wheel chairs or bed in order to perform its user's task. The iARM includes several user-interfaces: keypad, joystick and single switch. The patients with muscular dystrophy can operate the manipulator with these interfaces according to their conditions. However the information display device of the iARM is only a 7x5 matrix type indicator which is placed on the robot arm shoulder. Before using the robot arm, the user must know all the presented characters shown on the indicator and also the state transitions for the control mode of the manipulator.

We are developing a portable manipulator system for patients with less muscle power such as muscular dystrophy patients. The authors are developing user interfaces for the system. There are two types of interface. One is a joy-stick, the other is a single switch with a scanning menu panel.

In this paper, we report on the development of the single switch interface with RT-Middleware. The conditions of patients, whose behavior is restricted, are varied. Therefore the operational input for the robot should have many types and grades. These multiple types of input method can be adopted easily through RT-Middleware because RT-Middleware is a common software platform for robot systems.

In the following section, we discuss RT-Middleware and the OpenRTM-aist implementation of it. In the third section, we introduce the interface system constructed with RT-Middleware. In the fourth section, we describe the operation and evaluation by the patients for the interface and finally conclude this paper.

## II. RT-MIDDLEWARE

RT-Middleware is developed by AIST as an open robot software platform. "RT" means "Robot Technology", which is applied not only to the industrial field but also to the non industrial field such as human daily life support systems. RT-Middleware is a software platform for RT systems.

RT-Middleware aims at establishing a common platform based on distributed object technology that supports the construction of various networked robotic systems by the integration of various network enabled robotic elements called RT-Components. If the communication protocol between RT-Components is unified, any other RT-Components can be replaced with them. Modularization of RT elements and RT-Middleware have been studied and developed at AIST, which promotes the application of RT in various fields. RT-Middleware may raise the efficiency of robotic study and development, expand the RT target field and make a new robot market [3].

A. RT-Component Architecture

An RT-Component is the basic functional unit of RT-Middleware based systems. Fig. 1 shows the architecture block diagram of an RT-Component.

For an platform independencies, CORBA (Common Object Request Broker Architecture) is adopted as the distributed object middleware, is used for modeling RT-Components. An RT-Component consists of the following objects and interfaces.

- Component object
- Activity
- InPort as input port object
- OutPort as output port object
- ServicePort as service interface
- Configuration interfaces

The general distributed object model can be described as some interfaces that contain operations with parameters and a return value. On the other hand, the RT-Component model has a component object as the main body, activity as the main process unit and input ports (InPort) and output ports (OutPort) as data stream ports.

B. OpenRTM-aist

“OpenRTM-aist” is developed and distributed by AIST as an implementation of the RT-Middleware interface specification and RT-Component object model [4]. “OpenRTM-aist” consists of an RT-Component development frame work, manager and a set of tools.

Because RT-Middleware aims to improve the reuse of software, our single switch interface component and robotic simulator component can be replaced with the joystick component and real manipulator control component respectively.

Furthermore, when many types of interfaces are developed to correspond to the patient’s symptoms, these new components can be connected to the original components without any change in them and used, if the communication protocol of the components is the same.

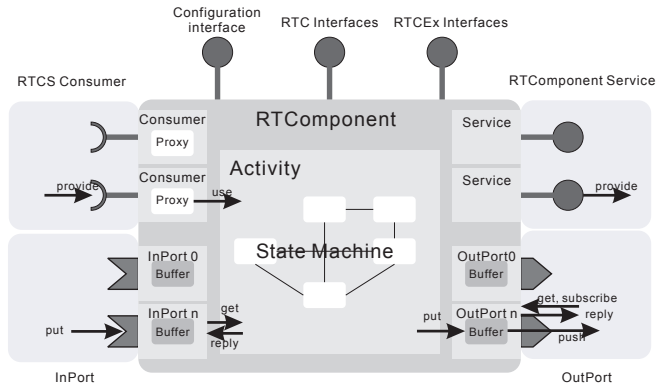


Figure 1. Architecture of RT-Component. RT-Component has a component object, command interfaces, an activity, InPorts, OutPorts and service ports

A. An Example of Single Switch interface

Single switch interfaces are used by disabled people to control computers. There are several commercial software packages [5, 6] for the single switch. For example, “Den-no-shin” is one application. In den-no-shin, the computer shows a panel of buttons on the console. The buttons’ color changes in order (scanning) and the user can select the high lighted button to push the single switch. This software enables the disabled people to input literals, to speak the input sentences, to start Windows applications, to control the mouse cursor with the cross cursor on the console and to click the mouse cursor.

However there are few single switch interface systems to control a robot. Control of a robotic wheelchair by single switch scanning has been reported in [7]. But the user can only give commands to change the direction of the wheelchair. We developed a single switch scanning interface component to control a robot arm.

B. An Outline of the System

Interfaces for disabled people have many variations because their symptoms are varied and the input methods available to them are restricted. Even if the interfaces are varied, the communication protocol of the interface and the robot components can be decided in advance. Under the protocol, many kinds of interfaces and robot components can be connected uniformly as RT-Components.

The overview of the user interface and the robot simulator on the computer console with the single switch is shown in Fig. 2. The user interface consists of a single switch and scanning menu panel interface. A manipulator simulator controlled by the interface is shown on the left of the panel interface.

The panel interface and the robot simulator are created from the interface component and the simulator component respectively.

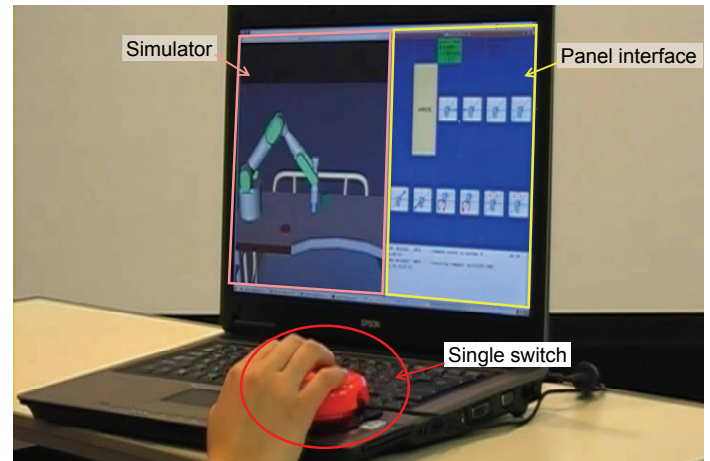


Figure 2. Overview of interface and robot components with a single switch

### C. Robot Component

The communication protocol of the interface and robot components depends on the input/output data of the robot component. The robot component should have the following functions.

1. Manual control of robot motion
2. Move to home position
3. Move to ready position
4. Hand open/close
5. Servo on/off

The robot arm has two control modes. One of them is a coordinate control mode. In this mode, the coordinates of the robot hand can be controlled. Another control mode is a joint mode. Joints of the robot arm can be controlled.

The home position is the closing configuration of the robot arm. The ready position is the initial position of the robot arm to start to move.

The robot hand can open and close to grasp an object.

The robot arm can suspend and restart with the servo on/off function.

In this paper, the robot arm has 5 degrees of freedom. The coordinate commands are given as cylindrical polar coordinates.

### D. Communication Protocol

To realize the above mentioned functions in the robot component, we decide the communication protocol between the robot component and the interface component as follows. The data of the output ports of the interface component are as follows:

1. The reference velocity of the robot hand
2. The reference joint angles

The robot component receives the above data and move along with them. The data of the input port of the interface component is as follows.

1. The current joint angles of the robot arm

The interface component can record the configuration of the robot with the data.

The service port is used for asynchronous commands. The service port commands from the interface component to the robot component are as follows:

1. Servo on-off
2. Set of the control mode
3. Move to target joint angles
4. Move to home position
5. Move to ready position
6. Hand open-close.

A precise diagram of component connections is shown in Fig. 3.

RT-Components are connected with “rtc-link.” Rtc-link is a GUI tool that manages connections of InPort/OutPort and Service Ports between RT-Components and performs activation/deactivation of an RT-Component. Rtc-link is a powerful tool which can be used when development and debugging of RT-Components is performed. Moreover, it can also be used in verification and experimentation with a robot system, performing the low level integration of components. Rtc-link is shown in Fig. 4.

Each component is described and executed in Linux and both components can be executed on Windows with VMware.

### E. Panel Interface

Our operational interface includes a button panel shown on the computer monitor. The buttons on the panel change their color in order (scanning in order). When the target button is highlighted, the user can select it with single switch pushing and control the robot arm (Fig. 5).

The panel includes three parts. They are a main panel, a subpanel and a message panel. The user can command the robot to select the button on a main panel and on a subpanel. The message panel shows the executed actions result and the errors.

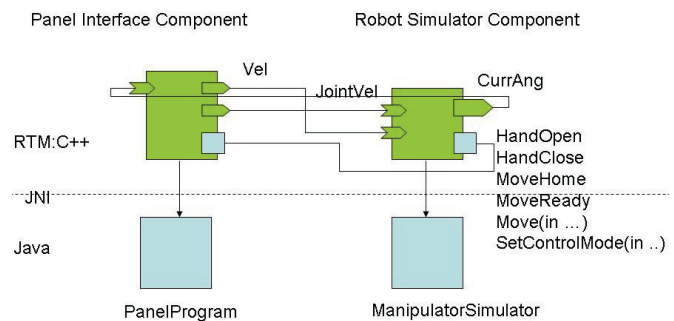


Figure 3. Connection of components

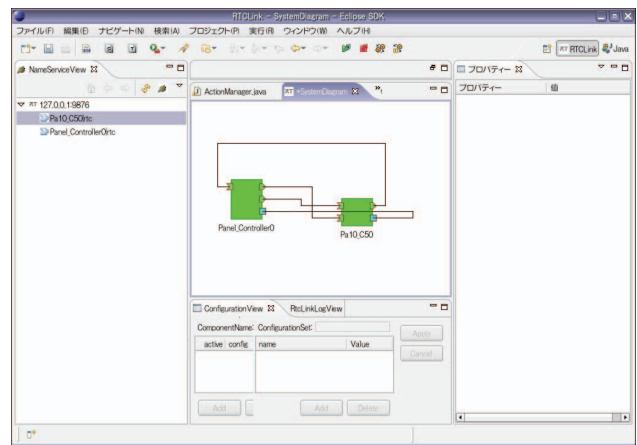


Figure 4. Components on rtc-link

The main panel has six buttons. The first button is to start and suspend panel scanning. The second button is to save and restore the robot's current position. The third button is the move command using the robot hand coordinates. The fourth button is open-close of the finger of the robot arm. The fifth button is move command of the robot's joints' angle. The sixth button corresponds to options.

1) Scan / Suspend Button:

When this button is selected, scanning of the panel is suspended. If the single switch is pressed again, scanning restarts.

2) Save / Restore Button:

This button opens a subpanel to save the current arm position and restore the position. The subpanel is shown in Fig. 6. The subpanel contains buttons to move to ready position and home position.

3) Coordinate Move Command Button:

This button opens a subpanel for the command button of the robot's hand coordinates (Fig. 7). In the beginning of our project, the target manipulator was taken to have 5 degrees of freedom. There are 10 commands. The user can select a button among these 10 buttons to indicate the direction of the robot hand's motion.

The selected buttons control the robot hand to move in the directions in Figs. 8. The directions and rotations of the robot hand motions are the same in the appearance of the simulation and the buttons' characters in the panel.

These buttons are toggle buttons. When the single switch is pressed and the button is selected, the robot starts to move. Then the single switch is pressed again and the robot stops .

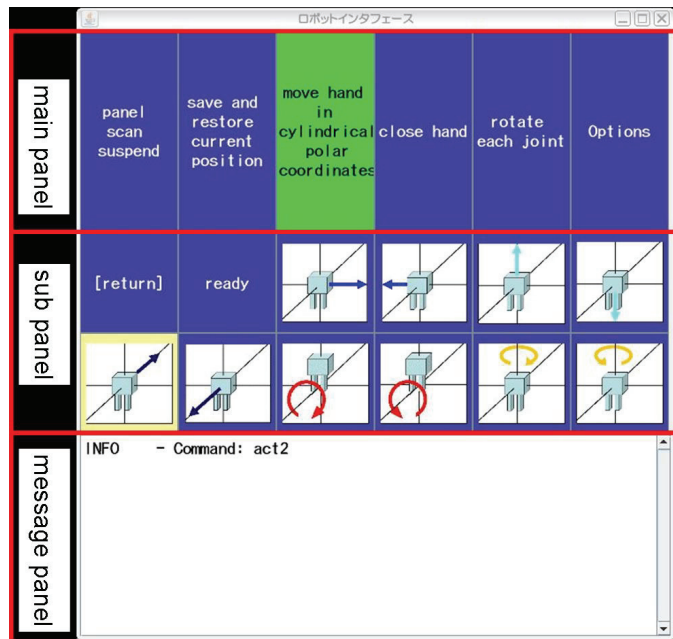


Figure 5. Input panel for single switch scanning

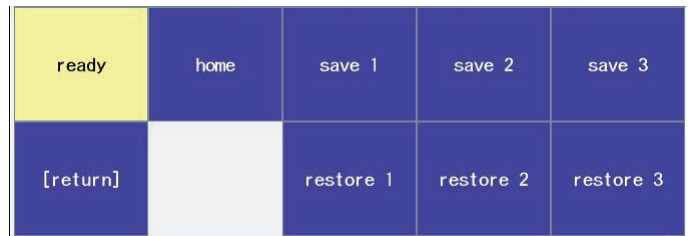


Figure 6. Subpanel for save and restore functions

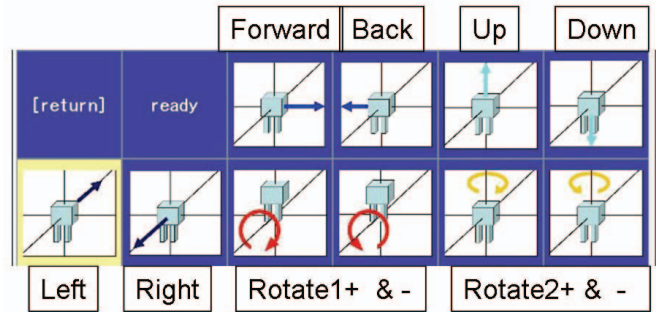


Figure 7. Subpanel for move command buttons

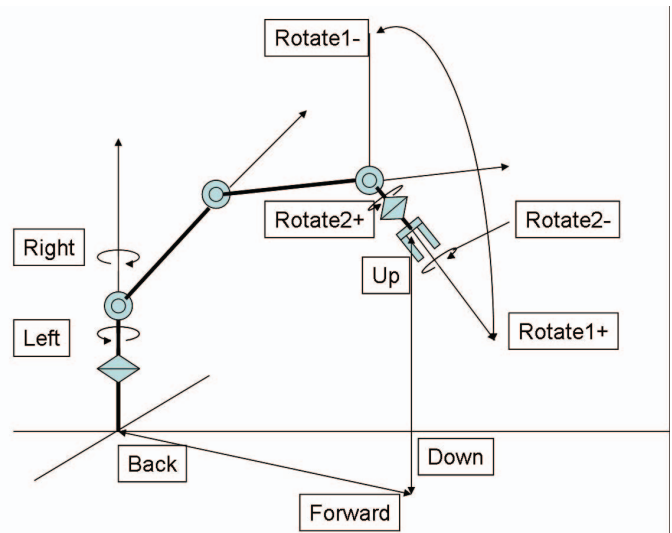


Figure 8. Directions of hand in the simulation

4) Hand Open / Close Button:

When this button is selected, the hand closes if the hand already is open. When this button is selected again, the hand opens.

5) Joint Command Button:

This button opens a subpanel of the joints' angle command of the robot arm (Fig. 9). The user can command 5 joints' rotations in + and - directions. In Fig. 10, the rotations of all the joints of the manipulator are shown.

These buttons are toggle buttons. When the single switch is pressed and the button is selected, the robot starts to move. Then the single switch is pressed again, the robot stops .

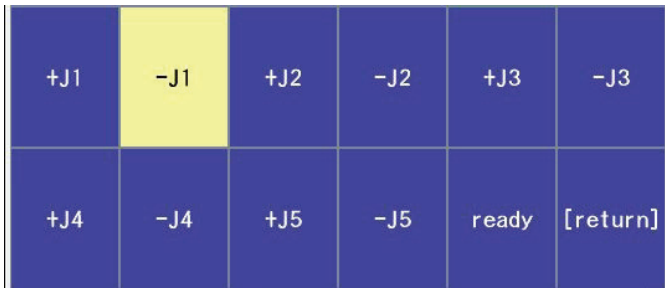


Figure 9. Subpanel for joint command buttons

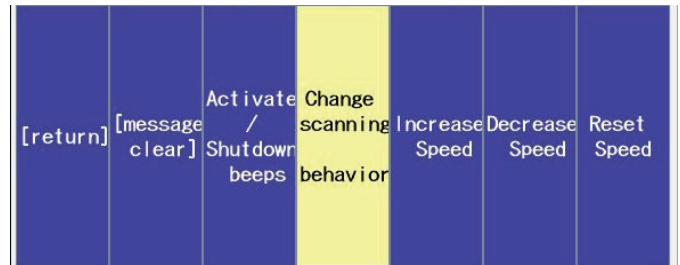


Figure 11. Subpanel for option buttons

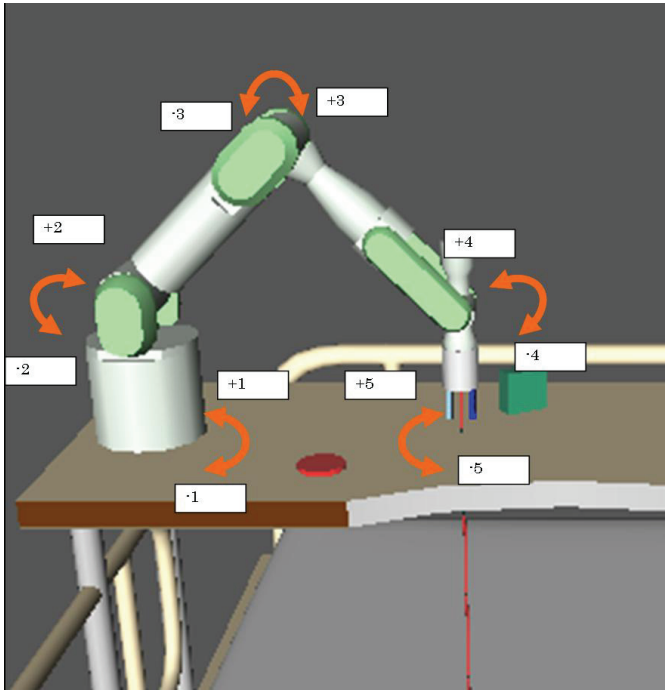


Figure 10. Rotations of joints in the simulation

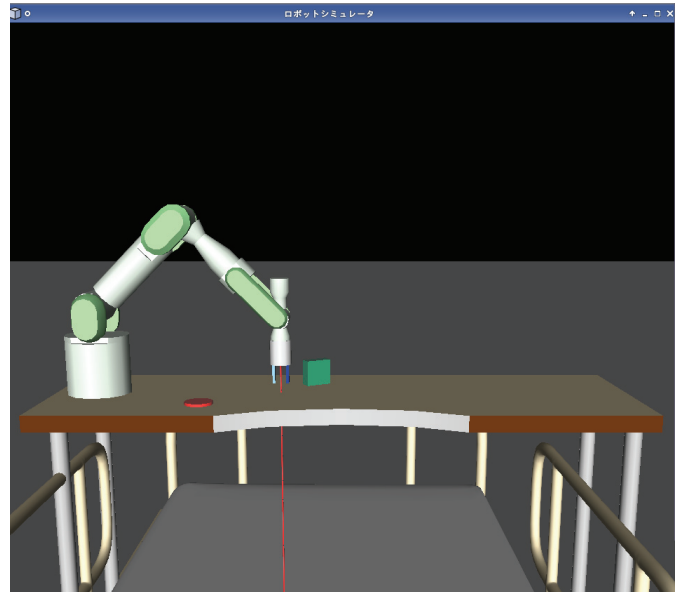


Figure 12. Moving robot in simulation

In our simulator, the contact between the robot hand and the grasping object or the environment is not calculated. When the hand closes near enough to the target object, the simulator decides that the robot has grasped the object and draws the object between the fingers of the robot as the grasping is completed.

The image of the simulator is drawn on a plane. Therefore the user cannot understand the robot hand position precisely in the depth direction. A red line like a laser pointer is drawn from the center of the hand on the vertical line of the hand to show the relative position of the hand in the task environment.

The simulator is written in Java similar to the interface panel and the C++ program calls it through JNI.

#### IV. OPERATION AND EVALUATION BY USERS

We cooperated with the Shimoshizu national hospital, Japan. The hospital has a special ward for muscular dystrophy patients. Two patients made trials of our interface and simulator for 30 minutes. Both of them usually used single switch scanning interface to control computers so they are accustomed to the single switch scanning interface. Their own single switches were connected to the interface computer through a USB connection.

First, examiners simply explained about the interface and the simulator, and demonstrated the task. Then each subject separately performed the task of taking the box and placing it

#### 6) Options Button:

This button opens a subpanel for options to change the panel settings (Fig. 11). Buttons to clear the literals in message panel, to set the on/off of the beep sound of panel scanning, to set the scanning order when the scanning restarts and to change the scanning speed of the buttons are shown in this panel.

All the button actions can be recorded in a log file with their executed time.

Our panel interface is written in Java. The C++ program calls the interface through JNI as shown in Fig. 2. Our system is developed with OpenRTM-aist-0.4.0.

#### F. Robot Simulator

We constructed a robot simulator to evaluate the single switch scanning interface (Fig. 12). The robot arm moves in the simulator window according to the commands from the operation of the single switch scanning interface.

We developed a simple task in the simulator. The task is to let the robot grasp the green box and place it on the red circle. The initial position of the green box and the red circle are alterable with a configuration file for the simulator.

on the target position in the simulator. During the task execution and after the task, the examiners asked subjects about general impressions, contents of buttons, scanning speed of buttons, safety issues and their desire to control robots with this interface. Their comments are shown in Table 1.

Both subjects stated that this interface is mainly good to use and they want to control a real robot with this interface by all means. They also commented on safety functions for the interface.

In this evaluation, only the move command of hand subpanel is used by the examiner and subjects to simplify the evaluation. Subpanels described in the pervious section have been refined using the patients' comments.

TABLE I. COMMENTS BY SUBJECTS

	Subject A	Subject B
Evaluations of easiness of operations	At first I was afraid that it may be difficult to control the robot. I feel easier to use it.	When the new robot will appear with this interface, I want to use them very much.  This robot will be much better than to call a helper.
Evaluations and desire about the panel	Configuration of the button on the panel seems good, but the contents of the button are difficult to understand.  Scanning of the button should start at the beginning of the list when the scanning restarts.  The scanning should be stopped unless the user want to use this interface.	I want to configure the scanning speed. The scanning can be continued anytime.  The number of the buttons on the panel seems to be necessary and sufficient. If I can select one of the many types of the panels, it seems preferable.  Beep sound at scanning of button is preferable.
Request about safety	I want the emergency stopping button to push the single switch for a long time or in the panel	It is better for me if the single switch involves emergency stopping function.  If the system can find the freezing of the computer and let the robot stop, I can feel safer.

## V. CONCLUSION

In order to support disabled people with less muscle strength such as muscular dystrophy patients, we developed a single switch scanning interface to control a manipulator system. The input menu panel is designed to perform various actions of the robot with the single switch and to be easy for users to understand. All the components of the interface and the robot simulator are constructed with RT middleware. Therefore other new interface components can be reconnected to our system without any change in them and utilized if the communication protocol of the components is the same.

Patients with muscular dystrophy tested the interface and made evaluations. Both patients stated that this interface is mainly good to use and they want to control a real robot with this interface by all means.

We will make several improvements based on the comments by the patients. Also we will compare our interface with the other interfaces so we can improve the manipulator. In this project, we will synthesize the interface and the real manipulator for the result. Moreover, this simulator can be a training kit to control a service robot for use in daily life.

## ACKNOWLEDGMENT

The authors appreciate the cooperation of the patients and staff of the Shimoshizu national hospital, Japan in this research. Also we would thank to the staff of the AIST project for the personal robot.

This research was carried out as one of UCROA (User Centered Robot Open Architecture) project at AIST [8].

## REFERENCES

- [1] <http://www.exactdynamics.com/>
- [2] J. Sijs, F. Liefhebber and G. W. R.B.E. Römer, "Combined Position & Force Control for a robotic manipulator," In Proceedings of 2007 IEEE 10th International Conference on Rehabilitation Robotics (ICORR2007), pp. 106-111, 2007 06, Noordwijk, The Netherlands.
- [3] Noriaki ANDO, Takashi SUEHIRO, Kosei KITAGAKI, Tetsuo KOTOKU \* and Woo-Keun Yoon, "RT-Middleware: Distributed Component Middleware for RT (Robot Technology)," In Proceedings of 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2005), pp.3555-3560, 2005.08, Edmonton, Canada.
- [4] <http://www.is.aist.go.jp/rt/OpenRTM-aist/html-en/>
- [5] [http://www.hitachi.com/Int-e/skk/welfare/welfare01\\_01.html](http://www.hitachi.com/Int-e/skk/welfare/welfare01_01.html)
- [6] K. Ozawa, "Barrier Free and Accessibility of Information Processing Equipment," The Journal of the Institute of Electronics, Information, and Communication Engineers. Vol. 81, No.2, pp. 166-169 (In Japanese).
- [7] H. A. Yanco and J. Gips, "Driver Performance Using Single Switch Scanning with a Powered Wheelchair: Robotic Assisted Control Versus Traditional Control.," In Proceedings of the Annual Conference of the Rehabilitation Engineering and Assistive Technology Society of North America, 26-30 June 1998. RESNA Press, 1998, pp 298-300. Minneapolis, Minnesota.
- [8] AIST 2005/2006 Annual Report