Development of a Personal Robot Considering Standardization

Moosung Choi*, Kwang-Woong Yang*, Dae-Heui Won*, Joon-Woo Park**, Sangduk Park*, and Ho-Gil Lee*

*Division of Applied Robot Technology, Korea Institute of Industrial Technology(KITECH), Ansan, Korea
(Tel : +82-31-400-3997; E-mail: {moosung, ygkgwg, daehee, sdpark and leehg}@kitech.re.kr)
**Fine Digital Inc., Korea
(Tel : +82-31-788-8852; E-mail: jwpark@finedigital.com)

Abstract: If a personal robot is popularized like a personal computer in the future, many kinds of robots will appear and the number of manufacturers will increase as a matter of course. In such circumstances, it can be inefficient, in case each manufacturer makes a whole platform individually. The solutions for this problem are to modularize a robot component (hardware and software) functionally and to standardize each module. Each module is developed and sold by each special maker and an end-product company purchases desired modules and integrates them. The standardization of a module includes the unification of the electrical, mechanical and software interface. In this paper, a few prototypes developed based on the concept of this study are introduced and possibility which can be standard platform is verified. Each prototype has merits and demerits, and a new structure of the hardware platform considered them is proposed Also the software architecture to develop the standardized and modularized platform is introduced and its detailed structure is described. The name of a method and the way to use that are defined dependently on the standard interfaces in order to use a module in other modules. Each module consists of a distributed object and that can be implemented in the random programming language and platform. It is necessary to study on the standardization of a personal robot after this steadily

Keywords: mobile robot, standardization, modularization, architecture

1. INTRODUCTION

It seems that the personal robot industry must follow the personal computer business in order to grow systematically and stably hereafter[1]. Personal robots are different from PC(Personal Computer) in the aspect of mobility and possibility that various platforms can appear according to object. But the group of business constructing these industry would be not very different each other. To begin with, the parts of robot are classified according to each function, and these classified parts are developed in the businesses specializing in developing robot modules. Then the integration company manufactures the robot platform considered design and function demanded by customer. The field of robot software, as well as hardware, like operating system and application program will appear to forms of specializing.

In general, personal robots are developed through these all processes in the single company now[2,3]. As the agreement of standardization is not erected, it is difficult to share the technical know-how of each company and possible to develop the overlapping technique. Also, there is a tendency to attach importance to developing hardware structure which is visible, and so it is insufficient to make on effort to develop the algorithm for executing the defined goal or the application program for implementation actually. In order to keep an increasing interest and respond to the demands of customer, it is necessary to develop technology ceaselessly and study the demanded technology in concentration.

This study started to develop the technology based on developing a robot systematically and in specialization. Through this development, the rate of developing technology is accelerated and it is possible to respond to the demands of customer more rapidly. The prototype developed in this study is not the final proposition for standardization and it must be erected through the continuous study.

This paper is organized as follows. In Section 2, a few prototypes developed based on the concept of this study are introduced and possibility which can be standard platform is verified. The software architecture to develop the standardized and modularized platform is introduced and its detailed structure is shown in Section 3. Finally, a few conclusions are drawn in Section 4.

2. HARDWARE STRUCTURE

A few prototypes have been developed since this study started. For the sake of specialized development, the parts of a robot are classified efficiently and modularized, and the easiness of integration must be considered. In this section, the developed prototypes are introduced, and then the detailed structure and merits and faults of each prototype are described.

2.1 CMR(Component based Modularized Robot)-P1

As shown in figure 2, a robot is divided into four modules(mobile, sensor, brain and vision) according to the function, and a robot is integrated as a method piling these modules[4,5,6]. The single board computer, which the operating system can be loaded, is taken in each module. The network media like IEEE1394 and TCP/IP are used for communication between modules. The mechanical and electrical standard interfaces are defined for combination between modules, and one module is able to call other modules as software standard interface is defined. There is a battery in each module, so hot-plugging is possible. Because

Fig. 1 CMR-P1

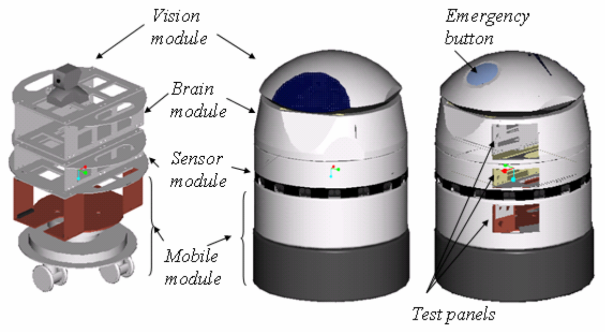

Fig. 2 Structure of CMR-P1

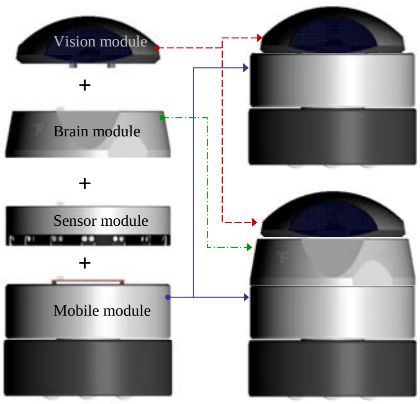

Fig. 3 Module combinations

the mechanical and electrical interface of all modules is the same, swapping between modules is possible. The openness of the software architecture is verified as showing that the mobile module with differential driving wheel structure, which is developed through this study can be replaced with the commercial product which has synchro driving wheel structure and manufactured at Hanool Robotics corp.. That can be realized by adjusting to the proposed standard interface.

2.2 CMR-P2

The modules of CMR-P1 should play the role of a frame, as they support the load of a whole robot and an external form is decided according to the shape of modules. Accordingly, a module is demanded on enough strength, and the external form of a robot must be considered when a module is developed. In case of rearrangement of modules, the mechanical and electrical connectors must be all unified. In order to do so, modules that necessitate small space like a brain module must be enlarged unnecessarily. And also it is a problem when dispersed sensors are attached and detached. For these reasons, it is difficult to be standardized and manufactured.

Above problems were solved by importing a frame shown in figure 7. Modules of CMR-P2 can be miniaturized and lightened, as a frame plays a role in supporting the load of a whole robot. Also the rearrangement of modules is possible and an external form can be changed easily by replacing with other frame. The sensors of variable sizes and shapes are attached and detached on a frame easily through the process of standardization.

CMR-P2 is composed of modules, sub-modules and a frame. A module, like brain, sensor, mobile and arm, has the operating system, but a sub-module, like laser, ultrasonic, vision, gyro and battery, does not and can not compute. The

Fig. 4 CMR-P2

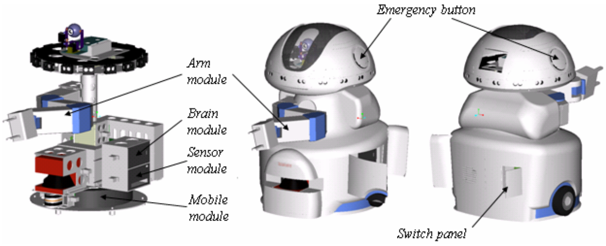

Fig. 5 Structure of CMR-P2

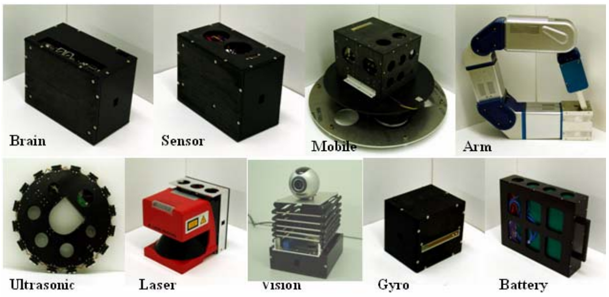

Fig. 6 Modules of CMR-P2

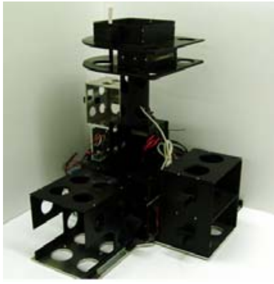

Fig. 7 Frame of CMR-P2

communication devices and the power management devices are set on a frame, and communication/power lines between modules are supplied.

As the standard interface is defined for combination between modules at CMR-P1, the standardizing proposition of the mechanical and electrical connector for hardware combination between a frame and a module is proposed and software standard interface is used at CMR-P2. The Sliding and locking mechanism using a stopping device is used for mechanical combination, and a proposed standard connector like figure 8 is used for electrical combination. Because several communication/power lines are demanded, model Type M of HARTING Electronics is used. This is composed

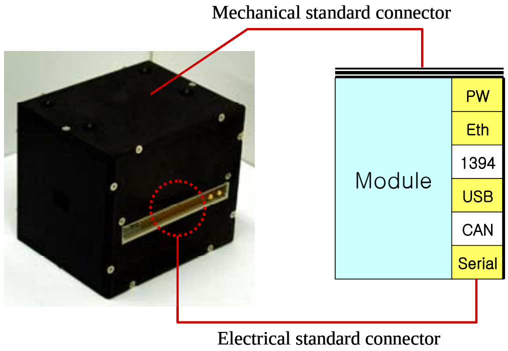

Fig. 8 Mechanical standard interface

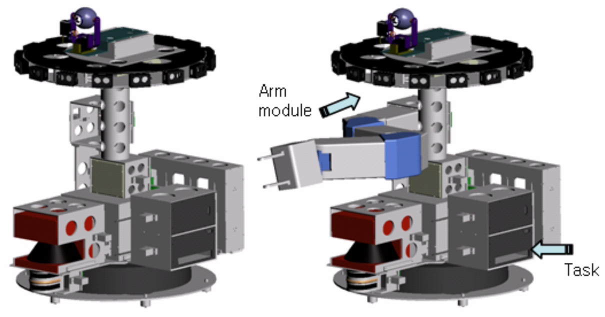

Fig. 9 Attachment and detachment of modules

of 4 power pins and 60 communication pins. 60 communication pins are divided into 2 TCP/IP, 3 USB, 2 IEEE 1394, 4 RS-232C, 2 RS-485 and 2 CAN.

2.3 CMR-P3

CMR-P2 has so many modules and sub-modules, and then the mechanical structure become complex and the size of a robot is made large. In CMR-P3, the mechanical modularization is sublated and the concept of a sub-module is removed. And option parts in return for that are added. These option parts can be laser scanner, infrared scanner, sonar sensor and camera, and so on.

CMR-P3 is composed of a mobile base and option parts. A mobile base and a frame of option parts are combined by using the locking device. Power and communication lines are connected by the standard connector according to each communication media, not a proposed standard connector used in CMR-P2. The reasons for that are as following:

- It is difficult to reflect the connecting properties of each communication media.
- It is considered that standardizing a connector which the various communication connectors unified into is not adequate judging from the increase and its acceleration of the communication media.
- There are not many occasions to use the proposed connector as the mechanical modularization is sublated.

A mobile base(mobile module) and a tablet PC(brain module) play roles of a module. TCP/IP is used as the communication protocol between modules. The frame of options parts has communication related devices and power management devices built in.

Fig. 10 CMR-P3

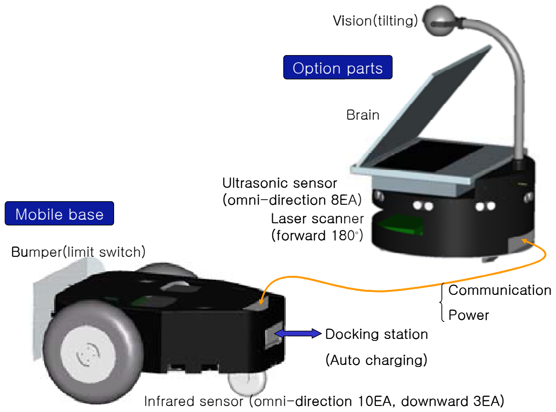

Fig. 11 Structure of CMR-P3

2.4 Proposed structure

Modules can be swapped by using the same mechanical interface, and hot-plugging is possible by distributing batteries in CMR-P1. However these functions are seldom needed to end-user using a robot practically and this structure is not reasonable as described in section 2.2.

Figure 12 is a proposed structure, which demerits of CMR-P2 are complemented and the structure of CMR-P3 is adopted partly. The concept of a sub-module in CMR-P2 is removed and option parts in return for it are added to the frame. The various kinds of sensors are added mainly. Because the shapes and kinds of sensors are various, the fitted location is not constant and new sensors are developed continuously, the mechanical standardization means nothing. It is more efficient that the high level controller(single board computer) is set alone and easy to upgrade than included in a module, because the product of higher performance is coming out rapidly. Also the number of controllers can be added according to the needed computation performance. Communication and power related devices are built in a frame. Mobile base and manipulator are defined as a module, as they are made of several parts and able to have the various constructions according to the combination method. These modules are connected to the controllers in a frame through USB, and so on, and they are used as a peripheral device of the personal computer. As installing device driver which has the information of their own and the software standard interface, modules will operate. The standard connector according to each communication media is used as an electrical connector of a module like CMR-P3. The

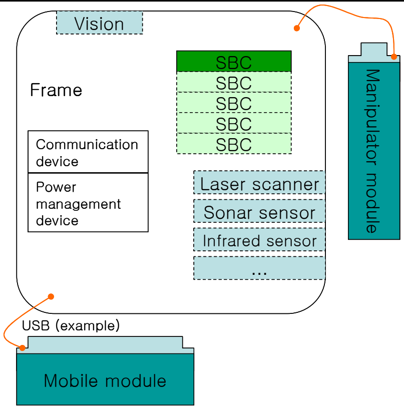

Fig. 12 Structure of proposed robot platform

mechanical connector would have to be a minimized form, which does not disturb developing a module.

3. SOFTWARE ARCHITECTURE

Various techniques which can be applied to a personal robot were developed for several decades. But these techniques were not integrated efficiently, so that is an obstacle to develop a robot. Also, as the application fields are various, the software to implement this various task comes to be complex. In this current circumstance, the new paradigm to develop a robot is needed in aspect of software as well as hardware. Some solutions for these problems are proposed in this study as following:

- Modularization and standardization of a robot in aspect of software and hardware
- Introduction of the distributed software architecture
- Development of device driver of each module and software library

In case of composing more than one module which uses various CPU and operating systems, the communication between modules must be considered. Also, if the application program is developed based on hardware or operating system which operates on, the structure of OS or hardware must be considered.

MRSF(Modular Robot Software Framework) proposed in this study is the architecture which separates an application program from hardware and OS and copes with distributed environment. MRSF architecture separates the robot software to a four level hierarchy and the development group is divided into four parts, which develop each software. Four parts are as following:

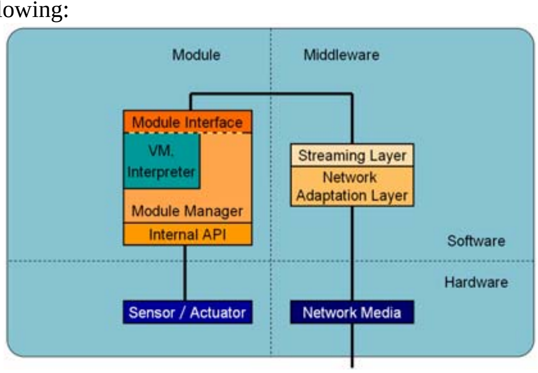

Fig. 13 MRSF architecture

- Development of the application program
- Development of the library and component
- Development of the architecture
- Development of the device driver

The application program is the highest level software like HRI(Human-Robot Interface). The lower level software like the device driver or library is used to develop this application program. A module software developer must offer the module possible to develop the application program or library as installs OS, MRSF architecture and needed library.

The library and the component are a collection of software which special function implemented. The software like localization and object recognition is an example. While the library depends on the programming language, the component is independent of the language due to have the interface defined as MIDL(Module Interface Definition Language).

MRSF architecture is provided not as the concept but as a state of source encoded practically. Accordingly, an application program developer can finish the development to correspond to own area without general understanding about the architecture.

The device driver which can be called Internal API is a collection of software which depends on hardware. If the module has I/O devices, the function for controlling and monitoring these devices should be implemented. This device driver is defined as standard interface and can be used in the library of the upper level and the application program. Also, it is included to the part of the architecture and implemented in the C language like the architecture.

In the preceding study, RPL(Robot Programming Language), RVM(Robot Virtual Machine) and middleware are developed[7, 8]. RPL is used to encode the library and the application program of a robot independently of a platform. RPL is developed based on C programming language which most programmers are familiar with and does not include the grammar easy to create the error and dependent on the platform like the pointer.

RVM is demanded to execute the byte code which is the result of compiling. In general, the virtual machine is defined as the environment which imitate a mechanical structure or a hardware platform or the virtual environment realized by this virtual structure.

In the middleware, the massage format and the communication method to provide the communication service between modules are defined. The massage between modules means that the module-control process executed in a module demands the module-control processing data in the remote site or is sent to call a function of the module-control process of other modules. The standard message format must be used between modules with the view of this function. Middleware is composed of two levels, which the first level is NAL(Network Adaptation Layer) and the second level is SL(Streaming Layer). In SL, the message data structure received from the application program is made a conversion to the stream. NAL accept various heterogeneous network. This layer sends data stream received from SL to remote NAL through the network media. And also, remote NAL sends to SL the accomplished data which is mixed the received stream as the arranged format and condition. Network media used on this occasion are Ethernet, Bluetooth, IEEE1394 and USB, and so on.

Table 1 Input output specification of software interface

	item	function
input	Position command	void Set_pos(int x, int y)
	Velocity command	void Set_vel(int drv)
	Control mode	void Set_ctl_mod(int mod)
	System constant	void Set_const(int id, int v)
output	Current position	int Get_pos_x() int Get_pos_y()
	Current velocity	int Get_vel()
	Control mode	int Get_ctl_mod()
	Constant & variable	int Get_const(int id)

4. CONCLUSIONS

In this paper a few prototypes developed based on the concept of this study are introduced and a structure of the standard platform is proposed. Also a software architecture developed to use this standard platform effectively is introduced and the standard interface is proposed.

If the hardware and software architecture proposed in this study are adopted as a standard structure, the groups of businesses as following can be derived.

- The company developing a module like mobile base and manipulator
- The company developing sensors for robots
- The company developing a frame
- The integration company
- The company developing the robot operating system and application program

From now on, it is demanded that the study on a standard proposition that the company developing components or robots can accept is made steadily. As a demand is created through the efficient and rapid development by establishing the reasonable standard proposition from the proposed standard proposition, the personal robot industry could be growing.

ACKNOWLEDGMENTS

This study is supported by The Project of Basic Technology Development for Personal Robots of Ministry of Commerce, Industry and Energy in Korea.

REFERENCES

[1] Yoffie, David B., "Introduction: CHESS and Computing in the Age of Digital Convergence", Harvard Business School Press, pp. 1 - 36, 1997.

[2] Aramaki, S., Shirouzu, H., Kurono, S., Mino, M., Uno, Y., Hara, K., Tanaka, H., Tsuruoka, T., "Development of autonomous mobile humanoid robot", The 25th Annual Conference of the IEEE Industrial Electronics Society, Vol. 2, pp. 529 - 534, 1999.

[3] Kaloutsakis, G., Tsourveloudis, N., Spanoudakis, P., "Design and development of an automated guided vehicle", IEEE International Conference on Industrial Technology, Vol. 2, pp. 990 - 993, 2003.

[4] Roh, S. G., Baek, S. M., Lee, D. H., Park, K. H., Moon, T. K., Ryew, S.W., Kim, J. Y., Kuc, T. Y., Kim, H. S., Lee, H. G., Choi, H. R., "Development of Personal Robot Platform : Approach for Modular Design", ICCAS, pp. 2313-2318, 2002.

[5] S. G. Roh, K. H. Park, K. W. Yang, J, H, Park, H. S. Kim, H. G. Lee, H. R. Choi, “Dynamic Infrastructure for Personal Robot : DynI”, ICCAS, pp.2039-2044, 2003

[6] S. G. Roh, K. H. Park, K. W. Yang, H. S. Kim, H. G. Lee, and H. R. Choi, "Development of Dynamically Reconfigurable Personal Robot," ICRA, pp.4023-4028, 2004.

[7] Gun Yoon, Hyoung Yuk Kim, Ju Sung Lee, Hong Seok Kim, Hong Seong Park, “Middleware Structure for Personal Robot,”ICCAS, pp. 153-157, 2003. 6.

[8] KwangWoong Yang, Hong-Seok Kim, Jaehyun Park, “A Virtual Machine for Modularized Personal Robot Controller,” Proceedings of ICCAS, pp.2170-2173, Oct. 16-19, 2002, Muju, Korea