Proposal for Teaching Manufacturing And Control Programming Using Autonomous Mobile Robots with an Arm

Shuji KUREBAYASHI[1], Hiroyuki AOKI[2], Toshiyuki KAMADA[3], Susumu KANEMUNE[4], and Yasushi KUNO[5]

[1] Shizuoka University eskureb@ipc.shizuoka.ac.jp
[2] Korea University hiroyuki.aoki@inc.korea.ac.kr
[3] Aichi University of Education tkamada@acm.org
[4] Hitotsubashi University kanemune@acm.org
[5] University of Tsukuba kuno@gssm.otsuka.tsukuba.ac.jp

Abstract. We propose a technology education curriculum for lower secondary school students using an autonomous mobile robot with an arm. The purpose of our curriculum is to teach the concept of systems that work with mechanics, electricity and computers. For this purpose, we have developed a control board and a computer language for an autonomous mobile robot with an arm. The benefit of this kind of robot is that students have to seriously think about the program for controlling the arm to lift and carry objects. This kind of serious thought is not necessary in programming simple mobile robots without arms[1]. In this paper, we will report a test conducted to evaluate our teaching materials and lessons in lower secondary school. As a result, our technology education curriculum satisfies requirements for students that have more incentive to learn the concept of systems.

1 Introduction

In today's information society, we are living with lots of embedded computers, such as those within refrigerators, cars, and so on. To foster zest for living in this information society, it is important to understand models of how embedded computers control machines.

However, in Japanese "Technology and Home Economics" subject, mechanics and computers are treated separately; mechanics controlled by embedded computers are not taught. Therefore, students can not get the idea of systems with embedded computers, in spite of widespread technology which we use everyday in our life. The downside of the fact is that they will not become citizens who understand the risks of technology and an evaluate the technological products correctly.

In Japanese lower secondary schools, robot contests are quite popular (used in more than 3,000 schools) in the manufacturing lessons of "Technology and Home Economics" curriculum. In the robot contests, students compete with each other to test their skills of controlling their hand-made robots. However, those robots are not autonomous robots; they are controlled by humans through remote controllers. Therefore, from such contests, students learn much about mechanics, but almost nothing about computer controlled mechanics. We believe that the heart of today's embedded systems is computer controlled automatism, and students should definitely learn such concepts. To enable such learning, (1) appropriate material and (2) effective curriculum are neccesary.

In this paper, we present an autonomous tri-axial robot (a mobile robot with an arm) that we have developed as teaching materials. Tri-axial means three motors, two of which are used to drive wheels to move the robot around, and one of which is used to lift the arm to carry things. Such robots can do some useful work (e.g. carrying things and so) are attractive to the students. We also present a curriculum we have developed which effectively make use of such robots. Additionally, results of two experimental classes, one for lower secondary school students and another for teachers, are reported. Finally, we discuss the effectiveness of those curricula and teaching materials for students to learn the computer controlled mechanics.

2 Overview of Our Hybrid Lessons

Our goal is to make the students comprehend the model in which mechanics are controlled by embedded computers (and programs running on them). However, such complex models cannot be taught in a single step. Therefore, we propose a step-by-step style of lessons as in the following:

(1) Manufacturing the robots
In the first step, students assemble mechanical parts of the robot car (body, gearwheels, motors, arms, and so on). They also learn wiring of electric parts so that they can understand what electrical parts (CPU board, power supply, motors, sensors and so on) their robots have, along with their connecting topologies.

(2) Primitive programming experiences
In the second step, students experience primitive programming. With this lesson, students understand that every action of a computer is controlled through programs, and programs work in a concise step-by-step manner.

(3) Programming to control robots
As the third step, students finally craft programs and download those programs to the robot car, which operates autonomously under program control.

Through these lessons, students can grab practical models of computer controlled mechanics along with collaboration among hardware and software. This goal has been difficult to achieve in traditional lessons in Japan in which mechanics and computers were taught separately.

3 Design of Teaching Materials

To carry out our lessons effectively, appropriate teaching materials are necessary. The requirements for the teaching materials are as follows:

- The material should support learning of how the mechanics work. Therefore, instead of being complete products, assembly kits, with which students compose various parts as gearwheels, arms, etc. together, are desirable.
- The material should support learning of computer hardware. Through wiring, and soldering to put the electric parts on a circuit board, students can understand how the electrical hardware such as CPUs, sensors and electric power supplies make the mechanics work.
- The material should support learning of embedded software with minimal effort. Spending lots of time learning programming language syntax or complex libraries is undesirable; the majority of time should be used to actually experience robot control. Therefore, easy-to-learn programming languages and/or environments with Japanese language support are necessary.
- The materials should be moderately priced, so that each of the students can have their own robot. To achieve this goal in practice, price of materials should be around ¥3,000 (about €20) or so.

There was no teaching material which satisfies all of the above condition. Therefore, we have developed (1) electrical board with CPU to control robot cars, and (2) educational programming system which runs cooperatively on electrical board and personal computer.

3.1 Control board

For the purpose of controlling robots, programming facility is required of the control board. Our control board uses a PIC (CPU, program/data memory and external interface packaged on a chip). Within the PIC, there is a small byte code interpreter, and students' programs are transferred from outside and stored as byte code programs. Some of the byte code operations put a signal on the external interface through which motor rotation is controlled.

In our curriculum, students put electrical parts on the control board and connect them through soldering and wiring. Through those experiences, students get a general idea of electrical circuitry and operation of control board. The circuit board we developed is shown in Fig.1.

As described earlier, our robots are tri-axial; two motors are used to control two wheels (to move the robot car around), and one motor is used to raise/lower arms. Therefore, our control board is capable of controlling three DC motors, and also has four input ports to connect various sensors/switches. The sensors include light sensor and touch switch (collision detector).

We use PIC16F630[2] as the micro controller on the board. This PIC has FLASH memory, and we record our firmware. The firmware is responsible for receiving the byte code program from the host computer and running (interpreting) the program. The maximum length of the byte code programs is 127 bytes.

Most of the byte code operations occupy 2 bytes (operation code and argument). Byte code programs are transferred to the board from the host computer using RS232C cable and interface.

Aside from the control board, the mechanics of the robot car is built using standard parts, such as general purpose frame, gearwheels, wheels and so on. The work of a student is shown in Fig.1.

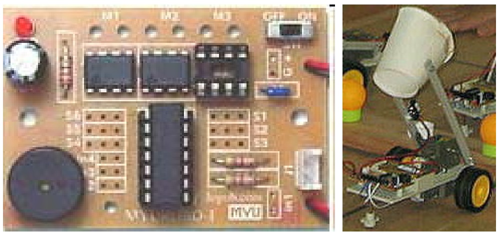


Fig. 1. The control board and student's work

3.2 Programming Language "Dolittle"

```
ROBO= MYU! "com1" create.                                // (1)
ROBO: sendProgram= [! beginRobot                         // (2)
  powerOnStart                                           // (3)
  [!                                                     // (4)
    [! 2senserNo ifLow ] then [! rightForward ]          // (5)
    else [! leftForward ] execute                        // (6)
    [! 1senserNo ifHigh ] then [! getBall ] execute      // (7)
  ]repeat                                                // (8)
endRobot].                                               // (9)
ROBO: getBall= [!                                        //(10)
  10 stop                                                //(11)
  [! 4senserNo ifLow] whileRepeat [! motorRight ] execute //(12)
  10 motorLeft                                           //(13)
  10 back                                                //(14)
].                                                       //(15)
ROBO! sendProgram.                                       //(16)
```

Fig. 2. An example control program

We adopted "Dolittle" [3] as a programming language and environment. Dolittle is an object-oriented educational programming language with compact syntax, and students can write programs using Japanese characters. Dolittle has turtle graphics (inherited from LOGO). Turtle graphics are well suited to primitive programming experiences because students can see the result of their programs on the screen.

Additionally, we have implemented a simple robot language translator written in Dolittle. With this translator, students can write their robot control programs using Dolittle syntax. However, those robot control programs are translated to byte code and transferred to the robot car.

Dolittle also has the facility to read/write RS232C communication ports, and this facility is used to control communication between the host computer and PIC monitor and transfer byte code programs.

Fig.2 shows a sample program for controlling a robot:

- (1) makes a robot object which will communicate through the port "com1".
- (2)–(9) defines a method "sendProgram".
- (2), (9) "beginRobot" and "endRobot" delimit the beginning and end of the robot control program.
- (3) specifies that the program of the robot starts soon after the power switch is turned on.
- (4)–(8) repeats infinitely.
- (5), (6) if the left sensor (sensor 2) touches a wall (or object), the robot moves to the right, otherwise to the left.
- (7) if the center sensor (sensor 1) touches an object, the robot calls up method "getBall".
- (10)–(15) defines method "getBall" : The robot stops, moves the shovel up to the limit, then goes back.
- (16) transfers compiled byte codes to the robot.

Adopting Dolittle[3] made it possible for lower secondary school students to learn this primitive program easily in a short period. This was essential to the success of our curriculum because we cannot invest many hours for teaching programming alone.

4 Experimental Classes for Teachers

To assess the usefulness and practicality of our materials and curriculum, we have given two workshops, or experimental classes, for teachers. One was given in August 2006 for five teachers at Fukuyama city in Hiroshima prefecture; another was in November 2006 for eight teachers at Fujieda city in Shizuoka prefecture.

In the workshop, teachers first manufactured mobile robots with an arm, then controlled them with their programs. The major goals of the lessons, when applied to students, are as follows:

- Students can learn widely about mechanics, electricity and computers.
- Students can learn outline of systems in which mechanics are controlled by electricity and computers.

We asked the teachers to asses whether our materials and curriculum meet the above goals, and to give us feedback as to what portion of the materials and curriculum need to be revised (where it is too difficult for students or has other problems). Kurebayashi took charge of two workshops.

Table 1 shows the contents of the workshops.

Table 1. Schedule of the workshop

description	#hours
robot manufacturing	3
robot programming	3

After the classes, we conducted a questionnaire survey to evaluate our materials. In the questionnaire, teachers answered each question choosing from a five level scale (5: strongly agree 4: agree 3: not sure 2: disagrees 1: strongly disagree). The questions and the numbers for each choice are shown in Table 2 and Table 3.

Table 2. Q1 and Q2: Difficulty for students

	Difficult				Easy
	5	4	3	2	1
Q1. robot manufacturing	0	2	6	5	0
Q2. robot programming	0	1	5	6	1

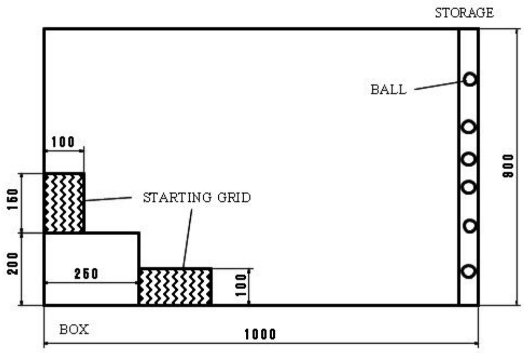
Table 3. Q3–Q6: The effect expected from our teaching material and curriculum

	Agree				Disagree
	5	4	3	2	1
Q3. Good for students to study informatics and computers	6	7	0	0	0
Q4. Good for students to study technology and manufacturing	6	6	1	0	0
Q5. You would like to perform robot contest in class	5	7	1	0	0
Q6. Good for students to foster creativity and invention skills	6	7	0	0	0

In total, we think that subjective evaluations from teachers have confirmed that our teaching materials and curriculum are effective for use in classes, both for teaching manufacturing and teaching computers in a unified manner, and also will have the effect of developing students' creativity.

5 Experimental Classes for lower secondary school Students

To evaluate effectiveness and practicality of our materials and curriculum with actual students, we conducted experimental classes in Hanashi lower secondary school (3rd grade, 123 students), Shizuoka prefecture, Japan. The lesson started with each student creating arobot and then composing a program for the robot. At the end of the lessons, a robot contest was held. Mr. Akiyama took charge of these lessons.



Rule of the Programming Robot Contest:

1. The robot must start from the starting grid
2. Pick up the ball on the storage
3. Carry it into the box

Fig. 3. The field layout and rules of the contest

5.1 Contents of the classes

The contents and the number of hours for the classes are shown in Table 4. After the overview, students practiced simple graphics programming using Dolittle. Then, students manufactured their own robots.

At the end of the arm of the robot, a paper cup with its bottom cut out and an elastic band is fastened; when the robot puts the cup over an ping-pong ball and raises the arm, the ball enters the cup and can be carried by the robot. Students can freely elaborate and/or enhance the mechanism at will.

For robot control programming and the final contest, the task of picking up ping-pong balls and carrying them was chosen. The layout of the field and the rules are shown in Fig.3. Scenes from the lessons are shown in Fig.4.

5.2 The result of the lessons

All 123 students were able to complete their robots and control programs on time. Also, there was no problem in learning the basics of programming. We think that

Table 4. Contents and schedule of the classes

contents	#hours
Let's learn about computer controls	1
Let's try making a program	4
Let's manufacture a robot	10
Create a program for robot control	3
Robot contest	3

Fig. 4. A student working on his task and Students programming their robot

the choice of Dolittle programming language and associated robot programming system contributed to this. Students were modifying and enhancing the sample program at will; after trying their programs on the robot, they could change the program to suit their intentions. As a result, the final robot contest was held successfully.

5.3 Questionnaire to the students

After completing the lessons, we gave a questionnaire to the students. The goal of the questionnaire was to evaluate the following points:

- Students' willingness to learn
- Ease/difficulty level of our materials

The questionnaire was composed of 5 closed questions and a text description. The selective questions were answered with a 5-level scale (5: Strongly agree, 4: Agree, 3: Not sure, 2: Disagree, 1: Strongly disagree). The content of selective questions and numbers of students with percentages those who answered positively (choice 5 and 4) are shown in Table 5.

The result suggests that students perceived both manufacturing and programming as "hard fun" [4]. Therefore, the difficulty of our materials and curriculum seems to be appropriate for lower secondary school students. Moreover, the ratio of answers "I would like to continue the lessons" was very high; these lessons raised students' willingness to learn considerably.

Table 5. Results of the enquiry

description	#students	%positive
1. Manufacturing robots were difficult.	88	71.5
2. Manufacturing robots were enjoyable.	115	93.0
3. Programming robots were difficult.	87	70.1
4. Programming robots were enjoyable.	91	74.0
5. I am willing to continue this lessons	96	78.0

We also requested students to describe freely at which points they felt difficulty. Multiple answers were allowed. The purpose of this question was to investigate what is the hard point for the students and what aspect of the classes were useful to them, from their own viewpoint. All students described something, resulting in 243 answers in total. We categorized the answers, whose summary is shown in Fig.5.

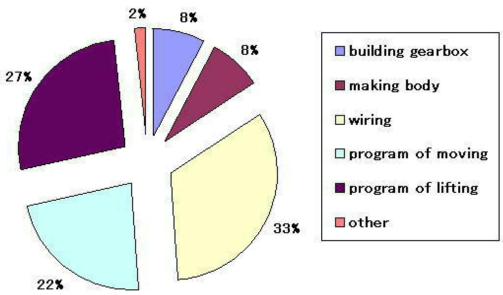


Fig. 5. Summary of “difficult points”

From the above summary, we can point out that students learned a lot about mechanics, electricity and computer programming. In other words, we think that our original goal of teaching students a combined system of mechanics and computer control was successfully achieved. Moreover, the ratio of answers concerning computer programming is very high (49% in total), indicating that students put lots of effort into computer programmings; an accurate reflection of today’s embedded system or computer-controlled devices.

5.4 Study on level of understanding embedded systems

The original motivation for our curriculum is to develop students' understanding of embedded systems in which computers control mechanics, because such systems are widely used, and today's citizens should understand the principles and risks of such technology. To assess this kind of understanding, we made another question to the students. The question concerns the mechanism of elevators, which are representative of computer controlled mechanics and also are familiar to the students. We showed them the picture of Fig.6 and asked the following questions:

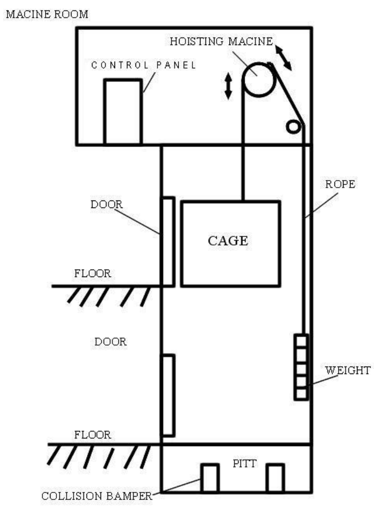


Fig. 6. Mechanism of elevators

Q1 Do you understand the meaning of the phrase "errors in control panel program?"

Q2 Please explain the role of the control panel.

For Q1, we asked students to choose one from "Yes," "Not sure," or "No." For Q2, we checked if the student could choose at least one part shown in Fig.6 and explain its role correctly.

As a control group, we made the same questionnaire to students of another public lower secondary school in the same city. This school does not teach about embedded systems at all. The number of students was 111. Fig.7 shows the ratio of students who chose "Yes" for Q1 and could correctly answer Q2 for two schools. Additionally, we asked the students to raise their hand if they thought that they understand the mechanism of the elevators, but no students raised their hands.

As a result, the ratio of students who think they can understand the meaning of the phrase "errors in the program of control panel" is almost the same, but there is a large difference among the two groups in the ratio of students who can write the role of control panel correctly.

In the control group (no embedded-system learning), there was a strong correlation in understanding the role of the control panel and understanding the meaning of the phrase "errors in the control panel program." On the other hand, in the experimental group (which experienced our curriculum), even when the students answered "No" in Q1, lots of them could write the role of control panel correctly in Q2. This means that there are many students who understand the principle of systems with computer controlled mechanics, even if they do not comprehend the somewhat difficult concept of "errors in the control panel program." The result confirms the effectiveness of our materials and curriculum in enhancing understanding of the principle of embedded systems.

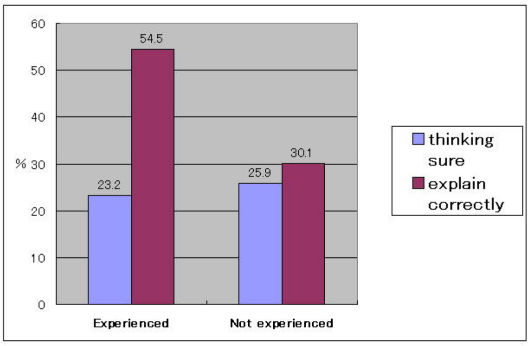


Fig. 7. The result of questionnaire on elevators

6 Discussion

We have developed teaching materials which can motivate students, through which students could understand the principle of embedded systems. Students did not feel much difficulty in manufacturing the robots and controlling them, as the robot consisted of processable materials such as plastic and brass, and programming was easy to learn.

In the class, students handled lots of parts, such as motors, sensor switches and control boards. Through these experiences, they could understand the principle of systems in which motors move a robot and computers control motors referring to signals from sensors. As a result, students thought "It's difficult but fun." and teachers thought "It's suitable for our classes".

The robot contest was effective for learning. Students could refine their control programs according to their intention, meaning that they could have concrete understanding of computer controlled mechanics.

The elevator questionnaire revealed that students who experienced our curriculum could answer properly, compared to the control group. This also indicates the effectiveness of our materials and teaching methods.

7 Related Works

LEGO Mind-storms[5]can control a tri-axial robot which is constructed by LEGO blocks. The robot can be programmed by a graphical language. The CPU and other parts are packaged, so we can't see the electrical parts inside the block. These features are suitable for novices. But we think that in technology class, students should have the experience of seeing and constructing electronic circuits in order to understanding systems, and they should have the experience of programming and debugging by coding.

Cricket [6]can control a tri-axial robot also. It can be controlled by a graphical language. It is useful for scientific education. But it is too expensive for students, so we didn't use it in our classes.

8 Conclusion

We developed autonomous mobile robots with an arm and conducted experimental lessons. The robot can be used for robot contests in classes. Students can learn the concept of systems by technology education and information education.

There are many system products, but they are a black box for many users. Users cannot understand what they are and how they work. To understand their principles, users should understand principles of software and hardware.

But not so many lower secondary school and higher secondary school students want to become specialists. Therefore teaching materials and concepts which students can be interested in are needed.

Our teaching materials for autonomous mobile robots with an arm and control programs are useful for hardware and software system education. By the experience of manufacturing robots and control programming, We can expect that students began to know the systems by analogy. Thus we are convinced of the effect to learn the concept of systems that work with mechanics, electricity and computers by using our teaching materials in Information and Technology education.

References

1. Kurebayashi, S., Kamada, T., Kanemune, S. : Learning Computer Program with Autonomous Robots, LNCS 4226, Springer, pp. 138–149, 2006.
2. Microchip Coop. : PIC16F630 Datasheet, 2007 http://ww1.microchip.com/downloads/en/DeviceDoc/40039E.pdf
3. Kanemune, S., Kuno, Y. : Dolittle: An Object-Oriented Language for K12 Education, EuroLogo 2005, Waruzawa, Poland, pp .144–153, 2005.
4. Papert, S. : The Children's Machine, Basic Books, 1993.
5. Resnick, M.: Behavior Construction Kits, Communications of the ACM, Vol.36, No.7 (1993), pp.64-71.
6. Martin,F.,Mikhak B.,Resnick,M.,Silverman B. and Berg,R. :To Mindstorms and Beyond :Evolution of a Construction Kit for Magical Machines. Robots for kids, Morgan Kaufman, 2000.