Computer Mechatronics:
A Radical Approach to Mechatronics Education

Martin Nilsson

SICS, POB 1263, 164 29 Kista, Sweden; Lund University; Mälardalen University.
e-mail: from.mekatronikmote2005@drnil.com

Abstract

This paper describes some distinguishing features of a course on mechatronics, based on computer science. We propose a teaching approach called Controlled Problem-Based Learning (CPBL). We have applied this method on three generations (2003-2005) of mainly fourth-year undergraduate students at Lund University (LTH). Although students found the course difficult, there were **no** dropouts, and **all** students attended the examination 2005.

1 Are students really more stupid nowadays?

Mechatronics is recognized as an important university subject, but there are a number of problems associated with teaching this subject. For instance, one often hears complaints that student knowledge has deteriorated to the point where it has become difficult to teach engineering subjects such as mechatronics. This is perhaps true for physics and mathematics knowledge. However, students are certainly not more stupid now, and in the field of computer science, we believe the knowledge level widely surpasses that of mathematics ever held. Another problem is that mechatronics courses tend to become expensive in terms of materials and teaching. Expensive equipment is necessary, and lab experiments are teacher-intensive.

In this paper, we introduce a novel, radical approach to teaching mechatronics, which we developed for a course on Computer Mechatronics at Lund University, Faculty of Engineering (LTH). We have now taught this course for three years to 3rd and 4th year undergraduate students in computer science and electrical engineering, as well as to graduate students in computer science and automatic control. Although the course has received a reputation as demanding, during 2005, **none** of the 14 students who joined the course dropped out. **All** students participated in the final exam. This paper summarizes some of the distinguishing features of the course.

The inspiration for this work has many origins. The first main source is the pioneering work on PBL methods in mechatronics education developed by Professors Mats Hansson and Jan Wikander's group at DAMEK, KTH, already in the early 1980ies [1]. The second main source is the advanced work on experimental robotics by the group led by Professors Hirochika Inoue and Masayuki Inaba at the Department of Mechano-Informatics at the University of Tokyo in Japan [2]. The third main source is the astounding, popular subculture in Japan in the field of mechatronics.

2 Fundamental ideas of Computer Mechatronics

2.1 *Controlled* Problem-Based Learning

We believe that classical problem-based learning (PBL) gives students too much freedom, at least initially. According to our experience, excessive freedom tends to overwhelm students, so we have developed a variant of PBL we call **Controlled Problem-Based Learning** (CPBL). Initially, students are guided and controlled quite strictly. As they gain confidence, they are gradually let loose. This is particularly important since they start their work around their own PC, which is the most fragile and expensive tool. All the features of Computer Mechatronics are adapted to the CPBL framework, and we describe the most significant of them in subsections below.

2.2 Base in Computer Science

Usually, courses in mechatronics assume that students have a background in mechanical engineering, or occasionally, electrical engineering. Instead, Computer Mechatronics assumes a solid programming background. We have found it is considerably easier for a programmer to learn electronics that it is for an electronics expert to learn programming. A base in computer science is a powerful tool, which many students already possess when they enter the university. This is an asset that should not be wasted but taken advantage of!

2.3 Smoothing out the right obstacles

Designing the core functionality of a system is easy. Students are frustrated by all the peripheral, obnoxious difficulties: What is a good power supply? How can I find and buy cheap parts? Where can I find documentation? How do I find a free C-compiler? What is a good communication protocol for bit banging? How can I protect my PC from short circuits? What is a basic set of hardware tools? *These* are the difficulties that hinder the students' progression most, and *this* is where the teaching effort needs to be spent. In comparison, the features of a microprocessor are nearly trivial. In fact, the course leaves the microprocessor to the student as manual reading homework (and it works!).

2.4 Students own and keep *all* their equipment

Students become much more careful and motivated if they own and are allowed to keep their equipment after the course. A holy principle of the course is that students should be able to continue in "tangential direction" after the course, using the same hardware and software, perhaps even starting up their own, small company. This excludes all software with academic licenses, and expensive equipment such as oscilloscopes. There isn't space to describe precisely *how* in this paper, but yes, it can be done [3]. All project work and experiments are designed so they can be performed at home, at the student's own pace. During the lectures, only an ordinary classroom is used. No special laboratory equipment is used, besides what students take to the classroom. The only extra equipment is two PCs, and extension cables providing power sockets for every student.

2.5 Overall emphasis on debugging

Anyone can build a robot. Not everyone can make it work. Mechatronics is primarily about **debugging** - making it work. Through the entire course, students are reminded that they *must design for debugging*. However fancy the system, if it doesn't work, it is just crap. Projects that don't work fail the exam. Period.

2.6 Group work forbidden – individual examination

The examination is strictly individual, and every student must be able to solve problems in all subfields, independently and on their own. *Group work* runs counter to this idea. However, *cooperation* is encouraged, except during the exam. During the course, every student builds a small mechatronic system, including opto-isolated host PC to microcontroller communication, at least one actuator, at least one sensor, and a closed feedback loop. The student must first demonstrate that the system works. Then, the student has to leave the room, while the teacher breaks the system. The student will *only* pass if he/she is able to repair it. This examination makes it difficult to cheat. Students can certainly copy a circuit from e.g. the Internet, but if they can't fully understand it, they will not be able to debug it. Actually, we *encourage* students to roam Internet or any other source for ideas and solutions.

3 Results and Conclusions

After three years of tuning, the course appears to work well. Students consider the course heavy work, but worthwhile. The course has also been highly successful in terms of the number of students following through all the way to the exam. A pleasant surprise is that the students appear to have accepted the concept whole-heartedly. There have been no complaints on the lecture room being crowded, or that students have to pay for part of the equipment. The students appear to consider the examination fair and adequate. We are continuing to develop the course as part of a larger framework, the **OpenMechatronics** platform [3]. This is a collection of methods for rapid prototyping in mechatronics suitable not only for students, but also for researchers, inventors, or anyone working with mechatronics on a severely limited budget.

4 Acknowledgments

We are grateful to Tekn. Dr. Klas Nilsson from the Department of Computer Science at LTH, who paved the way through the university administration for arranging the course. It would never have been approved had it not been for someone with special powers. Klas believed in the concept from the very start, and has been a great sparring partner. We are also grateful to all the enthusiastic students, who contributed with a never-ending stream of ideas and suggestions for improvements, especially to Tekn. Lic. Sven Gestegård Robertz and Tekn. Lic. Anders Nilsson, who passed the course with flying colours in 2003, and who have been teaching assistants in the course since 2004. We also want to thank Prof. Lars Asplund from Mälardalen University for many stimulating discussions.

References

[1] Hansson, Mats and Wikander, Jan: *Personal communication*. Dept. of Machine Design, KTH, Stockholm, Sweden. (1990-2005).
[2] Inoue, Hirochika and Inaba, Masayuki: *Personal communication*. Dept. of Mechano-Informatics, The University of Tokyo, Japan. (1984-2005).
[3] Nilsson, M.: *OpenMechatronics*. http://www.OpenMechatronics.org (2005-09-15).