A Development Platform for Mobile Robots

Jessica Howe

Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

The Problem: In robotics today there is a lack of choices when looking for robotic systems to be used in teaching, research, and experimentation. At MIT seniors and graduate students can enroll in 6.836, Embodied Intelligence. This course emphasizes the role of situatedness and embodiment in building intelligent robots, and encourages students to experiment with real robots. Unfortunately no platform exists that adequately meets these needs in terms of morphological flexibility, sensor and actuator expansion, cost, and time available to modify.

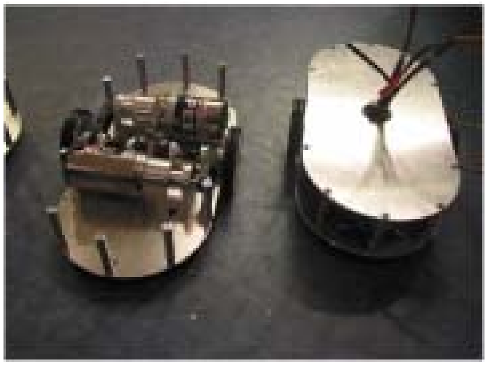

Figure 1: Wheeled robots in the process of being built

Motivation: When students build introductory robots, the Lego Mindstorm kit [5] is a common choice because of its simple construction scheme, straightforward simplistic programming language, and simple use of motors and actuators. At the same time it is fairly limiting in how complex a robot can be built. The programming language is not sufficiently expressive, there are few types of sensors and actuators provided by the kit, and the sole building unit is the Lego block. The students must be prepared to purchase such a kit on their own rather than using a resource provided by MIT.

The Handy Board [6] is also favored by students because of the ability to program in C and the ability to easily link to a variety of sensors and actuators. There is a limit to the number of sensors and actuators that may be used and, like the Lego kit, it is expensive.

For research a favorite is the Kephera robot [7]. Kephera is a small self contained wheeled robot complete with processor and basic sensors. Unfortunately, though a programming language is provided, there is no additional higher-level software support particular to robotic development, such as Behavior Language [2]. This platform is prohibitively expensive for a student to buy. It is difficult to situate within natural environments and it is usually contained within a man-made environment of flat floors and easily distinguishable walls. Its size also makes it difficult to add new components, so most people just use the sensors and actuators provided with the system.

What students need is a software architecture that supplies a flexible and powerful abstraction particular to robot programming. This software should run on a hardware architecture that supports fast processing for robot behavioral control, adequate networking, peripheral interface chips, PCBs for controlling actuators and allocating sensors. With this toolkit students will be able to build the robot they want and design it to accomplish sophisticated tasks in a reasonable amount of time.

Previous Work: Within this group at MIT we have developed Behavior Language [2], L, and used more general purpose languages such as C and LISP to program robots. We have built behavior based robots that have legs [1] as well as wheels, social robots that are active vision platforms [4], and developed multiple hardware architectures [3].

Approach: We aim to bridge the gap in available robotic systems by building an educational platform that contains a stable framework for building, developing, and implementing various robotic designs while allowing flexibility in morphological design and components. Our motor and sensor boards will be using pic 16F876 processors while the main computation will be done on a commercial Rabbit 2300 processor. Our design allows for stacking of boards to suit the particular quantity of components desired for a given project.

The programming language to be used is CREAL and is implemented with a compiler written in Emacs Lisp. The compiler is able to be run anywhere that runs Emacs, and comes complete with a back end that presently targets the Rabbit. Back ends may also be written to target other types of processors.

Impact: This educational package will facilitate student projects and encourage the building of robots in class settings. It will also be a platform on which multiple types of research may be done. There is flexibility in morphological form, sensor and actuator quantities and types, and design architectures. This platform will help to fill the gap that we have seen existing in systems available today.

Future Work: Our goal is to give students a development toolkit complete with detailed examples of how to construct multiple types of robots with varying body types and brain designs using this hardware and software architecture. We will provide a processor board for the Rabbit, a power board, motor and sensor control boards, and drivers for multiple types of sensors and actuators. The programming support will include the programming language CREAL, the CREAL compiler, example code, and tutorials to show how to implement various robot control architectures.

We intend to have a public online release of our material, in which we will release the hardware design, accessory drivers, the programming language CREAL, the CREAL compiler, example code, and tutorials to facilitate robot construction.

References:

[1] Rodney A. Brooks. A robot that walks; emergent behaviors from a carefully evolved network. *Nerual Computation 1:2*, pages 253–262, 1989.

[2] Rodney A. Brooks. The behavior language: Users guide. Technical Report 1227, MIT Artificial Intelligence Laboratory, April 1990.

[3] Rodney A. Brooks. Intelligence without reason. In John Myopoulos and Ray Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia, 1991. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

[4] Rodney Brooks Bryan Adams, Cynthia Breazeal and Brian Scassellati. Humanoid robots: A new kind of tool. *IEEE Intelligent Systems*, 15(4):25–31, July / August 2000.

[5] Mario Ferrari. *Building Robots with Lego Mindstorms: The Ultimate Tool for Mindstorm Maniacs.* Syngress Media Inc., 2001.

[6] Fred G. Martin. *The Handy Board Technical Reference.* http://handyboard.com/techdocs/, November 2000.

[7] K Team. *Kephera II user manual.* http://www.k-team.com/download/kephera.html, March 2002.