# RCUBE - a Multipurpose Platform for Intelligent Autonomous Systems

I. Boersch, J. Heinsohn, H. Loose, K.-U. Mrkor

*Department of Informatics and Media,*
*University of Applied Sciences Brandenburg / Germany,*
*PSF 21 32, 14737 Brandenburg an der Havel,*
*Tel.: (0 33 81) 35 54 28, Fax: (0 33 81) 35 54 99,*
*E-Mail: loose@fh-brandenburg.de*

## Abstract

*Since 1995 autonomous, intelligent or mobile robot systems have been investigated and included in the education process. Using commercial and/or open source platforms like LEGO-techniques with the 6.270-Board [2], the EyeBot controller [8] or the PIONEER robots [5] various aspects like mobility, controllability, sensor fusion, line or object recognition, navigation and control strategies were taken into account. The limited abilities of all these systems in computing power or image processing required the development of a new hardware platform called RCUBE [9] combining the features of a typical actuator-sensor-input-output-board with computing power and image recognition capabilities. The system will be suitable as a compact platform for universities, small enterprises and private developers.*

## 1. Introduction

The fascination of autonomous, intelligent or mobile robot systems has continued since the famous science fiction of Karel Capek and Isaac Asimov. In this period the American and European Mars missions including the activities of the robots Mars Rover Spirit and the robot Beagle 2 on the surface of the red planet can be obtained. The success and the failure are discussed.

Autonomous, intelligent or mobile robot systems have been investigated and developed at high specialised laboratories in research institutes, universities and companies. A number of service or personal robots have found their way into our world. They can be met in big halls cleaning the floor or the windows, in parking houses watching for troubles, in flats accompanying lonely people or in children's rooms collecting the toys.

Various principles of mobility from wheeled over legged up to flying mechanisms, different sensors and sensor systems from simple switches over ultra sonar up to stereo cameras, several hard- and software architectures from standby computer systems up to high integrated embedded microprocessor platforms were investigated, built in changing configurations and tested in ideal, industrial or hazardous environments. The humanoid robots or the Mars landing apparatuses are highlights of the beginning 21 century.

Consequently, Robotics - the scientific discipline about the understanding and the development of robots – was included into Mechanical, Electrical, Mechatronic Engineering or Computer Sciences Bachelor and Master courses.

In this paper examples of using autonomous, intelligent or mobile robots in the education process at the University of Applied Sciences Brandenburg are presented. The main problems were the limited abilities of the employed systems: not enough connectors to sensors and actuators, the low computing performance, the high power consumption, the lack of onboard image recognition capability and energy autonomy.

Most of the systems successfully established on the market have been developed by universities and then hived off as a product (see Eyebot, Handyboard, RCX, Real Robots).

The new developed hardware platform RCUBE combining the features of a typical actuator-sensor-input-output-board with computing power and image recognition capabilities is flexible, adaptable to an application case, easy to program and cost-effective.

## 2. Mobile LEGO-robots

In 1996 the concept of Learning Engineering by Designing LEGO Robots described in [1], [2] was introduced. The robots are composed of mechanical elements of the LEGO constructor system, of wheels and gears, of DC-and servo motors, of a variety of cheap sensors like switches, photo elements, infrared transmitters and receivers, batteries and a 6.270-Board with analogous and digital inputs & outputs (Fig. 1). The action control algorithms - the intelligence of the robot - are implemented at a usual personal computer using the high level “Interactive C” programming language [3].

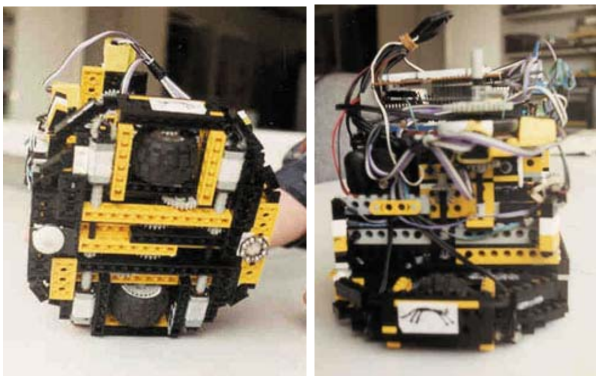

**Figure 1.** A typical LEGO-robot

In short time projects similar to real once where new products are developed in limited time with limited budgets using commercial subsystems teams of students of different specialisations have to create mobile and autonomous robots for given tasks contrary in their demands. For examples:

- *Minotaurus-Project*: Starting from the point A each of the robots has to find the Minotaurus placed in the target area and marked with an infrared sender and behead it (10 credits). Crossing the points B, C, D, E and F another 3 credits can be achieved.

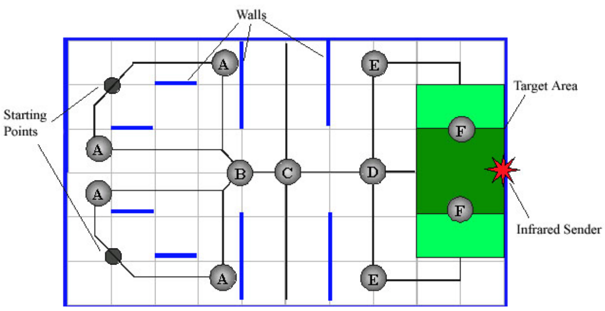


**Figure 2.** Area of the Minotaurus-Project

Different strategies solving the task are possible: 1. Line following (save strategy): The way is long and the speed is low, but the intermediate points will be crossed.
2. Directly towards the target: Ignoring the lines and moving immediately in the direction of the infrared sender up to the wall in front of the Minotaurus it can be beheaded with a long sword.

- *Hunting of a Robot-Hare*: 3 robots are included in the competition, one - the hare – is running away from the others – the hunters. Each of them has to localise the other both. Collision between the hunters should be avoided. The competition ends if there is a hunter clashed with the hare or if the time is over. Different tasks has to be solved: localisation of the other robots, collision avoidance, hunting strategy.

Beside the construction of a robust mobile mechanism armed with photo or infrared sensors and built for the given task the attention was concentrated on the control and navigation algorithms.

From the point of view of autonomy and intelligence the problems appearing during these projects were caused in the changing power of the accumulators onboard the robot and the lack of memory (32 kByte RAM) of the 6.270 board. So it was impossible to implement a functional library for various sensors and motion commands.

## 3. Genetic programming of the “servorium”

Aspects of the information technology and artificial intelligence are investigated in this project. There is no alternative to the idea to teach a “servorium” (Fig. 3) composed by randomly connected links, hinges and servo motors to achieve a desired motion [10]. An Eyebot-controller by Joker Robotics [8] is the brain of this autonomous system.

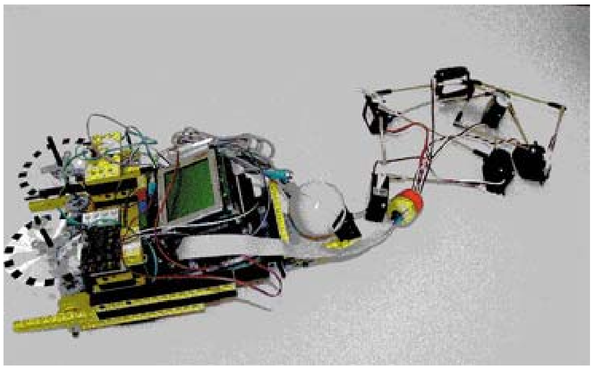

**Figure 3.** “Servorium” with odometer and control unit

The well known method of genetic programming was used to generate program sequences on the base of given subprograms. A population of 100 individual programs evolved guided by a fitness function. Generated programs got two seconds computation time on the real “servorium” to prove their performance in a cyclic execution. Performance of an individual was defined as a weighted sum of the movement measurements provided by odometric sensors and properties useful to direct the evolution in the desired direction. Because of the long duration of the experiments caused by coupling of fitness measure to real time and real bodies fitness components were integrated. Examples for fitness components are the proportion of servo and wait instructions or the complexity of an individual.

The performance of the "servorium" was the selection criteria for participation in basic genetic operations of crossover, mutation and reproduction. During artificial evolution some unwanted coevolutionary effects were observed in the sense of parasitic programs collecting fitness points using body configurations of other programs without any payback. After long series of experiments a program sequence was found moving the "servorium" and pulling its brain with a velocity of 3 m/h.

The experiment shows the applicability of genetic programming for automatic generation of motion patterns adapted to a complex morphology.

## 4. Energy autonomy

Energy autonomy of a robot could be considered like a precondition for surviving and lifelong learning. Service or personal robots working in a human environment can be recharged using electric power. Since 2002 different aspects of energy autonomy - the properties of control architectures, navigation strategies and docking concepts - were investigated.

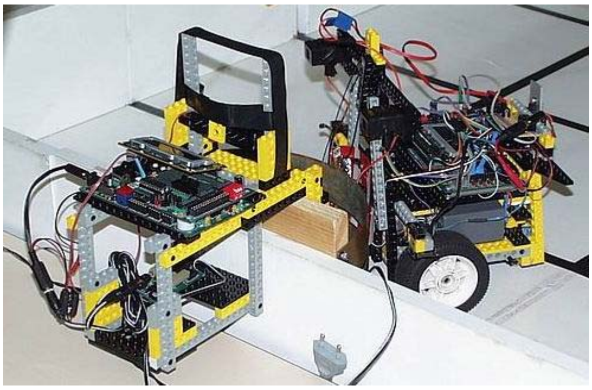

**Figure 4.** Robots at the base station

The robot is capable to dock at the base stations and recharge itself. A charge circuit board for autonomous 6.270-robots was developed and tested. Navigation and localisation of the base station is performed with an rotating laser beam and infrared beacons. The control architecture is a modified subsumption system, whose behaviours were provided with expectation horizons. These behaviours guarantee escalations, stress in more abstract activities and randomly selected actions at the top level, so that dead locks in the motion of the robot are avoided.

The robots survived self-governed more than 30 days in the laboratory until their external shut-down.

## 5. The Pioneer II – robots

Pioneer II [5] is a family of two-wheel-drive robots. They are small, intelligent. The architecture was originally developed by Kurt Konolige, containing all of the basic components for sensing and navigation in a real-world environment, including battery power, drive motors and wheels, position / speed encoders, and integrated sensors and accessories. They are all managed via an onboard micro controller and mobile-robot server software. It includes a complete addressable I/O bus for up to 16 devices and two RS-232 serial ports, one for communication between the robot server and a client computer and the other for the PTZ Robotic Camera.

**Figure 5.** One robot follows the other

Pioneer II comes with the Saphira client development suite with Colbert and the Pioneer simulator. It lets enable several built-in robotic behaviours including collision avoidance, image recognition, and self-navigation.

Since 1998 the Pioneer II platform is used as the experimental base in practical exercises, in various research activities and diploma theses. Basic features of motion planning, programming and control, of using sensors and vision systems, of different navigation schemes or mobility and autonomy are studied. In student experiments simple applications are realized using the hard- and software features of the robot system, including user defined behaviours and activities [7].

### 5.1.Get a bottle from outside of the lab

The standard experiment (Fig. 6) done at the end of the course consists of a sequence of subtasks:

- move along the wall up to a door is detected,
- move throw the door stopping in front of the glass wall,
- move along the corridor avoiding collision up to detecting bottles on the right hand side,
- move near to the bottle with the blue label, grasp and heave it,
- return to the starting point.

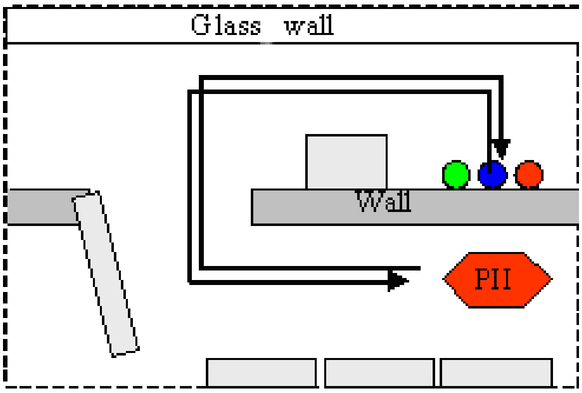


**Figure 6.** Map to the task "Get the bottle"

For the solution of this task the encoders, the sonar system, the robot vision, the internal map and the built in intelligence of Saphira can be used. An algorithm for the detection of the bottles and, especially the bottle with the blue cap must be integrated.

### 5.2. Vision based target detection and following

The main goal was to build a complete and reasonable fast visual module to localize moving objects and follow them with autonomous robot PIONEER II equipped with an active on-board camera. The locating is realized in real-time in laboratory circumstances on the base of color pictures obtained from the CCD camera. Color histograms are used to estimate the distributions of colors in a patch of objects in the presence of varying lighting conditions. The developed system is able to track a target at approximately rate of 6Hz on a standard laptop computer with no special hardware. Some experiments showed that proposed solution (soft- and hardware) works well (Fig. 5). All algorithms are designed with hardware implementation in mind.

The problems solving advanced tasks with the PIONEER platform are located in the vision system especially in unreliable video and control radio links and the difficulties to adapt control algorithms inside the Pioneer and Saphira operational system.

## 6. RCUBE - a hardware platform

Small autonomous systems especially mobile robots are a growing application area of intelligent systems. The economic commission of the UNO forecasts a boom concerning service robots (for instance for medical, cleaning, security purposes) and personal robots (domestic and entertainment robots) till 2005 [11].

For reasons of cost, time and capacity small enterprises and universities normally use widely spread basic blocks of microcontrollers (such as Basic Stamp, Handy board, C-Control) not suitable for real robotic applications including manifold sensor equipment (Table 1). For small and middle size autonomous systems the most important problems are the limited number of connections for sensors and actuators, the low computing performance and high power consumption.

Another drawback is the lack of an autonomous image recognition capability. Image recognition is realized on mobile robots either via radio linked hardware or by using relatively power consuming components.

In 2001 the project to develop a new intelligent platform for autonomous systems with vision capabilities sponsored by the Ministry of Sciences and Culture of the state Brandenburg/ Germany was started.

### 6.1. Requirements and results

The technical requirements to robot platform were identified as follows:

- image recognition onboard
- low power consumption
- high computing performance
- program persistence
- software open source
- independent from any radio links
- 32 ports for digital and analog sensors
- 12 ports for actuators, servos, DC-motors, bulbs
- small and modular
- cost effective

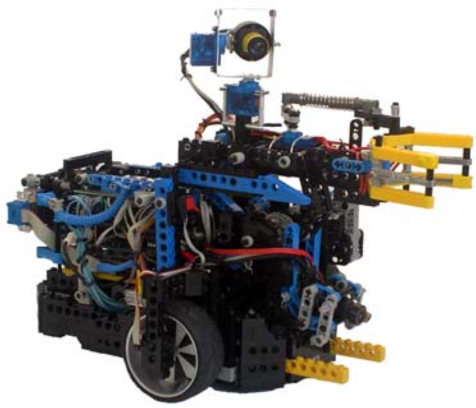

**Figure 7.** RCUBE robot with vision capabilities

**Table 1.** Overview about controlling parts for reactive robots
(Costs in Euro of one robot and time in month to build a simple robot are estimated)

| System | Source | Sensors | Actuators | Costs | Time |
|---|---|---|---|---|---|
| Eyebot | JokerRobotics | Many | many | 700 | 3 |
| 6.270-Board | self assembly | Many | many | 600 | 3 |
| Handyboard | self assembly | Few | many | 600 | 3 |
| C-Control | Conrad electronics | Few | very limited | 450 | 3 |
| Basic-Stamp | Parallax | Few | very limited | 350 | 2 |
| RCX | LEGO | very limited | few | 260 | 1 |
| Mobile-Robot | Fischertechnik | very limited | very limited | 300 | 1 |
| Real Robots | Eaglemoss | Few | very limited | 200 | 1 |

The RCUBE platform should be available in various configurations dependent on the demands of the application, for example (Fig.7) like a:

- cost-effective platform for reactive robots and private developers,
- powerful platform for intelligent robots with image processing capabilities suitable for research, development and education in the field of service robotics.

The architecture RCUBE consists of three of 3 ready-to-program hardware modules with a basic software layer:

- CPU board
- VIO board (video input/output unit)
- AKSEN board (actuator/sensor unit)

They can be combined in any number and connected via a field bus with 1 MBit bandwidth. Various interesting configuration and application are imaginable (Fig. 8).

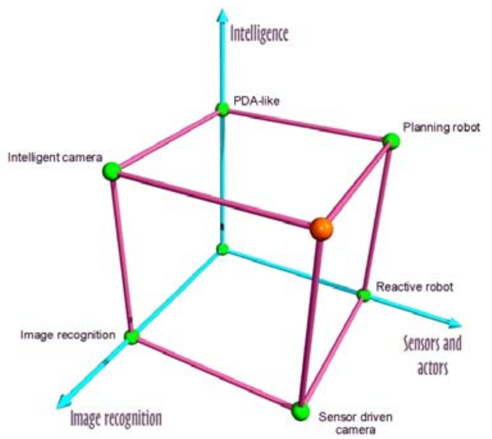


**Figure 8.** Configuration space of the RCUBE

## 6.2. CPU board

The CPU Board provides the location of computing power and therefore the intelligence of the system. A StrongARM processor with 200 MHz running ARM-Linux is its core component. It is programmable like any Linux machine using gcc. Libraries for communication with other modules are provided. Downloaded programs can be flashed to a nonvolatile memory.

For logging purposes or for receiving high level orders, a bluetooth connection with 115 KBit bandwidth is optional available. This provides a comfortable way for development and debugging.

## 6.3. VIO board

The VIO Board is a standalone image digitizing and recognition solution based on StrongARM and Linux. Up to 4 standard PAL cameras (for instance small and cheap CMOS cameras) can be connected to the board (multiplexed, 60 ms for a camera change).

The image stream is digitized with 25 fps (CCIR601 CIF 2:1, means 320x280x24) or 10 fps (CCIR601 full resolution 1:1). The image stream is provided in RGB or YCbCr format. Image recognition algorithms are programmable in C or C++ and connected to the image stream via a Video4Linux2 interface.

To support the development a video output of the input stream and the image analyzing results, such as colored segments, is available in real time.

## 6.4. AKSEN board

The AKSEN board provides connections for simple peripheral devices in a robotic environment, such as actuators and sensors. It was designed with two main purposes in mind:

- standalone controller for reactive robots,
- sensor and actuator server in a RCUBE system.

Main features of AKSEN controller board are

- 15 analog inputs (suitable for sensors for light, infrared, obstacles, lines, voltage etc.)
- 16 digital in-/outputs (freely configurable)
- 4 motor drivers for small DC-motors up to 1A (variable in direction and rotational speed)
- 4 driver (for infrared senders or bulbs)
- 3 servo outputs (extensible by software)
- 1 output for modulated infrared (localization)

- 3 fast encoder inputs (for instance odometry)
- 4 dipswitches
- RS232 interface,
- CAN interface (optional)
- LCD display (optional)
- bluetooth connection (optional)

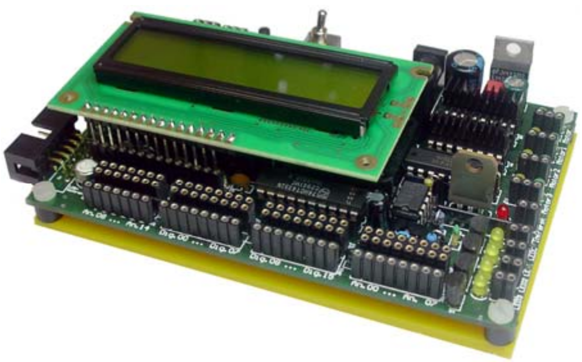

**Figure 9.** The AKSEN-Board

It is programmable like any Linux or Windows machine using freeware c compiler like sdcc.

## 7. Conclusion

Based on the experiences in investigating and teaching robotics for a long time and the practical success and failure with the 6.270 board, the EyeBot controller and the PIONEER II platform the RCUBE system was developed consisting of three main modules:

- CPU-board,
- VIO-board and
- AKSEN-board.

Any module is designed for low power consumption and interoperability and can be used standalone or combined in any number collaborating and communicating via CAN-bus or bluetooth with other modules or hosts. Each module survives a power blackout (or a long inactive period) without losing data and reboots after power reoccurs.

The CPU-Board based on a StrongARM processor (200 MHz, 32 MB RAM, ARM-Linux) provides necessary computing performance allowing the implementation of algorithms for task planning, navigation or skill learning.

The VIO-board allows really autonomous image processing for up to four cameras independent of radio links and base stations.

The AKSEN-board provides connections to usual devices of small robots (15 analogous inputs, 16 digital in-/outputs, 4 motor drivers for small DC-motors, 4 drivers for infrared senders or bulbs, 3 servo outputs).

The flexible configuration of a RCUBE system provides interesting application areas:

- reactive mobile robot (only AKSEN-boards),
- reactive mobile seeing robots (AKSEN-board and VIO-board),
- intelligent cameras (CPU-board and VIO-board),
- planning robots (CPU-Board and AKSEN-board)
- small service or personal robots (all three or more boards).

The system will be suitable as a research and education platform for universities, a basis for industrial applications and for private developers of robots.

## 8. References


[1] F. G. Martin, *Circuits to Control: Learning Engineering by Designing LEGO Robots,* Massachusetts Institute of Technology, 1994

[2] A. M. Flynn, J. L Jones, Mobile *Roboter*, Addison-Wesley, 1996.

[3] I. Boersch, J. Heinsohn , H. Loose, *RobotBuildingLab – praktische Mechatronik in der Ausbildung*, 2. Polnisch-Deutscher Workshop Werkzeuge der Mechatronik, 14./15.05.1998, Ilmenau, Proceedings, pp. 23-33.

[4] B. Kwolek, I. Boersch, H. Loose, *Verfolgung eines dynamischen Ziels mit dem autonomen mobilen Roboter PIONEER II*, 4$^{th}$ Brandenburg Workshop of Mechatronics, 2000, Brandenburg.

[5] *The PIONEER 2 Mobile Robot Operation Manual,* ActivMEDIA ROBOTICS, 1999, www.ActivMedia.com.

[6] H. Loose, *The walking robot "SimengDolores" – a project in education in Mechatronics*, 2$^{nd}$ European Workshop on Education in Mechatronics, (2001), Kiel, Proceedings, pp. 265-270.

[7] H. Loose, *Wheeled and Legged Robots in Collegiate Education*, International Colloquium on Autonomous and Mobile Systems, Magdeburg, June 25-26, 2002, pp. 179-182.

[8] Th. Bräunl, *Eyebot – Online Documentation,* http://www.ee.uwa.edu.au/~braunl/eyebot, 2002

[9] I. Boersch, J. Heinsohn , H. Loose, K.-U. Mrkor, *RCUBE A Platform for Intelligent Autonomous Systems*, IEEE Conference on Industrial Technology 2003, Proceedings, pp. 203-206, Maribor/Slowenien, 10.-12.12.2003.

[10] I. Boersch, *Genetisches Programmieren einfacher Roboterfähigkeiten*, *4$^{th}$ Brandenburg Workshop of Mechatronics,* 2000, Brandenburg.

[11] ECONOMIC COMMISSION FOR EUROPE (UNECE). *World Robotics - Statistics, Market Analysis, Forecasts, Case Studies and Profitability of Robot Investment.* Co-authored by the International Federation of Robotics, 2002.