robuLAB10 indoor research and service robot

**Introduction**

robuLAB10 is a multi-purpose mobile robot designed for various indoor applications, such as research and education, tele-presence, assistance to people staying at home. robuLAB10 is a generic mobile platform, on top of which standard or customized modules can be installed. Using an open control system called robuBOX™, based on Microsoft® Robotics Developer Studio, the robuLAB10 comes with basic robotic functions which can be completed by software developers. A licensing program is available to application developers willing to commercialize their own service robots based on the robuLAB10 platform.

**Applications**

- Robotics research and education
- Assistant robot at home
- Surveillance
- Tele-presence
- Entertainment/Education
- Cleaning

**Features**

- Remote control by wireless gamepad
- Teaching by showing missions
- Obstacle detection
- Compliant with EC 98/37 Machinery Directive

**Controlled by robuBOX®**

The core software of robuLAB10 provides all the basic functions needed for intelligent service robots: service orchestration, management and abstraction of hardware and sensors, navigation algorithms, anti-collision, path teaching-and-repeating, fleet supervision, remote control ... Using Microsoft® Robotics Developer Studio, these basic services can be enhanced or completed by customers.

**Main technical features***

<table>
<thead>
<tr>
<th>&gt; Platform</th>
<th>&gt; Control system</th>
<th>&gt; Sensors</th>
</tr>
</thead>
<tbody>
<tr>
<td>dimensions: 450 x 400 x 243 (L x W x H)</td>
<td>Pentium M 1.6 Ghz</td>
<td>Odometry</td>
</tr>
<tr>
<td>payload: 30 kg</td>
<td>512 Mb - 2 Gb Compact Flash</td>
<td>Laser range finder</td>
</tr>
<tr>
<td>weight: 23 kg, including batteries</td>
<td>Ethernet and USB connectors</td>
<td>Color camera CMUCam3</td>
</tr>
<tr>
<td>max speed: 3 m/s (1 m/s software limitation)</td>
<td>CAN, i²C, RS232</td>
<td>Ultrasonic sensors (rear and front)</td>
</tr>
<tr>
<td>4 lead sealed batteries (12 V, 9 Ah)</td>
<td>Wifi</td>
<td></td>
</tr>
</tbody>
</table>

*specs are subject to change at any time*
Extensions

Extension compartment: this compartment expands the space available inside the robot. It comes with 4 extra-batteries, to double the autonomy, as well as USB and Ethernet hubs. It can be used to install other equipments, inside or on top, such as an additional PC, transmitters, extra sensors ...

Docking station: to recharge batteries. An automatic docking function is available in the robot, based on laser navigation. The docking station works in 110 and 220 VAC.

Healthcare module: this module, including an extension compartment, is used for healthcare and tele-presence applications. It comes with a pan-tilt camera on top, and a panel PC used as Man Machine Interface.

6-dof manipulator: installed on top of the extension compartment, this versatile manipulator can be used to manipulate objects on a table. It has a payload up to 500g and an accuracy of 0,01° in cartesian coordinates. The manipulator comes with its own controller, and can be controlled through an ethernet connection.

You need a customized robot? Contact us ...
A distributed and open hardware architecture

The hardware architecture of robuLAB\textsuperscript{10} is based on one or several communicating PCs (one in the platform, other(s) in the extension modules). They all have expansion capabilities, run Windows XPe and robuBOX\textsuperscript{®}, and application software is developed using Microsoft\textsuperscript{®} Robotics Developer Studio distributed capabilities.
Programming tools and methods

**robuBOX®**: this is the core software under the hood of the robuLAB®. The robuBOX, fully based on Microsoft® Robotics Developer Studio, has been designed like a “Universal Robot Engine” to make easier and faster software developments using MRDS, thanks to a high level of hardware abstraction. It thus allows their portability to forthcoming new generations of robots. A robuBOX licensing programme is available to robot manufacturers willing to adopt this technology. Each robot is delivered with services and tools available “out of the box”.

> **robuBOX core**: this a set of MRDS services and standard data types: developers can use these services to build their own applications. It provides services such as hardware drivers for common hardware components, high-level services for sensor data processing, data types for representing positions, trajectories, localization data… The robuBOX core also provides a basic control architecture which “orchestrates” all services dealing with robot motion control.

> **UDP Server**: this server allows to take control of robot from a Linux platform or a non MRDS piece of code. This UDP server which runs in the robuBOX, provides all necessary information to perform a high level control of the robot: it allows to receive motion commands (velocity and rotation rate) and provides the odometry of the robot (dX, dY, dTheta), its localization (X,Y,theta), its battery level, the bumpers status and ultrasounds data. The server also offers a watchdog mechanism to stop the robot in case of loss of communication.

> **LUA Scripting language support (available Q4 2008)**: allows development of simple applications with an easy-to-use and popular scripting language. The C# methods and data types of main services can be used from the LUA scripting language, to be configured or extended. LUA source code samples come with the Robubox and show how to use the main services from LUA. It also come with a simple LUA source code editor (with syntax highlighting), that can be used to load/save/edit/run LUA scripts.

> **Sample applications**: with the robuLAB®, applications such as fleet management, exploration, remote control are provided. Some source code examples can also be used as basis for custom developments.

> **Web interfaces for main services (available Q4 2008)**: the main high-level services come with a web interface that can be used to configure/debug/test individual services or composite services. It can be used to move the robot, display sensor values, tune the motor drives, follow trajectories… Each action generated with the web interface can be performed in another web service or application using simple URLs and XML data exchanges. This is particularly useful for remote applications like controlling the robot from a mobile phone.

> **3D model of the robot (available Q4 2008)**: “3D model” means an accurate geometric and physic model, ready to be used in the MRDS simulation services. Most “client services” that control the 3D model in the simulator comes with a sample GUI. This client service is running EXACTLY THE SAME code that would be used for controlling the real robot. Just change the IP address and you can control the real robot!

> **3rd party hardware and software**: the full compatibility with MRDS allows using any commercially available hardware or software pack which has been developed for MRDS. Interactions with 3rd party OS or software can be developed using the .NET functionalities as well as classical web services or UDP/TCP services.
SimplySim – Apartment Environment.

Environment description:

An apartment interior environment, with all the usual feature of a modern urban home: living room, kitchen, a bedroom and a small bathroom.

Each room is populated with numerous physical objects to make the environment more realistic and interactive. The physics shapes of the objects have been defined using the SimplyPhy technology, enabling advanced physics interactions.

This environment has been specifically designed to develop and test software for the home usage of robotics: security, cleaning, assistance...

Environment Facts:

- Environment Size : ~150 m²
- Environment fully compatible with Microsoft Robotics Studio 1.5 or Microsoft Robotics Developer Studio 2008.
- Advanced physics shape on most of the dynamic objects.
- Living room - kitchen
- Working room
- Bedroom
- Bathroom
- Terrace
- A 2D map of the environment