The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Introducing Wanda - A New Robot for Research, Education, and Arts

Alexander Kettler, Marc Szymanski, Jens Liedke and Heinz Wörn

Abstract— **A new autonomous mobile robot for research, education and Arts is presented. The robot features a rich variety of sensors and an extendable processing unit while still maintaining very small size (approx. 51 mm in diameter). The robot can be used in combination with a very versatile arena for robot experiments that allows to setup and execute experiments in an easy and automated fashion, which also includes automatic recharging of the robot. We will provide an overview of the basic modular hardware design of the robot including a detailed description of the cpu, power management, bus system, actuators and sensors. A special section will be dedicated to a detailed description of the communication capabilities of the robot. Furthermore we will present the software architecture which provides high-level access to the robots hardware.**

I. INTRODUCTION

Designing small mobile robots for research in autonomous mobile robotics, evolutionary robotics, swarm robotics, artificial intelligence, and artificial life has a long history and reaches back to the mid 90's with the design of the Khepera robot at the EPFL [7]. Since then several robots, like the Alice [3], Khepera II-III, Jasmine [10] or ePuck robot [8], to give just a few of them, have been designed each especially strong in parts of those topics.

However, sometimes it seems that there is still much effort to be invested by the researcher to perform experiments with those robots and to properly shape the environment for the robot. With the design of the Wanda robot we did not just concentrate our efforts on the robot, but also on the robot's environment. We believe that the combination of a handy robot together with a very versatile arena for robot experiments is a key issue in the prior mentioned robotic research areas. The environment and the robot must be capable of repeating experimental runs in a simple and automated fashion, where data collection is an important factor.

The Wanda autonomous mobile robot for research, education and Arts with its small size of 45 mm in height and a diameter of 51 mm is perfectly suited for experiments performed on a desktop or in small arenas.

The robot's sensor system has been especially selected for such desktop experiments and includes three gray-value floor sensors, a touch and RGB/ambient light sensor to the front, an 3D accelerometer, wheel encoders, battery monitors and two light sensors to the top. A real novelty is the interrupt based infra-red (IR) communication system, which is comprised of six sensors for distance measurement and communication equally distributed around the robot without the need for an ADC. The communication system has been designed with an eye on the performance in a large swarm ($\gg 100$). It implements a self-organising *Time Division Multiple Access* (TDMA) protocol. The robot's mobile platform is based on a differential drive with a velocity of up to 300 mm/s. A ring of five RGB LEDs is arranged around the robot.

Wanda is equipped with two 250 mAh lithium polymer (LiPoly) cells providing about 2.5 hours autonomy. Due to sliding contacts at the robot's bottom it is possible to recharge the robot during experiments. Recharging the robot takes less than two hours in total.

The robot's central processing unit is a LM3S1960 micro controller (μC) from Texas Instruments (TI) [13]. Additionally, the robot can be equipped with a BlueTechnix Blackfin Board which enables the robot to run Linux. However, the Blackfin's (BF) power consumption makes an extra 1000 mAh LiPoly cell backpacked to the robot indispensable.

The robot is also equipped with a radio communication system based on Atmel's® ZigBee IC. This ZigBee communication can be used for monitoring the robot, collecting experimental data or uploading MDL2ϵ controllers.

The combination of a rechargeable robot, together with ZigBee and the Optical Data transmission system for Micro robots (ODeM) makes it possible to create such a versatile environment as mentioned above. Repeated experiments can now be performed in an automated way.

The following sections will give a closer look on the whole embedded system and the underlying design concept.

II. WANDA'S MODULAR DESIGN CONCEPT

The key idea of the Wanda robot's design is to make the robot as modular as the Khepera and ePuck robot but also to include improved recharging capabilities of the Jasmine robot. The size of the robot should be as small as possible but not that small that it leads to problems with the design of the differential drive and odometry system as we were facing it during the design of the Jasmine. Neither should it increase the problem of costs and spare parts as it seems to be the case with the Alice robot. Therefore the size has been proportioned on the one hand to fit the size of available low cost micro DC motors equipped with a planetary gear and on the other hand to fit the size of available 250 mAh LiPoly cells taken from an iPod Shuffle (generation I). This led to

This work has been supported by the Karlsruhe Institute of Technology Collective Robotics Lab (KITCoRoL), the GRK1194 and the SYMRBION and REPLICATOR projects.

A.Kettler, M.Szymanski, J.Liedke and H.Wörn are with Faculty of Computer Science, Institute for Process Control and Robotics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany alexander.kettler@kit.edu, marc.szymanski@kit.edu, jens.liedke@kit.edu, heinz.woern@kit.edu

an optimal PCB size of 45 mm in diameter, which is only 12 mm larger than the Jasmine robot but 25 mm smaller than the ePuck. Together with the chassis of the robot, which is important for better reflection of infra-red light and also supports the incremental encoders by preventing noise through ambient light, the robot has a diameter of 51 mm in total.

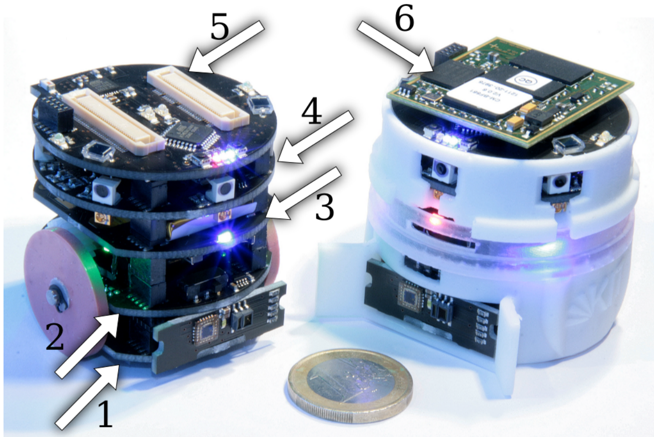

Fig. 1. The Wanda robot.

The Wanda robot's design has a layered, see Fig. 1, modular structure and can be built of three to six PCB layers. The single layers are from bottom to top:

1) **bottom board (BB):** floor sensors, accelerometer, USB jack for RS232 and JTAG via an FT2232D, LiPoly cell and battery management for recharging the robot via sliding contacts or USB, connector pads to RGB sensor board, power and recharging indicator LEDs.
2) **motor board (MB):** IR reflective wheel encoder sensors, power switch, 1 MB I^2C memory, motor control.
3) **auxiliary battery board (AUXB):** LiPoly cell plus battery management, recharging indicator LED, Micro SD-card slot, five RGB LEDs driven by a CAT9532.
4) **cortex controller board (CTB):** LM3S1960, IR communication, battery monitor ADC
5) **ODeM-RF board (OB):** ATMega168 for ODeM sensor control and ZigBee, AT68RF230 ZigBee IC, 8 KB I^2C memory, three status LEDs, two blue and two orange ultra bright LEDs, reset buttons and UART connectors.
6) **BlackFin board (BFB):** Bluetechnix CM-BF-561 Dual Core Blackfin board with ucLinux.

For a minimal functioning robot layers one, two, and four are sufficient. The typical arrangement includes layers one to five. The robot therefore has a height between 32–45 mm.

A. Micro Controllers

The selection of an appropriate micro controller is essential for the robot's usability. On the one hand, a too powerful μC with high power consumption reduces the robot's autonomy time. But on the other hand a μC with low power consumption usually has also low computational power. Another point that has to be considered from the software point of view is the driver support that comes with a μC. The effort of software implementation can be significantly reduced if a good driver library is available for a given μC. Here the cortex series from TI seems to be a good choice. All TI μCs are shipped with a very good drivers library in standard C. It has excellent computational power at an affordable power consumption.

From the very beginning, we decided that as many sensors as possible should be mostly independent from the μC. This means the μC should not be bothered too much with the analysis of series of incoming ADC data. We therefore selected a μC without on-board ADC but several communication busses, relatively large RAM and flash memory, a good number of 32 bit timers and several free general purpose input/output lines (GPIO) instead.

Our final choice for the robot is the LM3S1960 [13]. The complete sensor and actuator interaction with the LM3S1960 is explained in section II-C.

B. Power System

A good power supply is essential for a robots autonomy. Therefore we decided to equip the robot with one 250 mAh LiPoly cell which is always in the system and to additionally design a board with an additional auxiliary 250 mAh LiPoly cell. This has a major advantage: a power management based on the TPS2111 switches during runtime from the auxiliary battery to the steady battery if the auxiliary battery drops below approximately 3.4 V and backwards after a short periods as long as the auxiliary battery can still recover. This has the effect, as can be seen in Fig. 2, that the capacity of the auxiliary battery is used up extremely well. This way the batteries are discharged sequentially. However, recharging is done in parallel with one recharging IC for each cell. The robot is fully recharged at a charge-current of 150 mA within about two hours, see Fig. 3. This means that the robot has a longer autonomy than recharging time. The recharging current of 0.6C increases the LiPoly's lifetime.

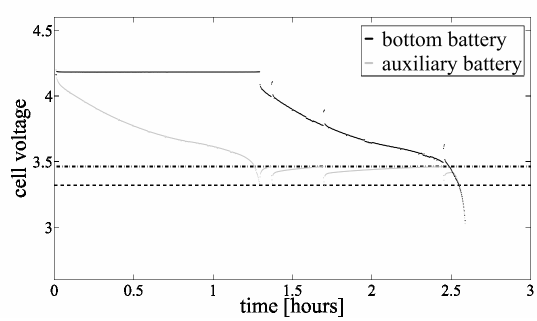

Fig. 2. Robot's discharging curve while running with a constant velocity of 50%. The dashed lines show the switching hysteresis of the TPS2111.

During recharging the power management system draws the energy for all electronics from the attached power supply and not from the batteries. The robot's capability of self-recharging is also essential for the design concept. The robots

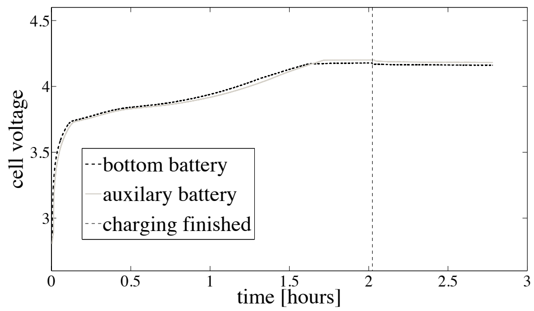

Fig. 3. Charging curve for the robot on the charging station.

can decide when to recharge or can be forced to recharge before the start of a new experiment by the automated arena system. Several types of recharging stations can be implemented which are based on for instance a checker board of alternating positive and negative polarity. We set up a simple recharging station based on two thin copper stripes on a black surface, which can be easily detected by the floor sensors. The recharging setup can be seen in the video [6]. To prevent damage from reversed polarity at the charge pads, the recharging system of the robot is equipped with a simple bridge rectifier.

C. Bus System

The robots bus system is divided into two parts as shown in Fig. 4. One goes from the CTB down to the AUXB, MB, and BB. The second goes up to the OB and BFB. Both include a separate SPI, I^2C, and UART bus, supported by several GPIOs used for interrupt handling and the configuration of the sensor system. The discrimination of those busses in two parts makes it easier to separate user designed boards from the parts of the robot that are important for the survivability of the robot.

One JTAG bus goes from the FT2232D via the CTB to the BFB and implements a JTAG chain for software debugging.

The bottom bus connects all sensors via I^2C with the cortex controller, the UART is connected to the FT2232D IC and is very important for accessing, configuring and debugging the robot. During a robot experiment the UART is not needed. The SPI is directly connected to the SD-card and not used by any other device.

The upper bus connects the ATMega on the OB with the Cortex and the BFB. The SPI can be configured with a dip switch to either connect all SPI devices to the bus or to just connect the CTB with the BFB and make a disjunct SPI bus between the ATMega and the AT86RF230. The slave select lines of the SPI bus are connected in a way that makes it possible for the BlackFin to function as SPI master and access all subsequent SPI devices. Alternatively, the Cortex can function as master and access all subsequent devices except the BF. This enables us to give full control over all SPI devices to the computationally strongest CPU currently in the system. A full ZigBee stack can therefore be implemented on either the Cortex or the BlackFin.

D. Actuators

The robot has two DC motors arranged as a differential drive. The wheels are made of the same PCB as the rest of the electronics but coated with silicon that increases the friction for a better grip. The wheel PCB implements also the encoder marks by gold coated copper pads surrounded with black finish.

For moving small objects the chassis has a passive gripper with enough space for e.g. a 2×2 LEGO® DUPLO brick.

As mentioned in the introduction the robot is surrounded with five RGB LEDs driven by a CAT9532, which can be perceived by the RGB sensor up to a distance of 30 cm. The light is dispersed by a transparent ring around the chassis. The CAT9532 can control each LED in one of four modes: on, off, PWM 0, and PWM 1. The PWM modes can be set up by selecting the PWM period and the duty cycle. This allows dimming or flickering of the LEDs without bothering the Cortex μC.

E. Sensors

The sensor system of the robot was designed with intend to make it very simple for the user to setup experiments for the robot even on a desktop without bothering too much about sensors and sensor data processing. From our experience it is often a very severe task on such small robots to discriminate different situations the robot is in. For example depending on the sensor system it could be very difficult to discriminate between a wall or a robot if only distance data is available. In this case a complex model has to be implemented to analyse the incoming data. This often leads to frustration before the experiment has even started.

To overcome such problems the Wanda robot has a rich set of sensors that can be used in different ways depending on the given problem. The floor sensors for example can be used either to do a simple line following in order to find a recharging station or to mark areas for different purposes like home area, food source, or even areas that should not be accessed by the robot. A setup on a desktop without any walls just painted with a black marker on a white sheet of paper can be used as an arena for robot experiments.

Each of the sensors can be used to fulfill multiple purposes. The sensors will be described more in detail in the following.

1) Floor and Touch Sensor: The floor sensor consists of three infra-red reflective sensors arranged in one line parallel to the axis of the robot. The touch sensor is located on the RGB sensor board to the front of the robot. All sensors are connected to an I^2C based ADC with 12 bit resolution. The ADC can be set up either in free running mode where it throws an interrupt if a given threshold has been over- or undershot. The threshold can be configured independently for each channel. However, it can also be actively polled by the Cortex μC. In this polling mode noise can be reduced by taking a measurement with the IR turned off and on and filtering the data with an Exponential Moving Average (EMA) filter.

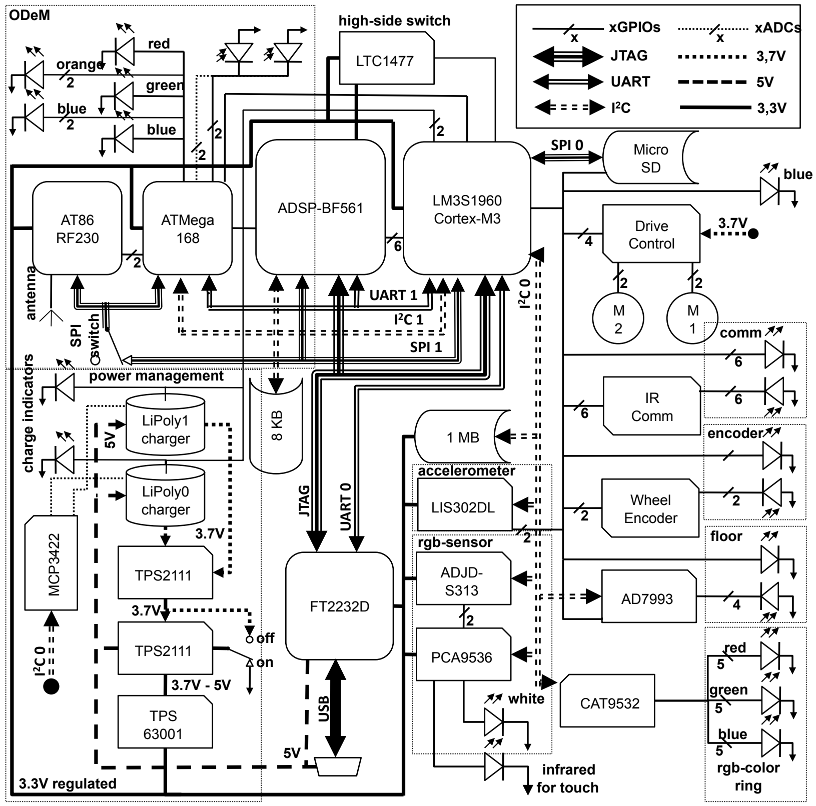

Fig. 4. Schematic drawing of the robot's bus system.

2) RGB Sensor: The RGB sensor is an ADJD-S313 and can be used for several purposes. It basically has three modes of operation that run concurrently: active color detection, ambient color detection and ambient light detection.

The active mode is triggered by the touch sensor, if an object is close enough to the robot to detect the color of the light reflected from the object while emitting from a white LED next to the RGB sensor. Here again we subtract measurements with light turned on (dark measurement) and off from each other to reduce noise coming from ambient light.

The dark measurement can also be used to measure the incoming ambient light, which is the minimum of the measured RGB values. By subtracting the ambient light from the other RGB values we get a good estimation of the main colors of the light received by the sensor. Phototaxis on differently colored light sources that mark different spots in the arena, or receiving the color of other robots is possible. However only true red, green, or blue values or combinations of two of them can be discriminated reliably in this way.

3) Accelerometer: The idea of the accelerometer is to use it for artificial intelligence and evolutionary robotics. The accelerometer should help to detect if the robot hit any obstacles. This serves as a direct and important feedback to online evolutionary robotics. It can also be used to detect if the robot is moving or not which is essential for e.g. collective transport.

Two more sensor systems are implemented on the Wanda robot. The infrared communication and ranging system and the Optical Data transmission system for Micro robots (ODeM). Those sensors are so essential that we dedicate a section for each of them.

III. INFRARED COMMUNICATION AND RANGING

The infrared communication and ranging system enables the robot to communicate with other robots while measuring the distance and direction to its communication partners at the same time for each packet being received. Furthermore, the robot is able to measure the distance and bearing to passive obstacles by receiving its own reflected messages.

A. Electronics

The robot is equipped with six IR-Diodes and six IR-Phototransistors, aligned symetrically around the center of the robot with a angular spacing of 60°. Each Phototransistor is connected to a highpass filter to eliminate ambient light. The output of the highpass filter is then fed into an inverting amplifier with high gain, whose output is directly connected to a schmitt-triggered GPIO pin on the μC, see fig. 5. This makes it possible to measure the intensity of an incoming light pulse simply by measuring the time between rising and falling edge on the GPIO pin, which offers some advantages over other commonly used solutions involving standard IR-Transceivers in conjunction with A/D converters. The system requires only a minimum of external electronic circuitry while still taking away much load from the μC by pre-processing sensory data. Fig. 6 shows the relation between the measured pulse length and the distance of the robot to a wall.

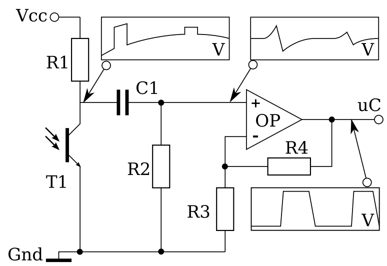

Fig. 5. Schematic drawing of one of the Infrared-Sensors with sketched signals

One drawback of the system is that the conversion from pulse intensity to pulse length prolongens the minimum time distance between two subsequent lightpulses that must be adhered in order to get correct intensity measurements. This limits the maximum transmission bandwith.

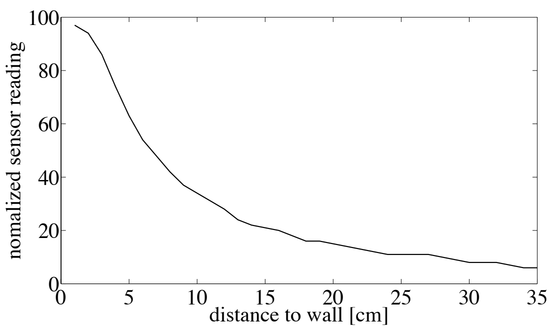

Fig. 6. Distance measurements in front of a white wall

B. Protocol

Due to the minimum time distance between two subsequent lightpulses, it is feasible to encode data by varying the distance between subsequent lightpulses, rather than by presence or absence of a pulse. A group of one starting pulse and five subsequent data pulses constitute a datagram. Each data pulse contains four bits of data, encoded in the distance between itself and its predecessor. This gives 16 bit user data plus four additional bits for control data and error detection per datagram.

To prevent collisions with datagrams from other robots, some kind of synchronisation is needed. Therefore, we have implemented a simple self-synchronising time division multiplexing algorithm:

Time is divided into frames of 100 ms length, each frame constituted of ten time-slots. Each robot occupies one time-slot and tries to minimize its slot-offset with regard to the other robots. During the timeslot, one datagram can be sent. Before sending, the robot checks that no other robot is sending and eventually changes its slot. As communication occurs only locally, ten timeslots have shown to be sufficient even for large groups of robots with an arbitrary high number of members because the average number of robots within direct sight will usually be below ten robots. Overall datarates are quite low right now (about 200 bit/s for each robot) but totally sufficient for most of the experiments conducted so far. Datarates could be further improved by using different protocols (e.g. burst-transmittions, where only some of the transmitted pulses are used for distance measuring).

IV. ODeM Arena

The Optical Data transmission system for Micro Robots (ODeM) is a system that implements an interactive arena for performing experiments with environmental changes during the course of an experiment and to repeat those experiments in an automated way. The system consists on the one hand of a specially designed sensor onboard the robot and an arena with camera and beamers, see fig. 7, which are used to project data in the arena. One robot can be connected to a PC and used as a Host Controller, to start, stop or setup experiments and poll data from the other robots via ZigBee. The ODeM originates from the I-SWARM project [9], [2] and has been designed for the Jasmine robot [10]. It has been transferred to the Wanda robot with slight improvements.

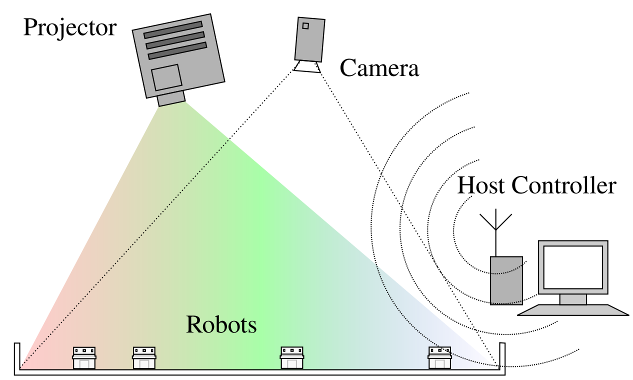

Fig. 7. The ODeM arena

The principle of the ODeM arena is to project three types of data in the arena. The data types that have been implemented are positional data, arbitrary local data, and virtual pheromones. Those different types of data are recognized by the ODeM sensor using two photodiodes on top.

Positional data consists of series of images that encode the absolute position in a grey code. Each photodiode can calculate it's own absolute position by analyzing this series. From this position estimate the position and orientation of the robot can be calculated by the ODeM sensor.

The same principle is used for the arbitrary data which just differs in a header frame from the positional data. This data can be used to mark areas like the home of the robot, hazardous areas, or food sources. In contrast to using the floor sensor, this kind of data can be altered during an experiment and can therefore also show the decrease of a food source or any other environmental change.

For using virtual pheromones a camera has been attached to the arena which detects the orange super bright LEDs on top of the robot. If a robot turns on its light, it is detected by the camera and added to the projected pheromone image. Diffusion and evaporation is done by simple gauss filtering of the pheromone image. Diffusion and evaporation factors can be adjusted by changing the gauss filter.

V. SOFTWARE ARCHITECTURE

Wanda's software is derived from the SymbricatorRTOS, which has been designed and implemented in the SYMBRION and REPLICATOR projects [5]. It is described more in detail in [11]. The SymbricatorRTOS is based on FreeRTOS, an open source embedded real-time operating system which was especially designed with a small foot print [1]. The main features of the SymbricatorRTOS are an embedded C++ API, a hooking system for intertask synchronization and communication and a command shell for robot calibration, configuration and debugging. It also features the possibility to connect the robot via the RS232 interface to a simulation system like player/stage [4] or Symbricator3D [14]. This enables the software designer to run experiments on the real hard- and software but receiving sensor information from the connected simulation environment.

A very generic implementation of a world model gives access to filtered and raw sensor data. It can be recorded on the SD-card. On a 2 GB SD-card, this allows to capture the full state of the robot for several days which is currently a unique feature for such a small robot system.

As described in [11] is it possible to implement almost every state-of-the-art robot control system with SymbricatorRTOS. We use the MDL2ϵ controller language described in [12] for controlling the robot. Several controllers can be uploaded via ZigBee on the SD-card or I^2C memory of the robot and then be started remotely. This enables us to easily perform and debug experiments with several dozens of robots which is important for large scale multi robot experiments.

MDL2ϵ is also used during experimental runs and then ported to the robot. A model of the Wanda robot including the world model, MDL2ϵ, and all sensors has been implemented in the stage simulation environment.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

The described Wanda robot together with the ODeM arena is a very promising tool for research and demonstration in several areas as swarm robotics, evolutionary robotics, sensor-actuator networks and multi-agent systems. The video [6] adjunct to this paper shows just a selection of performance tests done shortly after the production of the first robots. However, even those initial experiments show already how versatile the robot is.

B. Future Works

We intend to use the robots for swarm experiments to investigate the distribution of energy and labour within a swarm, we will also investigate collective pattern formation as a pre-state for a multi-robot 3D organism as intended by the SYMBRION and REPLICATOR projects.

We are also interested in analyzing the reality gap in multi robot simulations. Due to the possibility of running several experiments autonomously we are able to compare a large database with data from simulators. In this way we can analyze which parameters are vital for mapping simulations of large scale multi robot systems to the real world without loss of performance and quality.

VII. ACKNOWLEDGEMENTS

The authors are supported by the DFG Research Training Group 1194: "Self-organizing Sensor-Actuator-Networks" and the SYMBRION and REPLICATOR projects. REPLICATOR and SYMBRION projects are funded by European Commission within the 7th framework program. The Wanda robot has been funded by the KIT Collective Robotics Lab (KITCoRoL).

REFERENCES

[1] "Freertos - the standard solution for small embedded systems," www.freertos.org, 2008.

[2] A. Boletis, A. Brunete, W. Driesen, and J.-M. Breguet, "Solar Cell Powering with Integrated Global Positioning System for mm3 Size Robots," in *Proceedings of the 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2006)*, 2006, pp. 5528–5533.

[3] G. Caprari, T. Estier, and R. Siegwart, "Fascination of Down Scaling - Alice the Sugar Cube Robot," *Journal of Micro-Mechatronics*, vol. 1, no. 3, pp. 177–189, 2002.

[4] B. P. Gerkey, R. T. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *In Proceedings of the 11th International Conference on Advanced Robotics*, 2003, pp. 317–323.

[5] S. Kernbach, E. Meister, F. Schlachter, K. Jebens, M. Szymanski, J. Liedke, D. Laneri, L. Winkler, T. Schmickl, R. Thenius, P. Corradi, and L. Ricotti, "Symbiotic robot organisms: Replicator and symbrion projects," in *PerMIS 08*, 2008.

[6] A. Kettler and M. Szymanski, "Introducing Wanda — A new Robot for Research, Education and Arts," Video submission. Adjunct Proceedings of IROS 2010, March 2010.

[7] F. Mondada, E. Franzi, and A. Guignard, "The Development of Khepera," in *Experiments with the Mini-Robot Khepera*, ser. HNI-Verlagsschriftenreihe, Heinz Nixdorf Institut, 1999, pp. 7–14.

[8] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a Robot Designed for Education in Engineering," in *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, P. Gonalves, P. Torres, and C. Alves, Eds., vol. 1, no. 1, Portugal, 2009, pp. 59–65.

[9] J. Seyfried, M. Szymanski, N. Bender, R. Estana, M. Thiel, and H. Wörn, "The I-SWARM project: Intelligent Small World Autonomous Robots for Micro-manipulation," *From Animals to Animats 8: The Eighth International Conference on the Simulation of Adaptive Behavior (SAB'04) Workshop on Swarm Robotics*, 2004.

[10] M. Szymanski and S. Kernbach, "Jasmine - open source swarm robot," www.swarmrobot.org, 2007.

[11] M. Szymanski, L. Winkler, D. Laneri, F. Schlachter, A. C. Van Rossum, T. Schmickl, and R. Thenius, "Symbricatorrtos: a flexible and dynamic framework for bio-inspired robot control systems and evolution," in *CEC'09: Proceedings of the Eleventh conference on Congress on Evolutionary Computation*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 3314–3321.

[12] M. Szymanski and H. Wörn, "JaMOS - A MDL2ϵ based Operating System for Swarm Micro Robotics," *IEEE Swarm Intelligence Symposium*, pp. 324–331, 2007.

[13] *Stellaris®LM3S1960 Microcontroller – Data Sheet*, Texas Instruments, Austin, TX, USA, 11 July 2009.

[14] L. Winkler and H. Wörn, "Symbricator3D — A Distributed Simulation Environment for Modular Robots," in *ICIRA '09: Proceedings of the 2nd International Conference on Intelligent Robotics and Applications*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 1266–1277.