A DSP-BASED SOCCER ROBOT FOR FIRA MIROSOT

Thomas Klute
Norman Weiss
Simon Schulz
Thomas Pfeifer

Computer Science I
University of Dortmund, Germany
Contact: weiss@ls1.cs.uni-dortmund.de

Abstract: This work presents a soccer robot for the FIRA MiroSot league based on a DSP design instead of the more common microcontroller design to provide more flexibility in the use of the robot. The robot's mechanical and electronical design will be presented along with an overview on its control methods and software. Two additions to the robot – an on-board camera module and a 2.4 GHz radio module – will also be reviewed.

Keywords: robot control, robot navigation, robot vision, control algorithms, systems design

1. INTRODUCTION

In the last few years robot soccer has evolved into a widely used test bed for autonomous mobile robot technologies. The standardisation enforced by the two main robot soccer organisations – FIRA and RoboCup – has created environments highly suitable for demanding control and strategy development that allow a competition of different approaches through frequent robot soccer tournaments. Many different leagues within the organisations now exist. The scope ranges from truly tiny, centrally controlled robots in FIRA NaroSot (4.5 x 4.5 x 5 cm) to humanoid, fully autonomous robots in RoboCup Humanoid League (up to 180 cm in height).

In the FIRA MiroSot league (Kim, *et al*., 2004), the size of a robot is limited to 7.5 x 7.5 x 7.5 cm. Teams of up to 11 robots play against each other on fields sized up to 280 x 440 cm. Robot speeds reach 3.5 m/s during game play. Due to the very small size of the robots, they must be supported by a host computer (usually an off-the-shelf PC) which receives a picture of the field from a camera mounted about 2.5 m above the field. The host is responsible for image processing and strategic decisions. It transmits – via a radio link – movement information to the robots on the field, which they execute, thereby closing the control cycle.

Most FIRA MiroSot soccer robot designs are based on microcontrollers, mainly due to the strict size restrictions and the fact that a sufficient level of game play can be reached by simply transmitting wheel speed information to the robots. This limits the main tasks of the robots to retrieving data from the radio link and controlling wheel speeds, i.e. executing PID controllers.

This work demonstrates that a FIRA MiroSot soccer robot can also be designed on a DSP basis despite its small size. The DSP makes it much more flexible to use than a microcontroller-based design. The mechanical design of the robot will be presented in section 2, the electrical and electronic design in chapter 3. Section 4 presents the control methods and software on the robot, with section 5 summarising the results of an evaluation of acceleration sensors on the robot. Section 6 presents additions currently under development.

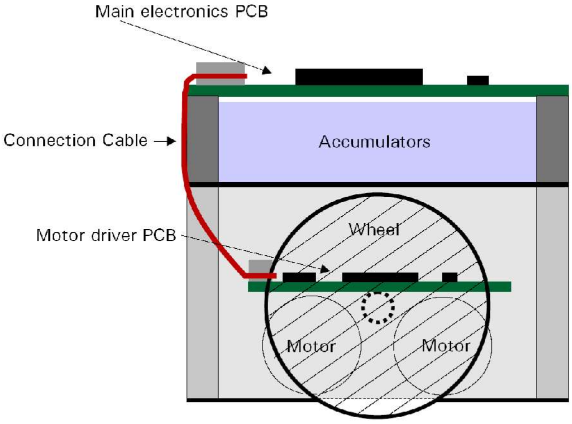

Fig. 1. Mechanical design of the robot (side view)

2. MECHANICAL DESIGN

Due to the small size, the mechanical design is mainly determined by the choice of motors. They need to be powerful in order to accelerate the robot sufficiently fast while at the same time being small, energy-efficient and easy to control. Again due to size restrictions, stepper-motors are unsuitable. That means that the choice is limited to linear motors, which in turn means that a rotation sensor is needed.

Out of the very few available and suitable motors, the choice fell on a Faulhaber 2224R DC linear motor (Faulhaber, 2003a), which is quite common in many FIRA MiroSot soccer robots. Its nominal voltage is 6V, although the supply voltage can be up to 9V without any harm to the motor. Its idle speed is 8200 rpm. The motor features an integrated encoder (type IE2-16), employing a QEP-encoding (Faulhaber, 2003b), where pulses and rotational direction are encoded by the edges of two square signals.

The two motors are built into the base of the robot (see figure 1). The two wheels are mounted on half-axes and driven by the motor via 1:8 gears. This results in 512 rotation impulses per wheel rotation (after QEP-decoding), along with information on the direction of rotation. The motor assembly – alongside with the motor driver PCB (see section 3) is fully encased in the bottom half of the robot. Above the steel enclosing is the (removable) accumulator, with the main electronics PCB on top of the robot.

3. ELECTRICAL AND ELECTRONIC DESIGN

The robot is based on a Texas Instruments TMS320F240 DSP (Texas Instruments, 2002). The DSP features 16 analogue-digital-converter inputs, 2 serial ports, 2 PWM outputs, 28 configurable I/O ports and a JTAG interface (for programming the DSP). It runs on an external clock of 10 MHz and an internal clock of 20 MHz and delivers 20 MIPS.

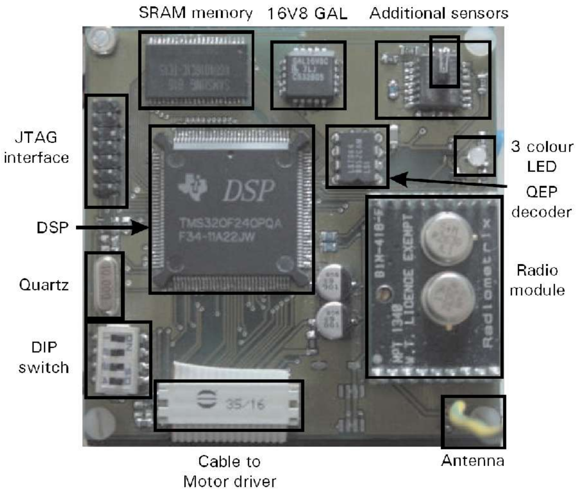

Fig. 2. Main electronics PCB (top view)

Since the internal memory of the DSP is very small (544 Words = 1088 Bytes for data), an external memory chip was added, providing the DSP with 64 kWords of data/program memory and 64 kWords of global memory. Since the DSP can access its internal and external memory at the same time, a 16V8 GAL IC decodes memory accesses from the DSP and routes them appropriately.

For software debugging purposes, the DSP can be set to a debug mode, in which the program is read from the external (volatile) memory. Prior to that, it must have been written to it using the JTAG port. For retaining the program permanently, it can be written to the DSPs flash memory, again using the JTAG interface. Switching between these two modes can be done anytime the main power is off by one of the switches in the DIP switches block on the main PCB. The three remaining DIP switches are connected to input ports used for preconfiguring the robots' identification number. As a method of outputting status signals, the PCB features a three-colour-LED connected to two output ports.

The two motors are driven by motor driver ICs – standard type L298 – on the motor driver PCB. These in turn are connected to the two PWM outputs of the DSP via a ribbon cable that also carries the sensor signals from the motors' rotation sensors and the power supply. One set of motor rotation sensor signals is connected to two input ports of the DSP that are capable of decoding QEP-encoded signals. Since for every QEP-encoded signal two QEP-enabled input ports are necessary and the DSP only features two, an external QEP-decoding IC had to be used for the second. That IC – type LS7084 (LSI, 2003) – is connected to two counter inputs of the DSP.

The main PCB (cf. figure 2) also carries the (exchangeable) radio module. Currently, Radiometrix BiM modules are used (Radiometrix, 2002) that offer a standard serial port connected to one of the serial ports of the DSP. The radio antenna – a

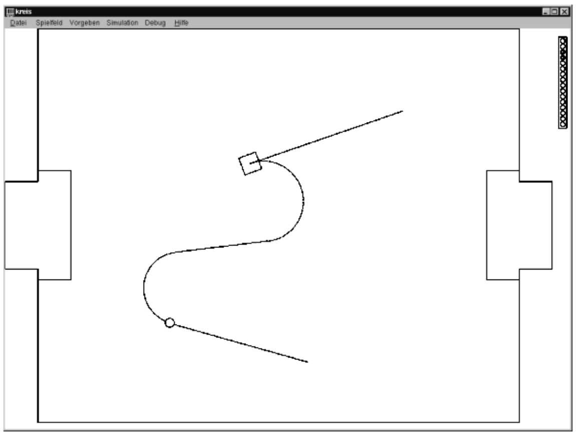

Fig. 3. An example of a s-curve path

simple 1/4λ wire antenna - is mounted next to the radio module. All electronic components are powered by a linear voltage regulator on the motor driver PCB. The motors are powered directly by the rechargeable battery pack, currently 6 NiMH AAA cells with a capacity of 800 mAh.

4. SOFTWARE AND CONTROL

By design, the operation of the robot is fully controlled by the DSP software. Wheel speeds are simply regulated by PID controllers implemented in software. QEP-decoding and PWM-encoding is done fully in hardware, so the software can directly read the number of rotation pulses and set the appropriate motor voltages without further ado. The radio module provides a transparent serial link to the host computer as long as it is used one-directionally only. Since the modules provide no packet control or error checking, a simple CRC-32 error-checking scheme is used in the DSP software. There is no guarding against packet loss, since the packets are sent continuously anyway.

In contrast to many other robots used in robot soccer leagues with central vision, this robot does not directly receive its wheel speeds from the host computer. Instead, it receives simple movement commands (or suitable parameters representing movement commands in a given algorithm). That means that a control algorithm capable of limited path planning is needed for this robot.

At first, a control algorithm derived from the CMU approach (Veloso, *et al*., 1998) was used. It is a reactive approach deriving the left and right wheel speeds from the desired orientation and speed in very frequent intervals. That means that the path is not preplanned. The movement of the robot is redecided on in every update cycle of the control algorithm with the characteristics of the algorithm ensuring steady movements of the robot. As a drawback the CMU approach inherently includes limitations which complicate precise control, namely the fact that the host computer cannot precisely determine which way the robot will move even when simulating the control algorithm.

As an alternative to the CMU approach the s-curve approach has been developed by Messom (1998; 2002) and refined for use on this robot. The s-curve approach is based on the concatenation of partial circular paths with straight lines. The straight line parts of the path allow a high acceleration of the robot, whereas the circular orbits allow a smooth beginning and end of the movement. The circular paths also allow an arbitrary orientation of the robot when it touches the ball. Please refer to figure 3 for a sample of a path planned with the s-curve algorithm. The position of the robot is marked by a square, the position of the ball to be hit in a specific direction is shown by a small circle. Both positions are connected by a path planned by the s-curve algorithm with two partial circular paths (referred to as "orbits") and one aligning straight line ("links").

The algorithm expects the position, the orientation, and the velocity of the start and end point as a vector. Sets of advantageous radii of the orbits can be given as preferences or can be calculated by the algorithm. The first step of the algorithm is to determine the centre points of the orbits . Two feasible centre points exist for each orbit, one to the left and the other one to the right. The decision which one to take depends on the mutual position of start and end point as well as on the start and the target orientation. Both orbits allow the calculation of a path, but one will result in a shorter section of the orbit. The link is only marginally effected by the choice of the orbit. The second step is the calculation of the link between both orbits, based on tangents to the orbits. The third step is the calculation of the radii, which takes centrifugal forces – which might be a limiting factor – into account. The details of the approach applied to this robot can be found in Hildebrand, *et. al.*, 2003.

5. ACCELERATION SENSING

At first, the robot design included an acceleration sensor (cf. figure 2, labelled “Additional sensors”). The sensor – an ADXL202 – can measure acceleration of up to 2 g in two axes and output them as either PWM or analogue values. It is manufactured by Analog Devices (1999). The sensor is sensitive to temperature, therefore an additional standard type temperature sensor was added on top of the ADXL202 IC. This way temperature changes can be accommodated for after an initial calibration. Since all PWM inputs of the DSP are already used, the sensor was connected to two A/D inputs of the DSP.

The intention of using an accelerometer on the robot was to sense acceleration precisely enough to assist the robot in keeping its own position by adding up acceleration values, to detect vibrations and therefore collisions with other objects and to detect slipping of the wheels.

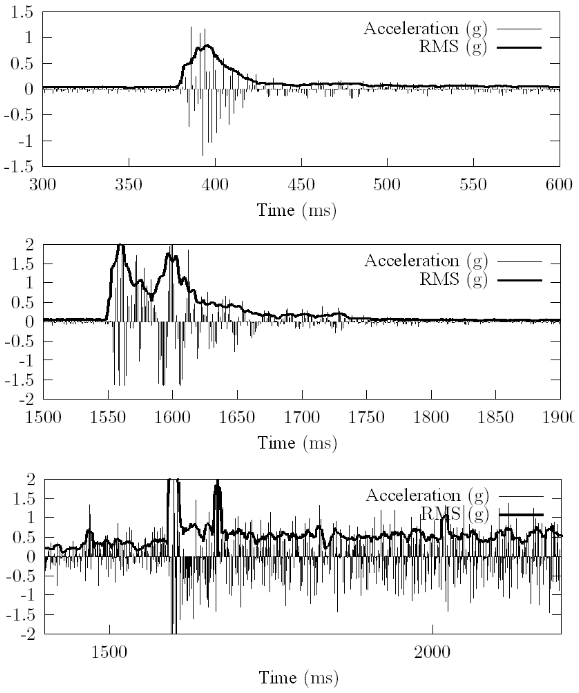

Fig. 4. Readings from the accelerometer during collisions with a ball (top), a robot (middle) and the field border (bottom). Displayed are the sensor readings and the root mean square.

After a short time it clearly emerged that due to the high noise of the sensor itself and the noise induced by the electromagnetic emissions of the DSP, position tracking by using the acceleration sensors is not feasible. Detection of wheel slip is possible but because of the high base noise of the sensor, this is only true for high deviations between expected and measured acceleration.

What emerged however is that a detection of collisions with objects can be done. Moreover, even a classification of the object the robot collided with is possible. Generally speaking, a collision with the ball will produce vibrations with a short single peak, while a collision with the border of the playing field (usually wooden) will produce prolonged, chaotic vibrations (cf. figure 4). Collisions with other robots lie inbetween.

Because of the clear problems that arose during the use of the acceleration sensor, we refrained from using it further and conclude that with a high likeliness the use of this sensor in FIRA MiroSot – although often discussed – may not meet the previous expectations.

6. ADDITIONS

Over time, various additions to the presented robot soccer design have been tested and implemented. Most of them are only feasible because the robot is based on a DSP design. Two recent examples are presented below.

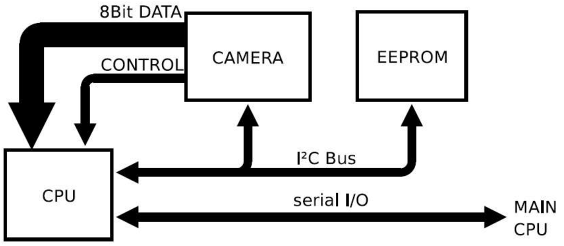

Fig. 5. Block diagram of the camera system

6.1 Camera module

An on-board camera greatly enhances the autonomy of a soccer robot. Naturally, the very small size of this robot poses strong restraints on the size of the camera module and its optics and the computational power needed for image processing. Lately, the strong demand for small cameras to be built into mobile phones has enabled advances on the field not possible before.

For the presented robot the Agilent ADCM-1650 CMOS camera module (Agilent, 2003) seemed well suited. The module includes everything needed to capture a video stream. The 8 x 7 x 5.4 mm casing contains a F2.6 lens, a CMOS array including an A/D converter and a control logic for onboard colour-system transformations and image resizing and scaling functions. The device draws 35 mAh at 2.8 Volts. The maximum resolution is 352 x 244 pixels at a frame rate of 15 fps. An advantage of these camera modules is that the window size and the frame rate are freely configurable via a serial I^2C Interface. In addition, they only need a few external components so that the vision system can be kept very small.

The DSP already on the robots' main PCB is not powerful enough for image processing, so an additional CPU is necessary. It is solemnly dedicated to image processing, with the robot's main DSP handling all other sensing, control and communication. The image processing CPU is an 8-bit CPU (type SX52) made by Ubicom (2004), formerly Scenix. The CPU runs at up to 75 MHz and delivers up to 75 MIPS. It features 2 configurable PWM outputs, 40 general purpose I/O ports, 4096 kB program space, 262 byte RAM and can run at voltages between 2.7 and 5 V. One of the advantages of these CPUs is the small size (PQFP package, 1 x 1 cm without pins). In this application the CPU will run at 50 to 75 MHz and with a voltage of 3 V, which is within the specification of the camera modules, so no voltage shifting circuits are needed for the I/O lines. The main clock source is a configurable DS1077L made by Maxim-IC (Maxim, 2001).

An I^2C EEPROM – type 24LCxx – is used to store configuration data like filtering thresholds, resolutions etc. Because the I^2C control code is needed anyway to control the camera module there is not much additional work to do to address this device.

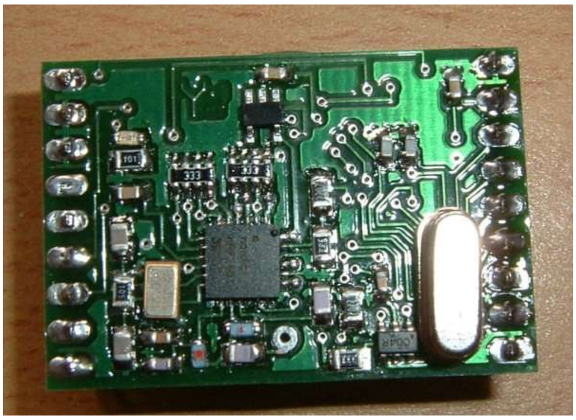

Fig. 6. Radio module (top view)

A larger EEPROM might be used in the future, e.g. to hold lookup tables to transform pixel coordinates to metric values.

The I^2C bus is used only for configuration of the camera module and to read/write configuration data to the EEPROM. The main data bus is the 8 bit wide bus between the camera and the CPU. The data synchronisation is done by 3 further signals (end-of pixel, end-of-line and end-of-frame). The serial line has a voltage level of 3 V so it can be easily interfaced with the main CPU with no additional components. For a PC connection, e.g. for debugging purposes, it must be connected to a hardware interface which converts the 3 V level to the PC serial ports voltage level. The camera module's clock is generated by the SX CPU by using one of its PWM outputs clocked at 12Mhz. Figure 5 shows a block diagram of the overall design of the camera module.

Because the camera-module is sending a continuous stream of data and the CPU RAM is limited to only 262 bytes, it is not possible to store a whole picture in RAM. Instead the data is processed on the fly. Every arriving pixel is analysed for its colour. The calculation of the ball position takes place in a small time slice between two frames. A conversion of pixel values to metric data is then done and the values are immediately transmitted over the RS232 link to the robot's main DSP. First tests show that a vertical resolution of about 80 pixels is enough to find the ball even if it is at the maximum distance of 2.8m possible in FIRA MiroSot. The horizontal resolution will be at less than 255 pixels. In order to set up different colour thresholds, window sizes etc. the robot's main DSP can send control packets to the vision CPU. The incoming data causes an interrupt, during which the new values are stored. Afterwards, the image capture progress is restarted. The entire software is written in C and is compiled with Bytecrafts SXC C-Compiler for the SX series CPUs.

The setup is currently in a working, but prototypic stage. As soon as a planned upgrade to a much more powerful DSP is realised, image processing will be fully possible.

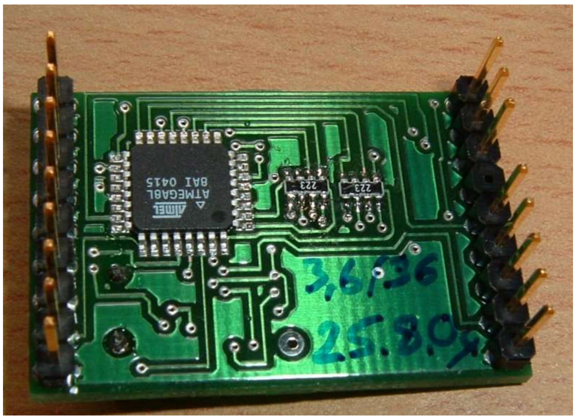

Fig. 7. Radio module (bottom view)

6.2 2.4 GHz radio link

A very limiting factor within FIRA MiroSot is the radio link between the robots and the host computer. Many teams use modules of the Radiometrix BiM series (2002). These transmit in the 433 or 869 MHz ISM bands with specified gross data rates of up to 160 kbps. They feature a very small footprint of only 23 x 33 x 4 mm and a low power consumption of less than 20 mA at 3 or 5 Volts. Unfortunately, due to various reasons, the net data rate will be around 35 kbps in this application, with the modules being half-duplex only.

Therefore, an updated radio module is an useful addition to the presented robot. This subsection shortly outlines a radio module operating in the 2.4 GHz frequency band that is pin compatible to the BiM modules. Please note that this addition is therefore obviously independent of the robot design being based on a DSP.

The module is based on a single chip GFSK transceiver with a data rate of up to 1 Mbit/s. The chip – a nRF2401 built by Nordic (2004) – transmits in the 2,4 to 2,5 GHz ISM Band, on one of 125 channels. With a short channel switching time of less than 200 µs, frequency hopping is possible. A special mode called “Shock Burst” uses an on-chip FIFO for packetising the data. In this mode, payload generation and CRC generation/checking is handled by the transceiver chip. The IC has a very low current consumption of about 11 mA while sending with an output power of –5 dBm and 18 mA in receive mode. It has two independent receivers, so it can receive from two channels simultaneously. The chip can be configured using a simple 3-wire interface. Data transfers are handled over two 3-wire synchronous interfaces. Besides the chip only a crystal and a passive antenna matching network are needed. It has a very small footprint, since it is available in a 5 x 5 mm QFN24 Package. It runs on 3,3 V, which are produced by a linear voltage regulator.

In order to establish compatibility to the Radiometrix BiM modules, the design includes an Atmel ATmega8 microcontroller (Atmel, 2004) that translates between the protocol used by the nRF2401 transceiver and the standard serial protocol used by the BiM modules. The ATmega8 is a low-power 8-Bit RISC microcontroller that runs at voltages between 2,7 and 5,5 V at a speed of 0 to 16 MHz, achieving at maximum of about 16 MIPS. In this design, it is run at 8 MHz and 5V.

The ATmega8 microcontroller is connected to the nRF2401 radio module through its asynchronous serial, the I^2C and the SPI interfaces. The different signal voltages of the chips are adjusted by simple voltage dividers. The software on the controller resets the radio chip and sets appropriate frequency, power and mode settings upon start-up. By default, the radio IC will be set into receive mode. Then, the microcontroller is powered down which results in a power consumption of less than 1 mA. Whenever a byte is received over the serial interface to be transmitted or a complete packet is received by the nRF2401, an interrupt will be generated to wake up the controller. Data received from the radio IC will be send directly to the serial interface. Data bytes on the serial interface will be stored in a software FIFO until they are ready to be sent as a complete packet.

The antenna matching network is based on the reference design by Nordic (2003). As the used transmission frequency is very high, specially selected parts need to be used. The antenna is a simple vertical wire with a length of 1/4λ, i.e. approximately 3 cm. This results in a symmetrical radial radiation, which is well suited for the use on rotating robots.

The module is in a prototypic stage (cf. figures 6 and 7), but works successfully so far.

7. RESULT

The presented robot design works successfully within the FIRA MiroSot league. Using robots of this type, the Chair Computer Science I of the University of Dortmund has won one second and two third places in the European FIRA MiroSot Robot Soccer Championships and one second and one third place in the World Championships. It reaches speeds of up to 2 m/s and allows for very flexible changes in the control algorithms and additional sensing. The on-board camera module is under major development and might provide much higher autonomy to the robot. It can make use of that higher autonomy and additional sensing only because it is based on a DSP design. First tests of the camera module have been very promising. The 2.4 GHz radio link works successfully within the limitations of its prototypic stage and its design might be of high interest to other FIRA MiroSot teams.

REFERENCES

Agilent (2003). *Agilent ADCM-1650-3011 CIF Resolution CMOS Camera Module Data Sheet.* Agilent Technologies, Santa Clara, CA, USA.

Analog Devices (1999). *Low Cost ±2 g/±10 g Dual Axis iMEMS Accelerometers with Digital Output ADXL202/ADXl210 Data Sheet Revision B.* Analog Devices Inc., Norwood, MA, USA.

Atmel (2004). *8-bit AVR With 8K Bytes In-System Programmable Flash ATmega8/ATmega8L Data Sheet.* Atmel Corp., San Jose, CA, USA.

Faulhaber (2003a). *DC-Kleinstmotoren 5mNm, Edelmetallkommutierung.* Dr. Fritz Faulhaber GmbH & Co. KG, Schönaich, Germany. (In German)

Faulhaber (2003b). *Magnetische Impulsgeber.* Dr. Fritz Faulhaber GmbH & Co. KG, Schönaich, Germany. (In German)

Hildebrand, L., B. Reusch, S. Bursch, C. Fenneker, M. Sölter (2003). Path Planning For Mobile Robots Using The S-Curve Algorithm. In: *Proceedings of the FIRA World Congress 2003.* Vienna, Austria.

Kim, J.-H., D.-H. Kim, Y.-J. Kim and K.-T. Seow (2004). *Soccer Robotics.* Springer Verlag, Berlin, Germany.

LSI (2003). *LS7083/7084 Quadrature Clock Converter Data Sheet.* LSI Computer Systems Inc., Melville, NY, USA.

Maxim (2001). *DS1077L 3V EconOscillator/ Divider Data Sheet.* Maxim Integrated Products Inc., Sunnyvale, CA, USA.

Messom, C. (1998). Robot Soccer: Sensing, Planning, Strategy And Control, A Distributed Real Time Intelligent System Approach. In: *Proceedings of The Third International Symposium On Artificial Life And Robotics (AROB III '98).* 422-426, Ohita, Japan.

Messom, C. (2002). Genetic Algorithms For Auto-tuning Mobile Robot Motion Control. *Res. Lett. Inf. Math. Sci.,* **3**, 129-134.

Nordic (2003). *Universal Low Cost USB DuoCeiver Using nRF2401 Application Note Revision 1.0.* Nordic Semiconductor ASA, Tiller, Norway.

Nordic (2004). *NRF2401 Single Chip 2.4 GHz Transceiver Product Specification Revision 1.1.* Nordic Semiconductor ASA, Tiller, Norway.

Radiometrix (2002). *869/914MHz High Speed FM Radio Transceiver Module BiM3 Data Sheet.* Radiometrix Ltd, Watford, UK.

Texas Instruments (2002). *TMS320F240 DSP Controller Data Sheet Revision E.* Texas Instruments Inc., Houston, TX, USA.

Ubicom (2004). *SX48BD/SX52BD Data Sheet.* Ubicom Inc., Mountain View, CA, USA.

Veloso, M., P. Stone, K. Han and S. Achim (1998). The CMUnited-97 Small-Robot Team. In: *RoboCup 97: Robot Soccer World Cup 1* (H. Kitano. (Ed)), 242-256. Springer Verlag, Berlin, Germany.