Advanced control structure for the autonomous mobile robot Lodur

Pierre Maurer[1], Micael Gagnebin[2]

HESSO-HEIG, West Switzerland University of Applied Sciences,
Yverdon-les-Bains, Suisse, [1] Pierre.Maurer@eivd.ch [2] Micael.Gagnebin@eivd.ch

Abstract

The Laboratory of Robotics and Automation (LaRA) of HESSO-HEIG has developed several autonomous mobile robots, featuring special control structures and architectures. These robots were designed to participate to the Swiss and European robotics cups EUROBOT. This paper describes the autonomous mobile robot "Lodur" which was used as a defending robot at the side of the autonomous mobile robot "Lomu", during EUROBOT 2004.

The control part of "Lodur" has the advantage of being contained in a minimal volume while presenting enough flexibility to allow for easy programming of strategies and for integration of a sufficient number of external components. The choice was made to use the control module BECK, which is a 16-bit controller coupled with an FPGA interface board assigned to carry out all the external operations with PC such as motor control, the counting of encoder pulses and the processing of some additional binary inputs/outputs signals. The control logic is integrated in an Altera CPLD and takes care of all the operations described above.

1. INTRODUCTION

At the beginning of the project, the Lab for Robotics and Automation, at HESSO-HEIG (formally EIVD) had already been designing mobile autonomous robots for about 10 years (e.g. [1-6]), for a variety of contexts, with a number of successful embedded control systems.

But to design the control unit of the robot "Lodur", a new challenge was set, relating to a very small available space, while keeping past absolute requirements in terms of decision and power autonomy. It mattered to choose a flexible solution, allowing for programming by means of tools available at the laboratory of robotics, and on which it was to be easy to carry out modifications on the level of control program. The solution, which has been designed, is based on the use of the Integrated PC-based, DK-40 module of Beck [7], augmented by an FPGA circuit specifically configured for additional IO's and fast motor control. For programming, the main unit runs our Piaget-light environment, and depending on strategies, the FPGA can be reconfigured.

Figure 1. The autonomous mobile robot Lodur

2. HARDWARE

The hardware components of Lodur consist in three main components : an integrated PC, an FPGA, and power circuits for motors.

2.1 IPC-MODULE

Figure 2. Module Beck

The IPC module features in particular:

- 8 digital inputs/outputs with output level of 15-30 V, with status LEDs;
- 2 serial interfaces RS-232 with TTL output level;
- 1 Ethernet network interface.

A major advantage of this module is that it has a real-time operating system RT-OS [8], compatible with the DOS of personal computers (PC). It is possible to store programs until a limit of 1 MByte, and to configure the start-up by means of an AUTOEXEC.BAT file. Thus, it is possible to develop a control program by using the same tools as those that are used to create a program on a PC. We wrote the control program using Borland C++, then, once the program was compiled and tested, we downloaded it into the module using the Ethernet connection.

The IPC DK-40 module of Beck however presents also some weak points. One can mention the fact that it offers only 8 binary inputs/outputs, which pretty much limits the possibilities for controlling a robot, even in the case of the Lodur robot, which has only a small number of functions. This is why it was decided to design and implement an intermediate board in order to extend the functions of the unit.

2.2 FPGA AND INTERFACING BOARD

The purpose of the interfacing board is to improve data communication for the DK-40 module, and to execute various tasks relating to robot operation. It includes some electronic components for DK-40 interfacing and a CPLD Altera EPM74'128 [9], which includes most of the logic circuits for data processing and outwards operations.

The 8 bits of inputs/outputs of the DK-40 module are used to communicate with the intermediate board: the 4 bits of low weights represent the transmitted user data, and the 4 bits of high weights are used to define 16 different destination addresses. Moreover, due to the fact that we can work in writing or reading mode, we have thus 64 bits of data in writing mode and 64 bits of data in reading mode. The intermediate board takes care of all data receiving and data transmissions towards outside: command of motors, command of auxiliary devices, pulse counting, reading of the signals from collisions detectors, and so on.

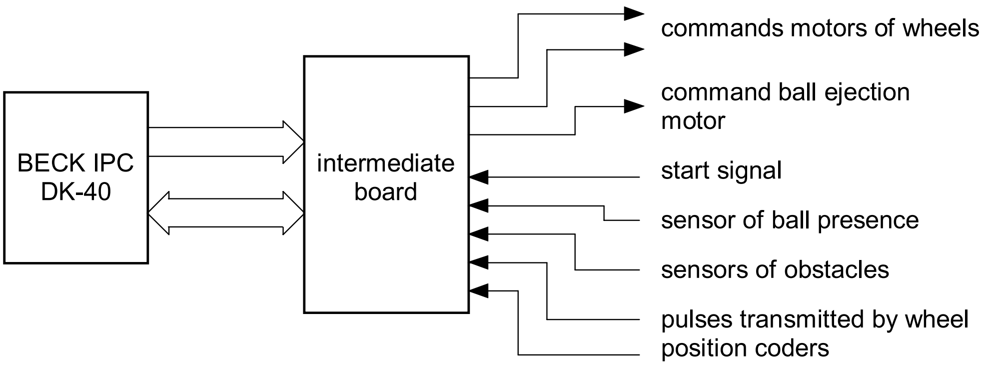

Figure 3. General overview of data transmission

Figure 4 presents in a more detailed way the organization of the signals inside the CPLD and the way in which the communication between the various internal registers and the BECK processor is established.

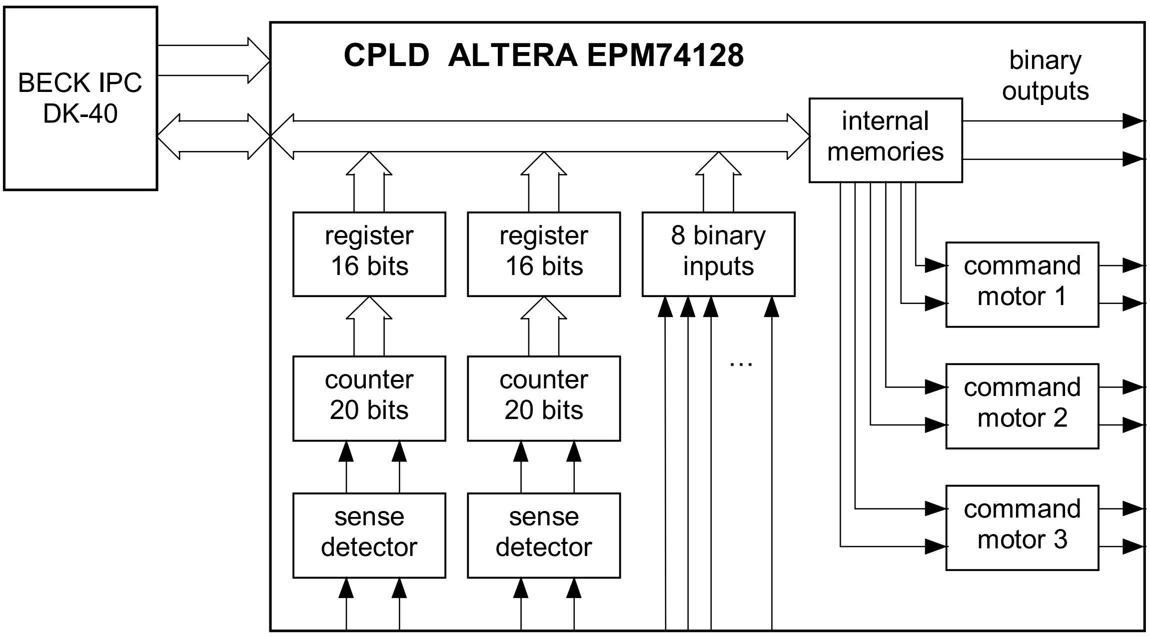

Figure 4. Schematic diagram of CPLD board

2.3 MOTOR CONTROL

Robot motion is ensured by DC motors, supplied through H-shaped bridges of MOS transistors.

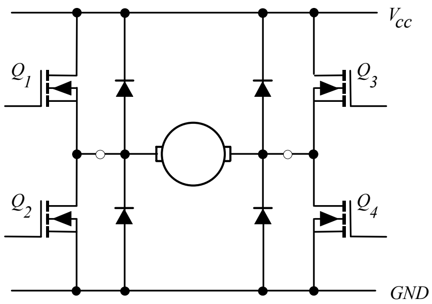

Figure 5. Wiring of motor supply

In this wiring diagram, 4 MOS transistors function like switches. Either they are blocked, or they are conducting, with a very small voltage drop. In this manner, they dissipate very little power by Joule effect and their heating remains moderate. To explain operation in a simplified way, one can say that the transistors Q1 and Q4 must be commanded by a logical signal A. When A = 0, Q1 and Q4 are blocked and when A = 1, Q1 and Q4 drive the current. In the same manner, the Q2 and Q3 transistors must be controlled by another logical signal B.

A first possible operating mode is to put the control signal A in logical state 1 and the signal B in logical state 0. Under these conditions, a current flows through the Q1 transistor, passes through the motor from left to right, and flows then through the Q3 transistor, returning to the ground of the supply. This current makes the motor turn. Another possibility is to put signal A in logical state 0 and the signal B in logical state 1. Under these conditions, a current flows in opposite direction through the motor and the direction of rotation is reversed. The table below summarizes 4 possible operating modes according to the 2 control signals A and B:

Case	A B	Functioning mode
0	0 0	Blocked transistors
1	0 1	Left wise rotation
2	1 0	Right wise rotation
3	1 1	Forbidden case

Table 1. Functioning modes

The combination A = 0, B = 0 results in blocking all transistors. The motor is not supplied any more, but it is not braked when it turns. The combination A = 1, B = 1 would cause that the 4 transistors would conduct current at the same time. That would cause a short-circuit of the supply and the intensity of the current in the components would be so high that it would cause their destruction. This situation of short-circuit in the command system must be avoided at all costs. In particular, when one passes from the situation A = 1, B = 0 to the situation A = 0, B = 1, a short-circuit is theoretically possible because conduction times and blocking times of MOS transistors are not identical. It is thus preferable to dwell a short time in the combination A = 0, B = 0 at each direction inversion of the command.

Some manufacturers of motor control circuits have envisaged another possible combination, which consists in blocking the transistors Q1 and Q3, and to make conducting the transistors Q2 and

Q4. This situation is interesting because it causes a short-circuit at the motor terminals, which results in an energetic braking of the latter.

The H-shaped bridge circuits, such as the one depicted in figure 5, lend themselves particularly well to the speed control of the motors using the pulse witdh modulation technique (PWM). In this case, the control signals change state periodically and the number of motor revolutions depends on the ratio of durations of the two signals. Figure 6 illustrates this operation. It represents the signals A and B and the current I_m in the motor as functions of time:

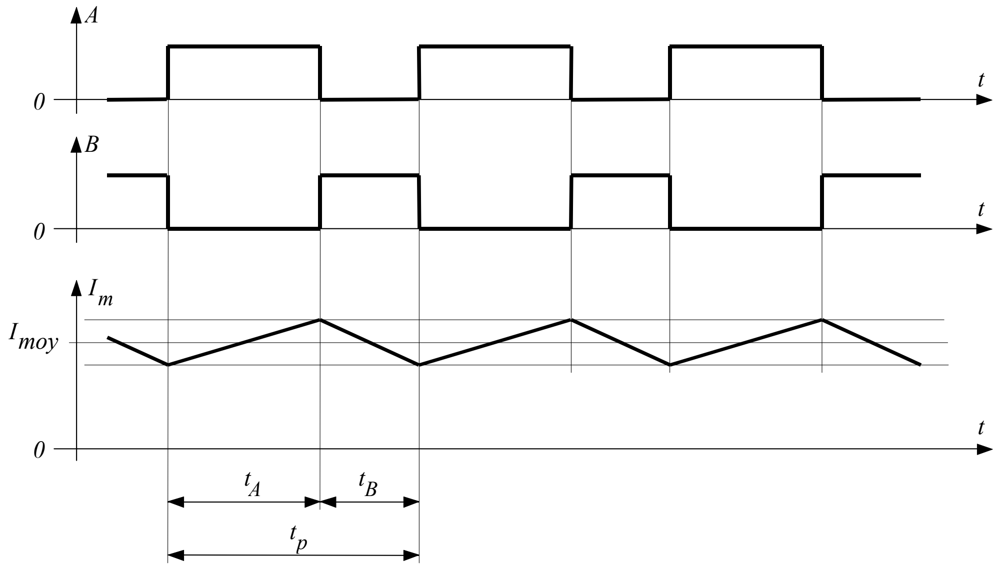

Figure 6. Control signals

The period t_p of a cycle of this signal must be selected rather short in comparison to the electric time-constant of the motor, so that the current varies rather slightly during the cycle and remains close to its average value. The motor turns all the more quickly as the ratio t_A / t_B is larger. If t_B is larger than t_A, the direction of rotation of the motor is reversed.

To create signals A and B in a cyclic way, we have used a binary 32-state counter. We estimated that with 32 states, that enabled us to have 16 different speed values for each direction of rotation and that that offered sufficient flexibility for motor control. The counting pulses are formed by a division of the frequency of the clock signal, which drives the CPLD so as to obtain a base frequency of 512 KHz for the counter. Since the cycle of the counter comprises 32 periods, the generated PWM signal frequency is thus 8 KHz. The relationship between the duration t_A and the period t_p is determined by a comparator, which compares the state of the counter with a reference value stored in an auxiliary register. The whole architecture of the motor control is represented schematically in the figure 7:

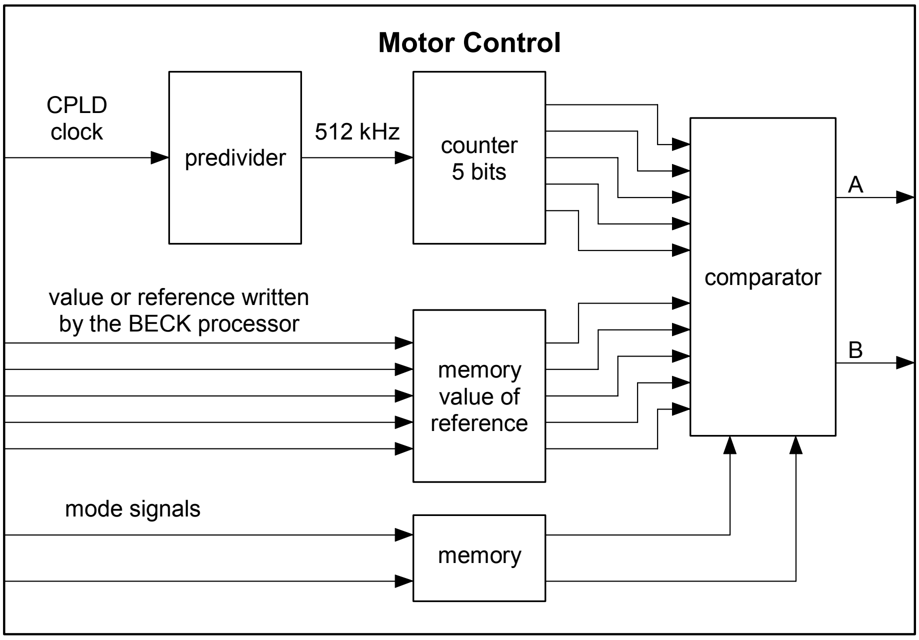

Figure 7. Motor control architecture

The formation of A and B signals for the circuit driving motors depends not only on the state of the counter and the value of reference stored in memory, but also on the mode signals sent by the BECK processor. With the mode signals, one can define four modes, which are "Left wise rotation", "Right wise rotation", "Blocked transistors", which is equivalent to "Motor shutdown without blocking", and finally the mode "Shutdown with blocking", if the power circuit, which receives the A and B control signals, allows this operating mode.

If one considers the mode "Right wise rotation", at the beginning of each cycle of the counter, i.e. when it contains the binary value "00000", the control signal A is put at the logical state 1 and the signal B is put at the logical state 0. When the state of the counter reaches the value written in the reference memory, the signals A and B are inverted. Actually, in order to avoid any risk of short-circuit in a branch of the H-bridge in the motor driving circuit, we envisaged a shift of commutations of signals A and B equivalent to one state of the counter, i.e. when one of signals A and B is reset to 0 for a given state of the counter, the other signal is set to logical state 1 at the next pulse of the counter. The signals A and B, such as they are actually generated, are represented below:

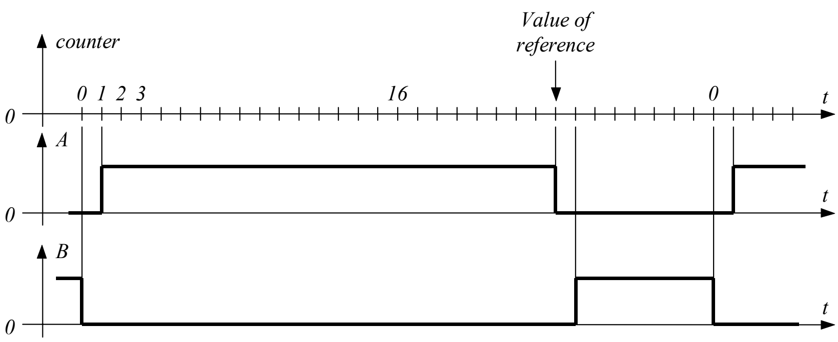

Figure 8. A and B control signals

This configuration is repeated three times in the CPLD, which contains the whole management logic of the motors. As a matter of fact, because of a limitation of the number of bistable trigger circuits available in the CPLD, only the motor controls of the wheels are equipped with a memory allowing registering a reference value from the BECK processor. For the third motor, this function has a less importance, because this motor is used to drive a belt or used for another internal function of the robot, and it is not necessary that the speed of this motor can be varied. Thus, the reference value used for driving the third motor has a fixed value, stored when programming the CPLD, and determined during the test phase of the robot.

3. PROGRAMMING

Lodur was the first implementation, in 2004, of the small robot concept presented above. For 2005, the same hardware architecture has been used again. On the software side however, a very different approach has been practiced. The proprietary Piaget environment developed in our group for the control of more powerful robots, relying on full-grade PC's, such as Sony VAIO UPG3 has been ported to the IPC in a lighter version. Called « Piaget-Light », this environment is implemented in C instead of C++, does not handle vision and ordinary graphic displays, but nevertheless allows for the convenient use of multi-agent and application-oriented language. Some parallel tasks consist in particular in wheel position-control, trapezoidal speed profiles, and direct kinematic solutions, so as to track position in Cartesian space and Cartesian coordinates available to the user.

4. CONCLUSION

After years of solutions relying on full-fledged PC's for robot control, recently two smaller robots have been designed to compete in Swiss and European Robotics Cups - EUROBOT. "Lodur" in 2004, and with minor changes, "Walter" in 2005 (in 2004, the CPLD did not allow yet for dynamic parameterization on 32 "levels" of the already existing PWM generators for motor control).

The control part of those robots brings advantage of being contained in a minimal volume (about 8 cm on the largest side), while presenting enough flexibility to allow for easy programming of strategies and the satisfactory handling of a sufficient number of external components (8bit of Boolean input, 8 bit of output, 2 16 bit-wide encoder counters, and 2 32-levels PWM generators). Choice was made to use BECK Integrated-PC as control module, which is a 16-bit controller, coupled with an FPGA (Altera CPLD) interface board assigned to carry out all external operations such as motor control, counting of encoder pulses and processing of additional binary inputs/outputs signals. In this minimal environment, it has been possible to run our Piaget-light, multi-agent environment, which ensures very user-friendly, fast and efficient definitions and updates of playing strategies.

Experience shows that the system fulfils the chosen specifications with high reliability. We can now imagine further improvements : Among the improvements considered, we can quote the introduction of a control system for the motor torques as well as a coordination of the movements of the driving wheels on the level of the CPLD. These two improvements would discharge the BECK-IPC module of some of its tasks. These tasks would be carried out better at the level of the CPLD, which has the possibility of intervening with a much shorter cycle time, and which could thus carry out them in a much more effective way. The BECK-IPC module could then perform more background tasks and it would be perhaps possible to assign operations such as processing images of the playing field and surrounding obstacles.

REFERENCES

[1] D. TURCHI, S. TROLLIET, L. BOURGEOIS, AND J.-M. VULLIAMY, "DESIGN AND LOCOMOTION CONTROL FOR A TELEROBOT OF SOJOURNER TYPE", 48TH SYMPOSIUM OF THE INTERNATIONAL

ASTRONAUTICAL FEDERATION, TORINO, ITALY, OCT. 97, PAPER ST-97-W.2.04 (RE.) PUBL. BY THE AMERICAN INSTITUTE OF AERONAUTICS AND ASTRONAUTICS, NEW YORK, USA.
[2] J.-D. DESSIMOZ, P.-F. GAUTHEY, M. ETIQUE, A. ROTZETTA, B. SAUGY, AND F. DRÄYER, "AUTOMATIC PATH GENERATION AND STEERING FOR A NEW, INTELLIGENT URBAN TRANSPORTATION SYSTEM", PROC. ICAM'98, 3RD INTERNATIONAL CONFERENCE ON ADVANCED MECHATRONICS, - INNOVATIVE MECHATRONICS FOR THE 21ST CENTURY, OKAYAMA, JAPAN, VOL.2, PP. 762-767, 3-6 AUG. 1998.
[3] J.-D. DESSIMOZ, PIERRE-FRANÇOIS GAUTHEY, MICHEL ETIQUE, BERNARD SAUGY ET ANDREA VEZZINI, "SERPENTINE - AN INTELLIGENT URBAN TRANSPORTATION SYSTEM FOR PASSENGERS, WITH AUTONOMOUS, MOBILE ROBOT PROPERTIES", CONF. AND PROCEEDINGS, SMC'99-IEEE, TOKYO, JAPON, OCT. 1999
[4] J.-D. DESSIMOZ, "IS A ROBOT THAT CAN AUTONOMOUSLY PLAY SOCCER INTELLIGENT?", PROC. INTERN. CONF. ON INTELLIGENT AUTONOMOUS SYSTEMS, NO 6, E. PAGELLO ET AL (EDS), IOS PRESS, CO-SPONSORED BY IEEE, VENICE, ITALY, JULY 2000, PP. 951-958
[5] M. DELATTRE, N. UEBELHART, S. VON WYL, S. BHATNAGAR, "POSITIONNING AND COLLISION AVOIDANCE FOR AUTONOMOUS MOBILE ROBOTS", NATIONAL WORKSHOP IPLNET 2004 ; AUTOMATION AND LOGISTICS FOR MANUFACTURING IN SWITZERLAND », MORAT, SEPT. 6-8, RESEAU NATIONAL DE COMPETENCES EN PRODUCTIQUE ET LOGISTIQUE INTEGRALES IPLNET,. (VOIR AUSSI SUR LE WEB. CF : WWW.IPLNET.CH)
[6] ANTOINE TRAD, JEAN-DANIEL DESSIMOZ, "AN ARCHITECTURAL DESIGN PATTERN PROPOSAL FOR MOBILE ROBOT SYSTEMS", CONFERENCE PROCEEDINGS OF THE INTERNATIONAL IMS (INTELLIGENT MANUFACTURING SYSTEMS) FORUM 2004, HELD IN CERNOBBIO - ITALY ON 17TH-19TH MAY.PP. 346-351,(HTTP://WWW.IMSFORUM2004.ORG)
[7] BECK, DK40 DATASHEET & HARDWARE MANUAL, HTTP://WWW.BECK-IPC.COM, 2005
[8] BECK, IPC@CHIP DOCUMENTATION INDEX - @CHIP-RTOS V1.04
[9] WWW.ALTERA.COM : MAX 7000 FAMILY: HIGH-PERFORMANCE CPLDS