iCC 1996

3rd international CAN Conference

in Paris (France)

Sponsored by

Motorola Semiconductor
National Semiconductor
Philips Semiconductors

Organized by

CAN in Automation (CiA)
international users and manufacturers group
Am Weichselgarten 26
D-91058 Erlangen
Phone +49-9131-69086-0
Fax +49-9131-69086-79
Email:headquarters@can-cia.de
URL: http://www.can-cia.de

G. Bourdon, P. Ruaux, S Delaplace.
Laboratoire de Robotique de Paris
10-12, av. de l'Europe
78 140 VELIZY-FRANCE
Email : bourdon@robot.uvsq.fr

C.A.N. for Autonomous Mobile Robot

In this paper, we describe the hardware and software architecture of an autonomous mobile robot. ROMAIN, INstrumented Autonomous Mobile RObot, is an experimental platform dedicated to a wide range of applications. Its main characteristics are flexibility, high speed velocity, energetic and decisional autonomy and real time processing. This mobile robot owns an open structure based around a Control Area Network for data transmission between microcontroller cards and a real time operating system. After the description of this structure we propose an application consisting in the tracking of an other robot in an unknown environment with ultrasonic sensors.

Introduction

The industrial applications of mobile robots are multiple, such as container carriage in factories, food-trays conveyance in hospital, specific intervention in radio-active cells. Researchers are working to increase robots autonomy in order to develop the field of this applications.

This objective was ours when we decided to build ROMAIN. This mobile robot is the second one realized in our laboratory after ROME0 (Experimental Mobile RObot number 0) [1]. This new robot had to be fast, flexible, offer real time processing capabilities, equip with adapted sensors and own energetic and decisional autonomy. It is the necessary characteristics to study a wide class of mobile robotics problems as navigation [2], dynamic control [3], landmark tracking [4].

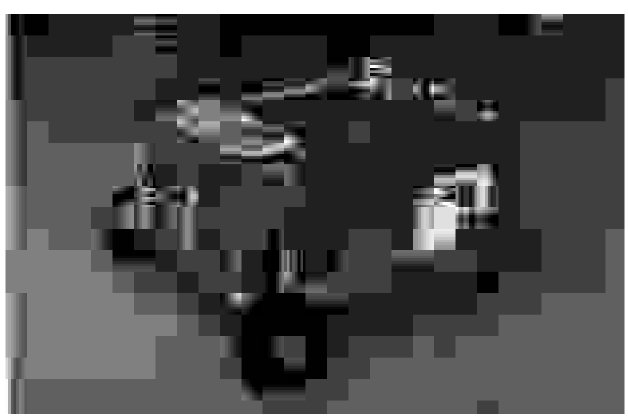

figure 1 : ROMAIN, INstrumented Autonomous Mobile RObot (LRP).

The locomotion structure of ROMAIN is simple but allows high velocity and platform stability

PC with an OS9000 operating system has in charge the high level computation.

The communication between high and low layers is ensured by a Control Area Network (CAN). This serial bus offers a high baud rate data transmission and a great flexibility for programming, node addition or removing.

Locomotion system

Several kinds of locomotion structure exist in mobile robotics. The choice of a mechanism is depending on the application and its criteria. The environment characteristics can be one of these factors, the ground nature (light, hard or hilly) or the task specification (carriage, supervising, exploration...). The different kinds of structure we can find (fig.2) are track (CENTAURE from CEA), peristaltic systems [5], legs [6], wheels [7] or hybrid of these systems [8].

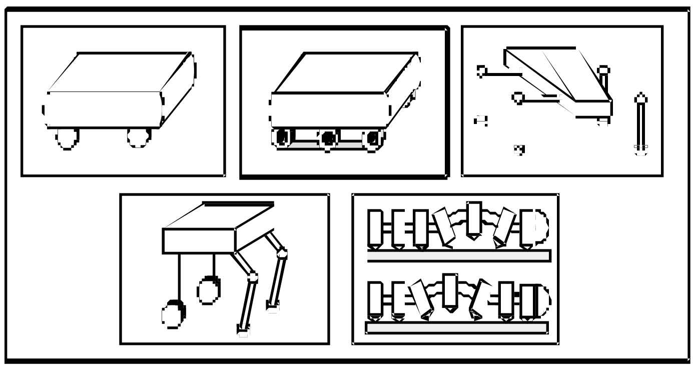

figure 2 : different locomotion structures in mobile robotics.

ROMAIN is a mobile robot with two rear driving wheels and two front free wheels. Its mass is about 100 Kg. The propulsion system is composed of two motor/reducer with a 16 reduction ratio, developing 500 W each (fig. 3). An 2000 points optical encoder is placed on each axle for a resolution of 128000 points/revs.

The power supply of 60 Volts is provided by 5 batteries. The energetic autonomy in a nominal working is about 4 hours. The maximum velocity and acceleration is respectively 4 m/s and 1.5 m/s_.

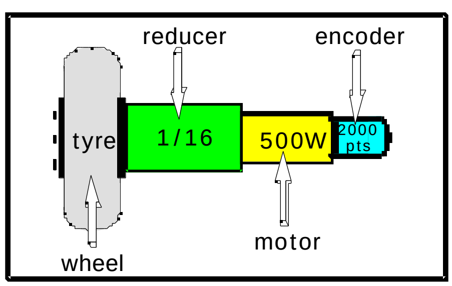

figure 3 :the encoder-motor-reducer set.

Kinematic model

This locomotion structure is classical but confers an easy utilization. It provides platform stability and allows high velocity. Nevertheless, such a choice presents some problems of control due to its non holonomy. Let the parameter of the system described fig. 4.

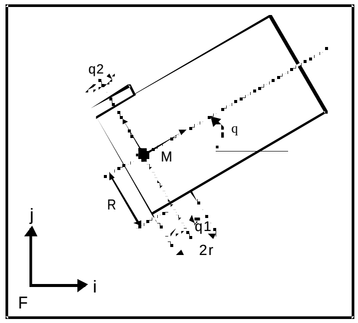

figure 4 : coordinates of mobile robot.

Suppose wheels are rolling without slipping, the kinematic model is given by :

$$\dot{x}_m \cos\theta + \dot{y}_m \sin\theta = \frac{r}{2}\left(\dot{q}_1 + \dot{q}_2\right) \quad (1)$$

$$\dot{x}_m \sin\theta - \dot{y}_m \cos\theta = 0 \quad (2)$$

$$\dot{\theta} = \frac{r}{2R}\left(\dot{q}_1 - \dot{q}_2\right) \quad (3)$$

with :

x_m, y_m : M coordinates in the absolute frame F.
θ : robot orientation in F.
$\dot{q}_1$: right wheel angular velocity.
$\dot{q}_2$: left wheel angular velocity.
r : wheel radius.
R : distance between a wheel and M.

One can demonstrate that equation (1) and (2) are non integrable, they are non holonomic constraints which limit the system motions. It means that the robot velocity is at each time, tangent to its trajectory.

General hardware architecture

The hardware conception was realized in order to perform parallel processing. To cope with different situations, an autonomous mobile robot has to manage several tasks including perception, decision and action in real time.

For this robot we adopted a decentralized system based on microcontroller cards and an industrial PC which form the nodes of a communication network the Control Area Network (CAN).

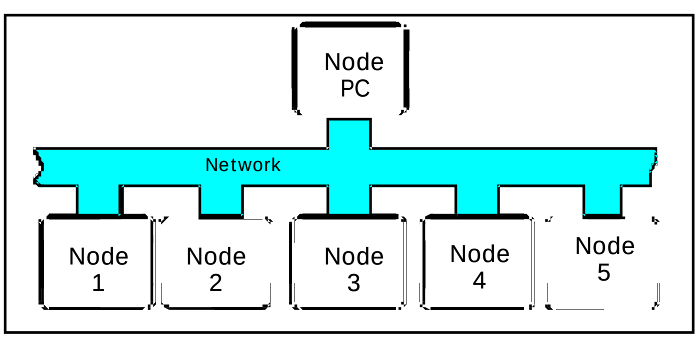

figure 5 : general hardware architecture.

The hardware is organized in two layers.

- The high layer is composed by a 486 DX2-66 industrial computer. An OS9000 operating system is installed allowing parallel task processing.
- Nodes of the second layer consist of microcontroller cards with their annex. One node is dedicated to a specific data processing.

system a real parallel architecture able to ensure real time processing.

The CAN network

To conserve the advantage of such a parallel decomposition, the data transfer had to be fast and with no heavy communication protocol. This is the reason why we choose the CAN solution.

Some CAN protocol properties are the following :

- prioritization of messages
- guarantee of latency times
- configuration flexibility
- multicast reception with time synchronization
- system wide data consistency
- multimaster
- error detection and error signaling
- automatic retransmission of corrupted messages
- distinction between temporary errors and permanent failures of nodes
- autonomous switching off of defect nodes.

The CAN presents a great flexibility because of the possibility to add or remove a node without network reconfiguration. Due to this property, our system becomes open and so allows an evolution of the platform.

A message length is from 0 to 8 bytes which are transmitted MSB first. It is marked by an identifier (11 bits) which indicates the nature of the contents. Several nodes can then receive a same message, a filtering function ensures message selection for each node.

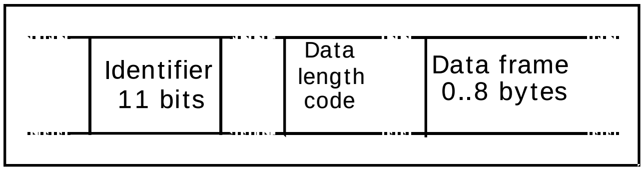

figure 6 : part of data frame.

Moreover, the CAN protocol guaranties a high degree of security with a bus monitoring, a CRC, a message frame check and stuff bit procedure.

Levels specification

We detail here the hardware conception of the different nodes. These cards have been completely designed in our laboratory.

High level node specification

High level consist of the industrial PC and its communication card interface (fig.7). The bus connection is ensured by a special designed CAN interface card with the communication controller 80C200.
It is equipped with a PCL 812PG data acquisition card (PC-LabCard) to manage other peripheral tools as joystick for instance.

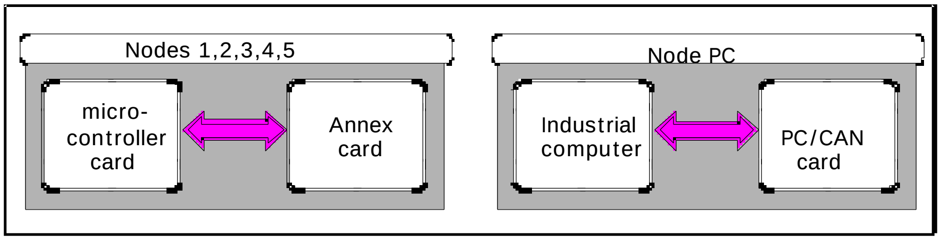

figure 7 : low and high level nodes.

Low level nodes specification

The low level is composed of nodes all based on the same microcontroller card MC (fig.7). A MC is built around an Intel 80C31 8 bits microcontroller (8032 for the modem node). A 128K RAM and a 256K ROM are the available extern memory. A 82C200 CAN controller ensures the bus connection. Such a card has in charge to process data sampled by its annex card. An annex card is attached on each MC dedicated to one type of computation (ultrasonic sensor management, modem data acquisition, motor control).
Figure 8 describe the general structure of this layer.

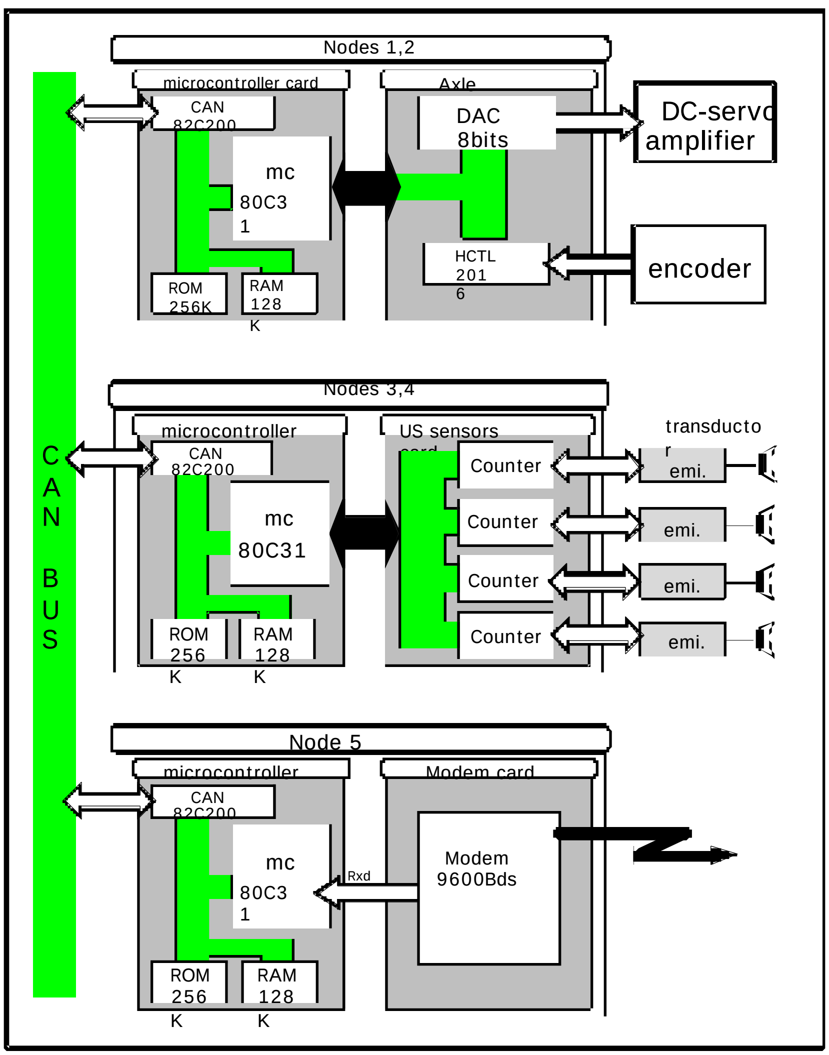

figure 8 : detailed hardware structure.

which has for inputs both signals of an optical encoder. An interrupt handler samples the counter value every 2 ms. According to the high level orders, the new command is computed and sent to the DC servo-amplifier (Copley 306) via a 8 bits Digital Analogic Converter. A message is writing on the CAN bus at each sampling time in order to compute position and orientation on the high level. The data length is 16 bits.

Nodes 3 and 4 : dedicated to the management of ultrasonic sensors.
The US annex card is able to control 4 Polaroid sensors through their emission cards. Four THCT 12016 give the distance of the first echo with an electronic precision of 0.33 cm. At each echo reception, these values are sampled by an interrupt handler. This value is sent by the MC to the high level through the network. The data length is 16 bits and the identifier indicates the receiver sensor.
A US wave is emitted 10 ms after the reception. The sampling, depending on the wave Time of Flight (TOF), is then asynchronous.

Nodes 5 : dedicated to the HF receiver management.
A HF receiver (433.92 MHz) is connected to the RS232 input of the 8032. These nodes permit to have a remote control of the robot, especially to stop it or to transfer simple commands. An interrupt handler is called at each reception. The acquired data is sending in a can message under the suitable format.

Software architecture

We describe in this sub-section the structure adopted for the software tasks managing.
The PC is equipped with an OS9000 operating system from Microware. He manages three task types : the *peripheral tasks*, the *data process tasks* and the *high level tasks* (fig.9).

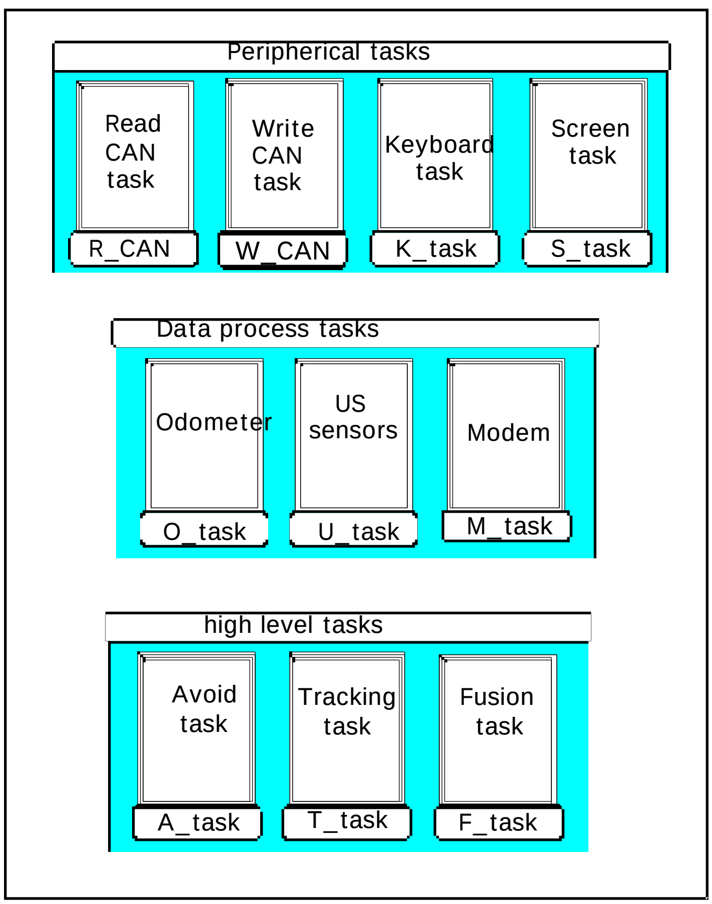

figure 9 : tasks definition.

- **Peripheral tasks**

R_CAN : this task reads the messages on the bus and stores them in different modules where the *data process tasks* can access.

W_CAN : it writes the commands sent by the high level tasks on the network with the appropriate identifier.

K_BOARD : manages the keyboard. Allows to an operator to send orders to control or stop the robot (works with interrupt).

S_task : permits to have a visual control for an operator.

- **Data process tasks**

These tasks access to specific modules where data are stored. Their roles are to do simple computation and especially to convert the received bytes in good format.

O_task : odometric task. Computes position and orientation of the robot in the absolute frame.
U_task : transforms the ultrasonic sensors data in distance.
M_task : converts the modem data in the appropriate format.

- **High level tasks**

These tasks do high level computation. That means complex algorithms of path planning, trajectory tracking, obstacle avoidance, target tracking.

Application example

To illustrate the tasks managing, we propose an example of collaboration between two mobile robots. The experimental procedure is described fig. 10.

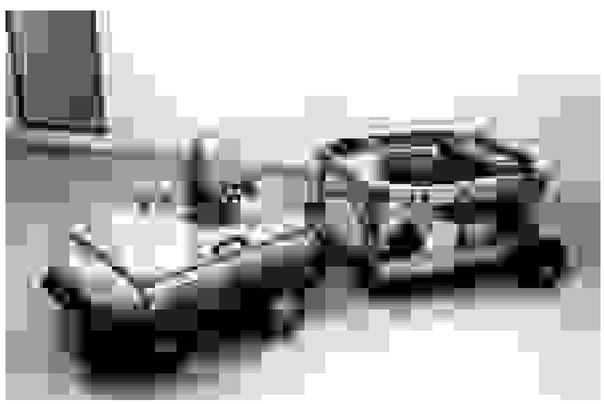

figure 10 : experimental tracking.

A leader, the Robuter II (ROBOSOFT), is sending its absolute position to ROMAIN via the modem with a 10 ms period. Our robot has to join and track the leader along its path. Our objective is to mount manipulator on each robot and study the problem of *mobile robot manipulators* cooperation.

Besides the tracking problem, ROMAIN has to cope with asynchronous events because it has no a priori information about the workspace.

The strategy built around fuzzy technics is devised in three parts, three high level tasks :

- the leader tracking (T_task) is based upon the rendez-vous point prediction and provides orientation and velocity according the leader position and processed velocity.
- the obstacle avoiding part (O_task) is realized according 8 ultrasonic sensor data and provides orientation and velocity variation commands.
- a fuzzy fusion module (F_task) gives the final commands to send to the motor system.

The software organization is represented figure 11.

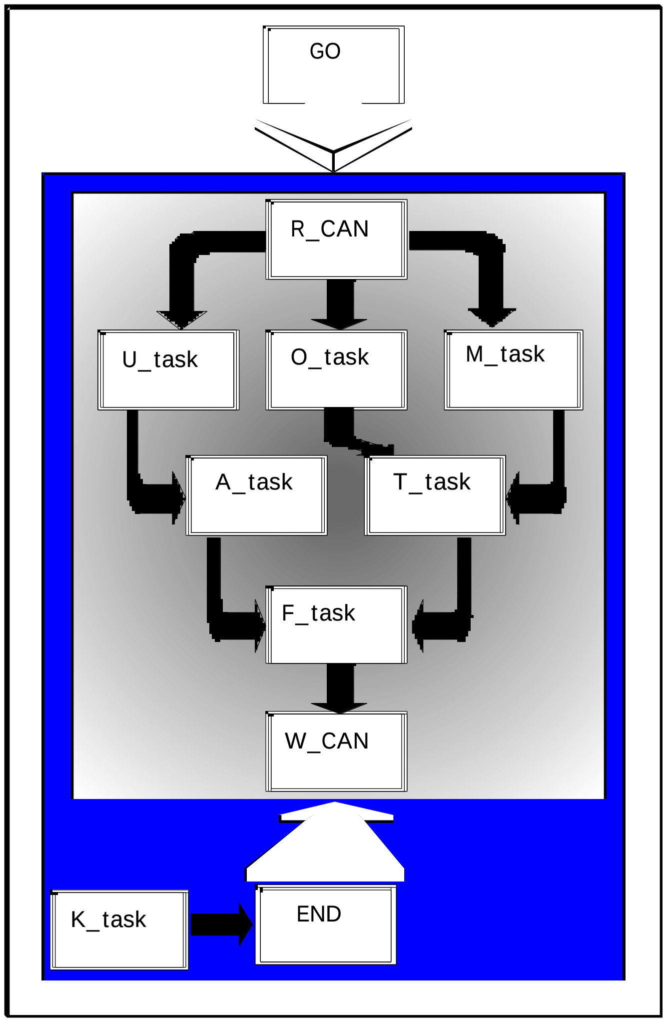

figure 10 : example of tasks organization.

R_CAN receives data from the low level nodes and transmits to the data process tasks which signal the information presence and their origin. These data are used by the high level tasks which compute the new orders written by W_CAN on the CAN network.

Conclusion and perspectives

We described the hardware and software architecture of an autonomous mobile robot. In order to approach to different characteristic problems in mobile robotics our proposed solution is built around CAN network. Indeed, the own properties of CAN, open structure, flexibility, complete communication protocol offers an optimal platform. The network is organized in two layers with specific nodes based on microcontroller cards and industrial PC. We shown in this paper that CAN is a good way to perform parallel processing, essential characteristics in mobile robotics.

In the future we desire to insert a middle layer with 80196 microcontroller in the view to release the high level from process as odometer computation or data fusion. Moreover this evolution in sub-network will allow to answer closely to real time requirement by grouping nodes with same bus access periodicity and so avoid any bottle neck.

The design of this open structure will make easier the adding procedure of cards for the sensor data process as accelerometers for slipping evaluation and inclinometers for the stability in hilly environment.

References

[1] S. Delaplace *Navigation a Coût Minimal d'un Robot Mobile dans un Environnement Partiellement Connu.* Thèse de doctorat de l'Université P. et M. Curie. 1991.

[2] G. Bourdon, P. Ruaux, S. Delaplace *Instantaneous Fuzzy Modeling for Obstacle Avoidance.* Int. Conf. SMC, oct. 1995, Vancouver.

[3] P. Ruaux, G. Bourdon, S. Delaplace *A rapid Mobile Robot Synthesis.* Int. Conf. SMC, Vol. 5, pp. 4073-4078, oct. 1995, Vancouver.

[4] P. Novikoff, R. Chellali, M. Milgram *Navigation of a Mobile Robot with Obstacle and Landmark Recognition by Monocular Vision.* ICARV'90, Int. Conf. on Aut. and Comp. Vision, pp. 480-483, 1990.

[5] P. Coiffet *La Robotique.* Ed. Hermes

[6] S. Hirose *A Study of Design and Control of Quadruped Walking Vehicle.* Int. Jour. of Robotics Research. Vol. 3, N°2, pp. 113-133, Summer 1984.

[7] R. Chatila, G. Giralt *A Task Oriented Navigation System for the Mobile Robot HILARE.* Proc. 2nd Int. Meeting on A.I., oct. 1880, Leningrad.

[8] N. K. M'Sirdi, N. Manamani, N. Nadjar-Gauthier *Modelization and Identification of a Pneumatic Legged Robot.* CESA'96, IMACS Multiconference, July 1996, Lille.