Integrated Architecture for Embedded Control-Command

Denis Genon-Catalot*, André Lagreze*, Jean Berezné**
**** Plate-forme Technologique de Valence***
Université Pierre Mendès-France
***** Laboratoire d'Astrophysique de l'Observatoire de Grenoble***
Université Joseph Fourier

Abstract

This paper presents a low-cost approach that allows distributed control/command architectures to be used for industrial systems. The goal of this project is to be able to run the systems thanks to simulation software, which can be replaced by the real parts as soon as they are built. This generic software can be used in the simulation and evaluation mode and also in the final operational mode.

Our approach can be summarized by the following three key points:

A distributed architecture of smart sensors and actuators, which are binded by a dedicated data network (CAN fieldbus).

An embedded version of Linux, which acts as an an ethernet gateway is developed and allows the supervision of a remote control points.

The connections of new elements are simplified thanks to a classification of sensors and actuators.

When new machine or systems are developed with requirements such as flexibility and real time constraints we chose an open architecture divided in two stages to ensure optimal data transmission security. The low-level activity is ensured by a priority-based fieldbus to connect hardware devices through low cost micro-controllers. The second level interface collects data flow and transfers control from the ethernet supervisor. This interface runs on an embedded PC under Linux operating system (OS). With such an open OS, a very compact and evaluative code has been produced.

A first demonstration model was developed in collaboration with the astrophysical Labs and Observatory of Grenoble. This model has allowed a validation of the proposed architecture for an experimental astrophysical instrument. It is noteworthy that such an architecture can be used in other systems such as those used in the area of digital printing.

Introduction

In the development of embedded systems, it is usually needed to ensure a connection to the low-level hardware with sensors and actuators and at least a link to the control desk. The global designs refer to the same concept but with exponential complexity associated with high costs due to experimental prototypes.

The control-command system that we propose can manage simultaneously electronic or electro-mechanical parts as a sub-function of the system For a new design several heterogeneous components need to be linked together through various links (Ethernet, serial, current loop, ISA, PCI or VME bus…) with the same number of communications protocols. However, such systems have to integrate news components for new application with high performance.

The work presented in this paper proposes a completed design approach, which is built upon smart sensors and actuators, connected through fieldbus. During the design period the system will be able to evaluate the influence of adding or changing devices without breaking down the software development. However, using this approach it is not required for a designer to have a deep knowledge of the internal workflow of the control-command system

Architecture

Three main constraints must be taken into account when this kind of system is designed: architecture versatility, real time and also whole system reliability.

Versatility means that adding and removing new sensors and actuators from the global system should be made possible in an easy way.

Reacting in real time a system reaction, we estimated that the system must react in less than one hundred milliseconds.

Finally, since this kind of system involves (high voltage for the piezoelectric translators or the electric motors, it must be reliable with a high immunity towards electromagnetic interference.

A distributed architecture build upon an embedded network has to fulfill all the above mentioned requirements. Figure 1 give a typical example of such architecture.

The interfaces boards between sensors and network are made up of a hardware part and a software part which implements distributed control-command protocol. These boards will be called “ network nodes ” in the following.

Mechanical, electrical and handshake features of interface between sensor and network node has been normalized to facilitate connection of additional sensors to the network.

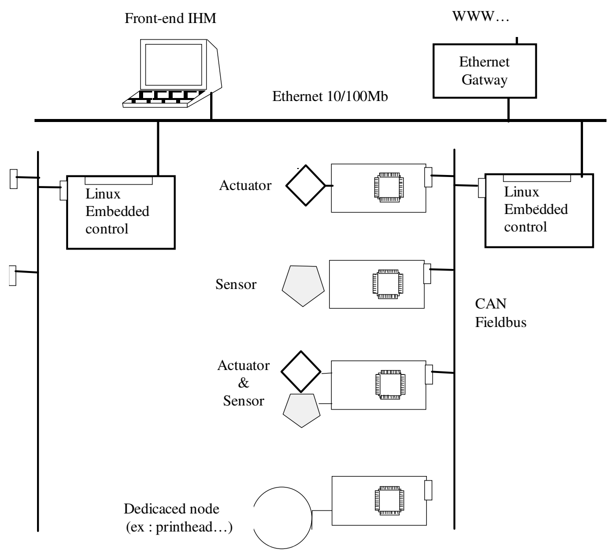

Figure 1. Global architecture

Hardware Design

CAN (Controller Area Network) Outlines

There is a wide variety of fieldbuses, each with its own specificity and its own field of application. Owing to the needed bandwidth for all sensors (more than 200 Kbitsps in raw rate), we should have a fast enough embedded network (in the sense of fieldbuses !) and also a network suitable for rough industrial environment.

With 1 Mbitsps baud rate on a forty meter long twisted pair, real time features and suitability for industrial field, CAN bus proved to be the fieldbus we needed to carry out the project.

CAN the BOSH company in Germany (ISO DIS 1159-2 standard) has developed protocol for interconnections between micro-controller stations. CAN specification is on the physical and data link layers (or more precisely the Medium Access layer). Several other standards normalize upper layers above CAN protocol and, in particular, application layer (DeviceNet, CAL, SDS ...). Our system application layer has been developed by our team in order to use network bandwidth in the best way.

The transmission medium can be a twisted pair (like in our system), a coaxial cable or an optical fiber. Bits are coded in NRZ with bit stuffing. This deliberately limits frame transmission time to meet better real time constraint (about 140 microseconds at 1 Mbitsps rate). The medium access method is CSMA type with nondestructive collisions by bitwise arbitration.

It means that the frame with the highest priority will always access the medium with a latency time shorter or equal than one frame length. This is very important to guarantee the short latency of high priority messages. Priority level directly depends on frame identifier (priority decreases as identifier binary value increases) and therefore, on frame contents. Access control is fully decentralized and every node has in its network controller all of the access mechanisms. This feature gives great network availability because when any network node is out of order, others can still access the medium.

Furthermore, CAN is a very reliable protocol due to several error detection procedures such as 15-bit CRC, acknowledgment bits, frame check and "fault confinement". This last one enables any network node which may be too often out of order to be disconnected from the bus (Bus-Off State). This high reliability is very important for the system we have developed since astrophysical observatories may be very remotely located.

Network Node Architecture

Besides the sensor a network node which binds the sensor to the network is implemented. This is a stand alone computer board with a Infineon SABC5015C micro-controller. This board is driven by an embedded program written in C language which is a part of the application layer. CAN controller is included in the chip, as the message objet in the Intel 82527, because of its advanced management of message objects and its bufferization capabilities (256-byte internal RAM). A external flash memory contains the embedded program and allows the downloading of network node programs through CAN network, thanks to its reprogramming capability. It’s a very useful feature which avoids long sensor dismantling periods. Sensor data are time stamped by network nodes using one of the microcontroller timers with an accuracy of hundred seconds. Microcontroller internal watchdog is utilized to monitor the node and reset the embedded program in case of accidental breakdown in processor instruction cycle. Control informations are locally displayed on a LCD display (identification of embedded program release, validity of sensor-network node interface ...).

Software Architecture

This part describes both the Linux OS kernel choice and the fieldbus control-command protocol. The IHM used will serve oriented software for application front-end as picture designer software is not described in this paper.

Embedding Linux

In our development we needed an operating system which should- support modern programming standard (C, JAVA, ..) ,TCP/IP stack implementation, and easy to program when it comes to writing devices drivers. At least the fact that Linux is highly configurable and comes with the sources makes it as attractive as in free distribution. The OS linux kernel is fully be supported by any small CPU card such as PC104 boards with X86 core. (Advantech, Jumptec…). We use RED-HAT vers. 6.2 (Gnu Public License) non commercial distribution without Kernel modification. For the multi-process software we have implemented PVM (Parallel Virtual Machine) which is originally involved in massive parallel computing.

Control-Command Protocol

Each network node bound to a sensor has its own embedded software in flash memory. This program is made up of two parts: a generic part common to all sensors and a specific part for each sensor. This last one is as reduced as possible thanks to sensor-network interface standardization.

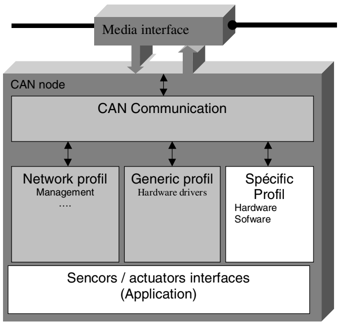

Figure 2. Network node architecture

As already stated, the first application layer we use is not standard, but "laboratory made". In fact, we use a fully decentralized medium access control by exploiting CAN protocol in the best way. A future version of network management and process data flow will be proposed with CAN-Open compliance protocol.

Conclusions

The distributed architecture is now operational. Our system was demonstrated to be a very useful for the demonstrator working at the Astrophysics Observatory in Grenoble (ASCCI project).

Now we are working on the next version of the network management in order to accept node from commercially available sensors or actuators.

In the same way we will propose a signal processor in node network to collapse more complex interface and reduce the bandwidth when designing design control-command system

Presently our work is enlarged towards other research fields especially in the digital printing area ans examples will be shown at the DPP2001 Conference.

Acknowledgements

The authors would like to acknowledge the contributions of Guillaume Mella and the Lork-system company in design, assembly and operation of the experimental apparatus (ASCII project). They also thank Prof. Arthur Soucemarianadin for introducing them to the subject of Digital Printing.

References

1. Nouvelles architectures réparties pour le contrôle commande en astrophysique, Denis Genon-Catalot, André Lagreze, Jean Berezné, Jean-Philippe Galvez , INNOCAP99, April 1999 Grenoble
2. "Environnement multicapteurs pour démonstrateur automobile de détection d'obstacles " A. Lagrèze , D.Genon-Catalot , INNOCAP99, April 1999 Grenoble.
3. "Integrated Environment multisensors data fusion system" A. Lagrèze , D.Genon-Catalot SPIE Intelligent System & Automated Manufacturing October 1997, Pittsburg PA
5. CAN Specifications ver2.0 Robert Bosch Gmbh 1991.

Biography

Denis Genon-Catalot was awarded a Ph.D. in Applied Physics in 1993 at the Joseph Fourier University Grenoble.

From 1994 to 1998 he is the Department Head for the Electronic Engineering Laboratories at ESISAR in Valence. Since 1998 he is with the Department of Networking and Telecommunication of Pierre Mendes-France University at Valence. Founder of the Research and Technology Platform in Valence, his work has primarily focused on embedded systems, including networking and architecture design. He is heavily involved in the technological development programs with industrial partners. He is a member of the SPIE.

Email : Denis.Genon@iut-valence.fr