

DI.SY.RE. – A Demonstrator for Distributed Industrial Systems Remotely Controllable

J. A. Fonseca[†], G.Rodrigues[‡], H.Proença[‡], J.Rodrigues[‡]; L.Barradas[‡], J. Ferreira[‡], M. Calha[‡]
jaf@det.ua.pt, gilnr@mail.pt, henrique@est.ipcb.pt, jorgecr@alunos.est.ipcb.pt, {claudio, jjf, mjc}@est.ipcb.pt

[†] DET-IEETA, University of Aveiro
P-3810-193 Aveiro, Portugal

[‡] EST, Polytechnic Institute of Castelo Branco
P-6000 Castelo Branco, Portugal

Abstract

This paper presents a co-operative work between two Portuguese universities in which a first step towards the development of a demonstrator for a distributed industrial system was attained. This demonstrator consists in a small, portable infrastructure with motors and encoders where experiments ranging from the test of control algorithms to the evaluation of real-time performance of medium access protocols can be done either locally or remotely.

Presently, the electromechanical part of the demonstrator, with three motor/encoder (M/E) pairs, is ready and operating. It is designed so that it can be used by students without any possibility of damage, it can be moved easily by a single person and it can pose challenging problems at least as a distributed system.

Electronic modules connected by a CAN-fieldbus control the motors. A host PC computer provides a local interface with the system. Adequate software tools developed under the project allow the remote control and visualization of the system through Internet. Work in progress includes the remote download of the code for the

microprocessor-based controllers used in the system.

1. Introduction

Fieldbus [1] based distributed systems are an important current topic of research and development namely when they are used as embedded systems [2] in safety critical real-time applications such as the ones in automotive industry.

In order to support teaching in related topics of disciplines like Distributed Systems, Real-Time Systems, Industrial Communication Networks, Embedded Systems (Control Systems, also), it was decided to develop a demonstrator offering similarities to typical industrial systems (e.g. machine tools). However, this demonstrator should present characteristics well adapted to teaching and demonstration purposes: it should be portable, scalable, capable of supporting failures or faults in its control system without destruction or physical risks, capable of posing challenging problems in the fields of interest, enabling assessment of performance by simple visual inspection.

Considering the current interest in the use of Internet to access and control field

devices and systems, the demonstrator also includes this feature. This is also of particular importance since the two institutions involved in the project, the University of Aveiro and the Polytechnic Institute of Castelo Branco, are 200 Km far away from each other.

Besides teaching, the demonstrator was also thought to provide a means to facilitate research in some topics of interest in Aveiro, namely in the field of medium access protocols for real-time performance in fieldbus based distributed systems and its influence in control applications [3],[4],[5],[6],[7].

2. The DISYRE demonstrator

The DISYRE demonstrator integrates three main parts: an electromechanical part with three M/E (motor/encoder) pairs, a distributed system to control each M/E pair and a set of tools to enable the remote access.

The electromechanical part, whose schematic is present in Figure 1 and photograph in Figure 2 is a small device controlled in real-time. It was designed to allow visual inspection of faults and to simulate a machine tool or some parts of a factory. It includes at this moment three DC motors each with an associated bi-phase incremental encoder.

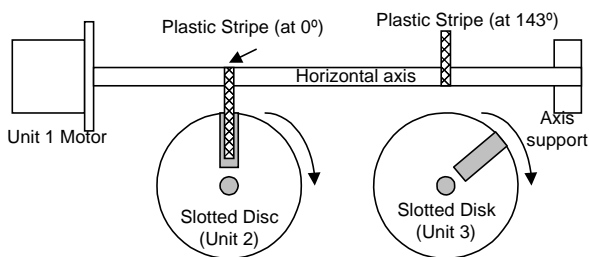


Figure 1. Schematics of the electro-mechanical part

As it can be seen in figure 1, one of the motors drives a 40 cm horizontally mounted rotating axis and the other two motors drive

two slotted disks that are also rotated horizontally. In the horizontal axis it is easy to mount two plastic ribbons that will rotate with it. The ultimate synchronization of the system consists in making the ribbons to pass in the slots of the two horizontal disks. If synchronization fails ribbons will touch the disks but no harm will be done to the parts. It is rather easy to add at least 4 more horizontal disks in this part as well as extend it with identical structures. This poses a challenging problem of synchronization to the system electronics and software. Besides, it is very easy to verify if the system is operating properly. Also, it is possible just to use the system for simple experiments of controlling the rotating speed of each of the individual motors.

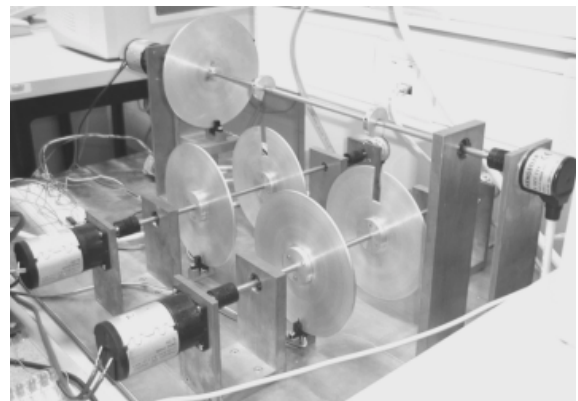


Figure 2. A view of the electromechanical part

The electronics of the demonstrator are organized as a distributed system based in the CAN – Controller Area Network fieldbus [8]. Each node of the system is a microcontroller-based board [9] and the interface electronics for the M/E pairs, a power driver and an encoder digital interface, as depicted in Figure 3. The special feature of the system is that code for the microcontrollers can be downloaded by the fieldbus through a similar board that acts as a gateway [10]. This feature is essential for future applications of the demonstrator.



Figure 3. Microcontroller based boards and interface electronics

The distributed system interacts with a PC, called the Operation PC where a video camera is also installed. This PC runs a software program, the Control Application, which sends commands to the CAN nodes and gathers data from the fieldbus. These commands can be issued either locally or remotely. In this case they come from a so-called Server Application that runs in a web server located in the campus of the Polytechnic Institute of Castelo Branco. Remote clients can then access the system through an Internet Browser that runs Java Applets that allow the opening of a TCP/IP connection to the Server Application.

At this moment the DISYRE demonstrator is only able to remotely change and monitor the rotational speed of the motors. The programming of the nodes can only be done in the operation PC but the work in progress will enable remote programming and control also. The first remote client to access the system can log in as controller [11] and can change the speeds. Several other clients

can, nevertheless, access the system and receive periodically the speed values. Real-time visualization of the system operation can be obtained through Real Video. The global system architecture and the remote control scheme used can be viewed in Figure 4.

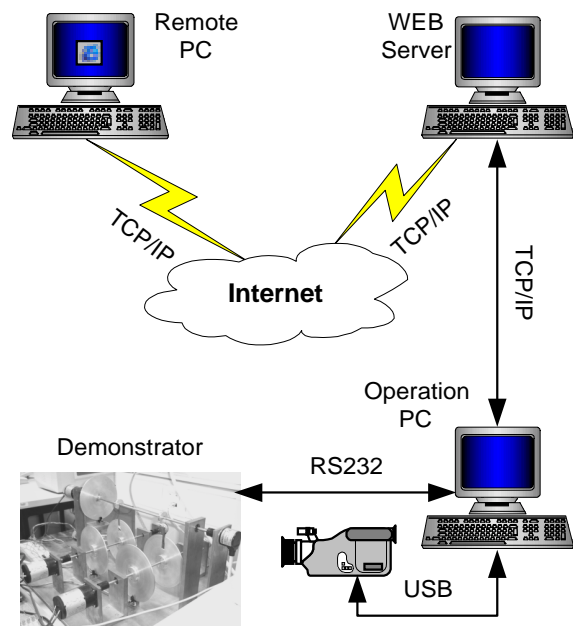


Figure 4. Global system architecture

3. The remote access infrastructure

The network architecture adopted is based on the client/server model as illustrated in Figure 5. According to Figure 5, a transaction is initiated by the remote PC, which requests a service to the web server. Then the web server transfers the request to the operation PC (the application server). This solution is simple and transparent, however if there was a firewall or a proxy server between the web server and the operation PC, then this simple scheme would not work, because the intranet would be protected and no transaction could be initiated outside it. Today and for security reasons, the existence of firewalls or proxy servers is almost mandatory. So this solution cannot be considered.

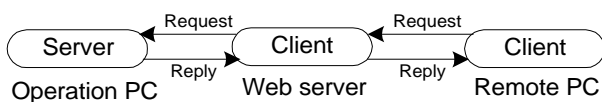


Figure 5. Client/server communication example

In the presence of a firewall or a proxy server, a transaction with a machine outside the intranet has to be initiated by a machine inside the intranet. So, the system architecture must be different than the one depicted in Figure 5.

The solution is changing the operation PC, formerly the server, to be client and the web server becomes the system server. Since the web server is placed upstream the firewall or the proxy server, all transactions must be initiated by the operation PC. This initialisation opens a channel through the firewall. After this setup, a client outside the intranet may now send requests to the Web server. The Web server acts, in this case, as a mediator between the remote client and the operation PC. This situation is illustrated in Figure 6.

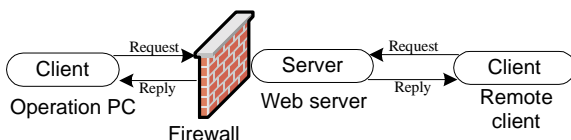


Figure 6. Implemented communication scheme

The main characteristics of the implemented communication model are:

- **Portability** – The operation PC and the web server can be easily replaced.
- **Flexibility** – It is possible to operate inside firewall or proxy server protected intranets.
- **Security** – A domain is used to access control.

Several applications were developed to implement the previously described

architecture. The overall layered structure of the software is presented in Figure 7 (the demonstrator internal software architecture is not included).

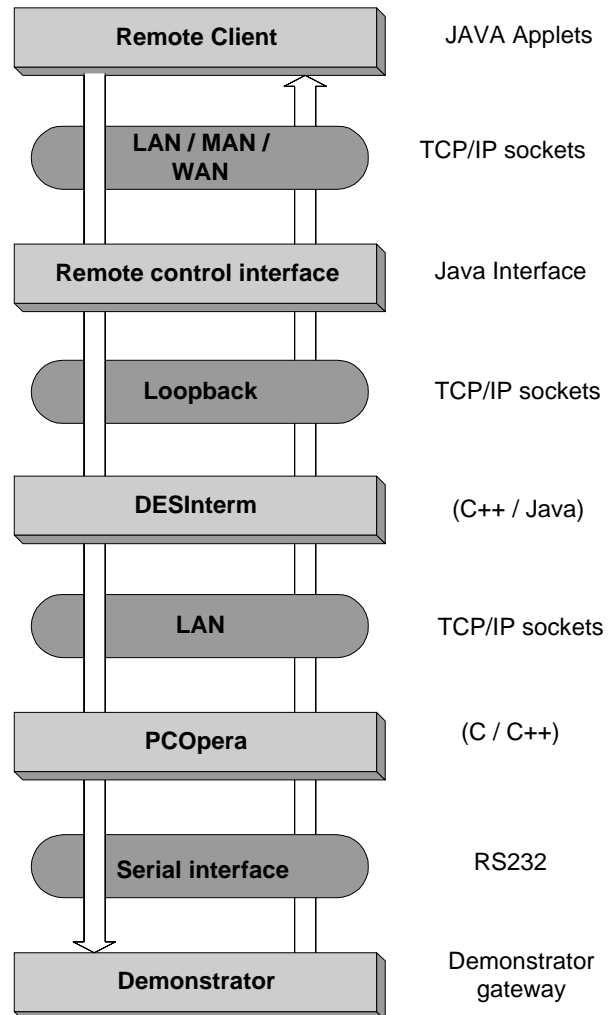


Figure 7. Layered structure of the implemented software

The application running in the operation PC (*PCOpera*) communicates with the demonstrator gateway through the serial port and with the internet server by TCP/IP sockets.

The software running on the Internet server (*DESIntern*) enables the communication between the remote PCs and the operation PC via TCP/IP sockets.

The remote control interface, to be executed in remote client PC, is made from several JAVA applets located in the internet service provider of the internet server.

The communication among these applications is based on a set of predefined messages.

4. The local distributed system

On the demonstrator side, there are two different types of nodes: the gateway and the controller nodes. The software developed reflects this situation. The gateway, besides controlling one axis, is also responsible for the interfacing between the demonstrator and the operation PC so, on top of axis control algorithm, a set of interfacing related routines were also implemented.

The developed software (Figure 8) provides interrupt based event detection for sensor reading and message processing and a global time base for motor control and synchronization. The software running in each node, starts by initialising the microcontroller and placing the axes in their initial position, afterwards the axes begin to rotate at the specified speed and from then on all subsequent actions are triggered by interrupts (timers interrupts for sensor reading and interrupt based message processing)

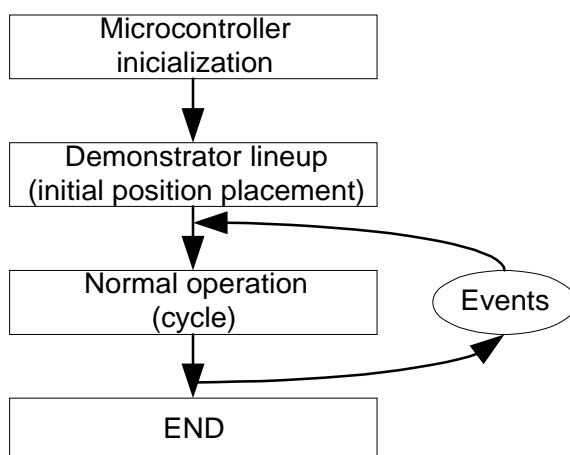


Figure 8. Demonstrator software

The messages processed by the gateway node are of two kinds: the CAN messages and the RS232 messages. The messages format was fully specified foreseeing the implementation of several control protocols.

5. Work in progress

The DISYRE system is yet in an earlier stage of development. Three lines of work in progress are being followed: enhancing the remote program download capabilities; development of a test bed for studies on control algorithms, including distance learning; demonstration and research of real time distributed systems.

In the first line the main improvement consists in allowing the download of programs from a remote client. This client sends the program to the Operation PC that, in turn, distributes it to the relevant nodes of the local CAN based distributed system. To do this a scheme to remotely change the operational modes of the nodes between *download* and *run* must be added. Also the techniques developed in [10] are being adapted to include new functionalities such as broadcast/multicast of fragments or entire programs and remote verification of program memory contents.

The rotating axes of the demonstrator can also be used as a test bed to teach and test control algorithms. In this case, at least in an undergraduate level, the details of the underlying architecture must be encapsulated so that the focus is on control issues. Two functionalities are then useful. One is the possibility to operate individually each of the M/E pairs so that it becomes possible to compare controller's performance. The ribbon contact, due to the absence of synchronization, acts then as a load disturbance. The second functionality consists on the isolation between the controller code and all the other node specific code (M/E driver, fieldbus drivers, etc). This can either be done at the client side, which implies the download of a full program, or at the node level using a resident kernel capable of accepting remote parameter configuration

(e.g. sampling period) and download of specific modules (e.g. the controller itself). The visual inspection is very useful at this level, so further improvements on image quality should be devised.

In the scope of the third line of work, the installation of medium access control protocols to improve real-time operation, namely the Flexible Time-Triggered CAN protocol [3] is being finished and will be used to achieve full synchronization at high rotational speeds. Also, the demonstrator will be used to show and study a novel session layer protocol, named Time-Triggered CAN, which standardization is now being finished by a task force working under ISO [12]. It will be possible to compare the performance of the overall distributed system under these or even other MAC protocols. The local part of the demonstrator will also be used to obtain experimental measures.

It should also be noticed that most of this work is being carried on by last year and graduate students in both Institutions.

6. Conclusions

In this paper, it was shown that the DISYRE demonstrator is a very useful tool for education and research in the fields of real time systems, distributed systems, control systems, embedded systems and also internet based automation. The development of several new features of the demonstrator is, on one side, a challenging activity for students to learn by problem solving methodologies and, on the other, provides a set of useful tools that can be used on distance education.

Some important characteristics of the demonstrator, besides remote access capabilities, are the portability and robustness of the local electromechanical equipment. The similarities with real industrial systems and the easy verification of performance are also valuable features.

Besides the work in progress described previously, some issues, as dependability

and security will also be addressed in the near future.

References

- [1] J.P. Thomesse, "A Review of the Fieldbuses", Annual Reviews in Control, 22 pp. 35-45, 1998.
- [2] H. Kopetz, "Real-Time Systems: Design Principles for Distributed Embedded Applications", Kluwer Acad. Publ., 1997.
- [3] L.Almeida, J.Fonseca, P.Fonseca "A Flexible Time-Triggered Communication System Based on the Controller Area Network" FeT '99 - Fieldbus Systems and their Applications Conference, Magdeburgo, Germany, September 1999.
- [4] P. Pedreiras, L. Almeida, "Combining Event-Triggered and Time-Triggered Traffic in FTT-CAN: Analysis of the Asynchronous Messaging System", Proc. WFCS'2000 – 3rd IEEE Int. Workshop on Factory Communication Systems, Porto, Port., 5-8 Sept. 2000.
- [5] F. Coutinho, J.A. Fonseca, J. Barreiros, E. Costa, "Jitter Minimization with Genetic Algorithms", Proceedings WFCS'2000 – 3rd IEEE International Workshop on Factory Communication Systems, Porto, Portugal, 5-8 September 2000.
- [6] L. Almeida, J.A. Fonseca, "FTT-CAN: a Network-Centric Approach for CAN based Distributed Systems", 4th IFAC Symposium on Intelligent Components and Instruments for Control Applications (SICICA'00), Buenos Aires, Argentina, 13-15 September 2000.
- [7] A. Mota, J. A. Fonseca, "Systems Modeling and Identification in CAN based Distributed Control Systems", DCCS 2000: 16th IFAC Workshop on Distributed Computer Control Systems, Sydney, Australia, 29 November-1 December 2000.
- [8] Bosch, "CAN specification version 2.0 - Tech. Report", Bosch GmbH, Stuttgart, Germany, 1991.
- [9] <http://www.microio.pt/produtos/canivete/canivete.htm>
- [10] P. Fonseca, F. Santos, A. Mota, J.A. Fonseca - "A Dynamically Reconfigurable CAN System" Proceedings ICC'98 - 5th International CAN Conference'98, S. Jose, USA, 8-9 November, 1998.
- [11] <http://www.est.ipcb.pt/laboratorios/ari/projectos/19992000/desire/>
- [12] ISO - International Standards Organisation: Time-Triggered Communications on CAN: TC 22 / SC 3 / WG 1 / TF 6