

A Bluetooth-based Wireless Distributed Data Acquisition and Control System

Ezequiel T. Coelho, Paulo Carvalhal, Manuel J. Ferreira, Luís F. Silva,
Heitor Almeida, Cristina Santos and José A. Afonso

*Department of Industrial Electronics,
University of Minho, Guimarães, Portugal*

{pcarvalhal,mjf,cristina,jose.afonso}@dei.uminho.pt, {lffsilva,heitor}@dem.uminho.pt

Abstract - This paper presents an application independent embedded platform for a wireless distributed data acquisition and control system, and describes its application in the scenario of controlling the information processing and communications between sensors and actuators onboard of an autonomous flying robot, in a “fly-by-wireless” approach. The system, which was designed and implemented, comprises a set of nodes composed by microcontrollers, wireless communication modules based on Bluetooth technology, and sensing/actuation devices. This paper also presents several experimental results which denote encouraging performance characteristics for the developed system, not only in the context of the proposed application, but for other wireless applications as well.

Index Terms - *Sensor Networks, Distributed Systems, Mobile Robots, Wireless Networks.*

I. INTRODUCTION

Due to its strict requirements in terms of delay and loss, the traffic of distributed data acquisition and control systems is normally supported by specific cabled networks which are known generically by the term fieldbus [1]. These networks are used instead of generic local area networks (LANs), because they possess characteristics that enable the provision of the QoS guarantees required by distributed control systems and other industrial applications, which present more demanding real-time requirements than conventional data communication applications, allied to a low tolerance to errors.

In this context, the replacement of conventional cabled networks by a wireless network introduces several advantages derived from the elimination of the cables, such as the mobility of the stations, opening a whole new range of potential applications. Other advantage is that the deployment of the network becomes much more flexible, faster and easier.

Currently there are many different wireless local and personal area network technologies available in the market. Section 2 presents an overview of these technologies and discusses their suitability to provide support to distributed data acquisition and control systems. Based on this analysis, we chose the Bluetooth wireless network for the development and implementation of the proposed platform. We use commercial-off-the-shelf Bluetooth modules which enable simple and fast integration of the whole system.

The developed wireless system is being used onboard of an unmanned aerial vehicle (UAV). The network comprises one master station, connected to the flight controller, and up to seven slave stations spread along the aircraft structure and connected to several sensors and actuators.

This developed system is integrated in a global project that involves the design and development of the aerial platform, as well as the electronics, communications hardware and software, flight control, artificial vision and systems integration, in order to provide autonomous takeoff, flight mission and landing maneuvers.

Next sections are organized as follows: section 2 presents an overview of relevant wireless network technologies in the context of this paper; section 3 discusses design and implementation issues of the developed platform; in section 4, the application of the platform onboard of an UAV is discussed; section 6 presents experimental performance results obtained with this system and section 5 presents the conclusions.

II. OVERVIEW OF WIRELESS NETWORKS

The most widespread wireless local area network (WLAN) technology currently available is based on the IEEE 802.11 standard [2]. These networks are available on multiple physical options and operating frequencies bands. However, all these versions use the same MAC (Medium Access Control) protocol; a contention based CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) mechanism know as DCF (Distributed Control Function), which is unable to provide the QoS guarantees required by distributed data acquisition and control systems due to the high probability of collisions, specially under high load conditions.

In order to support real-time traffic, the 802.11 standard defines an alternative MAC protocol know as PCF (Point Coordination Function) which is based on a polling scheme. This protocol is capable of providing the QoS guarantees. However, unlike the DCF protocol, the implementation of PCF is not mandatory, and the availability of products that support it is scarce. More recently, a newer standard, the IEEE 802.11e [3], designed to improve the efficiency and QoS support of 802.11 was released, but its availability on the market is also low.

Concurrently to the development of the 802.11, another wireless LAN (WLAN) standard, HIPERLAN/2 [4], was

developed by the ETSI. HIPERLAN/2 networks are designed to operate at the 5 GHz band using OFDM (Orthogonal Frequency Division Modulation). Its physical layer is similar to the one used by the IEEE 802.11a due to agreements made by the two standard bodies. On the other hand, the MAC protocols used by these networks are radically different. HIPERLAN/2 uses a demand based dynamic TDMA (Time Division Multiple Access) protocol, which is able to provide extensive support of QoS to multiple types of traffic, including those generated by distributed data acquisition and control systems [5]. However, the 802.11 standard won the battle for the wireless LAN market and as such no available HIPERLAN/2 products are known at the moment.

Due to design characteristics, 802.11 and HIPERLAN/2 present relatively expensive, bulky and power hungry modules. Although these networks can be suitable to interconnect devices like computers, there is an enormous potential market to provide wireless communication capabilities to smaller and cheaper devices running on batteries without the need of frequently recharging. Such devices include computer peripherals, biomedical monitoring appliances, surveillance units and many other sensing and actuation devices. To provide communication capabilities to such devices, various low cost short range networks, known collectively by the term personal area network (PAN), are being developed.

At the IEEE, the task of standardization of PAN networks is under the scope of the IEEE 802.15 group. One of these standards, the IEEE 802.15.4 [6] defines the physical and MAC layer of the ZigBee networks, which aim to provide a low power and low bit rate PAN with the main purpose of enabling sensor network applications. At the physical layer, the IEEE 802.15.4 relies on direct sequence spread spectrum (DSSS) to enhance the robustness against interference, and provides gross data rates of 20/40 kbps, at the 868/915 band, and 250 kbps, at the 2.4 ISM band. As in the 802.11 networks, the ZigBee MAC protocol is a contention based CSMA/CA mechanism.

A complementary scheme defined in the 802.15.4 MAC standard is designed to provide some QoS guarantees to real-time traffic. With this scheme enabled, the PAN coordinator defines a superframe structure containing a contention access period (CAP), where a slotted CSMA/CA mechanism is used, and a contention free period (CFP), where specific devices can transmit using guaranteed time slots (GTSs) granted by the PAN coordinator and announced previously.

One of the most popular ZigBee platforms is the MICAz mote [7]. Currently, the software shipped with these devices implements the CSMA/CA portion of the IEEE 802.15.4 standard, but not the GTS mechanism. As the CSMA/CA protocol is unable to provide the desired QoS guarantees for the distributed data acquisition and control application, the utilization of the motes becomes conditioned to the modification of the original software in order to introduce either the GTS mechanism or other MAC mechanism with QoS support.

Bluetooth [8], which is another PAN technology, operates in the 2.4 GHz ISM band using frequency hopping spread spectrum (FHSS) and provides a gross data rate of 1 Mbps. It operates using a star topology formed by one master and up to seven active slaves, and the transmissions can achieve a range of 10 or 100 m, depending of the class of the device. At the MAC layer, the Bluetooth devices uses a polling based protocol that provides support for both real-time and asynchronous traffic.

Bluetooth provides better overall characteristics than the other networks discussed here for the desired application. It drains much less power than the WLANs. It uses a MAC protocol that provides support for real-time traffic. It provides a higher gross data rate than ZigBee. Bluetooth spread spectrum covers a bandwidth of 79 MHz while ZigBee operates in a band of less than 5 MHz, what makes the former more robust against interference. Moreover, Bluetooth provides an adaptive frequency hopping mechanism that avoids bands affected by interference, which increase the robustness against interference even more.

III. PLATFORM DESIGN AND IMPLEMENTATION

We have chosen to interface with the Bluetooth stack at an application level. At this level of interface, all communication services are built on top of a Virtual Machine (VM), so Bluetooth stack details can be ignored and focus can be maintained on the application itself. The manufacturer's VM [9], implements a wireless multidrop access scheme where the master receives all frames sent by the slaves and all slaves can listen to the frames sent by the master, in a point-to-multipoint topology.

A. Physical Details

The physical part of the platform is built around a low power Texas Instruments MSP430 microcontroller, a Von-Neumann 16 bit RISC architecture with mixed program, data and I/O in a 64Kbytes address space. Besides its low power profile, which uses about 280 μ A when operating at 1MHz@2.2Vdc, MSP430 offers some interesting features, like single cycle register operations, direct memory-to-memory transfers and a CPU independent hardware multiplication unit. From the flexibility perspective, a flexible I/O structure capable of independently dealing with different I/O bits, in terms of data direction, interrupt programming, and edge triggering selection, two USARTs supporting SPI or UART protocols, and onboard 12 bit SAR ADC with 200 kHz rate, beyond PWM capable timers, are all relevant features.

The Bluetooth modules we use are connectBlue OEM serial adapter devices. The master unit is an OEMSPA33i module and the slave units are all OEMSPA13i modules, both including an integrated antenna. We plan to use external antennas in future versions of the platform, to be able to shield the modules in order to increase the reliability of the system against electromagnetic interference.

Fig. 1 displays the physical layout of a slave node. The Bluetooth module is in the upper left corner and the

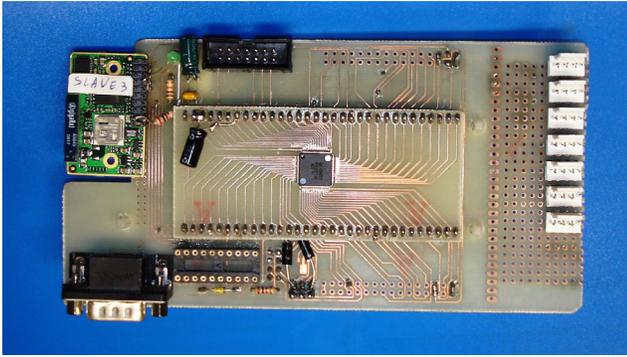


Fig. 1 Layout of a slave node of the network.

microcontroller is in the center of the board. Since it's only an experimental design, the board layout is not optimized yet. In terms of physical construction, the master node resembles the slave nodes, with the size of the Bluetooth module (which is larger in the master node) being the only visible difference between them. At this stage of design, no special care was taken with noise and interference issues, in spite of the known critical importance of this subject. In a future board release, layout, electromagnetic emission and compatibility will be of major concern. As the form factor of the board can be significantly improved, we expect to develop a new board comparable (in dimensions) to the Bluetooth module.

Mainly for debug purposes, a RS232 compatible serial connection is also available in each board, for local connection with an external monitoring computer (connector on lower left corner).

The module used on the slave nodes (OEMSPA13i), has a limitation, compared to the module used in the master node (OEMSPA33i), which is the fact that it allows only three simultaneous connections, while the OEMSPA33i allows up to seven simultaneous connections are possible. But since the slaves only need to establish one connection (to the master), this limitation does not represent a constraint to the system. The modules have the possibility to automatically connect themselves to a determined master node at startup. This functionality, among others implemented in the module's Virtual Machine, allows the system to provide a serial interface to the microcontroller, hiding the complexity of the Bluetooth stack, so it has only to care about character transmission, reception and processing.

A. Logical Details

The logical architecture developed is a two layered state machine implementation, composed by the transport layer and the application layer. The transport layer provides a packet delivery service, under control of master unit, capable of transparently deliver data packets across the network.

The transport layer is application independent, and interfaces with the top level application layer by means of a data space for buffering and a set of signaling control bits that allow total synchronization between the two layers. The hierarchy and signaling between the two layers is represented in Fig. 2.

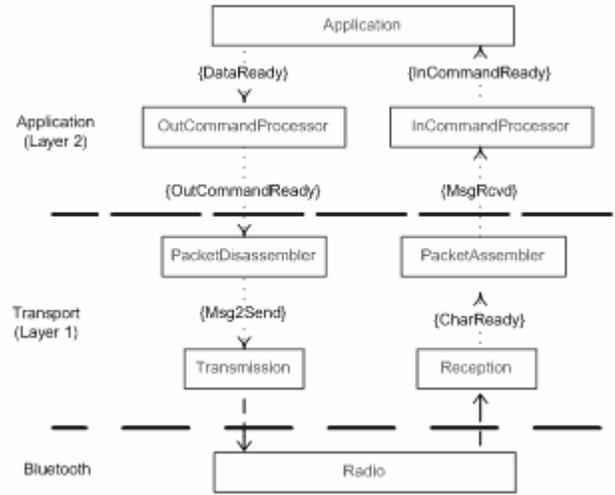


Fig. 2 Hierarchy and signaling at the logical level of the platform.

Walking from lower level to the upper level, it can be seen that asynchronous reception process delivers characters to upper processes. CharReady condition goes TRUE every time a new character arrives to the interface. The next process in the hierarchy is PacketAssembler, a state machine that performs packet re-assembly, reconstructing the original packet from a group of segments, and delivers packets for the next process in the hierarchy. When MsgRcvd (message received) goes TRUE, a new message is ready for processing. Thus, for incoming data, the model at layer 1 receives characters and delivers ready-to-process messages to the application layer.

When the application layer understands that the message is ready to process, a command processor for incoming messages is activated in order to decode the embedded command and semantics contained in the message, to eventually execute some action, and to pass relevant information for the final application.

For outgoing data, the resident application eventually makes available some data to transmit to the master, signaling this event with a DataReady signal. This causes the output command processor to execute its cycle, preparing one message to be sent. When the message is ready, OutCommandReady goes TRUE, signaling to the lower layer that there is a message to send to the network.

At this phase, frame segmentation starts (if needed) by means of a state machine for packet disassembly. This state machine breaks the original message in smaller segments prepared to be serialized. Each time a segment is ready to be sent, Msg2Send goes TRUE and serialization is triggered. So, for outgoing data, the transport layer receives messages from application layer, and sends segments to the radio module in order to be sent over the wireless medium.

The upper layer (application layer) is application dependent, and has no knowledge of the lower layer internal mechanisms, just the services and interfaces available from it. For the aircraft application, its main goal is to replicate a system table among all network nodes, at the maximum

frequency possible. This system table maintains all critical system values, describing the several sensors and actuators, status parameters, and loop feedback information.

Each network node is mapped to a table's section, where all related variables from sensing, actuators and metering are located. This layer is responsible for cyclic refreshing the respective table contents (based on local status), and also for cyclic actuation according to data sent from the master node (flight controller orders). This way, the whole system is viewed as a resident two dimensional array located at master, with different partial copies distributed and synchronized among slave nodes.

IV. THE UAV APPLICATION SCENARIO

Traditionally, UAVs present a processing system architecture consisting of one centralized and complex unit, with one or more CPUs, to which the instrumentation devices are connected by wires. At the same time, they have bulky mechanical connections. One of the main innovative aspects of the proposed platform concerns the flexible and distributed architecture of the onboard system, where the connections between the instrumentation and processing units are made using Bluetooth wireless communication technology. This revolutionary approach can be dubbed "fly-by-wireless". At the same time, the traditional monolithic processing unit is, in this approach, replaced by several less complex units (nodes), spread out over the airplane. In that way the nodes are placed near the sensors and near the controlled surfaces, creating a network of nodes with the capacity of data acquisition, processing, actuation and communication over a wireless platform.

The main advantages of this approach are the following: a) Reduction of the weight, which is achieved by the suppression of several physical connections; b) Easy and quick installation procedures; c) High flexibility drills, allowed by the easier mixing of the aileron and the elevators commands; d) Less cables maintenance; e) Less electromagnetic interference induced on wires; f) Small energy consumption; g) High modular development, allowed by the logical architecture of the nodes.

All the Bluetooth modules our platform are configured in non discoverable mode. The node discovery process of Bluetooth is a slow process, in the order of seconds, however it is not a problem since the master stores the addresses of all slave nodes that should participate in the piconet, so this process is avoided. As the piconet formation is performed on the ground before the takeoff procedure, the associated delay does not constitute a problem as well.

Given that free space propagation loss is proportional to the square of the distance, it is not expected that our network will either suffer or induce interference on other networks operating in the same ISM band, such as the widely deployed WiFi networks, since the former operates in the sky most of time, while the later are normally based on the ground.

The use of Bluetooth technology limits the piconet operation to a maximum of seven slaves. This limitation is not

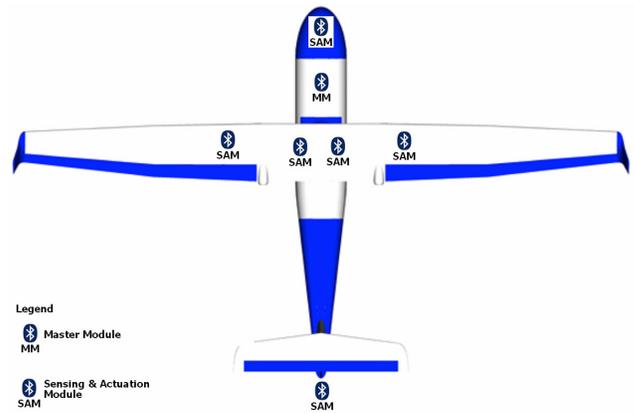


Fig. 3 Distribution of the Bluetooth nodes on the aircraft structure.

of major concern on this application, and can only impose some restraints if node number should be raised, which is not the case.

Fig. 3 shows distribution of the Bluetooth nodes on the aircraft structure. The master module (MM) is placed at the fuselage body, and as described earlier acts as the network and flight controller, onboard data logger, and communications controller for the link with the ground station. The other six nodes (Sensing & Actuation Modules - SAM) are slaves in the network. On each wing, there is a node for an electrical propulsion motor and for control surfaces like ailerons and flaps. Motor speed control, feedback loop, and operating temperature monitoring are responsibility of these wing nodes, as well as control surfaces actuation and position feedback. At the tail, there is another module for elevator and rudder control, and position feedback.

In the fuselage body, we have another two additional SAM type nodes, one for GPS and the other for electronic gyroscopes, which assure some basic information assessment for navigational purposes.

At the front node is the nose probe, which consists of a proprietary design based on six independent pressure sensors that give valuable digital information for flight control. Also in this node the information from an ultrasonic probe provides support for the automatic take-off and landing system.

Fig. 4 presents an overview of the global distributed system and identifies the relative positioning of each model layer as well as the service interactions. At the application layer, the focus is on system table replication and refreshing among all network nodes. Also at this layer all actuation and sensing is performed locally, without specific knowledge of network underneath.

Although the transport and application layers of all network nodes are similar, the master node includes an extra layer (to be implemented on an additional processor) for the flight controller. This layer decides what types of actions are to be taken at any given time, according to the system table contents maintained by the application layer.

Also included on the master logical architecture is the air-ground link, designed to provide permanent bidirectional

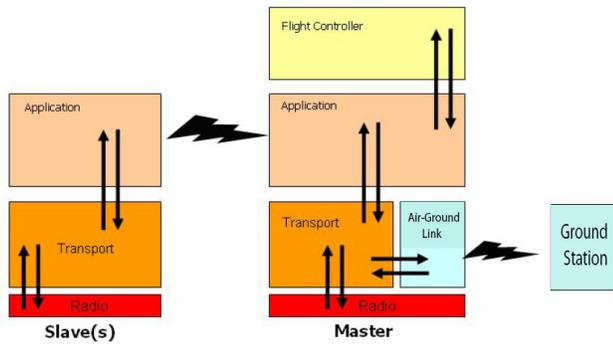


Fig. 4 Global system architecture.

radio connection with a ground station. At the ground station there is a flight instrumentation system with real-time telemetry readings from all system internal parameters, as well as simulators, remote aircraft control and other functions.

V. EXPERIMENTAL RESULTS

The performance of the developed system was measured in the laboratory under several conditions. The experimental setup used to produce the results presented in this section consists of 6 slaves that send uplink data periodically to the master at the same predefined sampling rate. Each sampling packet has a length of 15 octets, which is the maximum packet length supported by the transport layer due to a limitation imposed by the virtual machine used by the Bluetooth module.

Fig. 5 presents the aggregated uplink throughput that reaches the master node as a function of the sampling rate at the 6 slaves. Since Bluetooth uses a contention-free MAC protocol, the uplink transmissions are not affected by collisions, so the network throughput increases linearly with the offered load until the point it reaches saturation, which in this scenario corresponds to the situation where all the 6 slaves transmit data at sampling rates higher than 200 Hz.

Fig. 6 presents the packet loss ratio (PLR) averaged over the 6 slaves as a function of the sampling rate. As the figure shows, the PLR is limited to less than 0.5 % in the region where the network is not congested, but increases rapidly after the saturation point.

Next results concern to the delay experienced by the packets as they travel from the slaves to the master. Fig. 7 presents the mean, standard deviation, minimum and maximum values of the delay registered for one slave (results from other slaves are similar) as a function of the sampling rate. Each data point is calculated from a set of 1000 collected samples. There are no significant variations on the delay statistics for the range of sampling rates presented, which means that the delay is not adversely affected by a rise in the offered load, as long as the network operates below the saturation point. The minimum, mean and standard deviation of the delay are around 8, 27 and 16 ms respectively.

The previous results show that the system is able to reach a sampling rate of 200 Hz for each of the 6 slaves simultaneously (which represent a total of 1200 samples each

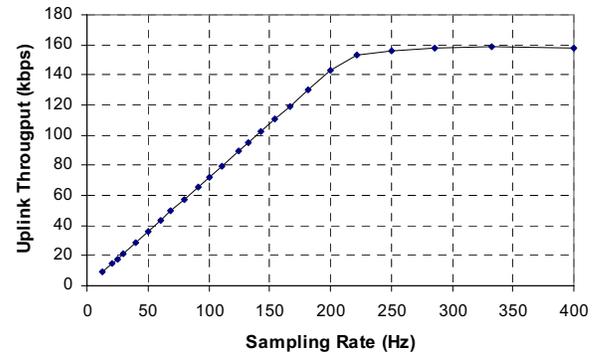


Fig. 5 Uplink throughput as a function of the sampling rate.

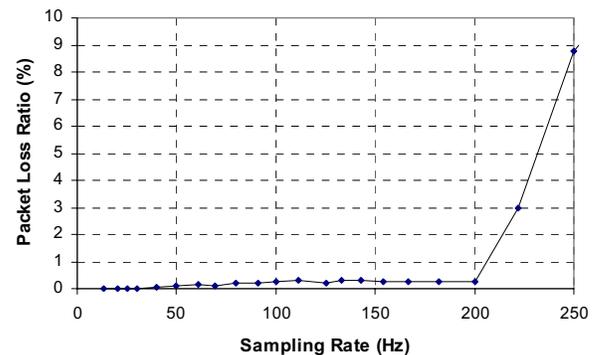


Fig. 6 Packet loss ratio as a function of the sampling rate.

second) without performance degradation in terms of throughput, loss or delay, which denotes a very good performance figure.

Fig. 8 presents the complementary cumulative distribution $P\{X>x\}$ for the random variable X , which represents the delay for all the samples collected using sampling rates in the range from 0 to 200 Hz. This chart allows us to see the probability that the delay exceeds a given deadline. We can see, for instance, that only 1 % of the samples suffer a delay higher than 90 ms.

VI. CONCLUSIONS

In this work we have described a short range wireless platform whose main application is to provide wireless communication for a distributed data acquisition and control system onboard of an UAV.

The distributed active network elements, with sensing and actuation capabilities, while maintaining a classical controller function in a master node, lead to a distribution and reduction of overall weight, derived from the utilization of wireless technology and consequent elimination of data cables, which also removes some aircraft design and construction constraints.

The Bluetooth's spread spectrum mechanism and its adaptive frequency hopping capabilities are paramount to guarantee reliable communications by adapting to dynamic spectrum interference conditions during aircraft operation.

REFERENCES

- [1] J.-D. Decotignie, "Industrial Electronics Handbook", Section 18.4 - Fieldbus, CRC Press, 1997, pp. 403-408.
- [2] IEEE 802.11 (ISO/IEC 8802-11:1999), "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 1999.
- [3] IEEE 802.11e draft/D4.0, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)", November 2002.
- [4] ETSI TR 101 683 V1.1.1, "Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; System Overview", February 2000.
- [5] J. A. Afonso e J. E. Neves, "Fast Retransmission of Real-Time Traffic in HIPERLAN/2 Systems", *Advanced Industrial Conference on Telecommunications AICT2005*, Lisbon, Portugal, July 2005.
- [6] IEEE Std 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), 2003.
- [7] J. Hill, M. Horton, R. Kling, and L. Krishnamurthy, "The Platforms Enabling Wireless Sensor Networks", *Communications of ACM*, vol. 47, no. 6, 2004.
- [8] Bluetooth SIG, Specification of the Bluetooth system, available at <http://www.bluetooth.com/>.
- [9] connectBlue, "Serial Port Adapter – 2nd Generation", User Manual, 2003, available at <http://www.connectblue.com/>.
- [10] Coelho et al., "Autonomous Platform for Distributed Sensing and Actuation over Bluetooth", *7th Portuguese Conference on Automatic Control*, Lisbon, Portugal, September 2006.

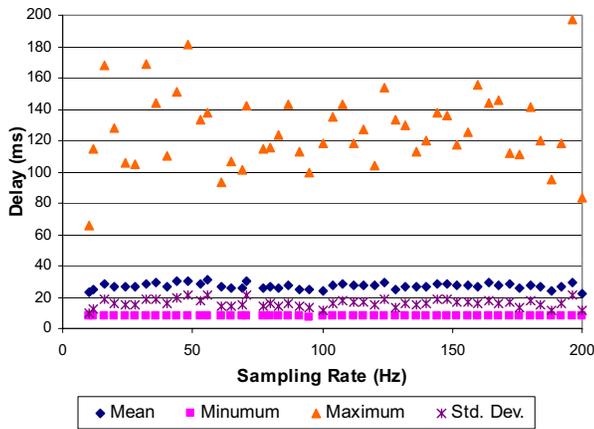


Fig. 7 Delay statistics from a slave to the master as a function of the sampling rate.

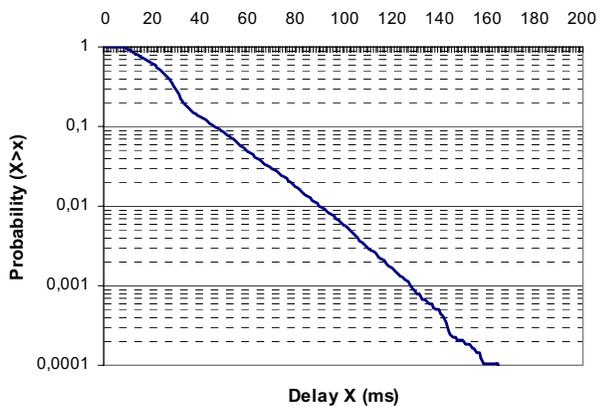


Fig. 8 Complementary cumulative distribution for the delay, calculated using all samples in the range from 0 to 200 Hz.

The use of the current VM provides the best way to focus on application design, by hiding Bluetooth protocol stack issues. The application oriented software design and a proprietary operating system, results in an efficient behavior with low computational power. The state machine supported design leads to an effective way of software development cycle, from design to deployment.

The experimental performance results are very satisfactory, pointing to the suitability of the developed platform for the proposed application. Nevertheless, further tests are needed in order to evaluate the behavior of the system under more harsh interference conditions, as well as in a real scenario onboard of the aircraft (which is currently being built).

This platform was designed to be application independent, and this fact was tested and validated by building other totally different applications over the same networking platform. In that way it has also been successfully applied to other areas, such as the control of agricultural greenhouses [10], and it was recently presented at the Hannover Fair 2006 in the context of a biomedical body monitoring system application.