
DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE COMUNICACIONES BASADO EN CAN PARA LA AVIÓNICA EN UN VEHÍCULO AÉREO AUTÓNOMO NO TRIPULADO

•
Jairo Miguel Vergara Díaz*
Andrés Yovanny Agudelo Toro**

•
Recibido: 17/04/2007
Aceptado: 02/05/2007

Resumen

La necesidad de diseñar el sistema de comunicaciones para la aviónica de un mini helicóptero robot basada en la arquitectura distribuida CAN es la propuesta presentada. El sistema de comunicaciones involucra los aspectos de hardware y software necesarios para permitir el intercambio de datos sobre una red o bus de aviónica desde los sensores y/o hacia los actuadores con el computador central o computador de vuelo. La principal característica de la arquitectura es que permite escalabilidad en la agregación de nuevos dispositivos, garantizando los requerimientos temporales necesarios para la adquisición de datos. Se presentan resultados de intercambio de datos sobre la red de aviónica mostrando las frecuencias de operación alcanzadas.

Palabras clave

Aviónica, CAN, Control, UAV.

* Ingeniero en Instrumentación y Control, especialista en Teleinformática y magíster en Ingeniería Informática, profesor de tiempo completo, programa de Ingeniería de Sistemas, Universidad de Medellín, e-mail: jvergara@udem.edu.co

** Ingeniero de Sistemas, Asistente de investigación, Universidad Eafit, e-mail: aagudel6@eafit.edu.co

DESIGN AND IMPLEMENTATION OF A COMMUNICATION SYSTEM BASED ON CAN FOR AVIONICS IN A ROBOT MINI-HELICOPTER

Abstract

This paper presents the design of the internal communication system for avionics of a robot mini-helicopter based on the CAN distributed architecture. The communication system involves several hardware and software aspects related to data exchange on avionics bus from sensors and actuators with the flight computer. The main characteristic of the architecture is scalability in the addition of new devices, maintaining time requirements for data acquisition. Results of data exchange on the avionics network showing the reached operating update rates for each node are shown.

Key Words

Avionics, CAN, Control, UAV.

INTRODUCCIÓN

Este trabajo presenta el diseño de un modelo para el sistema de comunicaciones de la aviónica para un vehículo aéreo autónomo (UAV) no tripulado denominado COLIBRÍ (Vélez, 2005). El sistema de comunicaciones está basado en una arquitectura distribuida CAN (Controller Area Network (Etschberger, 2001)), enfocada en los conceptos de integración, escalabilidad (adición simple de nuevos dispositivos) y cumplimiento de estrictos requerimientos temporales. Con el sistema de comunicaciones se busca integrar un conjunto de dispositivos en una sola red, de tal forma que pueda establecerse comunicación de los mismos con el computador de vuelo. Los componentes del sistema son en su mayoría dispositivos COTS (commercial off-the-shelf). Éstos incluyen una unidad de medición inercial (Inertial Measurement Unit o IMU), un magnetómetro, un GPS, un altímetro barométrico y un sonar. Este sistema se ensambló en una caja de aviónica hecha en aluminio de 10x30x18cm la cual se adaptó a los trenes de aterrizaje de un mini-helicóptero.

Los aspectos de hardware presentados en este artículo involucran elementos como el diseño de la interfaz CAN para los nodos. Esta interfaz presta la función de tomar la información entregada por los sensores y transmitirla sobre la red de aviónica hacia el computador de vuelo. Otro elemento de hardware presentado es el diseño de la interfaz CAN a través del puerto paralelo (interfaz paralelo-CAN) para la lectura/escritura de información desde el computador de vuelo hacia los nodos de la red de aviónica.

Los aspectos de software presentados involucran la plataforma de simulación y monitoreo, y algunos detalles sobre el sistema de procesamiento de datos. El computador de vuelo hace uso del sistema operativo de tiempo real QNX, especializado para ejecución de tareas en tiempo real.

Al final de este trabajo, para validar la efectividad de esta arquitectura, se presentan resultados del intercambio de datos entre los nodos y el computador de vuelo, teniendo en cuenta las frecuencias en las que éstos operan. Se dan, además, las ventajas de esta arquitectura sobre arquitecturas comúnmente utilizadas para este tipo de aplicaciones (RS-232, RS-485). Se muestran lecturas de las mediciones entregadas por la caja de aviónica, tales como velocidades angulares y aceleraciones en los ejes (IMU) y el rumbo o *heading* (magnetómetro). Se presta especial atención a la IMU, pues ésta demanda mayores frecuencias de muestreo en el sistema de aviónica (Dittrich, 2002).

DISEÑO DEL SISTEMA DE COMUNICACIONES

Se describe el diseño del sistema de comunicaciones en tres partes. La primera parte da la descripción de los sensores seleccionados para la implementación de la aviónica, en la cual cada sensor es un nodo independiente en la red CAN. Los nodos tienen una interfaz CAN para la comunicación de datos con el computador de vuelo a través del bus de aviónica.

La segunda parte describe el diseño de la interfaz CAN para el computador de vuelo a través del puerto paralelo, la cual permite el intercambio de datos entre el mismo y cualquier nodo CAN conectado al bus de aviónica.

La tercera parte describe la integración y el modelo general de comunicaciones.

Sensores

Para la recolección de datos del entorno del vehículo se utilizan sensores, los cuales dependen de la variable específica que se quiere medir. En el caso de la IMU, ésta pertenece al grupo de sensores de navegación y actitud. El control de actitud es uno

de los más importantes, dadas las características dinámicas y aerodinámicas de un helicóptero (Johnson, 1980; Padfield, 1999; Mettler 2002). Para percibir la actitud de un helicóptero se usan varios tipos de sensores: acelerómetros (para medir aceleraciones en los tres ejes) y giróscopos (para medir velocidades angulares (Corbasí, 1998)), los que conforman la IMU. Mediante una técnica de estimación de variables de estado llamada filtro de Kalman (Rogers, 2000; Myungsoo 1999), se pueden encontrar las variables de estados adicionales para conocer la actitud del helicóptero. A este procedimiento de estimación también se suma

otro sensor llamado GPS (sistema basado en satélites que permite obtener la latitud, la longitud y la altura). Al sistema completo se le conoce como NIU (Navigation Inertial Unit) (Rogers, 2000) o INS (Inertial Navigation System). Con la adición del GPS (Kaplan, 1996) y el magnetómetro se puede navegar en tres dimensiones (Corbasí, 1998). La altitud se mide mediante un altímetro barométrico.

Todos los sensores y actuadores que están conectados a la red CAN tienen un hardware asociado encargado de la transmisión y recepción de datos sobre la red CAN (ver figura 1).

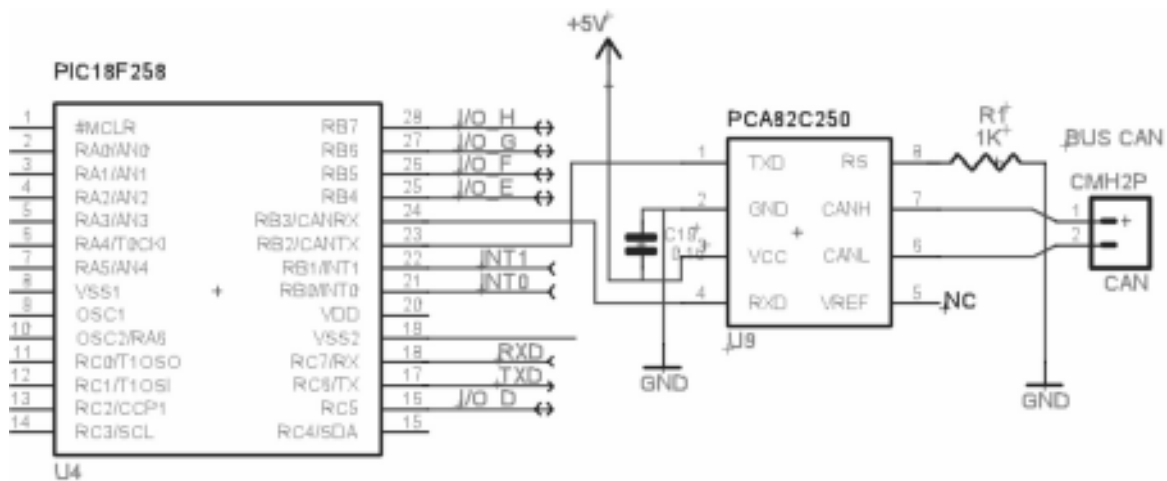


Figura 1: Interfaz CAN para nodos y actuadores.

El microcontrolador PIC18F258 es el encargado de recibir las señales del sensor de acuerdo con el tipo de señal o interfaz que utilice (análoga, digital o RS-232) y entregar los datos a través de la red CAN.

Diseño de la IMU

Para realizar la medición de las aceleraciones el PIC18F258 utiliza un temporizador. La IMU tiene la tarea de realizar las mediciones de velocidad angular en los ángulos de pitch, roll y yaw (cabeceo, alabeo y guiñada), además de las mediciones de aceleración en los ejes X, Y y Z. En

la IMU diseñada, el procesador que recibe las señales que provienen de los tres acelerómetros y de los tres giróscopos es también un microcontrolador PIC18F258; la razón de utilizar el PIC18F258 se debe a que éste posee un módulo CAN integrado.

La ubicación de los sensores para la medición en el eje respectivo es un factor clave, el diseño se estableció formando un cubo, en el cual cada uno de sus lados permite alojar ciertos componentes de la IMU. Los giróscopos tienen un rango de medición de 300 grados/seg y los acelerómetros de 10 g. (Ver figura 2)

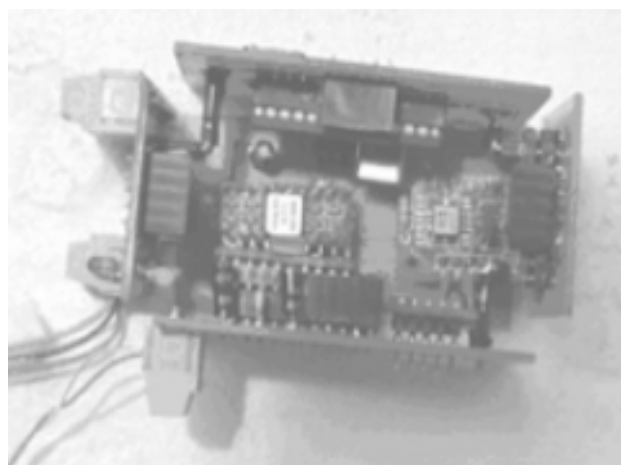


Figura 2: Diseño 3D final de la IMU

Sistema de posicionamiento global – GPS

Navegar es poder determinar en todo momento la posición, la velocidad y la actitud. Para el caso de un UAV, el GPS este es un elemento indispensable y la integración entre las mediciones entregadas por la IMU y el GPS permiten que la navegación sea realizable. El GPS utilizado tiene las siguientes características: 17 g de peso, 11 mm x 71 mm de área, recepción de un máximo de 17 satélites, 2 puertos seriales para comunicación y corrección diferencial para disminución del error (ver figura 3).

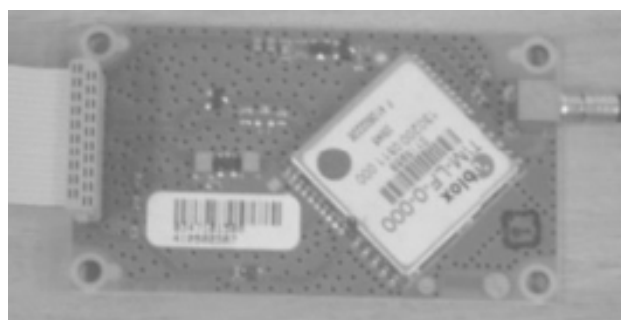


Figura 3: GPS Ublox.

Medición de alturas

En un UAV es necesaria la medición de grandes y bajas altitudes. La medición de baja altitud es útil

para los estados de despegue y aterrizaje; en este caso se utiliza un sonar altímetro (figura 4.a). Las mediciones de gran altitud (sobre el nivel del mar) son útiles en el vuelo por encima de los dos metros; para ello se utiliza un altímetro barométrico (figura 4.b)

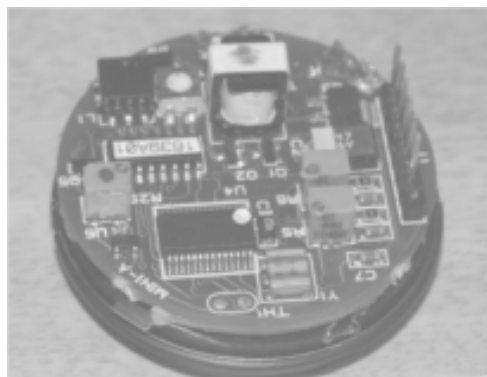


Figura 4: a. Sonar altímetro



Figura 4: b. Altímetro barométrico

Magnetómetro

El magnetómetro o brújula digital (ver figura 5) es un elemento indispensable y necesario para la orientación del UAV. Con este sensor es posible conocer los grados de orientación (yaw), que tiene el vehículo con respecto al norte magnético, además de mediciones auxiliares que este dispositivo entrega como el pitch y roll.

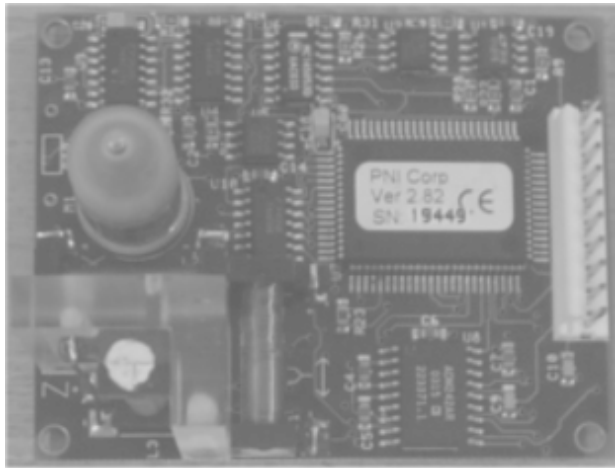


Figura 5: Magnetómetro

Diseño de la interfaz can para el computador de vuelo

El sensor COTS de velocidad angular escogido es el ADXRS300 de la compañía. Para la recolección de la información proveniente desde los sensores y para el envío de información hacia los actuadores por parte del computador de vuelo, es necesaria una interfaz CAN que permita la conexión al bus de aviónica donde se encuentran conectados todos los dispositivos (sensores y/o actuadores). Existen opciones comerciales de interfaces CAN para bus PCI, PCMCIA y arquitectura PC-104, las cuales tienen una restricción bastante fuerte debido al consumo de corriente, elemento que afecta de forma sustancial el tiempo de autonomía de vuelo para el helicóptero, el cual se alimenta por medio de baterías. Se decidió hacer la interfaz a través del puerto paralelo en el modo ECP+EPP que sigue el estándar IEEE 1284 implementado en 1994, el cual permite obtener un ancho de banda del orden de 500KBps hasta 2MBps, garantizando que no se generen cuellos de botella a partir del ancho de banda requerido para el bus CAN que es de 1 Mbps (125KBps).

El diseño de la interfaz se basa en un microcontrolador PIC18F258 que se encarga de recibir datos enviados por el computador de vuelo a través del puerto paralelo y enviarlos sobre la

red CAN para el caso de transmisiones (hacia actuadores) desde el computador de vuelo; el otro caso es en el que el PIC18F258 recibe datos (desde sensores) a través de la red CAN y los transfiere al computador de vuelo por el puerto paralelo para el caso de lecturas.

El consumo de corriente máxima en la interfaz CAN del computador de vuelo diseñada es de 100 mA, lo cual va a favor del tiempo de autonomía de vuelo y en comparación a una interfaz comercial donde el consumo promedio es de 700 mA. La figura 6 ilustra el circuito impreso de la interfaz CAN para el computador de vuelo.

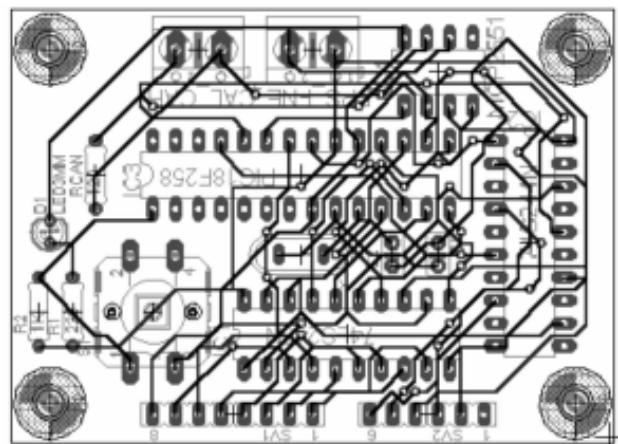


Figura 6: Diseño impreso de la interfaz CAN para el computador de vuelo.

Modelo del sistema de comunicaciones

El modelo del sistema de comunicaciones se basa en la conexión del computador de vuelo al bus de aviónica por medio de la interfaz CAN a través del puerto paralelo, con lo cual es posible intercambiar datos con cualquier nodo CAN conectado al bus.

En el modelo del sistema de comunicaciones se utilizaron identificadores de 11 bits para todos los nodos de la red CAN.

CAN es un protocolo de comunicaciones basado en una arquitectura de bus para la transferencia

de mensajes en ambientes distribuidos. Entre sus fortalezas está el permitir una arquitectura multi-maestro capaz de proveer características de respuesta en tiempo real, tolerancia a fallas en la recepción de mensajes y el mal funcionamiento de los nodos. Originalmente el CAN fue concebido para aplicaciones en el área automotriz, pero rápidamente emigró hacia áreas como el control y la automatización industrial, donde los buses de campo (*field bus*) proveen recursos de conectividad e integración de dispositivos de igual o distinta tecnología.

CAN está estructurado de acuerdo con el modelo OSI en una arquitectura de dos niveles (capa física y capa de enlace de datos). Distintas opciones están disponibles para la capa de aplicación como: CiA CAN Application Layer, CANOpen, SDS (Smart Distributed System), DeviceNet y CAN Kingdom [3]. La capa física se basa en dos líneas: CANL y CANH, por las cuales viajan los datos bajo un esquema de líneas balanceadas o diferenciales. La capa de enlace de datos (DLL) está estandarizada por ISO 11898. Los servicios de la DLL

en un nodo CAN son implementados por las subcapas de control de enlace lógico (LLC) y de control de acceso al medio (MAC), respectivamente. La subcapa LLC provee las funciones de filtro de aceptación (son aceptados solo los mensajes cuyos identificadores han sido previamente programados), notificación de sobrecarga y manejo del proceso de recuperación de errores. La subcapa MAC es responsable del empaquetamiento/desempaquetamiento de los datos, codificación de las tramas, manejo de acceso al medio, detección de error, señalización de error y serialización/deserialización de datos (Etschberger, 2001). Un factor importante con el uso de CAN es el hecho de poder manipular los errores sobre el medio y así tener mayor certeza sobre la información que se intercambia entre los nodos y el computador de vuelo.

En la figura 7 se aprecia la interconexión de todos los dispositivos (sensores/actuadores) que conforman la aviónica del UAV a través de la red CAN, además se ilustra la conexión inalámbrica 802.11b hacia la estación en tierra y líneas de alimentación de voltaje.

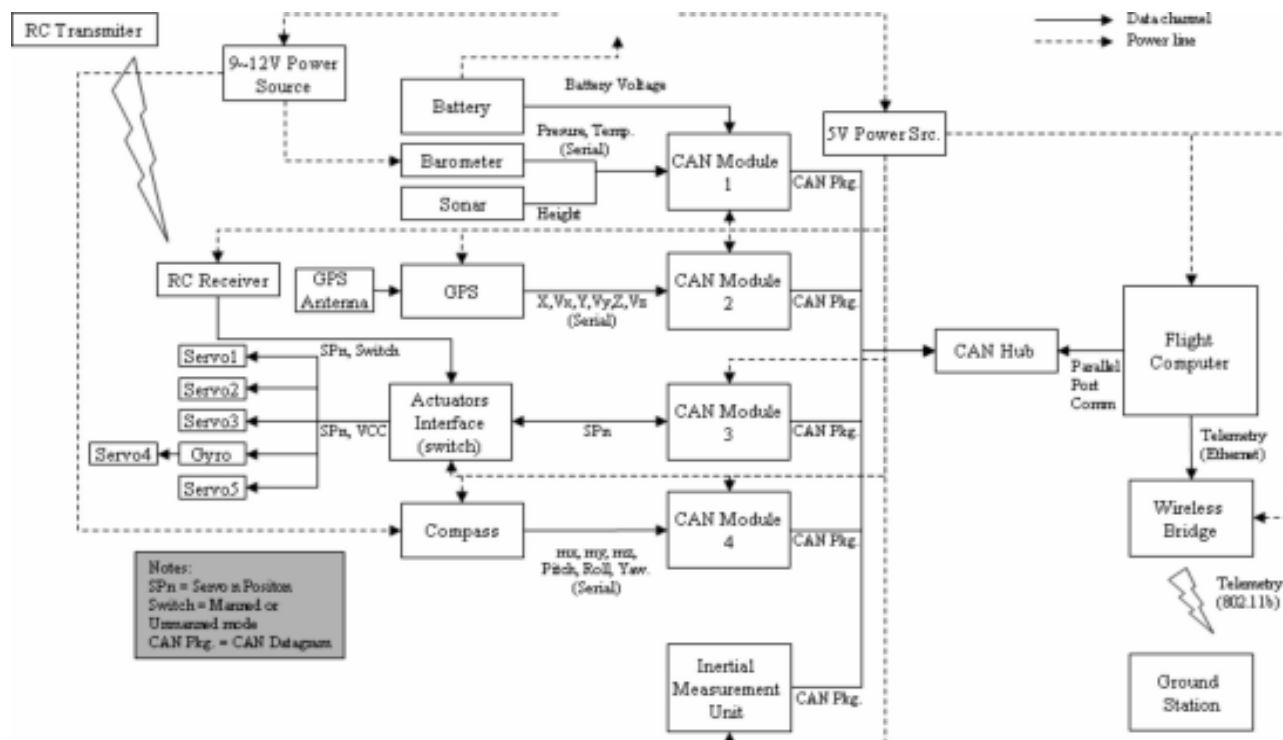


Figura 7: Modelo general del sistema de comunicaciones.

PLATAFORMA DE SIMULACIÓN Y MONITOREO

Las redes CAN se utilizan en áreas del control y automatización industrial como redes. Se requirió de una plataforma apropiada para la simulación y el monitoreo de las señales provenientes de los sensores. La plataforma debía ser fácilmente reconfigurable para diversas pruebas de hardware y de software. Se seleccionó Matlab/Simulink pues permite integrar fácilmente el control con la lectura de sensores y los actuadores. Simulink agiliza el desarrollo de prototipos y su simulación en tiempo real es una de las características más atractivas. La integración de la plataforma con el *target* (equipo de cómputo que ejecutará la lectura de sensores, control y actuación) se logra con Real-time workshop de Simulink. Se escogió QNX 6.3 como el sistema operativo para la ejecución debido a sus características de tiempo real. Un producto comercial (RT-LAB) y una plataforma propia han

sido usados para la generación automática de código y la transferencia al *target*.

Para la comunicación con el *target* se usaron tecnologías estándares como Ethernet, 802.11b y los protocolos UDP/IP y TCP/IP para transferencia de código y monitoreo. El helicóptero, por ejemplo, está equipado con un puente inalámbrico 802.11b. Una WLAN *ad-hoc* se forma con la computadora de vuelo del helicóptero y uno o más computadores portátiles que desempeñan el papel de estación de tierra o estaciones de supervisión. En pruebas de laboratorio se puede usar una red LAN.

Una sesión normal con la plataforma de simulación incluye: ajustar un modelo de Simulink para las pruebas deseadas, generar código, transferirlo automáticamente al *target*, ejecutar, conectar y monitorear. Esto último puede hacerse con gráficas de variables vs. tiempo, un panel de vuelo o un ambiente de realidad virtual (ver figura 8).

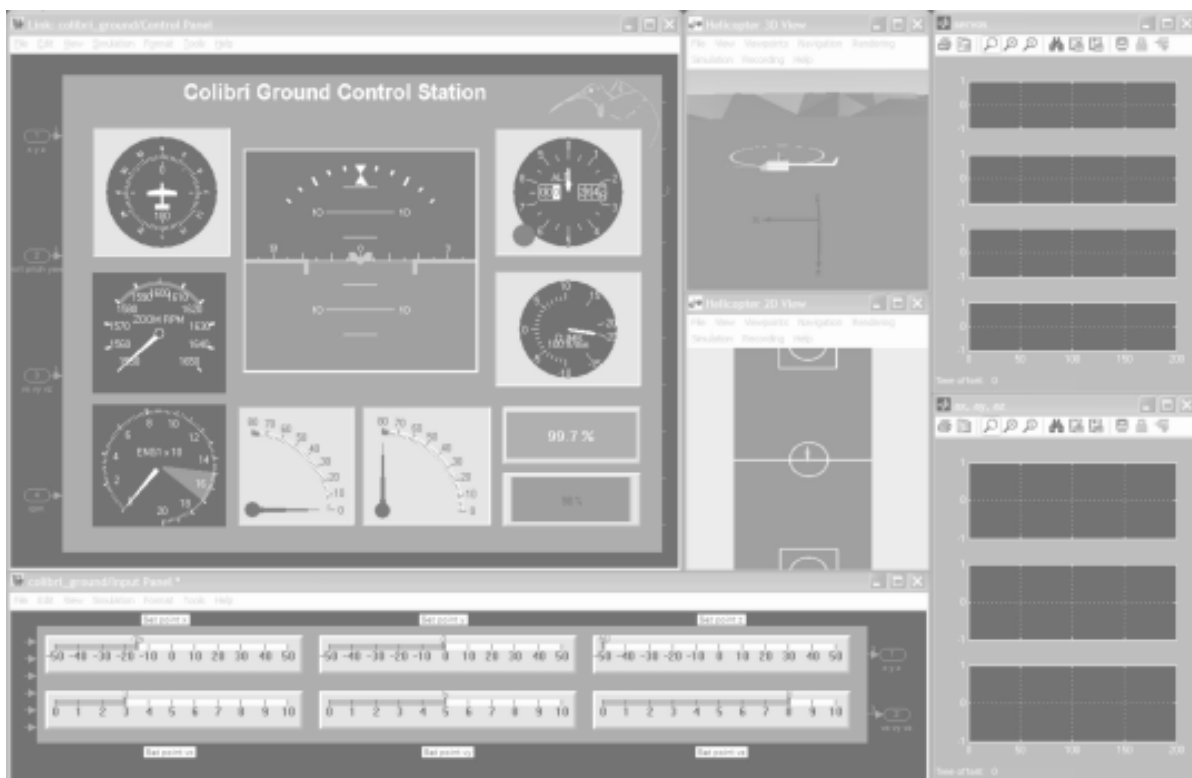


Figura 8: Estación en tierra

La telemetría durante el vuelo es de gran importancia debido a que las pruebas de vuelo requieren una vista actualizada del estado del helicóptero. Un *blockset* de Simulink fue diseñado para satisfacer esta necesidad. El módulo toma parámetros del modelo en ejecución y los envía por la red en paquetes UDP. Se eligió UDP debido a que los paquetes de los datos son enviados sin importar la pérdida de los mismos. Esta característica es deseable para comunicaciones en tiempo real.

ARQUITECTURA RS-232 VS CAN

El esquema que presentan la gran mayoría de los sistemas de aviónica actuales para UAV's es un sistema de comunicaciones centralizado basado en RS-232 (Kahn, 2001), lo que genera grandes problemas de escalabilidad en la agregación de nuevos sensores, como: sensor de presión atmosférica, sensor de temperatura para el tubo de escape del vehículo, sensor de temperatura para el motor del vehículo, sensor para el nivel del combustible, cámara, etc. Estos podrían ser más, dependiendo de los propósitos que se definan para el uso del vehículo. Existe una gran limitante para el computador de vuelo en cuanto al número de interfaces RS-232 disponibles, lo cual en los sistemas de hardware computacional actuales es bastante limitado. Para una arquitectura PC-104 comúnmente usada en UAV podrían ser tres o cuatro interfaces, y la posibilidad de agregar más interfaces tendría graves repercusiones tanto en el nivel de procesamiento como en el nivel de consumo de potencia, factor clave para el tiempo de vuelo en un UAV. En nuestro caso, se buscó diseñar un bus de aviónica utilizando la arquitectura distribuida CAN que pudiera integrar todos los elementos (sensores y actuadores) del vehículo, garantizando la posibilidad de agregar nuevos componentes de una manera fácil y sin afectar los requerimientos de hardware en el nivel de interfaces disponibles.

Otro gran problema tiene que ver con las restricciones de tiempo inherentes que posee el medio o una interfaz RS-232. En los mejores casos, la velocidad de transmisión es 57.600bps o 115.200bps, lo que conlleva a altos niveles de procesamiento en el computador de vuelo para obtener los resultados esperados en términos de las acciones de control a tomar. Esto no da lugar al mejoramiento de las frecuencias de muestreo que son necesarias por algunos algoritmos de control que se implementan sobre el computador de vuelo y que pueden proporcionar un mejor funcionamiento en la navegación para el UAV. El modelo RS-485 es una arquitectura distribuida y permite obtener grandes distancias entre los puntos de conexión o nodos, pero ésta utiliza el mismo ancho de banda del RS-232 y no posee las ventajas que provee CAN. Con una red CAN, el ancho de banda que se puede obtener es de 1Mbps, y con ello se abre la alternativa de poder agregar carga de datos sobre el medio, teniendo la posibilidad de garantizar los requerimientos temporales del vehículo de acuerdo con el sensor o actuador utilizado.

VALIDACIÓN DE LOS REQUERIMIENTOS TEMPORALES

El sistema de software que se ejecuta en el computador a bordo se encarga de ejercer el control sobre el helicóptero y enviar/recibir información de la estación en tierra (computador portátil). Se establecen, entonces, cinco tareas básicas: control y filtrado, lectura de sensores, actuación, envío de datos a la estación en tierra y recepción de comandos de la estación de tierra. La ejecución de estas tareas se hace a través de un hilo independiente por cada una. El uso de hilos garantiza una mejor distribución de la capacidad de procesamiento y la asignación de los tiempos necesarios para la entrada y salida de datos. El sistema operativo se encarga de repartir los tiempos por tarea. El control y el filtrado se ejecutan entre 50 y 100 Hz. La lectura de todos los sensores se hace a 250 Hz, aproximadamente.

La lectura de datos de los diferentes sensores (nodos CAN) se realiza de la siguiente manera: cada nodo sensor envía continuamente la información al nodo conectado al computador a bordo; una vez allí, la información es transmitida al computador y ordenada para el procesamiento; cada sensor envía a frecuencias diferentes; el hilo de lectura identifica de qué sensor proviene la información y la entrega al módulo de filtrado/control.

Tabla 1: Frecuencias de lecturas obtenidas.

Sensor	Frecuencia
IMU Acelerómetros	94.73 Hz
IMU Giróscopos	95.12 Hz
Magnetómetro	21.66 Hz
Heading	20.22 Hz
GPS X/VX	6.52 Hz
GPS Y/VY	6.26 Hz
GPS Z/VZ	6.13 Hz

Lectura típica de los sensores en mensajes por segundo. Cada mensaje contiene dos o tres variables de acuerdo con el sensor. La frecuencia total de lectura para este caso es 260.93 Hz.

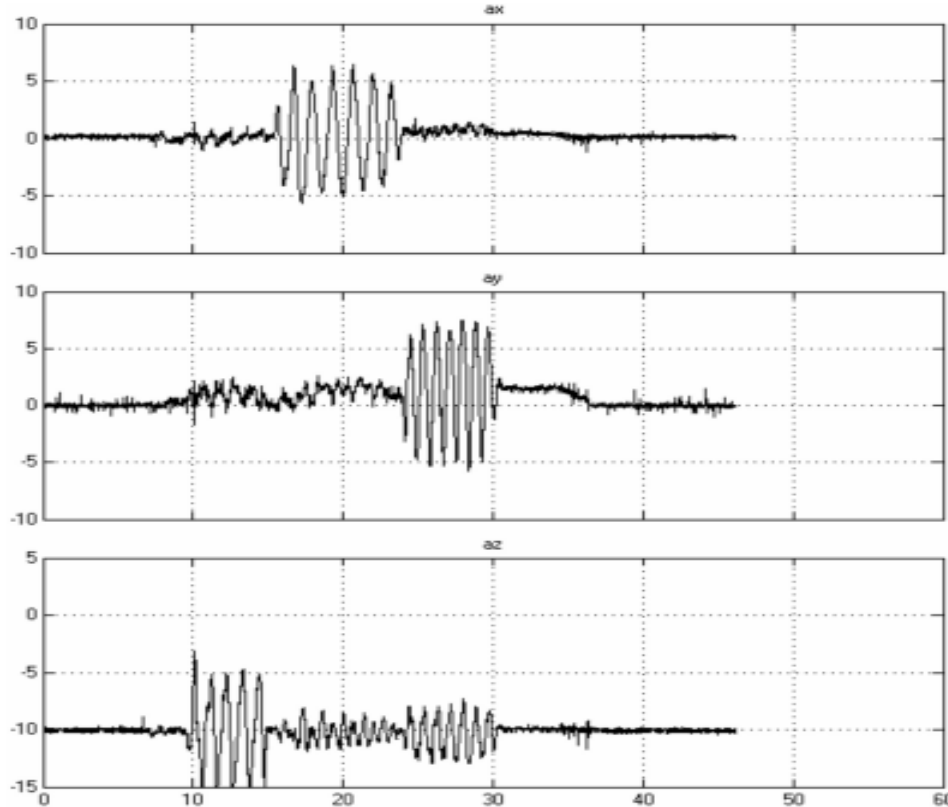


Figura 9: Lectura de acelerómetros

En la tabla 1 se pueden apreciar las frecuencias obtenidas para los diferentes sensores del UAV. Estas son comparables con las frecuencias comúnmente utilizadas (Kahn, 2001) descritas en la tabla 2.

Tabla 2: Frecuencias de lecturas típicas.

Sensor	Frecuencia
IMU Acelerómetros	100 Hz
IMU Giróscopos	100 Hz
Magnetómetro	20 Hz
Heading	20 Hz
GPS X/VX	5 Hz
GPS Y/VY	5 Hz
GPS Z/VZ	5 Hz

La frecuencia total de lectura para este caso es 255 Hz.

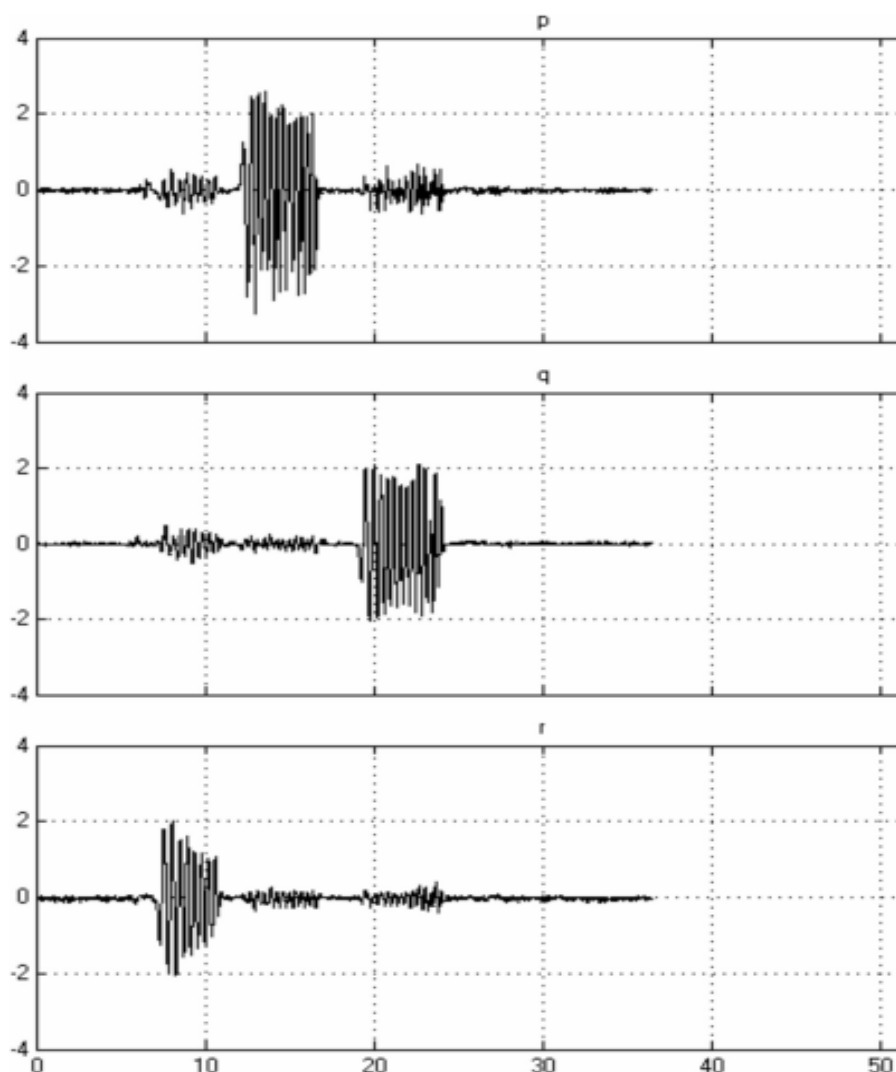


Figura 10: Lectura de giróscopos

En las figuras 9 a 11 se muestran diferentes lecturas de señales de sensores (IMU y magnetómetro) tomadas por el computador.

La figura 9 muestra el registro de los datos para las aceleraciones en los ejes X, Y y Z; para éstas se agitó la caja en el eje Z, X, y luego Y, respectivamente. La figura 10 muestra las velocidades angulares p , q y r . La figura 11 muestra el cabeceo, alabeo y guiñada. Todas las mediciones fueron almacenadas con las frecuencias mostradas en la tabla 1. Los datos fueron registrados con la conexión del sistema completo (caja de aviónica/computador de vuelo/estación en tierra).

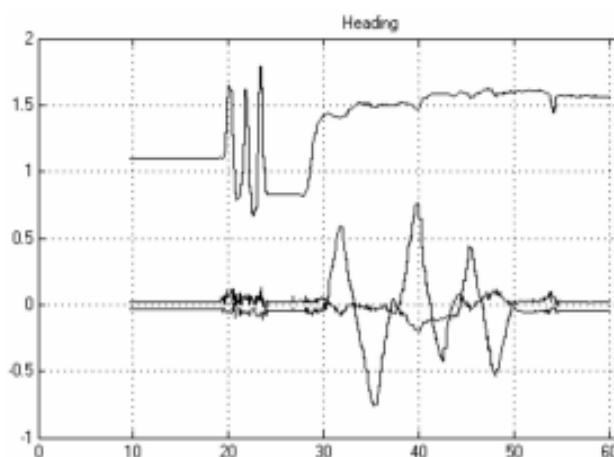


Figura 11: Lectura de magnetómetro.

Los estímulos se aplicaron de tal manera que se afectó cada variable de forma independiente para una mejor visualización.

CONCLUSIONES

La idea de desarrollar un sistema de comunicaciones basado en la arquitectura distribuida CAN para un UAV fue terminada completamente, y la escogencia de la arquitectura distribuida CAN fue un acierto, ya que sus características permiten llegar a aplicaciones como la de un bus de aviónica. La adición de dispositivos sensores y/o actuadores para el UAV es una realidad tangible.

La adición de nuevos nodos sensores y/o actuadores al bus de aviónica está contemplada en el diseño hardware/software, y así se posibilita el intercambio de información entre el computador de vuelo y un nodo CAN cualquiera.

Para cada nodo se calculó el tiempo de lectura de datos y se obtuvieron las frecuencias de operación. Con ello se realizó la validación de los requerimientos temporales para el intercambio de datos con el computador de vuelo.

La figura 12 muestra el diseño definitivo de la caja de aviónica, ensamblada conjuntamente con el mini-helicóptero.

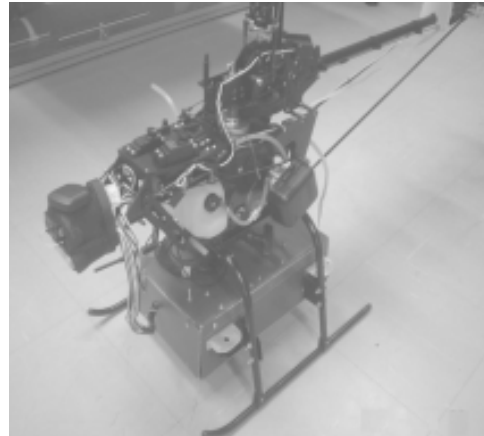


Figura 12: Mini-helicóptero con la caja de aviónica.

BIBLIOGRAFÍA

- CORBASÍ, A. Sistema de navegación. McGraw Hill, 1998.
- DITTRICH, Joerg S. Design and integration of an unmanned aerial vehicle navigation system. Thesis the Master of Science in Aerospace Engineering, 2002.
- ETSCHBERGER, Konrad. Controller area network- basics, protocols, chips and applications. IXXAT Automation GmbH, 2001.
- HELFRICK, A. Principles of avionics. Avionics Communications, 2002.
- JOHNSON, W. Helicopter theory. Dover, 1980.
- KAHN D., Aaron. The desing and development of a modular avionics system. Thesis the Master of Science in Aerospace Engineering, 2001.
- KAPLAN, E. Understanding GPS principles and applications. Artech House, 1996.
- METTLER, B.; TISCHLER, M. B.; KANADE, T. System identification modeling of a small-scale unmanned rotorcraft for flight control design. American Helicopter Society Journal, Vol. 47, No. 1, pp. 50-63, 2002.
- MYUNGSOO, J. State estimation via sensor modeling for helicopter control using and indirect kalman filter. Proceedings IEEE International Conference, 1999.

PADFIELD, G. D. Helicopter flight dynamics: the theory and application of flying qualities and simulation modelling. AIAA Education Series, pp. 514, 1999.

ROGERS, R. Applied mathematics in integrated navigation systems. Education Series AIAA, 2000.

VÉLEZ, C. M.; AGUDELO, A. Multirate control of an unmanned aerial vehicle. WSEAS Transactions on Circuits and Systems, Vol. 4, No.11, ISSN 11092734, pp. 1628-1634, 2005.