



**UNIVERSIDAD DE CONCEPCIÓN**  
**FACULTAD DE INGENIERÍA**  
**DEPARTAMENTO DE INGENIERÍA MECÁNICA**



**ESTABLECIMIENTO DE PROTOCOLO DE COMUNICACIÓN ENTRE X-PLANE  
Y MICROPROCESADORES EXTERNOS EN EL SIMULADOR DE VUELO DEL  
LABORATORIO DE TÉCNICAS AEROESPACIALES**

**POR**

**Germán Adolfo Quijada Arriagada**

Profesor guía:

Bernardo Andrés Hernández Vicente

16 de mayo de 2023

# Índice general

<b>1. Definición del problema</b>	<b>2</b>
1.1. Planteamiento del problema . . . . .	2
1.1.1. Contexto y justificación . . . . .	2
1.1.2. Software utilizado . . . . .	3
1.1.3. Otras soluciones . . . . .	4
1.2. Objetivos . . . . .	4
1.2.1. Objetivo general . . . . .	4
1.2.2. Objetivos específicos . . . . .	4
1.3. Condiciones de diseño . . . . .	5
1.3.1. Simulador . . . . .	5
1.3.2. Protocolo de comunicación . . . . .	5
1.3.3. Documentación . . . . .	6
1.4. Metodología de trabajo . . . . .	6
1.4.1. Simulador . . . . .	6
1.4.2. Protocolo de comunicación . . . . .	6
1.4.3. Documentación . . . . .	7
<b>2. Simulador</b>	<b>8</b>
<b>3. Protocolo de comunicación</b>	<b>9</b>

# Capítulo 1

## Definición del problema

### 1.1. Planteamiento del problema

#### 1.1.1. Contexto y justificación

Los programas de ingeniería aeroespacial además de impartir conocimientos de las ciencias básicas, abren el camino a la especialización ingenieril en las áreas aeronáutica y espacial, siendo uno de los tantos temas abordados la introducción a los procedimientos para ejecutar un vuelo y los fenómenos físicos involucrados que lo hacen posible, lo que abre las puertas a posterior investigación relacionada donde contar con el equipo adecuado que permita realizar simulaciones de vuelo que sustenten estos estudios puede ser muy útil [1].

Las simulaciones permiten acceder o modificar arbitrariamente variables de la aeronave y el entorno, con la ayuda de interfaces que extraen de la simulación el estado de los sistemas involucrados y lo exportan a otros medios en los que sea más fácil hacer uso de los datos, ya sea para visualizarlos en tiempo real en un panel de instrumentos con indicadores, registrar información de la condición de la aeronave durante el vuelo para posterior evaluación de rendimiento, o la modificación directa de la simulación para estudiar condiciones específicas, entre otros. Todo sin dejar de lado el realismo y por lo tanto autenticidad que ofrece el software de simulación evitando lo costoso de realizar los mismos estudios en aeronaves de verdad.

Sin embargo, existen barreras tecnológicas que ralentizan el uso de un simulador de vuelo por lo compleja que puede ser su interfaz gráfica o lo exigentes que sean en términos del manejo de una aeronave, el usuario debe en primera instancia contar con el hardware adecuado

que pueda ejecutar el simulador y familiarizarse con el funcionamiento interno del software para lograr sus fines. El objetivo de este proyecto de ingeniería abarca principalmente el caso de lograr comunicar el estado de la simulación a un hardware externo y que este pueda enviar comandos en respuesta, estableciendo una interfaz dedicada para ello que sea versátil y fácil de empezar a utilizar, junto con las instrucciones pertinentes.

En la simulación el uso de hardware externo que no sea la computadora principal es algo común por distintos motivos, como el querer emular la experiencia de un avión real lo que se traduce en la creación de controles e indicadores personalizados parecidos a los presentes en un panel de instrumentos, usualmente conectados a un microcontrolador que debe interactuar con el software por medio de alguna interfaz. Además, y sobre todo en un contexto de investigación donde se deban realizar múltiples pruebas iterativas, el que se trabaje con equipos externos evita que se utilicen recursos que pudiese requerir la simulación y de manera más importante también se evita el que se hagan modificaciones al software o hardware del equipo principal en la instalación de los programas para la investigación y luego en la desinstalación una vez los análisis concluyan, así el simulador se mantiene íntegro asegurando su disponibilidad para ser utilizado en un futuro con otro propósito.

### **1.1.2. Software utilizado**

Algunas soluciones existentes para comunicar simuladores de vuelo con microprocesadores son SimVimX [10] y MobiFlight [7], estos programas están diseñados para entusiastas que construyen paneles de instrumentos personalizados conectados a microprocesadores como Arduino o a otros computadores, habilitan la integración de los paneles proporcionando manuales e interfaces gráficas que ayudan a programar todo.

Debido al enfoque que tienen SimVimX y MobiFlight, las funciones que puede cumplir el microcontrolador están limitadas a hacer de interfaz para paneles de instrumentos, y no permiten mayores prestaciones sin modificar su código para su uso en otros fines.

Flight Simulator Universal Inter-Process Communication [1] o FSUIPC es un software que accede a la memoria interna de un simulador como Microsoft Flight Simulator [5] o Prepar3D [2], que permite a otros programas leer y modificar variables internas de la aeronave o el entorno, muchos plug-ins existentes dependen de esta interfaz para funcionar como por ejemplo MobiFlight. Además, el protocolo que establece FSUIPC puede ser considerado como un estándar, ya que existe otro software que lo reimplementa como XPUIPC [11] o XConnect [12] que habilita a los programas diseñados para funcionar con FSUIPC a que también lo

hagan con X-Plane [9].

Programar software que aproveche FSUIPC otorga las ventajas de compatibilidad con cualquier simulador soportado por la librería o sus reimplementaciones como XPUIPC, y el poder concentrar el desarrollo del establecimiento del protocolo a la comunicación externa con microcontroladores, puesto que la manipulación interna de memoria para cada simulador ya esta cubierta.

### **1.1.3. Otras soluciones**

X-Plane Connect [8] o XPC es otra interfaz diseñada por la NASA que cumple funciones similares a FSUIPC, pero específica para X-Plane sin posibilidad de adaptarla a otros simuladores. Lo interesante de esta alternativa es que incluye una librería para programar en MATLAB [3] y por consiguiente Simulink [4] lo que facilita el análisis de sistemas de control. SimConnect [6] desarrollado por Microsoft es una interfaz de bajo nivel como XPC o FSUIPC diseñada exclusivamente para Microsoft Flight Simulator 2020 también con características similares a las demás librerías.

## **1.2. Objetivos**

### **1.2.1. Objetivo general**

Reconfigurar simulador de vuelo del laboratorio de técnicas aeroespaciales, luego establecer un protocolo de comunicación entre el software del simulador y hardware externo, finalmente documentar la utilización del protocolo de comunicación y del simulador.

### **1.2.2. Objetivos específicos**

1. Reconfigurar la disposición del hardware del simulador con el objetivo de facilitar su uso y agilizar en el futuro la instalación de nuevo software. Se registrará y comparará el hardware actual del simulador con el hardware nuevo disponible en el laboratorio.

2. Puesta en marcha del software de simulación. Instalación de sistema operativo Windows 10 actualizado y X-Plane, configurar el panel de instrumentos y controles.

3. Establecer protocolo de comunicación entre software del simulador y equipos externos, primero definiendo un protocolo de comunicación sobre IP y Serial diseñado para micropro-

cesadores que soporten estas funciones, luego preparar con librerías existentes la extracción y manipulación de variables internas del simulador para finalmente combinar el protocolo de comunicación y la implementación de la librería en un programa personalizado.

4. Documentar uso del simulador preparando manuales instructivos referentes a la configuración de hardware y software del simulador, y el trabajo con el protocolo de comunicación con dispositivos externos.

## **1.3. Condiciones de diseño**

### **1.3.1. Simulador**

Actualmente el simulador en el laboratorio de técnicas aeroespaciales está limitado a X-Plane 9 debido a su disposición de 3 computadores de bajas prestaciones corriendo en paralelo, y cualquier mejora que se quiera hacer sobre el sistema debe cumplir con las condiciones que esa configuración implica, por esto se busca un diseño nuevo que permita más fácilmente trabajar en el simulador.

La nueva disposición de hardware consistirá en reducir la cantidad de computadores a solo uno, esto corta significativamente el tiempo que toma cualquier proceso que tuviese que realizarse en los tres computadores, elimina la necesidad de un router y cables que los conecten, y reduce la latencia entre las imágenes que muestran distintos monitores entre otras cosas.

Respecto al software, se va a realizar una instalación de Windows 10 compatible con gran parte si no todos los simuladores de vuelo en el mercado. Se instalarán X-Plane 9 y 10 disponibles por la universidad los cuales deberán configurarse de modo que trabajen con los múltiples monitores.

### **1.3.2. Protocolo de comunicación**

El nuevo código que se vaya a escribir para el protocolo de comunicación utilizara estándares ya establecidos en simuladores, y la implementación en el lado del hardware externo será en un lenguaje que compartan los microprocesadores o microcontroladores como Arduino y Raspberry con mínima modificación.

Reducir la complejidad del protocolo de comunicación es clave, pues se está buscando que esta interfaz pueda ser utilizada por más personas que no necesariamente estén familiarizadas

con la programación puedan cumplir sus objetivos, los cuales van más allá que solo crear paneles de instrumentos, como por ejemplo control y pruebas automatizadas.

### **1.3.3. Documentación**

Respecto a la documentación esta será en formato tipo manual, abarcará la nueva configuración del simulador y el trabajo con el protocolo de comunicación, desde el encendido de los equipos hasta el uso del protocolo con ejemplos para comenzar. Incluirá la solución a problemas comunes que se vayan notando los que no necesariamente deban ser técnicos.

## **1.4. Metodología de trabajo**

### **1.4.1. Simulador**

Para reconfigurar el simulador de acuerdo con las necesidades y las condiciones de diseño establecidas en primera instancia se realiza un levantamiento de antecedentes “en terreno” de la configuración actual a una profundidad suficiente para volver a implementarla en caso de que se quiera volver ella. El nuevo hardware disponible en el laboratorio también se analizará con su documentación para comparar sus capacidades con el hardware actual.

Hecho esto, se investigarán las alternativas y posibilidades de habilitar la máxima cantidad de monitores por computador considerando el hardware disponible. Finalmente se ejecutará el nuevo diseño del simulador y se instalara y configurara el software.

### **1.4.2. Protocolo de comunicación**

El comienzo del desarrollo del protocolo de comunicación consistirá en establecer el canal sistema operativo - microprocesador y la librería para trabajar con el simulador.

#### **1. Seleccionar canal de comunicación entre el sistema operativo y hardware externo**

Para lograr un enlace entre un microprocesador y el software de simulación primero es necesario establecer comunicación entre el microprocesador y el sistema operativo, esto requiere caracterizar los canales disponibles en el hardware externo que se vaya a utilizar, los microcontroladores por lo general cuentan con una interfaz serial e inalámbrica por Bluetooth o wifi, también habrá que averiguar los lenguajes en los que programarlos comparando sus características de acuerdo a los objetivos.

## **2. Seleccionar software que permita la extracción y manipulación de variables del simulador**

Primero se deben comparar las librerías que permiten trabajar sobre las variables del simulador según su lenguaje de programación, software de simulación que soportan y si sigue siendo activamente desarrollada.

Una vez tomadas las decisiones se procede a escribir el programa que se ejecutara en el sistema operativo, que actúe de interfaz entre el software de simulador y el microcontrolador, comunicados por medio de un protocolo personalizado que trabaje por sobre el de la librería y transmitido por el canal seleccionado.

### **1.4.3. Documentación**

Con el simulador reconfigurado, se invitará a miembros de la carrera interesados a probar y trabajar con el equipo para obtener realimentación sobre su uso y verificar el cumplimiento de los objetivos que se buscaban, con posibilidad de realizar modificaciones menores en respuesta a lo que se vaya reportando, además de registrar problemas comunes que deban ser mencionados en la documentación del uso del simulador.

Finalmente, con base en la experiencia y los registros tomados durante el desarrollo se crearán manuales que describan el uso y procedimientos para realizar trabajos de interés con el protocolo de comunicación, que estarán disponibles de manera física y online en un repositorio dedicado al simulador.



## **Capítulo 2**

### **Simulador**

## **Capítulo 3**

### **Protocolo de comunicación**

# Bibliografía

- [1] Peter Dowson y John Dowson. *Flight Simulator Universal Inter-Process Communication (FSUIPC)*. URL: <http://www.fsuipc.com/> (visitado 15-05-2023).
- [2] Lockheed Martin. *Prepar3D*. URL: <https://www.prepar3d.com/> (visitado 15-05-2023).
- [3] MathWorks. *MATLAB*. URL: <https://mathworks.com/> (visitado 15-05-2023).
- [4] MathWorks. *Simulink*. URL: <https://mathworks.com/products/simulink.html> (visitado 15-05-2023).
- [5] Microsoft. *Microsoft Flight Simulator*. URL: <https://www.flightsimulator.com/> (visitado 15-05-2023).
- [6] Microsoft. *SimConnect*. URL: [https://docs.flightsimulator.com/html/Programming\\_Tools/SimConnect/SimConnect\\_SDK.htm](https://docs.flightsimulator.com/html/Programming_Tools/SimConnect/SimConnect_SDK.htm) (visitado 15-05-2023).
- [7] *MobiFlight*. URL: <https://www.mobiflight.com/> (visitado 15-05-2023).
- [8] NASA. *X-Plane Connect*. URL: <https://github.com/nasa/XPlaneConnect> (visitado 15-05-2023).
- [9] Laminar Research. *X-Plane*. URL: <https://www.x-plane.com/> (visitado 15-05-2023).
- [10] Roman Sychev. *SimVimX*. URL: <https://simvim.com/> (visitado 15-05-2023).
- [11] Torsten Spiering. *X-Plane Universal Inter-Process Communication (XPUIPC)*. URL: <https://www.schiratti.com/xpuipc.html> (visitado 15-05-2023).
- [12] Mat Sutcliffe. *XConnect*. URL: <https://xplaneconnect.sourceforge.net/> (visitado 15-05-2023).