VOLTES FLY

Senior Design Fall 2011
University of Central Florida
College of Electrical Engineering &
Computer Science

Members:
Manuel Arredondo
Joe Paolini
Hector Bermudez

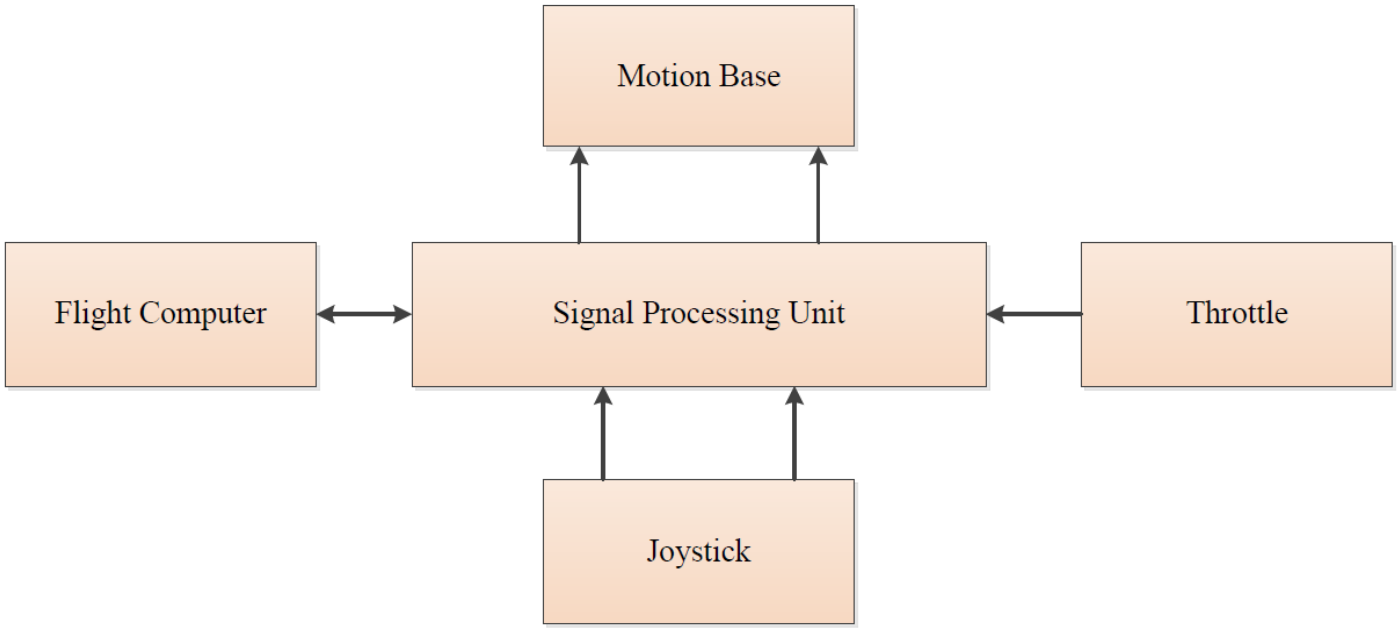
Motivation

- Homemade Flight simulator videos found online
- Relevant to industry in the Orlando Area
- Project paid off with it’s entertainment value!

Introduction

- Vestibular cueing flight simulator for a Cessna 172
- Throttle
- Joystick with haptic feedback
- Software Application

System Overview



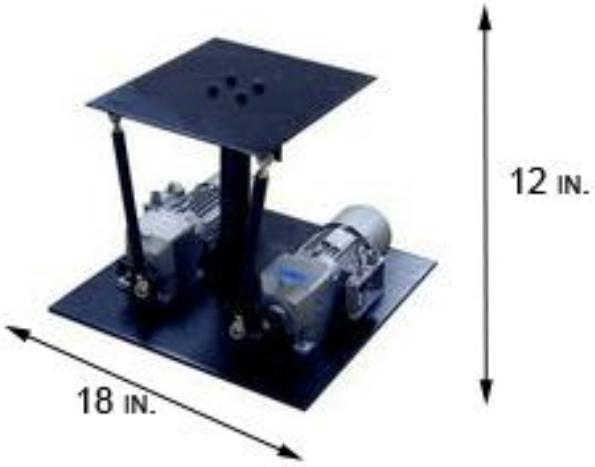
Voltes Fly Cockpit

Specifications

- Supports max weight and height of 210lbs and 6’5”
- Image Generator (X-Plane) runs at a minimum of 30 fps
- Instructor/Operator Station runs at a minimum of 30HZ

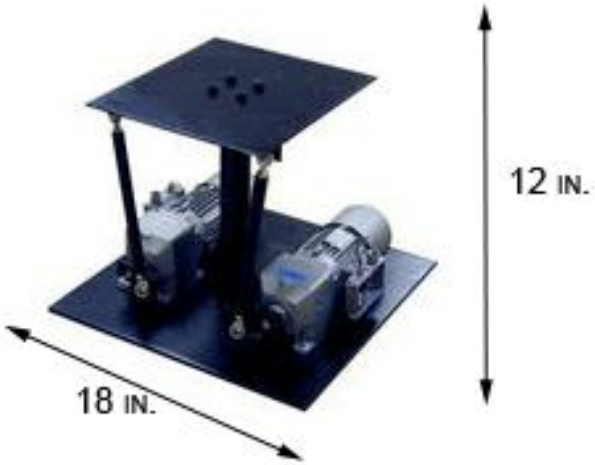
Motion Base

- COTS item loaned from Servos & Simulation Inc.
- 2 degrees of freedom achieved by 2 Servomotors, operational at 220V. (Transformer was provided)
- Max tilt of 20 degrees for pitch and roll



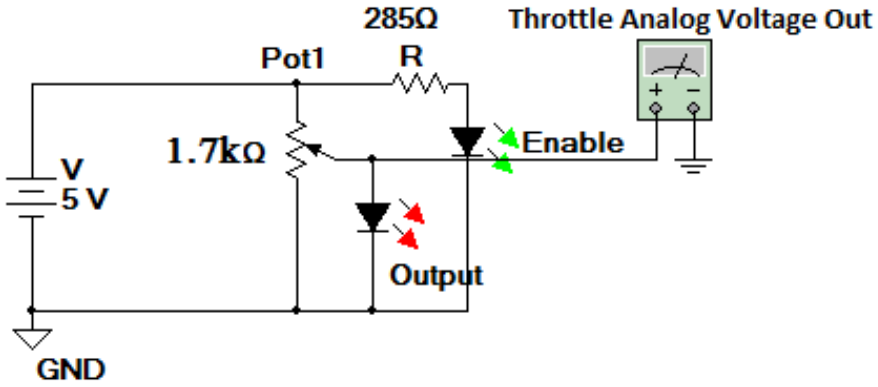
Motion Base Continued…

- Drive amplifiers allow each motor to maintain rotational position with a control signal range of +/-7 VDC
- 2 Potentiometers provide position feedback for implementation of a closed loop system
- Includes a 5/15VDC power supply which will be used for the other components



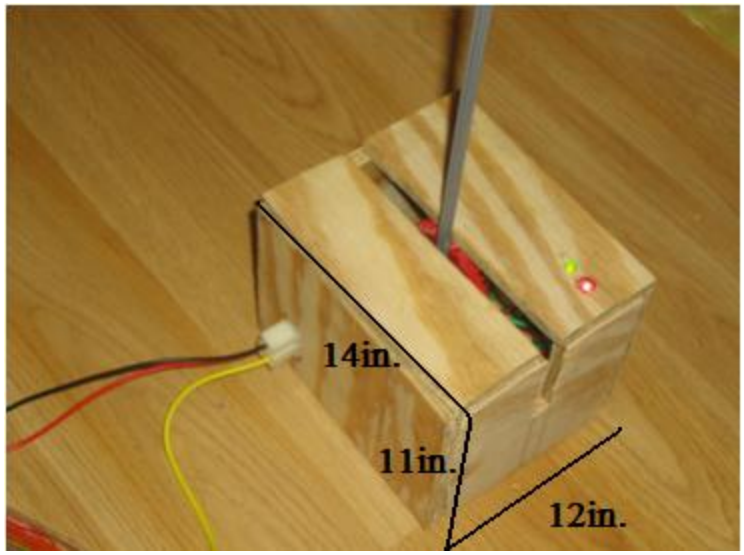
Throttle

- Linear response, rotary potentiometer used to track position.
- Wiper terminal outputs voltage ranges from 0-5V
- Ratio between 2 gears connecting shaft and lever allow for the complete use of 270°



Throttle

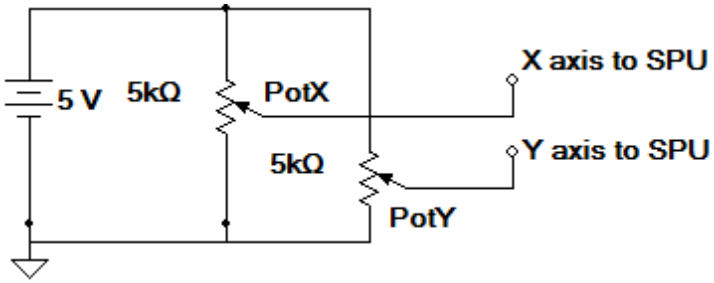
- Provides pilot with a lever range of 140°

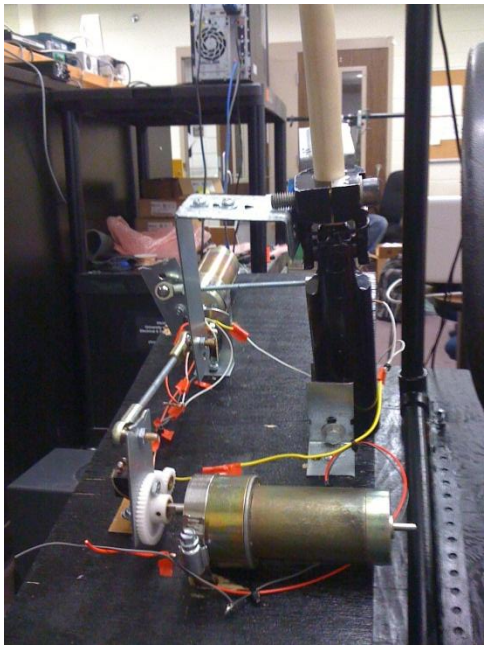


Lever Angle	Wiper Voltage(V)
7°	.04
30°	.17
50°	1.07
70°	1.66
90°	1.73
110°	1.84
130°	2.15
150°	2.52

Joystick

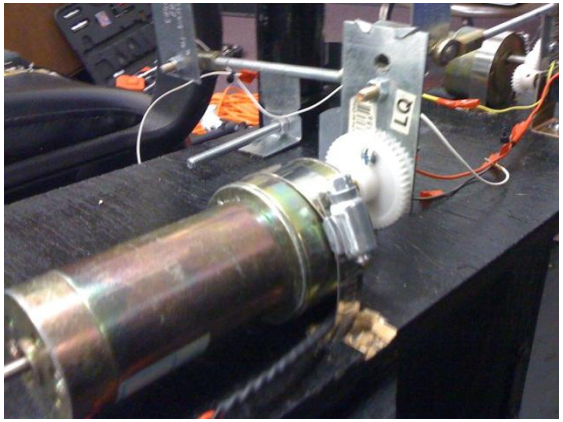
- Mechanically built to interface with rotary potentiometers on both the x and y axis and motors. 45° of handle rotation give 135° range on the potentiometer by using 3:1 gear ratio





Haptic Feedback

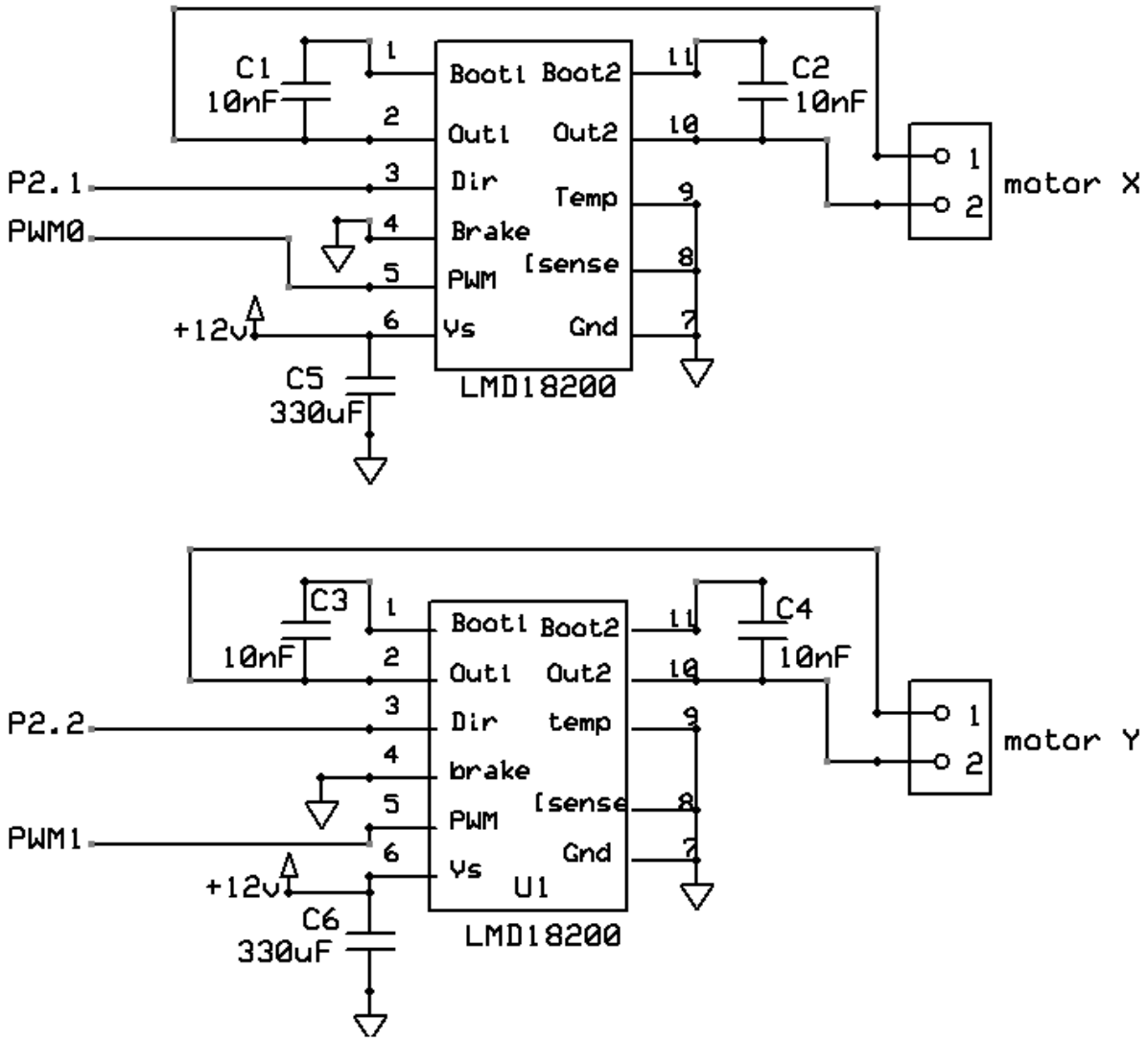
- Goal was to provide torque to counteract user's input
- Initial design would interface load cells with each axis to achieve force feedback:
  - Too expensive
  - Difficult to interface mechanically
- Joystick was interfaced with 2 DC motors:
  - Operate between 6-24V
  - Max Torque .0671Nm
  - Controlled digitally



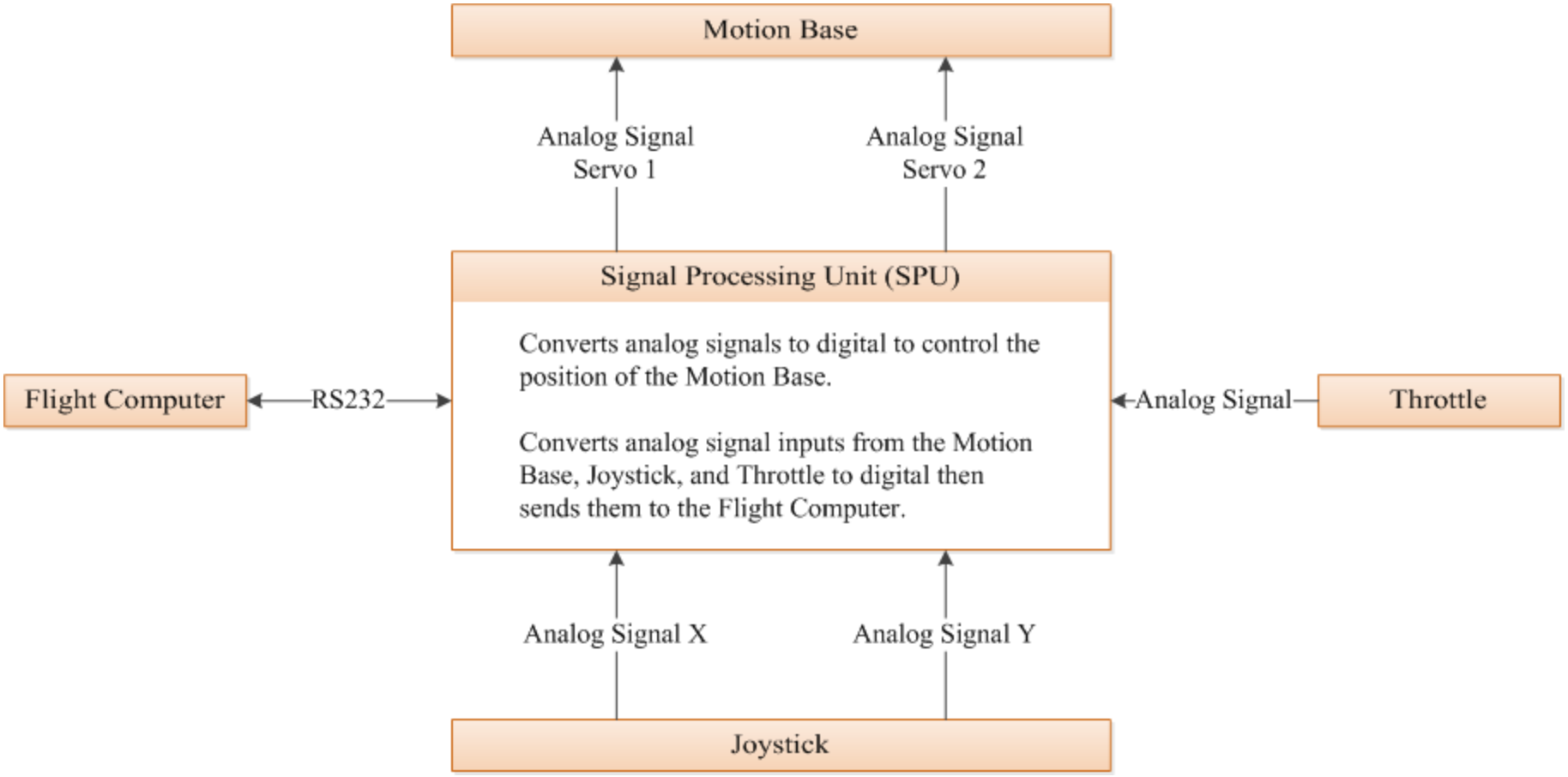
Haptic Feedback

- LMD18200 H-bridge along with ADUC841 processor from analog devices
- Handles up to 3A and 6A peak
- Motor control via PWM was accomplished but considered unsafe to fly simulator with feedback feature
- Issues were encountered with H-bridge
  - Stall currents and constant direction switching damaged IC's

Haptic Feedback Schematic



Signal Processing Unit (SPU)



SPU

- **What is the purpose of the SPU?**

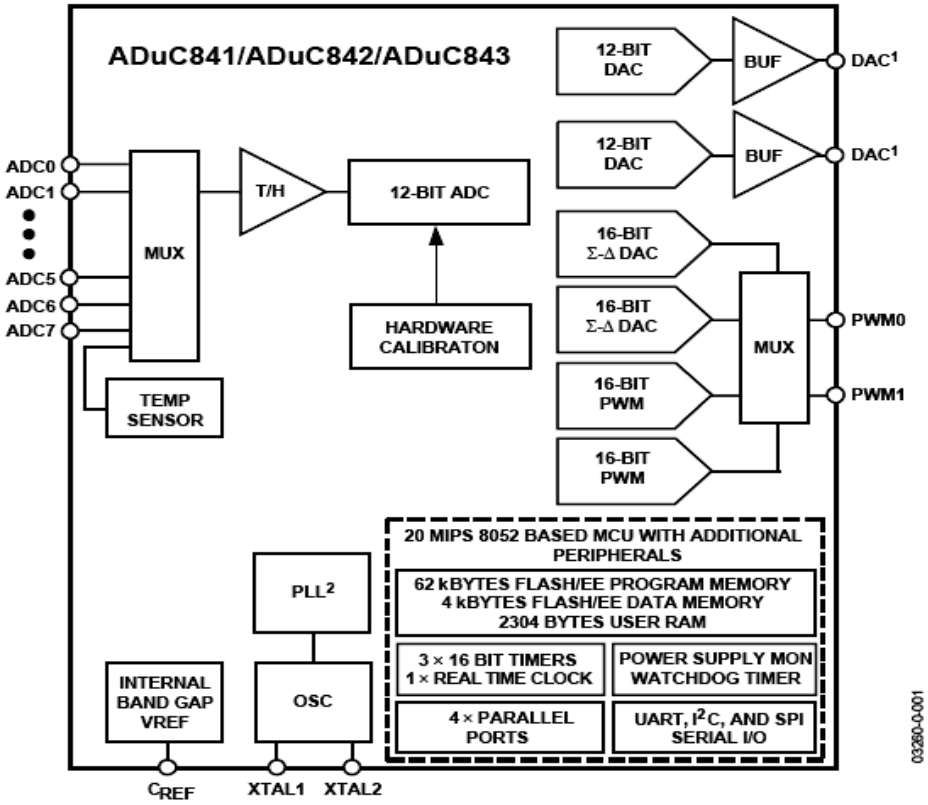
- Provide a centralized interface for communication between each system

- Perform secondary error checking on input and output range values

- Low cost solution

ADuC841 MicroConverter

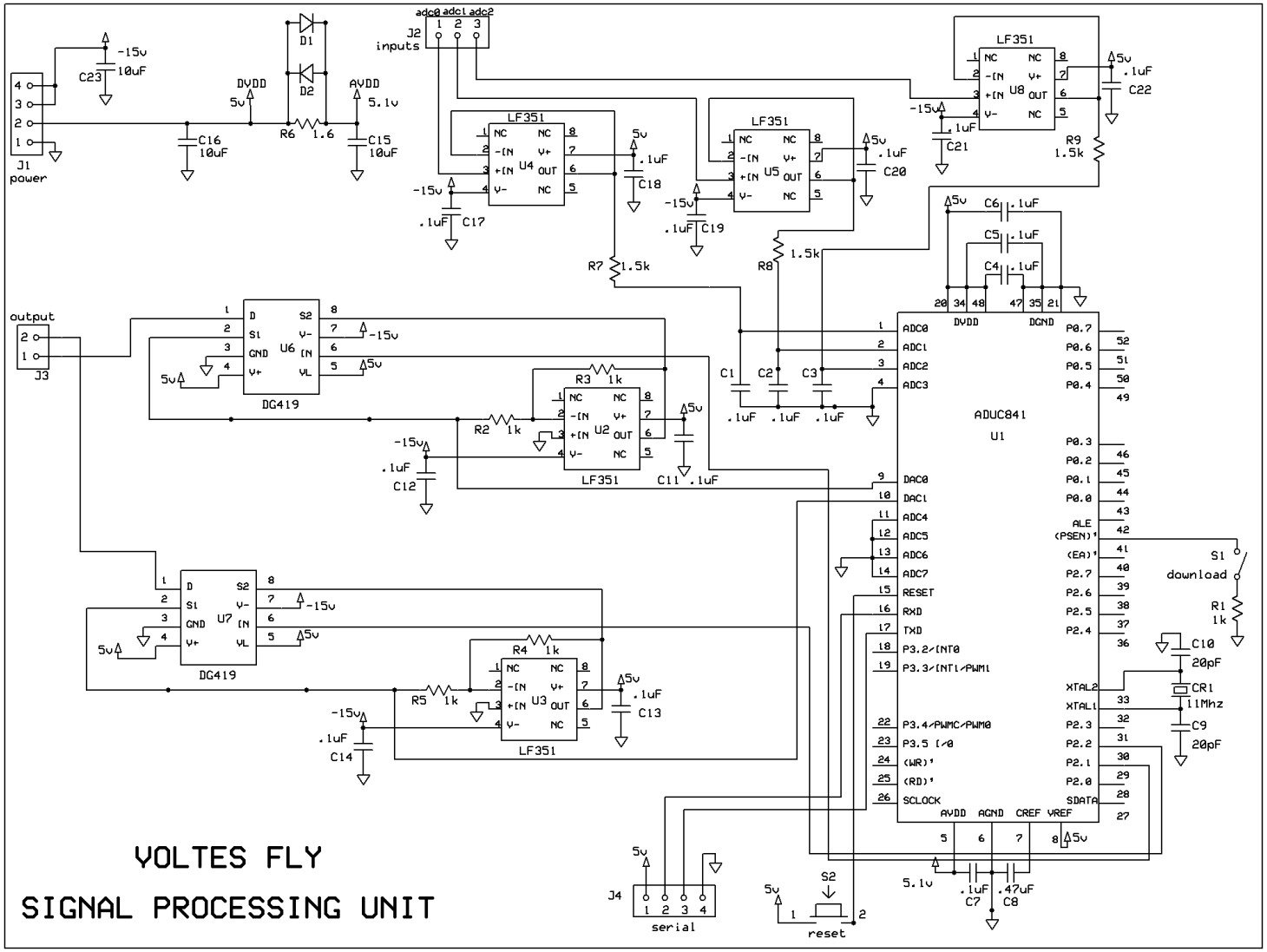
- **Specifications**
  - 2304 Bytes of SRAM
  - 8 Channel, 400kSPS, Self-Calibrating, 12-Bit ADC
  - Two 12-Bit Rail-to-Rail Voltage-Output DACs
  - Precision Voltage Reference , Serial Interface ports.



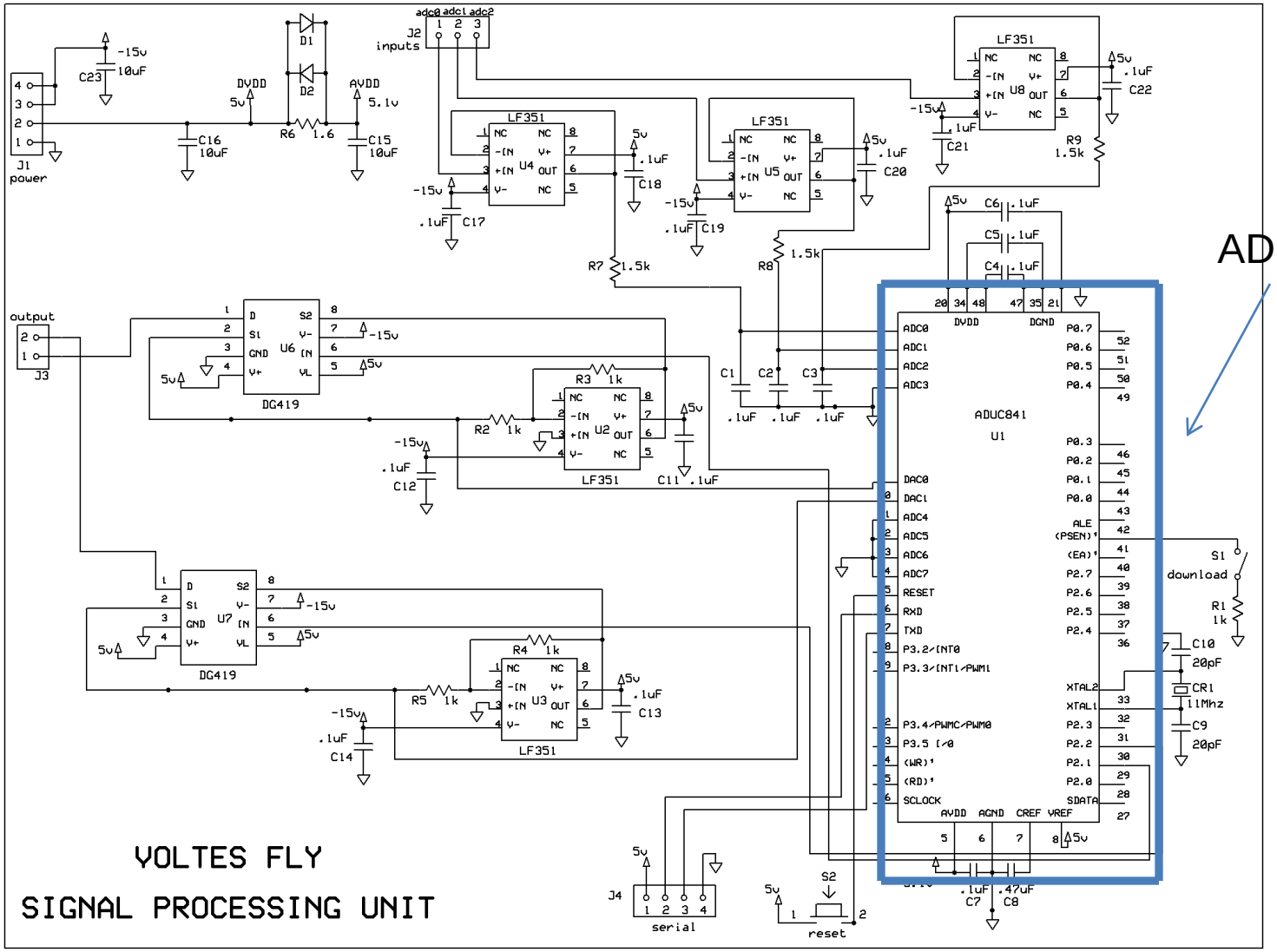
Printed Circuit Board

- Four layer PCB.
- Inner layers exclusively reserved for power and ground planes.
- Two copper layers on each end.
- Highly Portable.

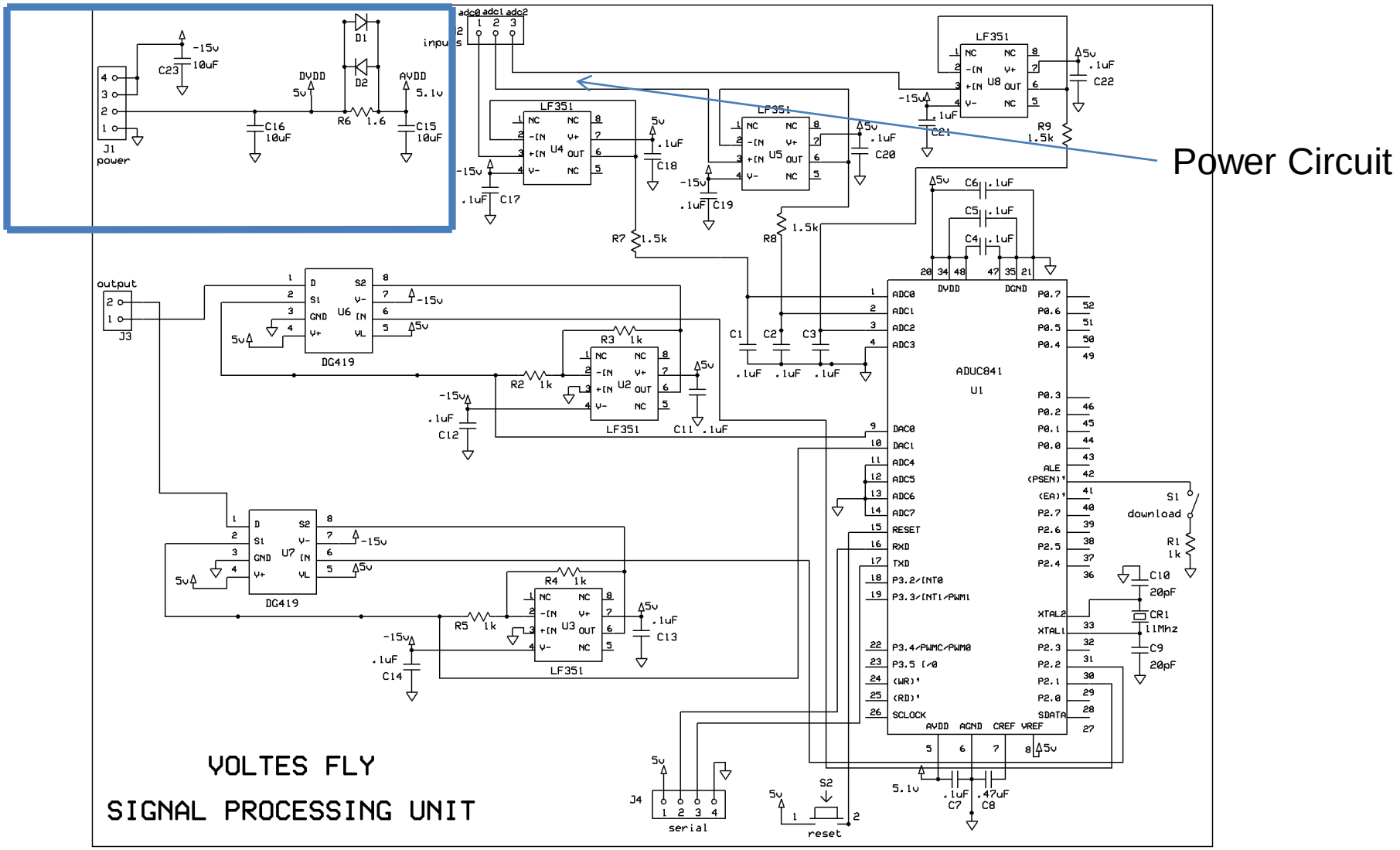
Printed Circuit Board Schematic



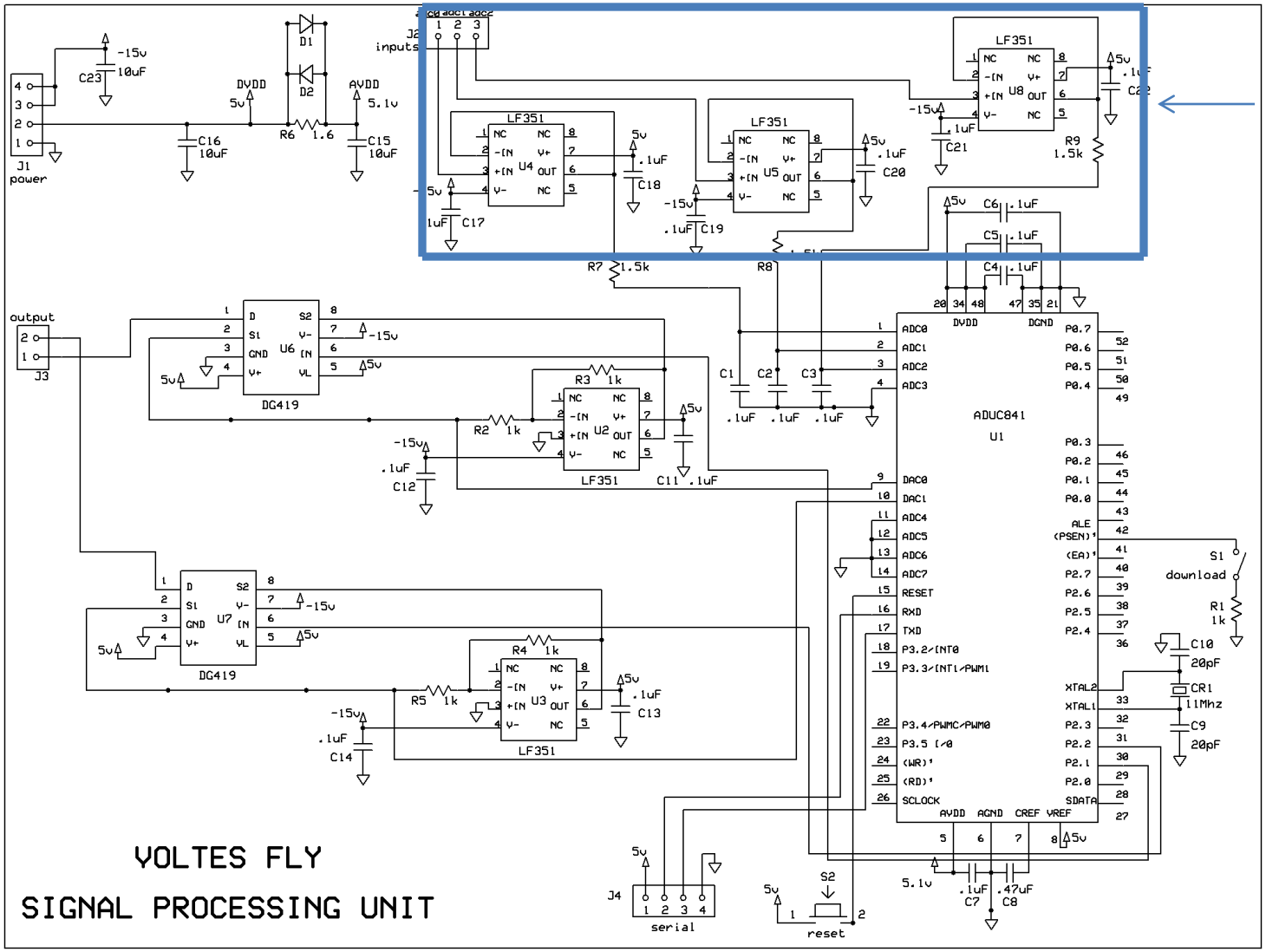
Printed Circuit Board Schematic



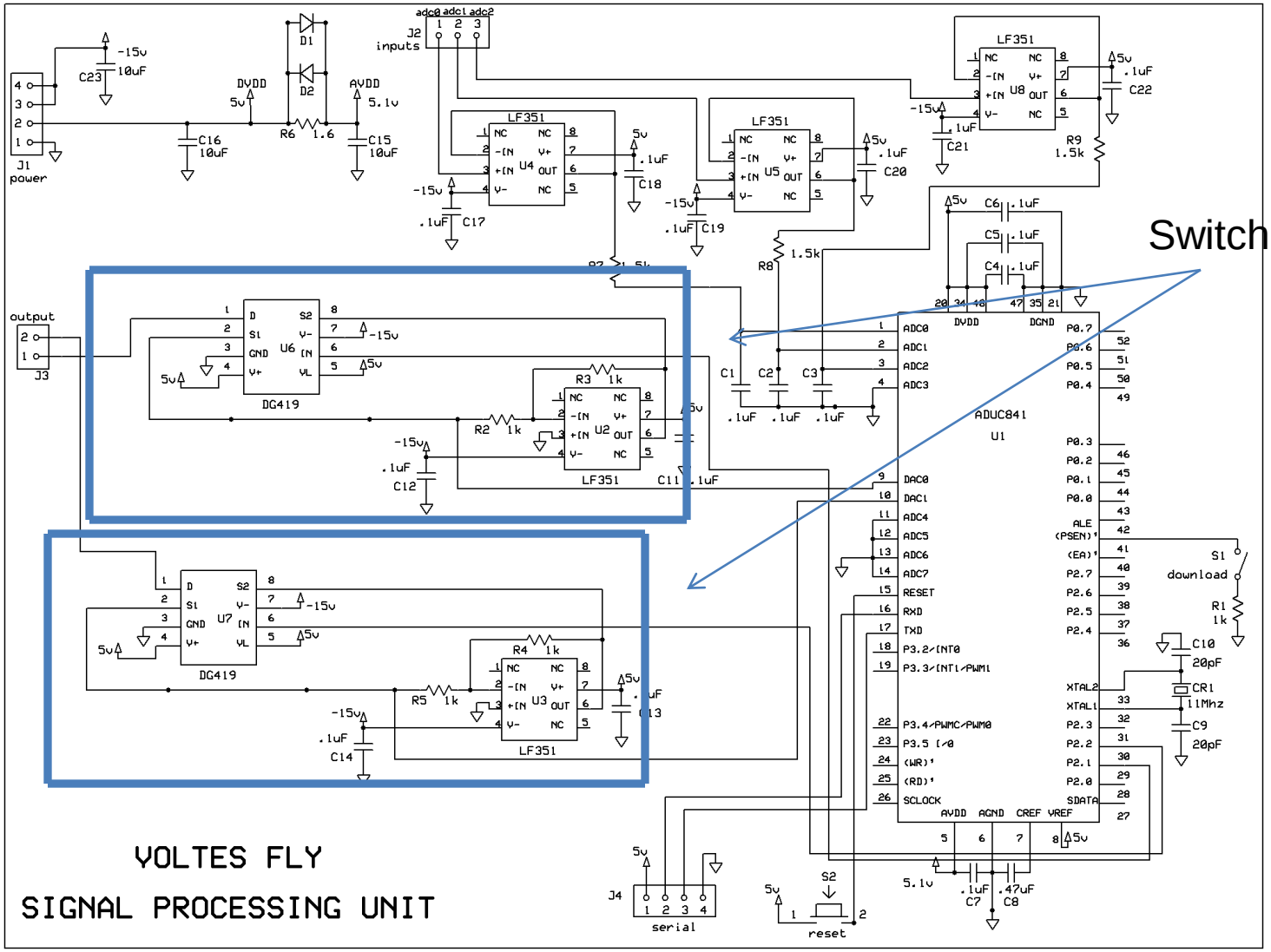
Printed Circuit Board Schematic



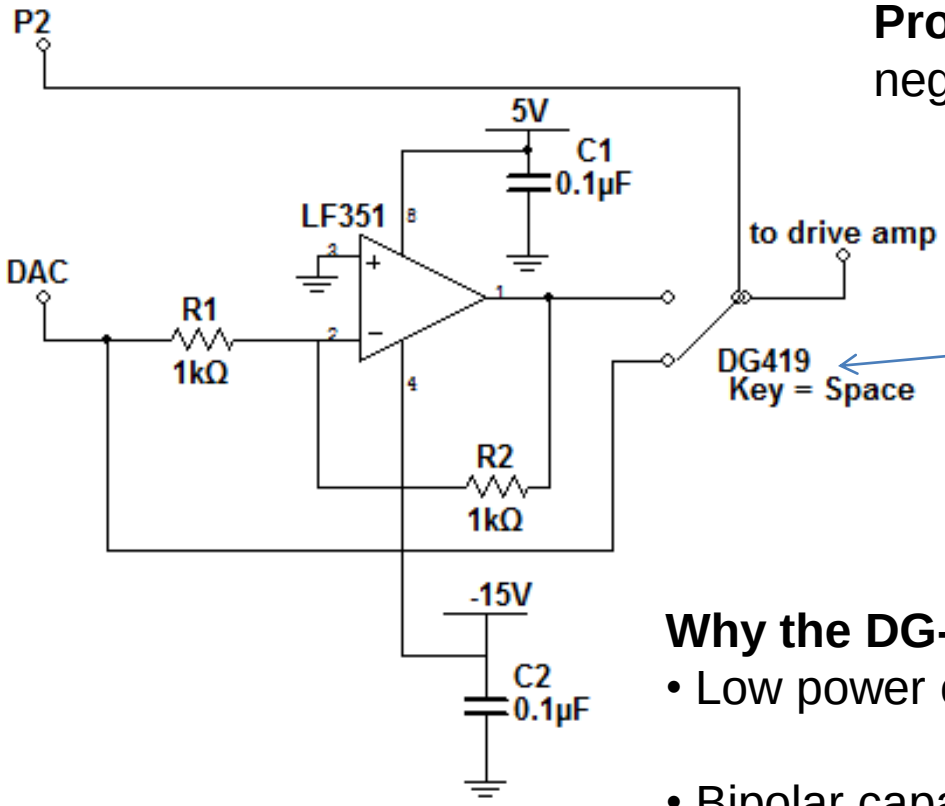
Printed Circuit Board Schematic



Printed Circuit Board Schematic



Switching Circuit



Problem: DACs do not output negative voltage values.

Solution: The use of a switching device, specifically the **DG-419**.

Why the DG-419:

• Low power consumption (35µW max).

• Bipolar capability - It can takes positive and negative input voltages.

• SPDT (single pole double throw) – Can pick from two signals based on a control input.

Packet Structure
Flight Computer to SPU

- **Header:** A smaller than symbol ‘<’, and the type and the length of the package values.

- **Voltage Values 1 & 2:** Represents the data payload of the packet, a 16 bit integer for both.

- **Check Sum:** It is a bitwise value containing the result of the check sum operation.

Packet Structure
SPU to Flight Computer

• **Header:** A smaller than symbol '<', and the type and the length of the package values.

•**Heartbeat:** It is an incremental value that tell us that the system is up and running.

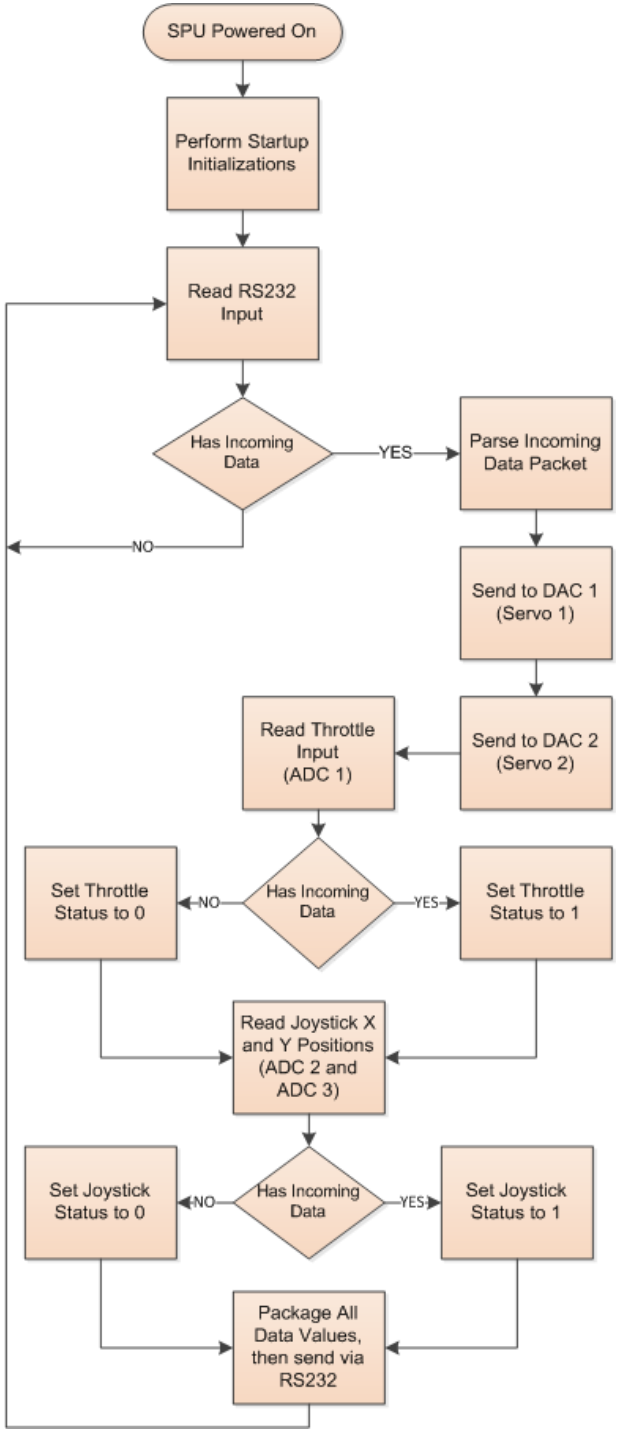
• **Throttle Position:** Value of the Throttle output.

•**Joystick X Position:** Joystick position along the X-Axis.

•**Joystick Y Position:** Joystick position along the Y-Axis

• **Joystick Status:** Notifies Flight Computer if the Joystick is connected to the SPU (1 if connected, 0 otherwise)

• **Throttle Status:** Notifies Flight Computer if the Throttle is connected to the SPU (1 if connected, 0 otherwise)



Embedded Software Flow

Main Functions:

• **main() :** Initialization of peripheral like the serial port, the DACs and the ADCs.

• **get_packet() :** Blocking function that process incoming packets and set the pace of the system.

• **process_packet() :** Perform swapping to incoming integers since the flight computer is little – endian and the microcontroller is big – endian.

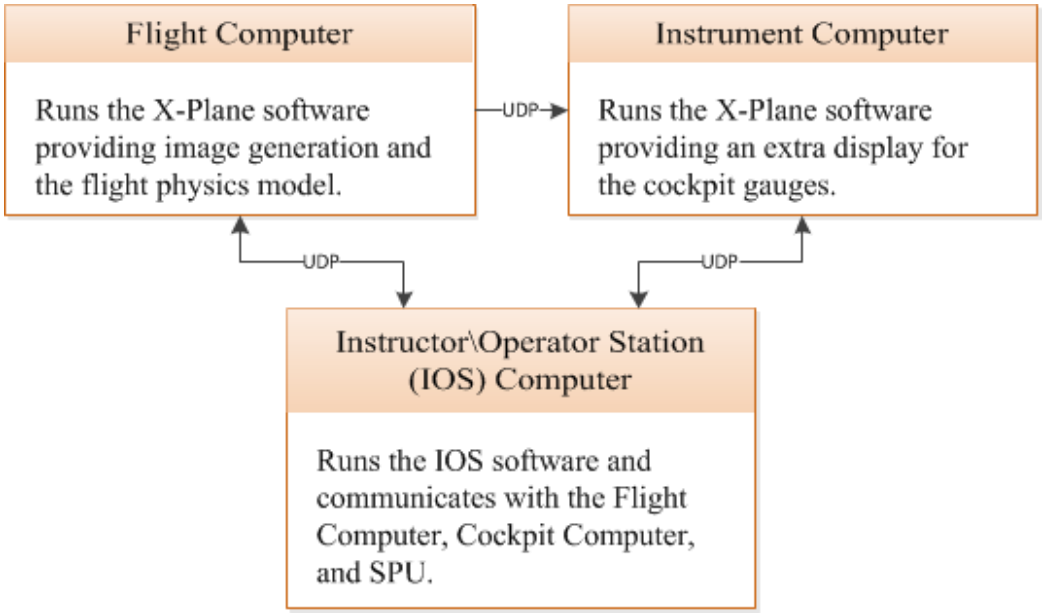
• **dac_out(voltage1, voltage2) :** Performs two actions.

a - Check the sign of the voltage value.
b – Send voltage values to both DACs.

• **send_serial_packet(package, length) :** Send package to flight computer.

• **do_crc(package, lenght) :** Perform Check Sum.

Flight Computer



- Specifications
  - Intel i7 Quad Core 2.80GHz
  - 8 GB RAM
  - Radeon HD 6570 graphics card with 1GB DDR3

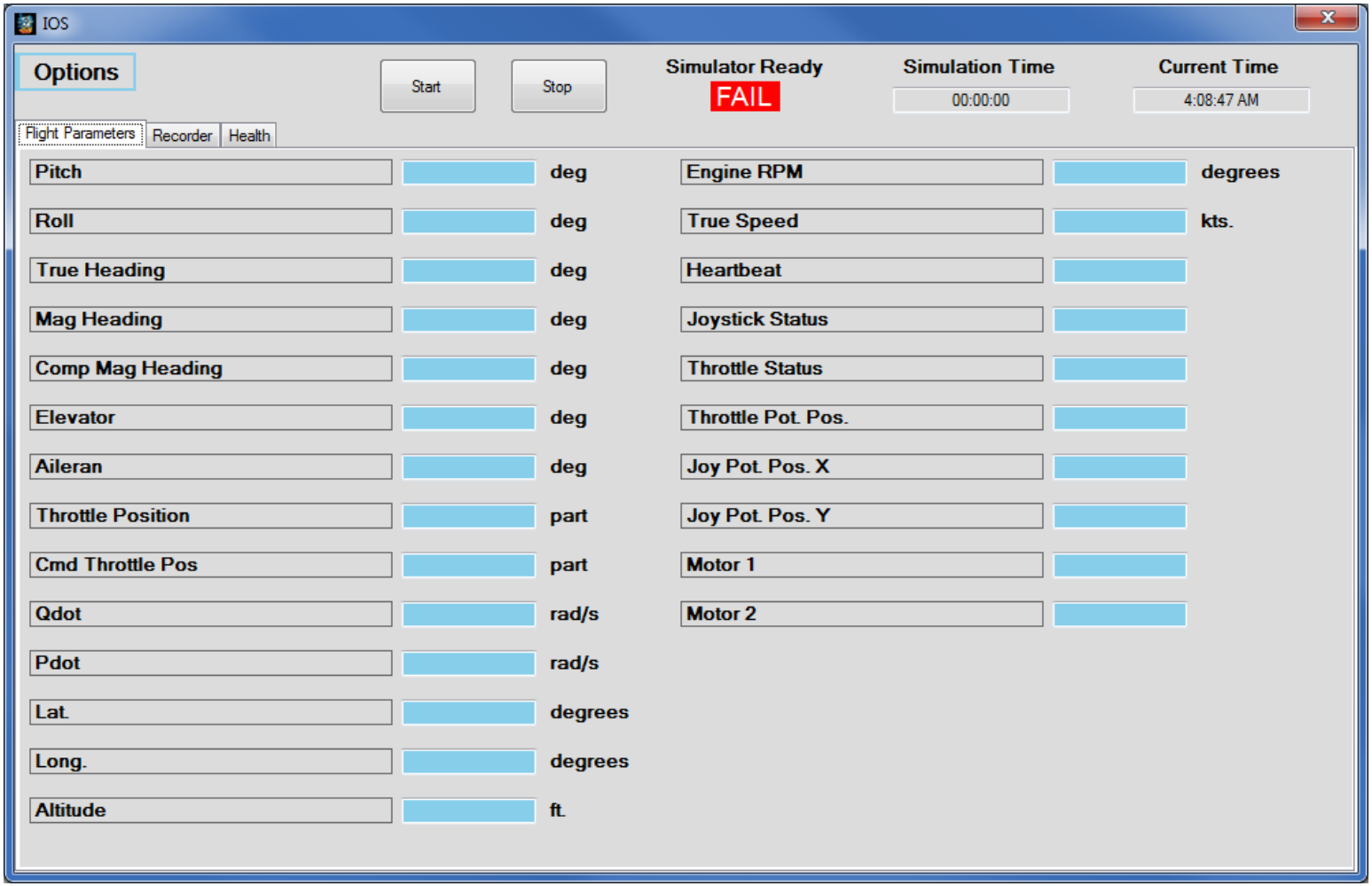
Instructor/Operator Station (IOS)

- Allows an instructor to monitor flight parameters for a simulation session
- Handles communication between X-Plane and the Signal Processing Unit
- 2 part design
  - GUI front end (called the IUI)
  - Simulation data processing interface (called the DPI)

IOS User Interface

- Scenario Page
  - Not implemented
- Flight Parameter Page
  - Displays flight parameters from the VoltesFly.xml file
    - Defines which values will be displayed and the variable in memory to update from
  - Allows instructor to control which values will be displayed without having to re-code
- Health Page
  - Displays the Health Status for each system
- Recorder Page
  - Allows the user to record any variable in memory and view either the raw data or as graph during an active simulation session

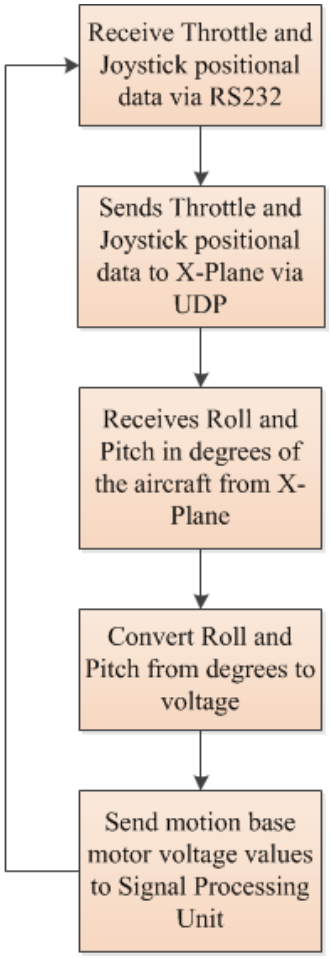
IOS User Interface Continued…



Simulator (backend)

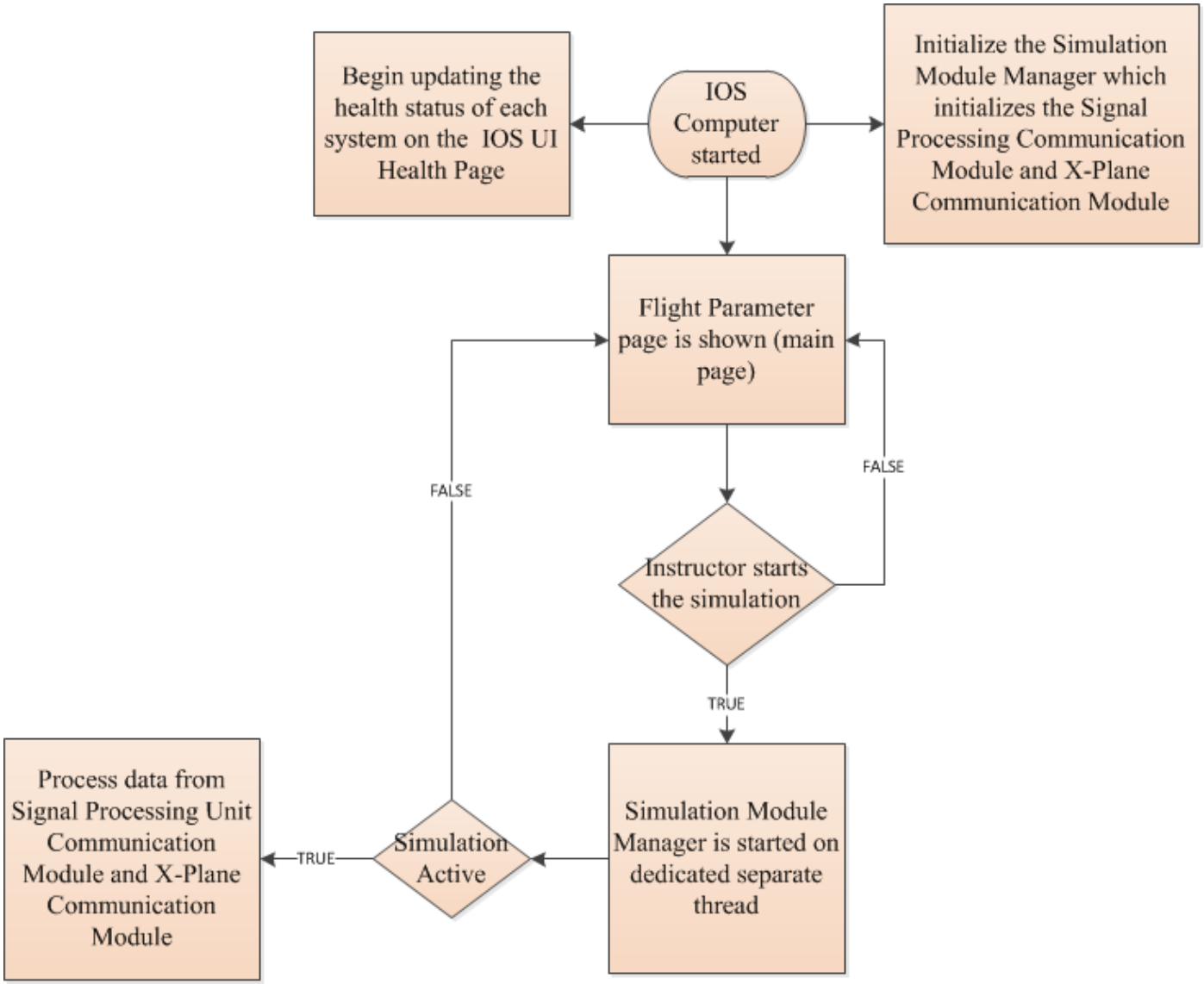
- Receives Joystick and Throttle data from the Signal Processing Unit via RS232
- Receives aircraft attitude (Roll and Pitch) data from X-Plane via UDP
- Converts Roll and Pitch from degrees to the voltage required for each motion base motor
- Data processing controlled by the transmission rate of X-Plane
- Follows a more functional programming design

Simulator (backend) continued…



- Runs on separate thread from IOS User Interface

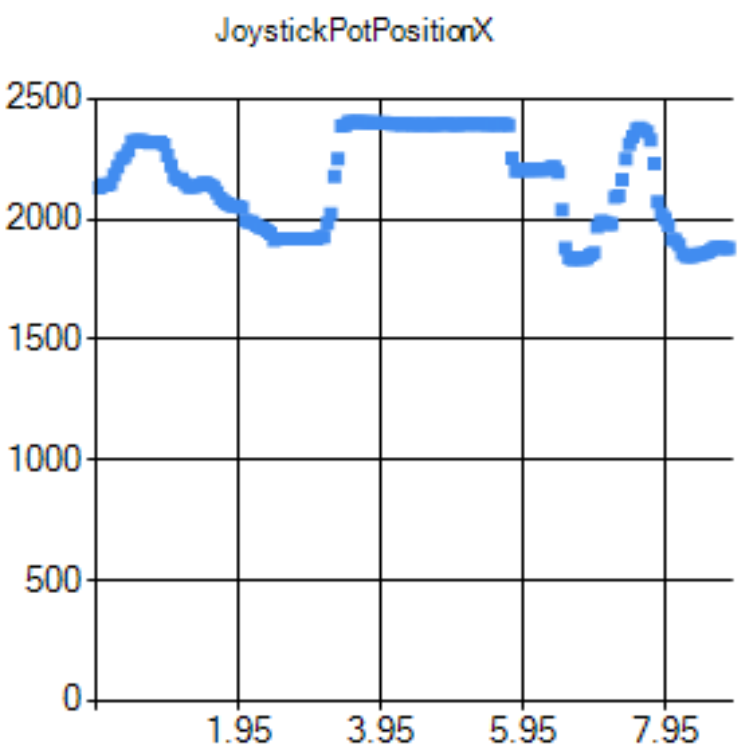
IOS Application Flow

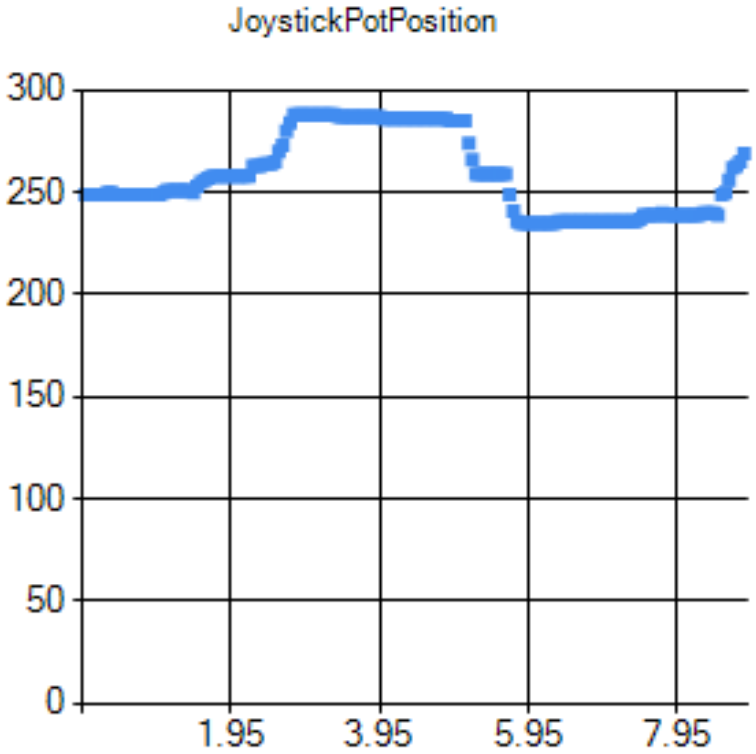


Testing

- Started with COTS Joystick, Throttle, and DAQ Data Acquisition (Measurement Computing)
- Replaced each component individually using a integrate and test approach
- Utilized Recorder functionality from the IOS

Testing


JoystickPotPositionX
2500
2000
1500
1000
500
0
1.95
3.95
5.95
7.95


JoystickPotPosition
300
250
200
150
100
50
0
1.95
3.95
5.95
7.95

Costs

Parts	Costs ($)
Cockpit construction	$200.00
Throttle	$19.87
Joystick	$228.00
SPU	$524.00
PCB	$108
Flight Computer	$0
X-plane flight simulator software	$29
Motion Base Platform	Loaned

- Self sponsored. Total without shipping reached $1108

Division of work

Parts	Member Responsible
Peripherals and Interfacing	Manuel Arredondo
Signal Processing Unit	Hector Bermudez
Instructor/Operator Station	Joe Paolini

Questions?