Introduction

This book is an attempt to lay the foundation for "friendlier" avionics. When all is well, there is time to thumb through books and charts or press the correct sequence of buttons to find the information needed on a Flight Management System (FMS). But, those of us that fly, know that when things go wrong, we must primarily rely on our training and memory. This is especially so when flying an aircraft certificated for single pilot operation, and there is no other crew to help.

This is not a theoretical book. The software included may be used to form the framework of a functioning system. It is hoped the avionics designers will use this as a starting point for future designs. The software is royalty free, but of course, provided as is, with no warranty. (Royalty free does not mean no credit need be given the various developers, in particular here, Carnegie-Mellon University for the Automatic Speech Recognition (ASR) and Nagoya Institute, Japan for the Text to Speech (TTS) system.)

There have been many articles written about Virtual Co-Pilots, Virtual Flight Officers, Electronic Pilot Assistants, among many other names. All suggest the ability of the crew to freely interact with the aircraft in a natural way using speech. The HAL 9000 in "2001, A Space Odyssey" was a memorable example of this natural interaction. Of course, what we heard was the voice of actor Douglas Rain (http://en.wikipedia.org/wiki/HAL_9000), but it was easy to believe that it was a computer speaking. As a side note, the voice of SAL in the sequel was Candice Bergen (Credited as Olga Mallsnerd) (http://en.wikipedia.org/wiki/HAL_9000#SAL_9000). Here, we will use the name Cognitive Pilot Assistant (CPA).

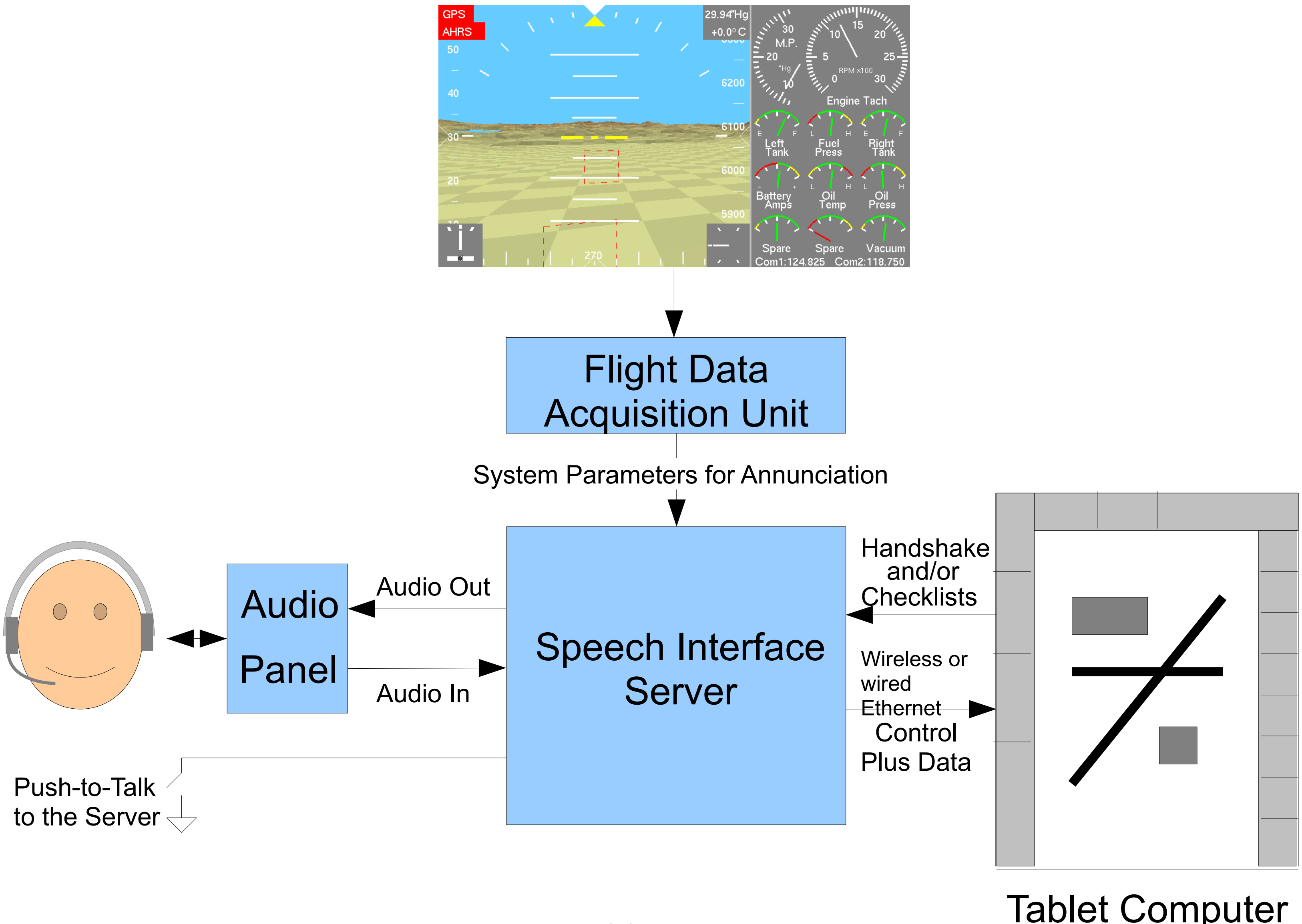
Automatic speech recognition (ASR) and text to speech synthesis (TTS) are not yet perfected today, but they can be helpful for interacting with avionics in useful ways. Tasks such as reciting checklists, reporting system anomalies, finding and displaying charts, are well within today's ASR/TTS systems capabilities.

In the chapters to follow, the code for a system that can do these tasks is discussed. The code for the aircraft interface is written entirely in the 'C' programming language. The code for the Android tablet is written in Android Java. The presentation will be a top down view. Some code details and snippets will be used to illustrate some points. But, not being a 'C' or Android programmer is not a hindrance to understanding the structure and flow of the programs. The code is "proof by example" that real systems may be designed that use the concepts presented.

The system to be described consists of three major components: An Android Tablet PC capable of displaying charts, checklists, aircraft system diagrams, etc. A data acquisition system (DAQ) for aircraft information such as engine data, air data, weather data, and any other sensor data that may be available and of interest to the crew. Lastly, a central device providing a wired interface to the DAQ, a wireless interface to the tablet PC, and a connection to the audio panel for implementing a conversational agent using ASR/TTS. Figure I.1 is a diagram of the overall system.

Within each of these three components are several software modules and sub-modules as well. These components and modules make it convenient to divide the book into three parts with chapters devoted to the various software modules. Little detailed attention will be given to hardware. It is constantly evolving, rendering obsolescent any particular hardware implementation that might be described. However, some general comments and a few examples will be provided along the way.

Part I details the software program flow for a Tablet PC. This provides a display of charts, checklists, and aircraft systems. These may be requested with the standard touch screen interface or by voice using the conversational agent embedded in the Cognitive Pilot Assistant software.



Speech Server System with Connected Tablet.

Figure II.1

The use of the term *Cognitive* in the name is intended to convey the notion that the system is aware of the state of the aircraft and will alert the pilot to anomalies, as well as offer assistance when needed.

Part II details the software program flow for the CPA. This includes the wireless access protocol, the automatic speech recognition and natural language processing software, the text to speech software, the interface to the Tablet PC, and the interface to the aircraft data acquisition system.

Part III discusses the software for the Avionics Data Acquisition System (ADAQ). Some examples of hardware are also provided. The interface to each model aircraft is likely to be different, so an attempt is made to provide sufficient information to address the many possible configurations.

To the extent possible, each Part may be read independently of the others. This requires a bit of repetition of some of the material, but not much. The Tablet, CPA, and ADAQ modules are well defined separate entities that communicate with each other, but are not dependent on each other.

It is not assumed that the reader has some experience writing 'C' code and/or Android applications. For those that do, the code will add to the experience of this book. But, as mentioned previously, its primary purpose is to give credence to the concepts and techniques presented. Understanding the details is less important.

All the software is on the accompanying CD-ROM. To the extent possible, it is well commented. In the chapters to follow, the software will be explained in a top down view. Some code snippets are used to illustrate particular points, but the software is not described in line by line detail.

The speech recognition software is either Sphinx 3 or PocketSphinx. If the reader elects to use either of these in a deployed system, please note and respect the copyright of Carnegie-Mellon University.

The text to speech software is a combination of Festival and HTS. These are copyrighted by the University of Edinburgh and Nagoya Institute of Technology respectively. Dr. Alan Black wrote the C code Flite+HTS which was modified for our application. Again, please respect their copyrights.