



**Escola Politècnica Superior  
de Castelldefels**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# TREBALL DE FI DE CARRERA

**TÍTOL: Disseny de pantalles cartogràfiques pel simulador de cabina d'un avió**

**TITULACIÓ: Enginyeria Tècnica de Telecomunicació, especialitat Sistemes de Telecomunicació**

**AUTOR: Joan Grasas León**

**DIRECTOR: Enric Pastor Llorens**

**CODIRECTOR: Xavier Prats Menéndez**

**DATA: 4 de setembre de 2005**

**Títol:** Disseny de pantalles cartogràfiques pel simulador de cabina d'un avió

**Autor:** Joan Grasas León

**Director:** Enric Pastor

**Data:** 5 de setembre de 2005

## Resum

La pantalla cartogràfica d'una cabina d'avió facilita al pilot una correcta adquisició visual del tràfic aeri circumdant, millora la seguretat i l'eficàcia d'operacions de vol i permet als avions obtenir posicionament i rumb més eficients en zones remotes i oceàniques.

L'objectiu d'aquest treball de fi de carrera és dissenyar una pantalla cartogràfica de baix cost, programar una llibreria que contingui totes les estructures necessàries per a la seva posterior programació així com realitzar un estudi dels mitjants maquinari necessaris per a la seva implementació.

El disseny de la pantalla cartogràfica s'ha basat en les recomenacions donades per *RTCA inc.* sobre el *Cockpit Display of Traffic Information*. El CDTI convencional obté les dades necessàries pel seu funcionament de múltiples sistemes de vigilància com ara ADS-B, TIS, TIS-B i TCAS. Valorant les característiques de les diferents fonts, s'ha triat el sistema ADS-B com el més idoni per a l'obtenció de les dades de vigilància per a la pantalla cartogràfica dissenyada ja que és el sistema amb una cobertura més ampla i és el sistema més extès.

ADS-B (*Automatic Dependent Surveillance – Broadcast*) és una aplicació a nivell d'enllaç de dades que transmet mitjançant un *transponder* dades de vigilància per mode *broadcast* a tots els usuaris que disposin d'un receptor que treballi a 1090 MHz.

En quant a maquinari, el sistema constarà d'emissió/recepció de dades, de processament i de visualització. L'emissió/recepció s'obtindrà mitjançant un *transponder Mode S ES 1090MHz* ja que és el mitjà triat tant a Europa com a EEUU. El processament i la visualització s'efectuaran mitjançant un *Panel PC* perquè integra d'una forma compacta i robusta pantalla TFT i computador d'altres prestacions a bon preu.

**Title:** Cartographic screen inside an aircraft cockpit design

**Author:** Joan Grasas León

**Director:** Enric Pastor

**Date:** September, 4th 2005

## Overview

The cartographic screen inside an aircraft cockpit makes easy to the pilot a correct vision of the aerial traffic around, improves the security and the efficiency of the flight operations and allows the aircrafts to obtain positioning and routes more efficiently in remote and oceanic areas.

The objective of this thesis is to design a low cost cartographic screen, to program a library that contains all the necessary structures for the subsequent programming and the study of the necessary hardware for its implementation.

The design of the cartographic screen is based on the recommendations of the *RTCA inc.* about the *Cockpit Display of Traffic Information*. The conventional CDTI gets the necessary data for its performance of multiple surveillance systems such as ADS-B, TIS, TIS-B and TCAS. Taking into account the characteristics of the different sources, the system ADS-B has been chosen as the most suitable one to obtain surveillance data for the designed cartographic screen as this is the most extended and most used range system.

ADS-B (Automatic Dependent Surveillance - Broadcast) is a data link application that transmits surveillance data through a transponder by means of a broadcast mode to users who have a receptor at 1090MHz.

As for hardware, the system will consist of data emission/reception, processing and visualisation. The Emission/Reception will be obtained by means of a transponder Mode S ES 1090MHz as it is the system chosen in both Europe and USA. The processing and visualisation will be obtained through a PC Panel that integrates in a compacted and robust way a TFT screen and a high asset computer at a low price.

# ÍNDIX

<b>INTRODUCCIÓ .....</b>	<b>1</b>
<b>CAPÍTOL 1. ESTÀNDARD CDTI.....</b>	<b>3</b>
1.1. Objectius operacionals .....	4
1.2. Conjectures .....	5
1.3. Aplicacions de vigilància del CDTI.....	5
<b>CAPÍTOL 2. FONTS DE DADES .....</b>	<b>7</b>
2.1. ADS-B .....	7
2.2. TIS.....	8
2.3. TIS-B.....	9
2.4. TCAS .....	10
2.5. Diferències entre les fonts de vigilància.....	11
2.6. Fusió de dades .....	13
2.7. Altres consideracions en el processament de dades de vigilància .....	14
<b>CAPÍTOL 3. DISSENY CDTI.....</b>	<b>15</b>
<b>3.1. Llibreria.....</b>	<b>15</b>
3.1.1. Classe <i>clibrary</i> .....	15
3.1.1.1. Constructors; Destructor .....	15
3.1.1.2. Atributs .....	15
3.1.1.3. Mètodes .....	16
3.1.2. Classe <i>cplane</i> .....	16
3.1.2.1. Constructor; Destructor .....	16
3.1.2.2. Atributs .....	16
3.1.2.3. Mètodes.....	17
3.1.3. Classe <i>cflightpath</i> .....	17
3.1.3.1. Destructor .....	17
3.1.3.2. Atribut .....	17
3.1.3.3. Mètode .....	17
3.1.4. Classe <i>ccoordinate</i> .....	17
3.1.4.1. Constructors.....	17
3.1.4.2. Atributs .....	18
3.1.5. Fórmules .....	18

3.1.5.1. Distància entre dos punts.....	18
3.1.5.2. Equivalència entre milles nàutiques i radiants .....	18
3.1.5.3. Equivalència entre milles nàutiques i metres .....	19
3.1.5.4. Equivalència entre metres per segon i nusos(kts).....	19
3.1.5.5. Equivalència entre peus i metres .....	19
<b>3.2. Aplicació visual .....</b>	<b>19</b>
3.2.1. Funcions característiques.....	19
3.2.2. Característiques generals de visualització .....	19
3.2.2.1. Característiques mínimes de visualització i informació de l'objectiu.....	20
3.2.2.2. Altres característiques de visualització de l'aplicació .....	20
3.2.2.3. Alertes de tràfic .....	21
3.2.3. Opcions de visualització.....	22
<b>CAPÍTOL 4. MAQUINARI .....</b>	<b>24</b>
<b>4.1. Introducció .....</b>	<b>24</b>
<b>4.2. Emissió/Recepció .....</b>	<b>25</b>
4.2.1. Elecció del mode de comunicació .....	25
4.2.2. Transponder Mode S ES.....	27
4.2.2.1. Honeywell.....	27
4.2.2.2. Rockwell Collins .....	28
4.2.2.3. ACSS .....	29
4.2.2.4. Garmin .....	30
<b>4.3. Processament i Visualització .....</b>	<b>31</b>
4.3.1. ATIOSys.....	31
4.3.1.1. ATP-120E.....	31
4.3.2. Protech.....	32
4.3.2.1. PPC-7360.....	32
4.3.3. Axiomtek .....	33
4.3.3.1. GOT-1120T .....	33
4.3.3.2. GOT-2770S .....	34
4.3.4. Acnodes .....	34
4.3.4.1. PC1060 .....	35
<b>CAPÍTOL 5. CONCLUSIONS .....</b>	<b>36</b>
<b>5.1. Objectius assolits .....</b>	<b>36</b>
<b>5.2. Desviació sobre el projecte inicial .....</b>	<b>36</b>
<b>5.3. Possibles millores i ampliacions futures .....</b>	<b>37</b>
<b>5.4. Impacte medioambiental.....</b>	<b>37</b>
<b>5.5. Conclusions personals .....</b>	<b>37</b>
<b>BIBLIOGRAFIA .....</b>	<b>39</b>

<b>ANNEX A. GLOSSARI .....</b>	<b>41</b>
<b>ANNEX B. CODI LLIBRERIA .....</b>	<b>43</b>
<b>B1. Coordinate.h .....</b>	<b>43</b>
<b>B2. Coordinate.cpp .....</b>	<b>44</b>
<b>B3. Plane.h.....</b>	<b>45</b>
<b>B4. Plane.cpp.....</b>	<b>47</b>
<b>B5. Flightpath.h .....</b>	<b>49</b>
<b>B6. Flightpath.cpp .....</b>	<b>50</b>
<b>B7. Library.h.....</b>	<b>51</b>
<b>B8. Library.cpp.....</b>	<b>53</b>
<b>B9. Formulas.cpp.....</b>	<b>56</b>
<b>B10. Fitxers de càrrega .....</b>	<b>58</b>
<b>ANNEX C. DATASHEETS PANEL PC .....</b>	<b>59</b>
<b>C1. ATP-120E .....</b>	<b>61</b>
<b>C2. PPC-7360 .....</b>	<b>63</b>
<b>C3. GOT-1120T .....</b>	<b>65</b>
<b>C4. GOT-2770S.....</b>	<b>67</b>
<b>C5. PC1060.....</b>	<b>69</b>

# INTRODUCCIÓ

## Idea i raó del projecte

L'espai aeri és limitat i amb la creixent demanda de transport aeri és necessari aprofitar-lo al màxim ja que és un bé escàs. Per tant, és molt important aconseguir gestionar-lo d'una forma el més eficient possible per tal d'optimitzar-lo i aconseguir que l'utilitzin el major nombre d'avions possible.

Una eina que s'utilitza per aquest fi és la pantalla cartogràfica de les cabines dels avions que permet al pilot fer-se una idea molt fidedigne de l'estat del cel per tal de poder planificar adequadament el seu pla de vol. A més a més, permet millorar la seguretat i l'eficàcia de les operacions de vol i obtenir posicionament i rumb més precisos en zones remotes i oceàniques.

Aquest projecte es basa en la realització del disseny d'una pantalla cartogràfica de baix cost. S'ha basat en les recomenacions de l'estàndard elaborat per *RTCA inc.* sobre *Cockpit Display of Traffic Information (CDTI)* [1] i enfocat a una aplicació en els avions d'extinció d'incendis del departament d'*Interior de la Generalitat de Catalunya – Direcció General d'Emergències i Seguretat Civil*.

## Objectius

L'objectiu d'aquest treball de fi de carrera és realitzar el disseny d'una pantalla cartogràfica, programar una llibreria que contingui totes les estructures necessàries per a la seva posterior programació així com realitzar un estudi dels mitjants maquinari necessaris per a la seva implementació.

## Estructura de la memòria

Al primer capítol es troba la informació relacionada amb l'estàndard de CDTI en que s'ha basat el disseny.

Al segon capítol es descriuen les fonts de dades de vigilància que poden ser emprades pel funcionament del CDTI.

Al tercer capítol s'exposa el disseny de la pantalla cartogràfica de baix cost realitzat així com una breu descripció de la programació realitzada per a programar la llibreria.

Al quart capítol es mostrarà el maquinari necessari per a la implementació del CDTI així com les principals característiques tècniques.

Al cinquè i últim capítol s'exposaran les conclusions tretes de la realització d'aquest treball així com els criteris d'ambientalització i les línies futures més rellevants.

Finalment, als annexos es troba el glossari, el codi extés de la llibreria així com els datasheets de diversos components maquinari escollits.



## CAPÍTOL 1. Estàndard CDTI

El *Cockpit Display of Traffic Information* (CDTI) té per funció presentar informació de vigilància del tràfic aeri circumdant a la tripulació de vol. La informació presentada inclou la posició relativa de les altres aeronaus respecte la mateixa. Aquesta informació del tràfic aeri de què disposa el CDTI pot ser obtinguda de una o més fonts, incloent, *Broadcast Automatic Dependent Surveillance* (ADS-B), *Traffic Information Service* (TIS), *Broadcast Traffic Information Service* (TIS-B), i *Traffic Alert and Collision Avoidance System* (TCAS). En el capítol següent s'explicaran les principals característiques d'aquests sistemes de vigilància.

Per tal de mostrar informació de tràfic, el CDTI pot ser utilitzat com a dispositiu de pantalla o com a dispositiu de pantalla compartida multifunció (MFD). Pel que fa a la presentació visual, la representació gràfica del tràfic pot ser mostrada des de diferents proximitats, diversos tipus de presentacions (p.e., auditiu, dispositiu *head-up display*), per tal que la informació pugui ser interpretada eficientment. La informació específica que es presenta pot variar en base a l'ús que es vulgui donar a l'esmentada informació.

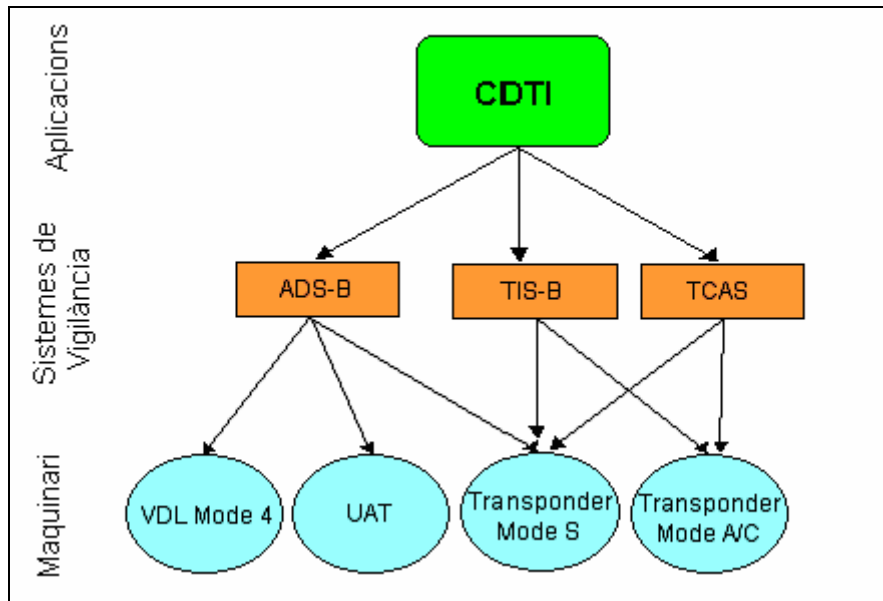
Per a distàncies curtes, la pantalla de tràfic pot presentar informació addicional provinent d'altres sistemes com pot ser el TCAS per tal de mostrar les situacions de conflicte i alerta amb altres avions. Els avions que no estiguin equipats amb TCAS hauran d'obtenir l'informació pel CDTI a partir de la informació del ADS-B i/o TIS.

El estàndard seguit és el realitzat pel Special Committee 186 de *RTCA inc.* i aprovat el 19/02/1998 [1]. Per tal de complir amb els estàndards de certificació, els fabricants estaran obligats a demostrar que CDTI funciona segons les seves especificacions i no interfereix amb cap altre sistema de l'aeronau.

El ADS-B MASPS (*Minimum Aviation System Performance Standards*) [2] defineix un rang ampli de usos operacionals del CDTI, aplicables a tots els dominis. Degut a la seva varietat i complexitat, les aplicacions poden ser desenvolupades per a diferents usos. Els dispositius futurs poden afegir prestacions per a noves o millorades aplicacions.

L'especificació, la transmissió i la recepció de dades de vigilància (TIS, ADS-B, TIS-B, etc.) són crítics per a la implementació del CDTI.

A continuació es presenta un esquema amb els diferents sistemes que intervenen en un CDTI (Figura 1.1).



**Fig. 1.1** Esquema dels Sistemes d'un CDTI

El CDTI obté les dades necessàries pel seu funcionament de diferents sistemes de vigilància; en la Figura 1.1 es presenten els més significatius ara per ara encara que s'hi podrien afegir alguns d'altres com ara TIS o algun altre sistema implementat en un futur. Aquests sistemes de vigilància obtenen les seves dades dels diferents modes de transmissió que seran explicats en el capítol 4.

## 1.1. Objectius operacionals

Les metes operacionals del CDTI són millorar la seguretat i l'eficàcia d'operacions de vol a través d'una capacitat accentuada de vigilància combinada amb nous procediments operatius. Les anteriors aplicacions del CDTI inclouen els següents objectius i aplicacions:

- Millorar la coneixença de la situació de vol
- Millorar la probabilitat d'una correcta adquisició visual del tràfic en comparació a la búsqueda visual sense ajuda
- Millorar les rutines de vol i la seguretat dels apropaments visuals
- Retardar el canvi que es produeix en les operacions i en la capacitat quan es passa d'unes condicions meteorològiques aptes, com per permetre aproximacions visuals, a unes altres que requereixen operacions instrumentals (IFR).
- Permetre als avions obtenir posicionament i rumb més eficients en espai oceànic i remot. Redueix les restriccions separació/altitud entre parells d'avions equipats amb CDTI mitjançant procediments millorats com per exemple els procediments "*In-Trail Climb/Descent (ITC/ITD)*" millorats.

Es pot obtenir millores immediates abans de tenir disponibles tots els avions del voltant amb la utilització d'aplicacions CDTI entre parelles d'avions. Tot i existir aquest tipus de beneficis, els quals es poden aconseguir sense necessitat de canvis en els procediments o en les responsabilitats de separació, existiran altres beneficis operacionals del CDTI que si què imposaran aquest tipus de canvis i què, per tant, hauran de ser ben determinats per tal de poder justificar les inversions necessàries.

## 1.2. Conjectures

Els requisits específics del CDTI dependran de les aplicacions derivades que se'n vulguin fer.

A la mercè de l'ús que li donem, el CDTI pot ser configurat per ostentar diferents tipus d'informació de tràfic.

Les fonts múltiples de dades de vigilància seran emprats per proveir al pilot amb la millor informació disponible.

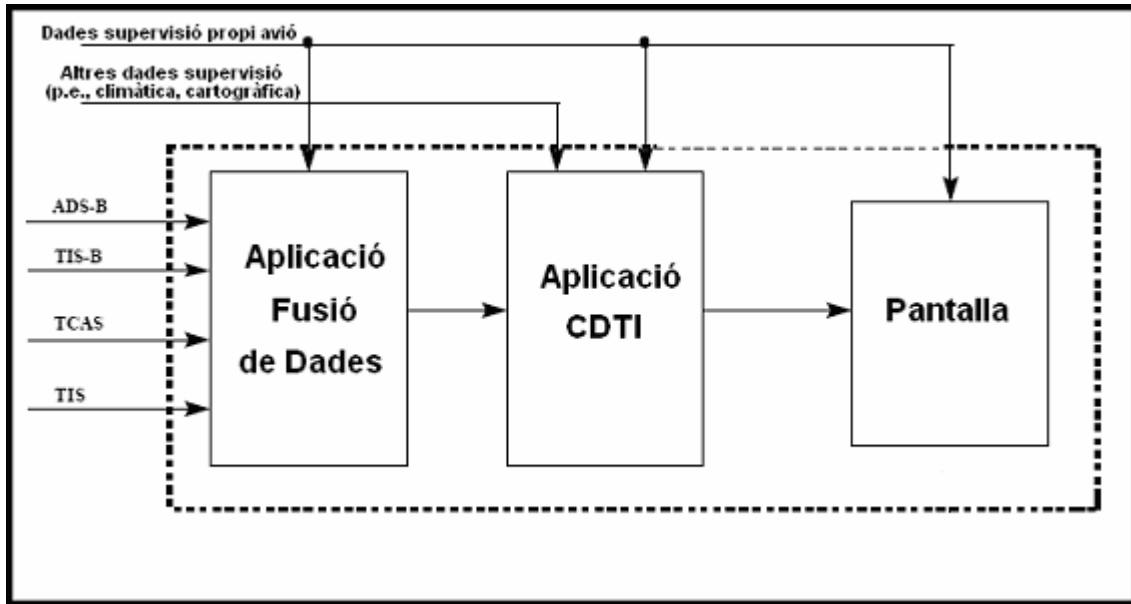
El disseny de l'equip CDTI està pensat per a la seva expansió i millora del programari associat. El CDTI haurà d'estar correctament integrat amb altres dispositius de pantalla i no haurà de interferir amb altres dispositius de més alta prioritat.

## 1.3. Aplicacions de vigilància del CDTI

La Figura 1-2 il·lustra la relació entre les funcions del CDTI, la funció de fusió de dades, fonts de vigilància (p.e., TCAS, TIS, ADS-B) i la pantalla. El CDTI obté fluxes de dades provinents de múltiples sistemes de vigilància. Aquests poden incloure: informació sobre les posicions relatives entre dos avions del TCAS; paquets transmesos avió a avió, dades TIS enviades via radars Mode-S, dades TIS-B provinents d'altres radars terrestres mitjançant un enllaç broadcast, i les dades obtingudes pels sensors propis de l'avió. Cada un d'aquests fluxes de dades tenen diferent contingut i qualitat de recepció i poden provenir de diferents equips o sistemes de l'avió. Les funcions del CDTI realitzen càlculs per transformar els fluxes de dades de vigilància en informació i prestacions per mostrar finalment per pantalla.

El dispositiu de pantalla és la interfície entre el processament de dades i el pilot. El dispositiu pot mostrar aquesta informació de forma auditiva, textual o gràfica.

Els elements requerits pel dispositiu i la qualitat depenen de l'ús que es vulgui donar a les dades.



**Fig. 1.2** Relació funcional del CDTI, aplicacions i fonts de vigilància

## CAPÍTOL 2. FONTS DE DADES

L'aplicació de fusió de dades que proporciona dades al CDTI pot rebre fluxes de dades de múltiples fonts.

Les fonts d'informació de vigilància de l'avió són obtinguts dels sistemes següents:

- ADS-B
- TIS
- TIS-B
- TCAS I i II

L'aplicació de fusió de dades pot incloure els següents càlculs: (1) la extrapolació temporal i transformació dels fluxes individuals de dades en coordenades en temps real per al CDTI, (2) la correlació dels diferents fluxes de dades per tal de poder mostrar un únic encapsulat per cada avió, i (3) la selecció o combinació de les dades més rellevants provinents dels diferents fluxes. L'aplicació CDTI pot incloure càlculs numèrics com ara els càlculs de proximitat que poden ser mostrats en diferents aplicacions. Alguns d'aquests càlculs han de tenir sub-processos específics. Per exemple, si algun dels diferents fluxes de dades no està disponible, els càlculs en el cas del (2) i del (3) no es poden realitzar i seran omesos.

Per facilitar una visualització compartida del tràfic, la tripulació hauria de tenir la millor informació disponible. Això pot fer-se utilitzant la font disponible més fiable (p.e., millor precisió i integritat) o utilitzant una combinació apropiada de les dades de múltiples fonts.

### 2.1. ADS-B

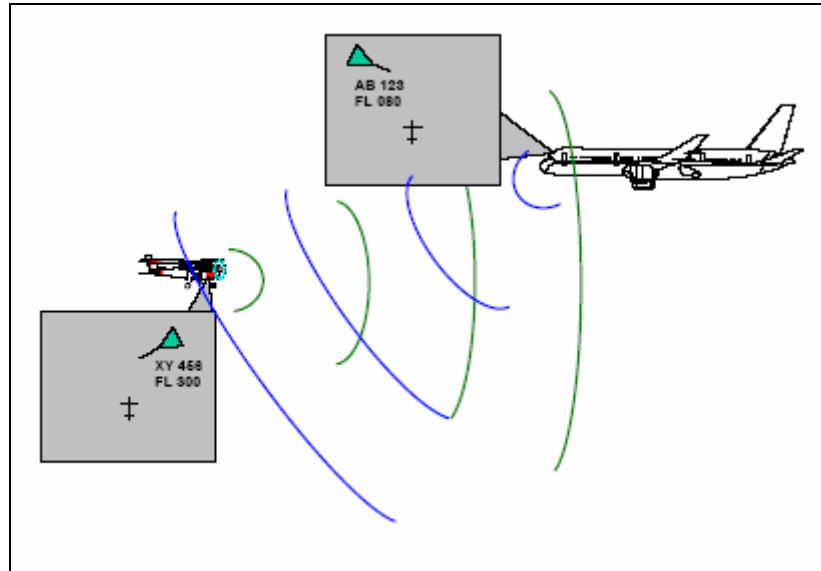
El sistema de vigilància ADS-B (*Automatic Dependent Surveillance Broadcast*) és una aplicació a nivell d'enllaç de dades que transmet des d'avions o vehicles terrestres dades de vigilància (com ara posició i identificació) per mode broadcast als usuaris que puguin rebre aquestes dades.

ADS-B pot suportar aplicacions de vigilància basats en radars terrestres, però també pot proveir de cobertura de vigilància en zones sense radars (p.e., oceans i zones remotes) i la capacitat de suportar eines de gestió automàtica del tràfic aeri com ara avís en situacions de conflicte, predicció de trajectories i monitorització de trajectòries de vol.

La informació es transmesa pel transponder a 1090 MHz. Com què ADS-B utilitza un protocol broadcast, qualsevol receptor que treballi a 1090 MHz podrà

rebre i decodificar les senyals per tal de determinar la posició de l'avió emissor de la senyal.

A continuació es presenta la Figura 2.1 que ens mostra les comunicacions intercanviades pels parells d'avions en una comunicació ADS-B, així com la visualització obtinguda per cada un d'ells.



**Fig. 2.1** Diagrama funcionament ADS-B

A nivell maquinari, hi ha tres sistemes emprats per a la comunicació entre parells d'avions amb el sistema de vigilància ADS-B. Aquests són el VDL Mode 4, UAT i per mitjà de Transponder Mode S. En el capítol 4 s'explicarà les característiques principals d'aquests sistemes.

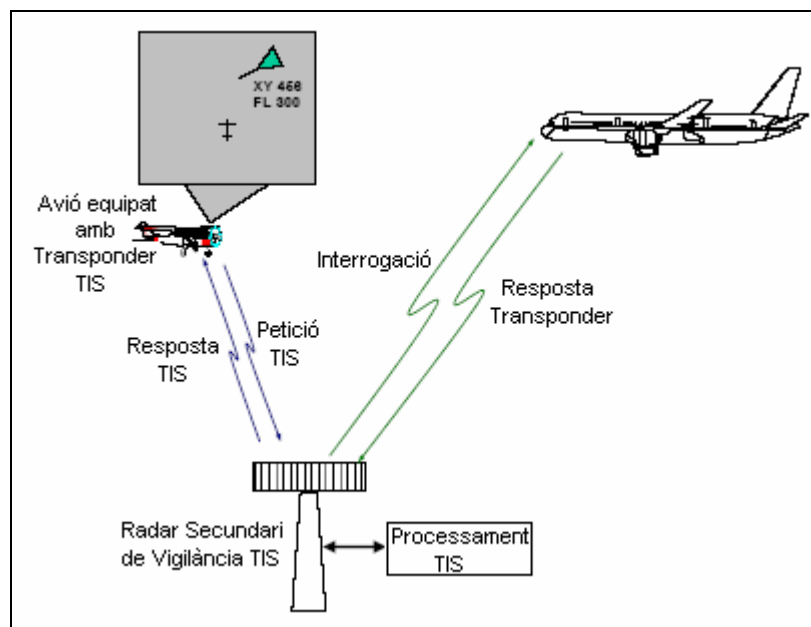
## 2.2. TIS

El sistema de vigilància TIS (*Traffic Information Service*) és un servei d'enllaç de dades Mode S a través del qual la informació de vigilància dels radars terrestres s'envia als avions per tal que puguin visualitzar-la a la cabina. La funció d'enllaç de dades proveeix visualització automàtica al pilot del tràfic pròxim i de les advertències o amenaces per a l'avió. Els transponders que estiguin dins d'una cobertura de 5 mn i 1200 peus d'altitud són incorporats a l'aplicació. TIS genera alertes per cada avió amb cobertura Mode S que porti un transmissor ATCRBS o Mode S. TIS proporciona la distància relativa, altitud, i localització dels avions més pròxims que poden ser mostrats al pilot. També proporciona informació de tendència vertical (velocitat de pujada o baixada) i advertències del tràfic similars a les del TCAS I.

El servei TIS està disponible pels avions equipats amb Mode S a la zona de cobertura de vigilància. El programari TIS dins els sensors Mode S proporciona a l'usuari informació sobre els avions equipats amb altres transponders propers.

El funcionament del sistema TIS consisteix en un intercanvi de informació entre el transponder d'un avió i els radars terrestres. La petició del propi avió al radar terrestre s'efectua a 1090 MHz. Aleshores, el radar contesta amb un paquet TIS a 1030 MHz amb la informació dels diferents avions de la zona més pròxima (rangs de cobertura descrits en aquest apartat). El funcionament d'aquest sistema es mostra a la Figura 2.2.

***Nota:*** Els Transponders Mode S responen a les crides ATC donant-los un codi d'identificació ATCRBS, codificant altitud i d'altres dades de camp.



**Fig. 2.2** Esquema funcionament sistema TIS

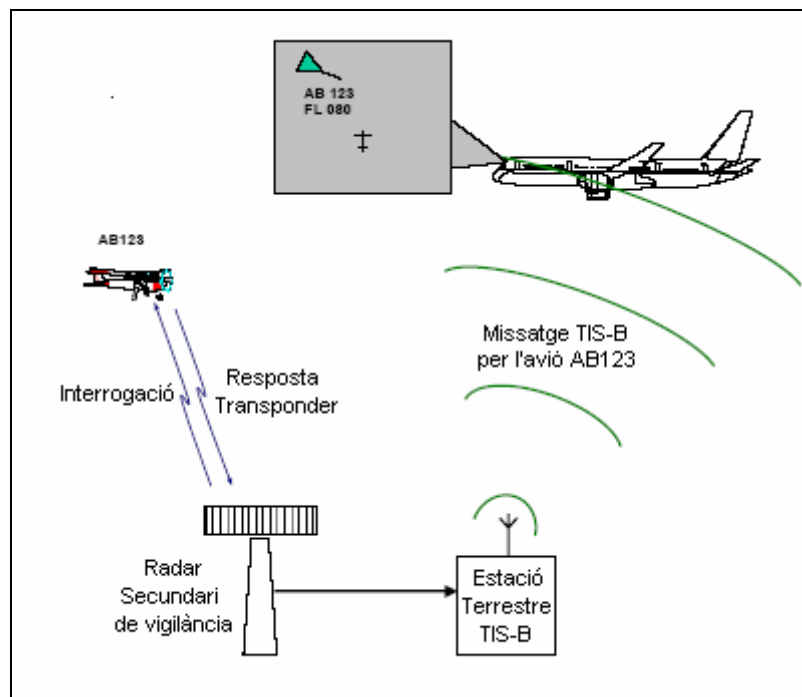
Aquest sistema està en desús ja que es necessita una sèrie d'estacions radar terrestres escampades pel territori i és inviable en zones remotes i oceans.

### 2.3. TIS-B

El sistema de vigilància TIS-B (Traffic Information Service Broadcast) és un servei en el qual les posicions de tots els avions dins una ampla àrea de cobertura són transmesos via broadcast a tots els usuaris amb la mateixa informació per a tots. Cada usuari, tenint coneixença de la seva pròpia posició, pot calcular la posició relativa del tràfic. Com el TIS, TIS-B permetrà als avions equipats amb ADS-B rebre informació limitada dels avions que no el portin

equipat quan estiguin dins la cobertura d'un radar de vigilància. Durant la transició a ADS-B, TIS-B proporcionarà informació del tràfic derivada d'un radar secundari de vigilància (SSR) dels avions que no estiguin equipats amb el sistema ADS-B. Com que el nombre d'avions equipats amb ADS-B augmenta, la necessitat d'incorporar el sistema TIS o TIS-B es reduirà.

Per aquest sistema també es necessari l'utilització de radars terrestres ja que aquests interroguen (1030 MHz) als avions situats en la seva zona de cobertura obtenint la resposta del transponder (1090 MHz) amb les seves dades. Aquest radar terrestre, emet via broadcast tota la informació referida al tràfic aeri circumdant. A continuació, la Figura 2.3 ens mostra la representació del funcionament del sistema TIS-B descrit.



**Fig. 2.3** Esquema funcionament sistema TIS-B

## 2.4. TCAS

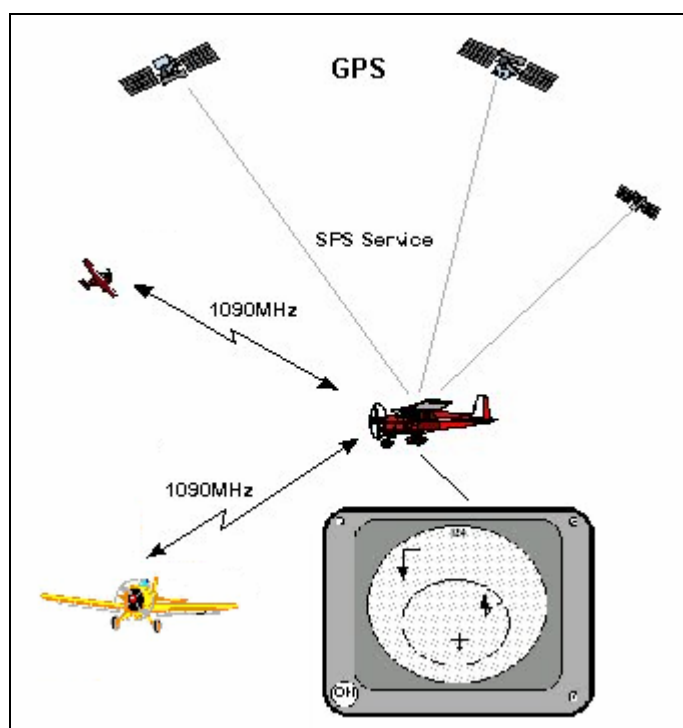
TCAS és un sistema d'evitació de colisions que utilitza un transmissor/receptor per tal de comunicar-se amb els transponders dels avions propers. TCAS determina l'àmbit de l'objectiu molt acuradament amb el temps que es tarda a contestar. TCAS gairebé determina la posició relativa del objectiu amb un sistema d'antena direccional. TCAS utilitza les mesures d'altitud rebudes de les replicues del transponder per determinar l'altitud de l'objectiu.

Hi ha dos sistemes bàsics amb capacitats diferents. El sistema TCAS I va ser dissenyat com a sistema de baix cost per a ús pels avions del *General Aviation*. Proporciona una visualització del tràfic i assisteix al pilot informant de les



situacions de perill o conflicte del tràfic aeri pròxim. El rang mínim de cobertura del sistema TCAS I és de 5 mn. Alguns sistemes tenen una cobertura molt més gran. Tots els sistemes estan subjectes a limitacions per interferències, el que pot reduir la seva cobertura.

El sistema TCAS II va ser dissenyat per a prevenir colisions. TCAS II té una cobertura de 20 mn, i una millor precisió de la posició relativa respecte TCAS I. TCAS II també dona la visualització del tràfic. TCAS II no només avisa al pilot de l'objectiu d'una alerta per colisió, sinó que també calcula la maniobra d'evasió vertical per prevenir la colisió, la coordina amb l'objectiu del TCAS II (si està equipat), i comunica consells de resolució al pilot. S'ha de dir que l'aplicació d'aquest sistema pot tenir conseqüències nefastes en el cas que un dels dos pilots no fagi cas dels consells del sistema TCAS II per evadir una colisió i acabin dirigint-se cap a la mateixa posició.



**Fig. 2.4** Esquema funcionament sistema TCAS

## 2.5. Diferències entre les fonts de vigilància

Cada una de les fonts de vigilància presentades tenen característiques operatives diferents. Aquestes diferències estan resumides a la Figura 2.5, abaix. A més, les següents diferències també són dignes d'esmena:

- Els errors de posició en el TCAS són diferents dels errors del ADS-B, en particular en la posició relativa. Els errors en el TIS són també diferents als errors del ADS-B. Els errors del TIS poden notar-se especialment en

les medicions en azimuth, especialment dins del rang de les 2 mn en relació a l'aeronau del receptor. La precisió en la posició relativa dels avions colindants en el cas del TCAS és molt acurada.

- El rang per a la detecció i visualització dels objectius ADS-B pot ser més gran que per els objectius TCAS. En els rangs on els objectius ADS-B són detectats però no així els TCAS (p.e., fora del rang normal de cobertura del TCAS), només els objectius amb ADS-B equipat es mostraran (p.e., si la relació d'usuaris amb el sistema equipat és baix, hi ha una menor certitud que tots els objectius siguin visualitzats). TIS proveeix la informació del tràfic pròxim en una àrea inferior a 5 mn de l'avió en qüestió. Depenent del rang màxim de visualització, el pilot pot seleccionar un rang major que el rang TIS nominal.
- TIS només està disponible quan el client i l'avió objectiu estan dins la cobertura del sensor Mode S del equip TIS. La cobertura del sensors Mode S pot ser poc fiable a baixa altura (p.e., altituds dels circuits de tràfic en aeroports), on la cobertura ADS-B es manté estable.
- ADS-B només pot detectar objectius que responguin a les crides. Encara que una gran proporció dels avions van equipats amb el sistema ADS-B, molts objectius continuaran sense detectar altres ADS-B per si mateixes.
- ADS-B pot proporcionar detecció a baixes altituds i a la superfície de l'aeroport, on la funcionalitat del sistema TCAS resta disminuïda o deshabilitada.
- TCAS només pot detectar avions que responen a les crides (equipats amb *transponders*). TCAS pot detectar la majoria dels transponders equipats a l'avió.

**Taula 2.1** Comparativa de les diferents fonts d'informació de vigilància

Paràmetre	ADS-B	TCAS	TIS	TIS-B
Posició de l'objectiu	· Latitud · Longitud · Altitud	· Distància relativa · Altitud · Angle	· Distància relativa · Altitud relativa · Angle	· Latitud · Longitud · Altitud
Rang Nominal (mn)	10-120 (1)	14-30	5 (3)	Basat en l'àrea de cobertura TIS
Període de mostreig (Hz)	3, 1.5 per aproximacions simultànies (2)	1 – 5	4 -- 12 (4)	4 -- 12 (4)
Equip requerit per tarjet aircraft	Transmissor ADS-B	Transponder Mode A/C o Mode S	Transponder Mode A/C o Mode S (5)	Transponder Mode A/C o Mode S (5)
Informació adicional de l'objectiu	· Identificació · Categoria d'avió · Estat emergència · Codificació (6) · Velocitat · Aceleració · Posició i velocitat Aproximada	· Identificació interna de tarjeta  TCAS (7) · Adreça de la Trajeta Mode S · Tasa vertical · Avís de tràfic o a avís emergència	· Indicació Sentit (Ascendent o descendent) · Advertència estat en el tràfic	· Indicació Sentit (Ascendent o descendent) · Advertència Estat en el tràfic

### Notes:

1. *El rang mínim de que està capacitat el sistema ADS-B és de 10 mn; el rang previst per la majoria de sistemes ADS-B està en 50-100 mn.*
2. *La taxa de transmissió d'un sistema ADS-B ha de ser més gran per tenir la certesa que la recepció està dins la taxa nominal.*
3. *TIS proporciona informació del tràfic i de les alertes per dispositius dins els 5 mn i els 1200 peus. Els missatges TIS poden arribar a cobertures de 7 mn.*
4. *El període de mostreig del sistema TIS depèn de la taxa de mostreig dels sensors Mode S. El període de mostreig del sistema TIS-B també depèn de la taxa de mostreig dels radars de vigilància que donen la informació de posició. Es pot incrementar la freqüència de mostreig utilitzant una combinació de radars com a entrada.*
5. *Els sistemes TIS/TIS-B poden detectar dispositius radars primaris.*
6. *ADS-B també proporciona informació sobre la capacitat dels equips per tal de suportar categories específiques d'aplicacions.*
7. *TCAS rastreja els objectius individuals i conserva la ID interna, anomenada ID de vigilància. Aquesta ID pot ser utilitzada per la funció de fusió de dades per correlacionar els paquets dels diferents dispositius.*

## **2.6. Fusió de dades**

Els avions equipats amb CDTI poden rebre dades del mateix objectiu de diferents fonts de vigilància. El procés de determinar quins paquets de dades es refereixen al mateix avió s'anomena "correlació". El CDTI hauria de mostrar un únic símbol encara que li arribi més d'un paquet de dades de vigilància d'un mateix avió. La correlació de la posició de l'objectiu ha d'emprar un algorisme que minimitzi el nombre d'objectius de cada avió visualitzats. L'algorisme haurà de minimitzar la correlació incorrecta de paquets dels diferents avions (p.e., declarant-los el mateix avió). Quan un únic identificador està disponible de cada font, com ara una adreça de dispositiu, el procés de correlació de l'objectiu identifica els paquets sense ambigüitat. Per altra banda, la proximitat en la posició obtinguda, i possiblement altres dades, són utilitzades per correlacionar paquets. Encara que hi hagi més d'un paquet per avió, el pilot no ha de ser informat de l'origen de la informació.

El CDTI ha de mostrar la millor estimació de la posició d'un objectiu. Aquesta estimació utilitza les dades de la font de vigilància més precisa, o pot combinar dades de varies fonts, tenint en compte la seva precisió, si es sabuda. Les dades de diferents fonts poden ser combinades si s'ha fet un apropiada previsió dels errors relatius de cada una d'elles.

El sistema pot mostrar una indicació de la disponibilitat i/o fallida de les fonts de dades.

La taxa de mostreig dels objectius ADS-B i TCAS pot ser major que la de les objectius TIS. És correcte actualitzar les dades ADS-B i TCAS quan els paquets arriben, sense esperar a que la informació del TIS hagi arribat.

Si un paquet es utilitza, una cua emmagatzema l'últim paquet per a cada objectiu. Aquest paquet ha de ser diferent al utilitzat pels algorismes de TCAS.

La informació del CDTI no pot interferir amb la lògica d'evitació de colisions.

## **2.7. Altres consideracions en el processament de dades de vigilància**

A més de l'esmentat a la fusió de dades, el període d'adquisició màxim hauria de ser considerat. La posició de l'objectiu pot ser extrapolada, si la precisió de la informació de velocitat està disponible i l'objectiu no ha estat actualitzat recentment. En aquest cas, fóra bo proporcionar un indicador al pilot que li indiqui que la posició mostrada no és l'última obtinguda. Alternativament, si la informació de l'objectiu no ha estat actualitzat recentment, la última posició rebuda pot continuar sent mostrada, o estimada.

Els objectius haurien de rastrejar vertical i horitzontalment per tal de suportar la generació d'alertes de tràfic i les tases amb indicacions verticals. El rastreig pot ser innecessari si la informació referida a velocitat és acurada.

## CAPÍTOL 3. DISSENY CDTI

L'aplicació de pantalla cartogràfica de baix cost està formada per tres parts programari i tres maquinari.

En aquest capítol descriurem dues de les tres parts: disseny i programació de la llibreria i el disseny de l'aplicació visual. L'aplicació d'adquisició i fusió de dades no s'implementarà en aquest projecte degut a la seva complexitat i dependència al programari triat per a la seva implementació.

El maquinari serà descrit en el següent capítol.

### 3.1. Llibreria

La llibreria està programada en C++ sobre *Microsoft Visual C++ 6.0*. Aquesta llibreria està formada per cinc classes i una sub-libreria on hi ha programades fórmules per fer càlculs matemàtics.

Aquesta llibreria inclou un generador de coordenades que genera en temps real les coordenades d'una sèrie limitada d'avions i els emmagatzema en una de les classes que a continuació es descriuran.

Nota: Per a més informació sobre el codi, vejeu Annex B.

#### 3.1.1. Classe *clibrary*

La classe *clibrary* consta de dos constructors i un destructor, vuit atributs i tres mètodes.

##### 3.1.1.1. Constructors; Destructor

- Constructors: n'hi ha dos. El primer inicialitza tots els atributs a zero i el segon se li passa per paràmetre un *string* on hi ha l'adreça completa de l'arxiu d'on volem importar l'informació.
- Destructor: si el vector *plane\_* no està buit s'elimina.

##### 3.1.1.2. Atributs

- *vector<cplane> plane\_* : vector dinàmic que conté les estructures dels diferents avions del sistema
- *double time\_* : utilitzat per emmagatzemar el temps del sistema. Unitat: segons

- *double collision\_radius\_* : s'estableix en el mètode *void asas(string file)* i conté el radi en què es considera zona de conflicte.
- *double collision\_height\_* : s'estableix en el mètode *void asas(string file)* i conté l'altura de seguretat en què es considera zona de conflicte
- *double alert\_radius\_* : variable que emmagatzema el radi que marca la zona d'alerta. Establert en el mètode *void asas(string file)*.
- *double alert\_height\_* : variable que emmagatzema l'altura que determina la zona d'alerta. Establert en el mètode *void asas(string file)*.
- *int plane\_selected\_* : variable emprada per saber quin avió de l'estructura és el centre de referència. Variable utilitzada al mètode *void asas (string file)* i que depèn de l'alpicació gràfica.
- *double period\_* : indica el període de mostreig de coordenades.

### 3.1.1.3. Mètodes

- *int load\_planes (string file)* : obre l'arxiu file per tal de llegir el nombre d'avions que contindrà el vector *plane\_* i les ubicacions de l'arxiu de *waypoints* de cada un d'ells.
- *void load\_alert (string file)* : mètode que importa de l'arxiu file la configuració del sistema per tal de poder començar a visualitzar. Configuració del sistema: *collision\_radius\_*, *collision\_height\_*, *alert\_radius\_*, *alert\_height\_* i *period\_*.
- *int alert (int pos)* : fil d'execució que calcula la distància relativa entre l'avió seleccionat com a centre i la resta i estableix les situacions conflictives o d'alerta entre ells.

## 3.1.2. Classe *cplane*

La classe *cplane* consta d'un constructor i destructor, cinc atributs, tres mètodes i un destructor.

### 3.1.2.1. Constructor; Destructor

- Constructor: inicialitza tots els atributs a zero.
- Destructor: destrueix el string *name\_* i el vector *coor\_*.

### 3.1.2.2. Atributs

- *string name\_* : nom de l'avió
- *int state\_* : informa de l'estat actual en què es troba l'avió; no carregat {-1}, parat {0}, normal {1}, en conflicte {2}, perill {3}.
- *cflightpath flightpath\_*

- *double bearing\_[2]* : conté el vector de trajectòria actual de l'avió. Coordenada x i coordenada y ocupen la posició 0 i 1 respectivament. Unitat: Segons.
- *vector<ccordinate> coor\_* : guarda les últimes *MAX* coordenades obtingudes de l'avió.

### 3.1.2.3. Mètodes

- *void add\_info\_flightpath (string file)* : Funció que carrega des de l'arxiu *file* els *waypoints* de l'avió juntament amb l'identificador *name\_*.
- *ccordinate calc\_coor (double time)* : Calcula la coordenada de l'avió en un instant determinat (només es passarà per paràmtre un valor de *time* inferior o igual al valor *time* de l'últim *waypoint*).
- *void calc\_bearing (void)* : Calcula el vector de direcció *bearing\_* de l'avió utilitzant les coordenades emmagatzemades al vector *coor\_*.

### 3.1.3. Classe *cflightpath*

Aquesta classe s'ha ideat per tal de contenir els *waypoints* de l'avió. Consta del constructor per defecte, destructor, un atribut i un mètode.

#### 3.1.3.1. Destructor

Si el vector *waypoint\_* no està buit s'elimina.

#### 3.1.3.2. Atribut

- *vector<ccordinate> waypoint\_* : conté els *waypoints* de l'avió

#### 3.1.3.3. Mètode

- *void calc\_waypoint\_time(int pos)* : calcula el temps en què l'avió arribarà al *waypoint* de la posició *pos* i l'emmagatzema.

### 3.1.4. Classe *ccordinate*

Aquesta classe té dos constructors i sis atributs.

#### 3.1.4.1. Constructors

Hi ha dos constructors: un inicialitza tots els atributs a zero i l'altre amb les variables passades com a paràmetres.

### 3.1.4.2. Atributs

- *double latitude\_* : conté la latitud en radiants
- *double altitude\_* : conté l'altitud en radiants
- *double height\_* : conté l'altura en metres
- *double time\_* : conté el temps en segons
- *double altitude\_* : conté l'altitud baromètrica en metres
- *double std\_* : conté la velocitat azimut de l'avió

### 3.1.5. Fórmules

Les fórmules presentades a continuació principalment són de conversió entre Sistema Internacional i unitats emprades al món de l'aviació però d'altres necessàries per a realitzar càlculs.

#### 3.1.5.1. Distància entre dos punts

La funció *double distance\_2points(double lat1, double lon1, double lat2, double lon2)* calcula la distància entre dos punts. Se li passa per paràmetre la longitud i latitud de dos coordenades. Et retorna el mòdul de la distància entre els dos punts. S'ha d'afegir que només es té en compte els plans X i Y per tal de fer el càlcul.

La unitat de les latituds, les longituds i la distància entre els dos punts és el radiant.

$$d = 2 \times \arcsin \left( \sqrt{\sin^2 \left( \frac{lat1 - lat2}{2} \right) + \cos(lat1) \times \cos(lat2) \times \sin^2 \left( \frac{lon1 - lon2}{2} \right)} \right) \quad (3.1)$$

#### 3.1.5.2. Equivalència entre milles nàutiques i radiants

$$\text{distància [mn]} = \left( \frac{180 \times 60}{\Pi} \right) \times \text{distància [radiants]} \quad (3.2)$$



### 3.1.5.3. Equivalència entre milles nàutiques i metres

$$\text{distància}[\text{mn}] = 1852 \times \text{distància}[\text{metres}] \quad (3.3)$$

### 3.1.5.4. Equivalència entre metres per segon i nusos(kts)

$$\text{velocitat}[\text{nusos}] = \left( \frac{3600}{1852} \right) \times \text{velocitat}[\text{m/s}] \quad (3.4)$$

### 3.1.5.5. Equivalència entre peus i metres

$$\text{distància}[\text{metres}] = 3,28024 \times \text{distància}[\text{peus}] \quad (3.5)$$

## 3.2. Aplicació visual

L'aplicació visual té per funció mostrar per pantalla la situació dels diferents avions en una zona determinada.

A continuació es descriurà els criteris seguits pel que fa al disseny gràfic de l'aplicació CDTI així com les diferents opcions de visualització.

### 3.2.1. Funcions característiques

El CDTI suporta els següents usos operacionals:

- Percepció de la situació del tràfic millorat
- Adquisició visual del tràfic millorat
- Operacions d'apropament visual millorades
- Suport en procediments de separació en oceans o zones remotes

### 3.2.2. Característiques generals de visualització

En un CDTI es pot visualitzar multitud de dades i, a vegades, aquest fet pot contribuir a dificultar la interpretació de les dades o, fins i tot, pot provocar una incorrecta interpretació. Per això destacarem l'informació imprescindible a visualitzar i, en un segon terme, s'enumerarà d'altres dades que podrien ser incorporades.

### 3.2.2.1. Característiques mínimes de visualització i informació de l'objectiu

- Símbol propi avió: La visualització del tràfic ha de contenir un símbol representant la localització del propi avió.
- Símbol avió objectiu: La pantalla ha de representar el tràfic proper al propi avió. Aquesta representació consisteix en el símbol de l'avió orientat en la direcció de l'avió.
- Altitud de l'avió objectiu: El sistema representa les altituds relatives dels altres avions (*Flightlevel*).
- Posició relativa de l'avió objectiu
- Brújula: S'hauria de visualitzar al voltant de la pantalla els quatre punts cardinals i, si es desitja, la graduació intermitja.

### 3.2.2.2. Altres característiques de visualització de l'aplicació

- Identificador (*Call Sign*): fins a set caràcters alfanumèrics de longitud obtinguts dels *tarjets* dels avions.
- Velocitat respecte el terre (*Ground Speed*): Velocitat representada com a mínim amb tres caràcters alfanumèrics.
- Indicacions *Ground Track*: Es la direcció que segueix el propi avió. Poden ser mostrades gràficament com per exemple mitjançant un segment des del símbol de l'avió. Si s'utilitza un segment, aquest podria ser proporcional a la velocitat respecte el terre, és a dir, representar el vector velocitat.
- Selecció de tarjet: Indica de quina font de vigilància s'ha obtingut les dades de l'avió mostrat.
- Cobertura extesa de pantalla: Serà necessària per algunes aplicacions. Per exemple, en zones oceàniques/remotes serà necessària una cobertura extesa de pantalla de com a mínim 60 mn i màxima de 120 mn.
- Indicació de direcció vertical: Indica si l'objectiu està en ascendint, descendint o a altura constant.
- Punts de pas del propi avió (*waypoints*).
- Alertes de tràfic: En preveure un conflicte potencial el símbol del propi avió podia variar de color i, si el maquinari ho permet, produir-se una alerta acústica. A continuació es mostra la taula de colors segons l'estat de l'avió.

**Taula 3.1.** Relació Colors-Conflicte

	Propi avió	Altre
Normal	Blau	Gris clar
Precaució	Blau	Gris Fosc
Conflicte potencial	Groc	Groc
Conflicte Inminent	Vermell	Vermell

**Nota:** Els conflictes dels altres avions només es mostraran en cas que el conflicte sigui amb el propi avió. Per a més informació, veure sub-apartat 3.2.2.3.

- Zones de protecció: Es podria visualitzar la zona de separació i zona de col·lisió del propi avió.

### 3.2.2.3. Alertes de tràfic

El disseny del CDTI està pensat per alertar al pilot de l'existència d'un conflicte potencial amb un altre avió. Aquesta aplicació les mostrarà de forma visual.

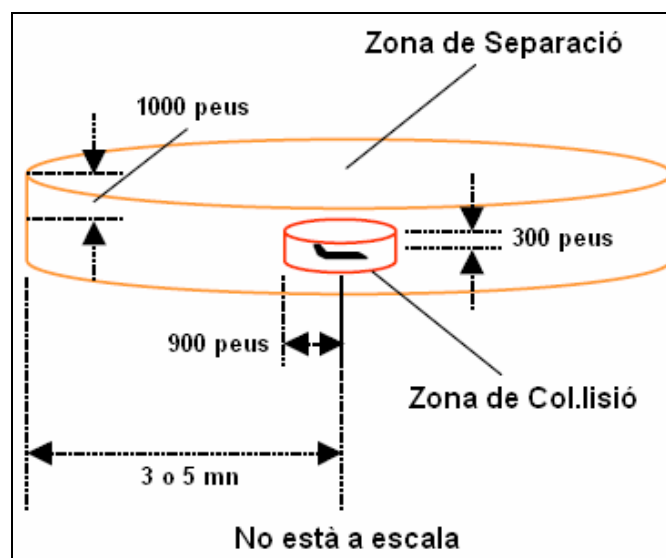
Les alertes de tràfic estan basades en dos zones al voltant del propi avió, anomenades zona de separació i zona de col·lisió.

Aquestes zones estan delimitades basats en les recomenacions del comitè *RTCA's Airborne Conflict Management (ACM)*.

La zona de separació representa la mínima separació legal al voltant d'un avió. Té 3 o 5 mn de radi, depenen de la localització de l'avió. Respecte a la vertical, l'altura de separació es de 1000 peus pels dos casos.

El conflicte està definit com la pèrdua de separació i una violació de separació ocorreguda si un avió penetra la zona de separació d'un altre avió. La radi de la zona de col·lisió és de 900 peus i l'altura és de 300 peus.

Aquestes distàncies són les que estan predefinides en l'aplicació encara que poden ser modificades mitjançant la funció *void load\_alert (string file)* cridada a través de l'aplicació visual.



**Fig. 3.1** Zones de Separació i Col·lisió

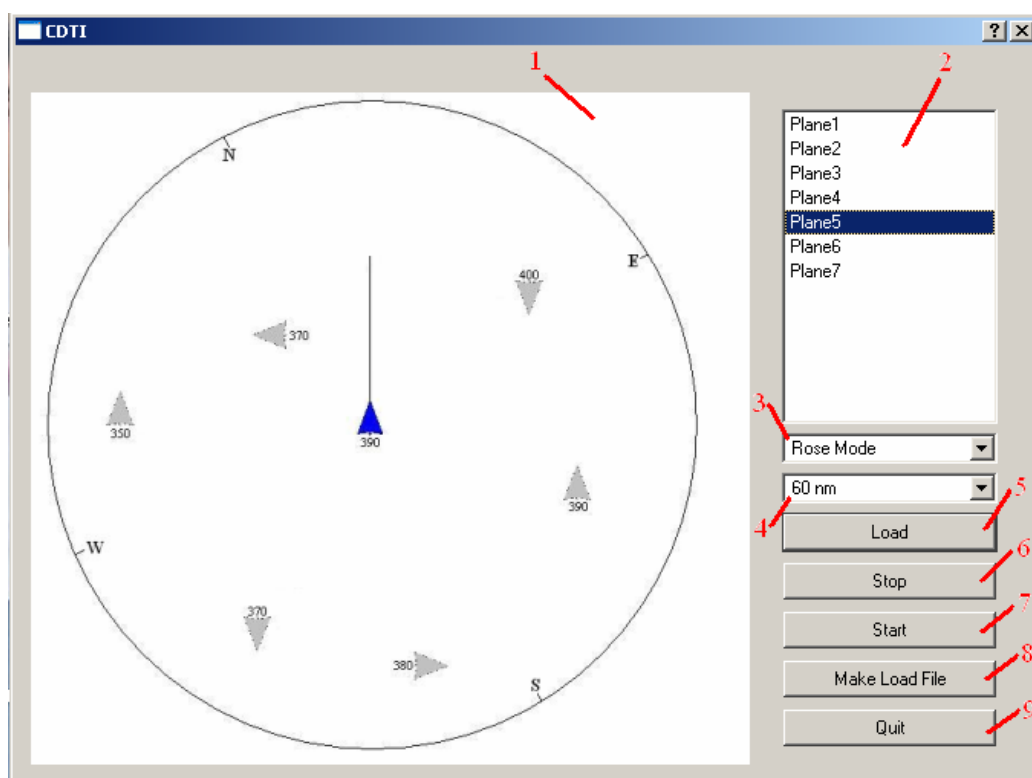
La definicions següents corresponen als criteris seguits per tal de definir els diferents estats en que es pot trobar un avió.

- Normal: No hi ha cap avió proper ni en la mateixa direcció que el propi.
- Precaució: L'avió objectiu pot creuar-se a llarg termini amb el propi ja que la direcció seguida per un condueix a l'altre. Per exemple, el propi segueix direcció Nord-Oest i l'objectiu Sud-Est.
- Conflicte Potencial: Aquesta conflicte es produeix quan un avió infringeix la zona de separació d'un altre. En aquest estat s'ha de intervenir modificant el pla de vol del propi avió i/o de l'objectiu.
- Conflicte Inminent: Aquest conflicte es produeix quan s'ha estat en estat conflicte potencial i no s'ha modificat el pla de vol i com a conseqüència s'ha entrat en zona de colisió. Aquest estat no s'ha d'arribar en cap cas ja que el temps de reacció es mínim.

*Nota:* Aquestes definicions poden ser modificades per tal d'ampliar les restriccions de separació obtenint un major temps de reacció.

### 3.2.3. Opcions de visualització

L'aplicació visual del CDTI té una sèrie d'opcions per tal d'obtenir una correcta adquisició visual del tràfic circumdant.



**Fig. 3.2** Exemple de visualització del CDTI

A continuació s'explicarà aquestes opcions de visualització així com la resta d'elements que formen l'aplicació visual.

- (1) Visor del tràfic aeri
- (2) Elecció del propi avió: Es pot variar el centre del visor i escollir l'avió que apareixi en pantalla. El nom que apareix és el Identificador o *Call Sign*.
- (3) Mode de visió: Es pot escollir entre el *Rose Mode* o el *Mode Arc Mode*. El primer es dona una visió del 360 graus i en el segon de 90. A la Figura 3.2 es presenta el *Mode Rose*.
- (4) Cobertura: La cobertura és el radi de visió de l'aplicació i la emprada per tal de realitzar els càlculs. Els radis de cobertura escollits són: 30, 60, 90, 120, 150 mn.
- (5) *Load*: Aquest botó s'utilitza per carregar la bateria d'avions que es mostren per pantalla.
- (6) *Start*: Aquest botó inicia o reinicia el simulador. Prèviament s'ha d'haver premut el botó *Load*.
- (7) *Stop*: Aquest s'utilitza per parar l'aplicació. Per reiniciar-la s'ha de polsar el botó *Start*.
- (8) *Make Load File*: Prement aquest botó s'obre un formulari per tal de realitzar l'arxiu que posteriorment es carregarà mitjançant el botó *Load*.
- (9) *Quit*: Sortida del programa.

## CAPÍTOL 4. MAQUINARI

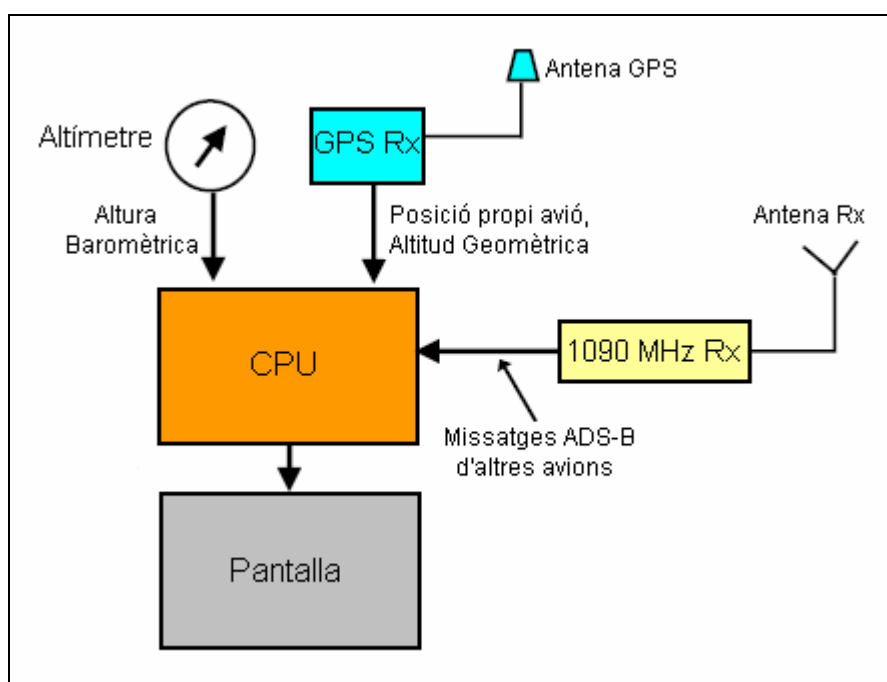
En aquest capítol s'exposa les necessitats maquinari del CDTI així com alguns productes que s'hi adequarien.

El sistema constarà de tres parts: la recepció/emissió, el processament i la visualització.

### 4.1. Introducció

El sistema es basarà únicament en les dades obtingudes del sistema de vigilància ADS-B ja que és el sistema més extès a l'aviació civil encara que en una futura millora del sistema s'hi podria incorporar algun dels altres sistemes descrits al capítol 2 (TIS, TIS-B, TCAS I, II) o algun altre que pugui implementar-se en un futur.

A continuació tenim la Figura 4.1 que ens mostra els principals components maquinari que trobem al sistema.



**Fig. 4.1** Diagrama Funcional CDTI

En la resta del capítol s'exposaran les diferents consideracions a tenir en compte per tal de posar en funcionament el sistema CDTI de baix cost així com diverses solucions comercials del sistema d'adquisició ADS-B, pantalla i CPU.

## 4.2. Emissió/Recepció

### 4.2.1. Elecció del mode de comunicació

Els equips que han de portar instal·lats els avions a bord per poder utilitzar l'ADS-B han de tenir certes característiques necessàries pel bon funcionament de l'aplicació. La generació de missatges i l'emissió d'aquests, i la interacció d'aquesta nova tecnologia amb els sistemes de navegació són importants pel seu funcionament.

Com s'ha mostrat a la Figura 1.1 d'aquesta memòria, hi ha tres modes de comunicació possibles de funcionament de l'ADS: Mode S ES 1090 MHz, UAT i VDL Mode 4. Tan als Estats Units com a Europa s'ha escollit el Mode S ES com a mitjà principal per l'ADS-B, ja que el mode S és l'utilitzat als transponders dels avions actuals degut a la seva utilització amb el TCAS. Degut a això, aquest mode serà l'utilitzat en aquest disseny.

Per permetre que un avió pugui enviar missatges amb squitter extés, el transponder s'ha de modificar i s'ha d'enviar la posició de l'avió i d'altres informacions de l'estat de l'aeronau al transponder. Tan a l'estació de terra de l'ATC com a l'equipament del TCAS en vol poden rebre aquestes informacions amb els receptors de 1090 MHz, i només s'hauria de millorar l'acceptació i el processament de la informació addicional. Per aquests canvis es requereix un programari de més nivell que l'actual al transponder i connectar-lo mitjançant cables amb el sistema de navegació, per obtenir la informació de la posició.

Totes les aeronaus que pertanyen a la Circulació Aèria General (CAG) i volen segons les normes IFR deuen estar equipades amb transponders Mode S de nivell 2 a partir del 31 de març de 2005. El mateix passa amb les aeronaus que volen en VFR a les quals se'ls hi ha deixat de marge fins al 30 de març de 2008, però abans del 30 de març de 2005 han d'anar equipades amb transponder Mode A/C o Mode S de nivell 2.

El transponder Mode S ha de complir les normes marcades per ICAO, referit a les telecomunicacions aeronàutiques, volum 4, referit a sistemes de radar de vigilància i sistema anticòlisió. Aquests han de ser de Mode S de nivell 2 i han de disposar de les següents funcions:

- Transmissió automàtica a terra de la identificació del vol (identificació del vol o matrícula)
- Gestió dels codis d'interrogacions de la identificació de vigilància (Suveillance Identifier - SI)
- Diversitat d'antena per les aeronaus amb una massa màxima superior a 5700 kg o que la seva velocitat vertadera màxima de creuer sigui superior a 250 nusos.

- Transmissió d'avisos de resolució (RA) emesos pel sistema d'anticol·lisió (ACAS) amb el qual l'aeronau està equipat.

La majoria de versions dels transponders conformes per la vigilància elemental, també seran capaços de fer squitter extés, i les modificacions seran mínimes.

Un transponder Mode S desenvolupat i qualificat per fer ES ha de complir certes condicions marcades a un cert nombre de documents que estipulen el seu concepte i la seva qualificació:

- Estàndards de funcions mínimes pel Mode S (inclouen el caràcter funcional de ES però no cobreixen totes les exigències que deuen respectar):
  - EUROCAE ED73B: especificacions de funcionament d'operació mínimes pels transponders SSR Mode S (gener 2003).
  - RTCA DO-181C: especificacions de funcionament d'operació mínimes per l'equipament a bord ATCRBS/ModeS (Air Traffic Control Radar Beacon System).
- Estàndards de funcions mínimes per l'ADS-B ES:
  - EUROCAE ED102: especificacions de funcionament d'operació mínimes per la funció ADS-B ES a Europa.
  - RTCA DO-260A: especificacions de funcionament d'operació mínimes per la funció ADS-B ES als Estats Units.
- Característiques de l'equip:
  - ARINC A718 Air Traffic Control Transponder (ATCRBS/Mode S): descriu les connexions físiques i funcionals requerides per la vigilància elemental i enriquida, i també per la funció de squitter extés. Especifica la informació per la concepció del transponder.
- Qualificació de l'equip:
  - JTSA C112a: especificacions de funcionament d'operació mínimes per transponders SSR Mode S, basats en la ED73A i que no inclou l'ES, permet al fabricant de l'equip de qualificar un transponder Mode S.
  - TSO C112A de la FAA: pel transponder Mode S, està basat en la RTCA DO-181, especificacions de funcionament d'operació mínimes que no inclouen el caràcter funcional de l'ES.
  - ATSO C1004: publicat a l'octubre de 2003 per la CAA australiana per l'ADS-B 1090 MHz.
  - La FAA prepara un nou TSO específic per l'ES.

També s'han de tenir en compte les connexions entre els transponders i el sistema de navegació per permetre la utilització de les informacions de posició dins de les transmissions ES.

Per poder utilitzar els equips per l'ADS-B, és necessari que hi hagi:

- Un emissor ES o un transponder Mode S o Mode A/C amb capacitat de ES.
- Equips que permetin fer arribar al transponder les dades de navegació i d'identificació del vol.



- Un enllaç compatible entre aquests tipus d'equipaments.

Tenint en compte tot el que han de dur les aeronaus per poder disposar de l'ADS-B, ens podem trobar en casos diferents, depenent dels equipaments que hi manquin o, simplement, només es necessitin modificar.

El primer grup d'aeronaus que podem trobar són les que disposen, en general, de sistemes de gestió de vol (FMS), sistemes de navegació (GNSS, central inercial) i transponder Mode S. Aquestes aeronaus únicament necessiten modificar el transponder per una versió amb capacitat de l'ADS-B ES, o activar l'emissió d'ES al transponder si aquest és d'última versió i ja té la capacitat d'emetre ES, i connectar el transponder amb els sistemes de gestió de vol o de navegació.

El segon grup d'aeronaus correspon a les que estan equipades amb el transponder Mode S. Els transponders deuran ser modificats o reemplaçats per disposar d'ES i complir les normes en vigor.

Dintre d'aquest segon grup, podem trobar tres tipus diferents d'arquitectures, que són les següents:

- Les aeronaus que disposen d'un receptor GPS adequat.
- Els avions que no tenen sistema de navegació, als quals se'ls hi haurà d'instal·lar les antenes corresponents i el cablejat necessari.
- Les aeronaus ja equipades amb GPS però que és necessari reemplaçar per un de nou, amb el cablejat adequat per connectar-lo al transponder.

El tercer grup d'aeronaus inclou aquelles que no estan equipades amb el transponder. En aquests casos, s'haurà d'instal·lar un transponder ADS-B 1090 i una o dues antenes ATC tot connectat amb el sistema de navegació existent. En aquest grup també podem diferenciar dos tipus d'arquitectures, segons si els avions disposen d'un receptor GPS o els que no el tenen, i necessiten de la instal·lació d'un sistema de navegació GPS, les antenes i la connexió amb el transponder.

## **4.2.2. Transponder Mode S ES**

En aquest apartat parlarem dels principals tipus de transponder Mode S amb capacitat per ADS-B ES que podem trobar al mercat actual. Dintre d'aquest mercat, els quatre fabricants principals són Honeywell/Allied Signal, Rockwell Collins, ACSS (Aviation Communications and Surveillance Systems) (L-3 Communications and Thales Company) i Garmin AT.

### *4.2.2.1. Honeywell*

Honeywell és una empresa que treballa en els camps de la tecnologia i la fabricació, que proporciona productes i serveis aerospacials, tecnologies de

control per edificis, habitatges i indústries, productes d'automoció, sistemes de generació d'energia, productes químics especials, fibres, plàstics i materials electrònics.

L'organització aeroespacial és proveïdora de components i sistemes d'aviònica, components per aeronaus i serveis de reparació i reglatge per línies aèria, fabricants de fuselatges, forces de defensa i governs. És un dels proveïdors d'equips i serveis aeroespacials més destacats del món, i presta serveis relacionats amb el transport aeri, l'aviació comercial i general, i als àmbits de defensa i espai aeri.

En el que a transponders es refereix, Honeywell ens ofereix dos transponders Mode S amb capacitat ADS-B ES. El primer que trobem és el TRA-67A dedicat al transport comercial. Aquest és un transponder Mode S de nivell 2, amb implementació dels codis d'identificació de vigilància (SI – surveillance identifier), que permet la transmissió de l'identificador del vol, el rumb magnètic, la velocitat del vent indicada, el nombre de Mach, la velocitat vertical, l'angle de balanceig, track angle rate, true track angle, la velocitat respecte al terra i l'altitud seleccionada. Aquest transponder compleix les normes en vigor i consta de dos models diferents, la versió 066-01127-1402 certificada i disponible a tots els nous Airbus i la versió 066-01127-1602 certificada i disponible per les aeronaus de nova producció de Boeing.

El preu d'aquest transponder és de 7700 \$, on s'inclou 6500 \$ del kit de muntatge i 1200 \$ de la mà d'obra.

L'altra transponder Mode S ofert per Honeywell amb capacitat per l'ADS-B ES és el KT73, enfocat a l'aviació general. Aquest transponder permet la transmissió de l'identificador del vol i els codis d'identificació de vigilància, i la posició, l'altitud i la direcció obtingudes del sistema GPS.

El preu del kit de muntatge del transponder KT73 és de 5230 \$.



**Fig. 4.2** Panell frontal del transponder KT73

#### 4.2.2.2. Rockwell Collins

Rockwell Collins és una empresa fundada al 1933 a Iowa, EEUU, per Arthur Collins, inicialment creada per dissenyar i produir equipaments d'ona curta, i que ha evolucionat al llarg dels anys, arribant a ser un dels majors proveïdors

d'electrònica i comunicació per l'aviació. Entre els productes principals d'aquesta empresa podem trobar productes de comunicació i electrònics de defensa pel govern, components electrònics d'aviació pel transport aeri comercial i productes electrònics per l'aviació regional i comercial.

Com es de suposar, entre els seus productes podem trobar dos models de transponders Mode S amb la funció ADS-B ES. Aquests dos models són el TDR 94/94D i el TPR 901.

El transponder TPR 901 és un transponder Mode S de nivell 2 dissenyat per l'aviació comercial, que opera amb transmissors Mode A i C d'ATCRBS, i també amb Mode S, i és compatible amb el sistema TCAS. Aquest transponder opera amb una funció de selecció de diversitat especial utilitzant dos receptors per millorar la vigilància entre aeronaus pel TCAS. Permet l'emissió d'ES. Aquest transponder també té dos models, el 822-1338-003 per Boeing i el 822-1338-021 per Airbus.

Els transponders TDR 94/94D (TDR 94: P/N 622-9352-007, TDR 94D: P/N 622-9210-007) estan dissenyats per aeronaus d'aviació regional i de negocis. El TDR 94D és de nivell 3A, d'acord amb els estàndards d'ICAO, i opera en Mode S.

#### 4.2.2.3. ACSS

Aviation Communication & Surveillance Systems és un dels líders en sistemes de seguretat per l'aviació amb productes que redueixen el potencial per col·lisions en l'aire i amb el terreny.

Dintre d'aquests sistemes de seguretat per aviació, ACSS proposa dos transponders Mode S amb capacitat per l'ADS-B ES, que són el XS950 i el RCZ 852. Tots dos compleixen les normes i reglamentacions en vigor respecte el Mode S, l'ACAS i l'ADS-B ES.

El XS950 està fabricat per ser utilitzat en aviació comercial. Aquest transponder és de nivell 4, d'acord amb els estàndards d'ICAO, i és capaç d'enviar i rebre informació. El preu d'aquest transponder està entre els 36000 \$ i els 41000 \$.



**Fig. 4.3** Transponder XS950

Els RCZ 852 està dissenyat per l'aviació general i de negocis. Aquest transponder és de nivell 3, d'acord amb els estàndards d'ICAO. Està fabricat amb compatibilitat amb el sistema TCAS. El seu preu és de 17000 \$.

#### 4.2.2.4. Garmin

Garmin és un dels líders en la tecnologia del sistema de posicionament global (GPS) i un innovador en electrònica. Treballa en el mercat de l'electrònica i l'aviació. Garmin es dedica al disseny, fabricació, i equipament de navegació i comunicació per l'aviació i pels consumidors. Els seus productes serveixen per l'aviació, la marina, l'automobilisme, wireless i aplicacions de recreació generals.

En el cas que ens trobem nosaltres, Garmin ens ofereix per aviació general, el GTX 330/330D. Aquests transponders operen en Mode S de nivell 2 amb capacitat per ES. El preu del GTX 30 és de 4835 \$.



**Fig. 4.4** Trasponder Garmin GTX 330/330D

### 4.3. Processament i Visualització

Per al processament i la visualització s'obtarà per un *Panel PC* ja que integra d'una forma compacta, robusta i a un bon preu pantalla TFT i computadora d'altres prestacions. Aquests models incorporen una pantalla tàctil per tal de facilitar el maneig del CDTI i eliminant la necessitat d'incorporar un teclat extern.

S'ha escollit per aquest propòsit una sèrie de models que s'adecuen a les característiques requerides per al CDTI de baix cost. L'elecció final dependrà dels requeriments del programa com ara processament, memòria, perifèrics, alimentació, tamany i resolució de pantalla i presupost.

Nota: Per a més informació sobre els *Panels PC*, vejeu *Annex C*.

#### 4.3.1. ATIOSys

*ATIO System, Inc.*, anteriorment *Industrial PC Division of Formosa Industrial Computing*, és una empresa subsidiària del *First International Computer (FIC) Group* i ofereix solucions i productes adequats a les necessitats industrials de les empreses. Estan especialitzats en les computadores empotrades, sistemes de video digitals i seguretat en xarxes.

Amb una experiència de més de 10 anys, és una empresa consolidada als EEUU, Japó i la Xina i en expansió a Europa i a la resta del sud-est asiàtic.

Actualment treballen per a la indústria japonesa en el control de maquinària, per Chinese Telecom, per la indústria militar chinesa, i per la indústria aeroespacial americana.

##### 4.3.1.1. ATP-120E

**Taula 4.1.** Especificacions ATP-120E

CPU	Via Eden 733MHz
Memòria	512 MB SDRAM
Pantalla	12" TFT Color TFT Touchscreen
SO	WinCE.net, IP65
Pes	5kg
Alimentació	AC90-264V, 100W
Dimensions	353x258x68.5(144.9)mm
Preu	1280\$
I/O	2xUSB1.1, 2xCOMs, COM2, Port Paral.lel, Ethernet



**Fig. 4.5** Visió Frontal-Lateral ATP-120E

### 4.3.2. Protech

Establerta al 1980, *Protech Systems Co.* S'ha especialitzat en el desenvolupament i creació de PCs industrials, productes d'automatització i computadores embotrades per a companyies de tot el món. Amb la tecnologia més avançada, Protech és una de les empreses líders del sector.

Protech basa tots els seus esforços en el seu departament de R+I+D (Recerca, Innovació i Desenvolupament) per tal de poder oferir als seus clients uns productes molt competitius i amb la última tecnologia.

Protech està certificada amb la ISO 9001 i la ISO 14001.

#### 4.3.2.1. PPC-7360

**Taula 4.2.** Especificacions PPC-7360

CPU	VIA Eden 733 MHz
Memòria	133-512M SDRAM
Pantalla	10.4" SXGA TFT LCD Touchscreen
SO	Win98SE/2k/XP/CE.net, Linux 8.0/9.0
Pes	6 kg
Alimentació	AC100W, DC80W
Dimensions	280x230x80mm
Preu	1244€
I/O	2xUSB1.1, 2xCOMs, COM2, Port Paral.lel, Ethernet



**Fig. 4.6** Visió Frontal PPC-7360

### 4.3.3. Axiomtek

Fundada al 1980 per un grup d'enginyers a Taiwan, Axiomtek és una de les majors companyies dissenyadores i productores en el camp dels PC industrials. Des de què es va establir, Axiomtek ha aconseguit el reconeixement a nivell mundial per els seus dissenys innovadors i amb la plena satisfacció dels seus clients.

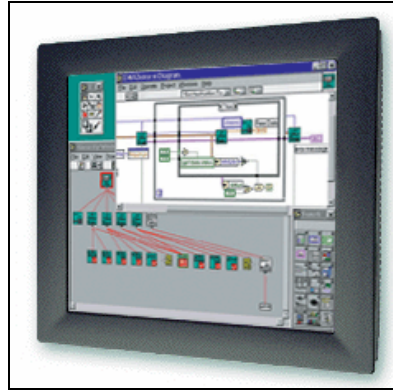
Els principals productes d'Axiomtek són les plataformes de computares empotrades, Panel PC, dispositius d'automatització industrials, plataformes IPC, dispositius de xarxes i dispositius d'emmagatzement. Té oficines a Taiwan, EEUU, Alemanya, França, Itàlia i Xina.

Axiomtek està certificada per la ISO-9001 i per ISO-14000.

#### 4.3.3.1. GOT-1120T

#### **Taula 4.4.** Especificacions GOT-1120T

CPU	Transmeta Crusoe 5400 (500MHz)
Memòria	128MB DRAM
Pantalla	12.1" SVGA TFT LCD Touchscreen
SO	Windows 95/98/NT/2k
Alimentació	DC24V, 0.8A
Dimensions	311x237x50mm
Preu	1708€
I/O	4xCOMs, port Paral.lel, port Ethernet, 1xUSB, 2xPS/2

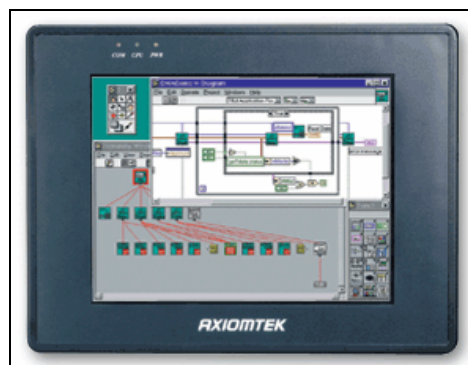


**Fig. 4.8** Visió Frontal GOT-1120T

#### 4.3.3.2. GOT-2770S

**Taula 4.5.** Especificacions GOT-2770S

CPU	Intel PXA255 200MHz
Memòria	64MB DRAM
Pantalla	7.7" VGA Color DSTN LCD TouchScreen
SO	Windows CE.NET/Compact Framework
Pes	1.2kg
Alimentació	DC24V, 0.5A
Dimensions	231x176x55mm
Preu	1092€
I/O	3xCOMs, port Paral.lel, port Ethernet, 1xUSB per ActiveSync



**Figura 4.9** Visió Frontal GOT-2770S

#### 4.3.4. Acnodes

*Acnodes Corporation* és un proveïdor de qualitat de computadores industrials per a control industrial, automatització en l'indústria i per a ús militar. Acnodes



produeix estacions de treball de tipus rack com ara pantalles planes, sistemes d'emmagatzemament, CPU, servidors i sistemes de control d'energia que permeten optimitzar i millorar els processos industrials. Els productes de Acnodes estan dissenyats per a treballar sota tot tipus de condicions amb un vida útil extensa.

Acnodes Corporation té la seu central a *City of Industry*, Califòrnia.

#### 4.3.4.1. PC1060

**Taula 4.3.** Especificacions PC1060

CPU	Via Eden 733MHz-1GHZ
Memòria	1GB DDR RAM
Pantalla	10.4" SVGA TFT Touchscreen
SO	WinCE.net 4.2/98/2k/XP, Linux 8.0/9.0
Pes	9 kg
Alimentació	AC110/120V, 100W
Dimensions	280x232x80mm
Preu	1495\$
I/O	2xUSB1.1, 2xCOMs, COM2, Port Paral.lel, Ethernet



**Fig. 4.7** Visió Frontal PC1060

## CAPÍTOL 5. CONCLUSIONS

En aquest capítol es valora fins a quin punt s'han complert les expectatives inicials, indicant les possibles causes de desviació. Es comprovaran si s'han obtingut els resultats esperats realitzant un balanç dels objectius fixats en un principi, es plantegen possibles millores i ampliacions futures així com el impacte mediambiental. Finalment s'exposen les conclusions personals sobre la realització d'aquest TFC.

### 5.1. Objectius assolits

Pràcticament s'han assolit tots els objectius marcats als objectius de la introducció.

En el disseny de la pantalla cartogràfica s'ha marcat unes pautes clares a seguir per a la seva posterior programació, indicant les característiques imprescindibles a visualitzar així com aquelles que podrien ser útils en cas que es volgués donar d'altres funcionalitats a l'aplicació.

La llibreria realitzada estructura d'una forma molt lògica les dades necessàries i implementa les principals funcions per al posterior funcionament de la pantalla cartogràfica. Entre aquestes, destaquem la funció de prevenció de conflictes, el simulador de trajectòries, el generador de coordenades així com la gestió per format d'arxiu \*.txt dels avions que formen l'aplicació i els *waypoints* del pla de vol.

Finalment, trobem l'estudi del maquinari necessari per a la seva implementació. En quan a sistema de recepció de dades de vigilància, s'ha determinat que el sistema més idoni és el ADS-B. Per tant, s'ha estudiat les diferents possibilitats en quant a mode de transmissió i s'ha determinat el Mode S com a tal. Com que no se sap en quin tipus d'avió es preten instal·lar aquesta pantalla cartogràfica, s'ha realitzat l'estudi en funció del possible maquinari existent a l'avió. En quan a processament i visualització s'ha triat un *Panel PC* perquè és el maquinari més semblant al que s'utilitzarà en la implementació final del projecte.

### 5.2. Desviació sobre el projecte inicial

L'estudi inicial realitzat ha sigut molt més extès del que s'havia fixat en un principi. Això bé justificat pel fet que en el món de l'aviació hi ha una normativa molt estricta sobre tots els temes que abarca. Per tant, ens em trobat finalment amb un treball amb una forta càrrega documental en deprimint del que s'havia pensat que seria el cos central del treball: la programació.

Dins aquesta programació, ha faltat per programar el motor gràfic de l'aplicació visual.

En quan a maquinari, en un principi d'havia pensat en una pantalla independent connectada a un PC industrial o un *Single Board Computer*. Però finalment es va escollir un *Panel PC* per la seva simplicitat.

### **5.3. Possibles millores i ampliacions futures**

El disseny podria millorar-se afegint-li més informació que pot ser d'utilitat com ara informació meteorològica, més dades del propi funcionament de l'avió o informació del terreny.

En quan a la llibreria, es podria millorar els algorismes de prevenció de conflictes en una ampliació futura.

Un cop fixat el tipus d'avió que incorporarà la pantalla cartogràfica, s'hauria de realitzar un estudi dels dispositius de recepció/emissió de què disposa l'avió i establir quin maquinari seria necessari incorporar, els dispositius d'interconnexió necessaris i el cablejat pertinent i realitzar el pressupost pertinent.

Finalment, s'hauria de dur a terme la programació final de l'aplicació visual tenint en conte les limitacions introduïdes per la computadora triada.

### **5.4. Impacte medioambiental**

Tots sabem que les radiacions de RF tenen efectes biològics i contraproductius per a la salut. Depenent de la potència, de la freqüència, de la geometria del sistema que s'exposa a elles i del tipus de teixit biològic, els efectes van des de l'estimulació del creixement a la mort de les cèl·lules (apoptosi) i a danya en l'ADN, sent els efectes tèrmics un factor important. Encara que en la nostra vida quotidiana el nivell d'exposició acostuma a mantenir-se per baix dels límits recomenats per la ICNIRP.

### **5.5. Conclusions personals**

La realització d'aquest projecte ha estat molt satisfactòria a nivell personal.

M'ha permès dissenyar una aplicació que pot ser d'utilitat a futurs estudiants de l'Escola Politècnica Superior de Castelldefels, participant a l'inici d'un projecte que espero que un pròxim estudiant acabi de realitzar.

Una de les tasques més importants ha estat la definició final dels objectius del projecte, i la seva finalitat.

Per una altra part, m'ha permès conèixer i aprendre un llenguatge de programació que pràcticament no havia vist, com és el C++, el qual només havia fet servir una vegada a una assignatura i de manera molt superficial.

Aquest projecte m'ha permès veure la importància que té la fase de disseny d'un projecte, ja que amb un bon disseny i estudi previ de la situació i dels objectius, la implementació és bastant més ràpida que sense.

Finalment cal destacar que la realització d'un projecte com aquest és una de les millors maneres de finalitzar la formació d'estudiants d'enginyeria ja que et mostra quines són les necessitats i els mètodes a l'hora de resoldre una sèrie de problemes plantejats i l'assoliment d'objectius.

## BIBLIOGRAFIA

- [1] *Guidance for Initial Implementation of Cockpit Display of Traffic Information*, RTCA SC-186 (1998)
- [2] *Minimum Aviation System Performance Standards for Automatic Dependent Surveillance Broadcast (ADS-B)*, RTCA DO-242 (1998)
- [3] *Cockpit Display of Traffic Information*, Airservices Australia - Technology Development (2005)
- [4] *3D-CDTI User Manual v2.1*, Nasa Ames Research Center - Flight Deck Research Group (2004)
- [5] Barhydt R., Palmer M.T., Wing D.J., *Use of a prototype Airborne Separation Assurance System for Resolving Near-term conflicts during autonomous aircraft operations*, Nasa Langley Research Center (2003)
- [6] Prinzo O.V., *ADS-B / CDTI: Innovations in Pilot-Managed Departures*, Federal Aviation Administration - Civil Aerospace Medical Institute (2002)
- [7] Alexander A.L., Wickens C.D., *Cockpit Display of Traffic Information: The effects of traffic load, dimensionality, and vertical profile orientation*, University of Illinois - Aviation Research Lab (2001)
- [8] *CAS 67A ACAS II Pilot's Guide*, Allied Signal Inc. (1999)
- [9] Owusu K., Dunstone G., *Development of Cockpit Display of Traffic Information (CDTI)*, International Civil Aviation Organization (2005)
- [10] *Campanya d'Estiu 2004*, Generalitat de Catalunya, Departament d'Interior - Direcció General d'Emergències i Seguretat Civil (2004)
- [11] *ADS-B Study and Implementation Task Force Meeting*, International Civil Aviation Organization (2003)
- [12] *Clarification Mode S Transponder in a Airport/A-SMGCS Environment*, European Organisation for the Safety of Air Navigation (2003)
- [13] <http://www.rtca.org/>
- [14] <http://www.iaopa.org/>
- [15] <http://www.gpc.se/>
- [16] <http://www.atiosys.com/>
- [17] <http://www.axiomtek.com.tw/>

- [18] <http://www.protech.com.tw/>
- [19] <http://www.acnodes.com/>
- [20] <http://www.ipcmax.com/>
- [21] <http://www.honeywell.es/>
- [22] <http://www.rockwellcollins.com/>
- [23] <http://www.acssonboard.com/>
- [24] <http://www.garmin.com/>
- [25] <http://www.am-avionics.com/>

## ANNEX A. GLOSSARI

ACAS	Airborne Collision Avoidance System
ACM	Airborne Conflict Management
ACSS	Aviation Communications and Surveillance Systems
ADS-B	Automatic Dependent Surveillance - Broadcast
ATC	Air Traffic Control
ATCRBS	Air Traffic Control Radar Beacon System
ATSO	Airborne Technical Standard Order
CAA	Civil Aviation Authority
CAG	Circulació Aèria General
CDTI	Cockpit Display of Traffic Information
CPU	Central Processing Unit
EEUU	Estats Units
EUROCAE	European Organization for Civil Aviation Equipment
FIC	First International Computer
FL	Flight Level
FMS	Flight Management System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
ICAO	International Civil Aviation Organization
IFR	Instrument Flight Rules
ISO	International Organization for Standardization
ITC	In-Trail Climb
ITD	In-Trail-Descent
JTSO	Joint Technical Standard Order
kts	Nusos
MASPS	Minimum Aviation System Performance Standards
MFD	Mutli-Function Display
mn	Milles nàutiques
nm	Milles nàutiques
RA	Resolution Advisory
RTCA	Radio Technical Commission for Aeronautics
SI	Surveillance Identifier
SSR	Secondary Surveillance Radar
TCAS	Traffic Alert and Collision Avoidance System
TIS	Traffic Information Service
TIS-B	Traffic Information Service - Broadcast
UAT	Universal Access Transceiver
VDL	VHF Data Link
VFR	Visual Flight Rules





## ANNEX B. CODI LLIBRERIA

### B1. Coordinate.h

```

#ifndef __COORDINATE__
#define __COORDINATE__

#include <vector>
#include <string>
#include <fstream>

using namespace std;

#include "formulas.h"

class ccoordinate
{
public : ccoordinate (void);           //Basic builder of ccoordinate

public : ccoordinate (double latitude, double longitude, double height,
                    double time, double altitude, double std);
//Builder of ccoordinate with the parameters. Units: latitude, longitude in
degrees - height, altitude in feets - std in kts - time in seconds

private : double latitude_;           //Latitude in radiants

private : double longitude_;          //Longitude in radiants

private : double height_;             //Height in meters

private : double time_;               //Time in seconds

private : double altitude_;           //Barometric altitude in meters

private : double std_;                //Velocity in meter/second

public : void set_lat (double latitude) { latitude_ = latitude; }
//Set coordenates from parameters: latitude. Units: latitude in radiants

public : void set_lon (double longitude) { longitude_ = longitude; }
//Set coordenates from parameters: longitude Units: longitude in radiants

public : void set_hei (double height) { height_ = height; }
//Set coordenates from parameters: height. Units: height in meters

public : void set_time (double time) { time_ = time; }
//Set time from parameter. Unit: Time in seconds

```

```

public : void set_alt (double altitude) { altitude_ = altitude; }
//Set barometric altitude from parameter. Unit: feet

public : void set_std (double std) { std_ = std; }
//Set STD (velocity) from parameter. Unit: kts

public : double latitude (void) { return latitude_; }
//Returns latitude value in radians

public : double longitude (void) { return longitude_; }
//Returns longitude value in radians

public : double height (void) { return height_; }
//Returns height value in meters

public : double time (void) { return time_; }
//Returns time value in seconds

public : double altitude (void) { return altitude_; }
//Returns barometric altitude in meters

public : double std (void) { return std_; }
//Returns STD (velocity) in meter/second

};

#endif

```

## B2. Coordinate.cpp

```

#include "coordinate.h"

//Basic builder of ccoordinate
ccordinate::ccordinate (void)
{
    set_lat(0.0),
    set_lon(0.0);
    set_hei(0.0);
    set_time (0.0);
    set_alt(0.0);
    set_std(0.0);
}

//Builder of ccoordinate with the parameters. Units: latitude, longitude in
degrees - height, altitude in feets - std in kts - time in seconds
ccordinate::ccordinate (double latitude, double longitude, double height,
                        double time, double altitude, double std)

```

```

{
    set_lat(latitude);
    set_lon(longitude);
    set_hei(height);
    set_time(time);
    set_alt(altitude);
    set_std(std);
}

```

### B3. Plane.h

```

#ifndef __PLANE__
#define __PLANE__

#include <vector>
#include <string>
using namespace std;

#include "flightpath.h"

class cplane
{
public : cplane (void);
        //Basic constructor of cplane --> Set plane number = 0

public : cplane (string file);
        //Constructor of cplane that load form file the coordenates of each plane

private : string name_;
        //Plane Name

private : int state_;
        //Plane state: NOT LOAD-STOPPED-NORMAL-CONFLICT-DANGER
        [-1,0,1,2,3]

private : cflightpath flightpath_;
        //Statement of cflightpath from the plane

public : double bearing_[2];
        //Plane Bearing Vector. Unit: seconds

public : vector<ccordinate> coor_;
        //Vector statement of ccordinate

public : void set_name (string name) { name_ = name; }
        //Sets the name of the plane

```

```

private : void set_state (int state)      {      state_ = state;      }
//Sets the state of the plane -- NOT LOADED-STOPPED-NORMAL-
CONFLICT-DANGER [-1,0,1,2,3]

private : void set_flightpath (cflightpath flightpath)
{      flightpath_ = flightpath;      }
//Sets the flightpath of the plane

private : void set_bearing (double x, double y);      //In seconds

private : void set_coor (ccoordinate coor);
//Sets the coordinate number i of the vector

public : string name (void)      {      return name_;      }
//Returns the name of the plane

private : int state (void)      {      return state_;}
//Returns the state of the plane -- NOT LOADED-STOPPED-NORMAL-
CONFLICT-DANGER [-1,0,1,2,3]

public : cflightpath flightpath (void)      {      return flightpath_; }
//Returns the flightpath of the plane

public : double xbearing (void)      {      return bearing_[0]; }
//Returns the X component of bearing. Unit: seconds

public : double ybearing (void)      {      return bearing_[1]; }
//Return the Y component of bearing. Unit: seconds

public : ccoordinate coor (int pos)      {      return coor_[pos]; }
//Returns the coordinate of the plane in the position coor

public : void add_info_flightpath (string file);
//Function that load from file the name and flightpath of the plane

public : ccoordinate calc_coor (double time);
//Returns the coordinate of the plane in a specific time

public : void calc_bearing (void);
//Calculate the plane bearing

public : ~cplane (void);      //Destructor of cplane

};

#endif

```

## B4. Plane.cpp

```
#include "plane.h"

#define MAX 5

//Basic constructor of cplane --> Set plane number = 0 and name = NULL
cplane::cplane (void)
{
    set_name("");
    set_state(-1);
    cflightpath flightpath_ = cflightpath();
    ccoordinate c_temp = ccoordinate();
    for(int i=0; i<MAX; i++)    set_coor(c_temp);
    set_bearing(0, 0);
}

//Sets the coordenate number i of the vector
void cplane::set_coor (ccordinate coor)
{
    coor_.push_back(coor);

    if(coor_.size() == MAX)
        coor_.erase(0,0);
}

//Sets bearing coordiantes
void cplane::set_bearing (double x, double y)
{
    bearing_[0] = x;
    bearing_[1] = y;
}

//Function that load from file the name and flightpath of the plane
void cplane::add_info_flightpath (string file)
{
    ifstream f;
    int i;
    string temp;

    f.open(file.c_str());

    getline(f,temp);
    f>>i;

    set_name(temp);

    ccoordinate c_temp = ccoordinate();
```

```

for(i=0;!f.eof(); i++)
{
    double lon, lat, hei, alt, std;

    f>>lat;
    f.ignore(1,':');
    f>>lon;
    f.ignore(1,':');
    f>>hei;
    f.ignore(1,':');
    f>>alt;
    f.ignore(1,':');
    f>>std;

    c_temp.set_lat(deg_to_rad(lat));
    c_temp.set_lon(deg_to_rad(lon));
    c_temp.set_hei(ft_to_m(hei));
    c_temp.set_alt(ft_to_m(alt));
    c_temp.set_std(kts_to_ms(std));
    flightpath().set_waypoint(c_temp);

    flightpath().calc_waypoint_time(i);
}

f.close();
}

//Returns the coordinate of the plane in a specific time
ccoordinate cplane::calc_coor(double time)
{
    int i;

    for (i=0; i<flightpath().waypoint_num() ; i++)
    {
        if ( time == flightpath().waypoint(i).time() )
            return flightpath().waypoint(i);

        else if ( ( time > flightpath().waypoint(i).time() ) &&
            ( time < flightpath().waypoint(i+1).time() ) )
            break;
    }

    ccoordinate c_temp = ccoordinate();
    double aux=0.0;

    c_temp.set_time(time);

    double time1 = time - flightpath().waypoint(i).time();

```

```

double      time2      =      flightpath().waypoint(i+1).time()      -
flightpath().waypoint(i).time();

aux = flightpath().waypoint(i).std() + (flightpath().waypoint(i+1).std() -
      flightpath().waypoint(i).std() ) * time1 / time2 ;

c_temp.set_std(aux);

aux      =      aux      =      flightpath().waypoint(i).altitude()      +
(flightpath().waypoint(i+1).altitude() -
      flightpath().waypoint(i).altitude() ) * time1 / time2 ;

c_temp.set_alt(aux);

aux      =      flightpath().waypoint(i).latitude()      +
(flightpath().waypoint(i+1).latitude() -
      flightpath().waypoint(i).latitude() ) * time1 / time2 ;

c_temp.set_lat(aux);

aux      =      flightpath().waypoint(i).longitude()      +
(flightpath().waypoint(i+1).longitude() -
      flightpath().waypoint(i).longitude() ) * time1 / time2 ;

c_temp.set_lon(aux);

aux = flightpath().waypoint(i).height() + (flightpath().waypoint(i+1).height()
- flightpath().waypoint(i).height() ) * time1 / time2 ;

c_temp.set_hei(aux);

return c_temp;

}

//Destructor of cplane
cplane::~cplane (void)
{
    name_.erase(0,name_.size());
    coord_.clear();
}

```

## B5. Flightpath.h

```
#ifndef __flightpath__
```

```

    #define __flightpath__

#include "coordinate.h"

class cflightpath
{

private : vector<ccoordinate> waypoint_ ;
//Array of waypoint coordinate

public : void set_waypoint (ccoordinate waypoint)
        {      waypoint_.push_back(waypoint);      }
//Sets waypoint coordinate at position i of the array waypoint_

public : int waypoint_num (void)      {      return waypoint_.size();      }
//Returns waypoint number

public : ccoordinate waypoint (int pos) {      return waypoint_[pos];      }
//Returns waypoint coordinate of the position pos of the array waypoint_

public : void calc_waypoint_time(int pos);
//Set time of date and hour from waypoint_[pos]

public : ~cflightpath (void);      //Destructor of cflightpath

};

#endif

```

## B6. Flightpath.cpp

```

#include "flightpath.h"

//Set time of date and hour from waypoint_[pos]
void cflightpath::calc_waypoint_time(int pos)
{
    if( !pos )
        waypoint(pos).set_time(0);
    else
    {
        double      d      =      distance_2points(waypoint(pos).latitude(),
        waypoint(pos).longitude(),
        waypoint(pos-1).latitude(), waypoint(pos-1).longitude() );

        d = nm_to_m( rad_to_nm(d) );
        double time = d / waypoint(pos).std();
        time = time - waypoint(pos-1).time();
        waypoint(pos).set_time(time);
    }
}

```



```

    }
}

//Destructor of cflightpath
cflightpath::~cflightpath (void)
{
    if( !waypoint_.empty() )
        waypoint_.clear();
}

```

## B7. Library.h

```

#ifndef __LIBRARY__
#define __LIBRARY__

#include "plane.h"

#define MAX_COOR 5

class clibrary
{

public : clibrary (void);
        //Constructor of clibrary: Set plane number as 0.

public : clibrary (string file);
        //Constructor of clibrary that loads the planes from file

public : vector<cplane> plane_;           //Array statement of cplane

public : double time_;                   //Time of the system

private : double collision_radius_;      //Collision radius. Unit: meters

private : double collision_height_;      //Collision height. Unit: meters

private : double alert_radius_;         //Alert radius. Unit: meters

private : double alert_height_;         //Alert height. Unit: meters

private : int plane_selected_;          //Plane selected in the visual aplicacion

private : double period_;                //Period in seconds

```

```
public : void set_plane (cplane plane) { plane_.push_back(plane); }
//Sets a plane in the position i. If i=-1 puts the plane at the end of the
vector

public : void set_time (double time) { time_ = time; }
//Time in seconds

public : void set_collision_radius (double radius)
{ collision_radius_ = radius; }
//Radius in meters

public : void set_collision_height (double height)
{ collision_height_ = height; }
//Height in meters

public : void set_alert_radius (double radius)
{ alert_radius_ = radius; }
//Radius in meters

public : void set_alert_height (double height)
{ alert_height_ = height; }
//Height in meters

public : void set_plane_selected (int pl_s)
{ plane_selected_ = pl_s; }

public : void set_period (double period) { period_ = period; }
//Period in seconds

public : int plane_num (void) { return plane_.size(); }
//Returns the number of planes

public : cplane plane(int i) { return plane_[i]; }
//Returns a clibrary from the position i of the vector

public : double time (void) { return time_; }
//Time in seconds

public : double collision_radius (void)
{ return collision_radius_; }
//Radius in meters

public : double collision_height (void) { return collision_height_; }
//Height in meters

public : double alert_radius (void) { return alert_radius_; }
//Radius in meters
```

```

public : double alert_height (void){      return alert_height_;    }
//Height in meters

public : int plane_selected (void) {      return plane_selected_;  }

public : double period (void)    {      return period_;      }
//Period in seconds

public : int load_planes (string file);
//Loads from file the number of planes and its location in hard drive
type --> 0 Coordinate_Simulator <-> 1 Flight_Simulator

public : void load_asas (string file);
//Load Collision and Alert parameters from file

public : int asas (int pos);
//Returns 0 if the situation of the plane is OK, -1 if there is a plane in the
alert zone, -2 if there's a plane in the collision zone

public : ~clibrary (void);                //Destructor of clibrary

};

#endif;

```

## B8. Library.cpp

```

#include "library.h"

//Constructor of clibrary: Set plane number as 0.
clibrary::clibrary (void)
{
    set_collision_radius (0.0);
    set_collision_height (0.0);
    set_alert_radius (0.0);
    set_alert_height (0.0);
    set_time (0.0);
    set_plane_selected (0);
    set_period(0);
}

//Constructor of clibrary that loads the planes from file
clibrary::clibrary (string file)
{
    load_planes(file.c_str());
}

```

```
}

```

```
//Loads from file the number of planes and its location in hard drive
```

```
// type --> 0 Coordinate_Simulator <-> 1 flightpath_Simulator
```

```
int clibrary::load_planes (string file)
```

```
{
```

```
    int i, num=0;
```

```
    ifstream f;
```

```
    string temp;
```

```
    f.open(file.c_str());
```

```
    if(!f.is_open())
```

```
        return 0;
```

```
    f>>num;
```

```
    f.ignore(1, '\n');
```

```
    set_plane_selected(0);
```

```
    cplane p_temp = cplane();
```

```
    for(i=0; !f.eof(); i++)
```

```
    {
```

```
        getline(f, temp);
```

```
        p_temp.add_info_flightpath (temp);
```

```
        set_plane(p_temp);
```

```
    }
```

```
    f.close();
```

```
    return 1;
```

```
}
```

```
//Load Collision and Alert parameters from file
```

```
void clibrary::load_asas (string file)
```

```
{
```

```
    ifstream f;
```

```
    int aux;
```

```
    f.open (file.c_str() );
```

```
    f>>aux;
```

```
    set_collision_radius(nm_to_rad(aux));
```

```
    f.ignore(1, ':');
```

```
    f>>aux;
```

```
    set_collision_height(ft_to_m(aux));
```

```
    f.ignore(1, ':');
```

```
    f>>aux;
```

```
    set_alert_radius(nm_to_rad(aux));
```

```

    f.ignore(1, ':');

    f>>aux;
    set_alert_height(ft_to_m(aux));
    f.ignore(1, ':');

    f>>aux;
    set_period(1000*aux);

    f.close();
}

//Returns 0 if the situation of the plane is OK, -1 if there is a plane in the alert
zone, -2 if there's a plane in the collision zone
int clibrary::asas(int pos)
{
    double d, h;

    for(int i=0; i<plane_.size() ; i++)
    {
        if(i!=pos)
        {
            d = distance_2points( plane(i).coor(MAX_COOR-
1).latitude(), plane(i).coor(MAX_COOR-1).longitude(),
plane(pos).coor(MAX_COOR-1).latitude(),
plane(pos).coor(MAX_COOR-1).longitude() );

            h = mod (plane(i).coor(MAX_COOR-1).altitude() -
plane(i).coor(pos).altitude() );

            if ( ( d <= collision_radius() ) && ( h <= collision_height() ) )
                return -2;
            else if ( ( d <= alert_radius() ) && ( h <= alert_height() ) )
                return -1;
        }
    }
    return 0;
}

//Destructor of clibrary
clibrary::~clibrary (void)
{
    if( !plane_.empty() )
        plane_.clear();
}

```

## B9. Formulas.cpp

```
#include <math.h>

#include "formulas.h"

#define PI 3.1415926536

//Returns distance between two points
//Units: latX, lonX and returned in radians
double distance_2points(double lat1, double lon1, double lat2, double lon2)
{
    return mod (2 * asin(sqrt(sin((lat1-lat2)/2)*sin((lat1-lat2)/2)+cos(lat1)*
        cos(lat2)*sin((lon1-lon2)/2)*sin((lon1-lon2)/2))));
}

//Returns the module of mod
double mod (double mod)
{
    if(mod<0)
        return mod*(-1);
    else
        return mod;
}

//Convert from degrees to radians. Units: d in seconds
double deg_to_rad (double d)
{
    return (d*PI)/180.0;
}

//Convert from radians to degrees. Units: Returned in seconds
double rad_to_deg (double d)
{
    return (d*180)/PI;
}

//Convert from radians to nautical milles
double rad_to_nm (double d)
{
    return (3/PI)*d;
    //distance_nm=((180*60)/PI)*distance_radians
}

//Convert from nautical milles to radians
double nm_to_rad (double d)
{
    return (PI/3.0)*d;
    //distance_radians=(PI/(180*60))*distance_nm
}
```

```
//Convert from nautical milles to killometers  
double nm_to_m (double d)  
{  
    return 1852*d;  
}
```

```
//Convert from killometers to nautical milles  
double m_to_nm (double d)  
{  
    return d/1852.0;  
}
```

```
//Convert from kts to meters/second  
double kts_to_ms (double d)  
{  
    return nm_to_m(d)/3600;  
}
```

```
//Convert from meters/second to kts  
double ms_to_kts (double d)  
{  
    return m_to_nm(d)*3600;  
}
```

```
//Convert from hours to seconds  
double h_to_s (double d)  
{  
    return d*3600;  
}
```

```
//Convert from feet to meters  
double ft_to_m (double d)  
{  
    return d/3.28084;  
}
```

```
//Convert from meters to feet  
double m_to_ft (double d)  
{  
    return d*3.28084;  
}
```

## B10. Fitxers de càrrega

- Arxiu de càrrega d'avions:

```

Archivo Edición Formato
Number_of_planes
Plane1_Location
Plane2_Location
.
.
PlaneN_Location

```

**Fig. B1** Estructura arxiu de càrrega d'avions

- Arxiu de càrrega d'un avió
  - LATITUDE in degrees<double>
  - LONGITUDE in degrees<double>
  - HEIGHT in feet<double>
  - ALTITUDE in feet <double>
  - STD(Velocity) in kts <double>

```

Archivo Edición Formato Ver Ayuda
Plane_Number
waypoint_numbers
LATITUDE:LONGITUDE:HEIGHT:ALTITUDE:STD(Velocity)
LATITUDE:LONGITUDE:HEIGHT:ALTITUDE:STD(Velocity)
...

```

**Fig. B2** Estructura arxiu de càrrega d'un avió

- Arxiu de càrrega de *Collision and Alert Parameters*
  - Collision\_radius in nm<double>
  - Collision\_height in feet<double>
  - Alert\_radius in nm<double>
  - Alert\_height in feet<double>
  - Period in seconds<double>

```

Archivo Edición Formato Ver Ayuda
Collision_radius:Collision_height:Alert_radius:Alert_height:Period

```

**Fig. B3** Estructura arxiu de càrrega Collision and Alert Parameters



## **ANNEX C. DATASHEETS PANEL PC**



## **C1. ATP-120**



## **C2. PPC-7360**



### **C3. GOT-1120T**





## **C4. GOT-2770S**



## **C5. PC1060**