2007

Software architecture for controlling an indoor hovering robot from a remote host

Ambika Asthana

University of Wollongong

UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or
study. The University does not authorise you to copy, communicate or otherwise make available
electronically to any other person any copyright material contained on this site. You are reminded of
the following:

This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part
of this work may be reproduced by any process, nor may any other exclusive right be
exercised, without the permission of the author.

Copyright owners are entitled to take legal action against persons who infringe their
copyright. A reproduction of material that is protected by copyright may be a copyright
infringement. A court may impose penalties and award damages in relation to offences and
infringements relating to copyright material. Higher penalties may apply, and higher damages
may be awarded, for offences and infringements involving the conversion of material into
digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do
not necessarily represent the views of the University of Wollongong.

Recommended Citation

Asthana, Ambika, Software architecture for controlling an indoor hovering robot from a remote host, MCompSc-Res thesis, School of

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library:
research-pubs@uow.edu.au
NOTE

This online version of the thesis may have different page formatting and pagination from the paper copy held in the University of Wollongong Library.

UNIVERSITY OF WOLLONGONG

COPYRIGHT WARNING

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.
SOFTWARE ARCHITECTURE FOR CONTROLLING AN INDOOR HOVERING ROBOT FROM A REMOTE HOST

A thesis submitted in partial fulfilment of the requirements for the award of the degree

Master of Computer Science – Research

from

University of Wollongong

by

Ambika Asthana

School of Computer Science and Software Engineering

2007
Declaration

I, Ambika Asthana, declare that this thesis, submitted in partial fulfillment of the requirements for the award of Masters of Computer Science (Research), in the Department of Informatics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Ambika Asthana
30 August 2007.
Abstract

To achieve stable autonomous control of an indoor flying robot is a challenging proposition in the field of robotics today. Many researchers are inspired by the echolocation of bats and vision of bees and attempt to duplicate this behaviour by using sonar sensors, cameras and onboard microprocessors. This project aims to achieve the same goal but with a different approach. We propose to build a software architecture for controlling a four-rotor helicopter, DraganFlyer, from a host computer. In order to do this, we equipped the DraganFlyer with communication devices, an Inertial Navigation Sensor (INS) and batteries.

The DraganFlyer is a four-rotor helicopter that can hover and move freely in air. Due to the near zero friction and damping at slow velocities it is marginally stable in six degrees of freedom. The aim of the overall research project is to understand the dynamics of the DraganFlyer and hence to achieve hover without drift and trajectory following without wandering. Development of software to achieve this level of control from a remote host poses a significant software design problem.

This thesis focuses on the software design problem, i.e. the design and development of real-time software for measuring the dynamics and for control of the DraganFlyer. The software runs on a host Macintosh and is divided into three main sections. One is the measurement of DraganFlyer motion with an INS. The second is the calculation of the control commands. The third is the control of the DraganFlyer via the radio control handset.

As these sections have different timing requirements, a significant part of the software design and testing time was spent examining how to decompose the system based on timing requirements and constraints. Then we had to determine how to couple these modules together to achieve overall timing goals without data loss.

Two types of experimental results are presented. The first results are to test the software, both the correctness of the calculations and their timeliness. The second are measurements of the open loop response of the DraganFlyer.
Acknowledgements

I would like to thank Phillip McKerrow for his guidance, support, advice and patience while supervising this project. He always believed I could do it!

A warm thank you to my parents and brother for their unflattering support and encouragement throughout the course of my research.

A special thanks to my friends: Lyris Rodrigues, Darryl Correa, Sheenal Shrivastava and Sangeetha Ramu for always standing by me and supporting me in every way possible.

Also, a big thank you to the Robotics Group for their enthusiasm and ideas!
Contents

DECLARATION ................................................................. I
ABSTRACT ................................................................. II
ACKNOWLEDGEMENTS .................................................. III
CONTENTS ................................................................. IV

1.0 INTRODUCTION ..................................................... 6
  1.1 Objectives .......................................................... 6
  1.2 Research Methodology ......................................... 6
  1.3 Software Issues ................................................. 7
  1.4 Literature Review ............................................... 8
  1.5 Overview of Thesis ............................................ 9

2.0 MODEL OF DRAGANFLYER ........................................ 11
  2.1 Model .............................................................. 11
  2.2 Coordinate Frames ............................................. 12
  2.3 Dynamics ......................................................... 12
  2.4 Control ............................................................ 12

3.0 INERTIAL NAVIGATION SENSOR (INS) ............. 15
  3.1 Physical Implementation of INS ............................ 15
  3.2 How INS Works ................................................. 16
  3.3 Advantages & Disadvantages of INS ..................... 17
  3.4 Reasons for Calibration .................................... 19
  3.5 Gyroscope ....................................................... 19
  3.6 Accelerometer .................................................. 20

4.0 INS MEASUREMENT SOFTWARE .............................. 21
  4.1 Hardware Set-up ............................................... 21
  4.1.1 Hardware .................................................... 21
  4.1.2 Advantages of Lithium Batteries ..................... 22
  4.1.3 Disadvantages of Lithium Batteries .................. 22
  4.2 INS Output ...................................................... 24
  4.2.1 Data Output Modes ....................................... 24
  4.2.2 Data Output Types ....................................... 25
  4.2.3 Packet types and Structure .......................... 25
  4.3 Software ........................................................ 29
  4.4 Data Extraction ............................................... 38
  4.5 Data Conversion ............................................... 39
  4.5.1 Zeroing Process .......................................... 40
  4.5.2 Integration ................................................ 40
  4.5.3 Sampling Time ............................................ 43
6.0 MOTION COMMAND SOFTWARE ........................................... 60
6.1 Bi-directional Hardware set-up ........................................ 61
6.2 PCBuddy ........................................................................ 61
   6.2.1 Servo Control Frames ........................................... 61
   6.2.2 Multi-Channel Frame ............................................ 61
   6.2.3 Downloading Servo Position Data ............................ 62
   6.2.4 Modulating the Transmitted Carrier ......................... 63
6.3 Software ....................................................................... 63
   6.3.1 Software Stages .................................................... 66

7.0 CONTROL SOFTWARE .................................................. 68
7.1 Control Theory ............................................................. 68
   7.1.1 Control Loop Basics ............................................ 68
   7.1.2 PID Controller Theory ........................................ 69
   7.1.3 PID Algorithm Implementation ............................. 71
   7.1.4 Limitations ......................................................... 71
7.2 Software ...................................................................... 72
   7.2.1 Software Stages .................................................... 73
7.3 System advantages/disadvantages ................................. 77
7.4 Open-loop control ....................................................... 77
   7.4.1 Test Design ......................................................... 77
   7.4.2 Test Result .......................................................... 79
7.5 Closed-loop control ..................................................... 83
   7.5.1 Software Design .................................................. 83
   7.5.2 Test Design .......................................................... 83
   7.5.3 Experiments ......................................................... 86

8.0 CONCLUSION ............................................................... 87
8.1 Future work ............................................................... 88

9.0 REFERENCES .............................................................. 89

10.0 APPENDICES ............................................................ 91
10.1 Appendix A ............................................................... 91
10.2 Appendix B ............................................................... 101