# Failsafe Control for DTUSat2

Kasper Bjørn Nielsen

Kongens Lyngby 2010
IMM-PHD-2010-1

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

# Abstract

The DTUSat-2 project is a student satellite project at DTU. Any software being run on the satellite could potentially make the satellite unresponsive if a serious error occured. Therefore, a failsafe mode has been developed which should catch theses failures so that the DTUSat-2 staff can investigate errors and prevent them from happening again by uploading new software. To operate the failsafe mode the staff needs both a console tool and a graphical user interface. It is the purpose of this project to investigate possible solutions and produce the software necesarry for the staff to be able to operate the satellite when in failsafe mode.

# Preface

This thesis was prepared at Informatics Mathematical Modelling, the Technical University of Denmark in fulfillment of the requirements for acquiring the Bachelor degree in engineering.

This bachelor thesis documents the Failsafe Control for DTUSat-2 software. It is aimed at the DTUSat-2 staff and should give any staff member a thorough overview of the requirements, design decisions, implementation specifics, test results, installation instructions and operating instructions of the software produced. A big thanks should go out to my supervisor Hans Henrik Løvengreen who has been very helpful with all aspects of the project.

Lyngby, June 2010

Kasper Bjørn Nielsen, s052808

# Contents

Appendix A

# Failsafe Commands

A list of the 20 failsafe commands

- `calculate_check_sum (address, length)`
- `call_function (address,parameter)`
- `copy_to_flash (from,to,length)`
- `copy_to_ram (from,to,length)`
- `delete_flash_block (address)`
- `download (address, length)`
- `download_sib`
- `execute (address)`
- `flash_test (address)`
- `health_status`
- `ram_test (address, length)`
- `read_register (address)`

- `read_sensor (address)`
- `reset`
- `reset_sib`
- `set_autoreset (value)`
- `unlock_flash`
- `upload (address, data)`
- `upload_sib (data)`
- `write_register (address, data)`