Flight Software Implementation and Verification on IRIS CubeSat

34th Small Satellite Conference
SSC20-WP1-22

Yun-Rong Yang, Ke-Yen Hsu, and Jyh-Ching Juang
Department of Electrical Engineering, National Cheng Kung University, Taiwan

jobo860723@gmail.com
http://unisectaiwan.ee.ncku.edu.tw/

Abstract:
Flight software (FSW) is essential in the operation of a satellite program. It is important to verify the functionality and performance of the FSW on ground before being deployed in space. We discusses the implementation and verification of the FSW of the Intelligent Remote-Sensing and Internet Satellite (IRIS) CubeSats.

01 Introduction

The IRIS program aims to design and launch two CubeSats: IRIS-A and IRIS-B.

Mission objective:

IRIS-A → Internet of Things (IoT) demonstration
- LoRa implementation
- Doppler frequency estimation/compensation

IRIS-B → In-orbit intelligent remote sensing

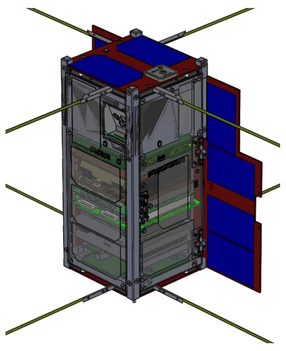
02 Overview of IRIS

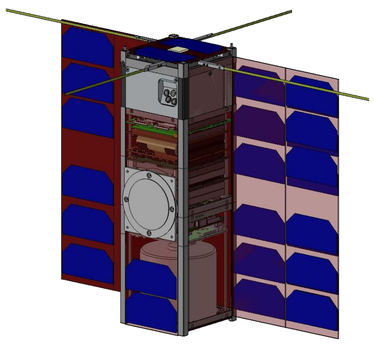
Subsystems:
- EPS
- OBDH
- TT&C
- ADCS
- Motherboard
- Daughterboard

Payload:

IRIS-A
- GPSR
- Reference clock
- LoRa receiver

IRIS-B
- GPSR
- RS payload





03 Flight Software Requirement

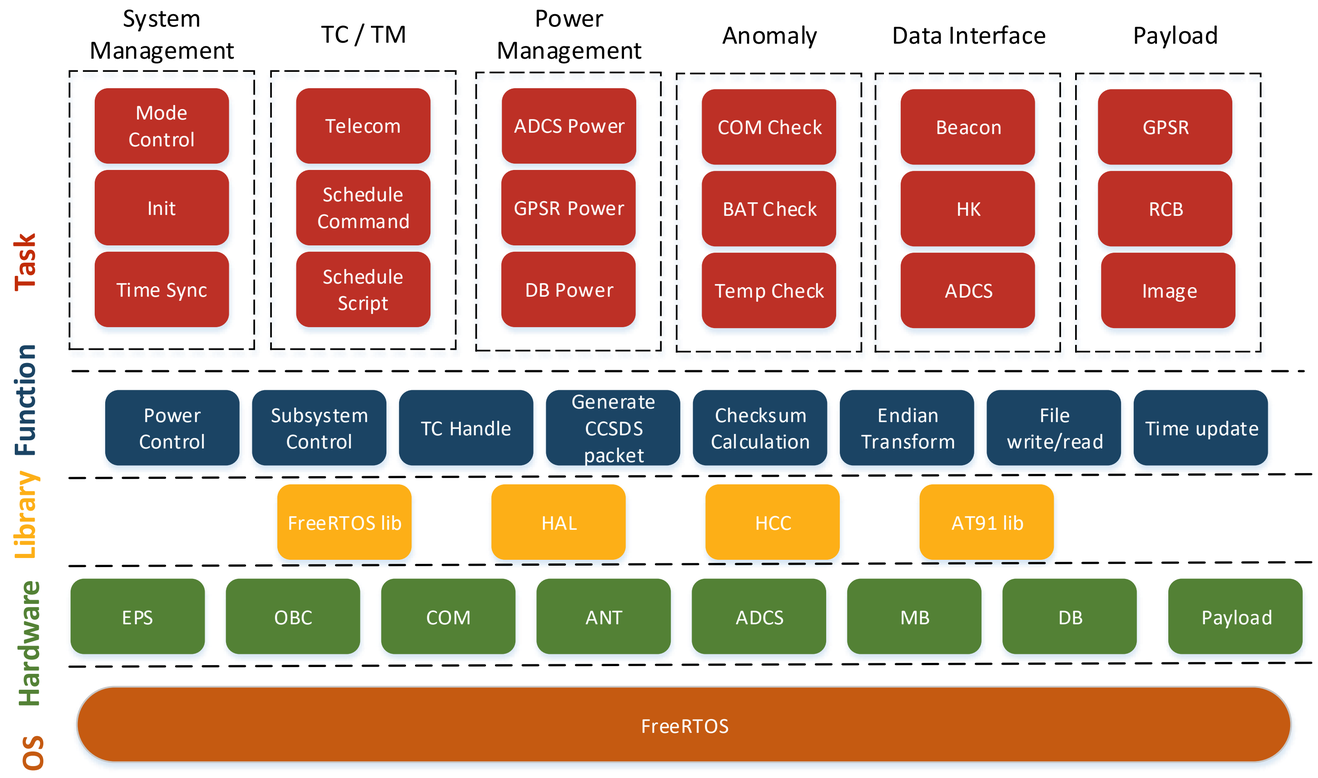
Robustness	The system should detect fault and recover from anomalies.
Modularity	The functionality can be easily added, modified or removed without affecting core of architecture.
Reusability	The software is able to migrate to different platform with least effort.
Autonomy	The system can do specific operation by itself.

04 Software Implementation

- Hardware: ISIS-iOBC (ARM-9, 32bit processor)
- Operating system: FreeRTOS
- Solution of the requirements:

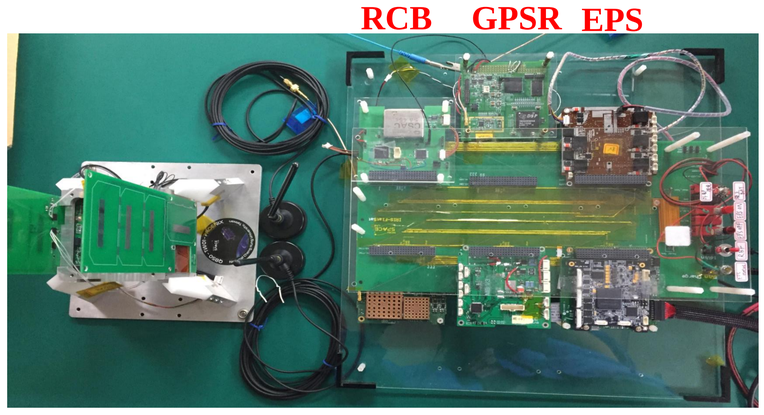
Robustness	FDIR mechanism (COM check, Temp check, BAT check)
Modularity	Task-oriented architecture
Reusability	Hierarchical architecture
Autonomy	Schedule telecommand, mission timeline

- Software architecture:

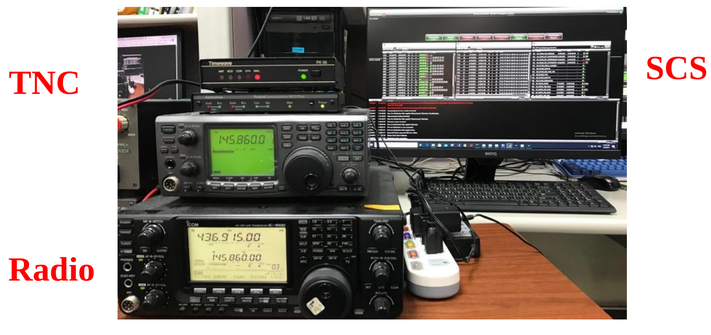


05 Software Verification

- Low-level test
  - > Test the basic function of the system
  - > Based on debugging console
- FlatSat platform
  - > Test interface communication
  - > Subsystem integration



- End –to –end communication
  - > Test communication with ground station
- Software test bed
  - > Use Arduino to simulate subsystem anomalies



06 Conclusion & Future Work

- Mission verification
  - > Real mode operation verification
- Software architecture tracking
- Code generation
  - > To reduce the effort of development