UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E ELETRÔNICA

Leonardo Kessler Slongo

NANOSATELLITE ELECTRICAL POWER SYSTEMS HARDWARE ARCHITECTURES: AN ANALYSIS ON ENERGY HARVESTING MAXIMIZATION THROUGH SCHEDULING ALGORITHM

Florianópolis

2017

Leonardo Kessler Slongo

NANOSATELLITE ELECTRICAL POWER SYSTEMS HARDWARE ARCHITECTURES: AN ANALYSIS ON ENERGY HARVESTING MAXIMIZATION THROUGH SCHEDULING ALGORITHM

Tese submetida ao Programa de Pós-Graduação
em Engenharia Elétrica para a obtenção
do Grau de Doutor em Engenharia.
Prof. Eduardo Augusto Bezerra, PhD
Universidade Federal de Santa Catarina
Orientador
Prof. Kleber Vieira de Paiva, Dr.
Universidade Federal de Santa Catarina
Co-orientador

Florianópolis

2017

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Slongo, Leonardo Kessler
Nanosatellite Electrical Power Systems Hardware Architectures: an analysis on energy harvesting maximization through scheduling algorithm / Leonardo Kessler Slongo ; orientador, Eduardo Augusto Bezerra, coorientador, Kleber Vieira de Paiva, 2017.
194 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós Graduação em Engenharia Elétrica, Florianópolis, 2017.

Inclui referências.

1. Engenharia Elétrica. 2. Nanosatélites. 3. Sistema de potência elétrica. 4. Algoritmo de escalonamento movido por energia. 5. Maximização de coleta de energia. I. Augusto Bezerra, Eduardo. II. Vieira de Paiva, Kleber. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Elétrica. IV. Título.

Leonardo Kessler Slongo

NANOSATELLITE ELECTRICAL POWER SYSTEMS HARDWARE ARCHITECTURES: AN ANALYSIS ON ENERGY HARVESTING MAXIMIZATION THROUGH SCHEDULING ALGORITHM

Esta Tese foi julgada aprovada para a obtenção do Título de "Doutor em Engenharia", e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica.

Florianópolis, 22 de Dezembro 2017.

Prof. Marcelo Lobo Heldwein, Dr. sc. ETH
Universidade Federal de Santa Catarina
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

---

Prof. Eduardo Augusto Bezerra, PhD
Universidade Federal de Santa Catarina
Orientador

Banca Examinadora:

Prof. Kleber Vieira de Paiva, Dr.
Universidade Federal de Santa Catarina
Co-orientador

Luigi Dilillo, PhD
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier

Prof. Fabian Luis Vargas, Dr
Pontifícia Universidade Católica do Rio Grande do Sul

Prof. César Augusto Missio Marcon, PhD
Pontifícia Universidade Católica do Rio Grande do Sul

Prof. Marcelo Lobo Heldwein, Dr. sc. ETH
Universidade Federal de Santa Catarina

Leonardo Kessler Slongo

NANOSATELLITE ELECTRICAL POWER SYSTEMS HARDWARE ARCHITECTURES: AN ANALYSIS ON ENERGY HARVESTING MAXIMIZATION THROUGH SCHEDULING ALGORITHM

This thesis is hereby approved and recommended for acceptance in partial fulfillment of the requirements for the degree of Doctor of Engineering in Electrical Engineering Graduate Program.

Florianópolis, 22^{nd} December 2017.

Prof. Marcelo Lobo Heldwein, Dr. sc. ETH
Federal University of Santa Catarina
Electrical Engineering Graduate Course Coordinator

Prof. Eduardo Augusto Bezerra, PhD
Thesis Adviser
Federal University of Santa Catarina

Examination Board:

Prof. Kleber Vieira de Paiva, Dr.
Federal University of Santa Catarina
Thesis Co-adviser

Luigi Dilillo, PhD
Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier

Prof. Fabian Luis Vargas, Dr
Pontifical Catholic University of Rio Grande do Sul

Prof. César Augusto Missio Marcon, PhD
Pontifical Catholic University of Rio Grande do Sul

Prof. Marcelo Lobo Heldwein, Dr. sc. ETH
Federal University of Santa Catarina

I dedicate this thesis to the Spirit of the living God, who teaches me all things...

ACKNOWLEDGMENTS

There is no better way to start this section but thanking the One who deserves all the honor and glory. I thank Jesus, not only for providing me what was needed to accomplish this thesis, but for giving me eternal life. I have no words to describe my gratitude to what He has done for me. He is the truly source of knowledge and wisdom, my Lord and Savior.

I would like to express my gratitude to the most special person on Earth, my wife Fernanda Roth Castellano Slongo, whose name stands now on my third manuscript. Throughout all this years she has been loyal, comprehensible, faithful, beautiful, merciful. She has always believed in me, and more important than all this, she has always believed in the Lord's plans for us. I would not be able to conclude this work (and many others in my life), if she was not there by me every single day, as she always did.

I sincerely thank my family, who brought me here. I am grateful to my father, Ademir Slongo, for his example of dedication, discipline, organization, hardworking, for his love and for all the support he provided along my life. I sincerely thank my mother, Marcia Kessler, for her immeasurable and unconditional love, for raising me in the Way, for her dedication, service and prayers. I thank my sister, Jaqueline Kessler Slongo, for her joy, her unexpected visits, for the conversations we have had, for minimizing my stressful moments with surf days, for her unceasing prayers. I thank my youngest sister Rafaela Slongo, who is a shining ray of joy, love and hope, which motivates me to be a good example to her. Also, I would like to thank the ones who, more recently, became part of my family as well: Rachel, Percilia, Rodrigo, Julie, Lorenzo, Enrico, Jorge and Marcela.

I thank those who always sustained me in prayers and advises: Rodrigo and Debora, Verner and Ticiane, Bruno and Luzi, Diego and Liliane, among many, who raise their shields and swords by my life and my house. I have no words to express my admiration and gratitude for their lovely and loyal friendship.

I thank professor Eduardo Bezerra, my adviser, who entrusted me to conduct this research and the responsibility of being part of the FloripaSat development team. I am grateful for the scientific meetings and for the valuable technical and personal advises. By the end of this work, I consider him not only as my adviser, but as a friend. I also thank professor Kleber Paiva, with whom I have been working for 12 years, for the opportunity to work again on the Microgravity project. I

express here my gratitude to him for trusting in my work. Once again, I have learned a lot from his dedication, effort, hard work, patience and love for science.

I sincerely thank Sara Martinez, Bruno Eiterer and Túlio Pereira, who have worked relentlessly to achieve the scientific and technological goals of this work. I thank them for their dedication and for executing what was beyond their responsibilities, for the sake of FloripaSat development. They have directly and extensively collaborated with this work, and they are undoubtedly co-authors of all the results presented in this manuscript. I also extend my gratitude to FloripaSat team, which was responsible for keeping an extraordinary work environment full of commitment, responsibility and enthusiasm.

I also thank professor Marcia Mantelli, who gave me the opportunity to work at Labtucal, and all the lab team that collaborated with my work on the Microgravity project. I specially thank Daniel Gaiki, João Gabriel and Pedro Seger, who directly collaborated with the Multi-Mission Platform development; Juan Pablo, Daniele Mangini and Luis Betancur for the partnership with the Microgravity experiments; and Luiz Domingos, Charles da Silva, Nelson Pabon, Luis Cisterna, for being always ready to lend a helping hand.

Finally, I would like to thank all Federal University of Santa Catarina staff and professors, for keeping and continuously building an extraordinary learning environment. They make this institution an example of success. One last but not least mention: I would like to thank Wilson Silva Costa for his dedication to the Electrical Engineering Graduation Program. His name is not frequently mentioned in acknowledgments, but his decades of dedication to the program allowed many excellent professionals to be formed. I am grateful to him for being always kind to me and ready to help with all the bureaucratic issues.

But the Comforter, which is the Holy Ghost,
whom the Father will send in my name,
he shall teach you all things, and bring all
things to your remembrance, whatsoever
I have said unto you.
John 14:26

Jesus Christ

RESUMO

Os nanossatélites tornaram-se uma oportunidade acessível para alcançar o espaço. São pequenos satélites com massa total que varia de 1 a 10kg, possuindo os subsistemas necessários (incluindo as cargas úteis) para satisfazer uma missão espacial. Esta tese concentra-se no sistema de potência elétrica (EPS) de um nanosatélite, uma placa de circuito impresso (PCB) que interage com fontes de energia (painéis solares), com unidades de armazenamento (baterias) e com outros subsistemas do satélite (Computador de bordo - OBDH; Sistema de comunicação - TT&C, Sistema de determinação e controle de atitude - ADCS; e cargas úteis). Um EPS ideal deve maximizar a extração de energia, armazenar de forma segura a energia e, finalmente, gerenciar eficientemente a distribuição de energia para outros subsistemas do satélite. Esta tese apresenta as quatro arquiteturas EPS mais adotadas para nanossatélites, comparando sua eficiência por meio de simulações e experimentos. Painéis solares, baterias e componentes do EPS foram analiticamente modelados e testados para projetar sistemas eficientes. Uma bancada de teste foi proposta para avaliar as arquiteturas EPS, emulando a irradiância solar e o consumo de energia do satélite. Além disso, a tese propõe um procedimento de teste de qualificação, que consiste em avaliar os subsistemas de nanosatélites em um vôo suborbital em um foguete de sondagem. Um sistema embarcado foi proposto para operar como uma interface entre os subsistemas do nanosatélite e a eletrônica do foguete. Os resultados da Missão Rio Verde são apresentados nesta tese, onde tanto o sistema embarcado de interface quanto o EPS do nanosatélite foram qualificados no espaço, em um voo do foguete VSB-30. Além disso, a tese propõe um algoritmo de escalonamento conduzido por energia, que permite melhorar a eficiência do EPS controlando a execução das tarefas do nanosatélite. Embora a arquitetura do circuito EPS seja o principal fator para a maximização da potência de entrada, a execução das tarefas do nanosatélite também afeta sua capacidade de captação de energia. Portanto, esta tese descreve um algoritmo de escalonamento que maximiza a entrada de energia do nanosatélite. O algoritmo foi testado para a arquitetura de Acoplamento Direto, melhorando sua capacidade de captação de energia em mais de 8 % para o teste de uma órbita. Finalmente, a tese aponta melhorias no algoritmo de escalonamento proposto, para também considerar efeitos de envelhecimento da bateria, previsão de entrada de energia, otimização

da prioridade das tarefas e dependência da arquitetura do EPS.

Palavras-chave: Nanosatélite; Sistema de potência elétrica; Teste de qualificação; Algoritmo de escalonamento movido por energia; Maximização de coleta de energia; Rastreador do ponto de máxima potência.

RESUMO EXPANDIDO

Introdução

Nanosatélites tornaram-se uma oportunidade acessível para se chegar ao espaço. Eles são pequenos satélites com massa total que varia de 1 a 10 Kg contendo todos os subsistemas necessários (incluindo cargas úteis) para satisfazer uma missão espacial. Por meio dos nanosatélites, as universidades podem permitir que seus alunos trabalhem em projetos reais de aplicação espacial. O baixo custo de lançamento dos nanosatélites e o curto tempo de desenvolvimento atraem qualquer entusiasta do espaço. Este crescimento no interesse por pequenos satélites cresceu ainda mais com a criação do padrão *CubeSat* em 1999. Desde então, muitas outras universidades e empresas ao redor do mundo têm trabalhado no desenvolvimento, teste, lançamento e rastreamento de nanosatélites (CORSO et al., 2011). Um nanosatélite pode ter um conjunto de subsistemas diferentes de acordo com o objetivo da missão. Os subsistemas mais comuns são os seguintes: Sistema de Energia Elétrica (EPS); Computador de Bordo (OBDH); Telemetria, Rastreamento e Comando (TT&C); Sistema de Determinação e Controle de Atitude (ADCS); antenas; painéis solares (ou outro dispositivo de conversão de energia); estrutura mecânica; baterias (ou outro dispositivo de armazenamento de energia) e cargas úteis. Embora alguns desses subsistemas sejam mencionados ao longo da tese, este trabalho se concentra no EPS. Um sistema de energia elétrica por satélite possui três funções principais: coletar energia, armazená-la e distribuí-la. O EPS é uma Placa de Circuito Impresso (PCB) que interage com fontes de energia (painéis solares, geradores termoelétricos, etc.), com unidades de armazenamento (baterias, supercapacitores, etc.) e com outros subsistemas de satélite (OBDH, TT &C, ADCS, cargas úteis, etc.). Um EPS ideal deve maximizar a captação de energia, armazenar com segurança a energia e, finalmente, gerenciar a distribuição de energia aos demais subsistemas do nanosatélite de uma forma eficiente. Os satélites executam muitas tarefas diferentes em órbita as quais devem ser organizadas para atingir os objetivos da missão. Uma vez que as tarefas a serem executadas podem ter diferentes prioridades, tempo de execução, recursos, etc., um algoritmo de escalonamento torna-se um elemento chave para uma missão de satélite bem sucedida. Este algoritmo de escalonamento deve definir qual e como as tarefas do satélite serão executadas. Esta tese demonstra como um algoritmo de escalonamento movido por energia

pode melhorar a capacidade de captação de energia de nanoatélites para uma arquitetura específica de hardware. O algoritmo de escalonamento deve ser projetado para ser ajustável aos requisitos de hardware e da missão. A tese apresenta portanto, diferentes modelos de arquiteturas EPS, simulações e uma discussão sobre esse assunto. Esta tese também levanta uma discussão interessante sobre como qualificar os módulos de nanosatélites. Este procedimento consiste em testar os subsistemas de nanosatélites a bordo de um foguete de sondagem. Finalmente, este trabalho apresenta uma discussão sobre melhorias que podem ser implementadas no algoritmo de escalonamento proposto. Além das modificações para arquiteturas EPS específicas, o algoritmo pode ser melhorado adicionando outras variáveis importantes como a previsão de potência de entrada, o efeito do envelhecimento da bateria e as prioridades de execução das tarefas.

Objetivos

O objetivo principal deste trabalho é melhorar a capacidade de captação de energia dos nanosatélites por meio de um algoritmo de escalonamento movido por energia. Alguns objetivos específicos foram definidos para alcançar o objetivo principal, a saber: Projetar diferentes arquiteturas de EPS para nanosatélites; modelar a entrada de energia do nanosatélite de acordo com sua posição em órbita; implementar o algoritmo de escalonamento movido por energia como um software embarcado modular que leva em consideração os atributos de hardware de cada arquitetura EPS; projetar uma bancada de testes para simular o consumo e a entrada de energia em órbita do nanosatélite; qualificar o EPS e o algoritmo proposto para aplicá-lo ao nanosatélite FloripaSat.

Metodologia

Quatro arquiteturas EPS foram projetadas para possibilitar a análise teórica e experimental de eficiência na captação e no gerenciamento de energia. Simulações em Simscape foram realizadas para a avaliação de desempenho das arquiteturas e comparação com os dados experimentais. Cada arquitetura foi testada separadamente em laboratório, emulando as condições de irradiância solar do espaço. Para isso, uma fonte luminosa constituída de LEDs de alta potência, controlado por uma fonte de corrente, foi utilizada. O controle da fonte de corrente foi estabelecido com base em um modelo de entrada de energia no nanosatélite, o qual é dependente da posição orbital. Tanto o modelo de posição orbital, quanto o modelo dos painéis solares são apresentados nesta tese. Além do teste em laboratório, uma das arquiteturas foi testada e qualificada em um voo suborbital, em um foguete de sondagem

(Missão Rio Verde). Este experimento possibilitou validar a metodologia de projeto das arquiteturas EPS. Por fim, um algoritmo de escalonamento foi proposto, com o objetivo de maximizar a captação de energia do nanosatélite ao gerenciar a execução das tarefas do mesmo. O algoritmo foi implementado e testado em laboratório, utilizando uma fonte controlada para emular o consumo de energia das tarefas em execução.

Resultados e Discussão

Esta tese apresentou uma visão geral sobre as arquiteturas EPS para nanosatélites. As arquiteturas mais adotadas foram modeladas, implementadas e testadas. A influência do hardware na operação dos painéis solares foi avaliada, procurando soluções para melhorar a capacidade de captação de energia dos nanosatélites. Um modelo preliminar de entrada de energia foi proposto, adotando o fator de forma de painéis solares e a posição do nanosatélite em órbita. Simulações foram validadas com experimentos realizados no laboratório. A partir dos modelos, simulações e resultados experimentais, pode-se concluir que a topologia do circuito EPS afeta fortemente a maximização e a eficiência da energia. Ainda que a arquitetura VLDO não possua um controle ativo de tensão dos painéis solares, foi classificada como a topologia mais eficiente. Ao longo da análise de arquiteturas EPS, este trabalho demonstrou que a capacidade de energia de nanosatélites pode ser melhorada ao se gerenciar a execução de suas tarefas. A discussão sobre as arquiteturas de hardware EPS esclareceu qual circuito pode se beneficiar das influências da execução das tarefas na operação do painel solar. Um algoritmo de escalonamento movido por energia foi proposto, melhorando em 8% a capacidade de energia captada pela arquitetura EPS Diretamente Acoplado para uma experiência de órbita única e quase 4,5% para uma experiência de três órbitas. Finalmente, um procedimento de qualificação foi proposto e executado, testando os subsistemas CubeSat em um foguete de sondagem. O modelo de engenharia do FloripaSat foi testado com sucesso no foguete VSB-30 ao longo da Missão Rio Verde, operado pelo Centro de Lançamentos de Alcântara, no Maranhão. Além dos subsistemas, uma plataforma de multimissão usada para testá-los foi qualificada. Este sistema integrado permite que uma variedade de experimentos sejam testados em foguetes suborbitais.

Considerações Finais

Os objetivos propostos para esta tese podem ser considerados alcançados. Os artigos publicados durante a pesquisa de doutorado (lista apresentada no final do manuscrito) confirmam o aspecto inovador e a re-

levância dos resultados apresentados neste trabalho. Importantes melhorias nesta linha de pesquisa foram apontadas ao longo do texto e certamente permitirão novos desafios técnicos e científicos para o grupo de pesquisa, para a universidade e para os parceiros de pesquisa.

Palavras-chave: Nanosatélite; Sistema de potência elétrica; Teste de qualificação; Algoritmo de escalonamento movido por energia; Maximização de coleta de energia; Rastreador do ponto de máxima potência.

ABSTRACT

Nanosatellites have become an affordable opportunity to reach the space. They are small satellites with total mass ranging from 1 to 10kg with the needed subsystems to satisfy a common satellite mission (including payloads). This thesis focuses on the nanosatellite's Electrical Power System, a printed circuit board (PCB) which interacts with power sources (solar panels), with storage units (batteries) and with other satellite's subsystems (On-Board Data Handling - OBDH; Telemetry, Tracking and Command - TT&C, Attitude Determination and Control System - ADCS; and payloads.). An ideal EPS should maximize energy extraction, safely store the energy and, finally, efficiently manage the energy distribution to other satellite's subsystems. This thesis presents four of the most adopted EPS architectures, comparing their efficiency through simulations and experiments. Solar panels, batteries and EPS components have been analytically modeled and tested in order to design efficient systems. A test stand is proposed to evaluate EPS architectures, emulating the solar irradiance and the satellite power consumption. Also, the thesis proposes a qualification test procedure, which consists of evaluating nanosatellite subsystems in a suborbital flight with a sounding rocket. An embedded system is proposed to operate as an interface between the nanosatellite subsystems and the rocket electronics. The Rio Verde Mission results are presented in this thesis, where both the embedded system interface and the nanosatellite's EPS have been space qualified through a VSB-30 flight. Besides this, the thesis proposes an energy-driven scheduling algorithm, which allows improving the EPS efficiency controlling the nanosatellite tasks execution. Although the EPS circuit architecture is the main factor for the input power maximization, the nanosatellite tasks execution also may affect its energy harvesting capability. Therefore, this thesis describes a scheduling algorithm that maximizes the nanosatellite power input. The algorithm has been tested for the Directly Coupled architecture, improving its energy harvesting capability in more than 8% for a single orbit experiment. Finally, the thesis points out improvements in the proposed energy-driven scheduling algorithm, which may consider also battery aging effects, energy input prediction, tasks priority optimization and EPS architecture dependence.

Keywords: Nanosatellite; Electrical Power System; Qualification test; Energy-driven scheduling algorithm; Energy harvesting maximization;

Maximum Power Point Tracker.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ADC *Analog to Digital Converter*

ADCS *Attitude Determination and Control System*

BCF *Battery Cycle Life*

COTS *Commercial Of The Shelf*

CR *Complete Rescheduling*

DoD *Depth-of-Discharge*

DVB-S2 *Digital Video Broadcasting – Satellite Second Generation*

EDA *Electronic Design Automation*

EEF *Energy-driven Scheduling Algorithm*

EOCV *End of Charge Voltage*

EOS *Earth Observing Satellites*

EPS *Electrical Power System*

ETSS *Enhanced Two Step Scheduler*

GEO *Geostationary Orbit*

GS *Ground Station*

LCS *Communications and Embedded Systems Laboratory*

LED *Light-Emitting Diode*

LEO *Low Earth Orbit*

MEO *Medium Earth Orbit*

MMP *Multi-Mission Platform*

MPPT *Maximum Power Point Tracker*

MTBF *Mean Time Between Failures*

MTEC *Mean Total Energy Consumption*

MTMF *Mean Time to Mission Failure*

NASA *National Aeronautics and Space Administration*

OBDH *On-Board Data Handling*

PCB *Printed Circuit Board*

P&O *Perturb and Observe*

PFmod *Proportional Fair queuing*

PL *Payload*

PR *Partial Rescheduling*

PWM *Pulse Width Modulation*

QoS *Quality of Service*

SAR *Synthetic Aperture Radar*

TCS *Thermal Control and Structure*

TDMA *Time Division Multiple Access*

TT&C *Telemetry, Tracking and Command*

UFSC *Federal University of Santa Catarina*

UnB *University of Brasília*

USART *Universal Synchronous/Asynchronous Receiver/Transmitter*

VI *Virtual Instrument*

VLDO *Very Low Dropout*

VV *Verification and Validation*

WSN *Wireless Sensor Network*

LIST OF SYMBOLS

V_{sp}	Solar panel voltage [V]
V_{bat}	Battery voltage [V]
V_{drop}	Voltage drop over components [V]
V_D	Diode voltage [V]
$V_{RS_{sp}}$	Solar panel shunt resistor voltage drop [V]
V_{BD}	Body diode voltage drop [V]
V_{DS}	MOSFET drain to source voltage drop [V]
I_D	Diode current [A]
I_S	Diode saturation current [A]
n	Diode ideality factor
V_T	Diode thermal voltage [V]
k	Boltzmann constant [J/K]
T	Temperature [K]
q	electron charge [C]
I_{sp}	Solar panel current [A]
R_{RSsp}	Solar panel shunt resistor [Ω]
V_{RSsp}	Solar panel shunt resistor voltage drop [V]
I_{bat}	Battery current [A]
V_{BD}	Body diode voltage [V]
n_{BD}	Body diode ideality factor
I_{SBD}	Body diode saturation current [A]
k'_n	MOSFET transconductance parameter [A/V^2]
I_{DS}	MOSFET drain to source current [A]
W/L	MOSFET channel aspect ratio
v_{GS}	MOSFET gate to source voltage [V]
V_{DS}	MOSFET drain to source voltage [V]
V_t	MOSFET threshold voltage [V]
R_{DSON}	On-state MOSFET drain to source resistance [Ω]
R_{RSbat}	Battery shunt resistor [Ω]
V_{RSbat}	Voltage drop over battery shunt resistor [V]
$V_{vldo_{in}}$	VLDO input voltage [V]
$V_{vldo_{out}}$	VLDO output voltage [V]

$I_{vldo_{out}}$	VLDO output current [A]
$P_{vldo_{diss}}$	VLDO dissipated power [W]
$R_{DSON_{VLDO}}$	On-state VLDO MOSFET drain to source resistance [Ω]
I_{Load}	Load's current [A]
V_{RSload}	Voltage drop over load shunt resistor [V]
V_{Load}	Voltage drop over the load [V]
V_L	Voltage drop over the inductor [V]
D	Boost converter duty cycle
R_{RSmppt}	MPPT integrated circuit shunt resistor [Ω]
VF_{bat}	Battery full state voltage [V]
VD_{bat}	Battery discharged voltage [V]
RD_s	Non-ideal diode series resistance [Ω]
$V_{RS_{bat}}$	Battery shunt resistor voltage drop [V]
I_c	Solar cell current [A]
V_c	Solar cell voltage [V]
I_{sc}	Solar cell short circuit current [A]
V_{oc}	Solar cell open circuit voltage [V]
I_{mpp}	Solar cell maximum power point current [A]
V_{mpp}	Solar cell maximum power point voltage [V]
I_{ph}	Solar cell photo generated current [A]
R_p	Solar cell parallel resistor [Ω]
R_s	Solar cell series resistor [Ω]
I_{R_p}	Solar cell parallel resistor current [A]
I_{R_s}	Solar cell series resistor current [A]
Δ_{Isc}	Coefficient for I_{sc} variation over temperature [A/K]
E	Solar irradiance [W/m^2]
E_0	Applied solar irradiance when obtaining I_{sc} [W/m^2]
T_c	Solar cell temperature [K]
T_0	Solar cell temperature when obtaining I_{sc} [K]
t_{orbit}	Period for the satellite to complete one orbit cycle [s]
t_{sh}	Period the satellite stays in shadow [s]
R	Earth's radius [km]
h	Satellite altitude [km]
G	Gravitational constant [N(m/kg)^2]

M	Earth's mass [kg]
A	Solar panel area [m^2]
η	Solar panel efficiency
β	Orbit angle from the Earth's Equatorial Plane [radian]
θ_s	Earth's shadow boundary orbit angle [radian]
α	Rotation form factor angle [radian]
θ	Satellite angular position in orbit [radian]
ϕ	Satellite rotation per minute
ψ	Rotation form factor coefficient
P_c	Solar panel power capability [W]

TABLE OF CONTENTS

1 INTRODUCTION

Nanosatellites have become an affordable opportunity to reach the space. They are small satellites with total mass ranging from 1 to 10kg with all the needed subsystems to satisfy a common satellite mission (including payloads). Through nanosatellites, universities may allow their students to work on real space application projects. Even small and medium size companies may have access to space technologies that until 20 years ago were mostly restricted to the National Space Agencies. Nanosatellites' launching "low price"and short development time attracts space enthusiasts. This growth in interest in small satellites was empowered by the CubeSat standard creation in 1999. Stanford University and California Polytechnic State University (Cal Poly) proposed a modular 10cm x 10cm x 11.35cm (1U) cubic shaped satellite intended for Low Earth Orbit and designed mostly with commercial off-the-shelf (COTS) components. Since then, many other universities an companies around the world have been working on nanosatellites' development, testing, launching and tracking (CORSO et al., 2011) (ALI et al., 2014) (The CubeSat Program, 2014).

A nanosatellite may have a set of different subsystems according to the mission goal. Most common subsystems are the following: Electrical Power System (EPS); On-Board Data Handling (OBDH); Telemetry, Tracking and Command (TT&C); Attitude Determination and Control System (ADCS); antennas; solar panels (or other energy conversion device); mechanical structure; batteries (or other energy storage device) and payloads. Although some of these subsystems are going to be mentioned along the thesis, this work focuses on the EPS. A satellite electrical power system has three main functions: energy harvesting, energy storage and energy distribution (Figure 1). The EPS is a printed circuit board (PCB) which interacts with power sources (solar panels, thermoelectric generators, etc.), with storage units (batteries, supercapacitors, etc.) and with other satellite's subsystems (OBDH, TT&C, ADCS, payloads, etc.).

An ideal EPS should maximize energy extraction, safely store the energy and, finally, efficiently manage the energy distribution to other satellite's subsystems. Mostly, these requirements conflict among each other or with other satellite subsystem's requirements. This work aims to solve this intricate requirements problem in an efficient and elegant way. Many other research groups have proposed efficient satellite EPSs, however, when considering nanosatellites, it is a completely

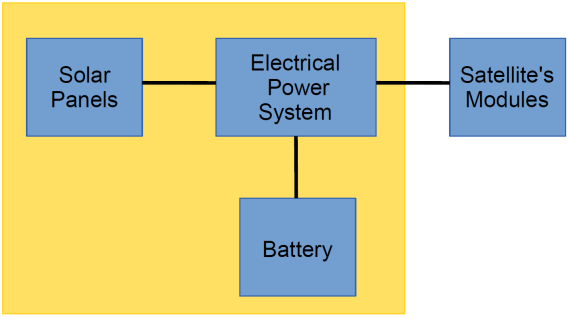


Figure 1: Electrical Power System overview.

distinct problem. Nanosatellites may be considered low power devices when compared to satellites with more than 100kg. Therefore, the energy harvesting maximization must be addressed differently. Also, the applications for LEO nanosatellites drastically differs from the medium earth orbit (MEO) and geostationary orbit (GEO) satellites' applications.

Satellites perform many different tasks in orbit and they shall be somehow organized in order to accomplish the satellite's goals. Since the tasks to be performed may have different priorities, execution time, resources, etc., a satellite scheduling algorithm may be a key element to achieve a successful satellite mission. Once the scheduling algorithm may define which (and how) the satellite's tasks are going to be executed, there shall be a relation between the algorithm and the EPS, after all, tasks execution demands energy.

The satellite scheduling problem is not new. It has been formulated with a variety of perspectives, with numerous proposed solutions (see Section 4.2 for some examples). Although there are distinct options for defining and solving the problem (DISHAN et al., 2013), (GAYTAN et al., 2015), (WANG et al., 2014), the goal is mostly the same: to optimize tasks execution from some perspective (maximize communication quality (CHRISTOPOULOS; CHATZINOTAS; OTTERSTEN, 2015), minimize system response time (KIM; CHANG, 2015), etc.). As described in Section 4.2, most of recent satellite scheduling algorithms are not designed for nanosatellites. Besides this, none of them are designed for energy harvesting maximization or nanosatellite lifetime extension. Chapter 4 demonstrates how an energy-driven scheduling algorithm may improve the nanosatellite energy harvesting capability for a specific EPS hardware architecture.

It is also important to mention that, besides the low power management issue, the scheduling algorithm shall consider the EPS hardware architecture. There are many different ways of harvesting energy in space. The same applies for the energy storage devices. Operating solar panels and recharging batteries, for example, allows lot of different technologies combinations and possibilities. Therefore, ideally, the scheduling algorithm shall be designed to be adjustable to hardware and mission requirements. Chapter 2 presents some EPS architectures models, simulations and a discussion on this issue.

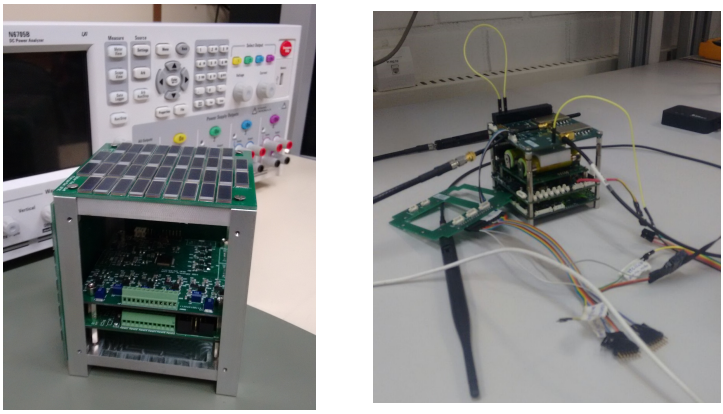
This thesis also raises an interesting discussion on how to qualify nanosatellite modules for flight. Unfortunately, due to their low rate of success, nanosatellites have been considered harmful for the space environment. Space debris are increasing and it is already a real problem. In order to decrease the nanosatellites fail rates, this thesis also proposes a qualification procedure for nanosatellite subsystems (Chapter 3). This procedure consists on testing the nanosatellite subsystems (engineering models) on-board a sounding rocket. This allows finding design fails on preliminary phases of the project, avoiding them to be propagated to the nanosatellite flight model.

Finally, this work presents a very fruitful discussion on improvements that may be implemented on the proposed scheduling algorithm (chapter 5). Besides the modifications for specific EPS architectures, the algorithm may be improved by adding other important variables as input power prediction, battery aging effect and tasks priority.

1.1 MOTIVATION

The main motivation for this work was to develop an efficient EPS for the nanosatellite projects ongoing at UFSC. At the end of 2013, a CubeSat project has started at the Communications and Embedded Systems Laboratory (LCS) at the Department of Electrical and Electronic Engineering - The FloripaSat. Soon, other UFSC departments joined the FloripaSat project: the Aerospace Engineering (UFSC Joinville), Mechatronics Engineering (UFSC Joinville), System's Control and Automation Engineering (UFSC Florianópolis) and Computer Science (UFSC Florianópolis). The project's main goal is to empower undergraduate and graduate students for working in the aerospace application field (SLONGO et al., 2016a).

The students have been organized in the following teams: Electrical Power System (EPS), On-Board Data Handling (OBDH), Tele-



(a) Prototype.

(b) Engineering model.

Figure 2: FloripaSat nanosatellite.

metry, Tracking and Command (TT&C), Attitude Determination and Control System (ADCS), Ground Station (GS), Verification and Validation (VV), Thermal Control and Structure (TCS) and Payloads (PL). Every team has a manager which realizes weekly meetings with the undergraduate students. Also, some general meetings are appointed allowing the students from different teams to interact with each other. Along the FloripaSat design process, four CubeSats models shall be obtained: prototype, engineering model 1, engineering model 2 and flight model. Figure 2 shows both the FloripaSat prototype (Figure 2a) and engineering model 1 (Figure 2b).

For the FloripaSat prototype, four different EPS hardware architectures have been proposed. Basically, they differ from each other on the method they operate the solar panels. Fabricating four different PCBs (EPS prototypes) allows an experimental analysis, which would guide the engineering and flight models designs. Besides this, the involved students would have the opportunity of working on the entire subsystem design (system conception, requirements, design decisions, hardware design, fabrication, electrical tests, software design, operational tests, etc.).

Besides the FloripaSat, the Federal University of Santa Catarina (UFSC) has joined the SERPENS I project, a 3U nanosatellite (Figure 3) developed by a consortium of Brazilian and foreign universities. The University of Brasília (UnB) has lead the SERPENS I project which was launched from the International Space Station in September 2015.

After a successful mission, UFSC has been commissioned to lead the SERPENS II project, which shall start in 2018.

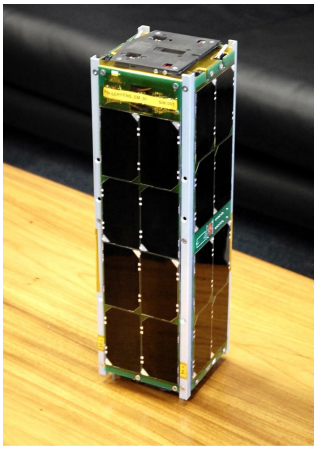


Figure 3: SERPENS I nanosatellite.
Image source: Valdivino Jr - Brazilian Space Agency (Brazilian Space, 2015).

The work presented in this thesis manuscript has been developed in this projects context, resulting on the design of the FloripaSat EPSs prototype, engineering model 1 and engineering model 2. All the knowledge acquired on these hardware and software designs may also be applicable in the next project, SERPENS II.

1.2 GOALS

The main goal of this work is to improve nanosatellites' energy harvesting capability. An energy-driven scheduling algorithm is proposed to achieve this. The following specific goals have been attended in order to accomplish the main goal:

1. To design nanosatellite Electrical Power Systems;
2. To model the nanosatellite power input according to its position in orbit;
3. To implement the energy-driven scheduling algorithm as an embedded modular software that takes into consideration the EPS hardware architecture.

4. To assemble a test stand to simulate the nanosatellite's in-orbit incoming energy and power consumption;

5. To qualify the EPS and the proposed scheduling algorithm in order to apply it to FloripaSat nanosatellite.

1.3 RESEARCH QUESTIONS AND CONTRIBUTIONS

The following are the questions which have guided this thesis to problem definitions and to their innovative solutions:

1. Is there any possibility of improving nanosatellite's energy harvesting performance by scheduling or controlling its tasks?

2. How the nanosatellite's tasks should be modeled in order to consider energy consumption?

3. Is the proposed scheduling algorithm an overhead for the nanosatellite EPS microcontroller?

4. How scheduling algorithms may be evaluated in therms of nanosatellite's energy harvesting maximization?

5. Does the EPS hardware architecture affects the scheduling algorithm strategy?

6. How does the EPS and other nanosatellite's subsystems may be tested to reduce the fail rates on this category of satellites?

These questions have been answered throughout this manuscript and have lead to the following contributions:

- Nanosatellite energy-driven scheduling algorithm to maximize solar energy harvesting for directly coupled EPS architecture: It is an embedded software, implemented in the C language, which is able to change the order and duration of the satellite executing tasks in order to maximize the EPS energy harvesting capability;

- EPS test stand to evaluate scheduling algorithms for different architectures: It is a platform consisting of equipments (power analyzer, voltage/current source, computer) and software blocks (implemented in LabVIEW and in C) in order to emulate the solar irradiance through high power LEDs and evaluate the EPS scheduling algorithm efficiency;

- FloripaSat EPS prototypes: Four different PCBs designed as distinct EPS architectures (directly coupled, very low dropout voltage regulator, maximum power point tracking with a dedicated integrated circuit, and maximum power point tracking architecture through a discrete boost regulator), which allowed the FloripaSat EPS team to experimentally understand the basics concepts of most well known EPS circuit designs.
- FloripaSat EPS engineering models: Two different versions (model 1 and model 2) of the FloripaSat EPS engineering model. These PCBs are the result of the EPS prototype analyses and improvements.
- FloripaSat energy aware software controlled task execution policy: It is the solution designed for the FloripaSat EPS engineering model 2, which considers the battery state of charge and the FloripaSat task sets to be triggered.
- Sounding rocket nanosatellite test procedure: a test procedure based on an embedded system designed to interface nanosatellite's subsystems and a suborbital rocket electronics. This embedded system as well as the FloripaSat engineering model main subsystems have been successfully tested on-board a VSB-30 rocket, proving the test procedure efficiency on qualifying nanosatellites PCBs on suborbital flights.

1.4 THESIS STRUCTURE

This manuscript contains six main chapters, including this introductory one. Three of the chapters are based on papers published/-submitted to international journals and conferences, which facilitates the understanding of each chapter by itself. Although this unconventional thesis writing approach has been adopted, the chapters have been smoothly structured in order to guarantee an easy reading and comprehension of all the proposed ideas and solutions as an unified work. Every chapter starts with an introduction, which places every individual piece of work in the research time line and in the state of the art. This brings to the thesis two important characteristics: Clarity in the research contributions presentation and an easier understanding of the analysis between the proposed/applied mathematical models and the experimental results, since they are tided together in every single chapter. The remaining of this manuscript is organized as follows:

- **Chapter 2** - This chapter discusses different Electrical Power System hardware architectures. It explains how each of them controls the solar panels, showing the benefits of each solution. Also, it presents mathematical models and simulations for each architecture. The models consider not only the EPS circuits, but the solar panels, the batteries, the satellite orbit position and power consumption. This allows a precise prediction of the nanosatellite energy harvesting, energy storage and energy distribution to other subsystems. The models are evaluated through experiments that emulate the nanosatellite in orbit.

- **Chapter 3** - This chapter shows an innovative qualification procedure proposed for nanosatellites using sounding rockets. After the tests performed at the laboratory (presented in Chapter 2) this is a necessary step to validate an embedded system for space applications. One of the four architectures presented in Chapter 2 was selected for this space qualification flight. The chapter discusses the importance of submitting the nanosatellite engineering model to a flight before finalizing the flight model design. An embedded system designed to interface the nanosatellite subsystems and the rocket electronics is shown. Also, the flight results for the FloripaSat engineering model (including the EPS described in Chapter 2) tested on-board the VSB-30 sounding rocket are presented.

- **Chapter 4** - This is the core chapter of this thesis where the energy-driven scheduling algorithm is presented. It discusses the performance of the algorithm on improving the EPS energy harvesting capability through tasks control. The algorithm is applied for a specific EPS architecture and, through simulations and experiments, it shows a considerable gain on energy harvesting when compared to the EPS operating without the algorithm. There is also a discussion on how to adapt the algorithm to other missions (other nanosatellites).

- **Chapter 5** - This chapter brings a discussion on further works for this research line. It presents additional parameters that may be considered for the energy-driven scheduling algorithms to improve even more the nanosatellite efficiency. Also, this chapter points out to the improvements for the FloripaSat EPS flight model that were discovered from the results of the sounding rocket mission.

- **Chapter 6** - This is the final chapter of this thesis where the

conclusion is addressed.

2 ELECTRICAL POWER SYSTEM ARCHITECTURES: MODELS, SIMULATIONS AND TESTS

2.1 INTRODUCTION

The Electrical Power System (EPS) is a printed circuit board (PCB) which shall provide energy to the remaining satellite subsystems. This energy shall be extracted from the space environment using some sort of energy conversion strategy. Different technologies and devices are available, as follows: solar panels may directly convert sunlight to electricity through the photo-voltaic effect; solar heat may be converted to operate some thermal power cycle; temperature gradients may be used by thermoelectric devices; solar heat may operate or catalyze chemical processes to produce or store energy (LIOR, 2001) (HARTY; OTTING; KUDIJA, 1994). Although there is a wide variety of options, solar panels are the most used devices to harvest energy to power satellites (GALATIS; GUO; BUURSINK, 2017).

The energy harvesting devices aforementioned normally produce an unstable power output. This may be related to the conversion physics or/and to the energy source instabilities. Although the Sun light is always present in the orbits around the Earth, satellites may not receive the same irradiance levels in all positions along the orbit. Therefore, most satellites have an energy storage device, which ensures to the subsystems the required power output stability. Then, besides managing to harvest energy, the EPS shall also control the energy storage charging/discharging process. As well as the conversion devices, there exist many technologies to store energy, like supercapacitors (SHIMIZU; UNDERWOOD, 2013) (WU et al., 2017), fuel cells (LEE; KIM, 2014) and batteries (HILL, 2011). Some satellites combine different storage devices technologies (i.e. battery and supercapacitor) in order to benefit from both characteristics: high discharge rate and high energy density.

The EPS shall not only harvest energy and store it. This subsystem must manage the satellite energy distribution, accordingly to the mission requirements. This means to provide power buses with voltage and current specifications to attend the subsystems need. Also, it ideally shall present a measurement unit to inform battery status and input power condition to the OBDH. This allows the satellite to efficiently execute its tasks, avoiding preempting them or missing their deadlines. Finally, an ideal EPS shall maximize the harvested energy, allowing the satellite to execute the largest number of tasks possible.

Therefore, this chapter is dedicated to comprehend the most adopted EPS hardware architectures for nanosatellites. Four different circuits are mathematically modeled, simulated, implemented and tested in order to compare their efficiency. This chapter also presents a test stand and a methodology to evaluate the EPS architectures at the laboratory.

2.2 STATE OF THE ART

The Electrical Power System may be considered one of the most important modules of a satellite. Without energy, no tasks may be performed by the spacecraft. Historically, more than 20% of the satellite failures are related to the EPS and the battery issues (CASTET; SALEH, 2010). Therefore, even it is not the most complex system of a satellite, it is worth desining it preciselly and efficiently. In addition to the safety issue, the more energy the satellite harvest and store, the more tasks it may perform in orbit.

The EPS design and efficiency evaluation problem is not new. Since the 1960s EPS models considering solar irradiance, solar panels, batteries, converters and loads have been proposed (BOMBERGER et al., 1963) (KIRPICH et al., 1963). However, most of them were planned for powering systems exceeding 100W. However, with the microelectronics development and with the emergence of the class of nanosatellites, the EPS designing challenge has changed. Low power - high efficiency converters, MPPT techniques, high efficiency solar cells and energy aware scheduling algorithms are some of the hot topics under research nowadays.

Aiming minimum space occupation and low power consumption, the EPS proposed for the AraMiS project has a boost regulator operate the solar panels close to their MPP (ALI et al., 2014). In order to reduce the number of components, the designers opted by the constant voltage MPPT algorithm, using only analog devices. The EPS steps up solar panels voltage from 4.4V to the distribution bus voltage level(14 ± 2V). It provides power buses with 3.3V and 5V voltage levels. The EPS efficiency analyzes is demonstrated, calculating the main components power losses. Although this is a very detailed and well described work, it does not compare the proposed solution with other EPS architectures.

A 3U EPS efficiency analyzis is presented in (GONZALEZ-LLORENTE et al., 2015). The subsystem is based on a buck converter, that steps down the solar panels voltage (15V or 7.5V) to the Lithium-Ion bat-

tery range of 3.3V or 6.6V. The authors define an optimum operation point for the dc-to-dc converter in order to achieve its maximum efficiency. The operation point is defined by the converter input voltage, output voltage and output power. Therefore, the discussion is based on different solar cells connections configurations, and battery cells arrangement, that result in the most efficient EPS configuration. Different scenarios have been analyzed to achieve 98% of peak efficiency on the best configuration: Input voltage of 7.4V, output voltage of 3.3V, output current varying from 0.1mA to 2.1A. Only the buck converter topology is discussed in this paper and it does not consider any MPPT technique.

An interesting solution adopted for the ERPSAT-1 picosatellite uses fuzzy logic to obtain energy harvesting maximization (NEJI et al., 2015). The developed subsystem has three main blocks, which are the fuzzy maximum power point tracker, the power conditioner, and the hierarchical fuzzy subsystem controller. The fuzzy logic algorithm has been implemented in a dedicated microcontroller that controls a boost converter. Simulations have been performed to demonstrate the better efficiency of the proposed method when compared to the Perturb and Observe MPPT algorithm. Although a comparison is provided in this work, it is restricted to simulation, and it considers only the algorithms tracking capability, without taking into consideration different hardware architectures possibilities.

Finally, a most recent work shows a design methodology to achieve high efficiency on the EPS topology based on a boost converter (PIOVESAN et al., 2016). The authors compare its boost converter optimal design to high efficiency commercial EPS options. Through their design methodology they achieved an efficiency of 97.467% for the case where $V_{in} = 5$V and $P_{out} = 2$W. Although the authors present an excellent result regarding the converter efficiency, the work does not discuss the energy input maximization. By changing the converter input and output conditions (when adopting a MPPT algorithm, for example), the converter efficiency shall not be the same. The ideal analysis shall take into consideration a global efficiency (input power maximization, power losses and delivered power to the load).

With the growth in the number of launches of nanosatellites, companies have started to design and commercialize subsystems. Two companies may be mentioned, specially for their advanced EPS architecture design: GOMspace and Clyde Space. The GOMspace offers the NanoPower P31u, able to power spacecrafts from 1W to 30W, based on a boost converter to operate the solar panels. The subsystem

has two Lithium-Ion batteries attached to it, with nominal voltage of 8V. It is provided with many features as: battery under-voltage and over-voltage protection, two regulated power buses (5V and 3.3V), onboard microcontroller with I^2C interface, a digital MPPT, among others (GOMSPACE, 2017). The Clyde Space 3G EPS is also provided with MPPT, battery under-voltage and over-voltage protection and it also has a dedicated onboard microcontroller with I^2C interface. On the other hand, it uses the concept of battery daughter boards, with no batteries attached to the EPS PCB. It has three regulated output buses (3.3V, 5V and 12V) (CLARK, 2014). Both GOMspace and Clyde Space EPS designs show the market trend of operating solar panels through boost regulators, added to a MPPT block. The drawback from both companies is to make not clear from the subsystem manual what is the MPPT control strategy and whether it allows parameters changes.

Although most of these works present their EPS efficiency analysis, none of them discusses and compares different hardware circuit architectures. Most analysis are presented after arbitrarily choosing an EPS architecture, without presenting reasons for that decision, and even worse, without a discussion that relates the EPS architecture, the design complexity, and mission requirements. Therefore, this chapter aims to cover this literature gap comparing the most adopted nanosatellite EPS hardware architectures and discussing the circuit influences on the subsystem overall efficiency.

2.3 FLORIPASAT EPS OVERVIEW

This work discusses four different EPS architectures in order to comprehend their particularities regarding the solar panel operation and energy management. However, before going to the circuits analyses, it is important to contextualize the EPS development. All four architectures have been designed in the context of FloripaSat project, an 1U CubeSat. The FloripaSat subsystems development has been planed in four phases: prototypes, engineering model 1, engineering model 2 and flight model. During the first phase (prototypes), four EPS different architectures have been proposed. The decision was to design, simulate and implement all of them in this preliminary phase, in order to compare the theoretical and experimental results. This decision has allowed to better comprehend the circuits in order to properly choose the solution for the FloripaSat project next phases.

The EPS architectures have many circuit blocks in common.

They are presented in this section, before moving for their particularities analysis. Figure 4 shows a block diagram of the directly coupled EPS, with the circuit blocks present in the remaining architectures shadowed.

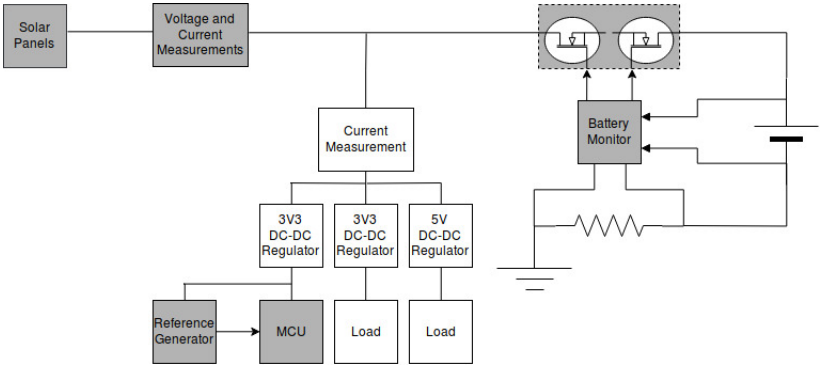


Figure 4: Directly coupled EPS block diagram.

All EPS architectures have been designed to operate six solar panels. Each EPS architecture has its own solar panels prototype set, because they differ in their maximum power point, as explained in Section 2.8. They all use the same solar cells (KXOB22-12X1L) (IXYS', 2016). Also, all solar panels contain the same number of solar cells (40 in total) in order to allow a fair energy harvesting capability comparison among the EPS architectures.

The FloripaSat CubeSat adopts the concept of distributed processing. All nanosatellite subsystems have their own microcontroller to manage their tasks. These microcontrollers communicate with each other through an I^2C bus in a master-slave mode. Therefore, all proposed EPS architectures have their own microcontroller (MSP430F6659). The MSP430 was chosen because of its low energy consumption and because it has already been successfully qualified in space by other nanosatellites' missions (Skycube, ZaCUBE-1 and Delfi-n3Xt). This microcontroller may actuate on some satellite dc-dc converters in order to enable/disable them. In critical situations (extremely low energy levels and/or OBDH failures condition), the EPS may disable the converters independently, managing the stored energy. However, in normal conditions, the EPS microcontroller works on slave mode.

Besides this, the EPS microcontroller is responsible for performing measurements, processing the collected data, and sending the

data to other satellite microcontrollers. All EPS architectures perform solar panels current measurement through a shunt resistor and a voltage amplification circuit. Also, the solar panels voltage are measured. Both solar panels current and voltage data is acquired by the microcontroller internal ADC.

Furthermore, the four EPS are also provided with a battery monitoring fuel gauge chip. This component plays an important role in the system. It is responsible for monitoring the CubeSat's battery status. Based on the information collected by this chip, the OBDH task scheduler may perform the decisions involving energy constraints. This chip communicates with the dedicated EPS microcontroller via the 1-Wire protocol. The information is than distributed to the other satellite's subsystems via the I^2C protocol.

Finally, the battery is also the same for all EPS architectures. One remark here is that the EPS MPPT with the boost regulator uses two batteries connected in series. The other three architectures have only one battery cell. It is a Lithium-Ion battery with nominal voltage of 3.7V and capacity of 3Ah (ICR18650-30A) (Samsung SDI Co. Ltd., 2007). Lithium-Ion batteries have been widely used in space applications due to their high charge density and low mass when compared to other battery technologies.

2.4 DIRECTLY COUPLED

The EPS directly coupled circuit is the simplest energy harvesting architecture. It requires few components, which reduces energy losses when compared to other architectures. Although it is known as "directly coupled", there are some protection components between the solar panels and the battery. However, these are passive components, which do not perform active control over the solar panels. Consequently, the solar panels may not continuously operate on their maximum power point. This is an interesting trade-off between design simplicity (and low power consumption) against no input power maximization (SLONGO et al., 2014).

Figure 5 shows the simplified EPS circuit diagram. This figure shows the architecture with only one solar panel, for the sake of simplicity. However, there are six solar panels, which are connected in parallel, in the point between the resistor RS_{sp} and the MOSFET M1. The diode connected to the solar panel avoids negative currents flowing through it when it is in shadow (the satellite is passing behind the Earth,

which blocks the Sun light). The RS_{sp} resistor is used for solar panel current measurement. The MOSFETs M1 and M2 compose the battery circuit protection, controlled by the battery monitoring chip. Finally, the RS_{bat} resistor is used to measure the battery current. Equations 2.1 and 2.2 show the solar panel dependency on battery voltage.

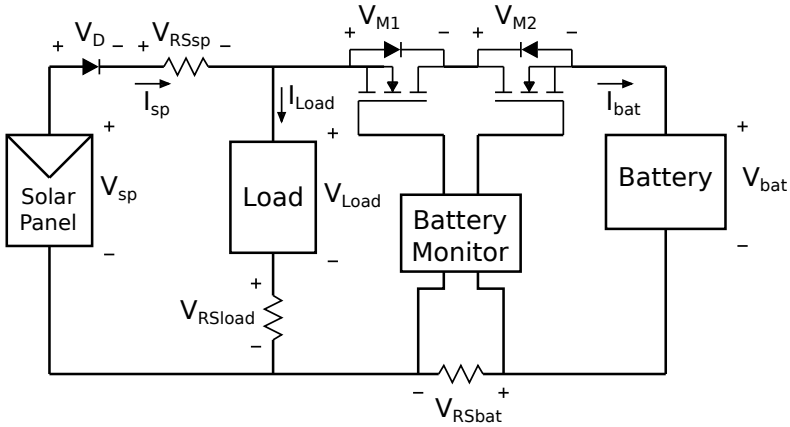


Figure 5: EPS simplified circuit diagram.

$$V_{sp} = V_{bat} + V_{drop} \tag{2.1}$$

where V_{sp} is the solar panel voltage, V_{bat} is the battery voltage and V_{drop} is the voltage drop over the components between the solar panel and the battery.

$$V_{drop} = V_D + V_{RSsp} + V_{BD} + V_{DS} + V_{RSbat} \tag{2.2}$$

where V_D is the diode voltage drop, V_{RSsp} is the solar panel shunt resistor voltage drop, $V_{BD} = V_{M1}$ is the body diode voltage drop of the first battery protection MOSFET, $V_{DS} = V_{M2}$ is the second battery protection MOSFET drain to source voltage drop and finally V_{RSbat} is the battery shunt resistor voltage drop. The voltage drop over the diode, the shunt resistors and the MOSFETs are current dependent. Thus, every right term of the Equation 2.2 may be redefined, starting from V_D:

$$I_D = I_S \left(e^{\left(\frac{V_D}{nV_T}\right)} - 1 \right) \tag{2.3}$$

where I_D is the diode current in forward bias polarity, I_S is the saturation current, n is the ideality factor ranging between 1 and 2 and V_T is the thermal voltage defined by Equation 2.4.

$$V_T = \frac{kT}{q} \tag{2.4}$$

where k is the Boltzmann constant, T is the temperature in Kelvin and q is the electron charge. Considering $I_D >> I_S$ Equation 2.3 may be simplified to Equation 2.5:

$$I_D \simeq I_S e^{\frac{V_D}{nV_T}} \tag{2.5}$$

From Equation 2.3 V_D may be obtained, as shown in Equation 2.6:

$$V_D = nV_T ln\frac{I_D}{I_S} \tag{2.6}$$

However, I_D is equal to the solar panel current I_{sp}. Finally the following equation is obtained:

$$V_D = nV_T ln\frac{I_{sp}}{I_S} \tag{2.7}$$

The voltage over the shunt resistor is directly obtained from Ohm's Law as shown in Equation 2.8, where R_{RSsp} is the shunt resistor through which flows the solar panel current I_{sp}.

$$V_{RSsp} = R_{RSsp} I_{sp} \tag{2.8}$$

For the circuit under study two n-channel MOSFET operate as switches to protect the battery against over current, overvoltage and undervoltage through the battery monitoring IC (Figure 5). Similarly to the V_D, the V_{BD} may be defined as Equation 2.9. It is important to note that the MOSFET body diode not necessarily has the same fabrication parameters as the aforementioned diode. This implies in a different n (n_{BD}) and I_S (I_{SBD}). Also, the current flowing through this diode is not the same. From Figure 5 it is clear that the body diode current is I_{bat}. Thus:

$$V_{BD} = n_{BD} V_T ln\frac{I_{bat}}{I_{SBD}} \tag{2.9}$$

Furthermore, the MOSFET's drain to source current I_{DS} is given by Equation 2.10.

$$I_{DS} = k'_n \frac{W}{L} \left[(v_{GS} - V_t) V_{DS} - \frac{1}{2} V_{DS}^2 \right] \tag{2.10}$$

where $k'_n = \mu_n C_{ox}$ is the transconductance parameter determined by the fabrication technology, W/L is the channel aspect ratio, v_{GS} is the gate to source voltage, V_{DS} is the drain to source voltage and V_t is the threshold voltage. Assuming that V_{DS} is considerably small, the term V_{DS}^2 may be neglected resulting in Equation 2.11:

$$I_{DS} \simeq k'_n \frac{W}{L} (v_{GS} - V_t) V_{DS} \tag{2.11}$$

This equation shows a linear relation between I_{DS} and V_{DS}. Rewriting Equation 2.11 the R_{DSON} may be defined, which is a common parameter found in MOSFET datasheets:

$$R_{DSON} = \frac{V_{DS}}{I_{DS}} = \frac{1}{k'_n \frac{W}{L} (v_{GS} - V_t)} \tag{2.12}$$

Therefore:

$$V_{DS} = R_{DSON} I_{DS} = \frac{I_{DS}}{k'_n \frac{W}{L} (v_{GS} - V_t)} = \frac{I_{bat}}{k'_n \frac{W}{L} (v_{GS} - V_t)} \tag{2.13}$$

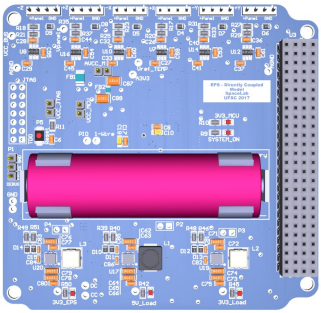
The voltage over the battery shunt resistor is also directly obtained from Ohm's Law as shown in Equation 2.14, where R_{RSbat} is the shunt resistor through which flows the battery current I_{bat}.

$$V_{RSbat} = R_{RSbat} I_{bat} \tag{2.14}$$

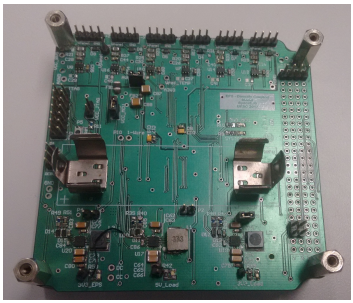
Finally, Equation 2.2 may be rewritten from Equations 2.7, 2.8, 2.9, 2.13 and 2.14 as follows:

$$\begin{aligned} V_{drop}(I_{sp}, I_{bat}) = {} & nV_T ln \frac{I_{sp}}{I_S} + R_{RSsp} I_{sp} + n_{BD} V_T ln \frac{I_{bat}}{I_{SBD}} \\ & + \frac{I_{bat}}{k'_n \frac{W}{L} (v_{GS} - V_t)} + R_{RSbat} I_{bat} \end{aligned} \tag{2.15}$$

Equations 2.2 and 2.1 shows that the solar panel voltage depends on battery voltage V_{bat}. This requires a good matching between the solar panel maximum power point voltage and the battery nominal voltage (see Section 2.8). Solar panel voltage also depends on battery



(a) Prototype 3D image.



(b) Prototype photo.

Figure 6: Directly coupled EPS prototype.

current I_{bat} and solar panel current I_{sp}. The solar panel current I_{sp} depends on temperature, solar irradiance and on its operating point (see Section 2.8). Therefore, this circuit presents a variable input power efficiency, which ultimately depends the satellite position in orbit (irradiance and temperature) and on battery conditions. Chapter 4 shows how this battery voltage dependence may be used to maximize the solar panels power input by controlling the satellite tasks execution. Figure 6 shows a 3D image of the EPS directly coupled architecture PCB as well as a picture of it.

2.5 VERY LOW DROPOUT VOLTAGE REGULATOR

This EPS architecture has a high efficiency linear regulator between the solar panel and the load. This figure shows the architecture with only one solar panel, for the sake of simplicity. However, there are six solar panels, which are connected in parallel, in the point between the resistor RS_{sp} and the linear regulator input. From the solar panel point of view, it may not be considered a controllable circuit. However, it changes the relation between the battery and the solar panel voltage when compared to the directly coupled circuit. Figure 7 shows a simplified circuit diagram of the very low dropout (VLDO) voltage regulator EPS architecture. The VLDO used in this circuit is a commercial regulator (TEXAS INSTRUMENTS, 2015b). The integrated circuit simplified internal block diagram is presented in Figure 8.

Differently from the voltage drop analysis proposed in Section 2.4, this analysis starts from the VLDO regulator dissipated power expres-

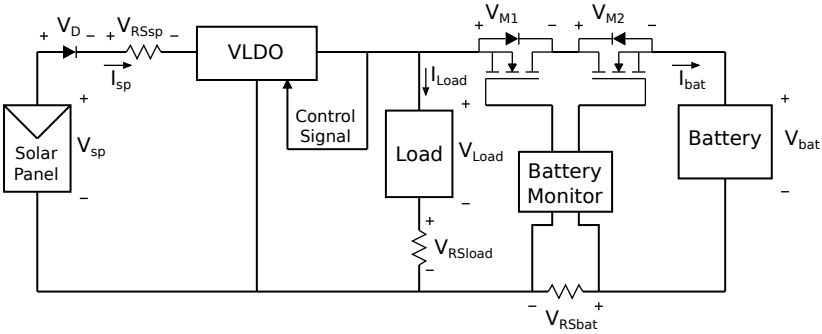


Figure 7: EPS VLDO regulator simplified circuit diagram.

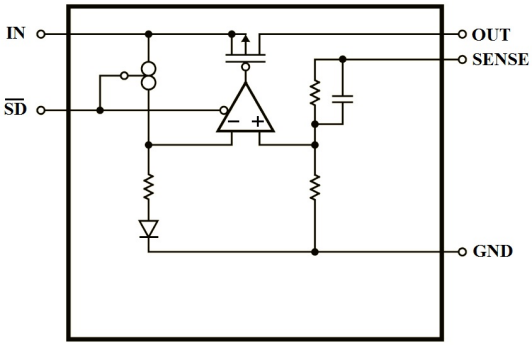


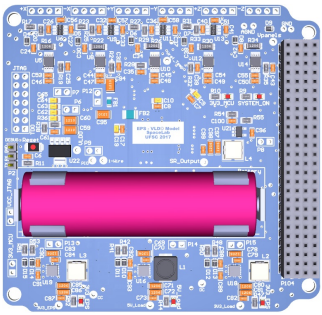
Figure 8: VLDO regulator simplified internal block diagram.
Image source: Component datasheet (TEXAS INSTRUMENTS, 2015b).

sion (Equation 2.16). Equation 2.16, may be rewritten to Equation 2.17 using the circuit parameters shown in Figure 7.

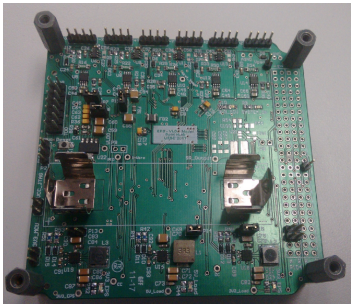
$$Pvldo_{diss} = (Vvldo_{in} - Vvldo_{out})Ivldo_{out} \tag{2.16}$$

$$\begin{aligned} Pvldo_{diss} = (V_{sp} - V_D - V_{RSsp})(I_{bat} + I_{Load}) \\ -(V_{BD} + V_{DS} + V_{bat} + V_{RSbat})(I_{bat} + I_{Load}) \end{aligned} \tag{2.17}$$

However, $Pvldo_{diss}$ may be described by Equation 2.18. Rewriting Equation 2.18 one may obtain V_{sp}, through Equation 2.19.



(a) Prototype 3D image.



(b) Prototype photo.

Figure 9: VLDO EPS prototype.

$$\begin{aligned} R_{DSON_{VLDO}} I_{sp}^2 = (V_{sp} - V_D - V_{RSsp})(I_{bat} + I_{Load}) \\ -(V_{BD} + V_{DS} + V_{bat} + V_{RSbat})(I_{bat} + I_{Load}) \end{aligned} \tag{2.18}$$

$$V_{sp} = \frac{R_{DSON_{VLDO}} I_{sp}^2}{I_{bat} + I_{Load}} + V_D + V_{RSsp} + V_{BD} + V_{DS} + V_{bat} + V_{RSbat} \tag{2.19}$$

Although Equation 2.19 is a long expression, one important conclusion may be stated: The load current (I_{Load}) strongly affects the solar panel voltage. This fact may be used to control the solar panel operation point by allowing the satellite to executing more or less tasks. Figure 9 shows a 3D image of the VLDO EPS prototype architecture PCB as well as a picture of it.

2.6 MAXIMUM POWER POINT TRACKER - BOOST REGULATOR

From all architectures presented in this work, this one is supposed to present the best MPP tracking performance. It is provided with a boost regulator between the solar panel and the load. Figure 10 shows the EPS boost regulator simplified circuit diagram. In this figure, only one power input channel is shown, for the sake of simplicity. However, there is one boost controller for each pair of solar panels connected in parallel. The solar panels connection occurs between the RS_{sp} and the

boost regulator. This results in three boost regulators controlling six solar panels. The regulator voltage output is imposed by the battery voltage. Therefore, by changing the PWM duty cycle, the solar panel voltage changes. This is the principle to control the solar panel voltage, in order to keep it as close as possible to the V_{mpp}.

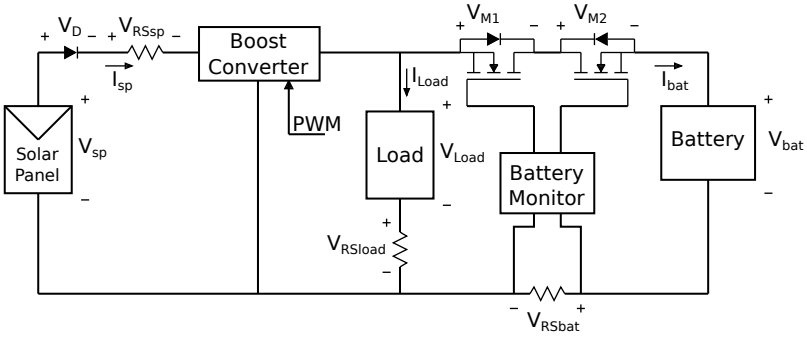


Figure 10: EPS MPPT boost regulator simplified circuit diagram.

This is the only circuit presented in this work that has two Lithium-Ion cells connected in series. Therefore, the converter must step up the solar panel voltage. For the sake of simplicity, the voltage drops over the diode, the solar panel shunt resistor, the MOSFETs and the battery shunt resistor will be neglected. Later in this analysis, they will be re-introduced into the final equation that describes the solar panel voltage and the circuit parameters.

The circuit of Figure 11 will be analyzed in two stages: first considering the switch S closed, and than considering it opened. For this, the inductor average voltage is considered to be null. Figure 12 shows the boost circuit in its first analysis stage, where the switch S is considered closed and the inductor voltage V_L equals the solar panel voltage V_{sp} (Equation 2.20).

$$V_L = V_{sp} \tag{2.20}$$

For the second stage analysis, the switch S is opened. The energy stored in the inductor than is discharged, flowing through the diode, and causing the V_L to change its polarity. Figure 13 shows the second stage circuit analysis, considering ideal components (diode conducting as a wire). In this case the inductor voltage is described by Equation 2.21.

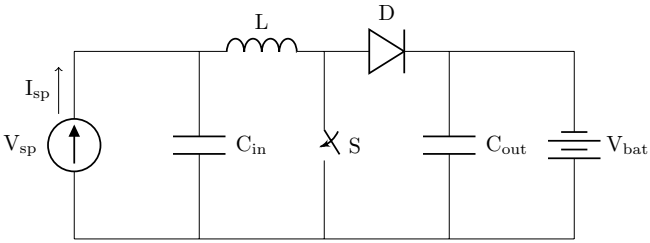


Figure 11: Boost regulator simplified circuit diagram.

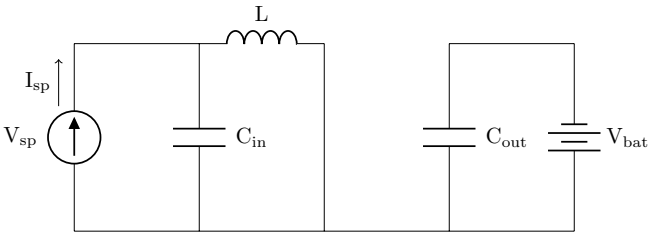


Figure 12: Boost regulator first stage analysis.

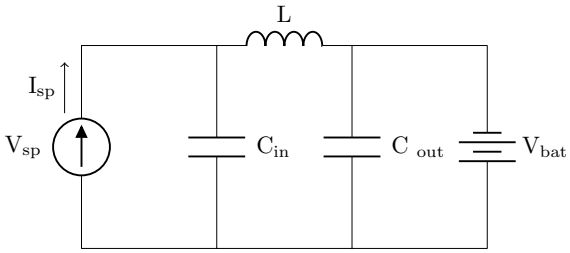


Figure 13: Boost regulator second stage analysis.

$$V_L = V_{sp} - V_{bat} \tag{2.21}$$

As presented in Figure 10, the switch S is controlled by a PWM signal, generated by the EPS microcontroller. This signal has a duty cycle D, ranging from 0 to 1, representing the percentage of time for which switch S stays closed. This PWM signal has an operating frequency f_D, and consequently a period T_D. Therefore, one may define the period for which the switch S stays closed as DT_D and the period for which it stays opened as $(1 - D)T_D$. Considering the circuit

two stages analysis, the average inductor voltage may be described by Equation 2.22.

$$V_L = \frac{1}{T_D}\left(\int_0^{DT_D} V_{sp}\,dt + \int_{DT_D}^{T_D} V_{sp} - V_{bat}\,dt\right) \tag{2.22}$$

Solving Equation 2.22 through equations 2.23 and 2.24, one may obtain the Equation 2.25.

$$V_L = \frac{1}{T_D}\left[V_{sp}\cdot DT_D + (V_{sp} - V_{bat})(T_D - DT_D)\right] \tag{2.23}$$

$$V_L = V_{sp}\cdot D + V_{sp} - V_{sp}\cdot D - V_{bat} + V_{bat}\cdot D \tag{2.24}$$

$$V_L = V_{sp} + V_{bat}(D-1) \tag{2.25}$$

As the inductor average voltage V_L is considered null along the period T, Equation 2.25 implies in Equation 2.26, which leads to Equation 2.27.

$$V_{sp} + V_{bat}(D-1) = 0 \tag{2.26}$$

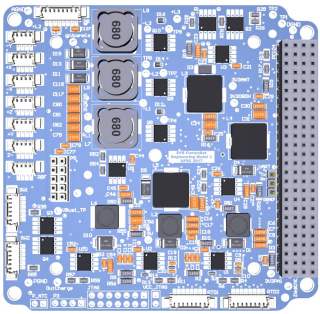
$$V_{sp} = V_{bat}(1-D) \tag{2.27}$$

From Equation 2.27 one may introduce the voltages drop over the diode (V_D), the solar panel shunt resistor (V_{RSsp}), the MOSFETs (V_{BD} and V_{DS}) and the battery shunt resistor (V_{RSbat}) presented in Figure 10. Also, recalling Equation 2.2, the Equation 2.27 may be redefined as Equation 2.28 and finally into Equation 2.29.

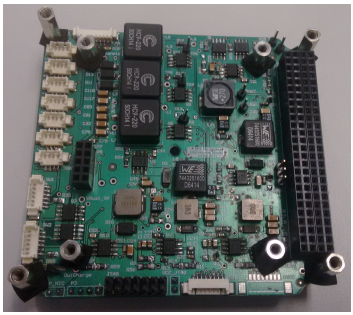
$$V_{sp} - V_D - V_{RSsp} = (V_{bat} + V_{RSbat} + V_{BD} + V_{DS})(1-D) \tag{2.28}$$

$$V_{sp} = V_D + V_{RSsp} + (V_{bat} + V_{RSbat} + V_{BD} + V_{DS})(1-D) \tag{2.29}$$

Equation 2.29 could be used as the solar panel control rule in order to establish the V_{mpp}. However, this is a relative complex equation to be executed in real time. Therefore, a simpler strategy is adopted



(a) MPPT boost EPS 3D image.



(b) MPPT boost EPS photo.

Figure 14: MPPT boost EPS.

to control the solar panel voltage, which is the Perturb and Observe MPPT algorithm. This formulation, however, is useful to the simulation implementation discussed in Section 2.10.

Another important conclusion from this mathematical description is that, although the battery voltage appears on Equation 2.29, the parameter D allows controlling the solar panel voltage for different values of V_{bat}. Therefor, when battery voltage changes, the microcontroller may actuate changing D, keeping the solar panel voltage on MPP, independently from battery voltage. Figure 14 shows a 3D image of the MPPT boost EPS architecture PCB as well as a picture of it.

2.7 MAXIMUM POWER POINT TRACKER - INTEGRATED CIRCUIT

This circuit has many similarities with the MPPT boost regulator EPS architecture. It also adopts a boost converter between the solar panel and the load. As well as for the architecture presented above, the solar panels are controlled in pairs (two solar panels connected in parallel and controlled by one boost regulator). However, this converter is built-in an integrated circuit (STMICROELECTRONICS, 2017). More than a converter, the chip implements the Perturb and Observe algorithm in hardware, to operate the solar panels on their MPP. Figure 15 shows the simplified circuit diagram of this architecture.

The formulation for this architecture is exactly the same as the one already presented in Section 2.6. The only remark here is that the PWM control is not performed externally by the EPS microcontrol-

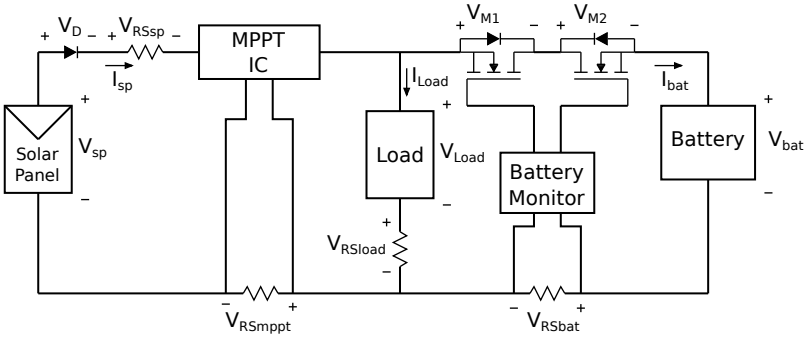


Figure 15: EPS MPPT IC simplified circuit diagram.

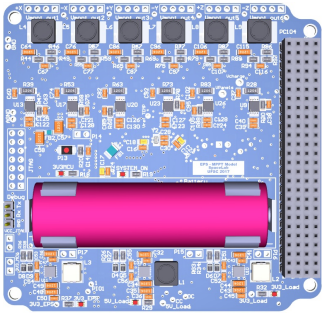
ler, as in the other MPPT architecture. The solar power calculation is performed inside the MPPT chip, by measuring the solar panel voltage (internal ADC) and current (through the resistor shunt V_{RSmppt}). Also, the perturbation caused on the solar panel voltage is determined by hardware, in the chip. This solution reduces the EPS microcontroller computation needs, however, it does not allow any modification/improvement on the control strategy.

Since the integrated chip is a boost regulator, the solar panels operating voltage shall be lower than for the other architectures. Therefore, a 4 by 10 has been implemented (see Section 2.8.1). Also, as any other boost regulator, its efficiency varies with its power output level. Due to the regulator power losses, it has been proved that it is not efficient for ultra low power applications (∼100mW output power range) (FR0HLICH; BEZERRA; SLONGO, 2015). However, for the CubeSat power consumption range, this may be considered an appropriate option to reduce the PCB design complexity and the microcontroller computational effort. Figure 16 shows a 3D image of the MPPT integrated circuit EPS prototype architecture PCB as well as a picture of it.

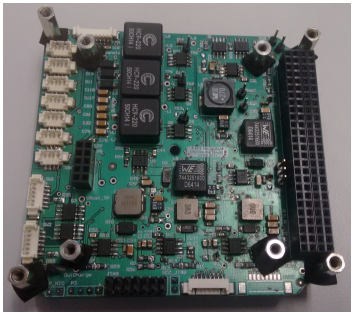
2.8 INPUT POWER MODEL AND SIMULATIONS

2.8.1 Solar panel

All circuits proposed in this chapter indirectly or directly operate solar panels. Therefore, a solar panel mathematical model is presen-



(a) Prototype 3D image.



(b) Prototype photo.

Figure 16: MPPT integrated circuit EPS prototype.

ted in this section (ERDEM; ERDEM, 2013) (BELLIA; YOUCEF; FATIMA, 2014). This model allows comprehending the EPS power input behavior as well as guides the EPS solar panel control circuit design. The solar panel is made of solar cells, connected in series or/and in parallel. Therefore, the presented model firstly focuses on describing a singles cell, and latter on modeling their different connection combinations.

An ideal solar cell can be modeled by a diode (D) in parallel with a current source (I). A more realistic model includes a parallel (R_p) and a series (R_s) resistors to consider non-idealistic power losses. The solar cell equivalent circuit is shown in Figure 17, where I_{ph} is the photo generated current, I_D is the diode current, I_{R_p} is the parallel resistor current, I_c is the solar cell current and V_c is the solar cell voltage. From the circuit, Equation 2.30 may be defined.

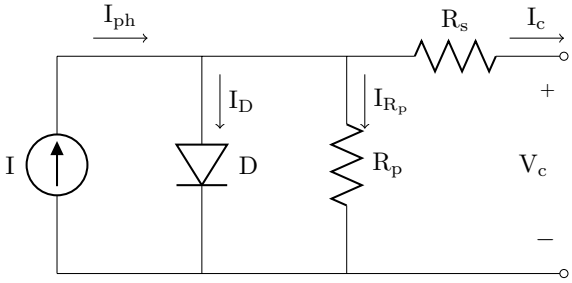


Figure 17: Solar cell equivalent circuit.

$$I_c = I_{ph} - I_D - I_{R_p} \tag{2.30}$$

The current I_{ph} may be obtained from the solar panel short circuit current I_{sc}, for a given solar irradiance E and solar cell temperature T_c, as described in Equation 2.31.

$$I_{ph} = I_{sc}\frac{E}{E_0}[1 + \Delta_{I_{sc}}(T_c - T_0)] \tag{2.31}$$

Where E_0 is the solar irradiance for which I_{sc} has been obtained (normally 1000 W/m^2), T_0 is the temperature for which I_{sc} has been obtained (normally 25°C), $\Delta_{I_{sc}}$ is the coefficient that expresses the I_{sc} variation with temperature. All these parameters may be obtained from the solar cell datasheet.

The diode current may be expressed by Equation 2.32, where I_S is the saturation current, q is the electron charge, n is the diode ideality factor, k is the Boltzmann constant and T_c is the solar cell temperature (KASAP, 2006) (BELLIA; YOUCEF; FATIMA, 2014).

$$I_D = I_S\left(e^{\frac{-q(V_c + I_c R_s)}{nkT_c}} - 1\right) \tag{2.32}$$

Analyzing the circuit of Figure 17 one may obtain the parallel resistor current I_{R_p}, which is defined in Equation 2.33.

$$I_{R_p} = \frac{V_c + I_c R_s}{R_p} \tag{2.33}$$

Combining equations 2.31, 2.32, 2.33 with Equation 2.30, the solar cell current-voltage relation may be described in therms of circuit parameters, as described in Equation 2.34.

$$I_c = I_{sc}\frac{E}{E_0}[1 + \Delta_{I_{sc}}(T_c - T_0)] - I_S\left(e^{\frac{-q(V_c + I_c R_s)}{nkT_c}} - 1\right) - \frac{V + I_c R_s}{R_p} \tag{2.34}$$

In the proposed model, Equation 2.34 has been implemented in MATLAB to generate the solar cell I-V current for a given temperature and irradiation. This model may be expanded to express the solar panel I-V, by combining multiple solar cells. Since every proposed EPS architecture has a different strategy to operate the solar panels, different combinations of solar cells have been implemented. Every

circuit architecture proposed in this chapter have been tested with a dedicated prototype solar panel. Both the EPS as the solar panels were designed aiming to harvest the maximum amount of energy as possible. Table 1 resumes the solar panels configuration for each EPS architecture, where V_{oc} is the solar cell open circuit voltage; I_{sc} is the solar cell short circuit current; V_{mpp} is the solar cell maximum power point voltage and I_{mpp} is the maximum power point current. The cells connections are described with the notation $a \times b$, which means a cells connected in series with b branches connected in parallel.

Table 1: Solar panel configuration for each EPS architecture.

EPS	**Cells**	$\boldsymbol{V_{oc}}$	$\boldsymbol{I_{sc}}$	$\boldsymbol{V_{mpp}}$	$\boldsymbol{I_{mpp}}$
DC	8 x 5	5.04V	250mA	4.0V	223.0mA
VLDO	10 x 4	6.30V	200mA	5.0V	178.4mA
MPPT chip	4 x 10	2.52V	500mA	2.0V	446.0mA
MPPT boost	10 x 4	6.30V	200mA	5.0V	178.4mA

The solar panels are made from commercial off-the-shelf solar cells (SolarBITs, from IXYS' - model KXOB22-12X1F) (IXYS', 2016). The usage of these simple and relatively cheap solar cells allows quick and inexpensive circuit validation. The solar panel is designed to connect SolarBITs in series and/or parallel to perfectly meet the circuits requirements, aiming input energy maximization (see Figure 18). All solar panels have 40 solar cells, therefore, the power generation capability is the same for all of them.

After mathematically describing the solar panel I-V relation, the circuit parameters must be defined in order to simulate the solar panel behavior. An important remark is that the circuit parameters change according to the solar panel temperature and irradiance level. Therefore, using only one set of components to model the solar panel for all temperature and irradiance levels leads to an imprecise model. In the other hand, to calculate circuit parameters for all the possible irradiance and temperature values is impracticable. Therefore, a layered technique is proposed, dividing the operating solar irradiance range in three zones, for which the circuit parameters are obtained. The irradiance zones were defined based on the test stand capability to generate power. Three irradiation boundaries have been defined for each solar panel model, as shown in Tables 2 to 4. This values have been defined in order to cover the irradiance level used during the experiments. The irradiance level is based on the orbit model presented in Section 2.8.2.

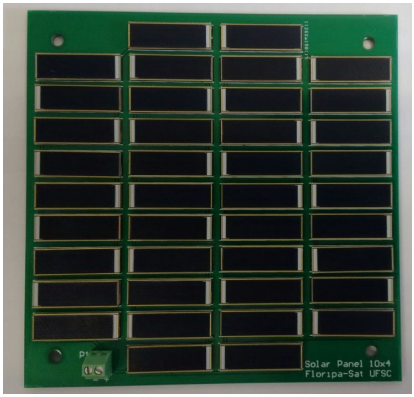


Figure 18: Solar panel prototype photo.

Although the satellite will receive more energy in space, the power level provided by the test stand is enough to validate the circuits and compare their efficiency. Then, for each solar panel, three sets of components were defined. Tables 2 to 4 summarize the circuit parameters for each irradiation zone.

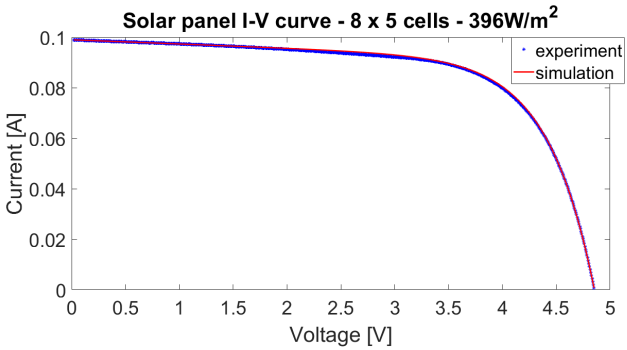


Figure 19: Solar panel I-V - 8 by 5 cells - 396W/m^2.

The components values have been obtained by fitting the experimental curve using the Equation 2.34. The experimental curves have been generated by submitting the solar panels to a known irradiance level (measured with a pyranometer) and sweeping the curve I-V with an electronic load (four quadrant voltage/current source - see Section 2.11). Figures 19 to 27 compares the experimental and simula-

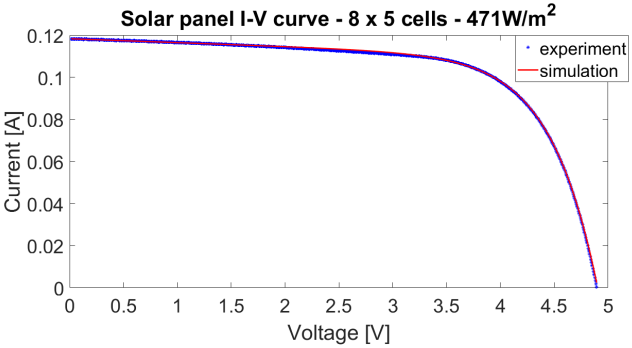


Figure 20: Solar panel I-V - 8 by 5 cells - 471W/m^2.

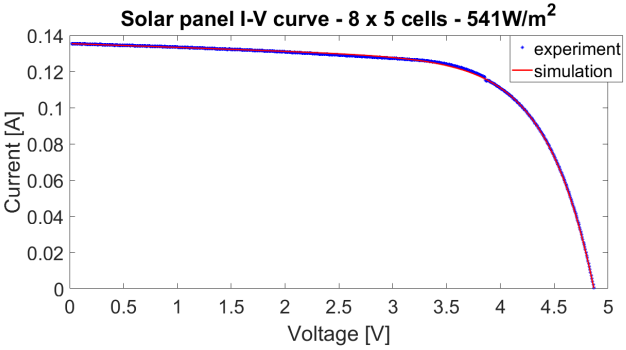


Figure 21: Solar panel I-V - 8 by 5 cells - 541W/m^2.

ted I-V curves for the different solar panels, under different irradiance conditions.

Table 2: Solar panel 8 x 5 circuit parameters.

Irradiance[W/m^2]	$R_s[\Omega]$	$R_p[\Omega]$	n	I_S[nA]
396	0.15	378.802	2	136
471	0.15	339.602	2	145
541	0.15	300.402	2	181

Figures 28 and 29 shows examples of solar panel power-voltage curves with irradiance and temperature variations. From the plots one may note that the power curves present a maximum value, which is

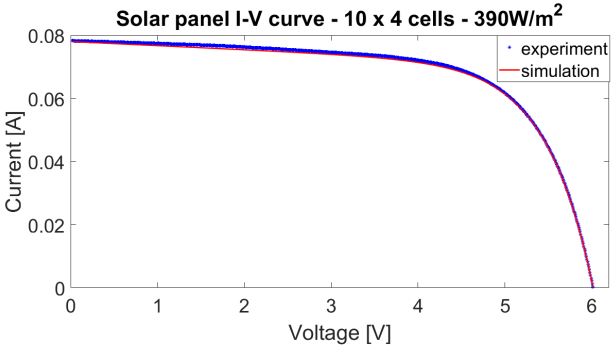


Figure 22: Solar panel I-V - 10 by 4 cells - 390W/m^2.

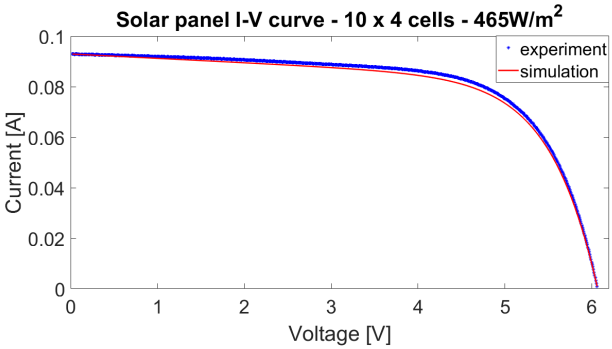


Figure 23: Solar panel I-V - 10 by 4 cells - 465W/m^2.

Table 3: Solar panel 4 x 10 circuit parameters.

Irradiance[W/m^2]	$R_s[\Omega]$	$R_p[\Omega]$	n	I_S[nA]
414	1.90999	559	2	145
490	1.90999	519.8	2	172
564	1.90999	486.602	2	190

known as the maximum power point (MPP). For each MPP there exist a voltage named maximum power point voltage (V_{mpp}) and a current named I_{mpp}. Therefore, one of the features of a well designed EPS is to operate the solar panel as close as possible to its MPP. The control technique to operate a device on its maximum power point

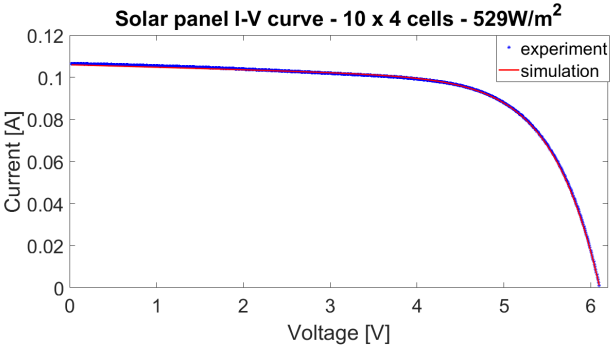


Figure 24: Solar panel I-V - 10 by 4 cells - 529W/m^2.

Table 4: Solar panel 10 x 4 circuit parameters.

Irradiance[W/m^2]	$R_s[\Omega]$	$R_p[\Omega]$	n	I_S[nA]
390	0.15	310.202	2	145
465	0.15	231.802	2	154
529	0.15	339.602	2	172

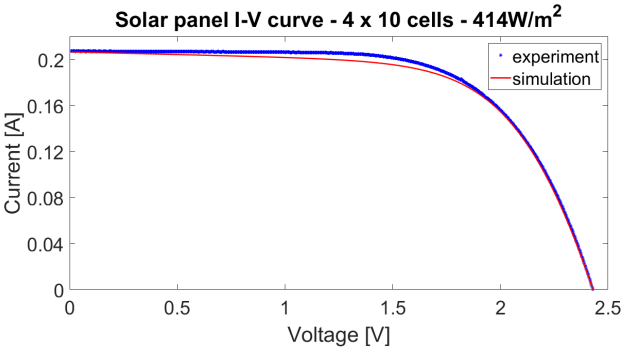


Figure 25: Solar panel I-V - 4 by 10 cells - 414W/m^2.

is called maximum power point tracker $MPPT$. Numerous solutions have been proposed in order to operate low power solar panels on their MPP (RAWY et al., 2017) (LOPEZ-LAPENA; PENELLA, 2012) (ARMBRUSTER et al., 2017). In this thesis the perturb and observe (P & O) technique has been adopted (see Section 2.7). Although it is known

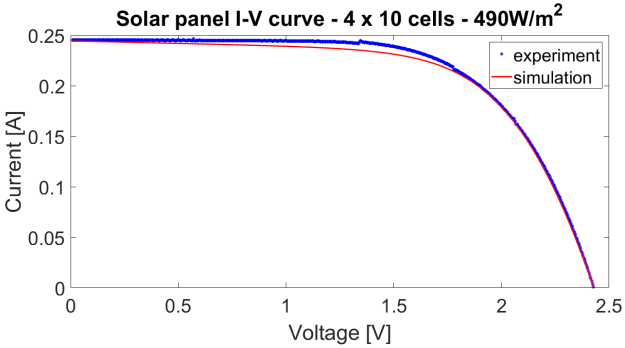


Figure 26: Solar panel I-V - 4 by 10 cells - 490W/m^2.

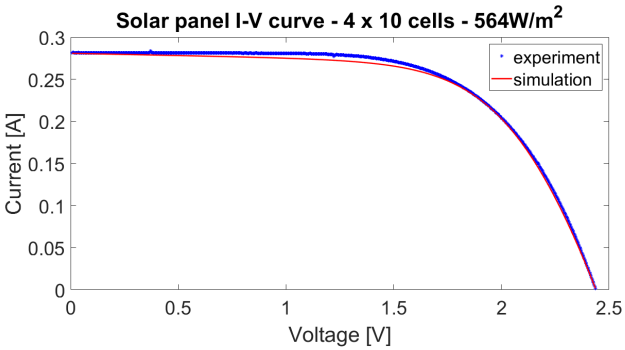


Figure 27: Solar panel I-V - 4 by 10 cells - 564W/m^2.

from the literature that this is not the most efficient technique, it was chosen due to its high cost-benefit ratio regarding precision and implementation simplicity (SUNDARESWARAN et al., 2016) (KILLI; SAMANTA, 2015).

2.8.2 Satellite orbit

The amount of energy that a satellite is able to extract from space determines the electrical load it is capable to supply. Considering the sunlight as the nanosatellite energy source, the factors that mostly affect the power input are the solar panel area, the solar panel efficiency and, the satellite orbit. These parameters are considered on a model

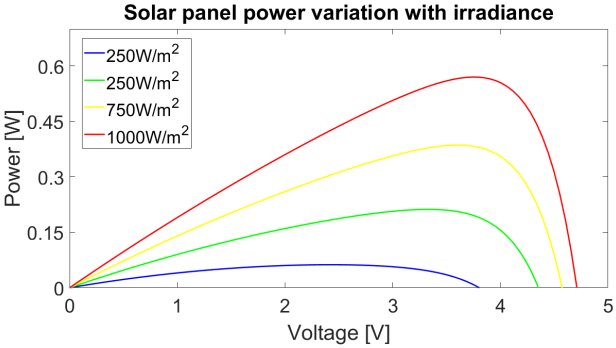


Figure 28: P-V curves with irradiance variation.

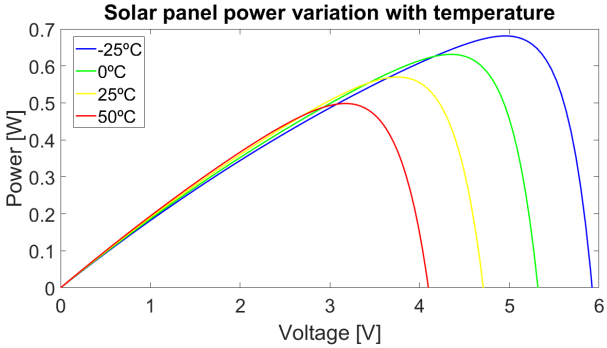


Figure 29: P-V curves with temperature variation.

that estimates the CubeSat's instantaneous available power.

The first parameter to be considered is the orbit altitude, which is part of the mission definition. For the FloripaSat project the assumed altitude is 310 km. Second parameter, also determined by the launcher, is the Sun-orbit-plane angle (β). When this angle is equal to zero, the period that the satellite stays in the Earth shadow achieves its maximum and the satellite shall orbit in the Equatorial Plane. Table 5 shows the parameters for the orbit period calculation (Equation 2.35). Equation 2.36 allows an estimation of the period the satellite stays in the Earth shadow. Some approximations were adopted for this calculation: the satellite orbit is considered as a circle and the Earth shape as a perfect sphere (ABOUEL-FOTOUH et al., 2006) (MOCANU et al., 2009) (JACOBSEN, 2011).

$$t_{orbit} = 2\pi(R+h)\sqrt{\frac{(R+h)}{GM}} \tag{2.35}$$

$$t_{sh} = \frac{t_{orbit}}{\pi} arccos\left(\frac{\sqrt{1-\left(\frac{R}{R+h}\right)^2}}{cos(\beta)}\right) \tag{2.36}$$

Table 5: Orbit simulation parameters.

Symbol	**Parameter**	**Value**	**Unit**
G	Gravitational constant	6.67×10^{-11}	$N(m/kg)^2$
M	Earth's mass	5.97219×10^{24}	kg
R	Earth's radius	6,371	km
h	Satellite altitude	310	km
β	Orbit angle	0	rad

From the equations 2.35 and 2.36 and from the values presented in Table 5 the orbit and the shadow period were determined: $t_{orbit} = 5434.67s = 90.58min$ and $t_{sh} = 2188.29s = 36.47min$. In order to estimate the CubeSat power input, the orbit was divided into zones, as described in Figure 30. This division was based on the CubeSat angular position in orbit (θ). Using a form factor, it is possible to calculate the percentage of the solar panels illuminated by the sun for each θ, and consequently, the nanosatellite instantaneous input power $P(\theta)$. The zones have been defined according to the orbit angular position. There are two angles to define the position that the satellite stays in shadow: θ_s and $2\pi - \theta_s$. The angle θ_s is defined in Equation 2.37.

$$\theta_s = arcsin\left(\frac{R}{R+h}\right) \tag{2.37}$$

The proposed orbit model also considers the rotation around the satellite z axis. The model assumes that the attitude control system is able to keep one satellite face always pointing to the Earth. This adds two more parameters to the model: the rotation form factor angle α and the satellite rotation per minute ϕ. Both parameters α and ϕ strongly depends on the $ADCS$ control strategy. Although α could assume values from 0 to 2π, the form factor repeats every $\pi/4$ step. Figure 31 shows a sequence of 8 CubeSat counter clockwise movements around 'z'

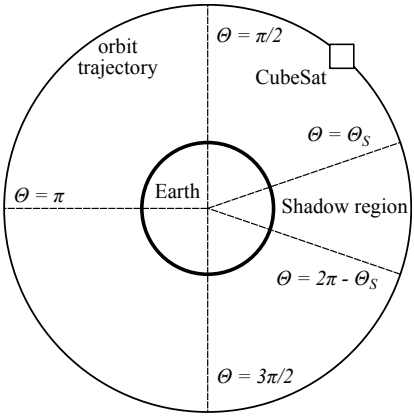


Figure 30: CubeSat orbit zones.

axis, with a step of 15 degrees between them. From this figure, one may note that the solar panel area hit by the solar irradiance in positions 1 and 7 is equal. The same occurs for positions 3 and 5, and also for positions 2, 6 and 8. Therefore, a rotation form factor coefficient ψ (ranging from 0 to 1) is adopted to redefine α, considering the form factor periodicity effect (Equation 2.38).

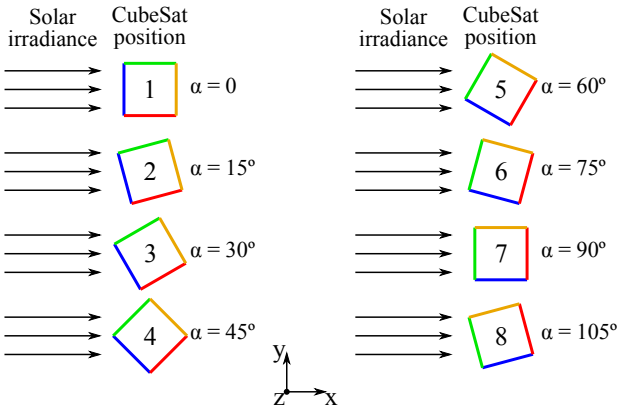


Figure 31: CubeSat rotation around 'z' axis.

$$\alpha = \begin{cases} \psi 2\pi & \text{if } 0 < \psi \leq 0.125 \\ \pi/4 - (\psi - 0.125)2\pi & \text{if } 0.125 < \psi \leq 0.25 \\ (\psi - 0.25)2\pi & \text{if } 0.25 < \psi \leq 0.375 \\ \pi/4 - (\psi - 0.375)2\pi & \text{if } 0.375 < \psi \leq 0.5 \\ (\psi - 0.5)2\pi & \text{if } 0.5 < \psi \leq 0.625 \\ \pi/4 - (\psi - 0.625)2\pi & \text{if } 0.625 < \psi \leq 0.75 \\ (\psi - 0.75)2\pi & \text{if } 0.75 < \psi \leq 0.875 \\ \pi/4 - (\psi - 0.875)2\pi & \text{if } 0.875 < \psi \leq 1 \end{cases} \tag{2.38}$$

After defining α, geometry equations were adopted to mathematically express the solar panels area illuminated by the Sun (Equation 2.39). These equations are valid for θ ranging from 0 to 2π. Table 6 shows the parameters to compute the nanosatellite instantaneous power input. These parameters depends on the solar panels characteristics and on the satellite position in orbit. The solar panels characteristics are known from the solar cells datasheet and from the solar panel mechanical dimensions. Figure 18 shows a picture of a FloripaSat solar panel prototype.

$$P(\theta, \alpha) = \begin{cases} 0 & \text{if } 0 < \theta \leq \theta_s \\ 0 & \text{if } (2\pi - \theta_s) < \theta \leq 2\pi \\ sin(\theta)(P_c sin(\alpha) & \\ +P_c sin(\pi/2 - \alpha)) & \text{if } \theta_s < \theta \leq \pi/2 \\ sin(\theta)(P_c sin(\alpha) & \\ +P_c sin(\pi/2 - \alpha)) - P_c cos(\theta) & \text{if } \pi/2 < \theta \leq \pi \\ -sin(\theta)(P_c sin(\alpha) & \\ +P_c sin(\pi/2 - \alpha)) - P_c cos(\theta) & \text{if } \pi < \theta \leq 3\pi/2 \\ -sin(theta)(P_c sin(\alpha) & \\ +P_c sin(\pi/2 - \alpha)) & \text{if } 3\pi/2 < \theta \leq (2\pi - \theta_s) \end{cases} \tag{2.39}$$

Where P_c is the solar panel power capability, defined by Equation 2.40.

$$P_c = \eta A E \tag{2.40}$$

Where η is the solar panel efficiency and E is the solar irradiance in space.

Table 6: Solar panel simulation parameters.

Symbol	**Parameter**	**Value**	**Unit**
A	Solar panel area	616×10^{-5}	m^2
E	Solar irradiance	$1,353$	W/m^2
η	Solar panel efficiency	0.186	-
α	Rotation angle	0 to 2π	radian
ϕ	Satellite RPM	*variable*	-

Finally, it is important to mention that when $\alpha = \pi/4$ and $\phi = 0$ the energy input is maximized (see the green curve in Figure 32). However, this would demand an extremely precise control strategy. Figures 32 to 34 show the satellite input power simulation for three different conditions:

- $\phi = 0$ and $\alpha = \pi/4$: no rotation around z axis, optimum α to maximize energy input;
- $\phi = 0$ and $\alpha = 0$: no rotation around z axis, α assuming its worst value;
- $\phi = 1$ and $\alpha = varying$: 1 rotation per minute around z axis and α assuming all possible values due to the satellite rotation around the satellite z axis;

All plots from figures 32 to 34 have their abscissas expressed in degrees, assuming values that may range from 0 to 360. This occurs due to the assumed circular movement of the satellite around the Earth. In order to express the occurrence of multiple orbits, the plots on this thesis may have their abscissas expressed in seconds or minutes. The angle to time conversion is based on Equation 2.35 (that calculates the orbit time) and on the assumption that the satellite has a constant angular velocity along the orbit. Then, using a simple linear conversion, it is possible to relate angle position with time along the satellite orbit. Figure 35 is an example of this angle to time conversion to express a magnitude varying over three orbits.

Figure 32 shows the effect of the satellite flight dynamics in power generation. All three different flight conditions present two discontinuity points on their power curve, which occurs when the satellite

enters/leaves the shadow zone. This discontinuity is caused by the model approximation that does not consider the penumbra region, which would result in a smother transition from shadow to illuminated zone. From Figures 33 and 34 one may note that the higher the satellite rotation around its own axis, the more lobes appear on the curve. Although this model is an approximation for the satellite flight dynamics, it is coherent with recent and more complex formulation (LEE et al., 2015). Besides this, the focus of this work is to compare the EPS architectures' efficiency, therefore, the power input approximation facilitates the experiments implementation and does prejudice the comparative analysis.

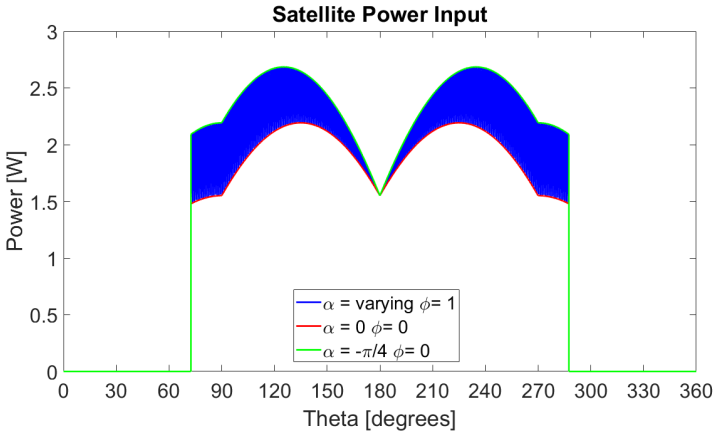


Figure 32: Satellite power input for different flight dynamic conditions.

Table 7: Satellite input power for different dynamic flight conditions.

$\boldsymbol{\alpha}$	$\boldsymbol{\phi}$	$\boldsymbol{P_{orb_{avg}}}$	$\boldsymbol{P_{orb_{max}}}$
varying	1	1.3108W	2.6850W
0	0	1.1355W	2.1923W
$\pi/4$	0	1.4014W	2.6850W

Table 7 shows the average ($P_{orb_{avg}}$) and the maximum power ($P_{orb_{max}}$) along the orbit for the three different flight dynamic conditions. The worst case scenario has been adopted to emulate the irradiance behavior experimentally. The condition where $\phi = 0$ and $\alpha = 0$

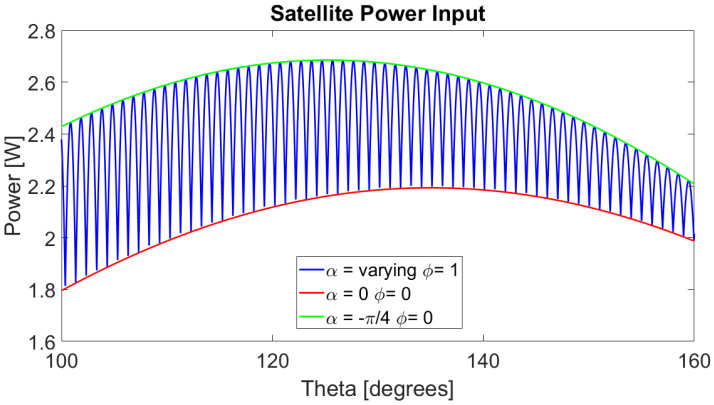


Figure 33: Satellite power input plot zoomed.

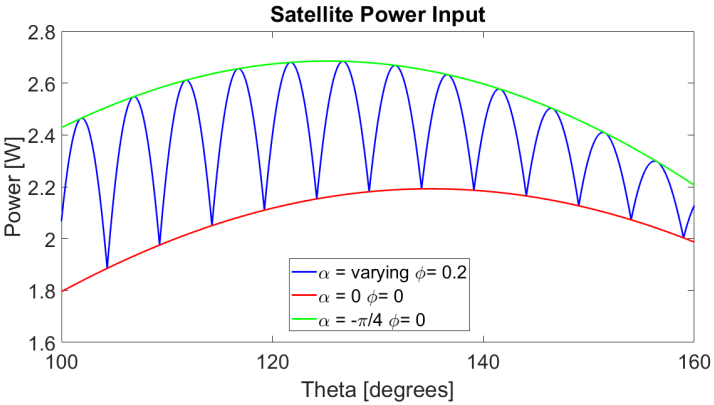


Figure 34: Satellite power input for $\phi = 0.2$.

has been reproduced by the current source that controls the test stand LEDs in order to generate light for the tests (see Section 2.11).

2.9 SATELLITE POWER CONSUMPTION DESCRIPTION

This section describes the FloripaSat engineering model power consumption. From this description, a current curve has been implemented to operate as an emulated load for the EPS architectures tests. The description is based on the subsystems' circuit blocks power con-

sumption, which have been calculated from the components' datasheets.

2.9.1 On Board Data Handling - OBDH

The OBDH is responsible for managing the nanosatellite tasks. It communicates with the remaining subsystems via the I^2C protocol operating as the system master. The OBDH has functions as: sensors measurements, data processing, data storage, etc. These functions are performed by circuit blocks or components. The power consumption of each circuit block/component is presented in Table 8. The power consumption is calculated multiplying the OBDH voltage bus (3.3V) by the current consumed by each circuit block/component. Some components are internally powered with different voltages, and their power consumption are calculated with the equations provided in their datasheets.

For this power consumption estimation the OBDH tasks under execution have been considered in continuous operation. Sensors measurements, for example, are not continuous, they occur by sampling. However, distributing the tasks in time to compute their combined power consumption would request a complex analyzes of the system. Since these power consumption variations represent a very small percentage (less than 1%) on the total satellite power consumption, the worst case scenario have been considered.

2.9.2 Battery heater

The batteries that store the satellite harvested energy shall operate within a certain temperature range (from -20°C to 60°C) (Samsung SDI Co. Ltd., 2007). Nanosatellite thermal analyzes show that lower temperatures than -20°C may be achieved in space environment (FILHO et al., 2016). Therefore, the FloripaSat is provided with a battery heater, which consists of two thin film heaters (resistances). Based on the thermal analyzes performed for the FloripaSat, one concludes that nearly 3W are necessary to warm up the batteries along the period that the satellite stays on Earth's shadow. Therefore, this power consumption is also emulated for the EPS architecture tests, as shown in Figure 35. In this power consumption description the heaters voltage were considered constant (5V).

Table 8: On Board Data Handling power consumption.

Component	Quantity	Current [mA]	Power [mW]
IMU (MPU-9250)	1	3.7 [a]	12.21
IMU (BMX055)	1	5.7 [b]	18.81
Voltage reference	1	0.026 [c]	0.0008
Operational amplifier	4	0.2 [d]	2.64
External watchdog	1	0.025 [e]	0.0825
microSD	1	0.25 [f]	0.825
Non-volatile memory	3	0.05 [g]	0.495
Microcontroller	1	8.39 [h]	57.1134
Current amplifier	1	0.23 [i]	2.277
Shunt (0.05Ω)	1	19.271	0.01857
Total			94.47

[a] (INVENSENSE, 2016)
[b] (BOSCH SENSORTEC, 2014)
[c] (TEXAS INSTRUMENTS, 2015a)
[d] (TEXAS INSTRUMENTS, 2016c)
[e] (TEXAS INSTRUMENTS, 2015c)
[f] (ENVOY DATA MEMORY, 2009)
[g] (ISSI, 2014)
[h] (TEXAS INSTRUMENTS, 2013)
[i] (MAXIM INTEGRATED, 2012)

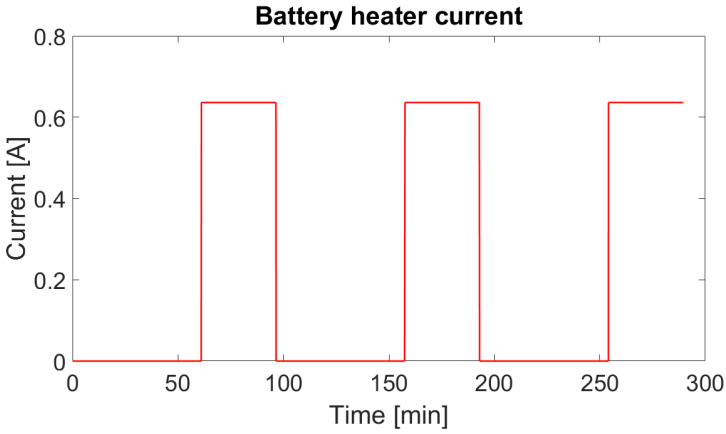


Figure 35: Battery heater current.

2.9.3 Telemetry, Tracking and Command - TT&C

The FloripaSat nanosatellite is provided with two radios: the transceiver and the beacon. These radios are part of the TT&C subsystem. The transceiver radio is controlled by the OBDH microcontroller, and it is responsible by sending to the ground stations the satellite main data package. The transceiver may operate periodically, or may send data by the ground station request. The beacon has its own dedicated microcontroller, and it is responsible by sending the satellite vital information to the ground station. It operates only in periodic mode.

For this power consumption estimation, the beacon radio is considered to send data every 60s, being active for 0.6s. The transceiver is considered to be active once per orbit, communicating for 10min. Table 9 summarizes the TT&C power consumption. The beacon microcontroller is the same that the OBDH one. Their power consumption have been considered the same. Also, it is important to note that only one radio is communicating at a time.

Finally, Figure 36 shows the FloripaSat OBDH and TT&C combined current consumption. From that figure, one may note a continuous current (the sum of OBDH components and TT&C microcontroller), and two different current pulses (the transceiver and beacon communication).

Table 9: Radios' power consumption.

Component	Triggering Period [s]	Active time [s]	Current [mA]	Power [mW]
PA (Transceiver)	60	2	600 [a]	1,980
PA (Beacon)	10	0.6	600 [a]	1,980
Microcontroller	continuous	continuous	8.39 [b]	57.1134
Total				2,037.1134

[a] (RFMD, 2014)
[b] (TEXAS INSTRUMENTS, 2013)

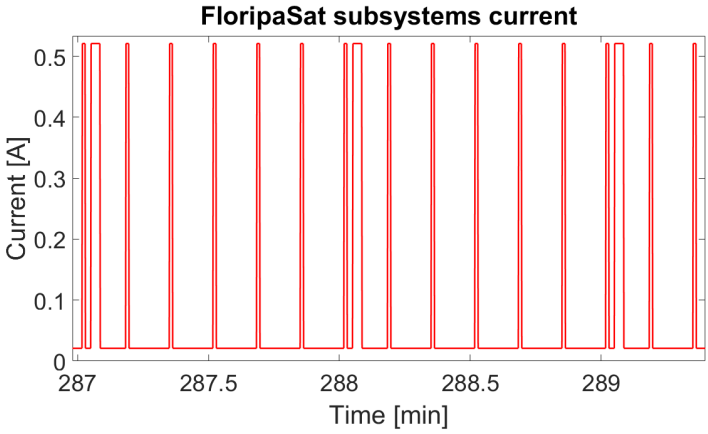


Figure 36: FloripaSat subsystems current - zoomed.

2.9.4 Electrical Power System

This section describes the EPS MPPT boost regulator power consumption (FloripaSat EPS engineering model). This power consumption **is not considered** for the EPS architectures tests (emulated current load of Figure 36), since it takes part on the EPS efficiency evaluation. Every single architecture has its own components and strategy to control the solar panels, which leads to more or less energy consumption. Therefore, the test condition would not be the same if the EPS architectures power consumption were part of the nanosatellite emulated electronic load. However, an example of this subsystem power consumption is presented in Table 10, only for the sake of comparison to the remaining nanosatellite subsystems.

The voltage regulator 5420 in Table 10 powers the digital blocks of the TT&C and of the EPS, therefore its consumption may be considered constant. In the same way, the voltage regulator 5410 consumption is considered constant, since it powers the OBDH digital blocks. Finally, the voltage regulator 54540 powers the radios power amplifiers, and for this reason its power consumption is dynamic.

Considering all the power consumption presented in last sections, it is important to mention that not all these electrical load shall be triggered at the same time. This would cause a current peak that the EPS would not be able to handle. Instead, the nanosatellite shall smoothly trigger its tasks. This soft start shall be considered on the mission

Table 10: EPS power consumption.

Component	Quantity	Current [mA]	Power [mW]
Voltage reference	1	0.026 [a]	0.0008
Op-amp	5	0.2 [b]	0.66
Current amplifier	7	0.23 [c]	0.759
Timer 555	1	0.250 [d]	0.825
External ADC	1	0.59 [e]	1.947
Microcontroller	1	8.39 [f]	57.1134
Kill-Switches	4	350	3.063 [g]
Battery monitoring	1	0.135 [h]	1.134
Battery protection	1	700	10.29 [i]
5420 regulator	1	49.456	206.588 [j]
5410 regulator	1	21.281	210.322 [k]
54540 regulator	2	600	149.032 [l]
Shunt (0.75Ω)	1	0.2	3
Shunt (0.05Ω)	6	508	12.903
Total			657.6372

[a] (TEXAS INSTRUMENTS, 2015a)
[b] (TEXAS INSTRUMENTS, 2016c)
[c] (MAXIM INTEGRATED, 2012)
[d] (TEXAS INSTRUMENTS, 2016b)
[e] (TEXAS INSTRUMENTS, 2016a)
[f] (TEXAS INSTRUMENTS, 2013)
[g] (VISHAY, 2011)
[h] (MAXIM INTEGRATED, 2011)
[i] (FAIRCHILD SEMICONDUCTOR, 2001)
[j] (TEXAS INSTRUMENTS, 2008)
[k] (TEXAS INSTRUMENTS, 2014)
[l] (TEXAS INSTRUMENTS, 2017)

planing in order to attend mission requirements without endangering the Electrical Power System.

2.10 COMPONENTS AND SYSTEM MODELS AND SIMULATIONS

The power input model described in Section 2.8 is an approximation based on satellite positioning in orbit and on solar panel efficiency. However, it does not take into consideration the EPS architecture influences on the solar panel control. This is clear when analyzing the Equation 2.40, which defines the solar panel power capability as a constant. Therefore, the goal of this section is to propose a more elaborated model, considering electrical characteristics of each circuit architecture.

Analyzing the circuit diagram of Figure 5 and equations 2.2, 2.1 and 2.15, one may note that the solar panel voltage depends on circuit parameters. This figure and equations describe the directly coupled EPS architecture, however, the same is valid for the remaining architectures. Therefore, there should be a model that is able to define the whole electrical power system, considering the following elements: solar irradiance, solar panel, solar panel operation strategy, power consumption, circuit components, and battery. The solar irradiance and the solar panel models have been already presented (Section 2.8). The satellite power consumption has also been defined (see Section 2.9). The solar panel control strategy has been described in Section 2.6. Therefore, this section is dedicated to model and simulate the remaining elements of that list (circuit components and battery) to obtain a more precise system simulation.

The idea is to simulate the satellite instantaneous power input using the solar irradiance as the model input. The solar irradiance may be obtained by the model proposed in Section 2.8.2. From the solar irradiance values, the system model must be able to calculate: solar panels voltage, solar panels current, solar panels power, battery voltage and battery current. It is clear that the solar panel voltage and current will depend not only on the solar irradiance values but also on the EPS architecture. Therefore, in order to calculate that values from the solar irradiance data, the EPS circuit must be modeled.

The EPS architectures have already been mathematically described through sections 2.4 to 2.7. Therefore, this section will present how the components values and control strategies have been defined in order to simulate the architectures. As the directly coupled architecture is the simplest one, and because its components are present in the

remaining architectures, its model will be the first to be presented.

Figure 5 is the basis for the circuit description. Starting from the diode, the first step was to experimentally obtain the n and I_S values, which are present in Equation 2.3. The experimental $V - I$ curve has been obtained, with a fix temperature of 23°C. However, independently of n and I_S values, Equation 2.3 did not fit to the experimental data. Two possible reasons for this phenomena has been raised: diode non ideality and measurement imprecision. To solve this problem, a series resistance has been added to the ideal diode model. Figure 37 shows the experimental and simulated curves. Table 11 shows the parameter used for simulation.

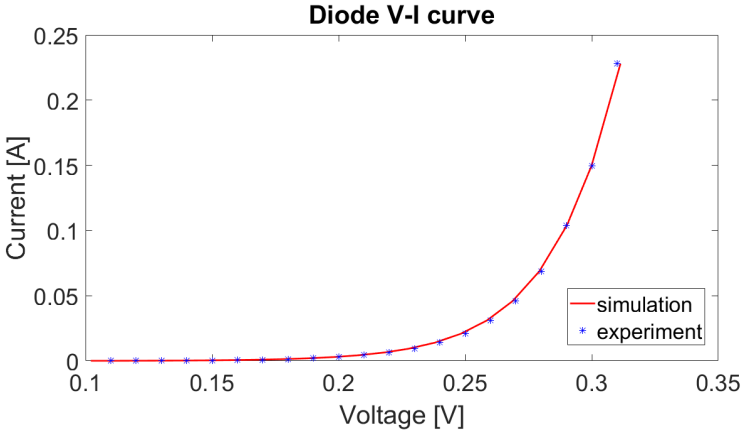


Figure 37: Voltage-current diode curve.

Table 11: Diode parameters experimentally obtained.

n	$RD_s[\Omega]$	$I_S[\mu A]$	$Temperature[°C]$
1.00404	0.0084	1.3032	23

Next component is the shunt resistor RS_{sp} used to measure the solar panel current. This component, as well as the RS_{bat} were defined simply by their resistance values. Each architecture has their shunt resistors to measure currents flowing through different blocks of their circuits. Table 12 present the resistor values used for simulation for all architectures.

Table 12: Resistances values used for simulation.

Architecture	$\boldsymbol{RS_{sp}}$	$\boldsymbol{RS_{bat}}$	$\boldsymbol{RS_{load}}$	$\boldsymbol{RS_{mppt}}$
DC	50mΩ	20mΩ	50mΩ	–
$LVDO$	50mΩ	15mΩ	50mΩ	–
$MPPTboost$	50mΩ	10mΩ	75mΩ	–
$MPPTIC$	50mΩ	20mΩ	50mΩ	25mΩ

Next components to be modeled are the MOSFETs M1 and M2 from Figure 5. As shown in Equation 2.10, among others, MOSFETs electrical behavior is dictated by fabrication parameters. Most of them are not present in components datasheet, therefore, they must be experimentally obtained. However, before planing any experiment, it is important to understand these MOSFETs functions and operation states. These MOSFETs are controlled by the battery monitoring chip and shall prevent the battery to be over and under charged. For the battery used in all proposed architectures, the battery full state voltage VF_{bat} is defined to be 4.2V and the battery discharged voltage VD_{bat} is defined to be 2.5V. With this values in mind, one may determine all possible operation for these pair of MOSFETs, as described in Table 13.

Measuring the MOSFETs' gate voltage and drain source current for all battery conditions described in Table 13, it has been confirmed that the MOSFETs operate only in three different forms: blocked (body diode reverse polarized), conducting (MOSFET in linear region), conducting (body diode directly polarized). The experiments show that when conducting the MOSFETs operate in the linear region. Therefore, they are modeled as resistors. The R_{DSON} average experimental resistance, for the two possible was gate voltage $V-g$ ($V_g = 4.3$V and $V_g = 9$V), was equal 0.00823455Ω (Figure 38). Finally, the MOSFET body diode has been modeled. The body diode $V-I$ curve is presented in Figure 39. The body diode parameters are presented in Table 14. This condition occurs only when the MOSFETs gate voltage is zero, that is when $V_{bat} < 2.5$V.

The battery model is a key component in order to simulate the whole EPS architecture. Battery modeling is a vast field of research, with different levels of complexity. Models vary from electrochemical reactions descriptions to equivalent circuits approximation (CHEN et al., 2014) (RODRIGUES et al., 2017). Since the goal of this work is to model the EPS subsystem, the battery model has been adopted from MathWorks (Generic Battery Block - Simscape Language). It is a

Table 13: Battery monitoring MOSFETs operation states.

Battery condition	**M1 state**	**M1 gate voltage**	**M2 state**	**M2 gate voltage**
$V_{bat} > 4.2$V - charging	conducting	9V	blocked	4.3V
$V_{bat} > 4.2$V - discharging	conducting	9V	conducting	4.3V
2.5V$< V_{bat} < 4.2$V - charging	conducting	9V	conducting	9V
2.5V$< V_{bat} < 4.2$V - discharging	conducting	9V	conducting	9V
$V_{bat} < 2.5$V - charging	conducting	0V	conducting	9V
$V_{bat} < 2.5$V - discharging	blocked	0V	blocked	9V

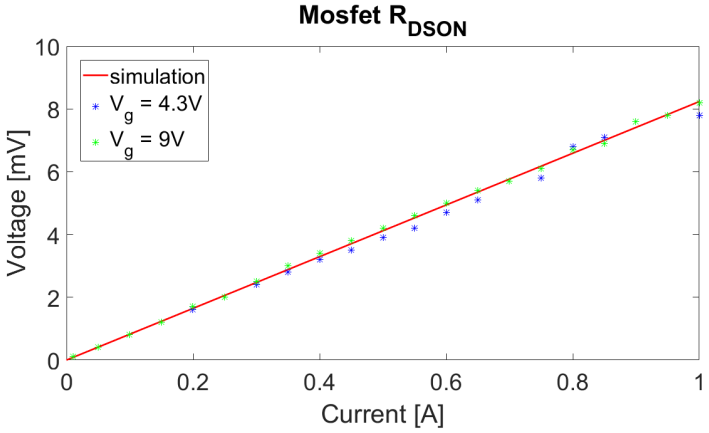


Figure 38: MOSFET R_{DSON} determination.

Table 14: MOSFET body diode parameters experimentally obtained.

n	$RD_s[\Omega]$	$I_S[\mu A]$	$Temperature[^oC]$
1.89492	0.27	0.126064	23

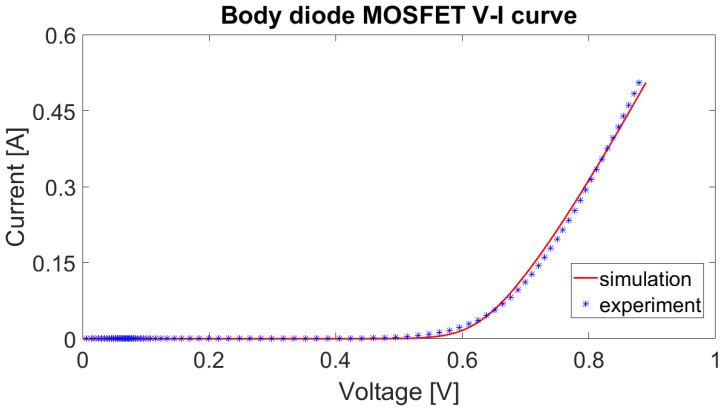


Figure 39: MOSFET body diode voltage-current curve.

lithium ion battery, for which the parameters have been configured according to the FloripaSat battery model(ICR18650-30A) (Samsung SDI Co. Ltd., 2007).

Finally, the circuit block between the solar panel and the load has been modeled. For the MPPT boost regulator architecture, the converter has been modeled as a voltage source controlled by voltage. The control voltage is used to defined the converter gain, and it is generated by the Perturb and Observe algorithm (implemented through a MATLAB function block). Appendix A shows the directly coupled and MPPT boost regulator simulation diagram, including all the components described in this section.

2.11 EPS TEST STAND

In order to test the EPS architectures a test stand has been proposed. The idea is to emulate the solar irradiance behavior, controlling high power LEDs through a current source. Four high power LEDs (100W each) have been used to illuminate the solar panels. In space, the solar irradiance may be considered constant, and the input power variation is caused by the satellite dynamic flight. However, in order to reproduce this behavior in laboratory, it is much easier to vary the light intensity, then create a motion system for the solar panels. Therefore, the LEDs current control curve is based on the model presented in Section 2.8.2. The power input worst case scenario was adopted to generate the current curve. Figure 40 shows the LEDs current curve for three consecutive satellites orbit, which has the same behavior as the satellite power input in space. Although the LEDs do not provide the same power level as available in space, the provided power is enough to compare all EPS architectures efficiency.

It is worthy to mention here that all tests performed for this thesis were based on three orbits experiments. This is the period for which the batteries are able to continuously power the system. After nearly three orbits, the EPS switches-off the batteries to avoid under voltage state. In a real satellite mission, the battery switch-off state is undesired, but it may occur in failure conditions of the energy harvesting subsystem (damaged solar panel, solar panel control block failure, etc). Normally, the batteries capacity shall be defined to continuously power the satellite, which demands knowing the satellite power input and the satellite power consumption. In this thesis, the calculation of both power input and power consumption are presented (Sections 2.8 and 2.9). The nanosatellite power consumption may be precisely emulated by controlled current sources but, the nanosatellite power input level could not achieve the one that is observed in outer space, by using

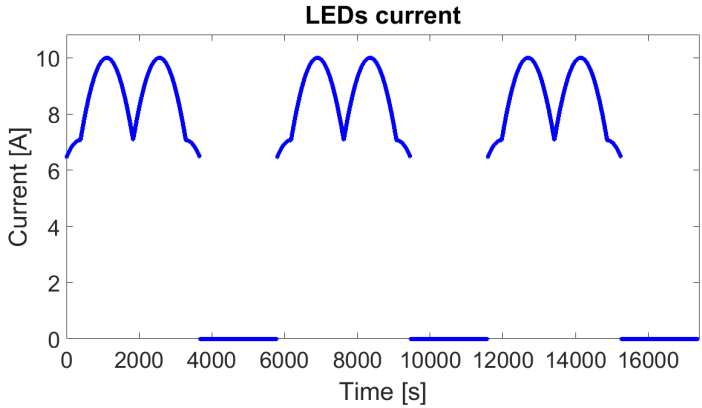


Figure 40: LEDs current for three consecutive orbits.

LEDs. Besides this, the solar panels adopted for the experiments are not as efficient as the ones planed for the mission (high cost space qualified solar panels). Therefore, the decision was to maintain the nanosatellite real power consumption and test the EPS architectures for as long as possible without entering the battery switch-off state, which resulted in experiments of three orbits.

The LEDs were attached to an aluminum heat-sink with a cooler on it, and placed on the top of a wooden case. Besides this, two coolers have been placed in one side of the case, in order to remove the heat from over the solar panels. Three solar panels were tested at time, since not all the six solar panels receive the Sun light simultaneously in orbit. Figure 41 shows a picture of the test stand. In order to control the LEDs current, a controllable voltage/current source has been used (TDK-Lambda GENH 40-19). A LabVIEW control software has been designed to generate the current curve. The irradiance provided by the high power LEDs was measured with a pyranometer (SMP6). This irradiance characterization allows reproducing the same scenario in simulations.

Besides the LEDs curve generation, the test stand also emulates the satellite power consumption. It uses a power analyzer (N6705B), with one channel configured as a four quadrant voltage/current source to emulate an electronic load. The electronic load is defined with a constant voltage (emulating the dc-dc converter voltage output) and a negative variable current. The load's current depends on the satellite electrical load specification, as described in Section 2.9. The load cur-

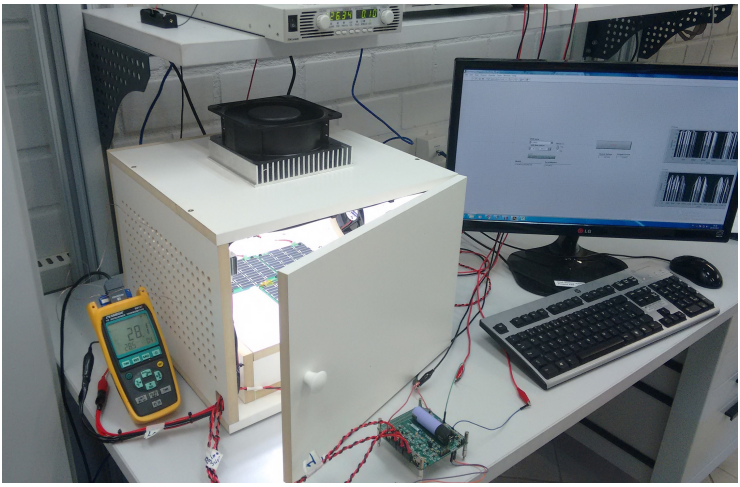


Figure 41: Test stand.

rent curve is defined by a data vector that may be loaded in the power analyzer.

The EPS under test is them connected to the computer, through a UART/USB driver. A debug embedded software is used to send all the EPS information to the computer via the EPS serial communication channel. The information is received and saved in the computer for further processing and analysis.

2.12 RESULTS

This section presents the performance comparisons among all EPS architectures. A discussion regarding the experimental results on the three orbit experiment is presented. Also, it compares the experimental results of each architecture with the simulations described in Section 2.10.

2.12.1 EPS architectures performance comparison

Figures 42 to 44 compares the solar panels voltages of the four EPS architectures during the three orbit experiment. From these figures one may note that the solar panel voltage of the MPPT IC architec-

ture is lower than the remaining ones. This is expected, as explained in sections 2.7 and 2.8, because the MPPT integrated circuit architecture is powered by solar panels with lower operation voltage (4 cells connected in series with 10 branches in parallel). As the MPPT integrated circuit operates as a boost regulator, it is able to step up the solar panel voltage from the 1.5V range to the battery voltage 3.7V range. It is also interesting to note the voltage control performed by the on-chip Perturb and Observe method, which causes the solar panel voltage to oscillate. The comparison also shows that the Perturb and Observe method implemented by the integrated circuit is more unstable than the one performed bay the EPS MPPT Boost Regulator architecture. The main reason for this is the method control increment step. For the MPPT IC it is defined in hardware, and there is no possibility to change it externally. For the MPPT Boost Regulator architecture, it is implemented in the MSP430 microcontroller, and may be changed in the embedded software. This is one of the biggest advantages of implementing the MPPT algorithm in software.

Also from figures 42 to 44, it is possible to note a different behavior in the solar panels voltage from the VLDO architecture in the experiment's first orbit. Differently from the other solar panels voltages, it clamps around 5V in the first orbit. This occurs due to the VLDO output voltage limitation, which is set for 4.2V. Therefore, when the battery voltage approach this value (see Figure 55), the voltage regulator output clamps in 4.2V and its input voltage operates in open state. In this case, the regulator input voltage (which is the solar panel voltage) is determined by the regulator output current. Therefore, the solar panel voltage is not affected by the battery voltage anymore, but by the load current. Later in Section 5.4, a discussion is presented on how to benefit from this phenomena to achieve energy harvesting maximization.

Besides this, figures 42 to 44 show variations on voltage levels which are not presented in the irradiation curve (see LEDs current control curve in Figure 40). These variations are caused by the load's power consumption variations. For the MPPT architectures (both IC and Boost Regulator) they are not that clear, due to the unstable solar panel voltage control. However, for the remaining architectures (DC and VLDO), these voltages variations may be noted. Figure 45 shows a zoomed portion of the solar panel 1 voltage, in order to better visualize these solar panels voltage changes caused by the load power consumption variations. The solar panel voltage from the MPPT IC is out of range in this figure.

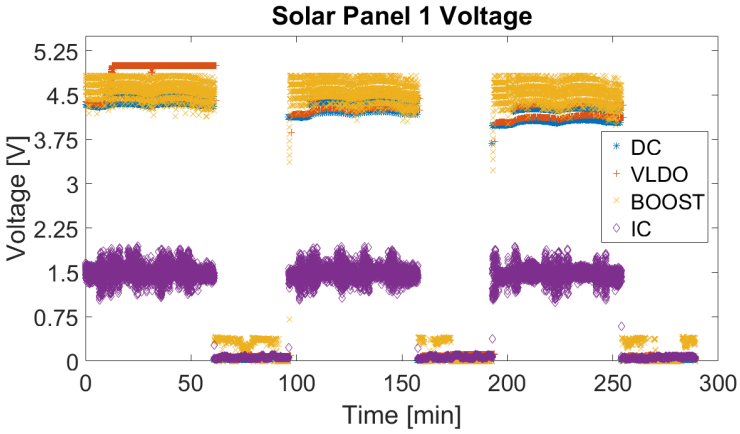


Figure 42: Solar panels 1 voltage.

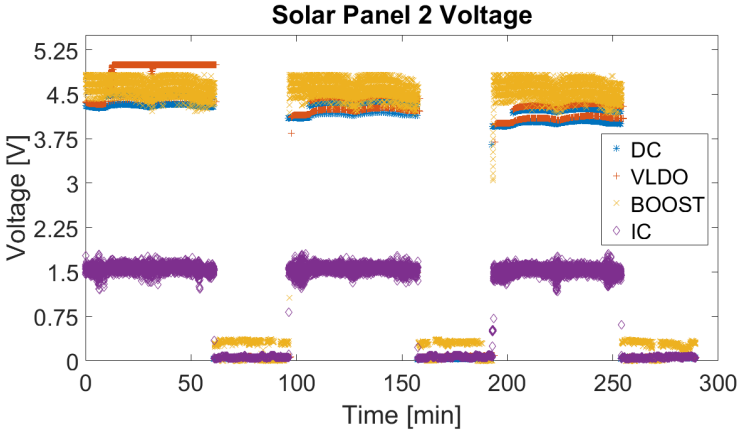


Figure 43: Solar panels 2 voltage.

Figures 46 to 48 present the solar panels currents comparison for all four EPS architectures. As expected, the solar panels from MPPT IC architecture present higher levels. Once again, this is due to the solar cells connections in the solar panel, the more cells connected in parallel, the higher the current, the lower the voltage. These figures also show the VLDO output clamping effect on solar panels current. Analyzing the first orbit, one may see that the current decreases after the clamping for the VLDO architecture. This occurs due to the low

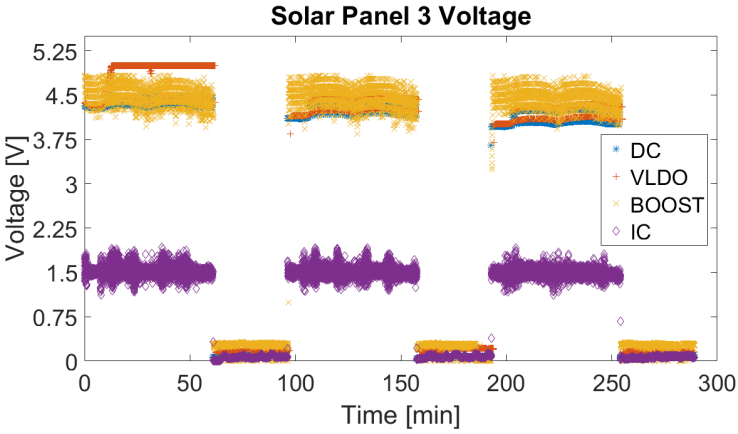


Figure 44: Solar panels 3 voltage.

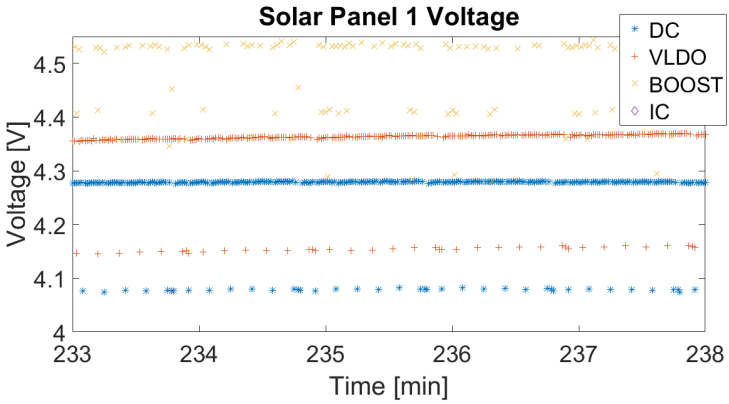


Figure 45: Solar panels 1 voltage - zoomed.

current demanded by the load, which is far form the I_{MPP} for that panel, causing the solar panel voltage to clamp far from the V_{MPP}, consequently delivering less current. Periodic variations on current level are observed for the DC and VLDO architectures, as they were on the solar panels voltages curves, also due to the power consumption variations (caused by radio periodic transmission - see Section 2.9.3). Figure 49 shows the current curve zoomed to observe its periodic variations.

Analyzing the voltage and current curves one may understand

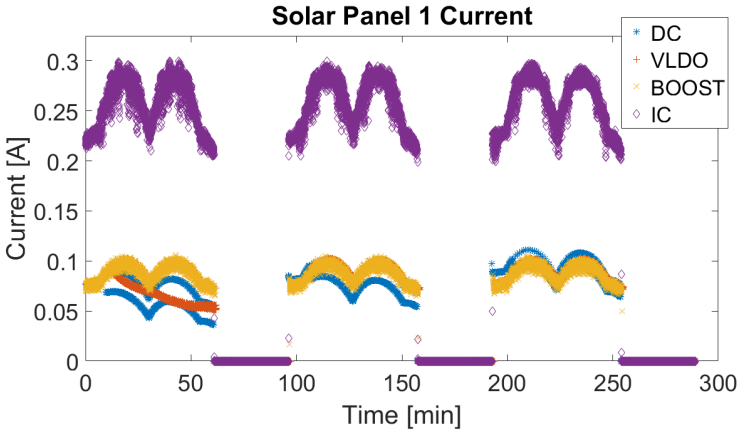


Figure 46: Solar panels 1 current.

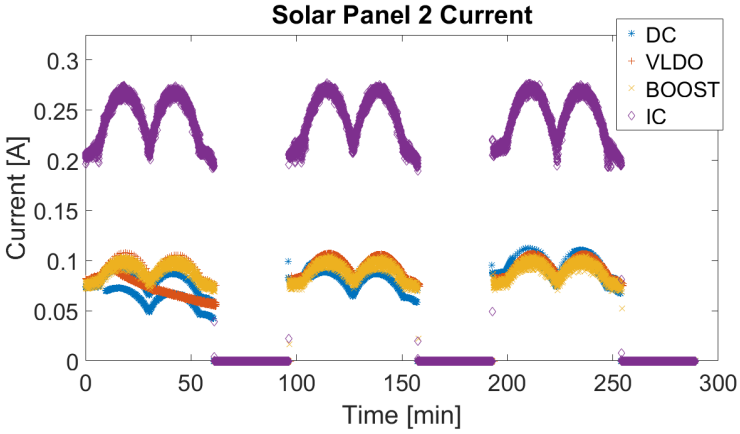


Figure 47: Solar panels 2 current.

how an EPS architecture operate the solar panels. However, no conclusion regarding the EPS efficiency may be stated by inspecting these curves separately. This occurs because the solar panels are different in their maximum operation point, due to the differences in their solar cells arrangement. Therefore, the solar panel power shall be analyzed. Since all solar panels have the same number of solar cells (40 each), their power output may be fairly compared. Figures 50 to 52 show the comparison between the solar panels output power for all four EPS

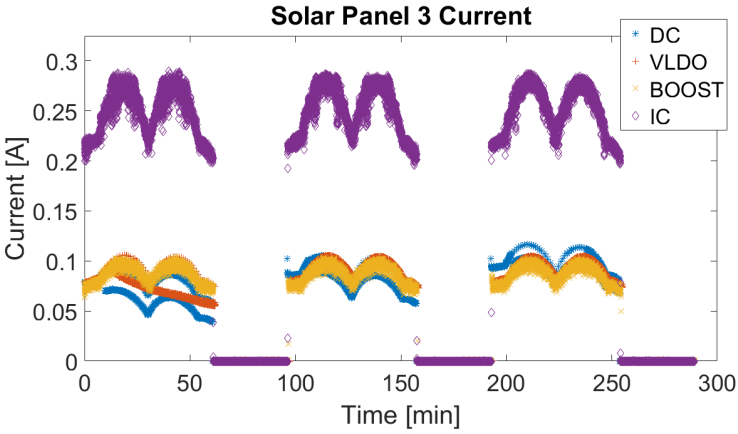


Figure 48: Solar panels 3 current.

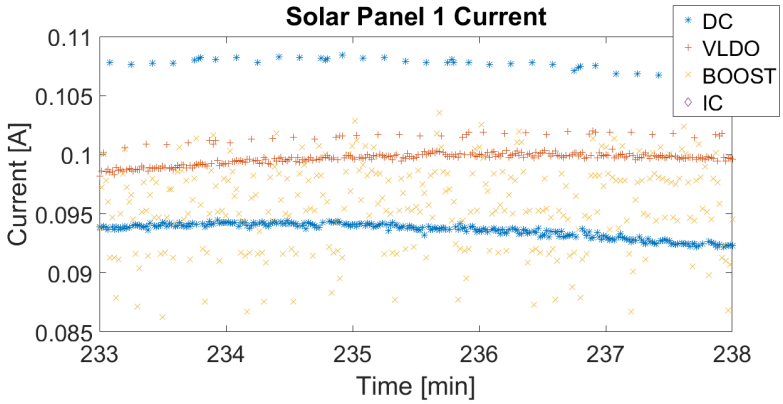


Figure 49: Solar panels 1 current - zoomed.

architectures. Firstly, one may note that the directly coupled EPS architecture solar panels power output were lower than the remaining ones. However, as the battery voltage decreases along the experiment, the power input increases. At the third orbit, the solar panels power output of this architecture is almost equal to the remaining ones. This confirms the battery voltage dependence mathematically demonstrated in Section 2.4. Chapter 4 shows how to achieve energy harvesting maximization with this architecture, by controlling the load power consumption (scheduling satellite tasks execution).

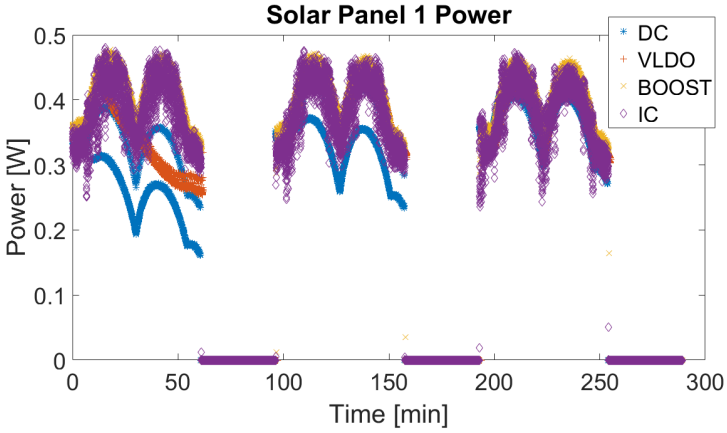


Figure 50: Solar panels 1 power.

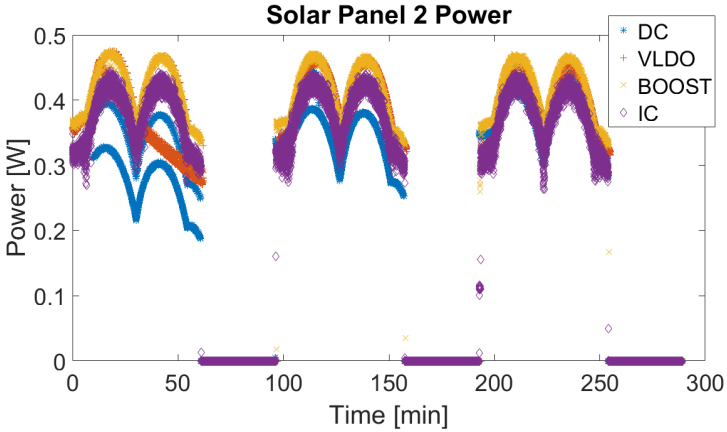


Figure 51: Solar panels 2 power.

From figures Figures 50 to 52 one may also note that the VLDO output voltage clamping impaired its energy harvesting performance. As mentioned before, this has occurred due to the current dependence in the clamping state. In case the load current were higher, the solar panel voltage would be lower, and this solar panel could operate closer to its MPP as well as the others. After leaving the clamping state, the VLDO architecture has presented better results in maximizing the power input than the DC and even than the MPPT IC architecture.

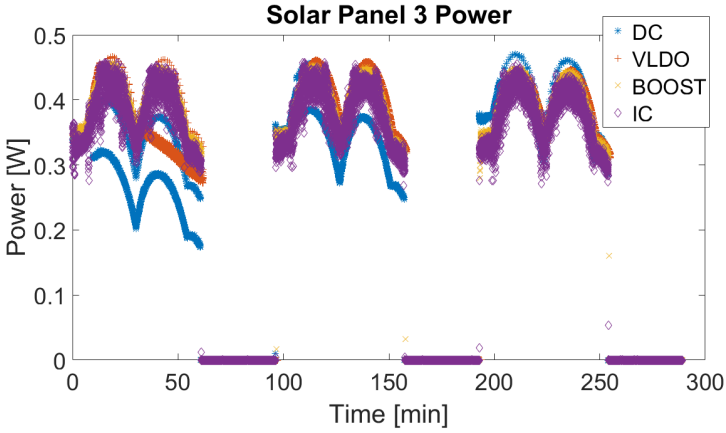


Figure 52: Solar panels 3 power.

The imprecise hardware control strategy implemented by the MPPT IC precluded it to operate the solar panels continuously at the MPP. The possible main reason for this is the fix (and too large) algorithm perturbation step. The MPPT Boost Regulator has presented the best performance in maximizing the solar panels power input. Figure 53 shows the solar panel 1 power curve zoomed, in order to better visualize what has just been concluded.

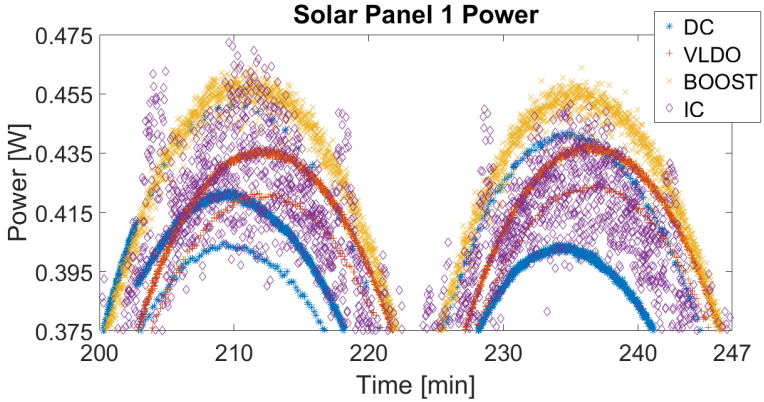


Figure 53: Solar panels 1 power - zoomed.

Before moving for further analysis, it is important to mention

that the results were not the same for all solar panels. Comparing the curves between panels 1, 2 and 3, for voltage, current and power, one may note that there are differences. Firstly, regarding the voltage, current, and power levels, they may differ from panel to panel for many reasons. The two most relevant appear to be the irradiance and the temperature distribution inside the case where the tests have been conducted. The three solar panels have been equidistantly distributed regarding the LEDs focus center. However, both temperature and irradiance measurements show that the solar panels were not at the same conditions. This is certainly not a problem in this analysis, since the comparisons are not made among solar panels 1, 2 and 3 only, but mainly on the overall power input delivered by the three solar panels together. Therefore, a numerical analysis, instead of only observing the plots, is essential to avoid an evaluation misunderstanding. For this reason, the three solar panels power were summed for each architecture (Figure 54) and integrated to calculated the total harvested energy per architecture, along the three orbit experiment.

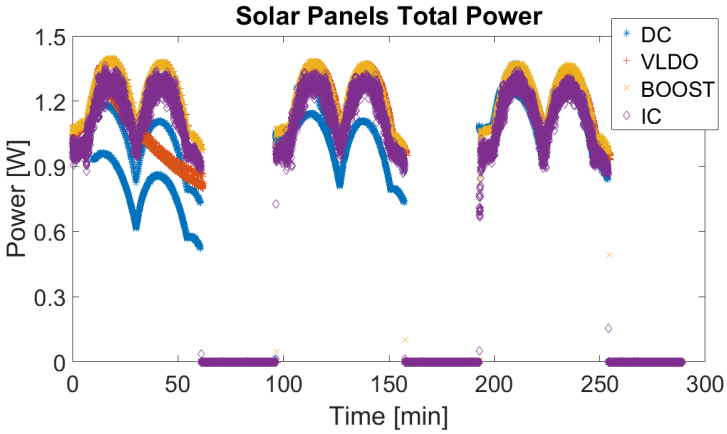


Figure 54: Solar panels 1, 2, and 3 summed power.

The energy harvested calculation result is shown in Table 15. As expected, the MPPT boost regulator architecture has harvested more energy than the other architectures. Surprisingly, the VLDO architecture has harvested more energy than the MPPT IC one. This demonstrates the importance of a good tuning for the Perturb and Observe algorithm parameters, which in this case is not possible because they are hardware defined into the chip. Also as expected, the directly

coupled architecture has harvested less energy than the others. The reason for this is the solar panel voltage dependence. However, it is worth mentioning that, with a good control on satellite tasks execution (load power consumption), the battery voltage could operate in a certain voltage that would cause the solar panels to operate close to their MPP. This topic is better addressed in Chapter 4.

Table 15: EPS architectures harvested energy along three orbits experiment.

EPS architecture	**Harvested energy**
Directly coupled	10.981kJ
MPPT integrated circuit	12.417kJ
VLDO regulator	12.654kJ
MPPT boost regulator	13.438kJ

Figure 55 shows the architectures battery voltage. It is worth remembering that the MPPT boost regulator architecture has two Lithium-Ion cells connected in series, differently than the others which have only one. However, the MPPT boost regulator architecture individually measures the batteries voltages, which has allowed to plot all battery voltages in the same figure.

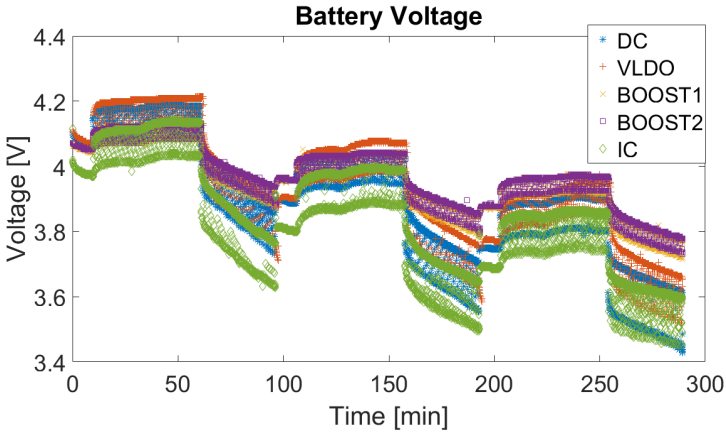


Figure 55: Batteries voltage.

Figure 56 shows the battery current comparison for all architectures. Positive values represent currents entering the battery. As

explained in Section 4.4.1, the load consumption is the same for all architectures, which causes the current load to have the same shape for all architectures. However, as the battery voltage decreases, the current shall increase to sustain the same output power to the load. Therefore, one may note the higher current consumption for DC and VLDO architectures in Figure 56. The MPPT boost regulator battery current is much lower than the others due to the two batteries connected in series. The battery voltage for this architecture is twice the value from the others, which causes the battery current to be nearly half value.

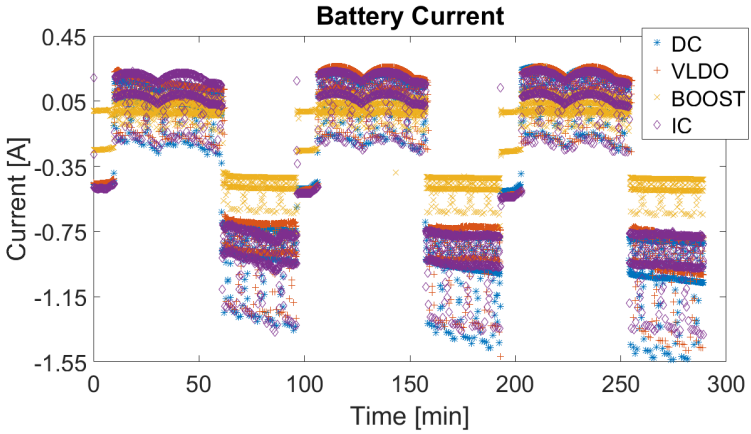


Figure 56: Battery current.

All the analysis made so far concerns to the energy harvesting capability. However, the EPS circuit itself consumes energy. Therefore, one of the most important issues addressed in this section is to evaluate not only the input power, but to analyze the "costs"of operating the solar panels. Since the load's power is the same for all architectures, it is reasonable to define the battery remaining capacity as a metric to evaluate the overall subsystem efficiency. This is valid because all the energy that is not delivered to the load is used to power the EPS. Therefore, the architecture which has the battery with more electric charge stored after three orbits may be considered the most efficient one.

The only exception for this analysis is the MPPT boost regulator architecture. Due to the usage of two Lithium-Ion cells connected in series, the battery current is lower along the experiment. Consequently,

the remaining capacity of the cells are going to be much higher than the others. This occurs because the remaining battery capacity is calculated by a Coulomb counter algorithm (implemented in the battery monitoring chip). Therefore, one may analyze the three architectures efficiency by Figure 57, but to compare all four architectures a different approach has been adopted and it is discussed later in this section. Figure 57 shows that the VLDO has been the most efficient architecture (not considering the MPPT boost regulator architecture yet), followed by the MPPT IC, followed by the directly coupled. It is worthy to remember here that all EPS architectures spent more energy than they are able to harvest, as already discussed in Section 2.11. This causes the batteries energy to decrease along the experiment. Although this is not the best configuration for a real mission, it does not affect the energy efficiency comparison.

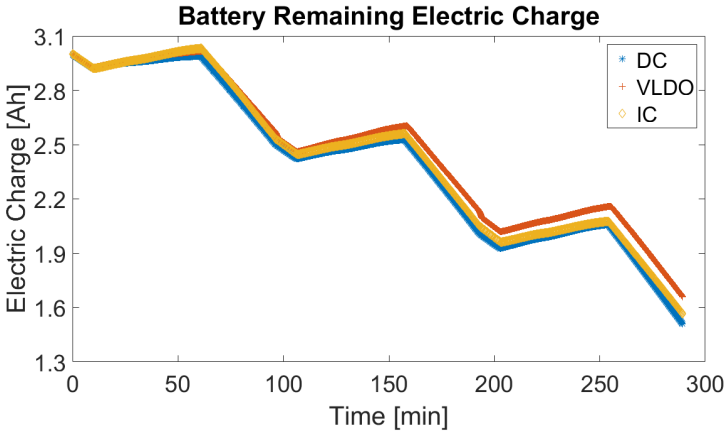


Figure 57: Battery remaining electric charge.

In order to compare all four architectures efficiency, the proposed evaluation method is to calculate the batteries delivered power (Figure 58) by multiplying their instantaneous voltage and current. Also, a numerical comparison may be performed by integrating the batteries delivered power to find out the total spent energy by each architecture along the three orbit experiment. Table 16 shows the global efficiency comparison results. Although the MPPT boost regulator architecture is the architecture that harvests more energy, it is the most inefficient circuit. The reason for this result is the losses in the boost regulator components, mainly in the inductor. The VLDO regulator architecture

demonstrated the better global efficiency, followed by the MPPT IC, and than the directly coupled architecture.

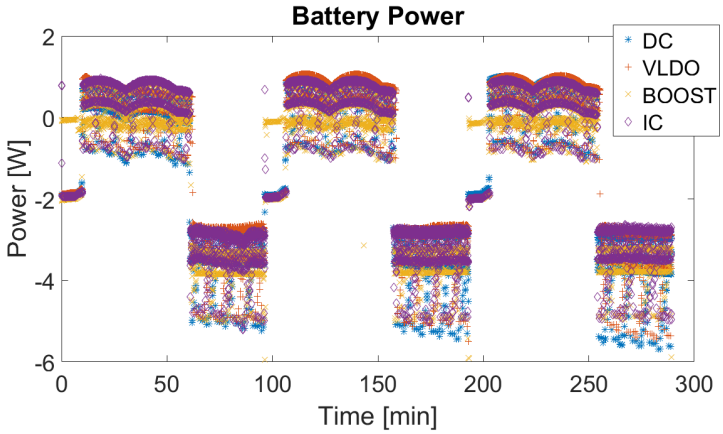


Figure 58: Batteries delivered power.

Table 16: EPS architectures spent energy along the three orbit experiment.

EPS architecture	**Spent energy**
VLDO regulator	18.102kJ
MPPT integrated circuit	18.931kJ
Directly coupled	20.052kJ
MPPT boost regulator	22.476kJ

Figure 59 shows the solar panels temperature. They were measured by two thermocouples placed between the three solar panels under test. The temperature variation behavior is very similar for all tests, although the curves show some amplitude variation. This differences occurred due to the room ambient temperature differences during the tests. Although the solar cell plays an important role on solar panel efficiency, the temperature differences (worst case nearly 10°C) were not significant to affect the architectures efficiency analysis.

Finally, all presented results may be summarized in Table 17, where the main features of each architecture are compared. Numbers from 1 to 4 are used to classify the architectures regarding their featu-

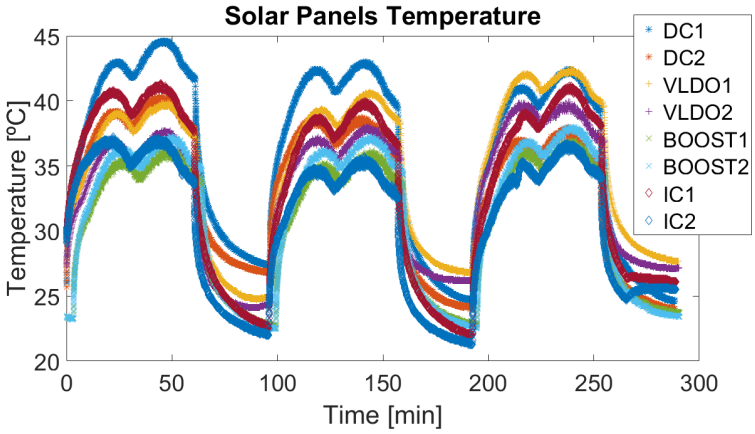


Figure 59: Solar panels temperature.

res, where the number 1 means the most costly, most complex to design and most efficient architecture.

Table 17: EPS architectures comparison.

architecture	**MPPT**	**complexity**	**cost**	**efficiency**
DC	no	4	4	3
VLDO	no	3	3	1
MPPT chip	yes	2	1	2
MPPT boost	yes	1	2	4

2.12.2 EPS architectures simulation results

Figure 60 shows the solar panel voltage (simulation and experimental) of the directly coupled EPS architecture. The comparison shows that the simulation curve fits on the experimental data. Although the voltage average values are consistent with the experiment, the experimental dynamic behavior was not perfectly reproduced by the simulation. The reason for this less sensitive behavior presented by the simulated solar panel voltage is related with the battery modeling, as discussed later on this section. Since the solar panel voltage is strongly affected by the battery voltage in this architecture, the slow

response of the battery model affects the solar panel voltage simulation. Even though, the simulation results are satisfactory, since the model has predicted the solar panel average voltage along all the orbits.

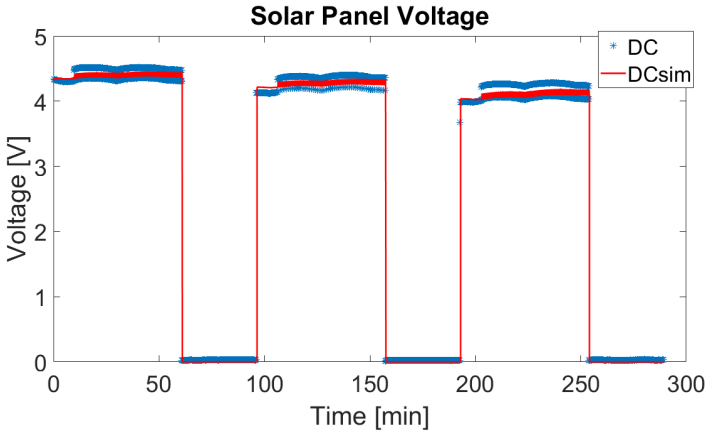


Figure 60: DC solar panel voltage simulation and experimental.

The solar panel current presented in Figure 61 shows that the predictions followed the current increase along the three orbit experiment. However, the simulated solar panel current level was slightly higher for the whole experiment. This effect is caused by the lower simulated battery voltage level compared to the experiment. The lower simulated battery voltage levels along the three orbits has caused the solar panels to operate closer to their MPP in the simulation. This lead to higher current levels in the simulation.

Figure 62 shows the comparison between the experimental and simulated directly coupled EPS architecture battery voltage. The simulated battery voltage has also followed the experimental behavior, though the differences increase as the battery voltage decreases along the experiment. Also, a voltage mismatch at the beginning of the simulation is verified. This effect is caused by the parameters definition in the battery modeling. Setting a higher voltage value for the full state of charge causes the battery voltage to be much higher during the eclipse. With the battery modeling limitations, this has been the best configuration to address the following inconsistency: battery voltage initial condition mismatch against battery voltage dynamic responsiveness.

Figure 63 shows the comparison between the simulated and experimental solar panel delivered power. The simulation has presented

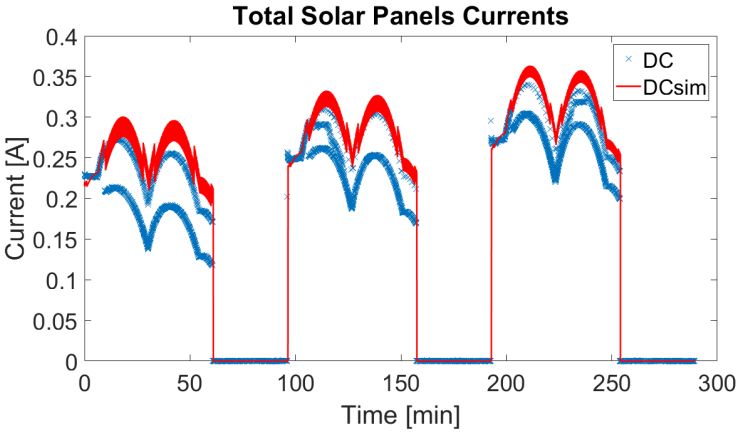


Figure 61: DC solar panel current simulation and experimental.

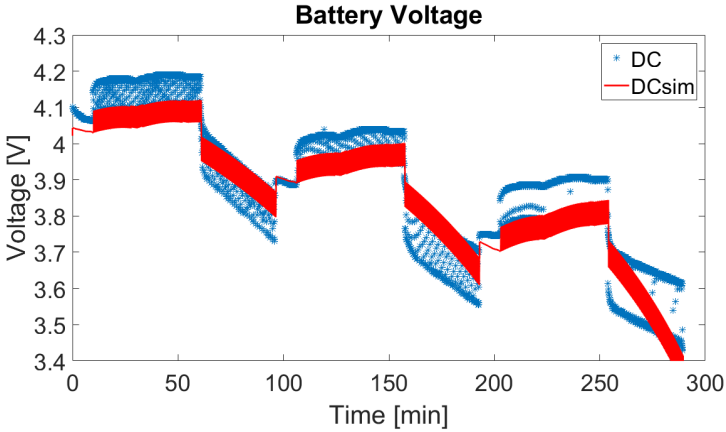


Figure 62: DC battery voltage simulation and experimental.

satisfactory results with a similar behavior of the curves. However, the solar panel current mismatch caused by the battery voltage prediction error strongly affects the power input prediction. Figure 64 shows an absolute error analysis on the nanosatellite power input prediction. One may note that errors are higher on the curve discontinuity points (when the satellite enters in Earth's shadow). Besides the visual evaluation provided by the absolute error plot, the average error has been calculated along the whole experiment. From this calculation, the simulation

has predicted an average power error of 20.9%.

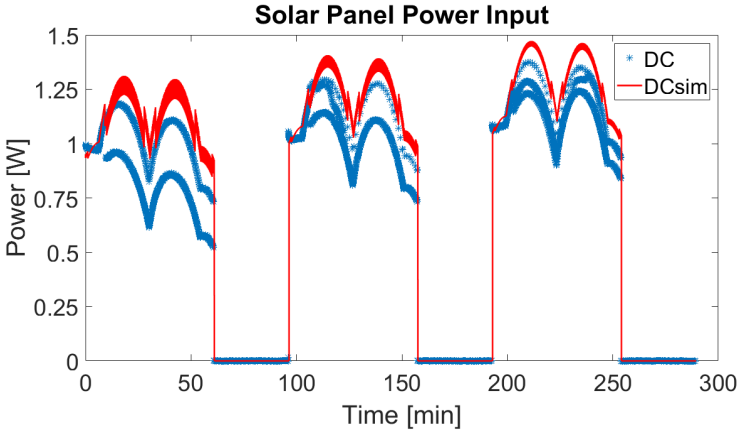


Figure 63: DC solar panel power simulation and experimental.

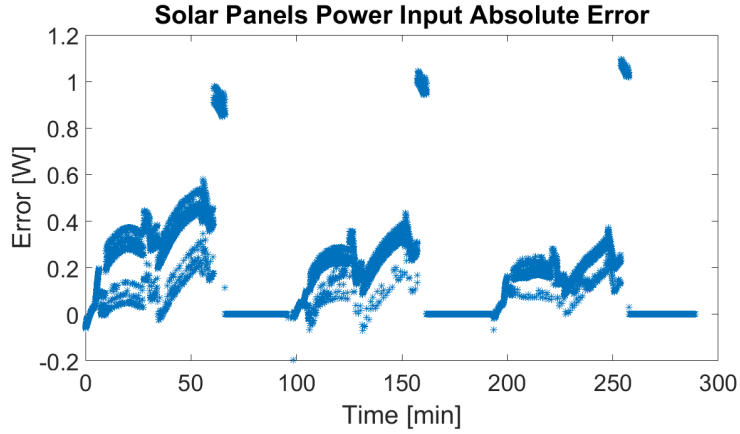


Figure 64: DC solar panel power absolute error.

Although the power input average error is relatively high, the simulation result may be considered satisfactory for two reasons: first, even for a complex feedback system, with several interdependent components, all the parameters evaluated responded coherently with the experimental results. Second, the battery modeling inaccuracy is evident from Figure 62 and it has strongly affected the final evaluation.

Therefore, improving the battery model will certainly reduce the overall simulation errors.

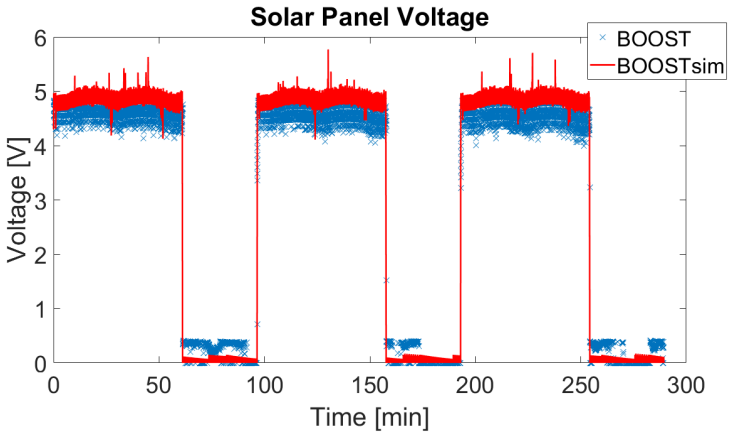


Figure 65: MPPT boost solar panel voltage simulation and experimental.

Figure 65 shows the solar panel voltage comparison for the MPPT boost regulator architecture. The MPPT control effects are visible in that curves. One may note the differences from Figure 60, which for the same irradiation input data, presents a much more stable behavior. Some instabilities on the simulated perturb and observe algorithm causes the solar panel voltage to present peaks. This instabilities may have been caused by the algorithm step definition. They have been defined as the same (for the real embedded system and for the simulation), however, the simulated system seems to respond more rapidly, causing undesired instabilities. Interesting to note here that the voltage level is nearly the same for the three orbits. This happens due to the solar panel voltage independence on the battery voltage, caused by the boost converter placed between the solar panel and the load.

Figure 66 shows the solar panel current comparison between the experiment and the simulation. The peaks verified on the simulated voltage curves also appear in this plot. However, one may note that the solar panel operation point independence from the battery voltage resulted in a much better fitting for this EPS architecture than for the directly coupled circuit. This result reaffirms that the drawback of the directly coupled EPS simulation is the battery modeling inaccuracy.

The simulated battery behavior was even worse for the MPPT

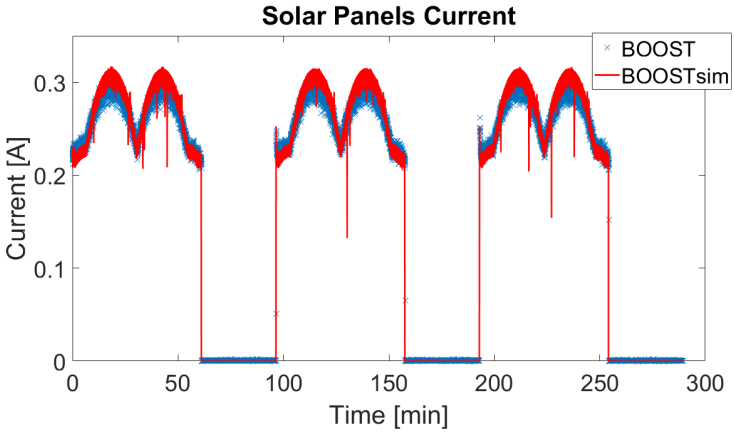


Figure 66: MPPT boost solar panel current simulation and experimental.

EPS boost regulator than for the directly coupled one. Note that, in this architecture, two batteries are connected in parallel. This contributed for the discharging anomaly presented in Figure 67. A second reason for the batteries do not discharge properly in the simulation is the fact that the boost regulator was ideally modeled. Therefore, in the simulation, more energy is entering the batteries, causing them to not discharge as verified on the experiment. A more realistic model for the boost regulator will certainly also improve the battery simulated behavior.

Figure 68 shows the comparison for the solar panel delivered power. The results are better than the simulation for the directly coupled architecture. Power peaks are presented in the simulation, as a result of the voltage and the current behavior, however, the simulated curve is closer to the experimental one.

Figure 69 shows an absolute error analysis on the solar panel delivered power. Also as occurred with the directly coupled simulation, errors are higher on the curve discontinuity points). A better positioning for the experimental and simulated vectors will certainly reduce these peak errors. This occurs because the experiment and the simulation shadowing event are not happening exactly at same time. Besides the visual evaluation, the average error has also been calculated for the MPPT boost regulator architecture. From this calculation, the simulation average power error was 13.2%. Although it is a much better

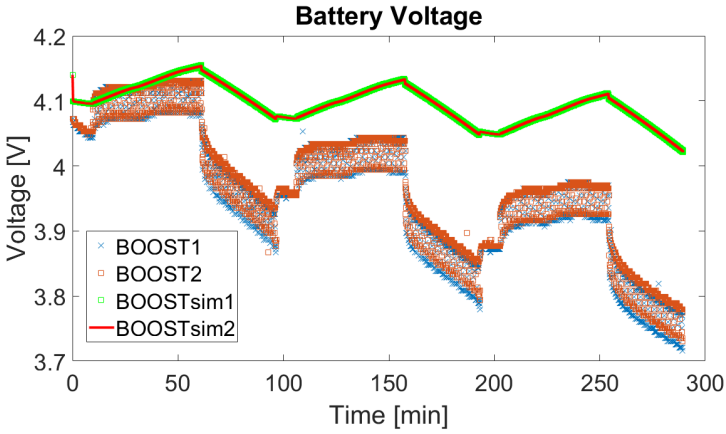


Figure 67: MPPT boost batteries voltage simulation and experimental.

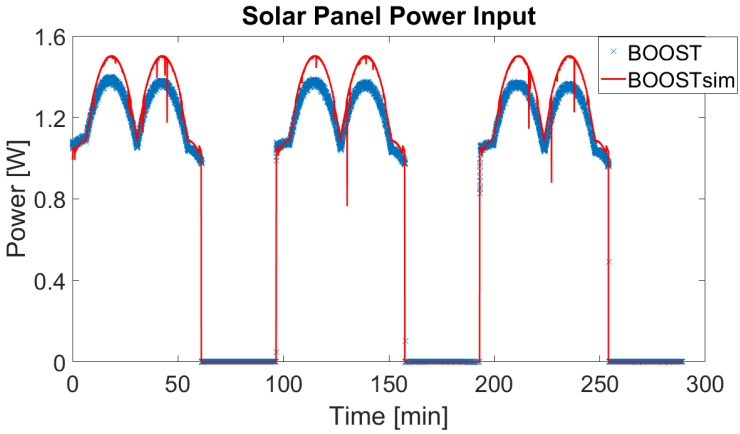


Figure 68: MPPT boost solar panel power simulation and experimental.

result than the obtained for the directly coupled architecture, it may be improved by adopting a better model for the boost regulator.

2.13 CONCLUSION

This chapter has presented an overview on Electrical Power Systems for nanosatellites. It has described four different EPS architec-

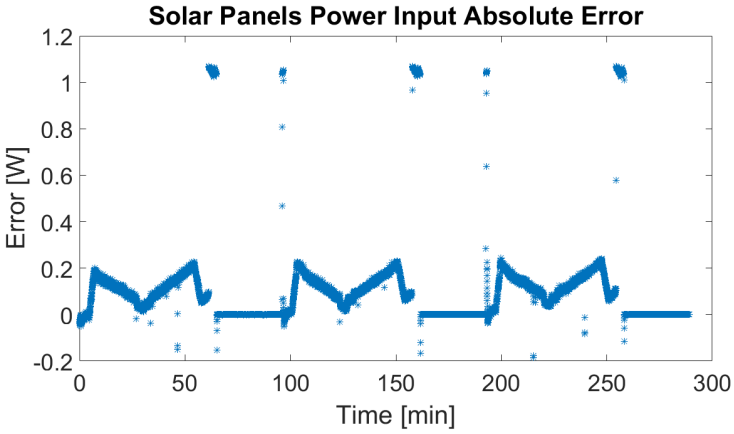


Figure 69: MPPT boost solar panel power absolute error.

tures designed in the context of FloripaSat project, an 1U CubeSat, powered by six solar panels. The architectures have been mathematically described, indicating the hardware influences on the solar panel behavior. Also, their circuit components have been modeled in order to compare their experimental evaluation with their simulated results.

A power input model has been proposed, considering the satellite positioning in orbit. This model allowed preliminary analysis on the satellite energy capability. The solar panels for each architecture have been modeled and experimentally validated. With these two models, a more elaborated system validation has been proposed. A three orbit experiment has been proposed, using high power LEDs to emulate the solar irradiance. Experimental results demonstrated that operating the solar panels on their maximum power point do not necessarily leads to higher efficiency. The higher power losses on the MPPT boost regulator precluded it to achieve better efficiency results. The lack of precision on controlling the perturbation steps has precluded the MPPT IC architecture to maintain solar panels on the MPP. The battery voltage dependence of the directly coupled EPS caused it to operate the solar panels far from their MPP along the first two orbits, drastically affecting its global efficiency. On the other hand, the VLDO has been the most efficient circuit topology, even though it do not perform an active control over the solar panels.

The EPS architectures simulation results were considered satisfactory, since the evaluated parameters were coherent along the three

orbit experiment. The simulation results for the MPPT boost regulator architecture were considerably better when compared to the directly coupled EPS. The main reason for that difference is the inaccuracy on the battery modeling, which strongly affects the directly coupled architecture.

In addition, this chapter mathematically demonstrated that the directly coupled architecture, as well as the VLDO, may have their efficiency improved by controlling the satellite tasks execution. This topic is better addressed in Chapter 4. The laboratory tests successfully accomplish the energy harvesting evaluation for all EPS prototype architectures. However, further tests shall be planned for the selected EPS architecture (and for all the FloripaSat subsystems), in order to avoid fails on the satellite flight model. Therefore, the next chapter presents an in-flight test procedure to validate nanosatellite subsystems on sounding rockets.

3 EPS QUALIFICATION ABOARD SUBORBITAL FLIGHT

3.1 INTRODUCTION

All the tests and simulations presented in the previous chapter are fundamental for a reliable EPS design. They may predict the electrical behavior of the subsystem in orbit, preventing fail conditions along the nanosatellite's mission. However, satellites are submitted to severe conditions both during launching and when operating in orbit. Intense levels of vibration, acceleration, temperature variations, may be faced by the spacecraft. Laboratories with special equipment may emulate these conditions. However, two important factors shall be considered: the tests' facility availability and costs; and the effects of submitting the nanosatellites through these severe test conditions before the flight.

With the emergence of the nanosatellites, there is a growing interest of the universities on participating in space projects. However, most of them have no facilities to perform proper tests in their satellite subsystems. Few university laboratories around the world are able to qualify subsystems to operate in space conditions. Therefore, there is a high cost associated to the qualification process. The nanosatellite qualification may cost even more than its launching. The reason is that many nanosatellites may share the same flight (due to their small size and weight), which drastically reduces the launching costs. On the other hand, the qualification test is individual and expensive. This leads to the critical situation of many nanosatellites missions failing for a lack of more elaborate tests (BOUWMEESTER; GUO, 2010).

Besides this, qualification tests may affect the nanosatellites' operability or even irreversibly damage it. Severe tests may be conducted on spare models, however, this idea conflicts with the nanosatellite low cost approach. The costs and fabrication time to produce two nanosatellites (qualification and flight models) may be off budget for most universities. Therefore, this chapter proposes a test procedure for nanosatellites subsystems using sounding rockets. This procedure suggests an intermediary qualification mission, for the nanosatellite main subsystems, using the engineering model. This motivates the team to obtain an operational flight-ready version of the satellite, which is submitted to all mission phases. In addition, data from real flight conditions may be collected, which are fundamental to identify design fails

before implementing the flight model.

The proposed EPS MPPT boost circuit described in the last chapter has been successfully submitted to a flight qualification, using the strategy described next in this chapter. Although this thesis focuses on the EPS, the proposed test procedure applies also for the remaining subsystems. Therefore, this chapter discusses the test procedure, presenting an embedded system (Muti-Mission Platform) to interface the nanosatellite subsystems and the suborbital rocket electronics. The results of all subsystems tests in flight are presented and discussed.

3.2 STATE OF THE ART

Along with the spread of the small sized satellite concept, a high failure rate has been noticed (BOUWMEESTER; GUO, 2010). Hardware and software design mistakes and failures in the integration process appear among some of the causes. Qualification and verification tests performed at laboratory are intended to avoid this problem (CORPINO; STESINA, 2014). A Platform to guide hardware and software design for small satellites has also been proposed. Through component-based topology, the design platform allows software reutilization, diminishing design time and increasing the system reliability (CAO, 2016). Also, a CubeSat mission design tool to be used in the risk assessment appears as an interesting solution to predict and mitigate failures for small satellite missions (GAMBLE; LIGHTSEY, 2014). Nevertheless, these strategies do not fully cover all the failure possibilities encountered during an actual space mission.

Therefore, some strategies have been proposed to address this issue. Tests with atmospheric balloons (MIYAZAKI; YAMAZAKI, 2013) have emerged as an inexpensive and effective solution to test nanosatellites. Although it allows important preliminary communication tests between the satellite and the ground station, it does not evaluate the satellite behavior under the severe conditions of a rocket launching.

In a more elaborated solution, sounding rockets (MATUNAGA et al., 2000) have been used for preliminary tests of nanosatellites, reducing failures before the mission. In this case, the sounding rocket ejects the nanosatellite after reaching a predetermined altitude. The satellite then communicates with a ground station for a period of time, before hitting the ground.

Although this is a more complete test than the ones performed with the atmospheric balloons, normally, nanosatellites only record/-

transmit data after they are released from the rocket. Actually, this occurs not only in preliminary tests with sounding rockets, but also during real missions. Normally, nanosatellites are activated only several minutes after they are ejected from the rocket. This precludes nanosatellites to record/transmit data during the rocket flight. Many nanosatellite failures may occur during the rocket flight and may not be analyzed and understood due to the lack of information in this critical period of the mission. Several nanosatellites do not transmit after the rocket ejection procedure and the researchers have no idea about what may have caused the failure(s).

In this chapter an innovative procedure to test nanosatellites subsystems on board sounding rockets is presented. Instead of ejecting the nanosatellite, the proposed test is completely performed aboard the sounding rocket. This allows the nanosatellite to record data during all the phases of the rocket flight (including the launching). The nanosatellite starts operating before the rocket lift-off and it remains operating along the whole flight. The satellite data are transmitted to the rocket electronics, which retransmits the data to the ground station. With this strategy, one may have access to nanosatellite failures information that may occur during the rocket flight. Temperature increases during flight, high acceleration levels, cables issues due vibration, powering fails due to battery damage, are among some problems that may occur during launching and flight, which may be detected with the proposed test procedure.

Besides the procedures of nanosatellite's subsystem configuration and integration, this chapter also describes the design of the Multi-Mission Platform (MMP), which is an embedded system conceived to allow nanosatellite tests (and other experiments) aboard sounding rockets. Also, the ground station decoder software is presented here.

Finally, the chapter presents the results of the FloripaSat (1U CubeSat) subsystems tested aboard a VSB-30 sounding rocket. Three engineering model subsystems have been tested: Electrical Power Subsystem (EPS); Telemetry, Tracking and Command (TT&C); and On Board Data Handling (OBDH). The flight data of each subsystem are presented and discussed. This analysis will prevent failures on FloripaSat flight model, as well as, it will guide the design to improve some of its features.

3.3 MULTI-MISSION PLATFORM

Sounding rockets have an electronic system (rocket electronics) responsible for receiving data from the payload experiments, and for transmitting them to a ground station. Therefore, any experiment which intends to transmit data to the Earth during the flight, shall communicate with the rocket electronics. The rocket electronics then sends the experiment data to a ground station through radio signal.

On the other hand, nanosatellites may communicate with ground stations by themselves, using their own radio and antenna. However, no external antennas (except the rocket main antenna) have been allowed for our sounding rocket flight. This has precluded the direct test of the TT&C radio system. Consequently, the nanosatellite data had to be sent to the rocket electronics, which transmitted the data via radio to the ground station, through the rocket main antenna.

In case of sounding rocket missions that allow the satellite to use its own antenna, the flight test of this subsystem could be completely accomplished, but other factors as rocket shielding and antennas interference should be considered.

Another important information is that in a rocket mission the payload is not always rescued after flight. Therefore, sending data through telemetry is extremely recommended for experiments tested on sounding rockets. Simultaneously, as a redundant option, the experiments' electronics shall save information in their own internal memory, for the case of accessing them later. From the experience obtained in previous sounding rocket missions, telemetry is also not always fully received and the internal memory may be the only way to obtain the flight data(PAIVA; MANTELLI; SLONGO, 2015)(PAIVA et al., 2008).

Beside this, the experiments are strongly recommended to provide communication with the bunker, through a specific cable named umbilical. Figure 70 shows the ideal pre-flight experiment electronics configuration regarding the communication with the rocket and the bunker.

With this in mind, there were two design options for our case: Either we modified the nanosatellite hardware and software to directly communicate with the rocket electronics and with the bunker, or we could create an intermediate embedded system to attend the sounding rocket mission specifications. Modifications on the nanosatellite design would imply in testing a different setup than the final version of the satellite. In addition, it is not recommended to make the nanosatellite more complex in order to test it. Therefore, a dedicated embedded

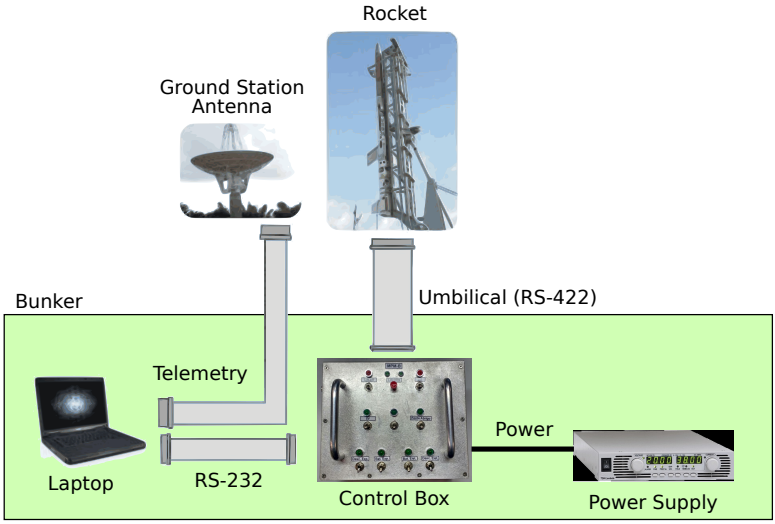


Figure 70: Rocket and bunker connections

system has been designed to test the nanosatellite subsystems (and other experiments) in sounding rockets. We named it the sounding rocket Multi-Mission Platform (MMP), shown in (Figure 71).

Besides the capability of testing nanosatellites, the Multi-Mission Platform is intended for scientific thermal experiments(PAIVA et al., 2010)(PAIVA et al., 2008). The embedded system consists of a stack of three printed circuit boards: power; data acquisition; and processing. It is beyond the scope of this work to describe in details the Multi-Mission Platform. However, some characteristics are important to be mentioned, in order to comprehend the nanosatellite subsystems test procedure and the needed hardware/software resources to accomplish this goal.

According the mission requirements, the communication between the experiment electronics and the bunker shall be through RS-422 standard due to the distance between the bunker and the rocket launching platform (See Section 3.4.4 for detailed information on the communication protocol). Besides this, there shall be a second communication channel (between the experiment and the rocket electronics). For safety reasons both communication channels must be isolated, avoiding any eventual electrical instability to be propagated to the rocket electronics. This is the first reason to have a Multi-Mission Platform, since this kind of communication is not implemented in nanosatellites.

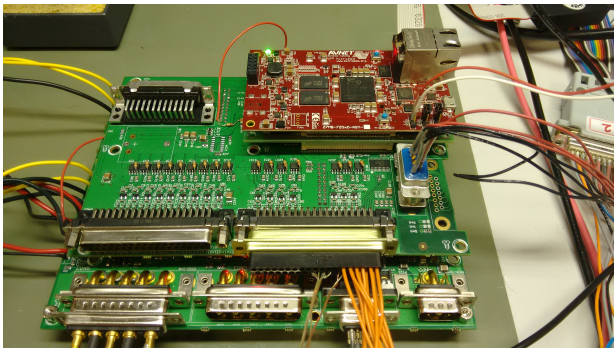


Figure 71: Multi-Mission Platform under test

The communication transceivers (ADM2682E) have been placed on the MMP data acquisition PCB. Figure 72 shows the transceivers electrical schematic, where signals with the label *uZed* come/go to/from the MMP processing PCB.

Also in the mission requirements it is stated that the experiment electronics must present a safety mechanical turn on/off system. The power on command shall come from the bunker, using a control box. This is not a requirement for nanosatellites, but for the rocket mission. Therefore, the Multi-Mission Platform is provided with an isolated push button circuit to switch it on and off. Figure 73 shows the push button electrical schematic. Figure 74 shows the control box, placed at the bunker.

Finally, the rocket electronics keeps all the experiments informed regarding the lift-off and the microgravity condition. This is achieved by switching two pins/lines to ground (lift-off and uG pins). It is extremely important to the experiments to obtain these events triggering time reference in order to map the experiments results with the flight dynamics. Therefore, the Multi-Mission Platform has an analog circuitry to identify the events, as shown in Figure 75. Besides the analog circuitry, the embedded software is responsible for including the events information on the data frame sent to the ground station (flight status byte).

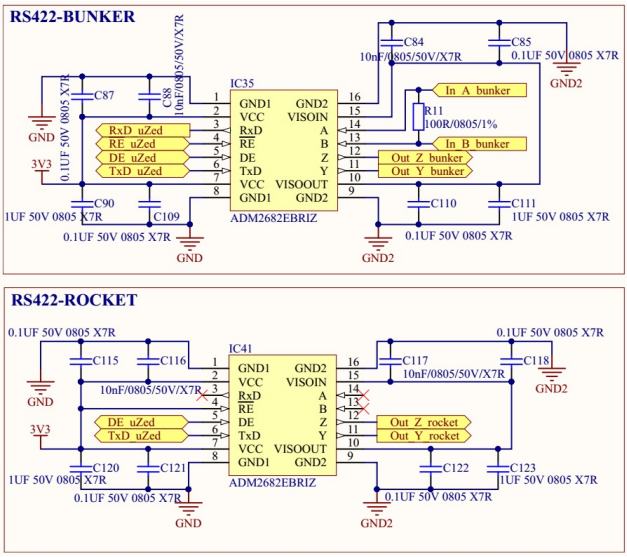


Figure 72: RS-422 transceivers electrical schematic

3.4 THE EXPERIMENT

The experiment to be performed under the sounding rocket is called New Medium Porous Technologies for Phase Changing Devices (MPM-A). It is carried out by an instrument which is implemented in a machined aluminum box with three compartments. The lower compartment contains both batteries (the thermal experiment batteries and the FloripaSat batteries). The second compartment is entirely dedicated to the thermal experiment (Phase Changing Devices). The higher compartment contains the MMP and the FloripaSat subsystems. The experiment's goal is to test a thermal device, as well as the nanosatellite subsystems, using a Multi-Mission Platform as an interface for the devices under test and the rocket electronics. This section focuses on explaining the FloripaSat subsystems design and configuration procedures in order to test them aboard the sounding rocket. Figure 76 shows an overview of the experiment electronics internal and external communication.

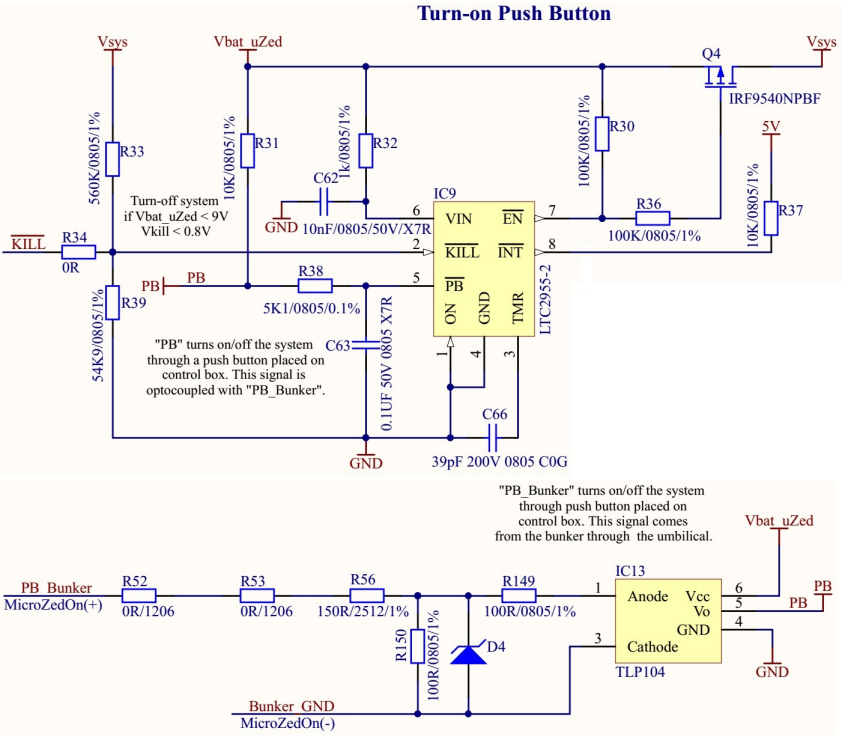


Figure 73: Push button turn on/off electrical schematic

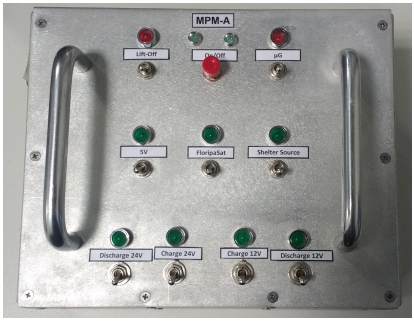


Figure 74: Control box

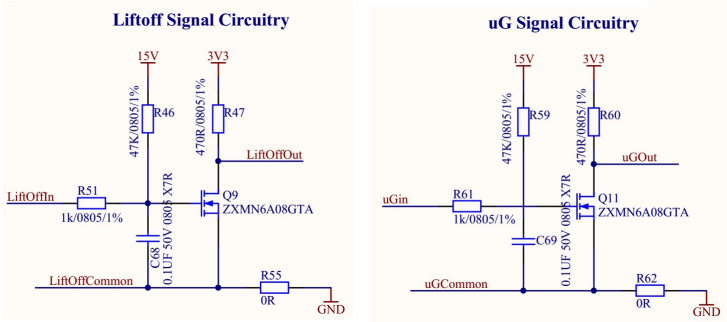


Figure 75: lift-off and uG circuitry

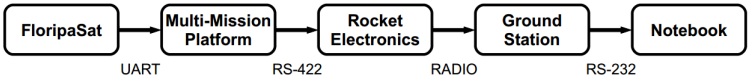


Figure 76: Experiment electronics communication diagram

3.4.1 FloripaSat architecture

Three FloripaSat subsystems have been tested aboard VSB-30 rocket: OBDH; TT&C; and EPS. The OBDH and the TT&C subsystems are located on the same PCB while the EPS is on a dedicated one. Combining functions of two different subsystems into a single subsystem/PCB is not a new idea and has been proven to be an effective approach for small satellites (VARATHARAJOO; FASOULAS, 2002) (ADDAIM; KHERRAS; ZANTOU, 2008). The flight model of FloripaSat shall have subsystems communicating with each other through different interfaces as shown in Figure 77.

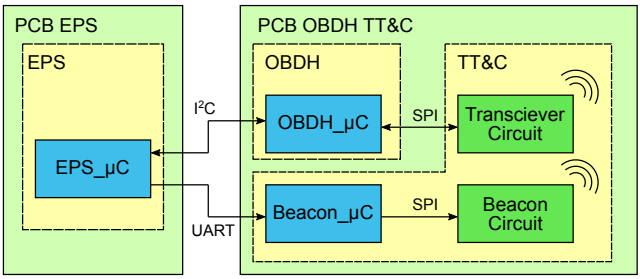


Figure 77: FloripaSat flight model architecture diagram

Normally, the OBDH shall build up the data frame and send it to the TT&C subsystem (via SPI), which sends the data to the ground station (via radio transceiver). Also, in case the satellite receives a telecommand (via radio transceiver), the TT&C subsystem sends the data to OBDH (via SPI) in order to process it. However, as mentioned in Section 3.3, it was not allowed to place an external dedicated antenna to directly test the TT&C communication. Therefore, a different communication configuration has been planed for the suborbital flight test with no hardware modifications made on the nanosatellite subsystems. For the sake of experiment electronics simplicity, only minor changes on the communication protocol were performed (see Sections 3.4.3 and 3.4.4).

The idea is to transmit data internally from the Beacon Radio to the Transceiver Radio. Even without the antennas, the radio frequency circuits should be able to send and receive data to each other, due to their proximity (both circuits were placed at the same PCB, as shown in Figure 77). The Beacon microcontroller generates data (a simple 2 byte counter) and sends it through SPI to the Beacon RF circuit. The data are transmitted to the Transceiver Circuit, which receives the RF signal, decodes it, and sends the data to the OBDH microcontroller via SPI. After receiving the Beacon data, the OBDH requires EPS data (via I^2C main bus). Finally, the OBDH microcontroller adds up its own information to the data frame and sends it to the Multi-Mission Platform Processing PCB via UART. Then, the data frame is sent to the rocket electronics via RS-422. Figure 78 shows the test configuration for flight.

3.4.2 Mission frame definition

Before discussing in details the FloripaSat's telemetry data, it is important to recall that the Multi-Mission Platform acquires data not only from the FloripaSat subsystems, but also from a thermal experiment, as previously stated. Before sending the data to the ground station, the Multi-Mission Platform packs the thermal experiment data (174 bytes) and the FloripaSat data (42 bytes) into a multiple frames containing 432 bytes (216 payload bytes) every sampling period. The Multi-Mission Platform transmits the acquired data split into 72 frames of 6 bytes. The first 58 frames concern the thermal experiment data frame and the last 14 frames concern the FloripaSat telemetry data. Further discussions regarding each of the thermal experiment data is

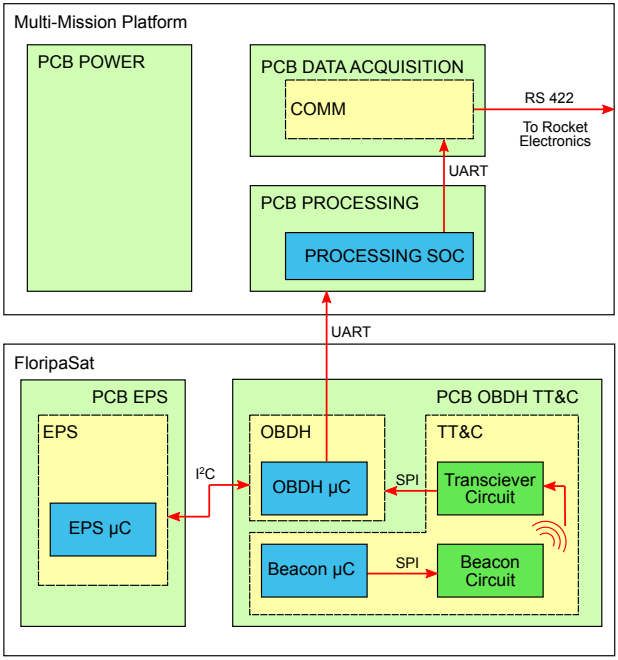


Figure 78: FloripaSat engineering model architecture diagram for the suborbital test

beyond the scope of this work. Therefore, the focus is kept on the FloripaSat. Figure 79 shows the structure of the data frame transmitted by the Multi-Mission Platform to the ground station.

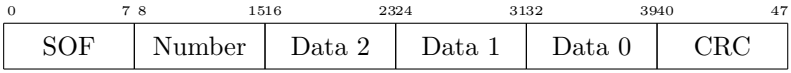


Figure 79: Data frame structure with the corresponding bit number above each field.

The Start of Frame (SOF) is the first byte transmitted to indicate to the ground station the beginning of a new frame and it is followed by the frame number used to identify the data field transmitted. The 3 following data bytes are the payload itself. The MMP generates a Cyclic Redundancy Check (CRC) considering 4 bytes of each frame: Number and the 3 data bytes. Due to the unreliable link between the ground station and the sounding rocket, the frame has only 3 payload bytes. By doing so, we avoid discarding a larger payload in case of data corruption at the expense of the overhead in each frame.

3.4.3 FloripaSat telemetry data

Unlike the thermal experiment data, the FloripaSat telemetry data are not generated at the Multi-Mission Platform. It is the nanosatellite's OBDH task to acquire data from subsystems and sensors, to generate an internal frame and to send it to the MMP through UART. Notice that the FloripaSat frame is then split by the MMP and encapsulated in its own frame previously described.

Since the FloripaSat OBDH main function is to control the nanosatellite tasks execution and build up the transmitted data frame, only few bytes were necessary to evaluate its own performance. Three OBDH features were tested with no need to include information on the data sent: 1 - receiving data from sensors and from other subsystems; 2 - the data frame generation routine; 3 - the communication with the transceiver. Additionally, an internal counter (two bytes for seconds and two bytes for milliseconds) has been implemented in the OBDH microcontroller. The counter data was sent as an internal parameter. Also, two bytes from microcontroller internal temperature have been included in the main data frame. Finally, a status byte has been generated, signaling eventual microcontroller anomalies in tasks execution.

The OBDH subsystem also has an Inertial Measurement Unit

(IMU), whose data have been included in the data frame sent by the FloripaSat. Both acceleration and rotation have been measured, in three different axis. Every measurement consists of two bytes, totaling twelve bytes of information.

The main EPS subsystem functionality is to control the battery charge and discharge process, as well as to ensure the proper power distribution for all the nanosatellite subsystems. Normally, EPS subsystems have a battery monitoring chip, circuit or subroutine. Therefore, the piece of information selected to evaluate the EPS performance was related to the battery monitoring. Input power is also an crucial aspect to be monitored, however, no external solar panels were allowed for the VSB-30 flight. Therefore, regarding the power input test, only the battery charge procedure (established on ground, before the rocket lift-off) has been monitored. Eleven bytes have been reserved for the EPS subsystem, as follows: batteries electrical input/output current (two bytes); batteries voltage (four bytes); battery monitoring chip internal temperature (two bytes); battery remaining electrical charge (two bytes); and finally the EPS status (one byte).

It is important to mention that the FloripaSat internal data frame integrity is firstly verified by the Multi-Mission Platform (FloripaSat - MMP UART communication integrity verification). Later, in a second verification level, the ground station processing software verifies the mission main data frame (rocket electronics - ground station radio communication integrity verification - see Section 3.5).

Therefore, in order to the MMP to identify the FloripaSat data frame (higher level verification), three bytes were defined as Start of Frame, and three bytes as the End of Frame (EOF). These bytes allow the ground station decoder software to correctly identify the beginning of new upcoming data frames, after receiving an eventual broken frame. Lastly, an eight bit CRC has been implemented to identify corrupted data frames. The list containing all field in FloripaSat data frame is shown below.

1. Start of Frame (SOF), 3 bytes: Start of Frame delimiter.

2. OBDH timestamp (s), 2 bytes: seconds representation of OBDH counter.

3. OBDH timestamp (ms), 2 bytes: milliseconds representation of OBDH counter.

4. OBDH temperature, 2 bytes: OBDH internal temperature.

5. OBDH status, 1 byte: OBDH internal status.
6. IMU acceleration x-axis, 2 byte: IMU acceleration in the x-axis.
7. IMU acceleration y-axis, 2 byte: IMU acceleration in the y-axis.
8. IMU acceleration z-axis, 2 byte: IMU acceleration in the z-axis.
9. IMU angular rate x-axis, 2 byte: IMU angular rate in the x-axis.
10. IMU angular rate y-axis, 2 byte: IMU angular rate in the y-axis.
11. IMU angular rate z-axis, 2 byte: IMU angular rate in the z-axis.
12. Radio counter 1, 2 byte: TT&C most significant transmitted data counter.
13. Radio counter 2, 2 byte: TT&C less significant transmitted data counter.
14. Batteries current 2, 2 byte: Current drawn from batteries.
15. Battery voltage 1, 2 byte: Voltage on battery 1.
16. Battery voltage 2, 2 byte: Voltage on battery 2.
17. EPS temperature, 2 byte: Internal EPS temperature.
18. Electrical charge, 2 byte: Battery remaining electrical charge.
19. EPS status, 1 byte: Status register for the battery protection circuit.
20. CRC, 1 byte: CRC of all previous field except for SOF.
21. End Of Frame (EOF), 3 bytes: End of frame delimiter.

3.4.4 Communication protocol specification

VSB-30 electronics receives data from payload by RS-422 standard. This standard is not common in nanosatellite applications due to its voltage requirements. Instead of including an RS-422 driver in the FloripaSat, in order to attend the rocket requirements, it was placed in the Multi-Mission Platform (see Section 3.3).

Two different communication channels have been implemented in the Multi-Mission Platform (see Figure 72). The first one is dedicated to the communication between the bunker and the experiment

electronics, before the rocket lift-off, where the main limitation on this channel is the distance between the bunker and the rocket (which is below the RS-422 standard limitation of 1500 m). This is a point to point channel and could be taken to the physical bandwidth limits defined by the standard. The second channel concerns the communication of the experiment and the rocket electronics used to transmit the data to the ground station. It is limited in bandwidth due to all the payload experiments on board the rocket using the same channel. The maximum allowed baud rate for each experiment on board was 57,600 bps.

The Multi-Mission Platform receives FloripaSat data frames through a UART port. There is no synchronization protocol between them but, to ensure that no data are lost due to overflows, the Multi-Mission Platform defines a maximum UART baud rate and guarantees that below that limit, it is able to store and transmit all the received data frames. The Multi-Mission Platform initially parses the received data to ensure it follows the data frame specification presented previously. It initially searches for the SOF bytes and then it buffers all the data it receives, until it finds the EOF bytes. Next, it checks if the CRC is valid. If it is, the data frame is stored for transmission with the rest of the experiment data. In case of a mismatch in the CRC, the data frame is discarded.

3.5 ON-GROUND DATA PROCESSING SOFTWARE

In order to send as much information as possible to the ground station, both the thermal experiment and the FloripaSat data were transmitted in raw format. As explained in Section 3.4.2, the data were packaged to frames with SOF, Number and CRC bytes. Therefore, a software application was implemented to allow data unpacking, processing and logging at the bunker during the rocket flight. The software was based on LabVIEW language, running in a notebook placed at the bunker.

As mentioned previously, the mission main data frame has 432 bytes (72 frames of 6 bytes) where the last 84 (14 frames of 6 bytes) are the ones related to the FloripaSat. Therefore, the LabVIEW software has been designed to identify the FloripaSat frames in order to unpack and process them. Since each frame contains 3 bytes of data (the other 3 bytes are SOF, Number and CRC), there are 42 data bytes in total which contain information from the FloripaSat (payload bytes). These are the bytes that must be converted by the LabVIEW software into

readable information to be saved during the rocket flight.

Figure 80 shows the FloripaSat data frame extraction from the mission data frame. It occurs in two parallel paths. The first one (upper part of Figure 80), extracts the frames 60 to 71, which are the FloripaSat data frames excepting the EOF and the SOF. These 12 frames go to the FloripaSat subVI in order to be processed. This VI converts the data, sends them to the LabVIEW Front Panel (column processed data, see Figure 81). After that, the converted data are saved in a .csv file. The second path extracts frames 59 to 72, which contains all the FloripaSat data (including EOF and SOF). This vector is directly sent to the LabVIEW Front Panel (raw data column, see Figure 81). After that, the raw data are saved in a .csv file.

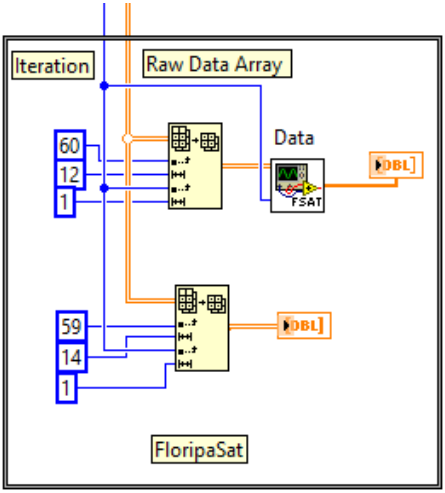


Figure 80: LabVIEW block diagram of FloripaSat data unpacking process

Figure 81 shows the LabVIEW graphical interface block related to the FloripaSat data. Two vectors can be seen: the one on the left hand represents the data after the conversion, and the one on the right hand represents the raw data. These vectors are updated to each received frame.

Figure 82 shows the FloripaSat subVI. This block receives the mission frames 60 to 71 (72 bytes - 12 frames of 6 bytes) and extracts only the FloripaSat 36 payload bytes (12 frames with 3 payload bytes

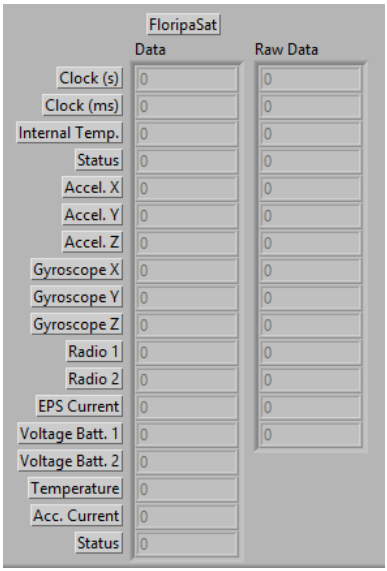


Figure 81: LabVIEW front panel for the FloripaSat received data

each). This process removes the frame bytes SOF, Number and CRC. This is achieved through a for loop, which does 12 iterations: each iteration receives a frame, extracts only the 3 payload bytes from it and allocates them to a new vector. At the end of the 12 iterations, the 36 payload bytes are stored in a vector that is forwarded to the next processing block.

The second processing block receives the vector with 36 bytes and, through another for loop, rearranges them in a vector with 18 positions. This reorganization is performed in order to generate the 18 data observed in Figure 81: timestamp, accelerometer, gyroscope, current, voltage, etc. In addition to rearranging the data, some of them go through formulas in order to be converted to meaningful data. This 18 position vector is the output of the FloripaSat subVI. It is reinserted into the mission data array (which also contains the processed data from thermal experiment) and then saved to a file. Therefore, the on-ground data processing software allows the converted and raw data being visualized during the flight and saved in files for further analyses.

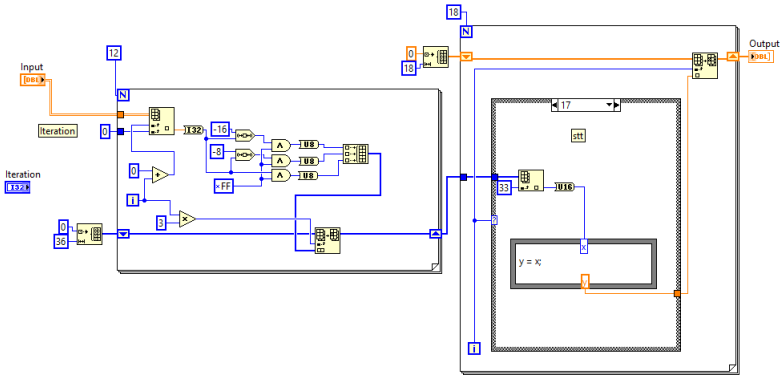


Figure 82: LabVIEW subVI of FloripaSat data decoder

3.6 INTEGRATION PROCESS

Nanosatellite subsystems integration to the MPM-A instrument has occurred in three levels. First, the FloripaSat subsystems have been integrated one with each other. Then, the subsystems were integrated with the MMP. Finally, the FloripaSat subsystems and batteries were integrated in the MPM-A machined aluminum box. The first integration level started with the nanosatellites subsystems placed side-by-side, connected by wires, with their software running in debug mode. Initially, powering tests have been conducted, using voltages supplies instead of batteries, in order to certify that the EPS was correctly suppling power to the OBDH and TT&C subsystems. After passing the preliminary powering tests, the FloripaSat batteries have been assembled. They consist of two lithium ion batteries connected in series (ICR18650-30A) and covered in silicon. Three wires were routed from the batteries (lower instrument box compartment) to the subsystems (higher instrument box compartment). The wires were connected to batteries at GND, intermediate voltage, and total voltage points.

After that, the FloripaSat PCBs internal communication was tested. To achieve this, a simple periodic routine was implemented to send a vector of fixed bytes from EPS to OBDH through I^2C protocol. To verify the correctness on this internal communication, the OBDH was configured to send the vector of bytes to a computer, through UART. After validating the internal communication, the EPS software has been modified to acquire real data from its sensors. Also, the

OBDH included real sensors data to the FloripaSat data frame sent to the computer. Finally, the communication between TT&C and OBDH has been validated. A counter has been implemented on the Beacon microcontroller, for each performed transmission to the transceiver. The SPI communication between these two subsystems has been validated by checking the counter increases on the final data vector sent from the OBDH to the computer.

After that, the integration with the MPP initiated. Both electronics were placed side-by-side and the activating system has been wired up from MPP to FloripaSat subsystems. It is important to understand how this solution has been implemented. Nanosatellites are normally turned on by deployment switches attached to the satellite's structure (The CubeSat Program, 2014). These switches remain deactivated when the satellite is inside the launcher, electrically disconnecting the satellite from the others. Once the satellite is launched, these switches electrically close and the satellite is powered on.

As shown in Figure 83, two MOSFET-P are connected to the deployment switches and to the RBF to turn off the subsystems power. When the RBF or the switches are activated, the P-channel MOSFETs do not allow the current to flow from the batteries to the load. As the P-channel MOSFET only conducts when it has a low level voltage in its gate, a NPN transistor was added between the MPP enable pin and the EPS P-channel MOSFET. This allowed the Multi-Mission Platform to enable the FloripaSat subsystems, as shown in Figure 84. Although, this seems a critical hardware modification, the implementations have been executed on the MPP side, therefore, there were no needed to change the EPS circuit.

After that, the FloripaSat was connected with the Multi-Mission Platform to test the switch on procedure and communication through UART. Data generated by FloripaSat were sent to the Multi-Mission Platform and then sent to a computer running the LabVIEW data processing software, which allowed the debugging of the FloripaSat running the release software version. During this phase of the integration, software bugs were found in the LabVIEW software which led to additional test procedures, such as the usage of an oscilloscope to verify the frequency which data were being sent by the OBDH.

Then, the flight version of the software was programmed into the boards before assembling them, since the programming connectors were not accessible after mechanical integration. After this step, the FloripaSat PCBs were stacked. The electrical connection among boards is performed through a PC-104 style connector. Metallic spacers have

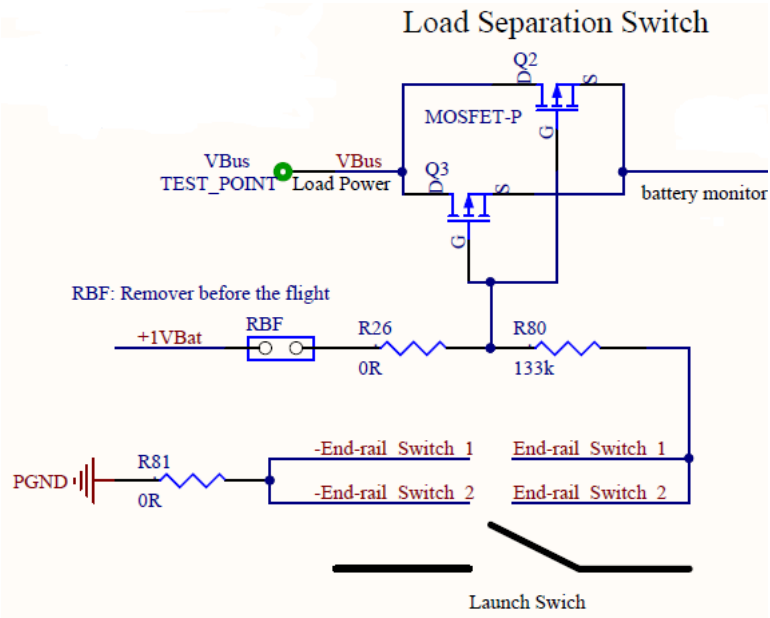


Figure 83: Original switching on circuit from FloripaSat

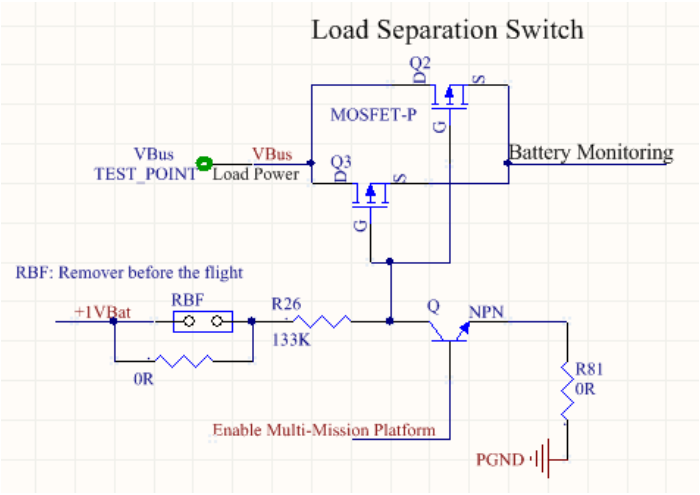


Figure 84: Modified switching on circuit to turn on FloripaSat subsystems by MMP

been screwed between PCBs to guarantee mechanical stability. The connections between FloripaSat subsystems and the MMP were made by soldering wires. To increase the mechanical stability of electronic components on the PCBs, silicone was poured over the most sensitive areas of all subsystems. Figure 85 shows a photo of the FloripaSat OBDH, EPS and TT&C subsystems under integration.

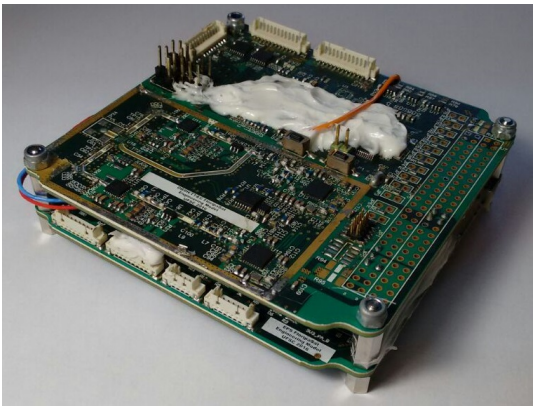


Figure 85: FloripaSat OBDH, EPS and TT&C subsystems under integration

After this test procedures, the FloripaSat subsystems were attached to the top part of the MPM-A aluminum case. Also, the batteries have been placed at the lower compartment of the aluminum case. Figure 86 shows a photo of the MPM-A instrument under integration, just before closing it.

3.7 ACCEPTANCE TESTS

After the integration process, carried out at the Federal University of Santa Catarina, the MPM-A has been sent to the Department of Aerospace Science and Technology (DCTA, in Portuguese), from the Brazilian Air Force, to be submitted to flight acceptance tests. This test procedure differs from the flight qualification tests, where the experiments are submitted to similar conditions to the rocket flight. The reason for applying only acceptance tests (less severe) instead of qualification tests, is that such research experiments normally do not have spare sets for severe tests that may cause irreversible damages to them.

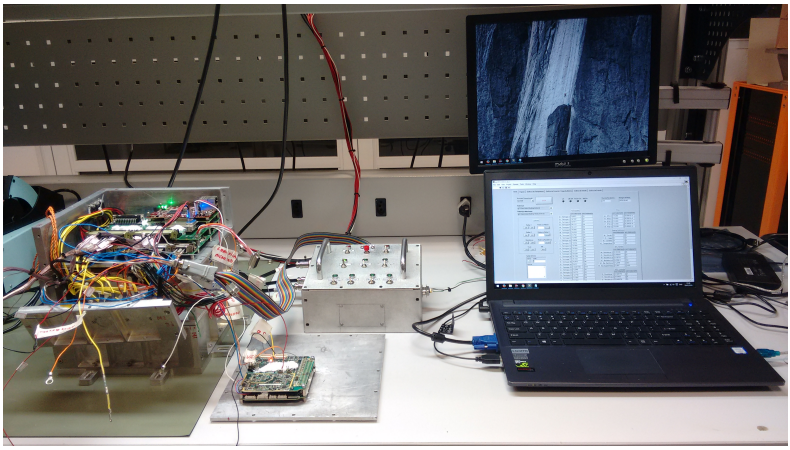


Figure 86: FloripaSat subsystems under integration with the MPM-A instrument

The lacking of spare sets are normally caused by the experiments' complexity, high cost and long development time.

The same situation applies to nanosatellites. The intrinsic idea of low cost and fast development time, comparing to bigger satellites, precludes the research groups of fabricating two identical versions of their spacecrafts. Therefore, the nanosatellites flight model are normally not submitted to severe qualification tests. This may also be one of the causes for a high failure rate on this class of satellites.

Therefore, this work suggests firstly submitting the nanosatellite subsystems engineering model to an acceptance test, which is faster, cheaper and less degrading. Then, testing these subsystems under flight, with the subsystems operating along the whole flight, recording data on severe and real flight conditions. Engineering model subsystems are faraway cheaper than the whole integrated flight model nanosatellite, which includes mechanical structure, solar panels, antenna, etc. In case the subsystems fail or crash at any moment on the flight, the data are recorded and may be further analyzed, avoiding committing the same design mistakes on the flight model.

After this level of tests with the engineering model subsystems, the flight model may be more reliably designed, and then the acceptance tests will be enough to preclude the satellite failures on the real mission.

Besides this, having an in-flight test during the satellite design phase motivates the team and sets up hard deadlines which are important for students to accomplish. Figure 87 shows a flow chart summarizing the proposed test procedure. Each box represents a step on the procedure, which was described in more details through the chapter's sections.

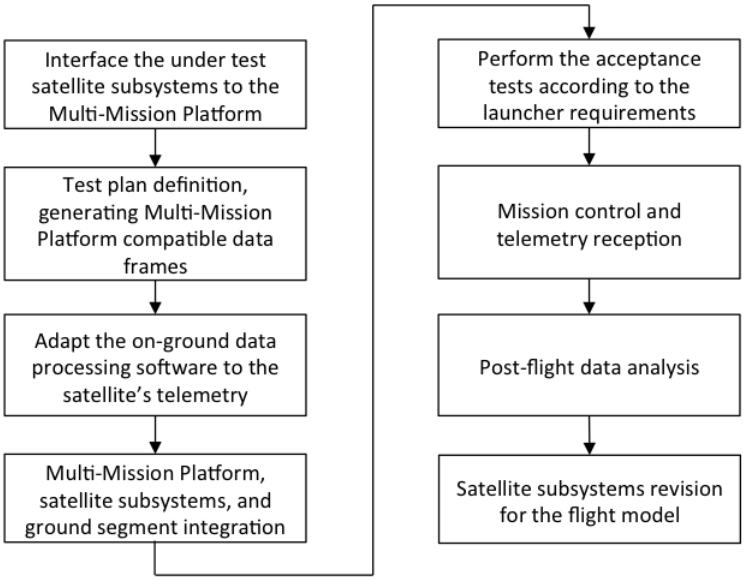


Figure 87: Proposed test procedure

Figure 88 shows the MPM-A instrument right before the acceptance tests at the DCTA. Due to military restrictions, it was not possible to take pictures of the MPM-A instrument under acceptance tests. The performed tests were: mass test; fit test; and vibration test. Vacuum tests were not needed because the rocket payload modules were pressurized and hermetically sealed.

After the MPM-A instrument has passed the acceptance tests, the rocket payload and its electronics were integrated and submitted again to a new acceptance test. Next, when the rocket payload has passed through all the acceptance tests, it was partially disassembled to be sent to the Alcântara Launching Center. After the final integration at the launching center, the rocket was ready for flight.

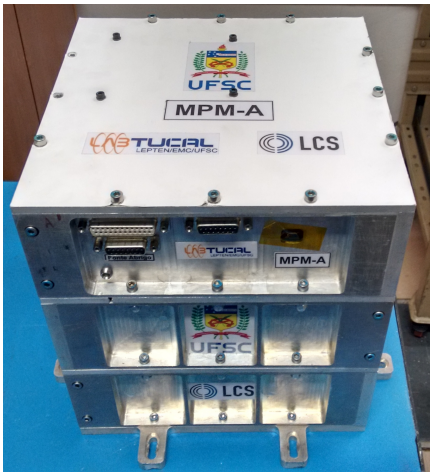


Figure 88: MPM-A instrument after acceptance tests - ready for flight

3.8 FLIGHT RESULTS

This section is dedicated to present the nanosatellite results during the sounding rocket mission. The intention is to verify the nanosatellite operation under severe conditions, once the nanosatellite was operating even before the rocket lift-off.

However, before analyzing the flight data, it is important to briefly describe the mission phases, in order to correctly interpret the results.

- **Phase 1** - The MMP and the FloripaSat have been turned on with the rocket at the launching platform 30 minutes before the rocket lift-off. At this moment, there was a physical communication between the MMP and the bunker, through the umbilical cable. This allowed receiving data from the FloripaSat, externally powering it and also charging its batteries (see Figure 70).

- **Phase 2** - Three minutes before the launching, the external powering process was intentionally interrupted, due to safety reasons, as a mission requirement. However, the MMP and the FloripaSat remained operating (powered by their batteries only) and sending data to the bunker through the umbilical cable.

- **Phase 3** - At the lift-off moment, the umbilical cable was discon-

nected and the rocket entered on flight mode. From this moment on, data have been sent through the rocket radio communication system only. Flight data were received by the ground station antenna and sent to the bunker to be processed by the on-ground data processing software.

- **Phase 4** - After nearly 1 minute the rocket should had entered the microgravity phase. Unfortunately, due to the rocket malfunctioning, which is not related to the experiments, this phase was not observed in this flight, and the respective related data are not available.

- **Phase 5** - After less than 3 minutes the rocket payload parachute system has been activated and the payload safely landed on sea nearly 6 minutes after the lift-off.

The rocket electronics generates two important signals to the payload experiments: lift-off and microgravity signals. These signals are important time references for data analyses. However, since there was no microgravity condition in this flight, only the lift-off signal has been received on the MMP. In our experiment, besides the lift-off signal time reference, the external powering interruption (which has occurred before the lift-off) has also been registered in order to be considered on mission data analyses. The parachute activating system generates no signal to the experiments, therefore, there is no specific time reference for this flight event.

The first results to be analyzed are the FloripaSat EPS data, acquired before and during the flight by the batteries monitoring chip. This chip communicates with the EPS microcontroller and measures batteries voltages, batteries current, batteries remained electrical charge and temperature. The acquired data are sent from EPS microcontroller to OBDH microcontroller through the I^2C protocol, as shown in Figure 78.

Figure 89 shows FloripaSat batteries voltage behavior along the time. During the phase that FloripaSat was operating with the rocket at the launching platform (nearly 27 minutes), one may note that both batteries remained nearly at the same voltage level. When the external powering process was interrupted, the batteries voltage instantaneously dropped, but remained on the expected operating level. Also, it is possible to identify a slightly voltage drop for both batteries during the flight, due to the batteries discharging process. No unexpected behavior has been identified on this result.

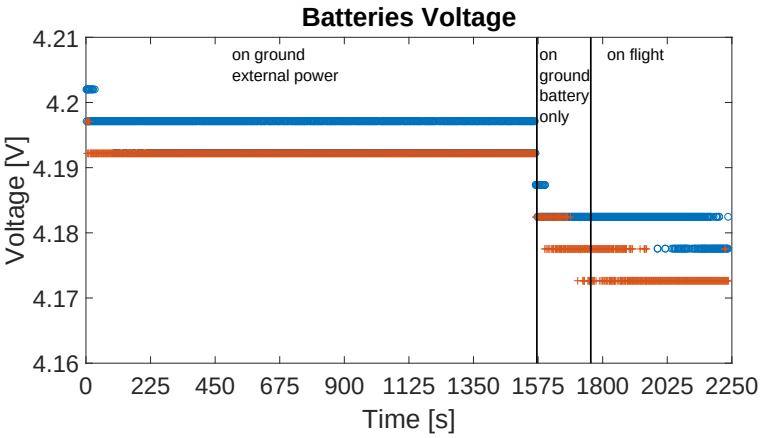


Figure 89: Batteries voltage

Figure 90 shows the FloripaSat instantaneous batteries current behavior during the mission. Negative currents represent the batteries discharging behavior. This plot shows that, even during the external powering process, batteries current remained negative. The first reason for this behavior is that the subsystems demanded more current than provided by the external power supply. The second reason was the adopted battery charging methodology, which is constant voltage charging. It is well known from the literature that constant voltage is not the most appropriate way to charge lithium-ion batteries (CHAOUI; GUALOUS, 2017), however, this was the safest manner to charge the batteries at that critical moment. Constant current charging could lead the batteries to over-voltage state (there was no external charging control system available), which would activate the batteries protection system minutes before the flight. In order to avoid this unnecessary risky condition, a most conservative approach has been chosen.

This charging issue faced during the mission has led to a charging circuit design, which will be used for batteries external charging procedure before the FloripaSat flight. Also, the research group is testing a constant current charging circuit block to be implemented between the solar panels and the battery. This could reduce the EPS harvesting energy capability, but extend the batteries lifetime. Therefore, this different architecture is under consideration for future EPS designs.

Figure 90 also shows that, after the lift-off, the batteries current

remained constant along the whole flight, on the expected level of nearly 23 mA (global current consumption of all modules).

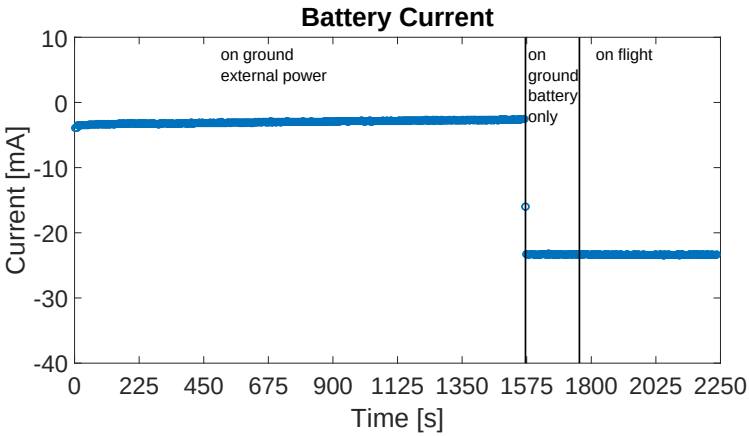


Figure 90: Batteries current

Figure 91 shows the batteries remaining electrical charge before and during the flight. The battery monitoring chip is provided with an instantaneous current integrator, which allows the EPS to obtain batteries state of charge information. This data is extremely valuable to the FloripaSat in order to schedule its tasks. Figure 91 shows a slightly decrease on the battery remained electrical charge before the flight. This occurs, as explained above, due to the subsystems higher consuming current than the external supply current. After the external powering interruption, one may note that the batteries discharging rate has increased. Also, this figure shows that during the flight, the batteries discharging rate was the same than before the flight, since the subsystems power consumption has remained the same.

Figure 92 shows the batteries monitoring chip internal temperature. A considerable temperature increase may be noted in this plot. Two are the main reasons for this effect: the first one is that the thermal experiment dissipates heat to the instrument aluminum casing (pre-heating operation mode), which leads to an overall temperature increasing inside the electronic compartment; the second reason is the chip internal self heating due to its operation. A relevant observation here is that the temperature increasing rate was higher after the launch. Again, two reasons may have contributed to this: the first one

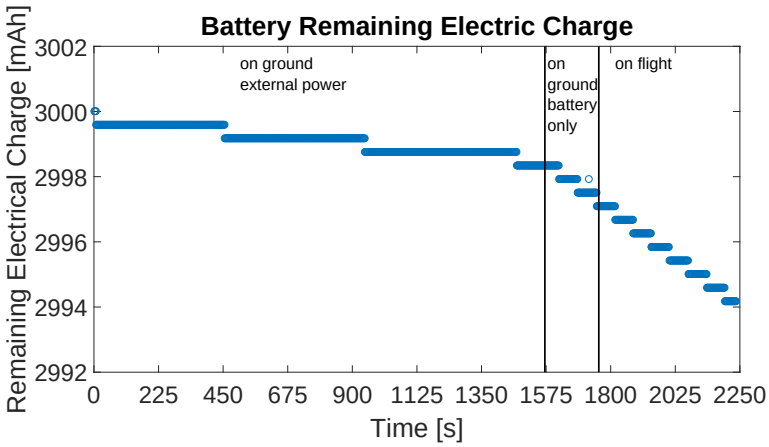


Figure 91: Batteries remaining electrical charge

is the rocket casing heating, due to the the air friction, which affects the whole experiment temperature behavior; the second reason is that the thermal experiment dissipated power was higher during the flight (heating up mode).

Figure 93 shows the OBDH microcontroller internal temperature. The temperature curve has a similar behavior than the one acquired by the EPS. This confirms the consistence on the results, since two different sensors, in two different PCBs have registered similar results.

Figure 94 shows the acceleration measurements performed by the OBDH IMU (MPU6050). From the plot, one may note only the g acceleration on z-axis before the launching, since the rocket z-axis was perpendicular to the Earth's surface at the launching platform. Also, as expected, there was no acceleration on axis x and y before the launching. Figure 94 shows the acceleration at the launching moment. Right after the lift-off, acceleration on z axis shows a peak of approximately 10 g and kept varying due to the ascendant movement of the rocket. As the rocket trajectory changes, x and y acceleration components were also increasing. Nearly at 1800 s the first stage separation occurred, causing high amplitudes on all axes. Due to a rocket belt separation fail, the acceleration level were higher than the expected and the rocket payload did not reach the correct velocity to obtain microgravity conditions. After 1830 s the rocket appears to be rea-

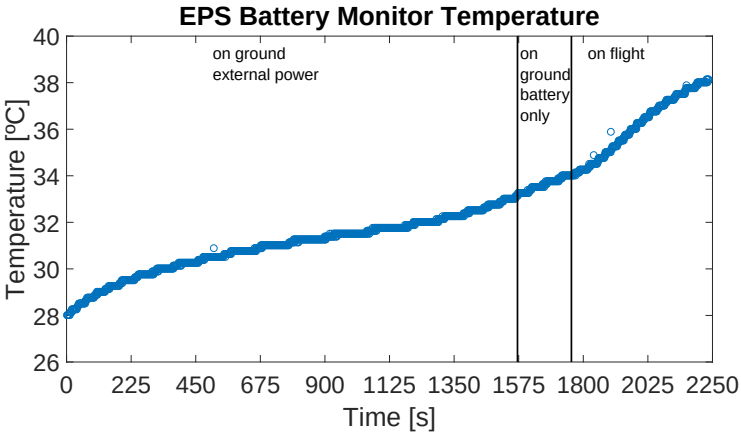


Figure 92: Batteries monitoring chip internal temperature

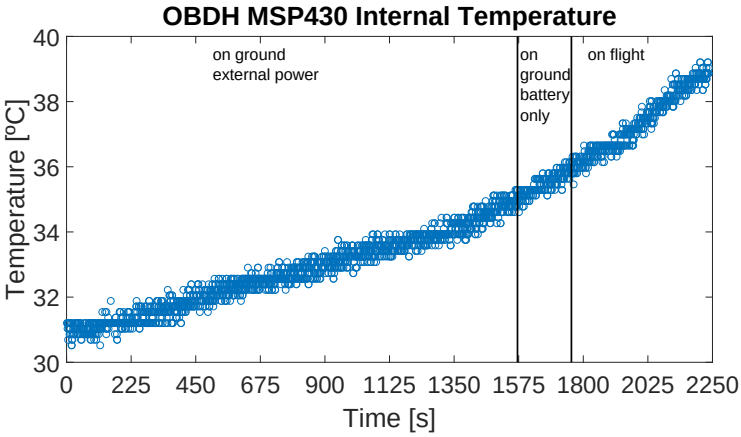


Figure 93: OBDH microcontroller internal temperature

ched low gravity environment, but due to the reduced time, it may be interpreted as a free fall before the parachute opening. After 2050 s, z axis component has returned to the same level before launching, and the other two axis acceleration returned to zero (rocket on vertical position).

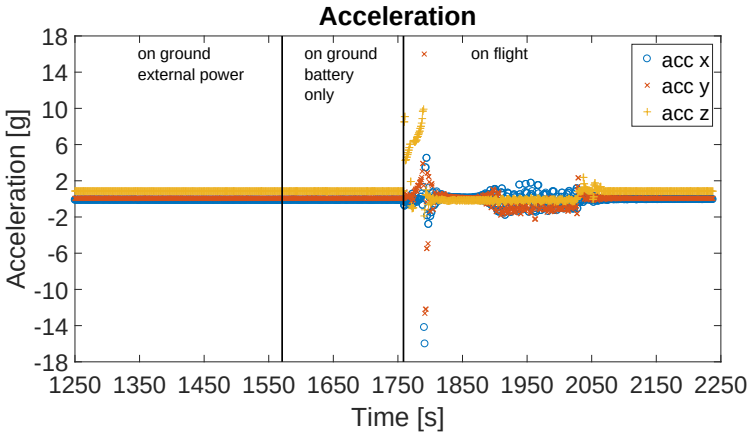


Figure 94: OBDH acceleration measurements

Figure 95 shows the rotation measurements performed by the OBDH IMU. Before the launching, as the rocket was attached to the launching platform, no rotation was measured on any axis. After launching, fast rotation around z axis was recorded, due to the ascendant spinning movement of the rocket. The plot shows a saturation on the z axis measurement, which could be avoided using a dynamic IMU range reconfiguration, achieved by software. This routine is already under development to be included on the FloripaSat flight model OBDH software. At 1800 s, rocket attitude control was activated but due to the belt separation problem it failed. At 1900 s parachute was opened. Figure 95 one may also estimate the landing time (close to 2150 second in the plot), since no rotation is verified in all axes.

Finally, and unfortunately, there was no communication between the transceiver (CC1125) and the beacon transmitter (CC1175). Further analyses allowed to detect an intermittent communication fail between the radios, which was caused by a design mistake on the transceiver circuit. For the transceiver to work properly with the frequency used on this project (437.5 MHz), the crystal should have frequencies higher

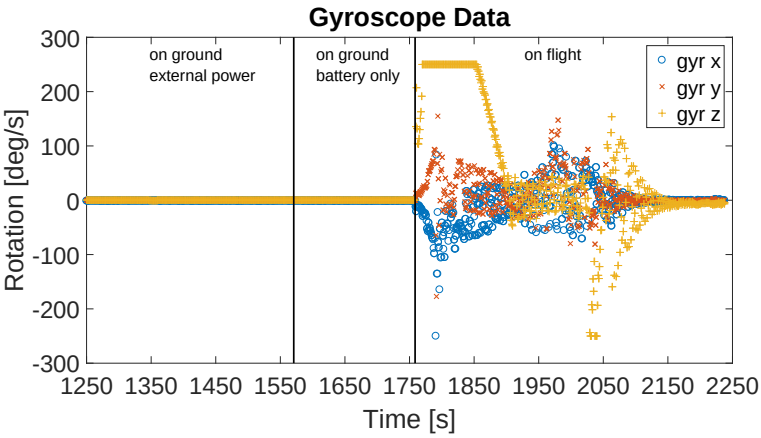


Figure 95: OBDH gyroscope measurements

than 40MHz. The lack on attending this requirement has caused a frequency shifting on the transceiver. Although the beacon radio probably transmitted correctly (verified by operational tests before and after the flight), the transceiver did not receive the data. This hardware bug has already been corrected and tests with the new hardware are being carried out at the group's lab. Fortunately, most FloripaSat data was sent to the MPP through the OBDH communication (via UART), which allowed to test all other subsystems' functions.

3.9 CONCLUSION

This chapter has presented a qualification test procedure for nanosatellites which allows analyzing the subsystems behavior under severe flight conditions. The idea behind the proposed test procedure is to fly the engineering model of a nanosatellite in a sounding rocket mission, before designing the satellite flight model. This chapter has described the hardware and software needs to build an interface between the rocket and the nanosatellite subsystems. The electronics interface (Multi-Mission Platform) allows to keep the tests as close as possible to the actual conditions that the satellite shall be submitted in the real mission. A case study has been presented, testing the FloripaSat engineering model subsystems, for which the sounding rocket

flight results have guided to important modifications in the satellite flight model. Most importantly, the flight results have qualified both the Multi-Mission Platform and the FloripaSat subsystem engineering model in flight. This result places this work in a select group of space qualified embedded systems dedicated to scientific missions.

The core features of the FloripaSat EPS engineering model have been tested on flight. **The results show that the subsystem operated perfectly for the energy management functions**. As described in this chapter, the solar panels input power control block was not tested during the flight. The reason was the rocket safety requirements, which precluded solar panels to be attached outside the spacecraft. Even though, the solar panels inputs have been used to externally power the nanosatellite subsystems and charge the batteries with the voltage source placed in the bunker.

As explained in chapter 2, the most complex architecture has been adopted for the FloripaSat engineering model: the discrete boost MPPT. This decision has been taken based on its theoretical efficiency performance, which surpasses the remaining. Although this input control architecture presented excellent results on laboratory tests, it has more failure probability than all other circuits. This conclusion motivated the main contribution of this thesis: to improve the simplest EPS architecture efficiency through software. Therefore, next chapter presents a very elegant solution to improve the Directly Coupled architecture efficiency through an energy-driven scheduling algorithm.

4 ENERGY-DRIVEN SCHEDULING ALGORITHM FOR NANOSATELLITE ENERGY HARVESTING MAXIMIZATION

4.1 INTRODUCTION

Chapter 2 has discussed different EPS hardware architectures, analyzing the circuits influence on energy harvesting capability. Although the circuit that controls the solar panels plays a major role on energy harvesting maximization, software policies may contribute to achieve this goal. As explained, the MPPT architecture is more complex and susceptible to fails when compared to the Directly Coupled one. Since the solar panel control feature (MPPT with boost regulator) was not tested in the sounding rocket mission, a simpler solution has been proposed as an alternative: Directly Coupled architecture with an energy-driven scheduling algorithm for energy harvesting maximization. This solution was firstly designed for the Directly Coupled architecture, however, along the work development, it has been noted its potential for other architectures as well (See Chapter 5).

Satellites perform tasks that are beyond energy harvesting and management. These tasks shall be somehow organized in order to accomplish the mission requirements. Since the tasks to be performed may have different priorities, execution time, resources, etc., a satellite task scheduling algorithm may be a key element to achieve a successful satellite mission. Once the scheduling algorithm may define which (and how) the satellite's tasks are going to be executed, there shall be a relation between the algorithm and the EPS, after all, tasks execution demands energy. This chapter shows how the satellite tasks execution may affect the solar panel behavior, mainly for the Directly Coupled architecture.

The satellite scheduling problem is not new. It has been formulated with a variety of perspectives, with numerous proposed solutions (Section 4.2). Although there are distinct manners for defining and solving the problem (DISHAN et al., 2013), (GAYTAN et al., 2015), (WANG et al., 2014), the goal is mostly the same: to optimize tasks execution from some perspective (maximize communication quality (CHRISTOPOULOS; CHATZINOTAS; OTTERSTEN, 2015), minimize system response time (KIM; CHANG, 2015), etc.).

As described in Section 4.2, most of recent satellite scheduling algorithms are not designed for nanosatellites. Besides this, none of them

aims energy harvesting maximization. Energy aware task schedulers have been widely discussed for wireless sensor networks (WSN) applications (HAN; LIU; LUO, 2013), (HOELLER; FRöHLICH, 2015). Although some of these works focus on reducing energy consumption to extend the nodes lifetime, the problem is totally different. Most wireless sensor networks have no energy input. Also, the periodicity (both in tasks execution as in power input) imposed by the orbital motion is not verified in most WSNs. Some authors, even when considering energy harvesting embedded systems, propose algorithms based on dynamic voltage and frequency scaling technique (TAN; YIN, 2016). This approach is restricted to reducing the processor power consumption only. Thus, this chapter introduces a solution to the satellite scheduling problem with a different approach for an emerging class of low power satellites (SLONGO et al., 2016b).

4.2 STATE OF THE ART

The task scheduling problem in satellites is not new, referring to the late 50s and early 60s, during the Space Race, where military artificial satellites started being developed and launched. At that time, the main concern was to maximize communication time. The system factors pertinent to the scheduling problem used to be classified into three categories: satellite availability, communication requirements and quality of communication. Linear programming approach has been one of the solutions to solve a set of mathematical equations in a maximization problem. Due to the computational limitation at that time, dynamic scheduling was considered an overhead (AHARA; ROSSBACH, 1967).

With the increasing number of launched satellites and the development of new channel access methods, the scheduling problem has become more sophisticated. For instance, scheduling algorithms have been applied to satellite systems communicating through time division multiple access (TDMA) to a channel. In this case, the proposed scheduling algorithm goal was to avoid or to reduce message conflict from ground stations when occupying a time slot. Also, the idea was to minimize the assignment procedure to shorten assignment time delay (DURRANI; JO, 1989). No power constraint is mentioned in this solution.

Later, scheduling techniques have been applied to Earth Observing Satellites (EOS). Some of these works considered energy cons-

traints in their scheduling algorithms. The Landsat 7 from National Aeronautics and Space Administration (NASA), for instance, implemented the so-called duty cycle constraint. A sensor should be limited to its operating time for a given period (GLOBUS et al., 2004). For the Landsat 7 a sensor should not be used for more than:

1. 34 minutes in any 100 minutes period,
2. 52 minutes in any 200 minutes period, or
3. 131 minutes in any 600 minutes period.

Since there is a correlation between the time the sensor is turned on and its power consumption, this can be considered an energy constraint. However, none of the evaluated algorithms solve the scheduling problem to reduce power consumption but to maximize the number of collected images from Earth.

An innovative work has considered fault-tolerant and real-time aspects to solve the task scheduling problem for multiple observation satellites (ZHU et al., 2015). In this innovative approach, the authors adopt the replication concept to ensure that a signed job is going to be executed. For this, they assume that a task primary copy is successfully allocated only if its corresponding task backup copy can be scheduled in another satellite. Otherwise, the primary copy shall be canceled. Even if one of the satellites fails in executing the task, the other one is able to execute it. Although the scheduling problem is rigorously well defined through a set of equations and assumptions that ensure the good performance of the algorithm, this work also does not mention power constraints or energy efficiency optimization.

Some recent works have been developed on solving the issue of ground station-satellite communication on multi-satellite missions. This problem also may be solved using scheduling algorithms. Recent ideas have emerged as applying mutation concepts of genetic algorithms to meet computation time and success rate mission requirements on satellite communication. Hybrid Dynamic Mutation has demonstrated outstanding performance in terms of speed and reliability when compared to other mutation strategies (ZHENG; GUO; GILL, 2017). Although both algorithms have proved to be efficient they are focused on the ground station side. They do not consider the satellite tasks management nor its energy consumption impact on mission accomplishment.

A dynamic scheduling approach is proposed by Wang et al. for emergency tasks on distributed imaging satellites (a satellite constellation for imaging proposes). The authors defined a multi-objective

mathematical programming model that contains five objects: tasks, resources, available opportunities, operational constraints and objectives (WANG et al., 2014). Energy consumption minimization is classified as one of the scheduling objectives. The authors present a so-called merging tasks technique, which allows tasks being executed simultaneously, reducing energy consumption in comparison with other algorithms. However, authors state that the scheduling main goal is to maximize the priorities of the scheduled tasks under operational constraints. Since the application is focused on emergency tasks, energy consumption is not the main concern.

An energy-driven scheduling algorithm has been proposed by Moser et al (MOSER et al., 2006). The authors considers an embedded system which is able to harvest energy. They are based on the principle that tasks deadlines may be attended only when there is enough energy to execute them. Therefore, properties as the energy source availability, capacity of the energy storage as well as tasks power consumption shall be considered. Their scheduling algorithm considers the input energy, as well as the stored energy, in order to properly schedule the tasks, avoiding the deadlines missing for energy lack. However their formulation does not implies in energy harvesting maximization. They are able to respect more tasks deadlines when compared to the traditional Earliest Deadline First (EDF) algorithm. However, the authors consider the input power actually fed into the energy storage. They do not analyze the task scheduling effect in the battery behavior and, consequently, in the harvested energy availability.

Finally, a very interesting and recent work has combined a nanosatellite application with an energy-efficient scheduling technique (PANG et al., 2015). The authors emphasize the advantages of using nanosatellite swarms for synthetic aperture radar (SAR) application. Nanosatellite swarms may be more efficient and less costly than a single satellite solution. Although part of the authors' work problem definition is application dependent (SAR), it does consider energy constraints. The authors define a power function which has a minimum value to perform an assigned task.

The authors also introduced a satellite failure probability aspect in their problem formulation. Satellite's failures are modeled through a reliability function, based on Weibull distribution (WEIBULL, 1951) (NASSAR et al., 2017). Besides this, the authors have created a set of scenarios considering energy consumption and communication bandwidth, adding the failures probability component. Finally, the problem is reduced to the failures and power consumption minimization. The authors

selected the following metrics to evaluate their algorithm performance: mean total energy consumption (MTEC), mean time to mission failure (MTMF), and mean time between failure (MTBF). Simulations have shown that their approach achieved better results than the general algorithms partial rescheduling (PR) and complete or full rescheduling (CR).

It is worth noting that this last discussed work took a step beyond the other mentioned above. It has not only added an energy constraint, but the proposed algorithm has been optimized to reduce power consumption. The next section shows the scheduling strategy proposed in this chapter, which moves further on the energy issue, aiming the nanosatellite's energy harvesting maximization.

4.3 PROPOSED SCHEDULING ALGORITHM STRATEGY

The proposed scheduling algorithm strategy applies for the EPS directly coupled circuit presented in Section 2.4 (see Figure 5). Recalling Equation 2.1, it is clear that the solar panel voltage (V_{sp}) depends on solar panel current I_{sp}, battery current I_{bat} and battery voltage V_{bat}. Solar panel current depends on the solar irradiance. However, the battery voltage and current, may be directly affected by the nanosatellite power consumption (tasks executions). This means that, according to the tasks being executed, the nanosatellite power consumption may vary, consequently, increasing or decreasing the battery voltage and current. This directly affects the nanosatellite power input.

Although Equation 2.1 shows that V_{drop} may be obtained from the solar panel current I_{sp} and from the battery current I_{bat}, there is no need for computing the V_{drop} in the EPS microcontroller. This is the elegance of the problem formulation. By using the Perturb and Observe (P&O) maximum power point algorithm (Section 4.3), all that the EPS microcontroller has to do is to trigger the tasks and measure the solar panel delivered power (solar panel voltage and current measurements). This Maximum Power Point Tracking (MPPT) algorithm has been implemented in C language, which runs on the EPS microcontroller, allowing energy harvesting maximization through the tasks execution control.

Based on the mathematical conclusion that the V_{sp} is affected by the V_{bat}, allowing power input to be maximized by tasks execution control, then, the scheduling algorithm may be designed. This section clarifies how the scheduling algorithm has been implemented, taking

into account the energy harvesting maximization and the tasks execution issues.

4.3.1 Energy harvesting maximization

As demonstrated in Section 2.4, the EPS directly coupled architecture causes the solar panels to depend on the battery voltage. Therefore, the nanosatellite's tasks execution affects the operating point of the solar panels through the battery charging and discharging effect. Then, the proposed algorithm controls the nanosatellite load's current in order to keep the solar panels working as close as possible to their MPPT. To achieve this, the Perturb and Observe (P&O) method (SARAVANAN; BABU, 2016) is implemented in the EPS microcontroller.

The MSP430F249 internal analog to digital converter (ADC) is used to measure the solar panels voltage. ADC channels are also used to measure the solar panels current through shunt resistors and amplification circuits. These measurements allow the instantaneous input power calculation in the microcontroller (Equation 4.1). The microcontroller calculates the input power in every iteration. For each power measurement the microcontroller compares the actual value ($P_{sp}(n)$) with the previous one ($P_{sp}(n-1)$). According to the measurements, more or less nanosatellite tasks are triggered, causing the battery voltage to vary and the solar panels to approximate to their optimums operating point.

$$P_{sp} = V_{sp} I_{sp} \tag{4.1}$$

Figure 96 explains the algorithm. Unlike a hardware MPPT, the proposed algorithm does not change the solar panels voltage directly. For points 1 and 2 in Figure 96, the energy-driven scheduling algorithm would reduce the number of tasks being executed, in order to increase battery voltage and, consequently, to increase the solar panel voltage (Vsp). This sets a new operating point for the solar panels, increasing the nanosatellite power input. The opposite behavior is verified in points 3 and 4, where the solar panel voltage increases and the input power moves away from the $Pmpp$. The algorithm shall identify this and shall increase the number of tasks to be executed in the next iteration. The conclusion is: solar energy harvesting maximization is ensured by the dynamic control of the task's execution. From this conclusion it is clear that the scheduling algorithm is energy-driven. Algorithm 1 shows a pseudo code for the scheduling algorithm, where

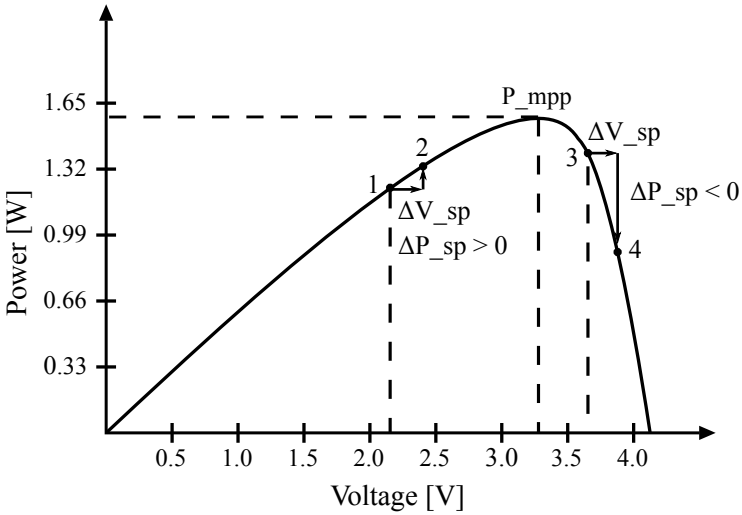


Figure 96: P&O algorithm.

perturb the system, in this case, means to execute more or less tasks.

For any set of tasks to be scheduled, independently of their power consumption and duration, it is possible to maximize the solar panels delivered power. Anytime that the battery voltage shall be changed, tasks will be anticipated or preempted. This means that the algorithm would give preference to maximizing the input energy rather than meeting the tasks' deadlines. The problem is that, not ensuring the tasks to be properly executed may be extremely harmful for the mission. For critical application satellites this approach may conflict with mission requirements. However, for scientific missions (most current nanosatellites' application), it may be more relevant to harvest more energy (increasing the overall satellite's number of tasks performed in orbit) than meeting non-critical tasks deadlines. Even though, the energy-driven approach shall be able to deal with tasks with different power consumption, duration and priority ensuring their proper execution. Therefore, next section explains how the energy-driven scheduling algorithm may address this issue.

Algorithm 1 MPPT Algorithm.

initialize variables
measure voltage $V_{sp}(n-1)$
measure current $I_{sp}(n-1)$
compute power $P_{sp}(n-1) = V_{sp}(n-1) \cdot I_{sp}(n-1)$
perturb the system in arbitrary direction
loop
    measure voltage $V_{sp}(n)$
    measure current $I_{sp}(n)$
    compute power $P_{sp}(n) = V_{sp}(n) \cdot I_{sp}(n)$
    if $P_{sp}(n) > P_{sp}(n-1)$ **then**
        perturb the system in the same direction as previous iteration
    else
        perturb the system in the opposite direction of previous iteration
    end if
    $V_{sp}(n-1) = V_{sp}(n)$
    $P_{sp}(n-1) = P_{sp}(n)$
end loop

4.3.2 Definition of tasks execution

The first thing to keep in mind is that the nanosatellite may be considered a multicore problem. Every single nanosatellite subsystem/-module has its own microcontroller, co-working in a distributed control architecture. Figure 97 shows how the nanosatellite tasks may be distributed in two subsequent orbit cycles. The tasks are executed by the nanosatellite subsystems/modules EPS, OBDH and TT&C. Therefore, for a given orbit O_i (where $i \in \mathbb{Z}$ represents the number of orbits) the tasks may be classified in three different sets: tasks already executed T_{ae}, tasks under execution T_{ue}, and tasks to be executed T_{tbe}. Thus, the total number of tasks T_T executed in an orbit O_i may be described as:

$$T_T = T_{ae} \cup T_{ue} \cup T_{tbe} \tag{4.2}$$

The energy-driven scheduling algorithm does not consider the already executed tasks in order to determine which tasks shall be executed next. Therefore, only a subset of T_T is taken into consideration,

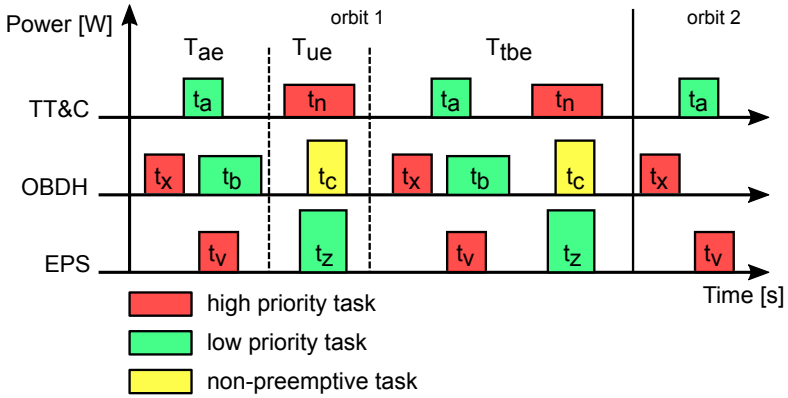


Figure 97: Tasks distribution in orbits.

which is the T_c, consisting of the tasks under execution and the tasks to be executed within the orbit O_i. This defines the set of tasks considered by the energy-driven scheduling algorithm for each iteration.

$$T_c = T_{ue} \cup T_{tbe} \tag{4.3}$$

From Algorithm 1 in Section 4.3.1, there are two options every time the energy-driven scheduling algorithm is triggered: tasks under execution shall be preempted (t_{pp}); or tasks to be executed shall be anticipated (t_{atp}). The question is: may all the satellite tasks be preempted without prejudicing the mission's goal? For most applications the answer is no. But, as already discussed, for nanosatellites under scientific missions, the number of preemptive tasks tends to be much greater than for critical application satellites. Even though, the energy-driven scheduling algorithm shall ensure that the non-preemptive tasks are going to be executed properly. Besides this, which task from the set T_{tbe} shall be triggered first when the algorithm identifies the need for tasks anticipation?

The solution proposed for these questions is to split the tasks' set T_c in three tasks' lists: high priority list L_{hp}; low priority list L_{lp}; and non-preemptive list L_{np}. These three lists shall be defined by the energy-driven scheduling algorithm designer, considering the nanosatellite mission requirements. Some rules must be defined in order to ensure that the energy-driven scheduling algorithm will execute all the tasks properly:

1. only one task from L_{hp} may be executed at time;

2. tasks to be anticipated are defined by the Earliest Deadline First (EDF) criteria;

3. tasks under execution from L_{np} must not be preempted.

The decision of executing only one high priority task at time is based on the nanosatellite hardware architecture. Although it has three microcontrollers to execute the tasks, they are connected to the same hardware peripherals, in some specific circuit blocks. This could lead to the situation of two or more microcontrollers trying to access the same component/sub-circuit with the same priority level. Therefore, to avoid this type of conflict, and to simplify the tasks priority model, only one high priority task shall be executed at time.

The traditional EDF algorithm has been already defined as inefficient for attending the tasks deadlines in such energy harvesting embedded systems (MOSER et al., 2006). However, in this case, it is combined with the energy maximization algorithm, which considers the energy input aspect every iteration.

Equations 4.6 to 4.8 show a hypothetical example for the lists configuration for the tasks of Figure 97. Besides the tasks distribution, Figure 97 uses colors to represent tasks nature: red - high priority; green - low priority; yellow - non-preemptive. Considering those lists and set of tasks, and supposing that the energy-driven scheduling algorithm has determined that a task shall be anticipated (t_{atp}), that task shall be t_a. Based on the proposed strategy, follows:

$$T_{ue} = \{t_n, t_c, t_z\} \tag{4.4}$$

$$T_{tbe} = \{t_x, t_a, t_v, t_b, t_n, t_z, t_c\} \tag{4.5}$$

$$L_{hp} = \{t_n, t_x, t_v\} \tag{4.6}$$

$$L_{lp} = \{t_a, t_b, t_z\} \tag{4.7}$$

$$L_{np} = \{t_c\} \tag{4.8}$$

$$t_{atp} = t_a \tag{4.9}$$

Although t_x has the earliest deadline among the tasks to be executed T_{tbe}, it is a high priority task in L_{hp} as well as t_n, and two high priority tasks must not be executed simultaneously. Figure 98 illustrates the incompatibility of two high priority tasks under execution. Recalling Figure 97, the task with earliest deadline in T_{tbe} (excluding t_x) is t_a, therefore $t_{atp} = t_a$, as shown in Figure 99.

Now, considering the same scenario of Figure 97, but supposing

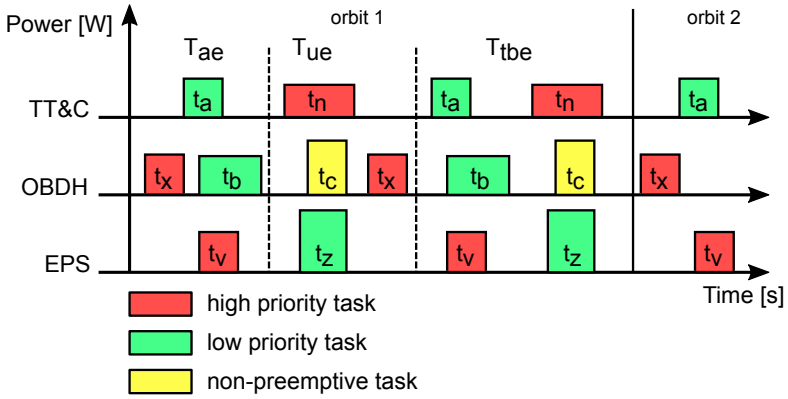


Figure 98: Incompatible scenario of two high priority tasks under execution.

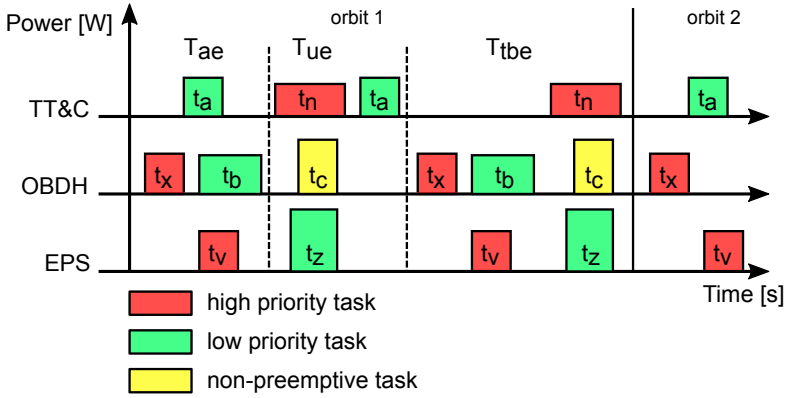


Figure 99: EDF modification causing t_a to be executed instead of t_x.

that a solar irradiance variation has caused the algorithm to preempt some task. That task shall be t_z. This occurs because t_c is a non-preemptive task (it is in L_{np}) and because t_n is a high priority task. This scenario causes the algorithm to preempt the low priority task t_z, as shown in Figure 100.

The tasks organization in these lists, as well as the rules to execute that tasks, allow both the energy harvesting maximization, as well as the tasks to be properly executed. Even for tasks with different power consumption and duration, the tasks' deadlines may be

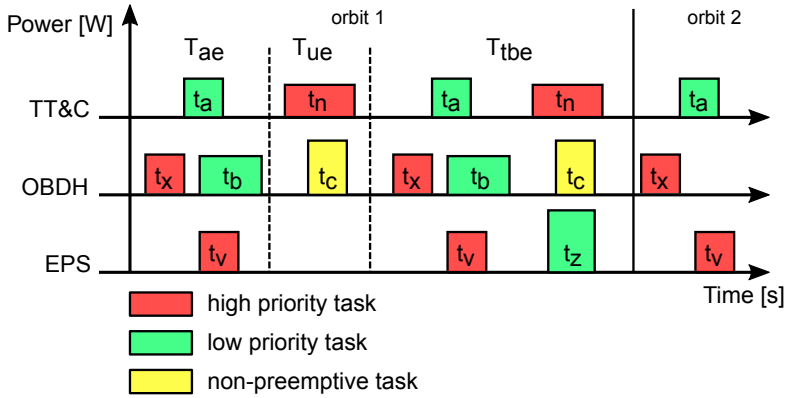


Figure 100: Task t_z preempted to achieve energy harvesting maximization.

accomplished.

However, an important remark shall be made here. The number of non-preemptive tasks to be scheduled is inversely proportional to the energy harvesting maximization efficiency. Also, the more energy the tasks takes the lower is the efficiency of the scheduling algorithm. This introduces the concept of **task granularity**, which may be considered equivalent to the tasks consumed energy. Keeping the grains small (low energy tasks) allows the algorithm to respond faster to the solar irradiance and temperature variations, which results in harvesting more energy.

$$t_E = \int_a^b t_P dt \tag{4.10}$$

where t_E is the task energy; t_P is the task power consumption; and $b - a$ is the time to execute the task.

Finally, it is worth mentioning that this lists' concept allows adopting this algorithm to any nanosatellite operating with the directly coupled EPS. Even though the nanosatellite set of tasks, subsystems or payloads are different from the FloripaSat, the algorithm may still be used. The only work that needs to be made is to distribute the nanosatellite into the proposed lists. A list defining tool may be used for this purpose, as explained in Chapter 5.

4.3.3 Algorithm iteration interval

In addition to the tasks execution issue, another important aspect that must be taken into account when designing the energy-driven scheduling algorithm is the iteration interval. Ideally, the shorter the scheduling iteration interval the better is the result on maximizing the energy harvested. This occurs due to the satellite fast dynamics (position on orbit and tasks execution). Since the scheduling algorithm controls the satellite's tasks in order to keep the solar panels voltage on the optimum value (maximum power point voltage V_{mpp}) and because the optimum condition continually changes, the algorithm shall be triggered as fast as possible. However two trade-offs must be considered to define the optimum algorithm iteration interval: power consumption and computational capacity.

Starting from the algorithm power consumption issue, as shown in Section 4.3.1, the operations to maximize the solar energy input are the following: measuring the solar panels voltage and current (ADC readings), computing the instantaneous delivered power, comparing the instantaneous input power with the previous value and enabling/disabling the satellite's tasks (IOs configuration or communication through I^2C). All of these operations are executed by an ultra low power microcontroller, which consumes few micro watts to execute them. Therefore, the power consumption to perform one algorithm computation may be considered insignificant when compared to the nanosatellite overall power consumption, the EPS power consumption, or even with the energy harvesting gain by triggering the scheduling algorithm.

However, the problem arises when the scheduling algorithm iteration interval is so small that the microcontroller ultra low power consumption level is higher than the energy harvested. This case rarely occurs before the microcontroller computational resources limit is achieved. However, it is worth defining limits to the iteration interval. The following three conditions shall be attended in order to achieve the best performance for the proposed energy-driven scheduling algorithm:

1. $t_{aii} > t_{aci}$;
2. $t_{aii} > t_{pc} \rightarrow E_{ai} < E_{hg}$;
3. t_{aii} shall be made as smaller as possible, respecting the two above conditions.

where t_{aii} is the energy-driven scheduling algorithm iteration interval, t_{aci} is the available interval for the computational resource to compute

the scheduling algorithm outputs, and t_{pc} is the interval limit for which the algorithm iteration energy consumption E_{ai} is smaller than the energy harvesting gain E_{hg}.

Considering an EPS based on a single core microcontroller, and that there is no operating system (bare metal execution) to allow multithreading, then, the scheduling algorithm low period iteration may be considered a computational capacity restriction problem. However, it is important to mention that, our case study (FloripaSat), as well as most of recent nanosatellites, is based on a distributed computational architecture. This means that each satellite module/subsystem (EPS, TT&C, OBDH, ADCS) has its own microcontroller. This drastically reduces the computational resource problem, since the EPS has a dedicated microcontroller, with only two main functions: battery management and scheduling algorithm execution. The satellite tasks control shall be performed by each module/subsystem. In most cases, the EPS function is only to inform, through the I^2C protocol (which connects all modules/subsystems), that more or less tasks shall be executed. In short, our suggestion is to firstly design the EPS battery management embedded software (the most critical function), allocating the remaining EPS microcontroller computational capacity to be used to the scheduling algorithm.

The optimal energy-driven scheduling algorithm iteration interval may be experimentally obtained, considering the mission requirements (computational resource, microcontroller power consumption, etc). For our case study (FloripaSat), t_{aii} was defined as 455 ms (see Section 4.4.2). Tests have been performed with smaller iteration interval (400 ms), leading to even better results (more energy harvested). This means that t_{aii} =400 ms> t_{pc}. However, the iteration interval could not be made smaller than 455 ms because this was the smallest interval to avoid conflicts with the critical functions of battery management, which are also performed by the EPS microcontroller.

4.4 EXPERIMENTAL SETUP

4.4.1 Test stand

In order to evaluate the above mentioned scheduling algorithm the test stand proposed in Section 2.11 is configured to perform the following tests:

1. **Solar panel emulation:** One of the four power analyzer channels is set with positive voltage and positive current, emulating the solar panel behavior. This first test allows the evaluation of EPS features regarding the energy input, like solar panel voltage and current measurements;

2. **Battery emulation:** One of the four power analyzer channels is set with positive voltage and positive current (discharging mode) or negative current (charging mode), emulating the nanosatellite battery. This test allows the evaluation of EPS battery monitoring features like: input/output battery current measurement, battery voltage measurement, overvoltage and undervoltage protection;

3. **Nanosatellite power consumption emulation:** This test may be performed setting one power analyzer channel with positive voltage and negative current, working as a controlled electronic load. The parameters (voltage and current) may be set by equations, by stored data vectors or even in real time, by a LabVIEW virtual instrument (VI). This feature allows to test the EPS with a variable electrical load as suggested in Section 4.3;

4. **Orbit-position input power emulation:** A light emitter system (high power LEDs) is controlled by one of the power analyzer channels. The light emitter system illuminates one or more solar panels, which deliver energy to the EPS. Since the power analyzer has three other available channels, one or more of the above mentioned tests may be performed simultaneously. For instance, the whole scenario may be emulated selecting channel one for controlling the light emitter system, channel two for the battery emulation and channel three for the power consumption emulation. In this configuration, a long term test may be performed, where many orbit cycles may be emulated, experimentally evaluating not only the EPS itself, but mainly, the scheduling algorithm;

5. **Computer communication:** The EPS has a debug USART serial interface for sending important data to the computer. This interface is used only for in-house tests. This allows, for instance, to identify all the decision taken by the EPS, including the triggered tasks.

4.4.2 Test configuration

The results presented in Section 4.5 have been obtained from a three orbital cycle test performed at the laboratory. A LED lamp had its light intensity controlled by one power analyzer channel working on current setting mode. For this channel, a current curve is generated, emulating the behavior of the power delivered to the solar panels in orbit (Figure 101), as explained in Section 2.8.2. Two solar panels have been connected in parallel and placed inside a box to avoid external light influences. The FloripaSat is provided with six solar panels, however, not all of them are going to deliver power to the system simultaneously. Due to the satellite rotation, only two or three panels are hit by the sunlight simultaneously (according to orbital positioning). Figure 102a shows a picture of the FloripaSat's prototype and Figure 102b shows the EPS under test.

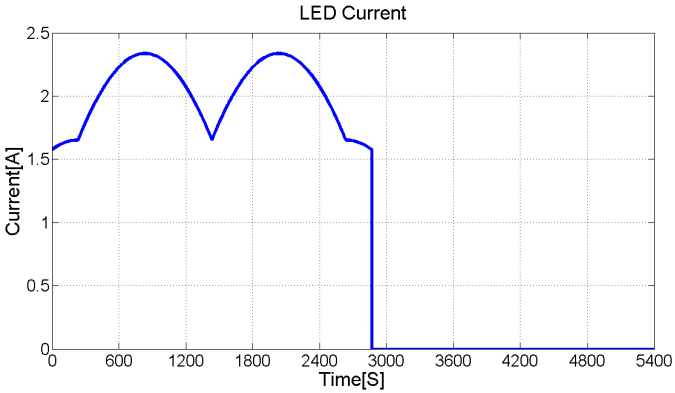
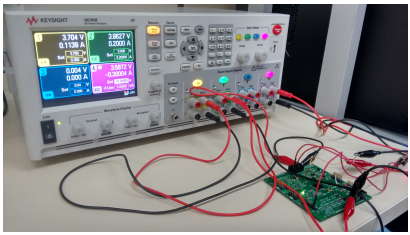


Figure 101: LED current.

The second power analyzer channel emulates the satellite power consumption (working as an emulated electronic load). It represents the nanosatellite tasks in execution. For each algorithm iteration the load current is increased or decreased in 10mA, according to the calculation performed by the microcontroller (every 455ms). The power analyzer control has been implemented in LabVIEW. The computer receives the EPS command (through serial interface) to increase or decrease the load current and the LabVIEW actuate on the power analyzer. In normal operation, the EPS shall send the commands to the OBDH, which will trigger the nanosatellite tasks. The decision of using an emulated elec-

(a) FloripaSat prototype.



(b) FloripaSat EPS under test.

Figure 102: FloripaSat prototype and EPS under test.

tronic load, instead of the real FloripaSat PCBs executing their tasks, was made to simplify the experiment. Not all the FloripaSat software control is ready to use yet. Besides this, the emulated electronic load provides more flexibility without compromising the test consistency.

Also, as discussed in Section 4.5 the temperature affects the solar panels efficiency. For this reason, the solar panels temperature is also measured. Two T-type thermocouples (one for each solar panel) have been attached on the solar panels surface. The temperature has been sampled every second and the data was sent to the computer through the Omega HH127 Thermocouple Datalogger and Thermometer.

4.5 RESULTS

The results presented in this section have been obtained from tests of three orbital cycles. Each cycle takes nearly 90 minutes, therefore, each test lasts for approximately 270 minutes. First, the EPS running the proposed energy-driven scheduling algorithm has been evaluated. The algorithm works only during the orbit period, where the sunlight hits the solar panels. During the eclipse the current provided to the load is kept constant (50mA). After the test with the scheduling algorithm is finished, the average current provided to the load is calculated. Then, a test without the scheduling algorithm is performed. In the test without the algorithm, the load current is kept constant (with the average current from the test with the scheduling algorithm)

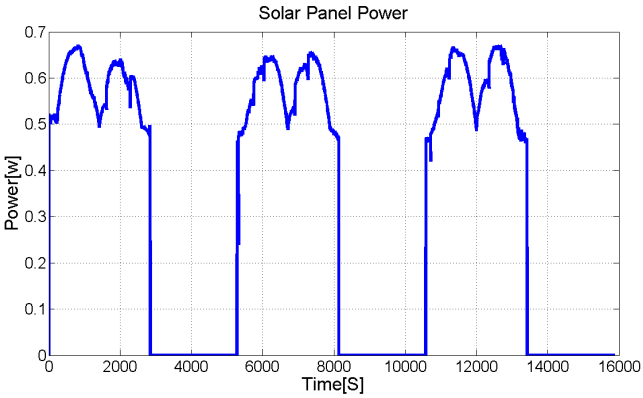
during the three orbit cycles.

Figure 103a shows the power delivered by the solar panels during the test with the scheduling algorithm. Comparing with the result obtained from the test without the scheduling algorithm (Figure 103b), one notes that the nanosatellite has harvested more energy with the energy-driven scheduling algorithm. Considering the complete test for three orbits, the scheduling algorithm allowed the EPS to harvest 4.48 % more energy than the EPS running without the scheduling algorithm. The greater difference occurs in the first orbit cycle, where the proposed algorithm has allowed an energy harvesting gain of 8.46%.

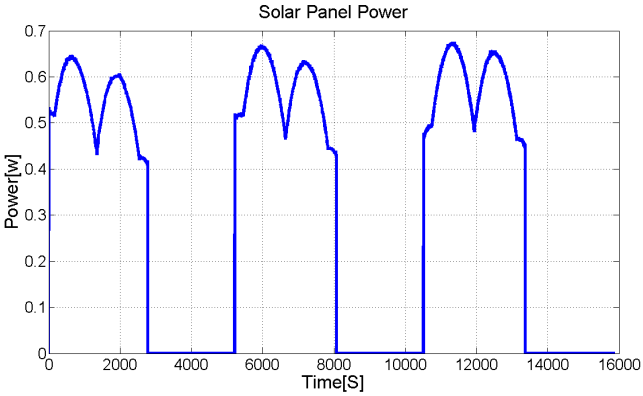
The solar panels voltage curve allows one to better understand why the difference was greater in the first orbit cycle. Figure 104 presents the solar panels voltage for both tests. Note that the solar panel voltage drops in the beginning of Figure 104a. The algorithm identifies that the battery voltage is too high, causing the solar panels voltage to operate far away from the MPP. Therefore, more tasks start to be triggered (electronic load current increased in steps of 10mA). The battery voltage drop (due to the increased current provided to the load) causes the solar panels voltage also to decrease. Therefore, the solar panels operate closer to the MPP in this cycle. Figure 104b shows the opposite behavior for the experiment without the energy-driven scheduling algorithm. Since the load current is kept constant in this test (no scheduling algorithm) the battery voltage increases as the solar irradiance increases. This causes the solar panels to operate far away from the MPP, harvesting less energy.

Note that for the two next cycles, the difference in the power delivered by the solar panels decreases (Figure 103). This happens due to the battery voltage drop in the test without the energy-driven scheduling algorithm. Even with no control of the load current, the solar panel voltage drops for the two last cycles, due to the natural battery discharge providing constant current to the load. Longer tests may show that, after the battery reaches its under-voltage limits the system without the energy-driven scheduling algorithm would be even more inefficient.

Another important issue to keep in mind is the matching between the solar panels and the battery (LEE; KIM; SHIN, 2013). For any satellite and specially for the EPS with the directly coupled circuit architecture, the correct definition of batteries and solar panels may drastically affect the overall satellite efficiency. For our case study, the solar panels may be designed to have a higher V_{mpp}. This would avoid the solar panels voltage to move far from their optimal operation point during the



(a) Test with the scheduling algorithm.



(b) Test without the scheduling algorithm.

Figure 103: Solar panel power.

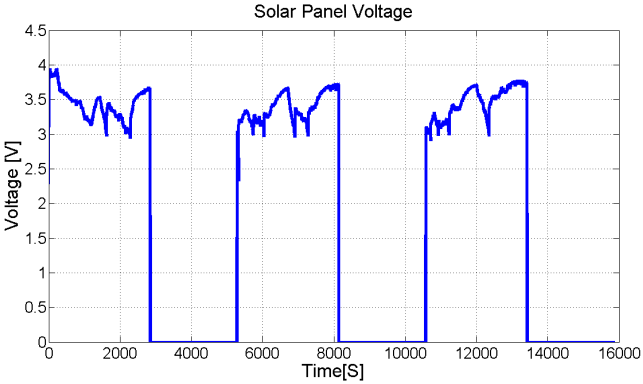
eclipse. Since the solar panels V_{mpp} is close to the battery voltage knee, it is harder to keep the battery voltage around this point. Therefore, the EPS hardware design may also affect the energy-driven scheduling algorithm.

Figure 105 shows the great difference in the solar panel current with and without the energy-driven scheduling algorithm. For all three orbit cycles one may note that the solar panel current has almost the same behavior for the test with the scheduling algorithm (Figure 105a). This demonstrates the efficient control to operate the system on the MPP. For the case without the energy-driven scheduling algorithm (Figure 105b), one may note that the harvested current increases every cycle, since the solar panel approximates the MPP, due to the natural discharge of battery. For longer tests, this efficiency shall also decrease when the battery voltage drops below the optimal voltage that causes the solar panels to operate close to the maximum power point.

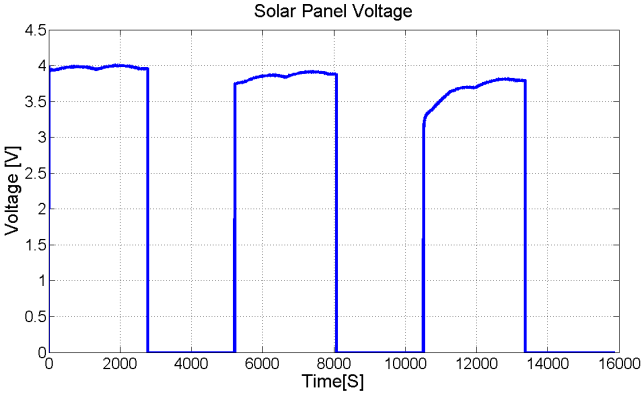
Figure 106 shows the battery voltage for both tests. Figure 106a shows the algorithm impact in battery voltage, by changing the load current. On the other hand, Figure 106b shows the natural charging and discharging process caused by the solar panel input current and the constant load current (104.8mA). Another important aspect in these plots is the under-voltage limit around 2.4V. Note that, due to the algorithm load current control, this limit is reached since the first orbit cycle, where the battery voltage is reduced when the solar irradiance reduces. For the test without the energy-driven scheduling algorithm this limit occurs only in the third orbit cycle.

Figure 107 also shows the battery current for both tests. In these plots the difference is more clear. For the case with the energy-driven scheduling algorithm (Figure 107a), the battery current varies significantly, in order to compensate temperature and irradiance changes, keeping the solar panel voltage as close as possible to the MPP. Figure 107b shows a more regular behavior, where the current varies only due to the solar panel current input (since the load current is constant for this case). From Figure 107 an interesting question may arise: is this battery charging current variation caused by the scheduling algorithm harmful to battery life? In order to properly answer this question, a battery aging effect/charging method analysis shall be conducted. This is out of the scope of this thesis, however, this topic is mentioned in Chapter 5, where a battery aging parameter is suggested to improve the scheduling algorithm performance.

Finally, Figure 108 shows the solar panels temperature during the tests. Two thermocouples have been asymmetrically placed over

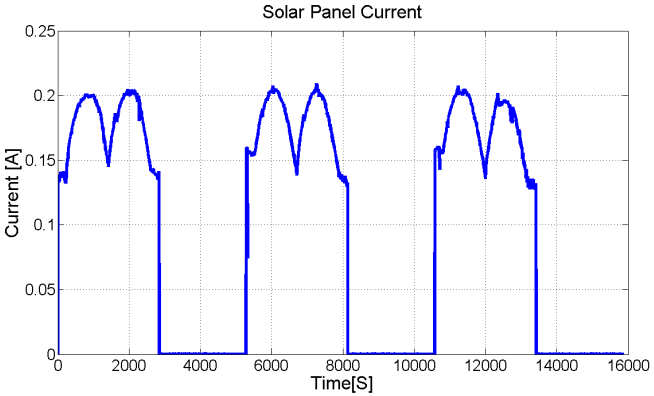


(a) Test with the scheduling algorithm.

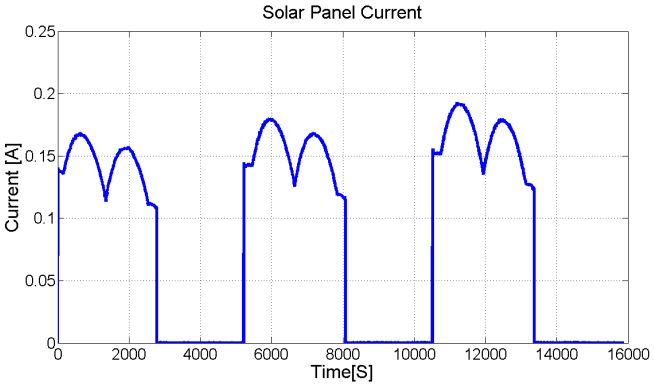


(b) Test without the scheduling algorithm.

Figure 104: Solar panel voltage.

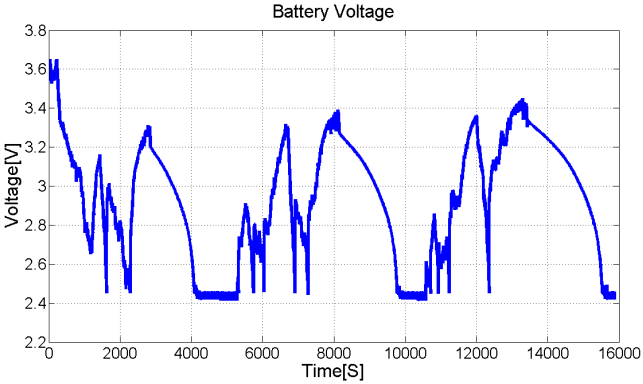


(a) Test with the scheduling algorithm.

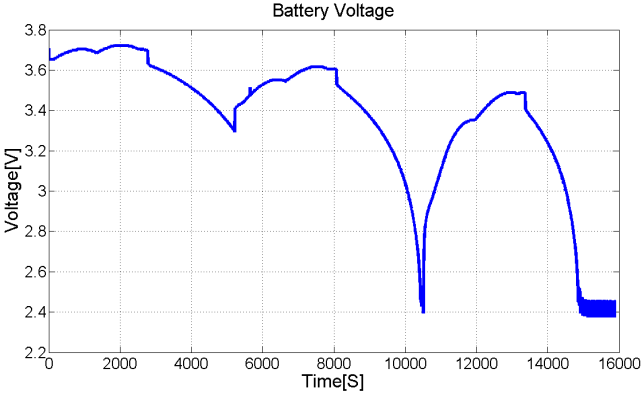


(b) Test without the scheduling algorithm.

Figure 105: Solar panel current.

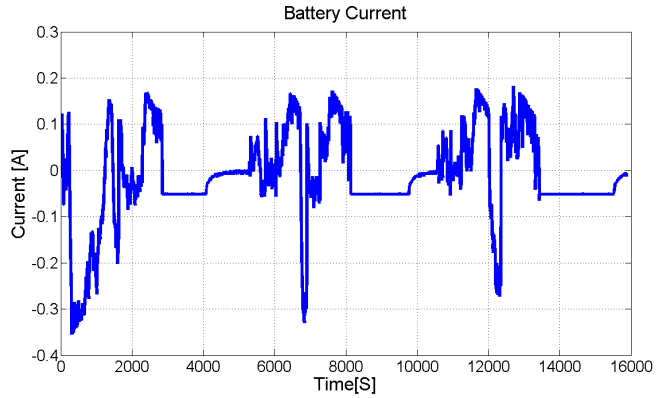


(a) Test with the scheduling algorithm.

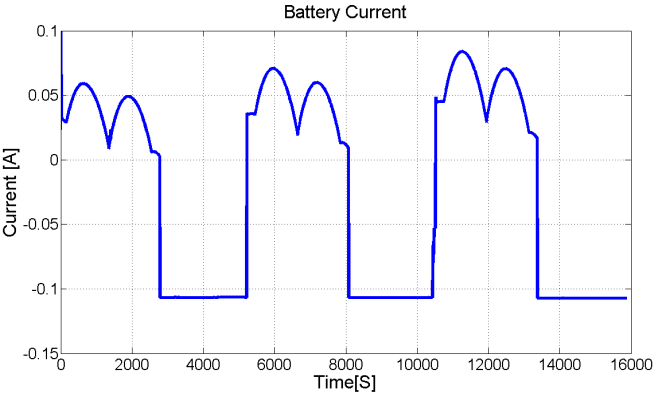


(b) Test without the scheduling algorithm.

Figure 106: Battery voltage.



(a) Test with the scheduling algorithm.



(b) Test without the scheduling algorithm.

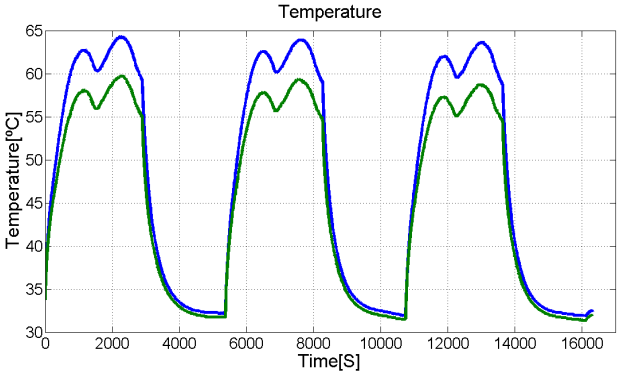
Figure 107: Battery current.

the solar panels in order to measure the temperature gradient along the experiment. They are identified by the green and blue lines in Figure 108. The experiments were performed inside a closed chamber, in order to avoid external light to affect the tests. LEDs lights were focused on solar panels increasing their temperatures due to irradiation effect. During the shadow period, solar panels temperature decreases, but it increases gradually as the LEDs are turned on again. A cooler to remove the warm air from the chamber was used, however it was not effective enough due to convection heat transfer limitation. A new test stand with two more powerful coolers has been designed to decrease the heating effect on the solar panels. Nevertheless, nanosatellites face similar conditions in outer space, where properly temperature control on solar panels is not possible. Indeed, temperature ranges will be different in orbit, but heating effect after leaving the shadow region will always be present on the solar panels.

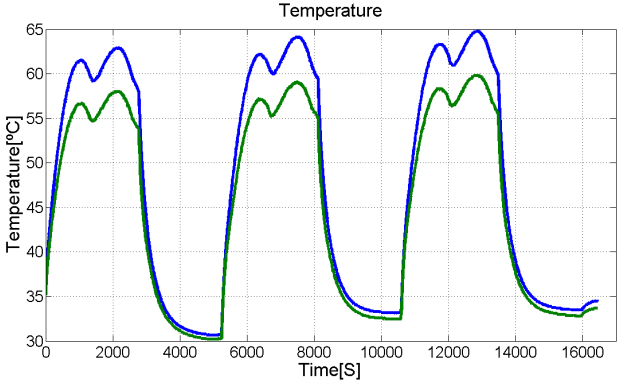
Temperature variation is the main cause to change the solar panel V_{MPP}. Comparing Figure 108 with Figures 105 and 103 one may note that the temperature variation affects the solar panel current and power output. The higher the temperature, the lower the delivered power. This occurs due to the great variation in the V_{MPP} for different temperatures. This justifies the power and current peaks difference in the same orbit cycle in Figures 103b and 105b. Note that for the test with the energy-driven scheduling algorithm this effect is reduced due to the dynamic change in load's current, reducing the battery voltage when the temperature increases (Figures 103a and 106a).

4.6 CONCLUSION

The proposed algorithm has demonstrated its energy harvesting maximization capability. Considering the complete test with three orbits, the EPS running with the energy-driven scheduling algorithm harvested 4.48 % more energy than the EPS running without the algorithm. An energy gain of 8.46% has been achieved for the first orbit cycle, where the initial battery voltage causes the solar panels to operate far from the V_{mpp}. The experiments also have shown that the natural battery discharge (operation without scheduling algorithm) momentarily approximates the solar panel voltage to the V_{mpp}. However, due to the cyclical nature of the battery charge/discharge process in orbit, tests longer than three orbit will show that the EPS harvesting efficiency without the energy-driven scheduling algorithm will decre-



(a) Test with the scheduling algorithm.



(b) Test without the scheduling algorithm.

Figure 108: Solar panel temperature.

ase again. Besides this, the temperature effects on solar panel power output has been reduced by the load's current control provided by the energy-driven scheduling algorithm. This is also a benefit of the proposed algorithm.

This work has experimentally proved that tasks control may optimize energy harvesting capability for directly coupled EPS. However, the tasks have been emulated as steps of 10mA in an electronic load. The next activity is to model the CubeSat tasks considering their power consumption and duration, using the lists' tasks organization proposed in Section 4.3.2. The Perturb and Observe algorithm shall then be tuned to operate with this set of tasks. Also, tasks priorities optimization shall be implemented in the new version of the energy-driven scheduling algorithm.

Besides this, other parameters may be added to the algorithm in order to make it even more efficient. Firstly, the battery remaining energy shall also be considered to avoid tasks preemption due to the lack of stored energy. In order to properly add an energy management strategy to this algorithm, the battery aging effect must be considered (AGARWAL et al., 2010). Finally, an energy input prediction model may also improve the rate of successfully executed tasks. Since the input energy behavior is periodic, the algorithm may execute tasks that would be preempted when analyzing only the stored energy. The next chapter discusses ideas that will guide future work, improving even more the Directly Coupled EPS architecture efficiency.

5 FUTURE WORK

This chapter points out solutions to improve the scheduling algorithm proposed in Chapter 4 and discusses the algorithm application on different EPS architectures. Even though the scheduling algorithm is energy driven, it may be modified to improve the quality of service. This would require a hybrid strategy that would optimize the tasks execution while maximizing the harvested energy. Also, the tasks preemption rate may be decreased if an energy input prediction parameter is added to the algorithm. Besides this, the battery aging effect shall be taken into consideration, since it affects the solar panel voltage and the tasks execution capability. Finally, particularities of the EPS architectures presented in Chapter 2 are discussed under the prism of scheduling algorithms.

5.1 TASKS EXECUTION OPTIMIZATION

This thesis has demonstrated that a satellite energy harvesting capability may be improved by scheduling its tasks. A strategy of distributing the nanosatellite tasks in lists is presented. However, this work does not provide a tool to automatically distribute those tasks. Thus, in order to adopt the proposed algorithm to any nanosatellite mission, the algorithm user must manually define and distribute the tasks of his satellite in the lists proposed in this work. The development of this tool would be a step forward in this research, allowing many research groups to use the algorithm.

In addition, Chapter 4 has addressed the tasks preemption issue. Although the proposed solution avoids missing hard deadlines for the high priority tasks, as well as avoids preempting the ones that shall not be preempted, it does not optimize their execution. Knowing the satellite tasks energy consumption, their priorities and preemptive characteristics may lead to an hybrid scheduling algorithm, which aims energy harvesting maximization and tasks execution optimization. This idea would not only improve the satellite energy harvesting capability, but it would also improve its quality of service.

5.2 ENERGY INPUT PREDICTION

The satellite movement in orbit may be predicted with models, as the one presented in Section 2.8.2. It also may be determined in real time by the ADCS. Differently from most energy harvesting systems operating on Earth, satellites are under well known and periodic solar irradiance level variations. This allows adopting energy harvesting prediction models which may significantly improve the satellite quality of service.

Most energy aware scheduling algorithms are based on the battery energy remaining capacity. This means that most algorithms analyze only the actual condition of the system. Scheduling algorithms that are able to predict the energy input may postpone, or even avoid, the preemption of tasks that normally would occur. This approach has been successfully used in energy harvesting wireless sensors networks (KOSUNALP, 2016) (SHIN; JOE, 2016). The techniques designed for wireless sensors networks may be even simplified to use in satellites. Nanosatellites ADCS real time orbit determination data, and the solar irradiance level periodicity leads to a simpler prediction problem. This may improve the satellite quality of service without causing processing overhead.

5.3 BATTERY AGING

The proposed energy-driven scheduling algorithm method could also consider the battery state of charge in order to choose the group of tasks to be performed. The battery monitoring chip already provides the state of charge information. However, the chip does not consider the aging effect caused by the charging/discharging cycles. Thus, the idea is also to include in the scheduling policy a battery aging prediction model (GHOLIZADEH; SALMASI, 2014), (SCHMALSTIEG et al., 2013). This will allow the scheduling algorithm to adapt its priorities based on the battery condition.

Normally these algorithms result in heavy calculation, which could cause an overhead for the OBDH processor, or for the EPS microcontroller. However, since battery aging takes time, one could use a ground station parameters update strategy. The ground station software could perform the heavy calculation with the EPS received data, and from time to time, update the EPS state of charge parameters by sending new ones. This idea is already under analysis at the LCS

laboratory both by the TT&C as well as by the EPS team.

5.4 EPS ARCHITECTURE DEPENDENCE

The experiments and simulations of Chapter 2 have demonstrated that the VLDO architecture was the most efficient of the four presented in this work. Therefore, instead of implementing the energy driven scheduling algorithm in the directly coupled circuit, the next step will be to migrate it for the VLDO architecture. A faster response than the DC EPS may be achieved, due to the direct relation between solar panel voltage and load current.

Besides this, a further analysis shall be performed to define the VLDO output voltage limitation. When the output voltage is clamp, the solar panel voltage dependence on load's current is even higher, since battery current decreases. Therefore, reducing the VLDO output voltage limitation to a level closer to the battery voltage plateau level (around 3.7V in this case), could improve the flexibility control on solar panel voltage.

Both MPPT architectures allow decoupling the load and battery from the solar panel. This occurs because the solar panel voltage depends both on the battery voltage as well as on the switching duty cycle. Therefore, ideally, independently from the battery voltage, the V_{mpp} may be achieved by controlling the duty cycle. Although the MPPT IC architecture leads to a simpler PCB design, it does not allow modifications on any MPPT algorithm parameter. Therefore, it may be not efficient as expected for some applications. On the other hand, the MPPT boost regulator architecture allows modifications on the algorithm parameters that may lead to operate the solar panels even more efficiently. Modifications on the perturb and observe algorithms may be adopted to improve the energy harvesting capability (KILLI; SAMANTA, 2015) (AHMED; SALAM, 2015).

Finally, an interesting option would be to reduce power losses on the MPPT boost regulator architecture (PIOVESAN et al., 2016). This decision could lead to a more complex PCB design, which always improve fails probability. However, the technological advancement in high-efficiency dc-dc converters for low-power applications is evident, including integrated solutions that may be useful in space applications (HONG; WU; WEI, 2017).

6 FINAL CONSIDERATIONS

This thesis has presented an overview on nanosatellite Electrical Power Systems. The most adopted circuit architectures have been modeled, implemented and tested. The hardware influences on the solar panels operation have been addressed, looking for solutions to improve nanosatellites energy harvesting capability.

A preliminary energy input model has been proposed, adopting solar panels form factor and the nanosatellite position in orbit. The proposed model has been improved considering the solar panel equivalent circuit, as well as the EPS circuit architecture. Simulations have been performed and validated with experiments performed at the laboratory. From the models, simulations and experimental results, one may conclude that the EPS circuit topology strongly affects energy maximization and efficiency. Even tough the VLDO architecture do not actively controls the solar panels voltage, it has been classified as the most efficient topology.

Along the EPS architectures analysis, this work demonstrated that the nanosatellite energy capability may be improved by scheduling its tasks. The discussion on the EPS hardware architectures has clarified which circuit may take advantage from the task execution influences on the solar panel control. An energy driven scheduling algorithm has been proposed, improving 8% the directly coupled energy capability for a single orbit experiment and almost 4.5% for a three orbit experiment.

Finally, a qualification procedure has been proposed an executed by testing CubeSat subsystems in a sounding rocket. The FloripaSat engineering model has been successfully tested in the VSB-30 rocket along the Rio Verde Mission, operated by the Alcântara Launching Center, Maranhão, Brazil. Besides the subsystems, a multimission platform used to test them has been space qualified. This embedded system allows a variety of experiments to be tested on sub-orbital rockets.

Therefore, all the goals proposed for this thesis may be considered achieved. The articles published during the PhD research (list presented at the end of this section) also confirms the innovative aspect and the relevance of the results presented in this work. Important improvements on this research line were pointed out along the manuscript and certainly will allow new technical and scientific challenges for the research group, university and research partners.

PUBLICATION LIST

Papers published in international journals

1. Paiva, K.V.; Mantelli, M.B.H.; Slongo, L.K.; Experimental testing of mini heat pipes under microgravity conditions aboard a suborbital rocket. Aerospace Science and Technology (Imprimé), v. 45, p. 367-375, 2015.

2. Frohlich, Antônio Augusto; Bezerra, Eduardo Augusto; Slongo, Leonardo Kessler; Experimental analysis of solar energy harvesting circuits efficiency for low power applications. Computers & Electrical Engineering, v. 45, p. 143-154, 2015.

Paper accepted in international journal

3. Betancur, Luis; Mangini, Daniele; Mameli, Mauro; Filippeschi, Sauro; Kessler Slongo, Leonardo; de Paiva, Kleber; Mantelli, Marcia; Marengo, Marco; Effect of the Condenser Temperature on the Start-up of a Pulsating Heat Pipe. Heat Pipe Science and Technology (printed), 2017.

Papers submited to international journals

4. Slongo, L. K.; Martínez, S. V.; Eiterer, B. V. B.; Pereira, T. G.; Bezerra, E. A.; Energy-driven scheduling algorithm for nanosatellite energy harvesting maximization. Acta Astronautica, 2017.

5. Slongo, L. K.; Reis, J. G.; Gaiki, D.; Martínez, S. V.; Eiterer, B. V. B.; Pereira, T. G.; Frata, M. S.; Neto, M. B.; Mera, J. P. F.; Seger, P. V. H.; Paiva, K. V.; Bezerra, E. A.; Dilillo, L.; Mantelli, M B. H.; Pre-flight qualification test procedure for nanosatellites using sounding rockets. Aerospace Science and Technology, 2017

Papers published in conferences

6. SLONGO, L. K.; BEZERRA, E. A. ; MARTINEZ, S. V. ; EITERER, B. V. B. ; PEREIRA, T. G. . Nanosatellite energy harvesting maximization through an energy aware scheduling algorithm. In: II Latin American IAA CubeSat Workshop, 2016, Florianopolis. Proceedings of II Latin American IAA CubeSat Workshop, 2016.

7. SLONGO, L. K.; MARTINEZ, S. V. ; EITERER, B. V. B. ; PEREIRA, T. G. ; KLEMZ, M. A. ; SALAMANCA, J. J. L. ; BALDINI NETO, M. ; PEREIRA, R. ; SILVA, F. G. ; LETTNIN,

D. V. ; BECKER, L. B. ; SPENGLER, A. W. ; TRAVASSOS JUNIOR, X. L. ; PAIVA, K. V. ; BEZERRA, E. A. . The Floripa-Sat experience: mission progress and satellite's development. In: II Latin American IAA CubeSat Workshop, 2016, Florianopolis. Proceedings of II Latin American IAA CubeSat Workshop, 2016.

8. MARTINS, V. M. G. ; SLONGO, L. K. ; VILLA, P. R. C. ; BEZERRA, E. A. . Soft Errors Analysis on FPGAs for CubeSat Missions. In: II Latin American IAA CubeSat Workshop, 2016, Florianopolis. Proceedings of II Latin American IAA CubeSat Workshop, 2016.

9. ARBOLEDA, L. A. B. ; MANGINI, D. ; FILIPPESCHI, S. ; SLONGO, L. K. ; PAIVA, K. V. ; MANTELLI, M.B.H. ; MARENGO, M. . Condenser Temperature Effect on the Transient Behavior of a Pulsating Heat Pipe. In: 18th Internacional Heat Pipe Conference and 12th Internacional Heat Pipe Symposium, 2016. Joint 18th IHPC and 12th IHPS, 2016.

10. SEMAN, L. O. ; SLONGO, L.K. ; HAUSMANN, R. ; BEZERRA, E. A. . O desenvolvimento de CubeSats diante da aprendizagem ativa: um levantamento com participantes de missões brasileiras. In: XLIV Congresso Brasileiro de Educação em Engenharia - COBENGE, 2016, Natal. 20 Anos de REENGE Caminhos da Engenharia, 2016.

11. PUTKA, R. M. F. ; ASCARI, L. B. ; SLONGO, L. K. ; BEZERRA, E. A. . Towards the Identification of Oversized Received Sequence Number in a FPGA Based Telemetry and Telecommand Satellite Unit. In: Simpósio Sul de Microeletrônica, 2014, Alegrete. Anais do 29° Simpósio Sul de Microeletrônica - SIM 2014, 2014.

12. SLONGO, L. K.; EITERER, B. V. B. ; MARTINEZ, S. V. ; BEZERRA, E. A. . Towards Cubesat Electrical Power System Efficient Designs. In: 1st Latin American IAA Cubesat Workshop, 2014, Brasília. Proceedings of the 1st Latin American IAA Cubesat Workshop, 2014.

13. VILLA, P. R. C. ; SLONGO, L. K. ; SALAMANCA, J. J. L. ; MARTINS, V. M. G. ; MARTINEZ, S. V. ; MARIGA, L. ; VIDAL, I. ; EITERER, B. V. B. ; MENEGON, V. E. ; COELHO, L. ; TRAVASSOS JUNIOR, X. L. ; PAIVA, K. V. ; SPENGLER, A. W. ; SOUSA, F. R. ; MELO, F. E. N. ; BECKER, L. B.

; LETTNIN, D. V. ; BEZERRA, E. A. . A complete CubeSat mission: the Floripa-Sat experience. In: 1st Latin American IAA Cubesat Workshop, 2014, Brasília. Proceedings of the 1st Latin American IAA Cubesat Workshop, 2014.

14. MARTINS, V. M. G. ; VILLA, P. R. C. ; SLONGO, L. K. ; SALAMANCA, J. J. L. ; SABINO, F. L. ; MARTINEZ, S. V. ; MARIGA, L. ; VIDAL, I. A. ; EITERER, B. V. B. ; BALDINI NETO, M. ; SPENGLER, A. W. ; MELO, F. E. N. ; LETTNIN, D. V. ; BEZERRA, E. A. . The experience of designing and developing the on-board electronics of a Cubesat in Brazil. In: 1st Latin American IAA CubeSat WorkShop, 2014, Brasília. Proceedings of 1st Latin American IAA CubeSat WorkShop, 2014.

REFERENCES

ABOUEL-FOTOUH, A. M. et al. The effect of satellite orientation on satellite surface temperature distributions. **Journal of Applied Sciences Research**, v. 2, n. 12, p. 100–107, December 2006. ISSN 1286-1292.

ADDAIM, A.; KHERRAS, A.; ZANTOU, E. Dsp implementation of integrated store-and-forward aprs payload and obdh subsystems for low-cost small satellite. **Aerospace Science and Technology**, v. 12, n. 4, p. 308 – 317, 2008. ISSN 1270-9638. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S1270963807001034>.

AGARWAL, V. et al. Development and validation of a battery model useful for discharging and charging power control and lifetime estimation. **Energy Conversion, IEEE Transactions on**, v. 25, n. 3, p. 821–835, Sept 2010. ISSN 0885-8969.

AHARA, C.; ROSSBACH, J. The scheduling problem in satellite communications systems. **Communication Technology, IEEE Transactions on**, v. 15, n. 3, p. 364–371, June 1967. ISSN 0018-9332.

AHMED, J.; SALAM, Z. An improved perturb and observe (p&o) maximum power point tracking (mppt) algorithm for higher efficiency. **Applied Energy**, v. 150, n. Supplement C, p. 97 – 108, 2015. ISSN 0306-2619. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0306261915004456>.

ALI, A. et al. Innovative power management, attitude determination and control tile for cubesat standard nanosatellites. **Acta Astronautica**, v. 96, n. Supplement C, p. 116 – 127, 2014. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576513004165>.

ARMBRUSTER, C. et al. Development of a high efficient mppt for space applications using gan power transistors. In: **PCIM Europe 2017; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management**. [S.l.: s.n.], 2017. p. 1–6.

BELLIA, H.; YOUCEF, R.; FATIMA, M. A detailed modeling of photovoltaic module using matlab. **NRIAG Journal of**

Astronomy and Geophysics, v. 3, n. 1, p. 53 – 61, 2014. ISSN 2090-9977. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S2090997714000182>.

BOMBERGER, D. C. et al. The spacecraft power supply system. **The Bell System Technical Journal**, v. 42, n. 4, p. 943–972, July 1963. ISSN 0005-8580.

BOSCH SENSORTEC. **Small, versatile 9-axis sensor module**. [S.l.], Nov 2014. Rev 1.1.

BOUWMEESTER, J.; GUO, J. Survey of worldwide pico and nanosatellite missions, distributions and subsystem technology. **Acta Astronautica**, v. 67, n. 7, p. 854–862, 2010. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576510001955>.

Brazilian Space. **Nanosat SERPENS Encerra Ciclo de Testes e Inicia Missão Espacial**. 2015. [Online; accessed 10-November-2017]. Retrieved from: <http://brazilianspace.blogspot.com.br/2015/11/nanosat-serpens-encerra-ciclo-de-testes.html>.

CAO, X. Flexible platform based micro-satellite design method. **Aerospace Science and Technology**, v. 53, p. 162 – 168, 2016. ISSN 1270-9638. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S1270963816301055>.

CASTET, J.-F.; SALEH, J. H. Beyond reliability, multi-state failure analysis of satellite subsystems: A statistical approach. **Reliability Engineering & System Safety**, v. 95, n. 4, p. 311 – 322, 2010. ISSN 0951-8320. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0951832009002531>.

CHAOUI, H.; GUALOUS, H. Adaptive state of charge estimation of lithium-ion batteries with parameter and thermal uncertainties. **IEEE Transactions on Control Systems Technology**, v. 25, n. 2, p. 752–759, March 2017. ISSN 1063-6536.

CHEN, R.-J. et al. Electric circuit modelling for lithium-ion batteries by intermittent discharging. **IET Power Electronics**, v. 7, n. April, p. 2672–2677, 2014. ISSN 1755-4535. Retrieved from: <http://digital-library.theiet.org/content/journals/10.1049/iet-pel.2013.0787>.

CHRISTOPOULOS, D.; CHATZINOTAS, S.; OTTERSTEN, B. Multicast multigroup precoding and user scheduling for frame-based satellite communications. **IEEE Transactions on Wireless Communications**, v. 14, n. 9, p. 4695–4707, Sept 2015. ISSN 1536-1276.

CLARK, C. **3rd Generation EPS plus some other stuff**. [S.l.], May 2014.

CORPINO, S.; STESINA, F. Verification of a cubesat via hardware-in-the-loop simulation. **IEEE Transactions on Aerospace and Electronic Systems**, v. 50, n. 4, p. 2807–2818, October 2014. ISSN 0018-9251.

CORSO, D. D. et al. Design of a university nano-satellite: the picpot case. **IEEE Transactions on Aerospace and Electronic Systems**, v. 47, n. 3, p. 1985–2007, July 2011. ISSN 0018-9251.

DISHAN, Q. et al. A dynamic scheduling method of earth-observing satellites by employing rolling horizon strategy. **The Scientific World Journal**, v. 2013, p. 11, 2013. ISSN 2356-6140.

DURRANI, S.; JO, K. Efficient scheduling algorithm for demand-assigned tdma satellite systems. **Aerospace and Electronic Systems, IEEE Transactions on**, v. 25, n. 2, p. 259–267, March 1989. ISSN 0018-9251.

ENVOY DATA MEMORY. **microSD 3.0 Memory Card Specification**. [S.l.], 2009. Rev. 1.6.

ERDEM, Z.; ERDEM, M. B. A proposed model of photovoltaic module in matlab/simulinktm for distance education. **Procedia-Social and Behavioral Sciences**, Elsevier, v. 103, p. 55–62, 2013.

FAIRCHILD SEMICONDUCTOR. **Dual N-Channel Logic Level PWM Optimized PowerTrench^{TM}MOSFET**. [S.l.], Oct 2001. Rev. C.

FILHO, E. M. et al. Numerical investigation of the influence of the orbit inclination on thermal control of a nanosatellite in leo. In: **II Latin American IAA CubeSat Workshop**. [S.l.: s.n.], 2016.

FR0HLICH, A. A.; BEZERRA, E. A.; SLONGO, L. K. Experimental analysis of solar energy harvesting circuits efficiency for low power

applications. **Computers and Electrical Engineering**, v. 45, n. Supplement C, p. 143 – 154, 2015. ISSN 0045-7906. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0045790614002328>.

GALATIS, G.; GUO, J.; BUURSINK, J. Development of a solar array drive mechanism for micro-satellite platforms. **Acta Astronautica**, v. 139, n. Supplement C, p. 407 – 418, 2017. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576517302412>.

GAMBLE, K. B.; LIGHTSEY, E. G. Cubesat mission design software tool for risk estimating relationships. **Acta Astronautica**, v. 102, p. 226 – 240, 2014. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576514002185>.

GAYTAN, L. D. C. H. R. et al. Dynamic scheduling for high throughput satellites employing priority code scheme. **IEEE Access**, v. 3, p. 2044–2054, 2015. ISSN 2169-3536.

GHOLIZADEH, M.; SALMASI, F. Estimation of state of charge, unknown nonlinearities, and state of health of a lithium-ion battery based on a comprehensive unobservable model. **Industrial Electronics, IEEE Transactions on**, v. 61, n. 3, p. 1335–1344, March 2014. ISSN 0278-0046.

GLOBUS, A. et al. A comparison of techniques for scheduling earth observing satellites. In: **AAAI**. [S.l.: s.n.], 2004. p. 836–843.

GOMSPACE. **NanoPower P31/P31uX Datasheet**. [S.l.], November 2017.

GONZALEZ-LLORENTE, J. et al. Improving the efficiency of 3u cubesat eps by selecting operating conditions for power converters. In: **2015 IEEE Aerospace Conference**. [S.l.: s.n.], 2015. p. 1–7. ISSN 1095-323X.

HAN, K.; LIU, Y.; LUO, J. Duty-cycle-aware minimum-energy multicasting in wireless sensor networks. **Networking, IEEE/ACM Transactions on**, v. 21, n. 3, p. 910–923, June 2013. ISSN 1063-6692.

HARTY, R. B.; OTTING, W. D.; KUDIJA, C. T. Applications of brayton cycle technology to space power. **IEEE Aerospace and Electronic Systems Magazine**, v. 9, n. 1, p. 28–32, Jan 1994. ISSN 0885-8985.

HILL, C. A. Satellite battery technology - a tutorial and overview. **IEEE Aerospace and Electronic Systems Magazine**, v. 26, n. 6, p. 38–43, June 2011. ISSN 0885-8985.

HOELLER, A. S.; FRöHLICH, A. A. Evaluation of energy-efficient heuristics for acobased routing in mobile wireless sensor networks. **Advanced Studies in Computer Science and Engineering, International Journal**, v. 4, p. 8, June 2015. ISSN 2278-7917.

HONG, X. E.; WU, J. F.; WEI, C. L. 98.1noninverting buck-boost dc-dc converter with smooth mode transition. **IEEE Transactions on Power Electronics**, v. 32, n. 3, p. 2008–2017, March 2017. ISSN 0885-8993.

INVENSENSE. **Nine-Axis (Gyro + Accelerometer + Compass) MEMS MotionTrackin^{TM}Device**. [S.l.], June 2016. Rev 1.1.

ISSI. **3V Serial Flash Memory With 133MHz Multi I/O SPI & QUAD I/O QPI DTR Interface**. [S.l.], March 2014. Rev. D.

IXYS'. **IXOLAR High Efficiency SolarBIT - KXOB22-12X1F**. [S.l.], June 2016.

JACOBSEN, L. E. **Power System of the NTNU Test Satellite**. [S.l.], December 2011.

KASAP, S. O. **Principles of Electronics Materials and Devices**. 3. ed. [S.l.]: Springer, 2006. 874 p. ISBN 978-0-07-295791-4.

KILLI, M.; SAMANTA, S. Modified perturb and observe mppt algorithm for drift avoidance in photovoltaic systems. **IEEE Transactions on Industrial Electronics**, v. 62, n. 9, p. 5549–5559, Sept 2015. ISSN 0278-0046.

KIM, H.; CHANG, Y. K. Mission scheduling optimization of sar satellite constellation for minimizing system response time. **Aerospace Science and Technology**, v. 40, n. Supplement C, p. 17 – 32, 2015. ISSN 1270-9638. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S1270963814002041>.

KIRPICH, A. et al. Design and laboratory evaluation of an electrical power system for a medium to synchronous altitude orbiting satellite. **IEEE Transactions on Aerospace**, v. 1, n. 2, p. 1335–1350, Aug 1963. ISSN 0536-1516.

KOSUNALP, S. A new energy prediction algorithm for energy-harvesting wireless sensor networks with q-learning. **IEEE Access**, v. 4, p. 5755–5763, 2016. ISSN 2169-3536.

LEE, D. Y. et al. Maximizing photovoltaic power generation of a space-dart configured satellite. **Acta Astronautica**, v. 111, p. 283 – 299, 2015. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576515000375>.

LEE, J.; KIM, E.; SHIN, K. G. Design and management of satellite power systems. In: **2013 IEEE 34th Real-Time Systems Symposium**. [S.l.: s.n.], 2013. p. 97–106. ISSN 1052-8725.

LEE, J.; KIM, T. Micro space power system using mems fuel cell for nano-satellites. **Acta Astronautica**, v. 101, n. Supplement C, p. 165 – 169, 2014. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576514001350>.

LIOR, N. Power from space. **Energy Conversion and Management**, v. 42, n. 15, p. 1769 – 1805, 2001. ISSN 0196-8904. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0196890401000401>.

LOPEZ-LAPENA, O.; PENELLA, M. T. Low-power focv mppt controller with automatic adjustment of the sample hold. **Electronics Letters**, v. 48, n. 20, p. 1301–1303, September 2012. ISSN 0013-5194.

MATUNAGA, S. et al. Titech micro-satellite model: Cansat for sub-orbital flight. In: **2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)**. [S.l.: s.n.], 2000. v. 7, p. 135–142 vol.7. ISSN 1095-323X.

MAXIM INTEGRATED. **2-Cell, Stand-Alone, Li+ Fuel-Gauge IC with Protector and Optional SHA-1 Authentication**. [S.l.], June 2011. Rev. 4.

MAXIM INTEGRATED. **High-Precision, Low-Voltage, Current-Sense Amplifier with Current Output and Chip Select for Multiplexing**. [S.l.], Nov 2012. Rev. 3.

MIYAZAKI, Y.; YAMAZAKI, M. A practical education of space engineering by using cansat and pico-satellite - fruitful collaboration with unisec for success of student satellite program -. In: **2013 6th International Conference on Recent Advances in Space Technologies (RAST)**. [S.l.: s.n.], 2013. p. 1081–1086.

MOCANU, B. et al. Determining optimal orbital path of a nanosatellite for efficient exploitation of the solar energy captured. In: **2009 First International Conference on Advances in Satellite and Space Communications**. [S.l.: s.n.], 2009. p. 128–133.

MOSER, C. et al. Real-time scheduling with regenerative energy. In: **18th Euromicro Conference on Real-Time Systems (ECRTS'06)**. [S.l.: s.n.], 2006. p. 10 pp.–270. ISSN 1068-3070.

NASSAR, M. et al. A new extension of weibull distribution: Properties and different methods of estimation. **Journal of Computational and Applied Mathematics**, 2017. ISSN 0377-0427. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0377042717306076>.

NEJI, B. et al. Hierarchical fuzzy-logic-based electrical power subsystem for pico satellite erpsat-1. **IEEE Systems Journal**, v. 9, n. 2, p. 474–486, June 2015. ISSN 1932-8184.

PAIVA, K. V. et al. Experimental tests of mini, pulsating and heat spreader heat pipes under microgravity conditions aboard suborbital rockets. In: **Proceedings of 15th International Heat Pipe Conference**. [S.l.: s.n.], 2010.

PAIVA, K. V.; MANTELLI, M. B. H.; SLONGO, L. K. Experimental testing of mini heat pipes under microgravity conditions aboard a suborbital rocket. **Aerospace Science and Technology**, v. 45, p. 367 – 375, 2015. ISSN 1270-9638. Retrieved from: <//www.sciencedirect.com/science/article/pii/S1270963815001819>.

PAIVA, K. V. et al. Experimental tests of wire mini heat pipe under microgravity conditions aboard suborbital rockets. In: **Proceedings of 9th International Heat Pipe Symposium**. [S.l.: s.n.], 2008.

PANG, C. K. et al. Nano-satellite swarm for sar applications: design and robust scheduling. **Aerospace and Electronic Systems, IEEE Transactions on**, v. 51, n. 2, p. 853–865, April 2015. ISSN 0018-9251.

PIOVESAN, T. et al. Cubesat electrical power supplies optimization - comparison between conventional and optimal design methodology. In: **2016 12th IEEE International Conference on Industry Applications (INDUSCON)**. [S.l.: s.n.], 2016. p. 1–7.

RAWY, K. et al. A self-adaptive time-based mppt with 96.2and a wide tracking range of 10 *mu* a to 1 ma for iot applications. **IEEE Transactions on Circuits and Systems I: Regular Papers**, v. 64, n. 9, p. 2334–2345, Sept 2017. ISSN 1549-8328.

RFMD. **3.6V, 100MHz to 1000MHz Linear Power Amplifier**. [S.l.], March 2014.

RODRIGUES, L. et al. A Temperature-Dependent Battery Model for Wireless Sensor Networks. **Sensors**, v. 17, n. 3, p. 422, 2017. Retrieved from: <http://www.mdpi.com/1424-8220/17/2/422>.

Samsung SDI Co. Ltd. **Specification of product for Lithium-ion Rechargeable Cell. Model ICR18650-30A**. [S.l.], November 2007.

SARAVANAN, S.; BABU, N. R. Maximum power point tracking algorithms for photovoltaic system a review. **Renewable and Sustainable Energy Reviews**, v. 57, p. 192–204, 2016. ISSN 1364-0321. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S1364032115014884>.

SCHMALSTIEG, J. et al. From accelerated aging tests to a lifetime prediction model: Analyzing lithium-ion batteries. In: **Electric Vehicle Symposium and Exhibition (EVS27), 2013 World**. [S.l.: s.n.], 2013. p. 1–12.

SHIMIZU, T.; UNDERWOOD, C. Super-capacitor energy storage for micro-satellites: Feasibility and potential mission applications. **Acta Astronautica**, v. 85, n. Supplement C, p. 138 – 154, 2013. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576512004857>.

SHIN, M.; JOE, I. Energy management algorithm for solar-powered energy harvesting wireless sensor node for internet of things. **IET Communications**, v. 10, n. 12, p. 1508–1521, 2016. ISSN 1751-8628.

SLONGO, L. et al. The floripa-sat experience: mission progress and satellite's development. In: IAA. **II Latin American IAA CubeSat Workshop**. [S.l.], 2016.

SLONGO, L. et al. Nanosatellite energy harvesting maximization through an energy aware scheduling algorithm. In: IAA. **II Latin American IAA CubeSat Workshop**. [S.l.], 2016.

SLONGO, L. K. et al. Towards cubesat electrical power system efficient designs. In: **I Latin American IAA CubeSat Workshop**. [S.l.: s.n.], 2014.

STMICROELECTRONICS. **High efficiency solar battery charger with embedded MPPT**. [S.l.], Feb 2017.

SUNDARESWARAN, K. et al. Development of an improved p&o algorithm assisted through a colony of foraging ants for mppt in pv system. **IEEE Transactions on Industrial Informatics**, v. 12, n. 1, p. 187–200, Feb 2016. ISSN 1551-3203.

TAN, Y.; YIN, X. A dynamic scheduling algorithm for energy harvesting embedded systems. **EURASIP Journal on Wireless Communications and Networking**, v. 2016, n. 1, p. 114, Apr 2016. ISSN 1687-1499. Retrieved from: <https://doi.org/10.1186/s13638-016-0602-8>.

TEXAS INSTRUMENTS. **2A Wide-Input-Range Step-Down Swift^{TM}Converter**. [S.l.], June 2008.

TEXAS INSTRUMENTS. **Mixed Signal Microcontrollers**. [S.l.], Oct 2013.

TEXAS INSTRUMENTS. **TPS5410 1A, Wide Input Range, Step-Down Converter**. [S.l.], Dec 2014.

TEXAS INSTRUMENTS. **Low-Noise, Very Low Drift, Precision Voltage Reference**. [S.l.], Feb 2015.

TEXAS INSTRUMENTS. **LP387x 1.5-A Fast Ultra-Low-Dropout Linear Regulators**. [S.l.], Jan 2015.

TEXAS INSTRUMENTS. **Voltage Monitor With Watchdog Timer**. [S.l.], Dec 2015.

TEXAS INSTRUMENTS. **ADS124x 24-Bit, 2-kSPS, Analog-To-Digital Converters With Programmable Gain Amplifier (PGA) For Sensor Measurement**. [S.l.], March 2016.

TEXAS INSTRUMENTS. **LMC555 CMOS Timer**. [S.l.], July 2016.

TEXAS INSTRUMENTS. **Low-Voltage Rail-to-Rail Output CMOS Operational Amplifiers With Shutdown**. [S.l.], Apr 2016.

TEXAS INSTRUMENTS. **TPS54540 4.5V to 42V Input, 5A, Step-Down DC-DC Converter with Eco-mode^{TM}**. [S.l.], March 2017.

The CubeSat Program. Cubesat design specification revision 13. **California Polytechnic State University, San Luis Obispo**, February 2014. Retrieved from: <https://static1.squarespace.com/static/5418c831e4b0fa4ecac1bacd/t/56e9b62337013b6c063a655a>.

VARATHARAJOO, R.; FASOULAS, S. Methodology for the development of combined energy and attitude control systems for satellites. **Aerospace Science and Technology**, v. 6, n. 4, p. 303 – 311, 2002. ISSN 1270-9638. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S1270963802011574>.

VISHAY. **P-Channel 1.8 V (G-S) MOSFET**. [S.l.], Feb 2011. Rev. A.

WANG, J. et al. Dynamic scheduling for emergency tasks on distributed imaging satellites with task merging. **IEEE Transactions on Parallel and Distributed Systems**, v. 25, n. 9, p. 2275–2285, Sept 2014. ISSN 1045-9219.

WEIBULL, W. A statistical distribution function of wide applicability. **Journal of Applied Mechanics, Transactions**, v. 18, p. 293–297, 1951.

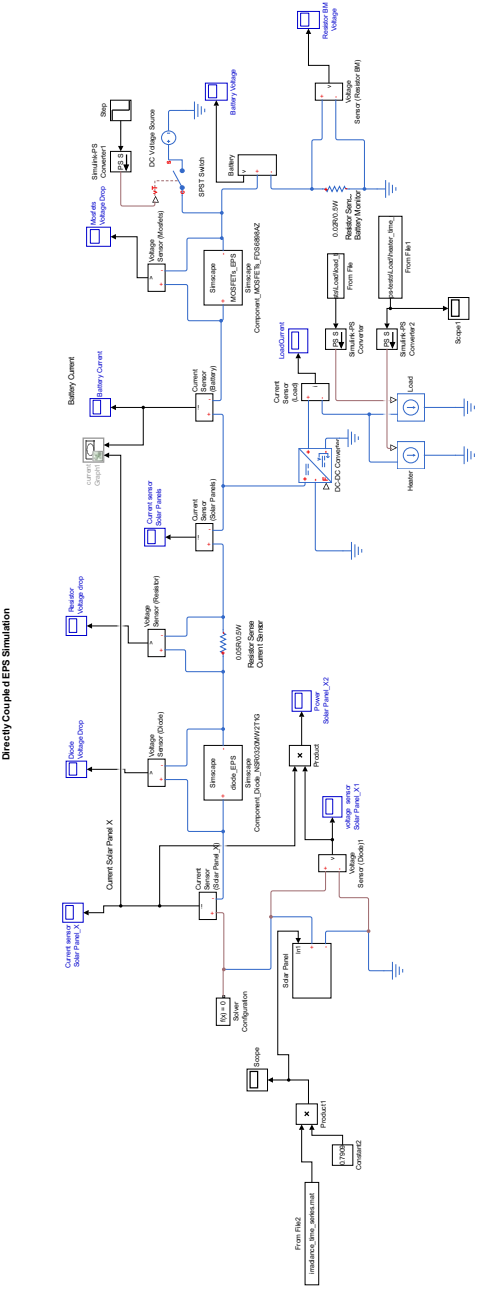
WU, Q. et al. Effect of supercapacitors supply on the high torque flywheel for satellite power system. **IFAC-PapersOnLine**, v. 50, n. 1, p. 9419 – 9425, 2017. ISSN 2405-8963. 20th IFAC World Congress. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S2405896317320323>.

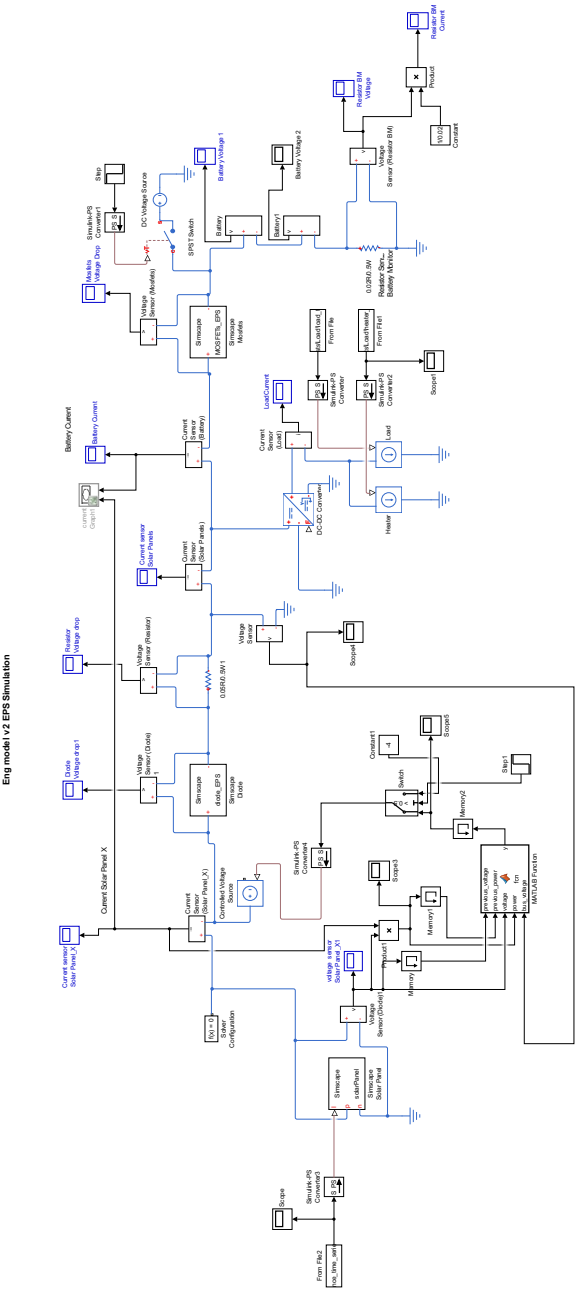
ZHENG, Z.; GUO, J.; GILL, E. Swarm satellite mission scheduling & planning using hybrid dynamic mutation genetic algorithm. **Acta Astronautica**, v. 137, p. 243 – 253, 2017. ISSN 0094-5765. Retrieved from: <http://www.sciencedirect.com/science/article/pii/S0094576516310396>.

ZHU, X. et al. Fault-tolerant scheduling for real-time tasks on multiple earth-observation satellites. **Parallel and Distributed Systems, IEEE Transactions on**, v. 26, n. 11, p. 3012–3026, Nov 2015. ISSN 1045-9219.

APPENDIX A – EPS architectures simulation diagrams

Directly Coupled EPS Simulation




Eng model v2 EPS Simulation
Current Solar Panel X
Battery Current
Resistor Voltage drop
Diode Voltage drop
Load Current
Resistor RM Current
Resistor RM Voltage
Battery Voltage 1
Battery Voltage 2
Current sensor Solar Panel_X
Current Sensor Solar Panel_X
Controlled Voltage Source
Voltage sensor Solar Panel_X1
Current Sensor (Battery)
Current sensor Solar Panels
Current Sensor (Solar Panels)
Current Sensor (Load)
Voltage Sensor (Resistor)
Voltage Sensor (Diode)
Voltage Sensor (Resistor RM)
Voltage Sensor (Mosfets)
Current Sensor (Heater)
Current Sensor (Resistor RM)
Current Sensor (Battery)
Current Sensor (Solar Panels)
Current Sensor (Load)
Voltage Sensor
Voltage Sensor (Diode)
Voltage Sensor (Resistor)
Voltage Sensor (Resistor RM)
Voltage Sensor (Mosfets)
Simscape Solar Panel
Simscape Diode
Simscape MOSFETs_EPS
Simscape Mosfets
Simulink-PS Converter
Simulink-PS Converter1
Simulink-PS Converter2
Simulink-PS Converter3
Simulink-PS Converter4
Simulink-PS Converter5
Simulink-PS Converter6
Simulink-PS Converter7
Simulink-PS Converter8
Simulink-PS Converter9
Simulink-PS Converter10
Simulink-PS Converter11
Simulink-PS Converter12
Simulink-PS Converter13
Simulink-PS Converter14
Simulink-PS Converter15
Simulink-PS Converter16
Simulink-PS Converter17
Simulink-PS Converter18
Simulink-PS Converter19
Simulink-PS Converter20
Simulink-PS Converter21
Simulink-PS Converter22
Simulink-PS Converter23
Simulink-PS Converter24
Simulink-PS Converter25
Simulink-PS Converter26
Simulink-PS Converter27
Simulink-PS Converter28
Simulink-PS Converter29
Simulink-PS Converter30
Simulink-PS Converter31
Simulink-PS Converter32
Simulink-PS Converter33
Simulink-PS Converter34
Simulink-PS Converter35
Simulink-PS Converter36
Simulink-PS Converter37
Simulink-PS Converter38
Simulink-PS Converter39
Simulink-PS Converter40
Simulink-PS Converter41
Simulink-PS Converter42
Simulink-PS Converter43
Simulink-PS Converter44
Simulink-PS Converter45
Simulink-PS Converter46
Simulink-PS Converter47
Simulink-PS Converter48
Simulink-PS Converter49
Simulink-PS Converter50
Simulink-PS Converter51
Simulink-PS Converter52
Simulink-PS Converter53
Simulink-PS Converter54
Simulink-PS Converter55
Simulink-PS Converter56
Simulink-PS Converter57
Simulink-PS Converter58
Simulink-PS Converter59
Simulink-PS Converter60
Simulink-PS Converter61
Simulink-PS Converter62
Simulink-PS Converter63
Simulink-PS Converter64
Simulink-PS Converter65
Simulink-PS Converter66
Simulink-PS Converter67
Simulink-PS Converter68
Simulink-PS Converter69
Simulink-PS Converter70
Simulink-PS Converter71
Simulink-PS Converter72
Simulink-PS Converter73
Simulink-PS Converter74
Simulink-PS Converter75
Simulink-PS Converter76
Simulink-PS Converter77
Simulink-PS Converter78
Simulink-PS Converter79
Simulink-PS Converter80
Simulink-PS Converter81
Simulink-PS Converter82
Simulink-PS Converter83
Simulink-PS Converter84
Simulink-PS Converter85
Simulink-PS Converter86
Simulink-PS Converter87
Simulink-PS Converter88
Simulink-PS Converter89
Simulink-PS Converter90
Simulink-PS Converter91
Simulink-PS Converter92
Simulink-PS Converter93
Simulink-PS Converter94
Simulink-PS Converter95
Simulink-PS Converter96
Simulink-PS Converter97
Simulink-PS Converter98
Simulink-PS Converter99
Simulink-PS Converter100
Simulink-PS Converter101
Simulink-PS Converter102
Simulink-PS Converter103
Simulink-PS Converter104
Simulink-PS Converter105
Simulink-PS Converter106
Simulink-PS Converter107
Simulink-PS Converter108
Simulink-PS Converter109
Simulink-PS Converter110
Simulink-PS Converter111
Simulink-PS Converter112
Simulink-PS Converter113
Simulink-PS Converter114
Simulink-PS Converter115
Simulink-PS Converter116
Simulink-PS Converter117
Simulink-PS Converter118
Simulink-PS Converter119
Simulink-PS Converter120
Simulink-PS Converter121
Simulink-PS Converter122
Simulink-PS Converter123
Simulink-PS Converter124
Simulink-PS Converter125
Simulink-PS Converter126
Simulink-PS Converter127
Simulink-PS Converter128
Simulink-PS Converter129
Simulink-PS Converter130
Simulink-PS Converter131
Simulink-PS Converter132
Simulink-PS Converter133
Simulink-PS Converter134
Simulink-PS Converter135
Simulink-PS Converter136
Simulink-PS Converter137
Simulink-PS Converter138
Simulink-PS Converter139
Simulink-PS Converter140
Simulink-PS Converter141
Simulink-PS Converter142
Simulink-PS Converter143
Simulink-PS Converter144
Simulink-PS Converter145
Simulink-PS Converter146
Simulink-PS Converter147
Simulink-PS Converter148
Simulink-PS Converter149
Simulink-PS Converter150
Simulink-PS Converter151
Simulink-PS Converter152
Simulink-PS Converter153
Simulink-PS Converter154
Simulink-PS Converter155
Simulink-PS Converter156
Simulink-PS Converter157
Simulink-PS Converter158
Simulink-PS Converter159
Simulink-PS Converter160
Simulink-PS Converter161
Simulink-PS Converter162
Simulink-PS Converter163
Simulink-PS Converter164
Simulink-PS Converter165
Simulink-PS Converter166
Simulink-PS Converter167
Simulink-PS Converter168
Simulink-PS Converter169
Simulink-PS Converter170
Simulink-PS Converter171
Simulink-PS Converter172
Simulink-PS Converter173
Simulink-PS Converter174
Simulink-PS Converter175
Simulink-PS Converter176
Simulink-PS Converter177
Simulink-PS Converter178
Simulink-PS Converter179
Simulink-PS Converter180
Simulink-PS Converter181
Simulink-PS Converter182
Simulink-PS Converter183
Simulink-PS Converter184
Simulink-PS Converter185
Simulink-PS Converter186
Simulink-PS Converter187
Simulink-PS Converter188
Simulink-PS Converter189
Simulink-PS Converter190
Simulink-PS Converter191
Simulink-PS Converter192
Simulink-PS Converter193
Simulink-PS Converter194
Simulink-PS Converter195
Simulink-PS Converter196
Simulink-PS Converter197
Simulink-PS Converter198
Simulink-PS Converter199
Simulink-PS Converter200
Simulink-PS Converter201
Simulink-PS Converter202
Simulink-PS Converter203
Simulink-PS Converter204
Simulink-PS Converter205
Simulink-PS Converter206
Simulink-PS Converter207
Simulink-PS Converter208
Simulink-PS Converter209
Simulink-PS Converter210
Simulink-PS Converter211
Simulink-PS Converter212
Simulink-PS Converter213
Simulink-PS Converter214
Simulink-PS Converter215
Simulink-PS Converter216
Simulink-PS Converter217
Simulink-PS Converter218
Simulink-PS Converter219
Simulink-PS Converter220
Simulink-PS Converter221
Simulink-PS Converter222
Simulink-PS Converter223
Simulink-PS Converter224
Simulink-PS Converter225
Simulink-PS Converter226
Simulink-PS Converter227
Simulink-PS Converter228
Simulink-PS Converter229
Simulink-PS Converter230
Simulink-PS Converter231
Simulink-PS Converter232
Simulink-PS Converter233
Simulink-PS Converter234
Simulink-PS Converter235
Simulink-PS Converter236
Simulink-PS Converter237
Simulink-PS Converter238
Simulink-PS Converter239
Simulink-PS Converter240
Simulink-PS Converter241
Simulink-PS Converter242
Simulink-PS Converter243
Simulink-PS Converter244
Simulink-PS Converter245
Simulink-PS Converter246
Simulink-PS Converter247
Simulink-PS Converter248
Simulink-PS Converter249
Simulink-PS Converter250
Simulink-PS Converter251
Simulink-PS Converter252
Simulink-PS Converter253
Simulink-PS Converter254
Simulink-PS Converter255
Simulink-PS Converter256
Simulink-PS Converter257
Simulink-PS Converter258
Simulink-PS Converter259
Simulink-PS Converter260
Simulink-PS Converter261
Simulink-PS Converter262
Simulink-PS Converter263
Simulink-PS Converter264
Simulink-PS Converter265
Simulink-PS Converter266
Simulink-PS Converter267
Simulink-PS Converter268
Simulink-PS Converter269
Simulink-PS Converter270
Simulink-PS Converter271
Simulink-PS Converter272
Simulink-PS Converter273
Simulink-PS Converter274
Simulink-PS Converter275
Simulink-PS Converter276
Simulink-PS Converter277
Simulink-PS Converter278
Simulink-PS Converter279
Simulink-PS Converter280
Simulink-PS Converter281
Simulink-PS Converter282
Simulink-PS Converter283
Simulink-PS Converter284
Simulink-PS Converter285
Simulink-PS Converter286
Simulink-PS Converter287
Simulink-PS Converter288
Simulink-PS Converter289
Simulink-PS Converter290
Simulink-PS Converter291
Simulink-PS Converter292
Simulink-PS Converter293
Simulink-PS Converter294
Simulink-PS Converter295
Simulink-PS Converter296
Simulink-PS Converter297
Simulink-PS Converter298
Simulink-PS Converter299
Simulink-PS Converter300
Simulink-PS Converter301
Simulink-PS Converter302
Simulink-PS Converter303
Simulink-PS Converter304
Simulink-PS Converter305
Simulink-PS Converter306
Simulink-PS Converter307
Simulink-PS Converter308
Simulink-PS Converter309
Simulink-PS Converter310
Simulink-PS Converter311
Simulink-PS Converter312
Simulink-PS Converter313
Simulink-PS Converter314
Simulink-PS Converter315
Simulink-PS Converter316
Simulink-PS Converter317
Simulink-PS Converter318
Simulink-PS Converter319
Simulink-PS Converter320
Simulink-PS Converter321
Simulink-PS Converter322
Simulink-PS Converter323
Simulink-PS Converter324
Simulink-PS Converter325
Simulink-PS Converter326
Simulink-PS Converter327
Simulink-PS Converter328
Simulink-PS Converter329
Simulink-PS Converter330
Simulink-PS Converter331
Simulink-PS Converter332
Simulink-PS Converter333
Simulink-PS Converter334
Simulink-PS Converter335
Simulink-PS Converter336
Simulink-PS Converter337
Simulink-PS Converter338
Simulink-PS Converter339
Simulink-PS Converter340
Simulink-PS Converter341
Simulink-PS Converter342
Simulink-PS Converter343
Simulink-PS Converter344
Simulink-PS Converter345
Simulink-PS Converter346
Simulink-PS Converter347
Simulink-PS Converter348
Simulink-PS Converter349
Simulink-PS Converter350
Simulink-PS Converter351
Simulink-PS Converter352
Simulink-PS Converter353
Simulink-PS Converter354
Simulink-PS Converter355
Simulink-PS Converter356
Simulink-PS Converter357
Simulink-PS Converter358
Simulink-PS Converter359
Simulink-PS Converter360
Simulink-PS Converter361
Simulink-PS Converter362
Simulink-PS Converter363
Simulink-PS Converter364
Simulink-PS Converter365
Simulink-PS Converter366
Simulink-PS Converter367
Simulink-PS Converter368
Simulink-PS Converter369
Simulink-PS Converter370
Simulink-PS Converter371
Simulink-PS Converter372
Simulink-PS Converter373
Simulink-PS Converter374
Simulink-PS Converter375
Simulink-PS Converter376
Simulink-PS Converter377
Simulink-PS Converter378
Simulink-PS Converter379
Simulink-PS Converter380
Simulink-PS Converter381
Simulink-PS Converter382
Simulink-PS Converter383
Simulink-PS Converter384
Simulink-PS Converter385
Simulink-PS Converter386
Simulink-PS Converter387
Simulink-PS Converter388
Simulink-PS Converter389
Simulink-PS Converter390
Simulink-PS Converter391
Simulink-PS Converter392
Simulink-PS Converter393
Simulink-PS Converter394
Simulink-PS Converter395
Simulink-PS Converter396
Simulink-PS Converter397
Simulink-PS Converter398
Simulink-PS Converter399
Simulink-PS Converter400
Simulink-PS Converter401
Simulink-PS Converter402
Simulink-PS Converter403
Simulink-PS Converter404
Simulink-PS Converter405
Simulink-PS Converter406
Simulink-PS Converter407
Simulink-PS Converter408
Simulink-PS Converter409
Simulink-PS Converter410
Simulink-PS Converter411
Simulink-PS Converter412
Simulink-PS Converter413
Simulink-PS Converter414
Simulink-PS Converter415
Simulink-PS Converter416
Simulink-PS Converter417
Simulink-PS Converter418
Simulink-PS Converter419
Simulink-PS Converter420
Simulink-PS Converter421
Simulink-PS Converter422
Simulink-PS Converter423
Simulink-PS Converter424
Simulink-PS Converter425
Simulink-PS Converter426
Simulink-PS Converter427
Simulink-PS Converter428
Simulink-PS Converter429
Simulink-PS Converter430
Simulink-PS Converter431
Simulink-PS Converter432
Simulink-PS Converter433
Simulink-PS Converter434
Simulink-PS Converter435
Simulink-PS Converter436
Simulink-PS Converter437
Simulink-PS Converter438
Simulink-PS Converter439
Simulink-PS Converter440
Simulink-PS Converter441
Simulink-PS Converter442
Simulink-PS Converter443
Simulink-PS Converter444
Simulink-PS Converter445
Simulink-PS Converter446
Simulink-PS Converter447
Simulink-PS Converter448
Simulink-PS Converter449
Simulink-PS Converter450
Simulink-PS Converter451
Simulink-PS Converter452
Simulink-PS Converter453
Simulink-PS Converter454
Simulink-PS Converter455
Simulink-PS Converter456
Simulink-PS Converter457
Simulink-PS Converter458
Simulink-PS Converter459
Simulink-PS Converter460
Simulink-PS Converter461
Simulink-PS Converter462
Simulink-PS Converter463
Simulink-PS Converter464
Simulink-PS Converter465
Simulink-PS Converter466
Simulink-PS Converter467
Simulink-PS Converter468
Simulink-PS Converter469
Simulink-PS Converter470
Simulink-PS Converter471
Simulink-PS Converter472
Simulink-PS Converter473
Simulink-PS Converter474
Simulink-PS Converter475
Simulink-PS Converter476
Simulink-PS Converter477
Simulink-PS Converter478
Simulink-PS Converter479
Simulink-PS Converter480
Simulink-PS Converter481
Simulink-PS Converter482
Simulink-PS Converter483
Simulink-PS Converter484
Simulink-PS Converter485
Simulink-PS Converter486
Simulink-PS Converter487
Simulink-PS Converter488
Simulink-PS Converter489
Simulink-PS Converter490
Simulink-PS Converter491
Simulink-PS Converter492
Simulink-PS Converter493
Simulink-PS Converter494
Simulink-PS Converter495
Simulink-PS Converter496
Simulink-PS Converter497
Simulink-PS Converter498
Simulink-PS Converter499
Simulink-PS Converter500
Simulink-PS Converter501
Simulink-PS Converter502
Simulink-PS Converter503
Simulink-PS Converter504
Simulink-PS Converter505
Simulink-PS Converter506
Simulink-PS Converter507
Simulink-PS Converter508
Simulink-PS Converter509
Simulink-PS Converter510
Simulink-PS Converter511
Simulink-PS Converter512
Simulink-PS Converter513
Simulink-PS Converter514
Simulink-PS Converter515
Simulink-PS Converter516
Simulink-PS Converter517
Simulink-PS Converter518
Simulink-PS Converter519
Simulink-PS Converter520
Simulink-PS Converter521
Simulink-PS Converter522
Simulink-PS Converter523
Simulink-PS Converter524
Simulink-PS Converter525
Simulink-PS Converter526
Simulink-PS Converter527
Simulink-PS Converter528
Simulink-PS Converter529
Simulink-PS Converter530
Simulink-PS Converter531
Simulink-PS Converter532
Simulink-PS Converter533
Simulink-PS Converter534
Simulink-PS Converter535
Simulink-PS Converter536
Simulink-PS Converter537
Simulink-PS Converter538
Simulink-PS Converter539
Simulink-PS Converter540
Simulink-PS Converter541
Simulink-PS Converter542
Simulink-PS Converter543
Simulink-PS Converter544
Simulink-PS Converter545
Simulink-PS Converter546
Simulink-PS Converter547
Simulink-PS Converter548
Simulink-PS Converter549
Simulink-PS Converter550
Simulink-PS Converter551
Simulink-PS Converter552
Simulink-PS Converter553
Simulink-PS Converter554
Simulink-PS Converter555
Simulink-PS Converter556
Simulink-PS Converter557
Simulink-PS Converter558
Simulink-PS Converter559
Simulink-PS Converter560
Simulink-PS Converter561
Simulink-PS Converter562
Simulink-PS Converter563
Simulink-PS Converter564
Simulink-PS Converter565
Simulink-PS Converter566
Simulink-PS Converter567
Simulink-PS Converter568
Simulink-PS Converter569
Simulink-PS Converter570
Simulink-PS Converter571
Simulink-PS Converter572
Simulink-PS Converter573
Simulink-PS Converter574
Simulink-PS Converter575
Simulink-PS Converter576
Simulink-PS Converter577
Simulink-PS Converter578
Simulink-PS Converter579
Simulink-PS Converter580
Simulink-PS Converter581
Simulink-PS Converter582
Simulink-PS Converter583
Simulink-PS Converter584
Simulink-PS Converter585
Simulink-PS Converter586
Simulink-PS Converter587
Simulink-PS Converter588
Simulink-PS Converter589
Simulink-PS Converter590
Simulink-PS Converter591
Simulink-PS Converter592
Simulink-PS Converter593
Simulink-PS Converter594
Simulink-PS Converter595
Simulink-PS Converter596
Simulink-PS Converter597
Simulink-PS Converter598
Simulink-PS Converter599
Simulink-PS Converter600
Simulink-PS Converter601
Simulink-PS Converter602
Simulink-PS Converter603
Simulink-PS Converter604
Simulink-PS Converter605
Simulink-PS Converter606
Simulink-PS Converter607
Simulink-PS Converter608
Simulink-PS Converter609
Simulink-PS Converter610
Simulink-PS Converter611
Simulink-PS Converter612
Simulink-PS Converter613
Simulink-PS Converter614
Simulink-PS Converter615
Simulink-PS Converter616
Simulink-PS Converter617
Simulink-PS Converter618
Simulink-PS Converter619
Simulink-PS Converter620
Simulink-PS Converter621
Simulink-PS Converter622
Simulink-PS Converter623
Simulink-PS Converter624
Simulink-PS Converter625
Simulink-PS Converter626
Simulink-PS Converter627
Simulink-PS Converter628
Simulink-PS Converter629
Simulink-PS Converter630
Simulink-PS Converter631
Simulink-PS Converter632
Simulink-PS Converter633
Simulink-PS Converter634
Simulink-PS Converter635
Simulink-PS Converter636
Simulink-PS Converter637
Simulink-PS Converter638
Simulink-PS Converter639
Simulink-PS Converter640
Simulink-PS Converter641
Simulink-PS Converter642
Simulink-PS Converter643
Simulink-PS Converter644
Simulink-PS Converter645
Simulink-PS Converter646
Simulink-PS Converter647
Simulink-PS Converter648
Simulink-PS Converter649
Simulink-PS Converter650
Simulink-PS Converter651
Simulink-PS Converter652
Simulink-PS Converter653
Simulink-PS Converter654
Simulink-PS Converter655
Simulink-PS Converter656
Simulink-PS Converter657
Simulink-PS Converter658
Simulink-PS Converter659
Simulink-PS Converter660
Simulink-PS Converter661
Simulink-PS Converter662
Simulink-PS Converter663
Simulink-PS Converter664
Simulink-PS Converter665
Simulink-PS Converter666
Simulink-PS Converter667
Simulink-PS Converter668
Simulink-PS Converter669
Simulink-PS Converter670
Simulink-PS Converter671
Simulink-PS Converter672
Simulink-PS Converter673
Simulink-PS Converter674
Simulink-PS Converter675
Simulink-PS Converter676
Simulink-PS Converter677
Simulink-PS Converter678
Simulink-PS Converter679
Simulink-PS Converter680
Simulink-PS Converter681
Simulink-PS Converter682
Simulink-PS Converter683
Simulink-PS Converter684
Simulink-PS Converter685
Simulink-PS Converter686
Simulink-PS Converter687
Simulink-PS Converter688
Simulink-PS Converter689
Simulink-PS Converter690
Simulink-PS Converter691
Simulink-PS Converter692
Simulink-PS Converter693
Simulink-PS Converter694
Simulink-PS Converter695
Simulink-PS Converter696
Simulink-PS Converter697
Simulink-PS Converter698
Simulink-PS Converter699
Simulink-PS Converter700
Simulink-PS Converter701
Simulink-PS Converter702
Simulink-PS Converter703
Simulink-PS Converter704
Simulink-PS Converter705
Simulink-PS Converter706
Simulink-PS Converter707
Simulink-PS Converter708
Simulink-PS Converter709
Simulink-PS Converter710
Simulink-PS Converter711
Simulink-PS Converter712
Simulink-PS Converter713
Simulink-PS Converter714
Simulink-PS Converter715
Simulink-PS Converter716
Simulink-PS Converter717
Simulink-PS Converter718
Simulink-PS Converter719
Simulink-PS Converter720
Simulink-PS Converter721
Simulink-PS Converter722
Simulink-PS Converter723
Simulink-PS Converter724
Simulink-PS Converter725
Simulink-PS Converter726
Simulink-PS Converter727
Simulink-PS Converter728
Simulink-PS Converter729
Simulink-PS Converter730
Simulink-PS Converter731
Simulink-PS Converter732
Simulink-PS Converter733
Simulink-PS Converter734
Simulink-PS Converter735
Simulink-PS Converter736
Simulink-PS Converter737
Simulink-PS Converter738
Simulink-PS Converter739
Simulink-PS Converter740
Simulink-PS Converter741
Simulink-PS Converter742
Simulink-PS Converter743
Simulink-PS Converter744
Simulink-PS Converter745
Simulink-PS Converter746
Simulink-PS Converter747
Simulink-PS Converter748
Simulink-PS Converter749
Simulink-PS Converter750
Simulink-PS Converter751
Simulink-PS Converter752
Simulink-PS Converter753
Simulink-PS Converter754
Simulink-PS Converter755
Simulink-PS Converter756
Simulink-PS Converter757
Simulink-PS Converter758
Simulink-PS Converter759
Simulink-PS Converter760
Simulink-PS Converter761
Simulink-PS Converter762
Simulink-PS Converter763
Simulink-PS Converter764
Simulink-PS Converter765
Simulink-PS Converter766
Simulink-PS Converter767
Simulink-PS Converter768
Simulink-PS Converter769
Simulink-PS Converter770
Simulink-PS Converter771
Simulink-PS Converter772
Simulink-PS Converter773
Simulink-PS Converter774
Simulink-PS Converter775
Simulink-PS Converter776
Simulink-PS Converter777
Simulink-PS Converter778
Simulink-PS Converter779
Simulink-PS Converter780
Simulink-PS Converter781
Simulink-PS Converter782
Simulink-PS Converter783
Simulink-PS Converter784
Simulink-PS Converter785
Simulink-PS Converter786
Simulink-PS Converter787
Simulink-PS Converter788
Simulink-PS Converter789
Simulink-PS Converter790
Simulink-PS Converter791
Simulink-PS Converter792
Simulink-PS Converter793
Simulink-PS Converter794
Simulink-PS Converter795
Simulink-PS Converter796
Simulink-PS Converter797
Simulink-PS Converter798
Simulink-PS Converter799
Simulink-PS Converter800
Simulink-PS Converter801
Simulink-PS Converter802
Simulink-PS Converter803
Simulink-PS Converter804
Simulink-PS Converter805
Simulink-PS Converter806
Simulink-PS Converter807
Simulink-PS Converter808
Simulink-PS Converter809
Simulink-PS Converter810
Simulink-PS Converter811
Simulink-PS Converter812
Simulink-PS Converter813
Simulink-PS Converter814
Simulink-PS Converter815
Simulink-PS Converter816
Simulink-PS Converter817
Simulink-PS Converter818
Simulink-PS Converter819
Simulink-PS Converter820
Simulink-PS Converter821
Simulink-PS Converter822
Simulink-PS Converter823
Simulink-PS Converter824
Simulink-PS Converter825
Simulink-PS Converter826
Simulink-PS Converter827
Simulink-PS Converter828
Simulink-PS Converter829
Simulink-PS Converter830
Simulink-PS Converter831
Simulink-PS Converter832
Simulink-PS Converter833
Simulink-PS Converter834
Simulink-PS Converter835
Simulink-PS Converter836
Simulink-PS Converter837
Simulink-PS Converter838
Simulink-PS Converter839
Simulink-PS Converter840
Simulink-PS Converter841
Simulink-PS Converter842
Simulink-PS Converter843
Simulink-PS Converter844
Simulink-PS Converter845
Simulink-PS Converter846
Simulink-PS Converter847
Simulink-PS Converter848
Simulink-PS Converter849
Simulink-PS Converter850
Simulink-PS Converter851
Simulink-PS Converter852
Simulink-PS Converter853
Simulink-PS Converter854
Simulink-PS Converter855
Simulink-PS Converter856
Simulink-PS Converter857
Simulink-PS Converter858
Simulink-PS Converter859
Simulink-PS Converter860
Simulink-PS Converter861
Simulink-PS Converter862
Simulink-PS Converter863
Simulink-PS Converter864
Simulink-PS Converter865
Simulink-PS Converter866
Simulink-PS Converter867
Simulink-PS Converter868
Simulink-PS Converter869
Simulink-PS Converter870
Simulink-PS Converter871
Simulink-PS Converter872
Simulink-PS Converter873
Simulink-PS Converter874
Simulink-PS Converter875
Simulink-PS Converter876
Simulink-PS Converter877
Simulink-PS Converter878
Simulink-PS Converter879
Simulink-PS Converter880
Simulink-PS Converter881
Simulink-PS Converter882
Simulink-PS Converter883
Simulink-PS Converter884
Simulink-PS Converter885
Simulink-PS Converter886
Simulink-PS Converter887
Simulink-PS Converter888
Simulink-PS Converter889
Simulink-PS Converter890
Simulink-PS Converter891
Simulink-PS Converter892
Simulink-PS Converter893
Simulink-PS Converter894
Simulink-PS Converter895
Simulink-PS Converter896
Simulink-PS Converter897
Simulink-PS Converter898
Simulink-PS Converter899
Simulink-PS Converter900
Simulink-PS Converter901
Simulink-PS Converter902
Simulink-PS Converter903
Simulink-PS Converter904
Simulink-PS Converter905
Simulink-PS Converter906
Simulink-PS Converter907
Simulink-PS Converter908
Simulink-PS Converter909
Simulink-PS Converter910
Simulink-PS Converter911
Simulink-PS Converter912
Simulink-PS Converter913
Simulink-PS Converter914
Simulink-PS Converter915
Simulink-PS Converter916
Simulink-PS Converter917
Simulink-PS Converter918
Simulink-PS Converter919
Simulink-PS Converter920
Simulink-PS Converter921
Simulink-PS Converter922
Simulink-PS Converter923
Simulink-PS Converter924
Simulink-PS Converter925
Simulink-PS Converter926
Simulink-PS Converter927
Simulink-PS Converter928
Simulink-PS Converter929
Simulink-PS Converter930
Simulink-PS Converter931
Simulink-PS Converter932
Simulink-PS Converter933
Simulink-PS Converter934
Simulink-PS Converter935
Simulink-PS Converter936
Simulink-PS Converter937
Simulink-PS Converter938
Simulink-PS Converter939
Simulink-PS Converter940
Simulink-PS Converter941
Simulink-PS Converter942
Simulink-PS Converter943
Simulink-PS Converter944
Simulink-PS Converter945
Simulink-PS Converter946
Simulink-PS Converter947
Simulink-PS Converter948
Simulink-PS Converter949
Simulink-PS Converter950
Simulink-PS Converter951
Simulink-PS Converter952
Simulink-PS Converter953
Simulink-PS Converter954
Simulink-PS Converter955
Simulink-PS Converter956
Simulink-PS Converter957
Simulink-PS Converter958
Simulink-PS Converter959
Simulink-PS Converter960
Simulink-PS Converter961
Simulink-PS Converter962
Simulink-PS Converter963
Simulink-PS Converter964
Simulink-PS Converter965
Simulink-PS Converter966
Simulink-PS Converter967
Simulink-PS Converter968
Simulink-PS Converter969
Simulink-PS Converter970
Simulink-PS Converter971
Simulink-PS Converter972
Simulink-PS Converter973
Simulink-PS Converter974
Simulink-PS Converter975
Simulink-PS Converter976
Simulink-PS Converter977
Simulink-PS Converter978
Simulink-PS Converter979
Simulink-PS Converter980
Simulink-PS Converter981
Simulink-PS Converter982
Simulink-PS Converter983
Simulink-PS Converter984
Simulink-PS Converter985
Simulink-PS Converter986
Simulink-PS Converter987
Simulink-PS Converter988
Simulink-PS Converter989
Simulink-PS Converter990
Simulink-PS Converter991
Simulink-PS Converter992
Simulink-PS Converter993
Simulink-PS Converter994
Simulink-PS Converter995
Simulink-PS Converter996
Simulink-PS Converter997
Simulink-PS Converter998
Simulink-PS Converter999
Simulink-PS Converter1000