Poster Reconfiguration of microcontroller payload driver
Romain Boyer, Frédéric Camps

To cite this version:
Romain Boyer, Frédéric Camps. Poster Reconfiguration of microcontroller payload driver. Cubsat Symposium 2023, Dec 2023, Louvain (BE), Belgium. 2023. hal-04330627

HAL Id: hal-04330627
https://hal.science/hal-04330627
Submitted on 8 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
NIMPH, Nanosatellite to Investigate Microwave Photonics Hardware

NIMPH summary

The project: NIMPH is an academic research project led by the University Space Centre of Toulouse (CSUT). It involves numerous Master students from Paul Sabatier University (UPS) and the ENSEEIHT engineering school, mainly through internships, research projects and practical work, under the supervision of professors and permanent staff from LAAS-CNRS.

Objectives:
This payload is heavy and involves significant costs. To minimize these, the project team focused on designing a bootloader that can reconfigure the onboard system in case of failure. This bootloader is designed to be used in various types of microcontroller payloads, such as those used in nanosatellites.

Nanopayload characteristics:
- Volume: 3U (300 cm³)
- Weight: 2 kg
- UHF radio (430 MHz)
- Sun-synchronous low-Earth orbit
- Power consumption: 7 W
- Mission duration: 2 years
- 3 payload slots: EDMon, XNGMon, M2M
- C phase

Reconfiguration of microcontroller payload driver

LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
Frédéric Camps, Romain Boyer, Cyril Campanella / Contact fcamps@laas.fr, rboyer@laas.fr

Reconfigurable memory architecture:

FLASH Memory Architecture:
- AtmegaS128's FLASH memory (128kB) is divided into two sections: Application section (blue, 120kB) and Bootloader section (orange, 8kB).
- Application Section holds main program, so that is flight measurement program.
- Bootloader Section holds bootloader program which is executed after a hardware reset.

Servey Section Subdivision:
- Application section is divided in six slots of 20kB.
- The first slot is in the running slot (where the current running program is located). Due to interrupt table location, only this slot can be run.
- Slot 1 to 3 are backup slots where backup programs are stored.
- Slot 4 is the security slot where the flight mission software program is stored. If an error occurred on UART bus, this slot is restored in running slot number 0.
- Finally, slot 5 is a configuration slot which keeps the actual configuration of memory as number of pages in each slot, I²C and software versioning, bootloader versioning ...

Communication protocol:

Reconfiguration Commands:
- On-Board Computer has four commands available in order to proceed to a reconfiguration function. Therefore, these commands are integrated in the bootloader program:
  - Reconfiguration Slot: Ask for write a program in a specific slot number. This slot number is asked by AtmegaS128 before data transfer.
  - Restore Slot: Ask for copy a specific slot number into the running slot (slot number 0).
  - Restart: Ask for a restart of the running program. This command simply exit bootloader program and jump to application section (address 0x00000).
  - Get configuration: Ask for the information in the configuration slot.

Communication protocol and Safety:
- Communication protocol is essential to ensure resiliency in reconfiguration. Here are some examples of communication messages between OBC and AtmegaS128.
- Robust design of this UART system is essential for the mission and durability of the nanopayload. Effective management of Hamming distance ensures increased resilience to space radiation, thereby ensuring the success of this space mission.
- To secure heavy data transfer, cyclic redundancy code (CRC) is applied to data. The algorithm used is CRC16-CCITT FALSE (ESA ECSS-E-ST-70-41C, Annex B.1). The same CRC is used in PUS protocol (Ground Station – NIMPH communications). This ensure resiliency of the written bootloader.