	QlevEr Sat PROJECT
	Student "stage" subject 2021-22
	CubeSat Payload-to-Platform interface

The Grenoble University Space Center (CSUG) (UGA - INP):

The project will be supervised by the CSUG (120 rue de la Piscine, 38400 Saint Martin d'Hères).

General context:

Started in 2020, the QlevEr Sat project aims at bringing Artificial Intelligence (AI) onboard a demonstration CubeSat for Earth Observation (EO). The overall project is led by the CSUG in collaboration with the Multidisciplinary Institute in Artificial Intelligence (MIAI UGA) and Teledyne e2v. QlevEr Sat will be surveying specific regions for deforestation. A 5m resolution is necessary to observe daily changes on a given target area. As the number of satellites increases, embarking AI algorithms directly on board will drastically reduce the bandwidth required for ground data transmission: only the post-analysis results can be downlinked, rather than images themselves.

As part of the Preliminary Definition phase (Phase B) of the overall project, a QlevEr Sat demonstrator is being designed as a ground engineering model, capable of acquiring some images, running embedded AI algorithms developed by the DSE partner (Data Science Experts) and transmitting the results to the main On Board Computer assembled by the U-Space partner (the 6U platform provider).

The payload of the satellite (CSUG part) only includes an optical lens, an imager, an FPGA bridge (camera function) as well as an ARM processor running a custom Linux distribution (based on Ubuntu 18.04 LTS). U-Space will deal with all other navigation and telecommunication equipment.

Within this phase, the present student project addresses the payload's embedded software system only (no optics, electronics, mechanics or thermal aspects). The AI itself is not included.

https://www.csug.fr/menu-principal/projets/qlever-sat/qlever-sat-751384.kjsp

CSUG	**QlevEr Sat PROJECT**
	Student "stage" subject 2021-22
	CubeSat Payload-to-Platform interface

Objective:

Investigate, configure, prototype and test an UART link to be used as interface between the satellite's payload and platform, in conformance to their respective requirements (QlevEr Sat mission requirements and technical constraints). An additional preliminary study concerning data transfer via QSPI between the FPGA bridge and Linux may be envisaged if time allows.

Description

QlevEr Sat acquires raw colour images (4096 x 4096 pixels) and computes associated binary maps that identify forest / non-forest or clouds / non-clouds areas, thanks to AI algorithms running on the ARM processor, on payload side. These binary maps as well as housekeeping telemetries (TM, data relating to the payload devices) have to be transferred to the platform, to be sent later to the ground. Conversely, the platform (PF) transmits to the payload (PL) the telecommands (TC) received from the ground.

For the considered use cases, and given the available interfaces, UART bus is a good candidate for the communication between the payload and the platform, and is in principle fast enough to ensure conveniently the above-mentioned transfers (processed data and TM/TC).

We intend to study and configure this link on the payload side, then we will test it when the PL is connected to another device. At the protocol level, the PUS standard will be considered. Apart from this, an additional work on configuring and using QSPI in Linux may be envisaged.

The tasks involve:

- Identify and configure in Yocto the appropriate driver(s) for UART (and QSPI), then create a new custom Linux for ARM µP, based on Ubuntu 18.04 LTS distribution.
- Find an approach so as to use these drivers from the user space, e.g. from a C program.
- Consider the Packet Utilisation Standard (PUS) used by the PF OBC, in order to specify the future data communications between the PL and the PF.
- Prove with appropriate tests that the UART link actually works.
- For each HW test, write a test plan, test all the options and document the results.
- Keep a written record of all the steps of all stages of the work.

Keywords: embedded Linux, Yocto, device tree, UART, QSPI, Qormino®, C (C++).

NB: The AI Python code is already developed.
The working language for all documentation is English.

Contacts:

Software Supervisor Engineer: lian.apostol@@univ-grenoble-alpes.fr
QlevErSat Project Manager: tania.mcnamara@univ-grenoble-alpes.fr