Adrien CHARDON

✉ adrienchardon AT mailoo.org
cv.nodraak.fr/en.pdf

Adrien is a software and electronics engineer, with an expertise in embedded and critical systems. He is interested in robots, launchers and inter-planetary probes. He cares about code quality, automation and open source software.

Work Experiences

Software Engineer - ASTRIS kickstage

PTS
Since 2021-01
Berlin, Germany

- The ASTRIS kickstage is a 5 tons spacecraft developed by Ariane Group to be flown on Ariane 6 to provide additional capabilities and flexibility to payloads. In collaboration with Ariane Group, **PTS designed the full avionics of ASTRIS** up to PDR. The PTS developed avionics covered the **electronics of the OBC**, telemetry, power, data acquisition and valve and pyro control subsystems, as well the **low level software of the OBC, and complete software of the other subsystems**.
- Within the software team, I contributed to the following topics:
- **Architecture and design**: requirements flow down, functional architecture decomposition, design of the CANopen-based internal bus.
- **Documentation** (following ECSS standard): unit and validation test plans, software design document (SDD) and interface control document (ICD) ; With a focus on the Datapool, Telemetry pipeline and IOs drivers (UART, CAN, SPI).
- **Implementation and integration**: ARM uc BSPs (Zynq 7020, SmartFusion2), RTOS (RTEMS), NASA cFS, drivers (UART, CAN, SPI). Version control and CI (Git and GitLab), maintaining three execution targets for different levels of tests (x86/Linux, ARM/QEMU, ARM/Zynq), code coverage (GCOV), static analysis (Cppcheck, SonarQube, MISRA), UTs (VectorCast).

Software Engineer - CLAM (IoT for logistics)

PTS
Since 2021-01
Berlin, Germany

- CLAM is a monitoring and optimization solution for logistics (containers and trucks): it provides the container loading level, position, temperature, humidity, ...
- **As the main responsible for software**, I researched, designed and developed both the embedded software running in the containers and the backend to administrate the units and visualize the data:
- **Embedded software:** ESP32, GSM and GNSS, distance sensor (VL53), environmental sensors (temperature, humidity), IMU.
- **Backend:** Django, SQLite, Grafana, Nginx.
- **Electronics:** prototype PCB layout, routing and soldering, testing.

Software Engineer - Exomars

GMV
2019-2020
Madrid, Spain

- Exomars rover: as the **main technical responsible** I developed and validated a middleware to interface the autonomous navigation libraries (CNES) with the rover platform (Airbus UK). During development, I configured a **CI pipeline to ensure high quality code and up to date reports**: running automated test on x86 and Sparc, tracking coverage and static analysis results, and automatically generating documentation.
- Exomars cruise and descent modules: I **helped in the development of the GNC algorithms**: code and bug fixes, tests and validation, and improvement of the software quality metrics.
  - ***Skills:*** *programming (embedded C, Python), unit and integration testing (VectorCast, Tsim, Leon 2 FT on Rasta), quality tools (Cppcheck, Gitlab CI, code coverage), software development standards (MISRA-C, ECSS-40B).*

Version public - 2022-10-21 - 47ad241

Project Leader (Student) - Smallsat ECE3Sat

ECE Paris
2017-2018
Paris, France

- In collaboration with a team of 25 people, I designed and built a nano satellite in order to study a new de-orbit technique using Earth's magnetic field.
- **I managed the team** (5 people) responsible for the satellite architecture and the on-board computer. **We implemented the on-board communication** bus with CAN and ASN.1 to ensure a reliable communication between subsystems.
  - ***Skills:*** *distributed architecture design, sizing and µcontrollers choice, team management and coordination.*

Software Engineer (Student) - Robot Gali X

ECE Paris
2016-2017
Paris, France

- With a team of 5, **we designed and built an autonomous robot** for the French Robotic cup.
- **In charge of the software**, I designed a **distributed architecture** to allow easier reuse for the future robots.
- I implemented a **telemetry GUI** to monitor the robot status and a **simulator** to assess the performances (recompilation of ARM code for execution on x86).
  - ***Skills:*** *embedded C/C++, Python (telemetry GUI, software-in-the-loop simulation), drivers for CAN bus and other peripherals, ARM µcontrollers, Git.*
  - *Video demo and source code available at* `https://cv.nodraak.fr`

Education

2018 — **Advanced Master**, *TAS Astro: space systems design.*
ISAE-Supaero - Toulouse, France

2013 - 2018 — **Engineering Degree**, *Majoring in embedded systems.*
ECE Paris - Paris, France

2015 - 2016 — **Bachelor of Science**, *Electronics & IT.*
Aalborg University - Aalborg, Denmark

French	**Native**.
English	**Fluent (C2)**.
German	**Conversational (B1)**.
Spanish	**Conversational (B1)**.

Hobbies

Writing	`https://blog.nodraak.fr`
Tourism	Museums and cities
Sports	Running (HM: 1h29), swing dancing, roller/ice skating
Learning and making	Software and robotics projects - Recently: learning Rust lang
Playing	Kerbal Space Program

Version public - 2022-10-21 - 47ad241