

## Summary

---

- Come from EPITA a computer science engineering school with a mainly C/Unix oriented background.
- Specialization in programming microcontrollers.
- Good skills on motor control, basic skills on Inertial Measurement Unit.
- Explored by myself other domains like electronics or feedback and automatic control systems (Scilab/Scicos).
- Worked on several drones successfully commercialized.

## Software Development Experience

---

### Parrot Faurecia Automotive

*2017 – today: Automotive middleware*

- C/C++ development in hypervised solution with 3 virtual machines: Linux, Android and real time OS.
- C++/java development in the Android (N.car then O) car services framework. Implemented the Vehicle Hardware Abstraction Layer (VHAL) aka an abstraction of the physical transport layer (like CAN/LIN) for other Android services (java).
- Qt and QML development of a proof of concept car cluster/dashboard displaying vehicle information from the VHAL and tested on a concrete car.
- Linked the vehicle climate actuator to the Android climate control application through the VHAL.
- Esterel, C development of the prototype of an ASIL real time application supervising the cluster display through the support of hardware.
- Made protocol libraries for allowing virtual machines to communicate together (display output checker).
- Received a formation on ISO-26262.
- Wrote a methodology for knowing if a proposed architecture will run in real-time (Petri, Grafset, Max-Plus algebra).
- Made the portage of Android N.car car services to Android L for an older Parrot project.
- C, C++, Java, QML, Qt, Ethernet socket, unit tests. Architecture specification. Methodology Scrum, Gerrit, Git, Docker, gdb server.

### Parrot Drones

*Jan. 2011 – Jan. 2016: Microcontroller developer*

*Development alone on the firmware of several generations of Electronic Speed Controller (ESC) for the following drones: AR Drone 1 and 2, Bebop 1 and 2, Exom 1, Disco 1, Anafi. ESC are boards driving brushless motor with a propeller.*

- Wrote the specification document (SA/RT).
- Wrote the brushless motor model and its regulator in Scicos.
- Developed multiple firmware based on the 6 steps motor commutation. Motors speed are drive with a closed loop regulation for speed: PID, feedforward, Kalman filter for the speed estimation. Power On Self Test are made like detecting if the motor is plugged to its ESC. C/ASM for Atmel Atmega8. C and logic gates programming (FPGA-style) for Cypress PoSC3 and Cypress PoSC5LP. A single chip drives four motors.
- Developed bootloader firmware, bootloader protocol, and flasher tool. Motor firmware are upgraded when the drone user updates its smart-phone application.
- Developed protocols based on I<sup>2</sup>C, spi, UART (C/Linux/ $\mu$ C).
- Developed motor test bench controlling peripherals (power supply, tachymeters...).
- Worked with the hardware team for evolving the hardware of the ESC.

### Parrot Drones

*Jan. 2016 – Jan. 2017: Microcontroller developer*

*Evolution of motor commutation. Worked on Field Oriented Control (FOC) solution: motor torque and speed are estimated and controlled.*

- Internship supervisor for implementing Field Oriented Control firmware for controlling a motor in angle for a gimbal project.

- Wrote a FOC model with Scicos for simulation.
- Firmware finally not developed internally by Parrot.
- Supervized the firmware delivery (developed by Active Semi PAC2553). In contact with a US engineer for the firmware evolution and development.
- Integrated the firmware for controlling in position a gyro-stabilized pod (aka a camera gimbal). Integrated the firmware for spinning propellers.
- Developed bootloader firmware, bootloader protocol, bootloader tool and ICSP flasher tool based on STM board.

## Parrot Drones

*2 months: Thermal calibration for IMU sensors*

*Development of a fast IMU thermal calibration process embedded in the drone. Project constraint: thermal calibration is made during the factory process and shall be as fast as possible.*

- Studied the state of art, worked with thermal chamber and turntable pods, write specification document for the factory process.
- Sensors thermal biases are removed.
- The IMU temperature is maintained fixed by the drone (closed loop regulation) during its fly.
- Worked with hardware team, developed driver for reading sensors.
- One patent realized.

## Parrot Drones

*Canceled drone based sub-projects*

- Read outputs of different remote controls like PPM, PPM-Sumed, Spektrum by an ATmega8 (canceled project).
- Reading the state of art for Simultaneous Localization And Mapping (SLAM) algorithm.

## Eurogiciel

*2007 – 2010: Consulting for Sagem*

*Worked on fixing bugs in the code source of an aircraft inertial measurement unit (IMU).*

- Analysis the IMU code source, locate and fix the bug.
- Reproduce the issue on a test bench by writing non-regression scripts, summarized with document..
- Respect the DO-178B norm.
- Got a formation on the Esterel V6 software.

## Internships

---

### INRIA

*September – December 2005: Code generation*

*Four months internship at the National Institute for Research in Computer Science and Automatic Control on the development of SynDEX a software used for generating the code of distributed real time applications.*

- Development in OCaml, CamlTk, OCamllex, OCamlYacc and M4 for evolving GUI, lexer and parser for allowing it to generate the hosted code source that the user can store inside the GUI.
- Learning the law control and writing tutorials for SynDEX where cars are following the previous car.

### INRIA

*January – July 2007: Car platooning*

*Continuation of the previous internship. Development of an algorithm, based on the video stream of a single low cost camera, for platooning electric cars named CyCab.*

- Retro-engineering the CyCab electronic architecture (RTAI Linux and MPC555 boards) as well as its law control (C and assembly code).
- Implement the original CyCab law control in Scicos and translated it into SynDEX.
- Implement the platooning algorithm based on image processing (PID control for the maintaining the distance with the previous CyCab, Kalman filter for tracking the CyCab in the picture), read pictures from a firewire camera.
- Buy a newer embedded desktop allowing a RTAI Linux and for displaying the camera.
- Test on real situation with the CyCab.

## Education

---

**2001–2007** EPITA school: Ecole Pour l'Informatique et les Techniques Avancées.

**French** Mother tongue.  
**English** Fluent, 775 at TOEIC.  
**Spanish** Beginner.

## Computer skills

---

**Languages** C, C++11, Esterel, OCaml, Delphi, Forth, currently learning Julia.  
**GNU/Linux** git, bash, M4, Makefile, gdb(server), valgrind, emacs, flex/bison.  
**Lib** GTKmm, OpenGL core, L<sup>A</sup>T<sub>E</sub>X.  
**Tests** gmock/gtest, crpcut, cppunit, gcov.  
**CI** Travis-CI, OpenSuse Build, coverity scan.  
**Sys. Analysis** Real-Time Structured methods (SART), plantuml.  
**Embedded** Digital electronics, oscilloscopes, Cypress PsoC3/5, Active Semi PAC2553, Cortex M0, PIC16F (assembly language), dsPIC30F (C language), UART, I<sup>2</sup>C, SPI.  
**Control** Scilab, Scicos, SynDEX, Automatic control systems learnt with internships and projects.  
**Norms** ISO-26262, DO-178B.

## Personal Github projects

---

**Chess** Chess project in C++ for learning neural network. The AI shall learn how to move pieces on the chessboard. Learning Julia language and frameworks like Knet and TensorFlow.

**SimTaDyn** Project for creating dynamic geographic maps and manipulate them as spreadsheet. C++, Forth, GTKmm, OpenGL core.

## EPITA projects

---

**Helicopter** Studies and realization of a four-rotor heads micro indoor model helicopter : computer-based control (Scilab) and embedded system (2 dsPIC micro-controllers, acceleration sensor, 4 brushed motor drivers) 1 year, 1 people, annual report.

**Car** 3D simulator game made in Delphi and OpenGL. The player drives a car inside a city with its traffic jam. The car dynamic is simulated (1 year, 2 peoples, annual report).

**Processor** Study and simulate of a CISC micro-processor based on an FPGA emulation software (MaxPlus2).

**Bash** Developed an Unix Shell commands interpreter in C (1 month, 6 peoples).

**Tiger** Developed a Tiger language compiler in C++, Flex and Bison (4 months, 4 peoples).

**Lisp** Developed a Common Lisp language interpreter in OCaml (1 week, 3 peoples).

**Forth** Developed a Forth language interpreter in C, for the SimTaDyn project (alone). Forth is a stack language running inside a virtual machine. Forth has no syntax but the syntax can self evolve.

**Corewar** Developed in C a virtual machine executing assembly language files simulating concurrent viruses with the theme a race tournament.

**Recalage** Find a rigid transformation which matches a cloud of points with a surface (2 weeks, 1 people).

**PDE** Solver of partial derivative equations with the finished difference method written in C.

**Bistro** Big numbers calculator written in C (2 weeks, 2 peoples).

## Other

---

Fish tank, chess player, role play games.