Marc Garrido Casas
EXTENDED CURRICULUM VITAE

Firmware / Software Embedded Engineer

+34 618 133 038

Garridocho@gmail.com

https://github.com/Garriden

https://www.linkedin.com/in/marcgarrido/

https://marcgarrido.es

PERSONAL DETAILS

Surname : Garrido
First name : Marc
Date of birth : 14-10-1995
Place of residence : Barcelona
Nationality : Spanish

PROFILE & AMBITION

I have eight years of experience as a **Software Embedded Engineer** (sometimes labeled as **Firmware Engineer**). All my work is related to **Low-level** programming, **Real-time** systems and High Performance Computing (**HPC**). My specialization lies in software efficiency, especially at deeper level. I can be laveled as a **Performace Engineer**.

My expertise is in modern **C++** (C++17 and C++20).
I also have worked in C and Python. I have some familiarity with languages such as Go or C#.
My career has taken me through a variety of environments, I worked in small, medium and large companies, startups, both spanish and international companies.

I thrive on bringing new ideas to life.
I constantly look for new challenges, allowing me to expand my comfort zone.

I eagerly seek opportunities where my ideas could be heard while giving innovation.

My goal is to continuously improve my technical skill while exploring fruther domains such a robotics, automotive, probability, and artificial inteligence, which fascinate me and offer new paths for exploration.

EDUCATION

	Period	Education & Projects	Diploma
Universitat Oberta de Catalunya (UOC)	2020 -2022	Master's Degree in Data Science	Yes
Universitat Politècnica de Catalunya (UPC)	2013 -2019	Bachelor's Degree in Informatics Engineering Specialization in Computer Engineering	Yes

COURSES

Period	Description course
3 days	Debugging C in automotive with Trace
3 days	Real-Time Design Patterns for Modern C++
40 hours	Deep Learning
40 hours	Machine Learning & Python
Udemy Courses	Wireshark software
Udemy Courses	Golang language
40 hours	HTML / CSS / Javascript
40 hours	3D Design & Blender

WORK EXPERIENCE & ADDITIONAL QUALIFICATIONS

Work experience:

	Period	Department and function
Deep Detection (Startup)	12/2023 - 04/2025	Senior Software Embedded Engineer / Firmware Engineer. Photon counting (R+I+D) C++20 Successfully designed and implemented new functionalities to optimize the camera, achieving the maximum speed while improving the robustness and reliability within a tight timeframe. Keywords: C++, Real-Time, Embedded.
Ficosa automotive, Panasonic	02/2022 - 12/2023	Senior Software Embedded Engineer / Firmware Engineer (R+I+D) ADAS. C Implemented new functionalities based on client requirements and bug-solving. Developed a personalized Bootloader using AUTOSAR classic. Worked in a Git workflow environment, with Can Bus, Automotive tools. Reduced 20% of bugs during production. Keywords: C, Automotive, Autosar, Real-Time, Embedded.
Derivco sports, Betway	07/2021 - 02/2022	Software Engineer (Backend), Betting Services. GO / Devops Developed new code in Go and refactored existing code in Elixir. Worked with DevOps, Azure, Continuous Integration pipelines, and unit-tests. Collaborated with an international team following strong Agile 2-week demos. Keywords: Go, Dev-Ops.

Some of the positions are described in more detail in the appendix

	Period	Department and function
HP Printing Solutions	09/2019 - 07/2021	Software Embedded Engineer / Firmware Engineer (R+I+D) for Large Format Printers. C++11 Added new functionalities, fixed bugs, and added Gtests with C++ Successfully developed C++ code for real-time servo movements. Achieved a 2x increase in printing speed while maintaining quality. Improved the lifespan of a printer motor by 4x and redesigned its movement. Conducted master's thesis on optimizing PID values for a specific motor. Worked in a large team following agile methodologies with strict deadlines. Utilized embedded systems, RTOS, SVN, and cross-compilation Keywords: C++, Real-Time, Embedded.
Promax Electronica (Internship)	02/2019 - 07/2019	Software Embedded Engineer / Firmware Engineer (R+I+D) C++17 Conducted bachelor's thesis on creating a cost-effective FM decoder. Keywords: C++, Real-Time, Embedded.
RearSim (Startup)	09/2017 - 07/2019	Computer Engineer (R+I+D) Backend, Frontend, Firmware. DLLs. C+ / C# / Arduino Programmed the movement of a physical motorcycle simulator. Developed a C# software with real-time control using threads. Improved the product reaction time by 5x by optimizing servos and PID. Keywords: C++, C# Real-Time, Embedded, DLL.

EXPERTISE

	Average	Good	Very Good
Operating Systems			
Windows		**X**	
Linux			**X**
RTOS			**X**
Programming/Software/Design			
C++20			**X**
C++			**X**
C			**X**
Python		**X**	
C#	**X**		
Go	**X**		
Bash		**X**	
Equipment/Hardware/Tools			
GIT			**X**
SVN		**X**	
Visual Studio Code			**X**
CMake		**X**	
Wireshark		**X**	
Trace		**X**	
Vector CANoe		**X**	
Planning and management			
Jira			**X**
Agile / Scrum			**X**
Autosar		**X**	
Other Strengths			
Problem Solving			**X**
Debugging			**X**
Multi-Threading		**X**	
Web Scraping		**X**	
Memory Scanning		**X**	
TCP / UDP sockets			**X**
Arduino		**X**	
Ethical Hacking	**X**		

LANGUAGE SKILLS

	SPEAKING	WRITING
English	Fluent	Fluent
Spanish	Native	Native
Catalan	Native	Native

EXTRACURRICULAR ACTIVITIES and PROJECTS

	Period	Function
Autonomous Bot	From 2022	Created a C++ bot for autonomous play in an online game. Implemented memory scanning, threading, and TCP/IP packet handling.
Sensor parking (arduino)	2021	Inserted a distance sensor on my car to beep while I'm parking.
Web scraping and Statistics	From 2018	Extracted information from HTML using Python and processed the obtained data.
Airport management program	2016	Developed an airport management program with heuristics using C++.
CodeClub Biblioteques	2016	Teach programming to a 8-10 years old kids in libraries with Scratch.

HOBBIES

When I'm not working, you'll probably find me playing chess, hitting the tennis court, going into the forest for a run, or calmly reading a book.
I also like to code in my free time – it's kind of a hobby that sometimes feels like work, but in a good way. I love to invent new things.
When I'm more relaxed, I really enjoy going out with friends, going to the cinema or, again, reading a new book – anything from sci-fi to self-awareness.

PORTFOLIO

Deep Detection (Startup)
12/2023 - Present
Senior Software Embedded Engineer / Firmware Engineer (R+I+D).

Project:

Develop the embedded software of X-ray imaging cameras in C++.

Challenges:

- Integration of the final industrial product and the construction of the future technology platform while creating scalable and optimized software systems that meet the client needs across the manufacturing industries.
- Review and optimize system code to achieve high speed, multichannel data transfer and deliver the command control requirements.
- Program, test, debug and document software following the company procedures.
- Assure the interfaces to onboard logic and system on chip processors of the hardware are functioning. Incorporate configurable features at the manufacturer and user level.

Tasks:

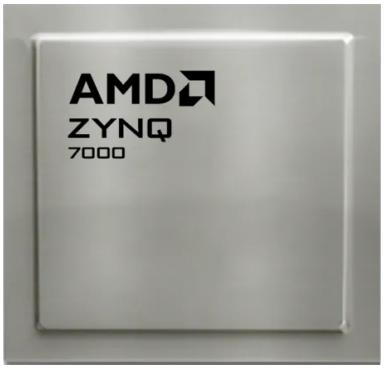
- Requirements analysis: Understand the needs of the client.
- Software design: Choosing the right architecture of the embedded software with the needed algorithms and the right protocols communications to make it work efficiently.
- Implement: Adding new features, following the tickets and the documentation related to the electronic board behavior. Zynq7000 (Arm Cortex A9), Petalinux.
- Debugging: Identifying and investigate when and why is the bug occurring. Modify the code and finally fix the bug, test it and prevent similar bugs in the future.
- Integration: Add the work of external engineers to our product, testing it and integrating it to ensure a correct behavior. Deploy the product and have a correct version scheme in git between releases.
- RTOS: Understand the scheduler behaviour behind the Petalinux to maximize the efficiency of the two main threads. Characterize processes timings.

Results:

- 20% improvement in camera speed acquisition, merging two threads into one in a critical part of the code. I discovered those two thread were dependent from each other, it would solve time eradicating useless context-switching.
- Migrated into a new board, from one core to two cores, putting the non-critical threads on the second core, making the first core more CPU efficient when running the main threads.
- Integrate new version of the embedded application, reading into memory instead of DMA, done by an external engineer, into our product.
- Made our embedded application robust, error prune, being able to work for hours and hours without crashing while working with the desired features.
- Discover and solve a critical bug, present for a long time, where the camera had a bottleneck where it couldn't reach the maximum speed.
- Implemented a new variant of the UDP communication protocol to send much more information in a single packet, reducing the number of packages sent by 90% and increasing the throughput by 15% without losing data.

Tools / Technical Environment:

- C++
- Zynq7000 (Arm Cortex A9)
- Real-Time
- MultiThreading
- UDP / TCP sockets
- Embedded
- Linux
- Petalinux
- Version Control: Git



FICOSA Automotive, Panasonic
02/2022 - 12/2023
Senior Software Embedded Engineer / Firmware Engineer (R+I+D) ADAS.

Project:

Develop the embedded for a software automotive board in C.
Focus on the Bootloader inside the Basic Software team. (ADAS)

Challenges:

- Client Requirements: Work with system engineers to meet the requirements to add new functionalities based on client. Re watch it from time to time and bug-solving the possible issues.
- Keep developing a personalized Bootloader coded with state machines and interruptions using AUTOSAR classic. Maintaining the real-time performance and respecting the timings for each command.
- Work in “V-shape” workflow environment, 2 week sprints in Jira with releases.

Tasks:

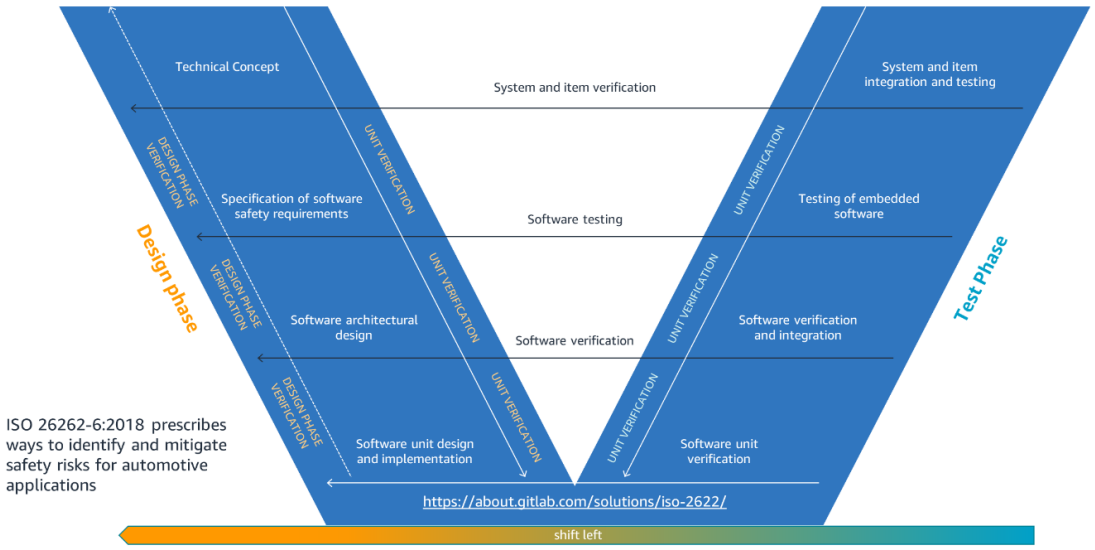
- Be in charge of the Bootloader, Second Bootloader, MCU, DCM and other small modules in the Basic Software Autosar.
- Implement: Adding new features, following the tickets and the documentation and ISOs / Doors. STM32 (Arm Cortex A9).
- Debugging: Investigate, replicate and solve critical bugs related to real-time. Adapt the code maintaning the requirements, test it and prevent. Used a Trace and CANoe tool to debug better. Also I needed to create and revive unit tests.
- Integration: Migrate old architecture into the common code developed in the Basic Software team, Bootloader included.
- Understand the personalized real time operating system already coded bare metal to fit all the hard deadlines. Preemptive behavior implemented via interruptions.

Results:

- About 20% improvement in bugfix burnout during the last part of the project.
- Successfully integrated the Bootloader into two different projects.
- Made our Basic Software robust, scalable and easy to fit into another projects.
- Unit test the entire Basic Software and applying MISRA-C.
- Solve “last-hour” issues when deploying the Friday release.

Tools / Technical Environment:

- C
- STM32 (Arm Cortex A7)
- Autosar
- Bootloader
- Misra C
- Wireshark
- Real-Time
- Bare metal
- CAN
- Vector CANoe
- Trace
- Embedded
- Linux
- Version Control: Git / Bitbucket



HP Printing Solutions
09/2019 - 07/2021
Software Embedded Engineer / Firmware Engineer (R+I+D).

Project:

Develop the embedded software of Large Format Printers in C++.

Challenges:

- Working on a project which our main objective was to accomplish the same results and quality as the old printer, but two times faster the speed.

Tasks:

- Implement: Adding new features, following the tickets, bug-fixing.
- Integration: Work with other software engineers, mechanic and electronic engineers to evolve our product.
- Boot and maintain the new printer prototypes.

Results:

- Conducted a master's thesis about optimizing the PID values for the servomotors attached in our printers, using IA.
- Improved the lifespan of specific servomotor by 4x, working faster then the previous project. Redesigned its movement.
- Migrated our software into a new electronic board.

Tools / Technical Environment:
- C++
- Arm Cortex A7
- Petalinux
- Yocto
- Servomotors
- Real-Time
- Embedded
- Linux
- Version Control: Git / SVN