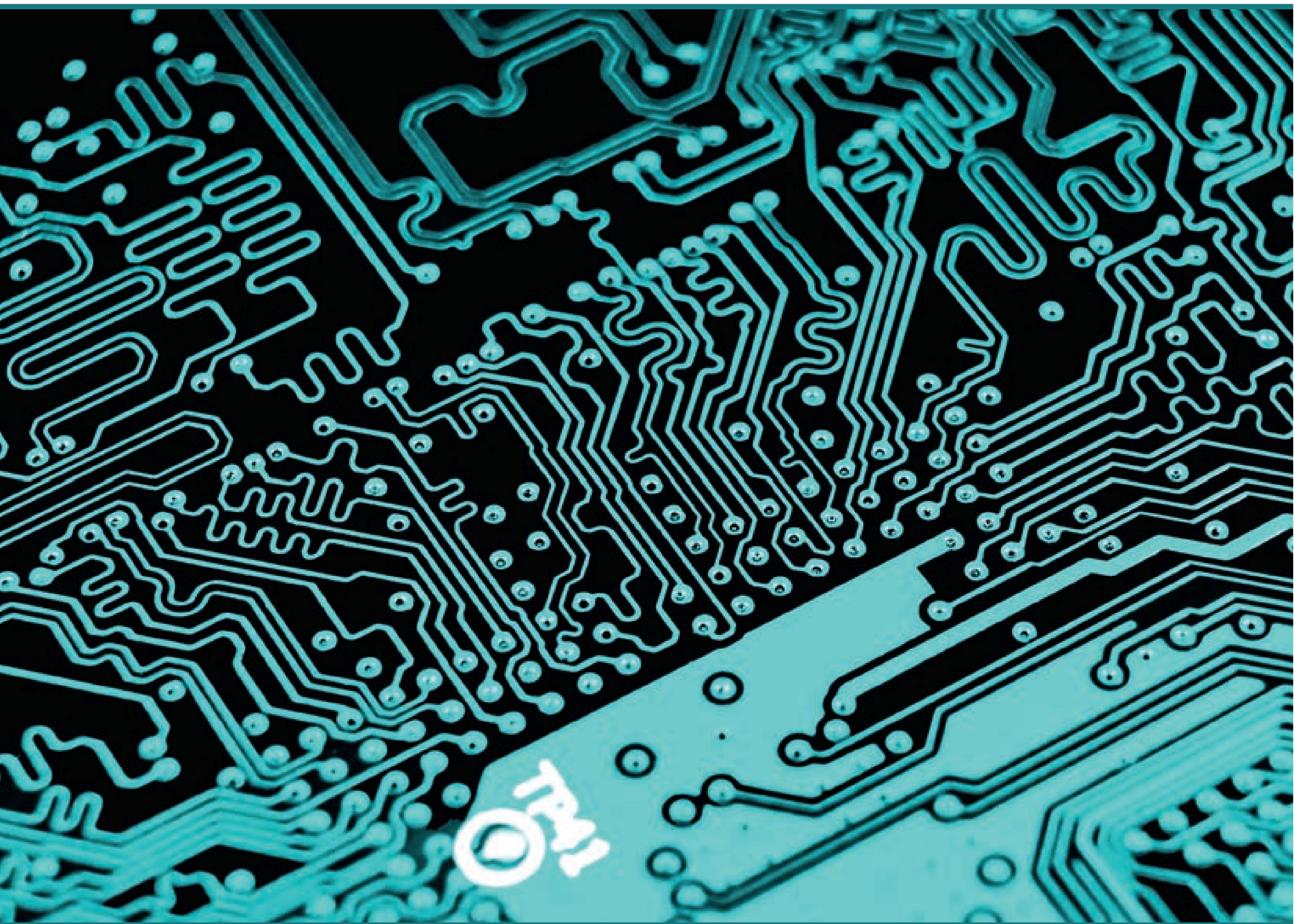
RIGA TECHNICAL UNIVERSITY

Donato Repole

RESEARCH OF PARALLEL COMPUTING NEURO-FUZZY NETWORKS FOR UNMANNED VEHICLES

Summary of the Doctoral Thesis



RTU Press
Riga 2021

RIGA TECHNICAL UNIVERSITY
Faculty of Electrical and Environmental Engineering
Institute of Industrial Electronics and Electrical Engineering

Donato REPOLE
Doctoral Student of the Study Programme
"Computerized Control of Electrical Technologies"

RESEARCH OF PARALLEL COMPUTING NEURO-FUZZY NETWORKS FOR UNMANNED VEHICLES

Summary of the Doctoral Thesis

Scientific supervisor
Assistant Professor Dr. sc. ing.
LESLIE ROBERT ADRIAN

RTU Press
Riga 2021

Repole D., Research of Parallel Computing Neuro-fuzzy Networks for Unmanned Vehicles. Summary of the Doctoral Thesis. – Riga: RTU Press, 2021. – 37 p.

Published in accordance with the decision of the Promotion Council “RTU P-14” of 30 August 2021, Minutes No. 04030-9.12.1/9.

https://doi.org/10.7250/9789934226922
ISBN 978-9934-22-692-2 (pdf)

DOCTORAL THESIS PROPOSED TO RIGA TECHNICAL UNIVERSITY FOR THE PROMOTION TO THE SCIENTIFIC DEGREE OF DOCTOR OF SCIENCE

To be granted the scientific degree of Doctor of Science (Ph. D.), the present Doctoral Thesis has been submitted for the defence at the open meeting of RTU Promotion Council on 28 December 2021 at 12.00 at the Faculty of Electrical and Environmental Engineering of Riga Technical University.

OFFICIAL REVIEWERS

Dr. habil. sc. ing. Peteris Apse-Apsitis
Riga Technical University

Dr. habil. sc. ing. Raja Mazuir Raja Ahsan Shah
Coventry University, UK

Dr. habil. sc. ing. Andrés Gabriel García
Universidad Tecnológica Nacional, Bahía Blanca, Argentina

DECLARATION OF ACADEMIC INTEGRITY

I hereby declare that the Doctoral Thesis submitted for the review to Riga Technical University for the promotion to the scientific degree of Doctor of Science (Ph. D.) is my own. I confirm that this Doctoral Thesis had not been submitted to any other university for the promotion to a scientific degree.

Donato Repole ……………………………. (signature)
Date: ………………………

The Doctoral Thesis has been written in English. It consists of 7 chapters, inclusive of the introduction; 111 figures, 19 tables; 88 equations; the total number of pages is 249. The Bibliography contains 66 titles.

Annotation

The presented Doctoral Thesis illustrates the author's research in the field of VHDL based "neuro-fuzzy controllers". The author's academic investigations involve numerous applications of "neuro-fuzzy controllers", and part of the doctoral research focuses on evaluating different implementation methods. The decision of VHDL as "controller's hardware description language" is the outcome of the author's academic research, which is the core of his international papers.

The presented work starts with an overview of autonomous mobile robotics applications, automotive applications and small autonomous unmanned aerial vehicles (derivative from RC planes), which describes the context of the Doctoral Thesis.

Then, the dissertation moves to the motivations behind the decision process of the selection of VHDL as "controller's hardware description language", strictly correlated to the flexibility and the advantages of using an FPGA instead of a multi-core MCU. A significant focus is given to the FPGAs parallel processing functionality. Part of the Thesis scrutinises methods to mitigate the complexity of a VHDL based description and the implementation of advanced learning processes.

The Thesis examines a novel software tool for the high-level "neuro-fuzzy controller" description capable of executing controller simulations, optimisation tasks, performing learning / training tasks, and exporting the controller in VHDL code.

The Thesis proposes an application case for the VHDL based "neuro-fuzzy controllers" research, aiming at the use of learning / training controller's capability to off-load the mechanical design. This approach targets the controller fine-tuning through a replicable process, which shall allow adapting the controller's parameters to the mechanical characteristics of the RC plane that shall be converted into a small unmanned aerial vehicle. A series of mechanical and electrical / electronic hardware assumptions and definitions are made as pre-requisites for the controller conception. The proposal's focus is the controller's design strategy, scrutinising the design process, the description and the simulation of the "neuro-fuzzy controller".

Since the system's pre-requisites and boundary conditions are finalised to deliver a general aerial vehicle controller, the Thesis aims to deliver a "neuro-fuzzy controller" capable of replicating a human being pilot behaviour. Efforts are made to establish fuzzy controller's simulation (fuzzy controller is the core of the "neuro-fuzzy controller" before the learning / training process and the optimisation process), a learning / training process, and an optimisation process.

A learning capable controller design may result in a very sophisticated design, and the designer shall rely on robust software tools; the selection of the learning / training acceleration tool becomes a crucial step of the dissertation application case. Even more important for the dissertation are the definitions of the "Simulation Conditions" on which the "core fuzzy controller" shall be tested. In fact, a mandatory condition for an appropriate learning / training process is to use a "core fuzzy controller", already capable of performing basic tasks, as the heart of the system.

What is drawn, between the lines, by the proposed work is the introduction of a design strategy that is looking for developing solutions for complex controller architecture of mobile robotic vehicles (of any nature) or even for multiple industrial application. This work enables further investigative research into autonomous robotics, particularly into the physical implementation of an autonomous aerial unmanned vehicle from an inexpensive RC plane.

A simplified RC plane design may be used as a worst-case scenario for the controller design, where a 3D printed homebuilt aircraft may be turned into an AUAV through the process and the algorithms disserted. Replication of the learning / training process and their iteration on different mechanics and different RC planes to be adapted into AUAV may result in information gold mining for the researchers. Indeed, the determination of reliable processes allows researchers to reutilise the same principles for totally different applications, circumscribed only by the researcher's imagination.

The Doctoral Thesis has been written in English. All summaries and conclusions and the research results are related to the hypothesis hypotheses and the relationship between them. Research outcome has the potential to evolve into other projects consisting of various methodologies extracted from the investigations.

The Thesis consists of 7 chapters, inclusive of the introduction and the subsequent conclusions. The bibliography contains 66 reference sources and 12 appendices. The volume of the present Doctoral Thesis is 249 pages. It has been illustrated with 111 figures, 88 formulas and 19 tables.

CONTENTS

SUBJECT INTRODUCTION

The autonomous vehicles applications (mobile robotics applications, driverless automotive applications, AUAV and UAV applications, etc.) define the context where the Doctoral Thesis is implanted. However, the Doctoral Thesis objective is the research and development of VHDL based "neuro-fuzzy controllers" for small autonomous unmanned aerial vehicles (derivative from RC planes).

For many years, fuzzy logic has been an attractive technology for the designers of industrial, consumer and automotive products. Conversely, achieving the right balance between cost and performance results is a difficult task. In fact, fuzzy algorithms can be executed on low-cost MCUs, but as these have architectures that were not designed to handle fuzzy logic, the software overhead often makes the performance inadequate. Dedicated fuzzy processor microchips can meet the most demanding performance requirements, but it is mainly an expensive ASIC solution. Indeed, only a few full custom or semi-custom integrated fuzzy controllers exist, and most of them are assembled from standard cells at the gate level. At the moment, an FPGA based solution is a valid option capable of delivering both good system performances and high flexibility to the designer. Important scientific literature proves the strength of the use of FPGAs for the implementation of neuro-fuzzy controllers. FPGA's parallel processing capability results in a significant advantage over the use of conventional MCUs, which are operating a serial data processing.

The selection of VHDL as "controller's hardware description language" is based on the VHDL efficiency and reliability for complex hardware, verified in numerous scientific articles, such as an AUAV controller.

Tasks of the Thesis

The primary task of the Thesis was to develop a VHDL based "neuro-fuzzy controller" proposal for small autonomous unmanned aerial vehicles (derivative from RC planes).

The secondary task includes the initial investigations of appropriate learning / training methods enabling the controller to adapt to different platforms with different flight dynamics models.

The learning / training process to be effective requires a primary fuzzy flight controller capable of performing a limited flying operation. From the listed functionalities, the author's goal is to emphasise raw simulation to check the "controller response" in certain conditions, such as: take-off, landing, route adjustments, steady-state flight (stable flight), gusty winds compensation manoeuvre, etc.

The assumption is to use the outcome of this simulation to implement an iteration based adjustment of the Rulebase's "weights" and "functions" (or weighted rules) to perform any learning processes.

Scientific Novelty

The project's primary focus is a small AUAV (it is assumed to be an autonomous RC plane) "neuro-fuzzy controller" capable of replicating a human being pilot behaviour. Efforts are concentrated on the design, the description and the simulation of the "neuro-fuzzy controller". The Doctoral Thesis introduces a design strategy capable of supporting complex controller's development for mobile robotic vehicles of any nature or even for multiple industrial application.

The dissertation examines a novel software tool for the high-level "neuro-fuzzy controller" description capable of executing controller's simulations, optimising tasks, performing learning / training tasks, and exporting the controller in VHDL code.

The resulting outcome is a flexible, innovative system capable of being adaptable to a different use through a training-based process for adjusting, weighting, and optimising the neuro-fuzzy control algorithm.

Practical Application

The Thesis proposes an application case for the VHDL based "neuro-fuzzy controllers" research, aiming the use of learning / training controller's capability to off-load the mechanical design. This approach targets the controller fine-tuning through a replicable process, which shall allow adapting the controller's parameters to the mechanical characteristics of the RC plane that might be converted into a small AUAV. A simplified 3D printed homebuilt RC plane is an extreme study case. Turning it into a basic AUAV, thanks to the use of core algorithms properly adapted and developed through a series of proposed optimisation and learning / training processes, may represent a sensational achievement.

Research results apply to a wide range of autonomous robotics, not only to the physical implementation of an AUAV from an inexpensive RC plane. The determination of reliable processes may allow reutilising the same principles for different kind of applications.

Tools and Methodology of Research

The Doctoral Thesis scrutinises methods to mitigate the complexity of a VHDL based on the description and the implementation of advanced learning processes.

A learning capable controller design may result in a very complex project, and the designer shall rely on powerful software tools. The selection of the learning / training acceleration tool becomes a crucial step of the dissertation application case. Even more important for the dissertation is the definitions of the "simulation conditions" on which the "core fuzzy controller" should be tested. In fact, a mandatory condition for an appropriate learning / training process is to use a "core fuzzy controller", already capable of performing basic tasks, as the heart of the system.

Many of the processes of theoretical calculations and graphical representation of the results have been obtained utilising a menagerie of software systems, including:

- Aforge.net (C# framework);
- ALDEC Active-HDL (VHDL compatible FPGA design creation and simulation environment);
- Altium Designer (hardware design environment);
- Cadence-OrCAD (hardware design environment);

- fuzzyTECH (Fuzzy/Neural GUI for fuzzy logic modelling and programming algorithms);
- Lattice Diamond (Lattice Semiconductors VHDL design environment);
- LT Spice (hardware design environment);
- MATLAB (multi-paradigm numerical computing environment);
- Maplesoft Maple (symbolic and numeric computing environment);
- Microsoft Excel (tables and spreadsheets);
- Microsoft Paint 3D (2D parts design);
- Microsoft PowerPoint (2D parts design);
- Microsoft Visio (2D parts design);
- Microsoft Word;
- Model-Sim (Mentor Graphics);
- Neural.NET (neural guided learning software);
- Pspice (Circuit modelling and analysis);
- Synopsys Synplify PRO (VHDL compatible FPGA synthesis software);
- Solidworks (3D parts design environment);
- XFL3 (Xfuzzy 3 GUI development environment for Neuro-Fuzzy system design, optimisation and simulations).

Structure and Volume of the Thesis

The Thesis objectives are related to the field of unmanned vehicles (mobile robotics applications, driverless automotive applications, AUAV and UAV applications), in particular to the research and development of VHDL based "neuro-fuzzy controllers" for small autonomous unmanned aerial vehicles (derivative from RC planes).

The Thesis presents a development project for the design of a VHDL based neuro-fuzzy controller. The premise, VHDL is the working platform for the system implies that the fuzzy system description must be synthesisable. In a synthesisable VHDL algorithm, there is a requirement to implement and tune the characteristics of the neuro-fuzzy controller to the physical hardware implementation (FPGA printing).

VHDL language imposes some limitations when compared with the flexibility and expressiveness of other fuzzy-logic oriented languages. To achieve a behavioural modelling, a VHDL description style is proposed in order to allow a distinct system structure description (fuzzy sets, rule base, etc.) and an operator description (connectives and fuzzy operations). Making it possible to describe both the fuzzy system structure and the processing algorithm independently [1]

A control system design advantage is pursued using high-level descriptions of neural fuzzy-logic based systems in a more remedial way. The consequent dissertation action is the analysis of the behavioural description achieved by "XFUZZY XFL version 3.5" (or XFL3 GUI) developed by "Instituto de Microelectrónica de Sevilla (IMSE-CNM). It allows high-level descriptions of the fuzzy-logic controller (or the neuro-fuzzy logic controller after the learning / training process) and then translates this description into a valid VHDL code.

Before generating the raw fuzzy controller's VHDL code and proceeding to the learning / training operations, the dissertation illustrated a series of raw simulations to check the "controller response" in certain conditions, such as: take-off, landing, route adjustments, steady-state flight (stable flight), gusty winds compensation manoeuvre and, etc

Then, a learning / training structured procedure is illustrated. The procedure targets the replication of the learning / training process in different uses, making it possible to adapt the controller's algorithm to many different AUAVs.

Chapter 1 is devoted to an overview or introduction to unmanned vehicles (mobile robotics applications, driverless automotive applications, AUAV and UAV applications, etc.), the topicality and associated problems. The author's assertions are discussed along with the methods of research and primary hypothesis. The scientific novelty, practical novelty and application are therein outlined.

Chapter 2 gives a brief outline of unmanned vehicles control strategies. In specific, it is taken as an example in the control strategy for a civil UAV. Three different control strategies to govern a UAV are illustrated:

a) linear control;
b) non-linear control;
c) AI, learning-based control.

Chapter 3 analyses the typical architectures of electric vehicles, particularly the safety regulations associated with the vehicle's REESS. Such research and investigations are precious because automotive industry standards and regulation are the benchmarks for the hardware proposal design illustrated in the Thesis.

Chapter 4 illustrates a detailed academic theoretical work based on selecting the most effective methods to implement a parallel computational capable, neuro-fuzzy controller. The investigations support the utilisation of the FPGA as the physical controller's hardware and the VHDL as the hardware description language. The theoretical benefits of the use of high-level descriptions of the fuzzy-logic controller (or the neuro-fuzzy logic controller after the learning/training process) and then the translation of this description into a valid VHDL code (in particular the behavioural description achieved by "XFUZZY XFL version 3.5" developed by "Instituto de Microelectrónica de Sevilla")

Chapter 5 introduces the proposed hardware used as a baseline for the flight controller design.

Chapter 6 is the final chapter of the Thesis, which illustrates the proposed neuro-fuzzy flight controller. The description includes:

- RTL views of the synthesisable system's VHDL code;
- full description of the "neuro-fuzzy" controller;
- an optimisation strategy for the "Controller";
- a controller's learning / training execution strategy;
- a complex simulation analysis of the raw fuzzy-logic flight controller.

With the presented research and the derivative neuro-fuzzy controller design, the author aims for a precise final delivery: to demonstrate the feasibility of a flexible-cost effective controller able to mimic a human pilot's driving behaviour and be capable of behaviours corrections with learning / training processes.

Dissemination of Research Results

The author's academic investigations touched on several different applications of "neuro-fuzzy controllers", and part of the research focused on evaluating different implementation methods. The decision of VHDL as "controller's hardware description language" is the outcome of the author's academic researches, which are the core of the author's international papers.

The following ten publications are presented in the Doctoral Thesis:

1. L. R. Adrian, **D. Repole,** and L. Ribickis, "Proposed neuro-guided learning for obstacle avoidance in AMBO a robotic device", 2015 56th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, 2015, pp. 1–5.
2. Janis Voitkans, Leslie R. Adrian, **Donato Repole**, "INVESTIGATION OF ELECTRICAL PARAMETERS FOR PCB TRANSFORMER" 15th International Scientific Conference: Engineering for Rural Development 25–27.05.2016 Jelgava, LATVIA, pp. 1445–1452.
3. L. R. Adrian, **D. Repole** and L. Ribickis, "High efficiency modular DC-DC power converter for adaption to industrial & hybrid robotics", 2016 57th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, 2016, pp. 1–5.
4. L. R. Adrian and **D. Repole**, "Intelligent autonomous environmental monitoring based on the AMBOA robot sensory system", 2017 IEEE 58th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, 2017, pp. 1–6.
5. **D. Repole** and L. R. Adrian, "Fuzzy nano piezo hybrid for fault detection in automotive power PCB", 2017 IEEE 37th International Conference on Electronics and Nanotechnology (ELNANO), Kiev, 2017, pp. 400–404.
6. **D. Repole** and L. R. Adrian, "Evaluation of GaN MOSFET for Unmanned Aerial Vehicles BLDC Motor Drive", 2018 IEEE 59th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 2018, pp. 1–4.
7. **D. Repole** and L. R. Adrian, "Introduction to Parallel MAS Control for MAS – Smart Sensor Networks", 2019 IEEE 60th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 2019, pp. 1–5.
8. L. R. Adrian, **D. Repole** and A. Rubenis, "Comparative study of Lithium-Ion hybrid super capacitors", 19th International Scientific Conference Engineering for Rural Development, 20–22.05.2020 Jelgava, LATVIA, pp. 906–912. DOI:10.22616/ERDev.2020.19.TF217.
9. **D. Repole** and L. R. Adrian, "VHDL based Neuro-Fuzzy Lithium Ion Hybrid Super Capacitors management, Advantages of the high level descriptions of neural fuzzy logic based systems", 2020 IEEE 61th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 5–6 Nov. 2020.

10. Krists Kviesis, Leslie Robert Adrian, Ansis Avotins, Olegs Tetervenoks and **D. Repole,** “MAS Concept for PIR Sensor-Based Lighting System Control Applications”, 8th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE'2020), Vilnius (Lithuania) 2021, accepted for publication (ID: PID014).

1. UNMANNED VEHICLES – INSIDE THE ENVELOPE

Nowadays, unmanned vehicles are getting more popular. Although during the last decades unmanned vehicles mainly had a military application, today, it is possible to observe unmanned vehicles in factories, streets, civil airfields and city parks. It is possible to divide unmanned vehicles into three groups:

a) unmanned ground vehicles;
b) unmanned aerial vehicles;
c) unmanned underwater vehicles.

In the previous decades, u**nmanned ground vehicles** (UGV) were primarily associated with robots, especially indoor robots used for carrying goods from deposit to factory working station. Over the last few years, an impressive technological acceleration brought to market many new UGVs for both military and civil application. The most illustrious military applications are:

a) hazardous object manipulations;
b) explorer;
c) delivery of supplies.

Civil applications for UGV are in line with the Thesis research, and it is possible to highlight a few fascinating applications:

a) industrial applications (autonomous robots or AR);
b) utility unmanned ground vehicles (UUGV);
c) human transportation.

Fig. 1.1. **E**xample of industrial ground vehicle.

Unmanned aerial vehicles (UAV) is the most famous category of unmanned vehicles, mainly due to UAV's impact on military combat strategies and techniques. The civil market is also observing a broad interest in UAVs for multiple purposes. Increased availability of cheap and durable batteries mixed with the low cost and high-performance available electronics for controls and power electronics made it possible to diffuse small-sized hobby UAVs (widely accessible in consumer electronics store). It is common to observe in our parks flying quad-copters or small model based aeroplanes controlled by a smartphone or just flying fully autonomously.

Unmanned underwater vehicles (UUV) may be divided into two categories: remotely operated underwater vehicles (ROUV) and autonomous underwater vehicles (AUV). Previously, UUVs have been used for a limited number of tasks dictated by the technology available. It is possible to highlight four main applications of UUVs.

- **Commercial –** UUVs are very popular in oil and gas industry for a large variety of uses.
- **Military –** the navies of multiple countries are currently producing UUVs to be used in oceanic warfare, with particular attention to the sea exploration to eradicate underwater mines threats or the sea exploration to detect unfriendly objects.
- **Research –** scientists rely on a heterogeneous variety of UUVs to study lakes, seas, and the ocean floor.
- **Hobby –** many robotic enthusiasts enjoy constructing and operating UUVs as a hobby.

2. UNMANNED VEHICLES CONTROL TECHNIQUES

Each kind of unmanned vehicles has a specific control strategy that allows the performing of particular tasks. In specific, the control strategy of a civil UAV it is taken as an example. Generally, three different control strategies to govern a UAV might be applied, which are:

a) linear control;
b) non-linear control;
c) AI – learning based control.

Basically, a linear control tries to approximate a complex non-linear problem, although described by a simplified pattern of equations, into a linear system that may be reduced as a classical control feedback loop. Due to the complexity of the dynamics, there are two basic strategies for the control design.

The first method is to continue the decomposition in the previous section to identify components of the dynamics that are well controlled by specific choices of the actuators and then perform successive loop closure [2].

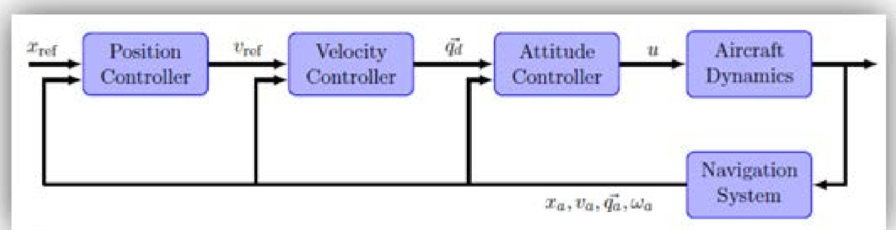


Fig. 2.1. Successive loop-closure control architecture [2].

The second approach is to design a controller for full dynamics, either linear or non-linear. The advantage of this approach is that it employs state space control approaches to handle the fully coupled dynamics. However, it is challenging to handle the actuator's saturation and very hard to include state constraints. Furthermore, unless done with extreme care, these controllers, especially in high-performance flight, can be very sensitive to modelling errors and omissions [2].

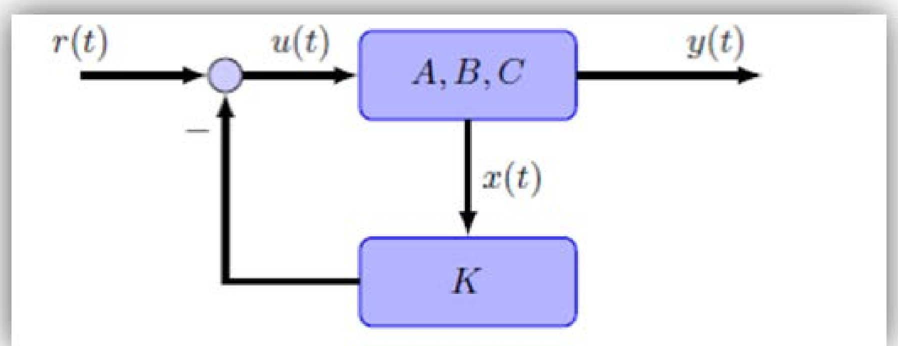


Fig. 2.2. Full-state feedback controller [2].

The scientific literature shows that for LTI systems, the controller's design may result in a simple task, but in front of a non-linear system, some other more complex techniques should be considered, with particular attention to the system stability. The most common non-linear control techniques are:

- linearization (approximation);
- feedback linearization;
- Lyapunov Stability;
- CPWL (continuous piecewise linear approximation).

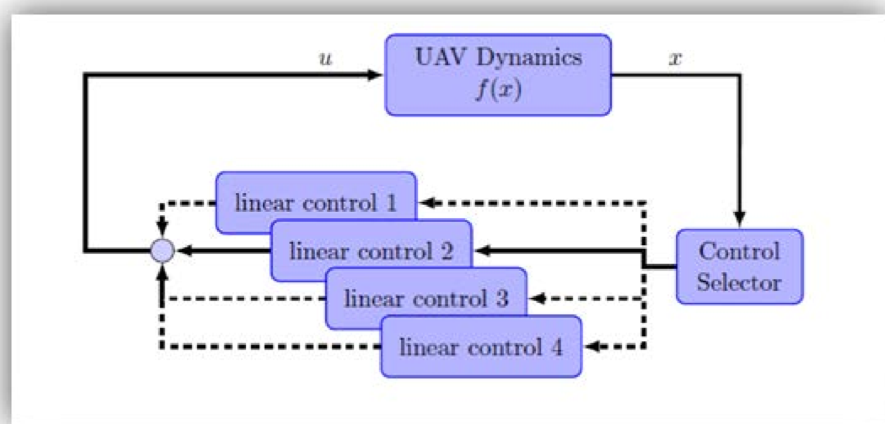


Fig. 2.3. Schematic of a gain scheduled scheme for UAV control [3].

Learning-based control (LBC) is an alternative approach to control an unmanned vehicle and could be based on a hybrid neuro-fuzzy network tuned by a genetic algorithm.

Practically the system uses the system's available parameters, or digitally processed differential parameters, as inputs of specific fuzzy membership function.

The fuzzy system (MIFs, MOFs, FIS, etc.) could be processed in a dedicated neuro-fuzzy network. A successive interaction of neuro-fuzzy modules and genetic algorithms produces a neuro-fuzzy controller tuned by a genetic algorithm (training process). The training system produces a more elaborated and accurate fuzzy inference system (FIS).

Usually, a hybrid neuro-fuzzy controller uses a combination of several layers, and only two of them will be completely readable because inner layers, as usual for a neuro-fuzzy system, are the neuro-fuzzy hidden layers.

The first layer contains all "membership input functions" (MIFs), which for a UAV controller could be divided into:

- flight dynamics membership functions;
- trajectory membership functions;
- energy estimation membership functions.

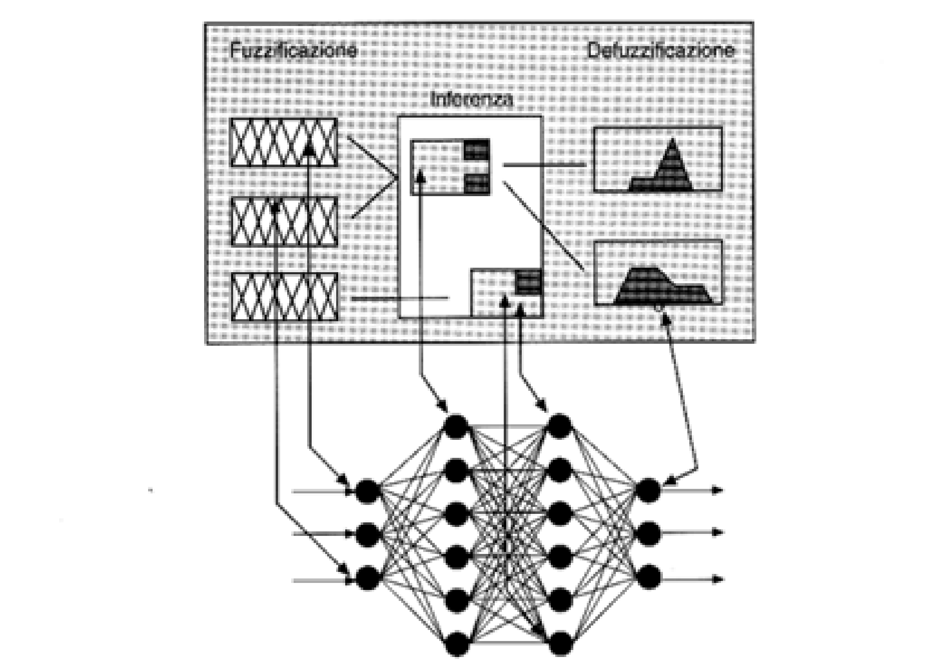


Fig. 2.4. Neural network blocks scheme representation.

Each MIF activates a specific group of neurones, and each membership function activates the status neurone; neuro-fuzzy training (achieved into hidden layers) defines the features and weights of the second layer neurones network.

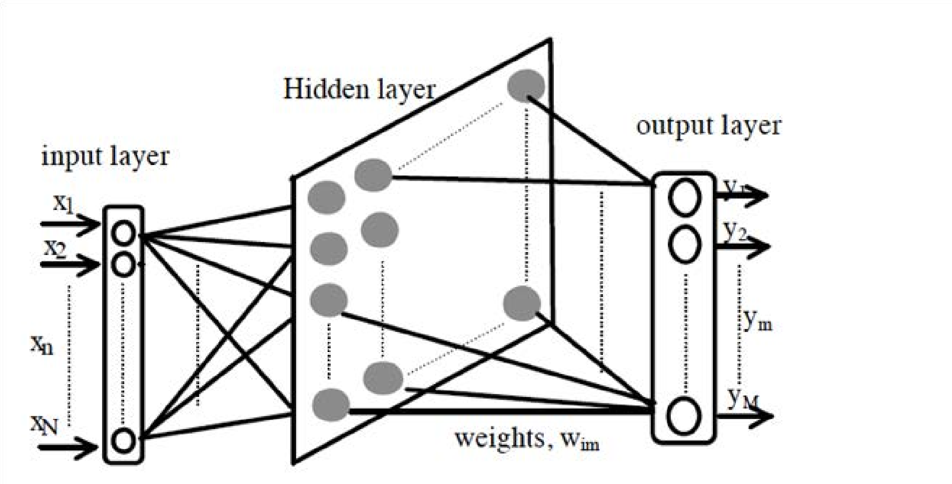


Fig. 2.5. Hybrid neuro-fuzzy network: hidden layer representation.

This second network activates the output layer associated with its actuators and its propulsion hardware in the case of a UAV or a robot.

3. TYPICAL ARCHITECTURES OF ELECTRIC VEHICLES

Autonomous unmanned vehicles (AUV) development and evolution is a process rigorously connected to the vehicles electrification process. Part of the author's research also analyses this process, using the automotive applications as a significant study case.

Electrification of the automotive market is in constant growth, and it is forecasted that the incoming regulations will strengthen the trend. In fact, across the globe, numerous cities and countries are opting for severe diesel combustion engines limitation in the urban areas and, in a few cases, some substantial limitations also for petrol combustion engines. The e-mobility process will affect the whole urban transport (private and public), which means that in the function of the vehicle's application, a specific architecture will be used to satisfy the new regulations at the lowest production cost. In the merit of the EV/HEV characterisation, the most relevant regulation to highlight is the "UN/ECE-R100". This regulation defines the main requirements in terms of electrical safety and makes a clear architectural distinction in the function of the vehicle's REESS (rechargeable energy storage system) [4].

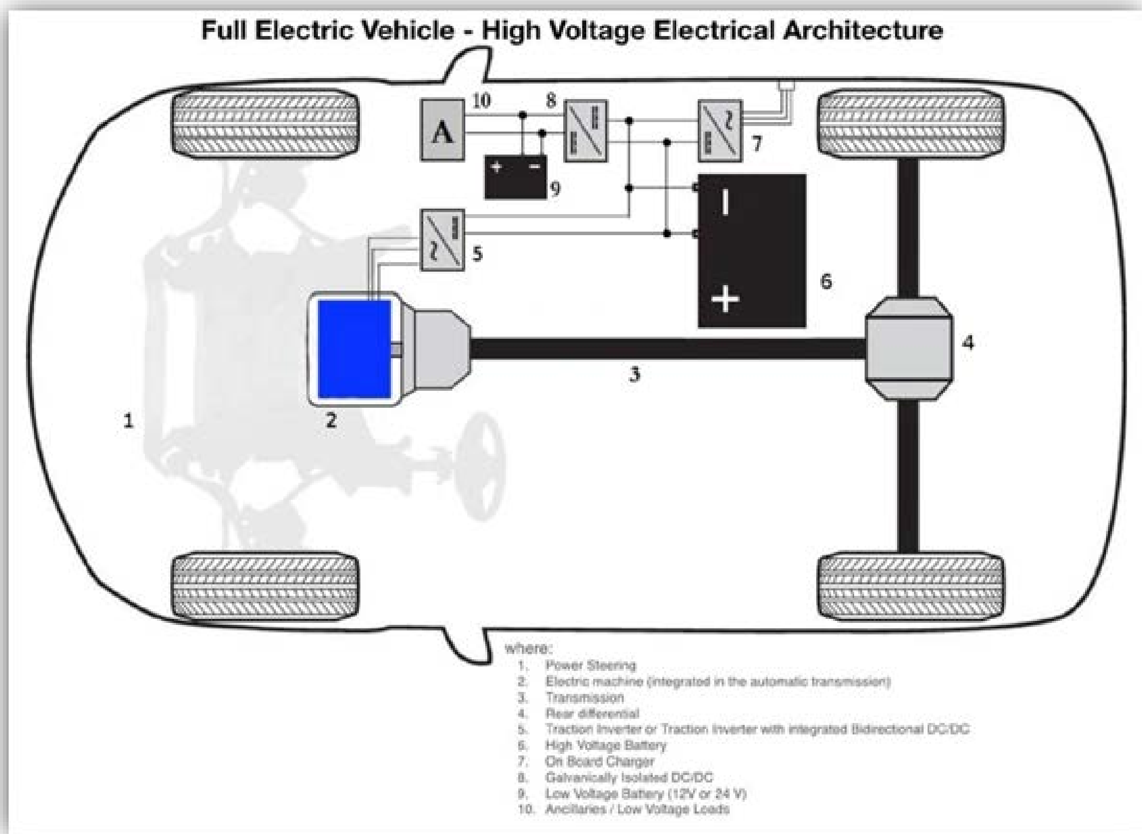


Fig. 3.1. Example of EV high voltage architecture representation.

4. THEORETICAL FRAMEWORK

A learning capable AUAV controller design is the core of the Thesis project; the author's preferences lie in the field of fuzzy logic and neural networks.

Fuzzy algorithms can be executed on low-cost conventional microcontrollers, but as these have architectures that were not designed to handle fuzzy logic, the software overhead often makes the performance inadequate. Dedicated fuzzy processor chips can meet the most demanding performance needs [5].

The dissertation starts with the scrutiny of a high-level design approach. The usage of high-level description methodologies for modelling fuzzy controllers reduce the development time significantly, making a rapid design of custom fuzzy hardware possible. VHDL for design capture and VHDL-based logic synthesis proved to be an efficient method for designing complex hardware.

However, for describing regular structures like finite state-machines, a different approach could be more appropriate. For describing such structures, state-charts could be used. In addition, a commercial tool based on state-charts incorporates a VHDL generation facility for generating a synthesisable code. The employment of state-charts formalism for capturing a rule base of a fuzzy control system is a comprehensive used approach in the scientific literature.

Fuzzy controller relies upon conventional principles for the interface and the information exchange. In the controller, an external device's information (such as a sensor) is converted into an output control signal to drive a device (or multiple devices) via the process of fuzzification, rule evaluation and defuzzification. Such kind of processes are based on a set of membership functions and FIS; numerous publications, such as [5], [6], [7], [8], and [9], illustrate the processes' details. The motivation of implementing a fuzzy controller in VHDL was driven by the need for an inexpensive hardware implementation of a generic fuzzy controller for use in industrial and commercial applications. This approach has several more advantages. Field programmable gate array (FPGA) is used as a hardware platform because FPGA allows very high logic capacity [10], [11]. FPGAs offer more flexibility than ASICs because the chip can be reprogrammed, allowing the redesign of the system's circuits portions for optimisation [12]. With the use of small, cost-effective FPGA for implementing the fuzzy logic controller, it is possible to fully benefit from the parallel computational capabilities of the fuzzy logic (and neural networks) [13].

After the parallel computational process, a significant advantage of using an FPGA is the possibility of having interchangeable blocks-based software, where the objects associated with each rulebase represent an independent VHDL component. For each VHDL Component, if a standard porting layout is used, it could be possible to tune the controller within the main state machine algorithms performing only two operations:

- adapt the algorithm's objects (adapt or replace at the component level the rules, the membership functions, etc.);
- define the new weights of each rulebase (algorithms can be trained to achieve a perfectly tuned system) [13].

This solution enables a dynamic FIS tuning via the cloud or a dedicated learning process typical of the neural network design. Enabling this functionality, it is possible then to influence the behaviour of a specific device, adapting the controller's behaviour to particular tasks or environmental conditions.

One primary driver of the fuzzy systems, neural networks, and neuro-fuzzy systems success is their ability to approximate continuous non-linear functions. In this area within the fuzzy and neuro-fuzzy systems, a large number of works are cited to support the utilisation of FPGA and VHDL for the implementation of complex non-linear neuro-fuzzy controllers.

Particular attention is given to the study of novel description formats to describe both the fuzzy system structure and the processing algorithm independently. Such kind of description formats makes the use of linguistic hedges possible to compact the rules defining the system's behaviour. In these circumstances the use of integrated software capable of translating a fuzzy-logic oriented language into a VHDL code results beneficial for the design process [14], [15].

The evaluation of the "XFUZZY XFL version 3.5" GUI developed by "Instituto de Microelectrónica de Sevilla (IMSE-CNM)" [15] results are beneficial for the description of the fuzzy logic controller (or the neuro-fuzzy logic controller after the learning / training process) and then for the implementation of a valid VHDL code.

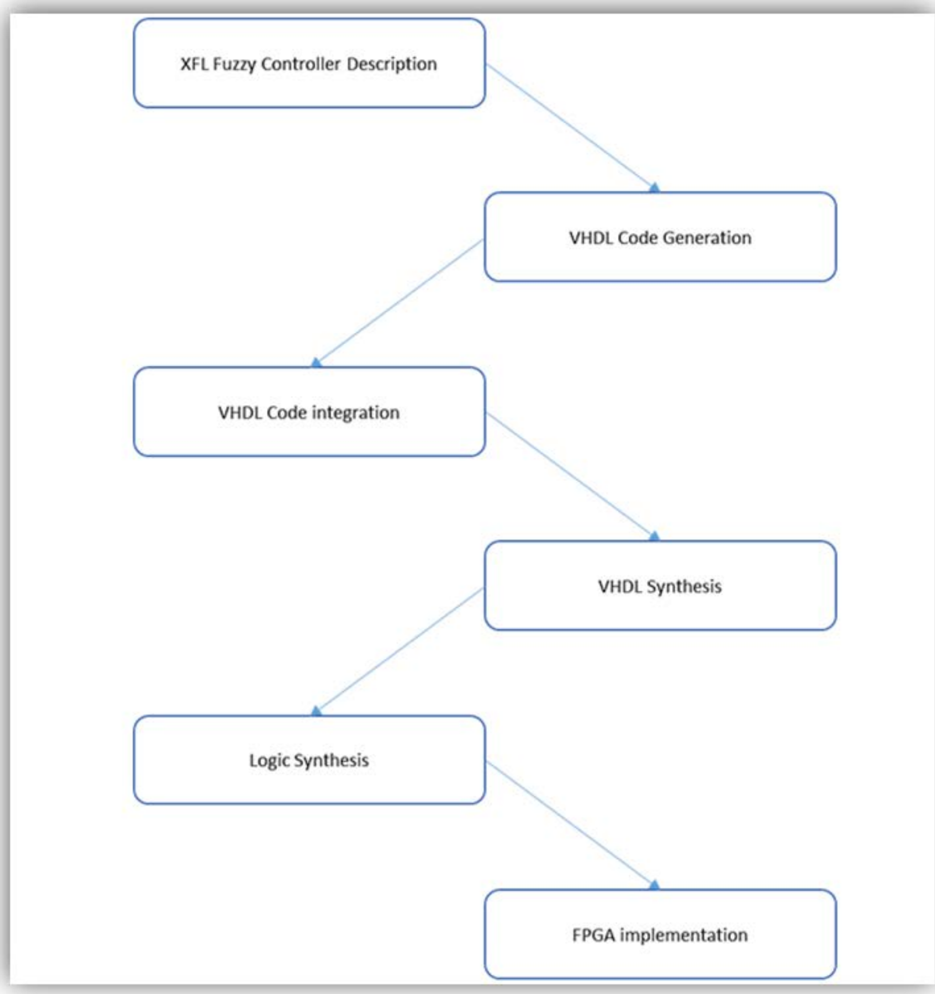


Fig. 4.1. Proposed design flow.

5. SYSTEM'S HARDWARE DESIGN PROPOSAL

As previously introduced, it is assumed to use a simplified RC plane mechanical design as a worst-case scenario for the controller design, where a 3D printed homebuilt aircraft may be turned into AUAV through the process and the algorithms disserted. A twin-motor fly-wing platform is used as a benchmark for describing the controller's design .

For the proposed work, the electronic hardware design proposal has been developed in order to perform few functions, which are:

- to define the core electronics / hardware required by the autonomous flight mode controller;
- to define the hardware for the neuro-fuzzy controller;
- to define the hardware for the learning / training process.

Figures 5.1 and 5.2 describe the block diagrams of the employed electronic hardware; the system's core unit is the FPGA, which acts before as the system's gateway (collects and digital processes all peripherals information) and then acts as the system's controller performing a parallel computation of the collected information.

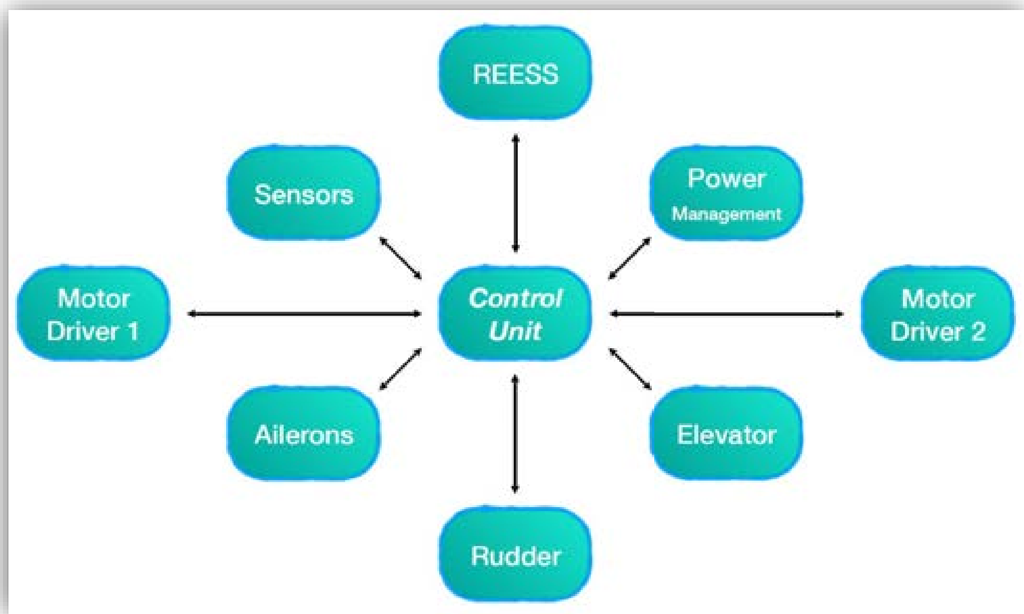


Fig. 5.1. HW high-level block diagram.

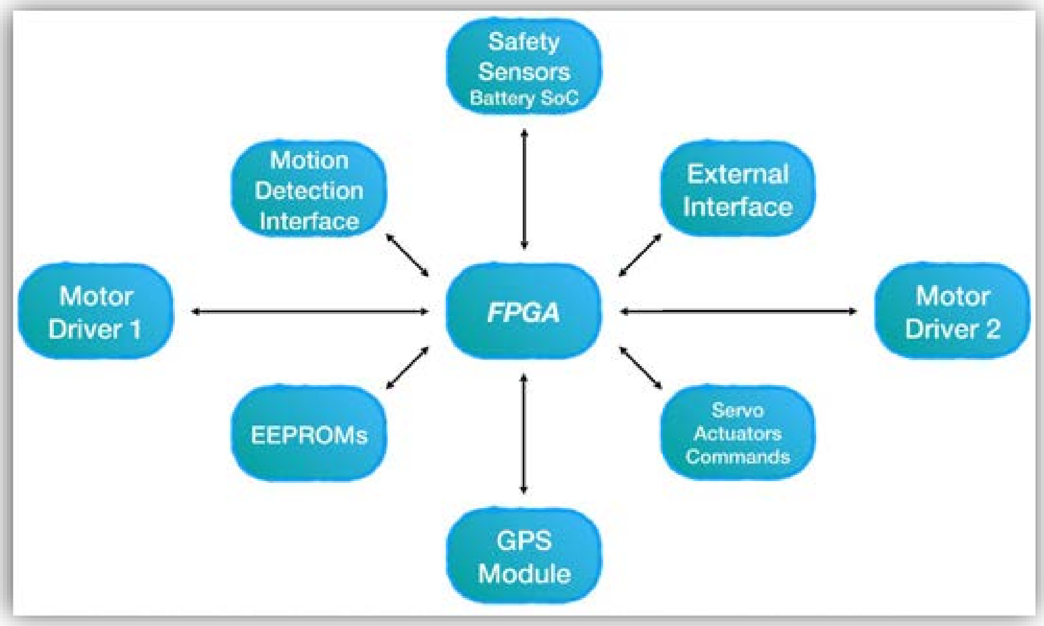


Fig. 5.2. FPGA's peripherals block diagram.

6. CASE STUDY: CONTROLLER'S DESIGN

As previously described, the scope of the proposed work is to minimise the hardware design giving the most of the design load to the VHDL-neuro fuzzy controller. This design strategy moves by the minimisation of the sensor installed on the vehicle, seeking a cost-effective compromise between mechanical constraints, electronics hardware / sensors available, and flight control principles.

As previously described, the first step of the controller's design is the identification of the "System's Inputs", as "System's Environment Variable" or as "Shell Variable", the "Actuators" or "System's Outputs" and then link them with a "Transfer Function". As proposed work is oriented on a neuro-fuzzy controller, the "System Transfer Function" is the set of all MIFs, all MOFs and all rulebases (or the hedge block) [16].

The consequent action targets the definition of a set of bespoke VHDL components; each component associable with a specific system's peripheral, acting as a system's peripherals interface. It is assumed that the controller's core requires the following input parameters:

a) altitude,
b) speed,
c) pitch angle,
d) rolling angle,
e) yaw angle,
f) estimated position,
g) flight reference parameters,
h) proximity sensor,
i) battery's SoC.

Then, the consequential system's design assumption is that the neuro-fuzzy controller generates the following output signals:

a) left E-Motor torque demand;
b) right E-Motor torque demand;
c) ailerons SERVO-Motor, PWM control signal;
d) elevator SERVO-Motor, PWM control signal;
e) rudders SERVO-Motor, PWM control signal.

The core of the flight controller development is associated with the FIS design, which is achieved by describing a set of five "Rulebases" within the XFL3 environment.

XFUZZY GUI encases a set of membership functions into a "Type", usually associated with a sensor or an actuator. GUI's definition of the controller physical "Input Variables" and "Outputs Variables" requires an existing environment "Type" to be linked with it.

The **creation of a fuzzy system in the "XFUZZY" (or "XFL3 GUI") environment usually starts with the definition of the "Operator Set".** An "Operator Set" in "XFL3 GUI" is an object containing the mathematical functions assigned to each fuzzy operator. Fuzzy operators can be binary (like the T-norms and S-norms employed to represent linguistic variable connections, implication, or rule aggregations), unary (like the C-norms or the operators related with linguistic hedges), or can be associated with defuzzification methods' [15].

In the description of a fuzzy system, the second step is the creation of the linguistic variable types, using the "Type Creation Interface". A new "Type" requires an identifier and universe of discourse (minimum, maximum and cardinality). The interface includes several pre-defined types corresponding to the most usual partitions of the universes. These pre-defined types contain homogeneous triangular, trapezoidal, bell-shaped and singleton partitions, shouldered-triangular and shouldered-bell partitions. Other pre-defined types are equal bells and singletons, commonly used as a first option for output variable types. When one of the previous pre-defined types is selected, the number of membership function of the partition must be introduced [15].

An XFL3 type is an object that describes a type of linguistic variable. It defines its universe of discourse, names the linguistic labels covering that universe, and specifies the membership function associated with each label. Linguistic labels can be defined in two ways: free membership functions or members of a family of membership functions. In the last case, the family of membership functions must be previously defined. A free membership function uses its own set of parameters, while the members of a family share the list of parameters of that family. It is useful for reducing the number of parameters and representing constraints between the linguistic labels (such as the order or a fixed overlapping degree) [15].

The types so defined inherit the universe of discourse and the labels of their parents automatically. The labels defined in the body of the type are either added to the parent labels or overwritten if they have the same name.

The third step in defining a fuzzy system is to describe each “Rulebase” expressing the relationship among the system variables.

The accurate definition of the “Operator Sets”, “Variable Types”, and “Rulebases” is propaedeutic for the fuzzy system design progress. The definition of the global input and output variables, using the “Variable Properties” interface (GUI’s window), is the following design step.

The concluding operation in a fuzzy system definition is the description of its hierarchical structure.

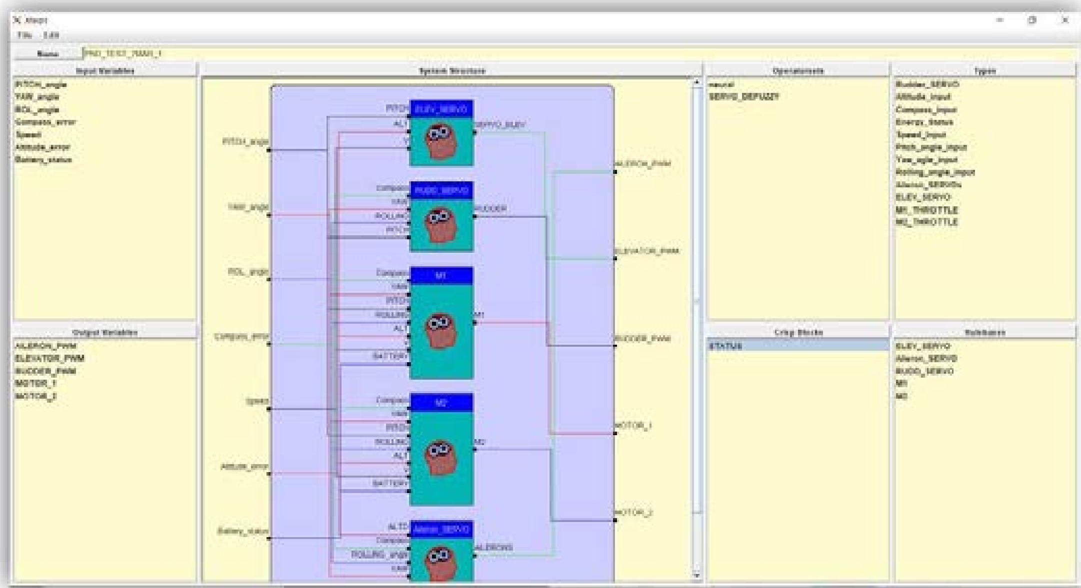


Fig. 6.1. Flight controller’s description in XFL3 GUI.

Before exporting the fuzzy flight controller in the VHDL language, it is recommended to implement a series of simulation and, if possible, optimise the fuzzy flight controller. There are several options available able to perform a fuzzy controller simulation. The author appreciates the functionalities available on the XFUZZY environment and in its GUI. The GUI's functionalities used for the proposed work are:

- controller graphical representation;
- optimization of the type's membership functions;
- optimization of rulebase;
- monitorization of fuzzy controller's behaviour.

From the listed functionalities, the author decided to emphasise the utilisation of the "Inference Monitor" functionality to perform a raw simulation of the controller. The goal of the raw simulation is to check the controller response in certain conditions, such as: take-off, landing, route adjustments, steady-state flight (stable flight), gusty winds compensation manoeuvre, etc.

The assumption is to use the outcome of this simulation to implement an iteration-based adjustment of the rulebase's "weights" and "functions" before exporting the VDLH algorithms or perform any learning processes.

The simulation process delivers the possibility to test the VHDL controller behaviour, which shall be encased in a complex, and broader physical controller. It has a multi-layer structure, associable to any hierarchical hardware schematics design. The "Top Layer" describes the peripherals interfaces and the interaction between the peripherals with the core of the flight controller. A typical VHDL schematics RTL view, generated after the algorism's synthesis, allows the user to observe the whole system as an array of blocks connected to each other with data bus and independent digital signals.

The VHDL code design incorporates bespoke functionalities to perform data mining operation (vehicle's telemetry), intended to operate in both circumstances: in case the system is working in "autonomous mode" or in "training mode".

The last design iteration is associated with the execution of the learning / training process. The dissertation describes a proposal, which utilises the "Learning Tool" of the XFUZZY GUI. A preliminary condition is the utilisation of a training file compatible with the "Learning Tool". It requires that the raw data stored in the external memory shall be transferred on the user PC and then processed in order to ensure the conformation of the input / output information pattern with the "Learning Tool" standards (as well as shall be given to the file the extension ".trn").

Configuration starts with the selection of the learning technique. The author justifies its predilection for supervised learning techniques because it focuses on the system behaviour given by a set of training input / output data.

The successive configuration step is the selection of the learning algorithm. The author's decision is to opt for gradient descent algorithms, which are well-known algorithms employed in fuzzy systems learning processes. Although the most common variant is the back propagation algorithm, it is preferred to use the Manhattan algorithm. The root of the decision is in the principles behind the algorithms. Back propagation algorithm modifies the parameter values proportionally to the gradient of the error function in order to reach a local minimum. Since the convergence speed of this algorithm is slow, modifications such as a different learning rate for each parameter or adapting, heuristically, the control variables of the algorithm are beneficial. RProp algorithm follows this strategy and improves significantly the convergence speed taking into account the gradient value of two successive iterations providing information about the curvature of the error function. Manhattan algorithm is chosen because it represents a good trade-off between the algorithm's accuracy and computational power required.

The subsequent configuration step is the selection of the error function. The author assumes the default function, mean square error function, as an appropriate option.

Following configuration tool selection is the identification of the best processing algorithm to use for the designed fuzzy system simplification, since the processing algorithms can be applied to the system before the tuning process (pre-processing option) or after it (post-processing option).

The author's choice is to use the post-processing option; to complete the process, it is necessary to select the pruning algorithm that prunes the rules and reduces the membership functions. Author's preference is the method of pruning the worst "N" rules.

The last configuration step is the definition of the end condition; it is mandatory to specify how the learning process shall be concluded. This condition limits the number of iterations, the maximum error goal, or the maximum absolute or relative deviation (considering both the training and the test error).

Author's preliminary "end condition" setup imposes a limit of 25 iterations. Further author's research may focus on the enhancements quantification of more complex and application customised application setups. Not mandatory tool configurations are not considered in this dissertation, although they may potentially be used in further investigation.

7. CONCLUSIONS

The dissertation's introduction starts with a quick overview of unmanned vehicles, currently available technologies, and future technological evolutions. The introduction concludes with the analysis of the author's investigations that are exposed in:

- dissertation's topicality;
- dissertation's primary hypothesis and intentions;
- methods of research and development;
- dissertation's scientific novelty;
- dissertation's practical application of research results;
- dissemination of research results.

The second part of the work builds up a theoretical framework for the application proposal associated with the Thesis. This section clearly defines the goals of the work and draws the strategy to achieve them. Significant attention is given to the academic research analysis of applications and implementations of fuzzy logic controllers and neuro-fuzzy controllers. Efforts aim to construct the fundaments on which to build the controller's design strategy proposal, assuming that

- the main goal of the proposed work is to implement a controller capable of replaying the behaviour of a human pilot while driving a small RC aeroplane;
- the secondary aim of the proposed work is to create a controller capable of being tuned in the function of the hardware (the small physical UAV, or RC aeroplane capable of autonomous operations);

Overall, it may be said that the preliminary study specifies:

- a neuro-fuzzy controller is the control strategy aimed for the small UAV;
- the assumptions made is that the neuro-fuzzy controller has seven inputs and five outputs;
- a detailed study of the academic literature suggests the use of an FPGA to process the controller;
- the choice of FPGA is driven by its flexibility and by its capability to process multiple functions in parallel (parallel computation capability);
- VHDL will be used as the working platform for the systems, although the VHDL language imposes some limitations if compared with the flexibility and expressiveness of other fuzzy logic oriented languages;
- in order to achieve behavioural modelling, the Author's suggestion is to use a VHDL description style where the system's structure description (fuzzy sets, rule base) and the operator's description (connectivity, fuzzy operations) are defined separately (this makes it possible to describe both the fuzzy system structure and the processing algorithm independently);
- the description format makes possible the use of linguistic hedges in order to compact the rules defining the system behaviour (a significant advantage of using this approach is the availability of a tool able to translate a fuzzy logic oriented language with a GUI interface into a VHDL code);

- the proposed work, to describe the fuzzy logic controller and then translate this description into a valid VHDL code, utilises the "XFUZZY XFL3.5" GUI (or XFL3) developed by Instituto de Microelectrónica de Sevilla (IMSE-CNM);
- XFL3 description language is a development environment that eases the specification, verification and synthesis of fuzzy inference systems;
- a set of essential functions, called the XFL library, performs the parsing and semantic analysis of XFL specifications and stores them using an abstract syntax tree.

The third part of the Thesis starts from the hardware description of a small RC plane transformable into a small AUAV. The hardware description covers the electronics hardware description and marginally the mechanical hardware description. The mechanical description includes the description of a simple RC plane powered by a low voltage REESS (according to Reg. 100 definition of low voltage REESS) and a set of two independent low voltage BLCD e-Motors. The description of the electronic systems targets a comprehensive definition of the hardware characteristics, indispensable for a proper controller`s definition. The proposal's primary goal, which targets to move the design's load from the mechanical design to the controller's design, is the use of the neuro-fuzzy controller to adapt itself to the vehicle's characteristics, allowing the simplified mechanical design of the drone (or RC plane).

The core of the research work and the technical proposal is the controller's design based on a multi-layer structure. The design starts with the identification of the System's Inputs", the "Actuators", or "System Outputs", and then links them with a "Transfer Function". Since the proposed work is oriented on a neuro-fuzzy controller, the "System Transfer Function" is implemented by a specific set of MIFs, MOFs, FIS (Rulebases) and a learning process from a training file.

The VHDL controller's multi-layer structure is associable with any hierarchical hardware schematics design. The "Top Layer" describes the peripherals interfaces and the interaction between the peripherals with the core of the flight controller.

The first VHDL "Top Layer" section is populated with a set of independent blocks specifically designed to process the "System's Inputs". Each block, independently (in parallel), manages a determined sensor and then digitally processes the relative information before broadcasting the data to the neuro-fuzzy controller (the core of the controller).

The second section of the VHDL "Top Layer" algorithm incorporates the VHDL algorithms exported from the XFUZZY GUI. The neuro-fuzzy flight controller is encased in a bespoke VHDL component called NEURAL, and its VHDL code represents the second hierarchical layer of the algorithm's structure. The "NEURAL" VHDL component utilises a set of five sub-components, each of them built on a specific "Rulebase". For hierarchical systems, a VHDL description is generated for each "Rulebase", which acts independently and populates the linked sub-component. The VHDL code population of these sub-components embodies the third hierarchical layer of the physical controller structure.

The third VHDL "Top Layer" algorithm's section is populated with a set of independent components specifically designed to process the "System's Outputs". Each VHDL component is associated with a single neuro-fuzzy controller's output. Each block, independently (in parallel), is interfaced with the neuro-fuzzy controller and digitally processes the relative information before broadcasting to the electro-mechanical actuators the control signals.

The primary focus of the controller design is the neuro-fuzzy unit. For this design, the XFUZZY GUI results are exceptionally effective for the controller description, design, simulation, optimisation and the learning / training process. The technical proposal's pre-requisites are:

- By assumption, a significant hardware design simplification (mechanical and electronic) is pursued.
- The final goal is to compensate the hardware simplification with a controller capable of being easily tuned and capable of learning.
- The mechanical hardware architecture is defined as a baseline for the controller design.
- The electronic hardware architecture is defined as a baseline for the controller design.
- Chapter 6 investigations deliver the following significant results:
    - the RTL views of the synthesisable system's VHDL code;
    - a full description of the neuro-fuzzy controller;
    - an optimisation strategy for the "Controller";
    - a controller's learning / training execution strategy;
    - a detailed simulation analysis of the raw, fuzzy flight controller.

With the presented research and the derivative neuro-fuzzy controller's design, the āuthor aims for a precise final delivery: to demonstrate the feasibility of a flexible and cost-effective controller able to mimic the driving behaviour of a human pilot and also be capable of behaviour corrections with learning / training processes.

The author relies on the simulation analysis presented in Chapter 6 to demonstrate the controller's basic behaviours, flexibility, robustness, and potential future developments. Although the simulation analysis covers a wide range of cases, a particular emphasis is given to the controller behaviours during complex manoeuvres such as the take-off and the landing.

For each analysed manoeuvre, the analysis describes a controller that takes time by time the expected decision, the expected decision that a human being pilot may most likely take if facing similar environmental conditions. Remarkably, a controller not yet optimised and not yet trained can produce excellent results. Although the outcome is in line with the theoretical research done, it may be expectable that a series of learning / training operations may be necessary before establishing a system capable of performing a fully autonomous flight.

An analysis and conclusions of the proposed control strategy quality:

The appraisal of the proposed controller's quality in the field of AUAV constructed on an adapted small RC plane is both varied and subjective. Many claims are made regarding the controller's quality to perform individual manoeuvres outside sets of pre-defined environmental conditions. The learning / training process limitation is that in front of unpredictable conditions, a learning / training process, by definition, is not applicable (by definition, it is not possible to define training for unpredictable conditions). It means that the author cannot rely entirely on the learning process' contribution, although it is an important strategic asset.

So far, as relates to the controller ability to perform manoeuvres under pre-defined environmental conditions where the landscape is known, where the vehicle position in the space *is under control, and the target manoeuvre is known (it is intended a manoeuvre that a human being is capable of defining and then mimicking), it is possible to expect that the controller will, with a reasonably good margin of error, react accordingly.*

The controller efficiencies may be affected by the accuracy / reliability of the particular type of sensors, the quality of the information associate with the system's environmental variables (or global environmental information). The learning / training process quality targets to mitigate such risk. With future research it is expected that a characterisation of this mitigation factor is given.

8. PLANNED FUTURE RESEARCH

The description made gives all the information necessary to progress with the project and obtain the necessary economic resources required for the high development cost of such systems. The sourcing of all components required for the complete system's physical realisation is not the only cause of the delay, ensuring the proper testing environment and facilities allocation results being significant braking elements for physical tests implementation. Physical installation of the system in a real-world environment is currently underway and is a result of the Thesis work.

Future research, relying on the availability of a "physical vehicle" and the results of a learning / training process described in Chapter 6, will include the addition of a refined neural network to the controller. Supplementary plans are to adapt different models of RC planes to carry the electronic hardware described in Chapter 5 and then verify the controller's flexibility and its capability to adapt to new mechanical characteristics using the learning / training process.

Future research will include but not be limited to:

- industrial applications of the neuro-fuzzy controller built on FPGA;
- cloud-based neural networks for industrial applications;
- cloud-based neural networks for environment safety-critical monitoring;
- automotive application of neuro-fuzzy controller built on FPGA;
- FPGA automotive applications;
- automotive applications for artificial intelligence;
- self-driving system for automotive applications;
- data collection for pseudo-memory applications;
- practical applications for swarm robotics manipulation through memory harvesting;
- long-range exploration technologies for fully autonomous vehicles;
- safety modelling for closed environment robotics;
- investigation into appropriate control methods for data access, including MOB, cloud or other access methods for single robots, swarm robots or remote exploration robots;
- full design of worker or swarm accessories to compliment the system.

BIBLIOGRAPHY

[1] A. Barriga, S. Sánchez-Solano, P. Brox, A. Cabrera, I. Baturone, “Modelling and implementation of fuzzy systems based on VHDL”, International Journal of Approximate Reasoning, Volume 41, Issue 2, 2006, pp. 164–178, ISSN 0888-613X, https://doi.org/10.1016/j.ijar.2005.06.018.

[2] How J. P., Frazzoli E., Chowdhary G. V. (2015) Linear Flight Control Techniques for Unmanned Aerial Vehicles. In: Valavanis K., Vachtsevanos G. (eds) Handbook of Unmanned Aerial Vehicles. Springer, Dordrecht.

[3] Girish C. V., Emilio F., Jonathan H.P., Hugh L. (2015) Nonlinear Flight Control Techniques for Unmanned Aerial Vehicles. In: Valavanis K., Vachtsevanos G. (eds) Handbook of Unmanned Aerial Vehicles. Springer, Dordrecht.

[4] UN/ECE-R100/Rev.2, Regulation No. 100, Revision 2.

[5] K. Tanaka (Translated by T. Niimura), 1997, “An Introduction to Fuzzy Logic for Practical Applications”, Springer- Verlag, New York, Ch. 4, 5, pp. 86–136.

[6] Zadeh, L.A., “Fuzzy Sets”, Information and Control, 8, 338–353, 1965.

[7] Brubaker, David I., "Fuzzy Logic Basics: Intuitive Rules Replace Complex Math," EDN, June 18,1992, 111 p.

[8] Glenn A, “Fundamentals of Fuzzy Logic Part I & II”, SENSORS, April 1993.

[9] Earl Cox, The Fuzzy Systems Handbook (1994), ISBN 0-12-194270-8.

[10] SHABIUL ISLAM, NOWSHAD AMIN, M.S.BHUYAN, MUKTER ZAMAN “FPGA Realisation of Fuzzy Temperature controller for industrial application” WSEAS TRANSACTIONS on SYSTEMS and CONTROL Manuscript received June 16, 2007; revised Sep. 17, 2007.

[11] Brown, S. D., Francis, R. J., Rose, J., and Vranesic, Z.G., Field-Programmable Gate Arrays, Kluwer Academic Publishers, 1996.

[12] K. T. Tho, K. H. Yeow, F. Mohd-Yasin, M. S. Sulaiman, and M. I. Reaz, VHDL Modeling of Boolean Function Classification Schemes for Lossless Data Compression, WSEAS Transactions on Computers, Vol. 3, No. 2, 2004, pp. 365–368.

[13] D. Repole and L. R. Adrian, “Introduction to Parallel MAS Control for MAS - Smart Sensor Networks”, 2019 IEEE 60th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 2019, pp. 1–5.

[14] A. Barriga, S. Sánchez-Solano, P. Brox, A. Cabrera, I. Baturone, “Modelling and implementation of fuzzy systems based on VHDL”, International Journal of Approximate Reasoning, Volume 41, Issue 2, 2006, pp. 164–178, ISSN 0888-613X, https://doi.org/10.1016/j.ijar.2005.06.018.

[15] FUZZY LOGIC DESIGN TOOLS, Xfuzzy 25th, V. 3.5, March 2018, Instituto de Microelectrónica de Sevilla (IMSE-CNM), http://www2.imse-cnm.csic.es/Xfuzzy/Xfuzzy_3.5/download.html#DISTRIBUTION.

[16] Silvio Cammarata, “Reti Neuronali, Dal Perceptron alle reti caotiche e neuro-fuzzy”, seconda edizione, ETASLIBRI, 1997.

APPENDICES

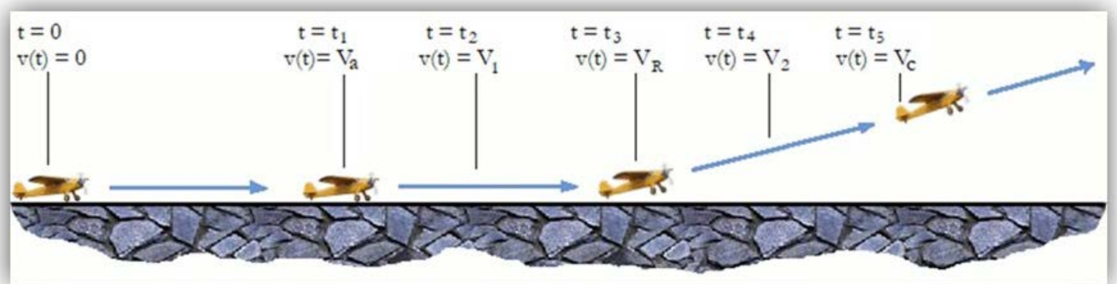


Fig. A.1. Take-off manoeuvre, graphical animation.

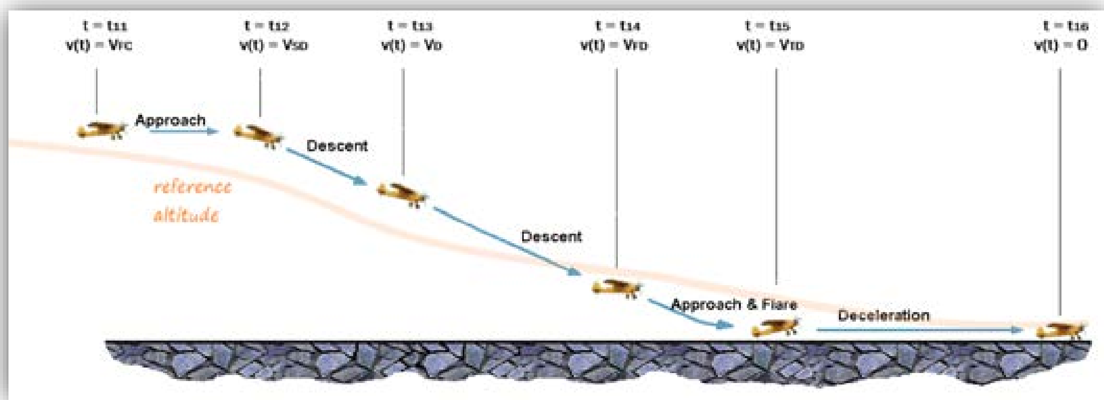


Fig. A.2. Reference landing manoeuvre altitude against vehicle's manoeuvre position, graphical animation.

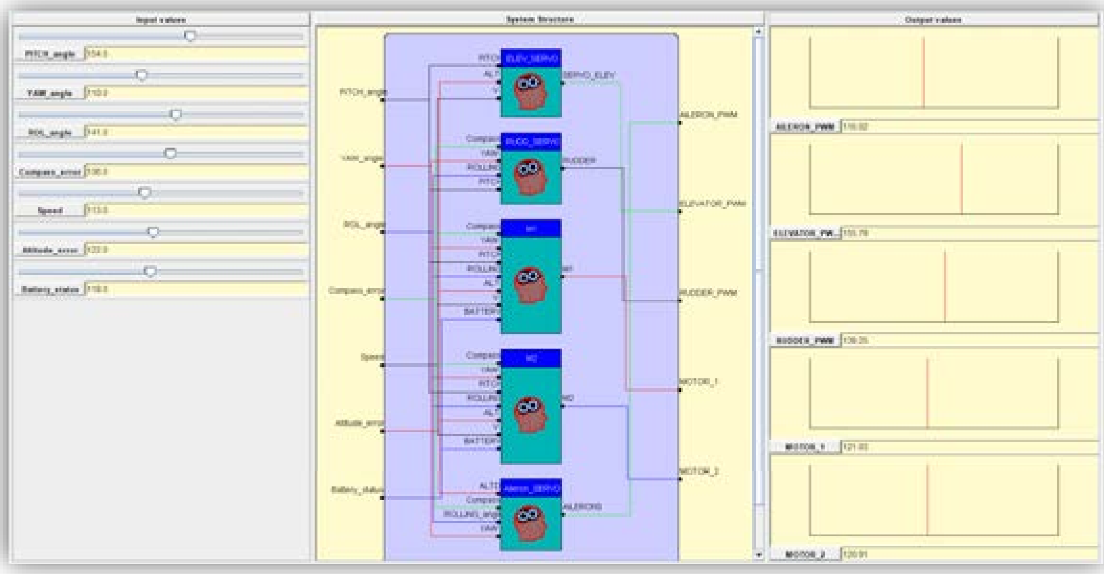


Fig. A.3. “XFUZZY Inference Monitor” tool applied to the proposed controller.

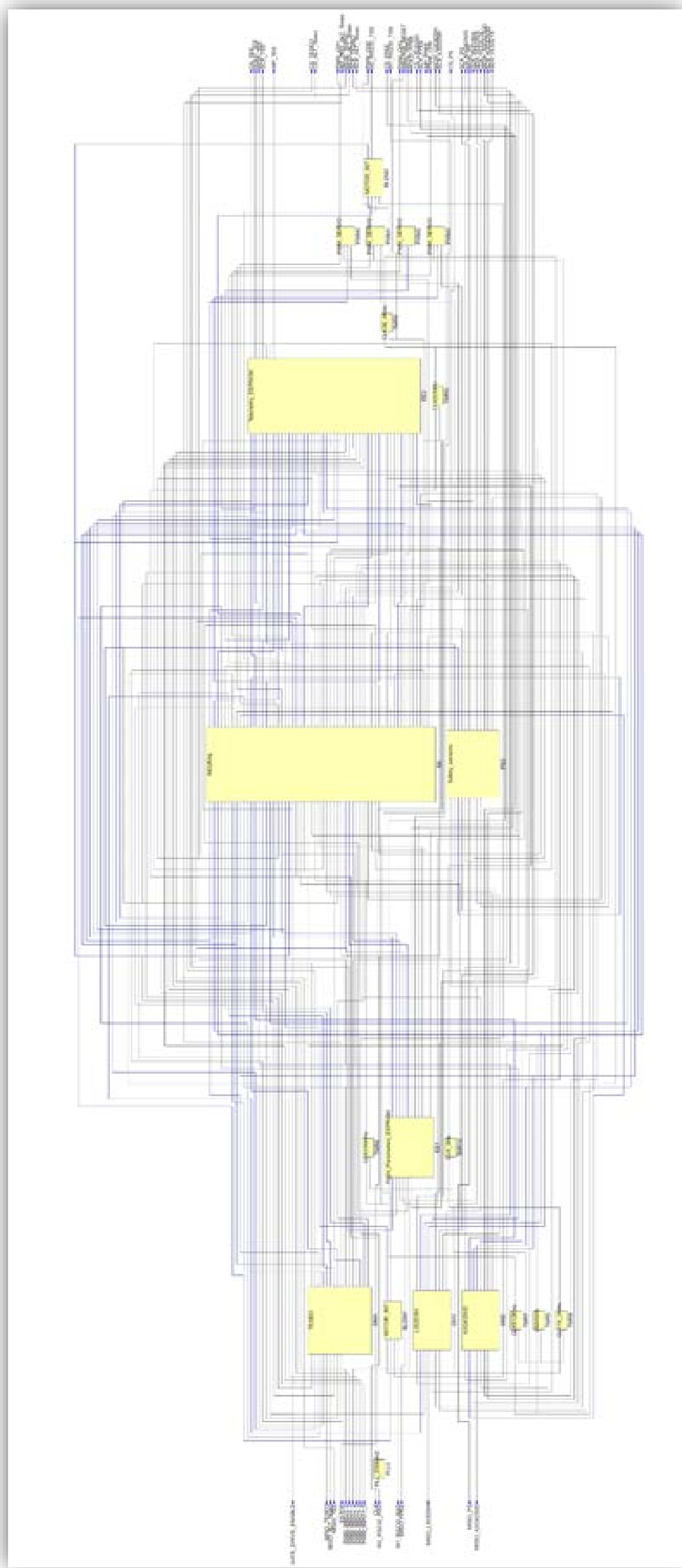


Fig. A.4. RTL view of the outcome of the main VHDL algorithm's synthesis.

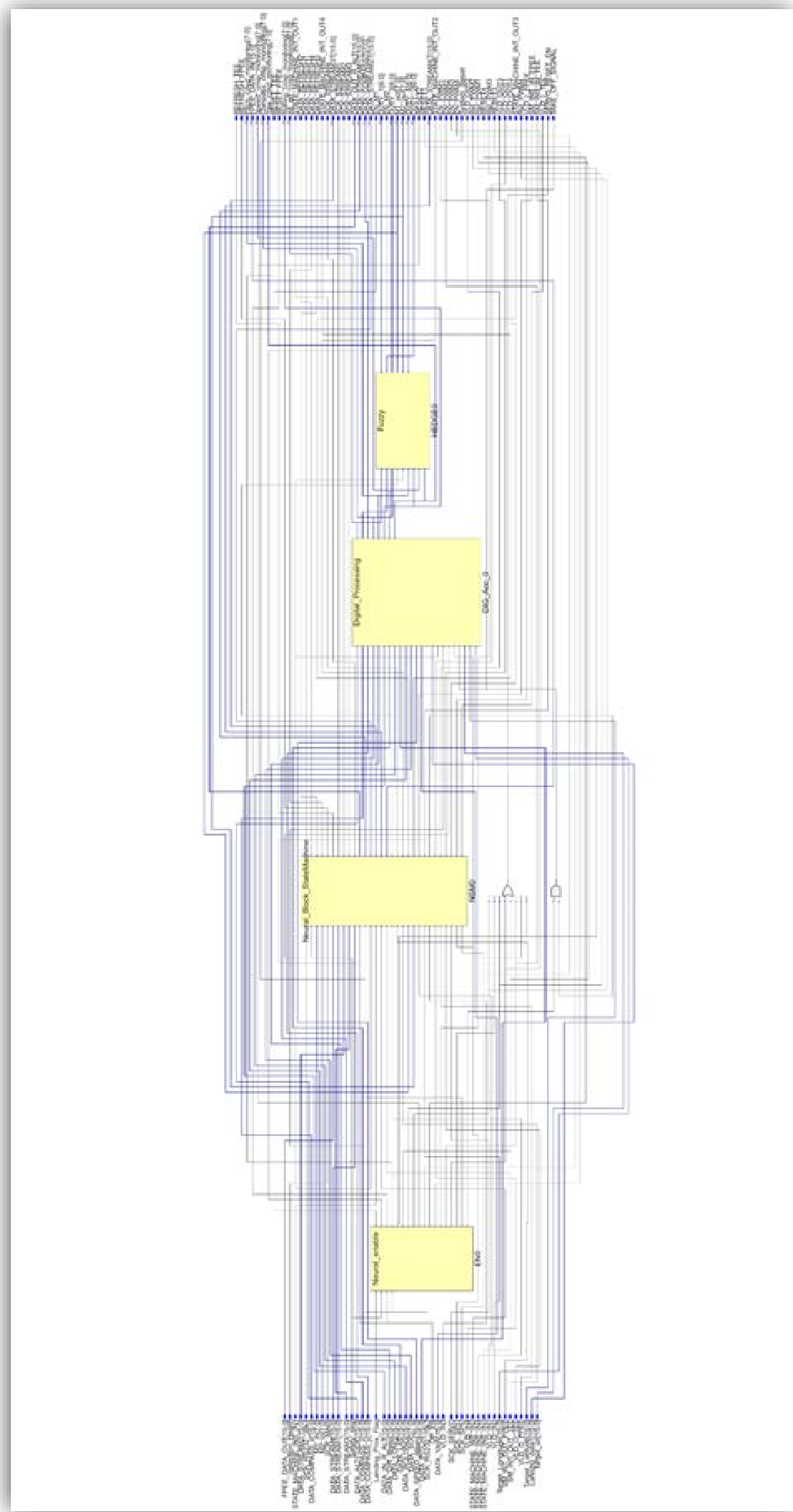


Fig. A.5. RTL view, the outcome of NEURAL VHDL component's code synthesis.

Donato Repole received his BSc degree from the University of Salerno (Italy) and a Master's degree in Electronic Engineering from Riga Technical University after studies in the Institute of Power Electronics and Electrical Engineering of Faculty of Electrical and Environmental Engineering.