23rd ABCM International Congress of Mechanical Engineering
December 6-11, 2015, Rio de Janeiro, RJ, Brazil

PLATFORM FOR SIMULATION AND ADJUSTMENT OF QUADROTORS CONTROLS USING ARM CORTEX-M4 AND X-PLANE

Alessandro Corrêa Mendes
Helosman Figueiredo
Ícaro Bezerra Viana
Luiz Carlos Sandoval Góes
Osamu Saotome
Instituto Tecnológico de Aeronáutica, São José dos Campos, Brazil 12228-900
alcomen@univap.br
helosman@gmail.com
icaro@ita.br
goes@ita.br
osamu@ita

Abstract. *Unmanned Aerial Vehicles (UAVs) have been developed for various applications, such as search and rescue, border monitoring, building surveillance and inspection of electrical transmission lines. Aiming to test control algorithms implemented in embedded devices, this paper proposes a Hardware-in-the Loop (HIL) simulation to assess control laws. The architecture of the simulation platform consists of a simulator, that reproduces a quadrotor vehicle, interconnected to an embedded module comprising an ARM Cortex-M4 microcontroller. Thus, the aircraft dynamics is simulated on a dedicated computer that uses the X-Plane flight simulator as part of an HIL loop. To assess the embedded control system, control laws were implemented to altitude and attitude. The main result of this work is the development of testing procedures and a simulation tool to test embedded control laws before the start of the flight test campaigns.*

Keywords: *Quadrotors, X-Plane, Hardware-in-the-Loop, Embedded Systems*

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have become desirable for many applications where manned operation is considered inefficient or too risky. Such devices have attracted increasing interest from companies and government applications that according to Alexis *et al.* (2011) are search and rescue, wild fire surveillance, buildings inspection, agricultural services, mapping, marine operations, battle damage assessment, border interdiction, and law enforcement. Other applications are identified in Tarhan and Altug (2010) as reconnaissance, inspection for natural disasters, as a remote sensor for atmospheric measurements, inspection of pipe lines and power lines, or for aerial images. Among the various UAVs platforms, such as fixed wing Teo *et al.* (2004) and blimp Zufferey *et al.* (2002), our focus in this paper is the quadrotors. The quadrotors are vehicles of VTOL (Vertical Take-off and Landing) type and with four independent rotors as shown in Figure 1. The VTOL systems have specific characteristics that allows several diverse applications. As such, it is possible to accomplish vertical flights, hovering and flight in low velocity. Its concept offers better useful load and is simpler to construct and control Bouabdallah *et al.* (2004). Its small size and high maneuverability allow to flight in indoor and outdoor environment Santana and Braga (2008).

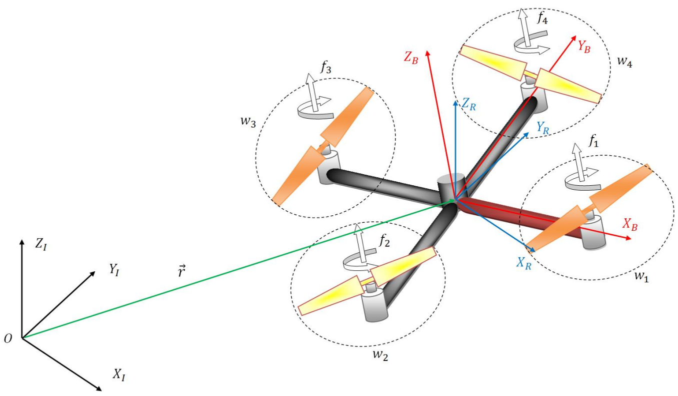


Figure 1: UAV type quadrotor.

The characteristics mentioned above have motivated many researchers to propose flight controllers capable of stabilizing UAVs in various applications. However, from the point of view of control, these aircraft are unstable, non-linear, subactuated with highly coupled and complex dynamics. These characteristics can make the experimental test of these UAVs, regardless of their nature, to pose a great risk to not only for the equipment under development, but also to any bystanders. This is due as these vehicles can fly at high speeds and have high speed rotors.

This problem motivated the creation of simulators with a high degree of realism, which is important for the development of these technologies.

Simulations reduce the development time of new technologies, because it is possible to perform faster tests and to be less dangerous than performing a practical experience. Through simulations, such as hardware-in- the-loop (HIL), it is possible not only to validate the system before an experimental test, but also to project a compatible hardware and to avoid physical injury. Pizeta I. H. B. and R. (2012).

Safety is therefore a primary issue that is being considered in conducting actual flight tests Cai *et al.* (2008). HIL simulators are used to test the hardware, as this process includes the hardware into the simulation loop. By performing tests with different environmental conditions and analyzing the hardware response, a comprehensive assessment system can be made. It is worth mentioning that this is one of the last steps before conducting a real experiment Coombes *et al.* (2012), because once the hardware is validated within this environment, it is capable of being tested in a practical experiment.

Given the presented context, this work aims a HIL simulation capable of test control laws for quadrotor UAVs. This simulation environment integrates the flight simulator X-Plane with an ARM Cortex-M4 microcontroller.

The remainder of the article is organized as follows. Section II describes the Hardware-In-The-Loop simulation environment. Section III presents the Hardware-In-The-Loop implementation. Section IV shows Simulation Results and Section V contains the conclusion and suggestions for future developments.

2. HARDWARE-IN-THE-LOOP SIMULATION ENVIRONMENT

The simulation environment provides capabilities for the assessment of control algorithms for UAVs of quadrotor type. It contains a computer with enough computational capacity to run the flight simulator X-Plane and to set communications with an ARM Cortex-M4 microcontroller from Texas Instruments Inc. as shown in Figure 2.

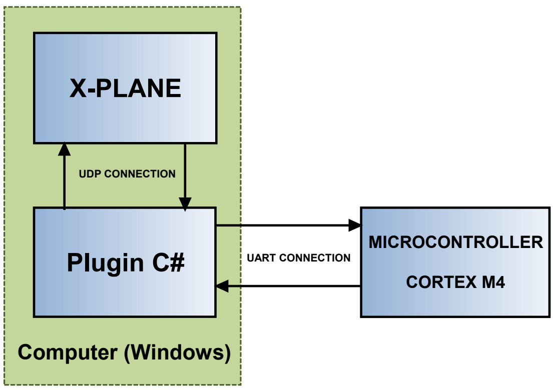


Figure 2: Environment.

The purpose of this paper is to provide an embedded control adjustment tool using the technique in HIL for after use the same hardware to be validated in real flight.

2.1 X-Plane Flight Simulator

Demand for simulators increased together with the UAV market. Currently the market has several options for aircraft simulators such as: Flight Gear, Microsoft Flight Simulator and X-Plane. The X-Plane was chosen for this work because it differs from traditional simulators, The key difference is the method used to simulate the dynamics of the aircraft. The chosen simulator uses the simulation method called Blade Elements to consider all parts of the aircraft in flight, different from traditional method by equations of dynamics. The Blade Elements method has a higher computational cost, however they have a gain in the level of realism of Simulation. The software is certificated by FAA (Federal Aviation Administration), and it also allows for real-life conditions simulation, such as wind and turbulence. The simulator also provides an aircraft construction model called Plane-Maker, based on blade elements where, given the aircraft geometry, engine and propellers parameters, it is possible to build the desired aircraft, besides facilitating simulator data insertion and extraction through User Datagram Protocol (UDP). Indriyanto and Jenie (2010).

The aircraft model used for simulation is the same platform simulation quadrotors developed in the work Figueiredo and Saotome (2012). The aircraft is shown in Figure 3. The quadrotor has four independent fixed-pitch propellers.

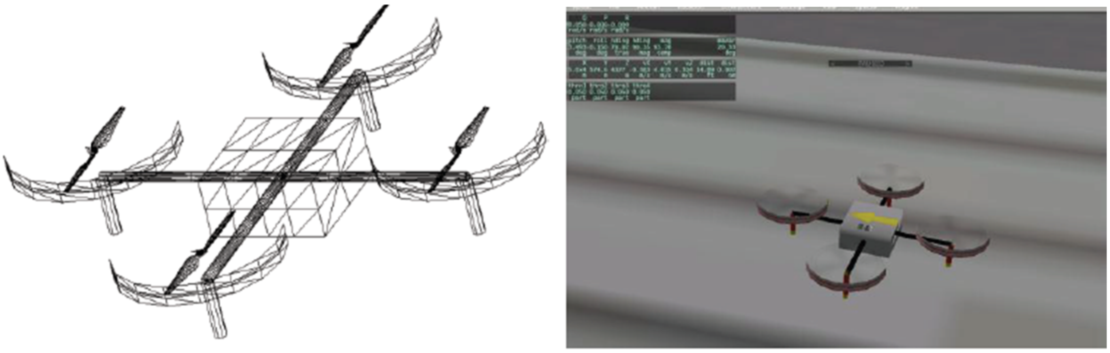


Figure 3: (a) Quadrotor built in the Plane-Maker software. (b) Quadrotor flying in the X-Plane flight simulator.

2.2 ARM Cortex-M4 microcontroller

In this project it was chosen the ARM Cortex-M4 microcontroller (TM4C123GH6PM) from Texas Instruments because this microcontroller is able to work directly with floating-point variables, which contributes to the accuracy of control calculations carried out by the microcontroller, and increases the computing power.

To run this application, the CPU clock was configured in 16 MHz. This microcontroller also features: high performance 32-bit applications, up to 80 MHz CPU clock, Floating-Point Unit (FPU), low Power consumption, 8-channel serial (UART), channel SPI, 6-channel I2C, two CAN channels among others.

3. HARDWARE-IN-THE-LOOP IMPLEMENTATION

The X-Plane sends the flight data through Ethernet interface the microcontroller, this data is processed by the control algorithm embedded in microcontroller that return the precessed throttle values from each motor to X-plane, as described in the Figure 4. To change gain values and references from embedded controls in real time during the simulations, a HMI (human machine interface) using a color touch screen LCD.

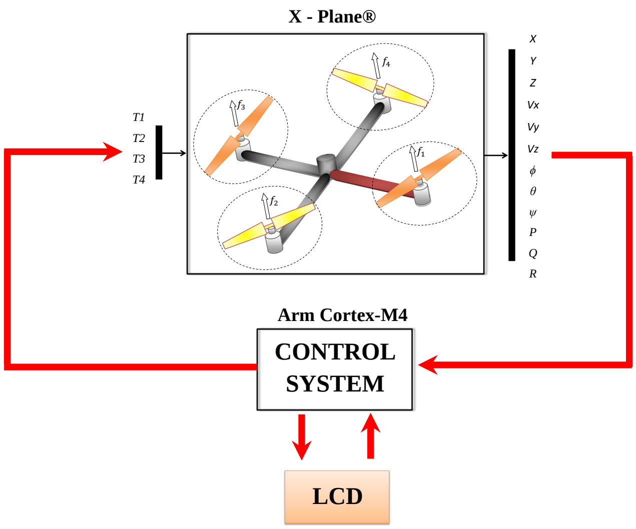


Figure 4: Simulation environment data flow.

As previously mentioned, the quadrotor is an unstable aircraft, so in order to perform HIL simulations, proportional (P) controllers were added to attitude and altitude. Figure 5 shows the block diagram of the implemented control system.

The proportional controllers were implemented with multiplicative gains. The controller gain was adjusted empirically during flights with some adjustments using HMI interface on embedded control.

In order to facilitate adjustments in the loop control during the tests, we developed a human machine interface (HMI), with a LCD touch screen whose display allows the user to adjust the references and the gains of controls system. This

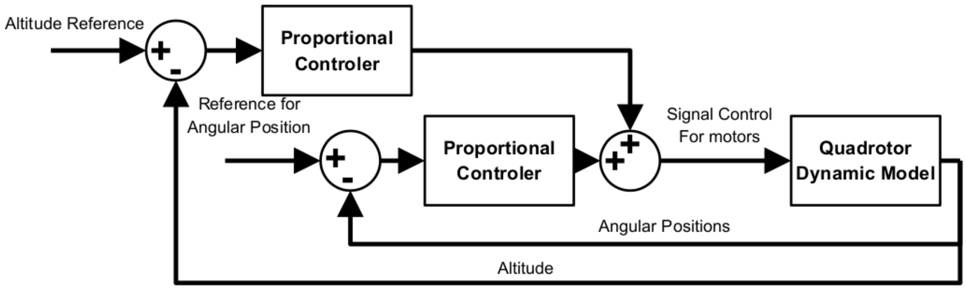
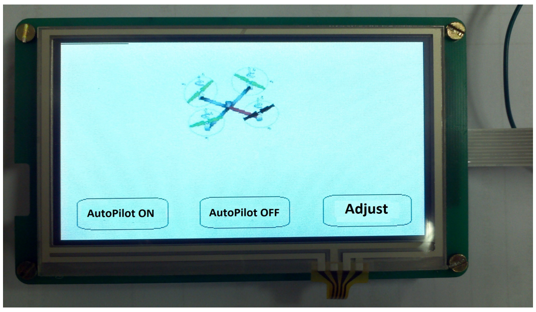
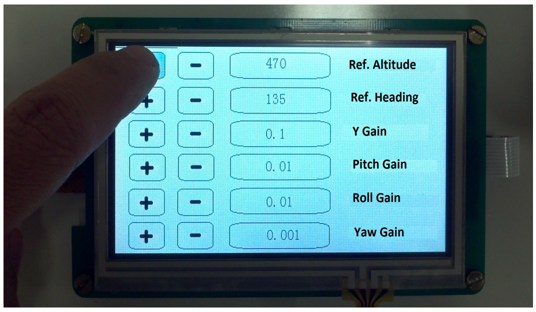


Figure 5: Control Block Diagram

facilitates the study of the control behavior when there is need to change the reference values while the aircraft is in flight, without the need to recompile the microcontroller software. This interface is shown in Figure 6.



(a) Setting screen.



(b) References and gains adjusts screen.

Figure 6: Screens

3.1 Communication

The Software X-Plane communicates with the outside world through one input port and one output port on standard UDP protocol. The port address number can be configured by the user. Due that the microcontroller lacks an UDP interface, it was necessary to find a form to communicate with the X-Plane. This problem was solved by developing a plugin in C# where the data originated from the X-Plane UDP port are transmitted on standard UART (Universal Asynchronous Receiver/Transmitter) at 115200 bps (bits per seconds) to the serial port of the microcontroller, and the data originated from the Microcontroller serial port is converted and sent on UDP to X-Plane, as shown in Figure 7.

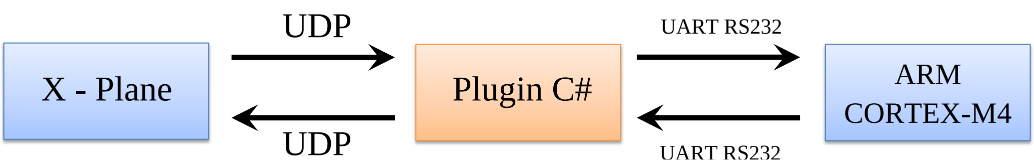


Figure 7: Block Diagram of communication.

After the packages were received and unpacked, the variables (Pitch, Roll, Yaw and Altitude) received from the X-Plane are loaded, then they are used at the input of control loops. The control loops calculates the motor throttle (T1, T2, T3 e T4) to keep values attitude and altitude and then send back these values to the X-Plane software. The UDP data packet sent by the flight simulator is shown in Figure 8.

3.2 Hardware

This project used the development kit TIVAC EK-TM4C123from Texas Instruments, which contains an ARM microcontroller M4 32bit floating point. The gain with floating point capability enables the use of control techniques that require greater computational load. With this micro-controller can implement since proportional controls, up to advanced control techniques as optimal control and predictive control. This hardware is the same that will be embedded in the aircraft for flight will be integrated into this hardware inertial sensors and barometer.

This microcontroller has several peripherals, it was used a serial port UART to communicate with the computer, who in turn sends and receives X-Plane data packets, therefore forming a closed loop.

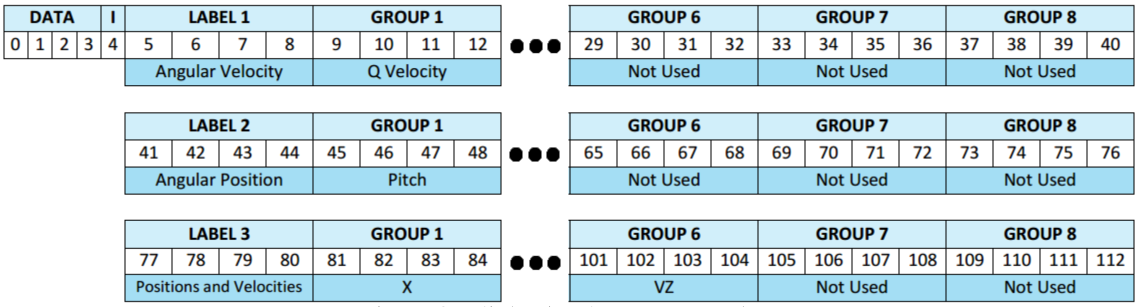


Figure 8: Flight simulator UDP packet.

The control was embedded into the microcontroller, and it was enabled the debugging line by line of code during runtime. The data from the X-Plane are received through the microcontroller UART serial port using a serial interrupt. Data is stored in a buffer, which will later be treated and split into fields used in the control system calculations. Fields are separated into Pitch, Roll, Yaw, Altitude, Heading, Speed X, Speed Y and Speed Z.

After evaluating control calculations, variables are created accordingly to the acceleration of each motor. These variables are in proportion to acceleration and are packaged and sent to the X-Plane by the same UART serial interface.

In Embedded Systems the time to process each fuction is very important. This will show if the time used is enough to run the application in real time, if the microcontroller has the capability to evaluate calculations of all the control laws, process data receive and send to XPlane. In flights this times is import because all delay of times process will be a fail on attitude control.

The time measurements were carried out using the same hardware. Four pins of the microcontroller were enabled to measure the time process. When start of a process this pin goes to logic HI (high) level and on the end process of that task goes to logic LO (low). The test points of the one process is shown in Figure 9. This signal can be read and measured using an oscilloscope as shown inTime measurements were carried out using the same hardware. An unused pin from the microcontroller was used as output. When a task starts, this pin output raises to logic HI (high) level and by the end of that task it falls back to logic LO (low). This signal can be read and measured using an oscilloscope as shown in Figure 10.

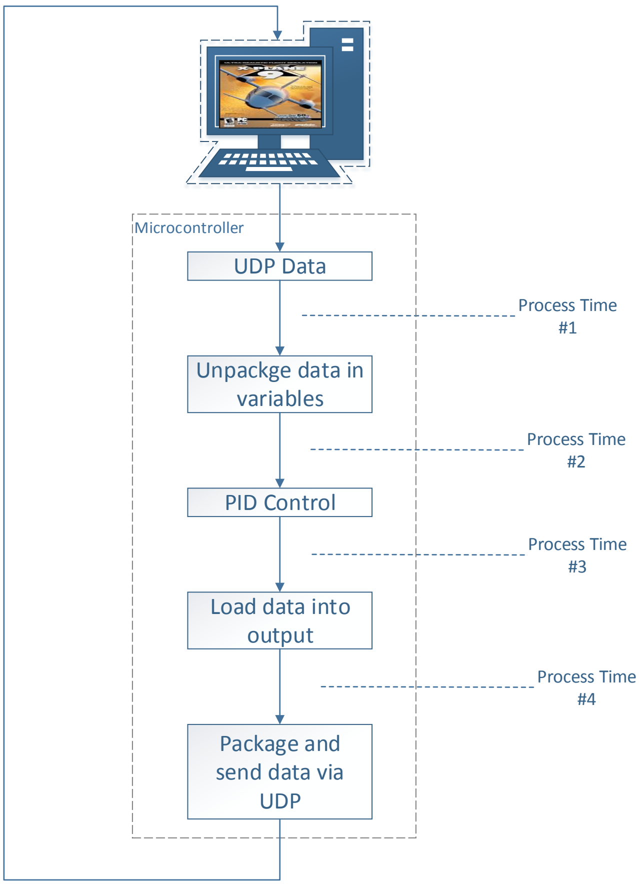


Figure 9: Test point to measure each process time.

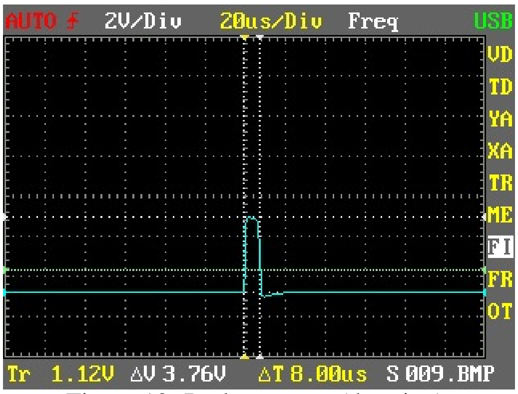


Figure 10: Packages sent (duration).

Tasks were separated into: rate of packages are received from X-Plane, time to unpackge the UDP values, time to calculate all controls, load the data into output and time to send packages to X-Plane. Some time values are shown in Table 1.

Table 1: Timing from hardware.

Nr.	Task	Time	Rate
1	Packages received from X-Plane		60 Hz
2	Control process (duration)	28 μs	
3	Packages sent (duration)	8 μs	

4. RESULTS

For simulation tests, it has been implemented control laws in order to stabilize the quadrotor, the controlled variables are altitude and attitude (roll,pitch and yaw), where the gains of the controllers are tuned empirically. Gains adjustments were performed with the following experiment, it took the system to the point that maintained oscillations, near this condition the gains were being reduced gradually each adjustment were performed new simulations to evaluate the system response. This method was used because it is this initial conception work platform HIL, and is not suitable for actual aircraft use. A simulation was performed for a period of 35 seconds and obtained the following responses that are shown in Figures 11 and 12.

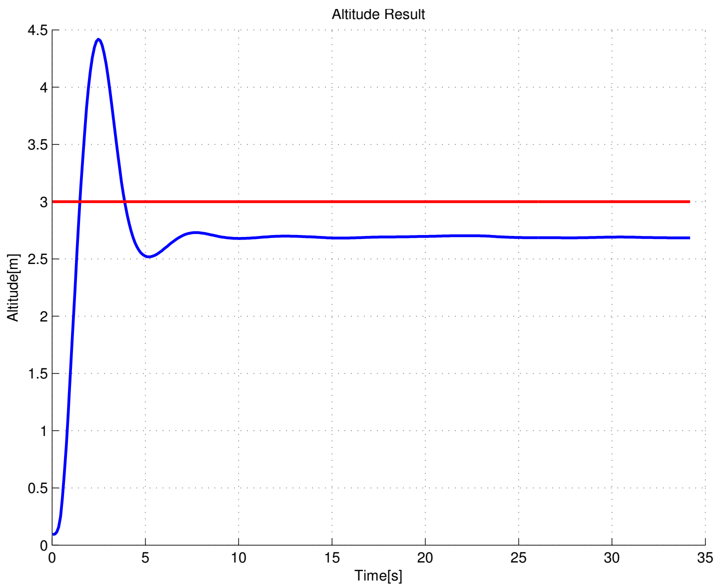


Figure 11: Altitude Result.

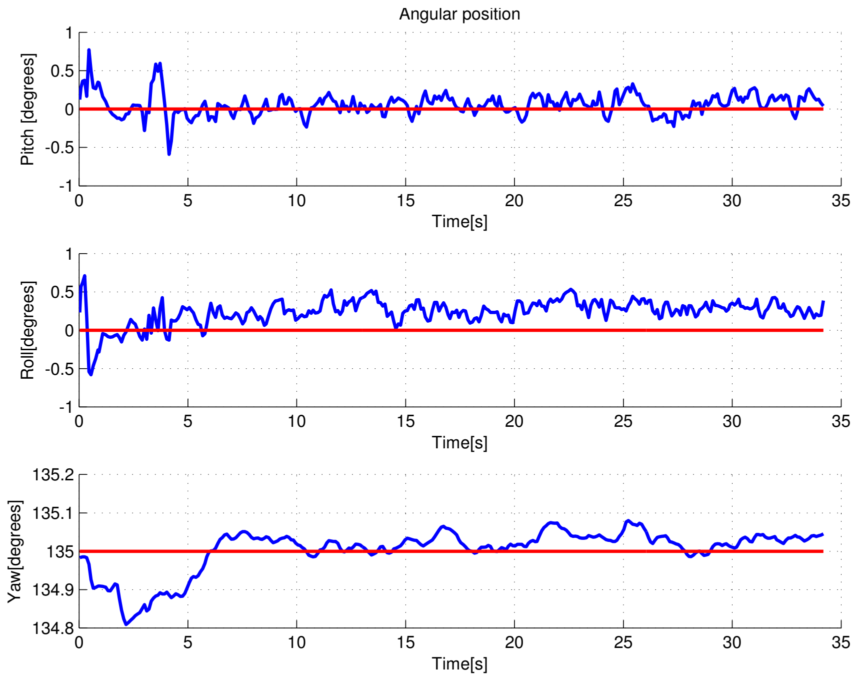


Figure 12: Angular Position.

As it can be observed that the system can track the reference signal, is worth mentioning that the proposed controller and its demonstration of stability is outside of the scope of this paper. On the other hand, the simulation environment allows for a sensory reading and demonstration of the quadrotor real time values, enabling analysis via a graphical user interface. Such visualization facilitates the analysis of the behavior of the aircraft during flight, as shown in Figure 13 and the experimental test in Figure 14.

Figure 13: Simulation Environmental in X-Plane.

The addition of touch screen interface enables this platform to analyse reconfigurations of control in flight, in this work this reconfigurations have been made to adjust the values of the gains.

5. CONCLUSION

We developed an environment for testing control laws applied to UAVs of quadrotor type. The environment allows the analysis of the behavior of the embedded controller. The simulation environment assists in UAVs development, since it allows the inclusion of data in the simulation through the LCD screen as well as the use of these data for the effective control in real time. The purpose of the Environment is to propose a simulation that emulates as close as possible the conditions, so that the most adverse situations can be observed, allowing for validation of the proposed controller, making it reliable to be used in a practical experiment. In the next work plan to conduct all phases of control system design, such as modeling, simulations, project controllers, and discretization for embedded systems.

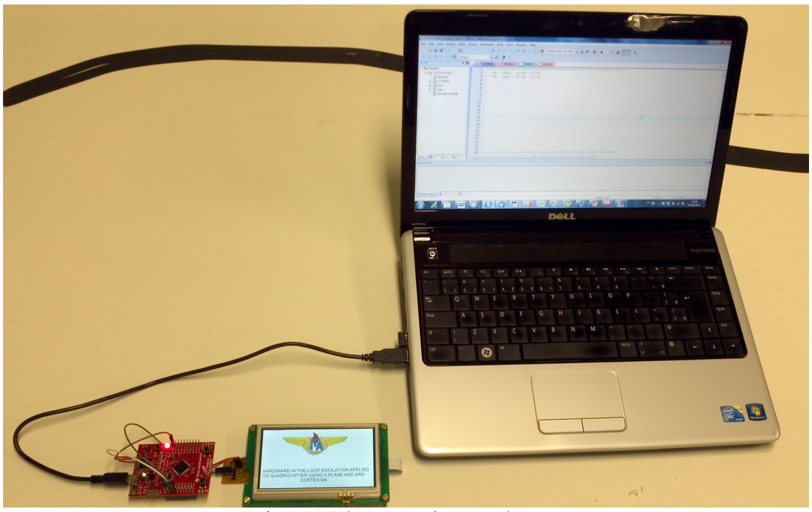


Figure 14: Experimental test.

6. ACKNOWLEDGEMENTS

The first author thanks UNIVAP for their support and encouragement, and is also thankful to Texas Instruments, who in the person of Eng. Nuncio always provided support for this project, and their Development Department for providing samples of components.

7. REFERENCES

Alexis, K., Papachristos, C., Nikolakopoulos, G. and Tzes, A., 2011. "Model predictive quadrotor indoor position control". *19th Mediterranean Conference on Control and Automation (MED)*, Vol. 57, pp. 1247–1252.

Bouabdallah, S., Murrieri, P. and Siegwart, R., 2004. "Design and control of an indoor micro quadrotor". *IEEE International Conference on Robotics and Automation*, Vol. 5, pp. 4393–4398.

Cai, G., Chen, B.M., Lee, T.H. and Dong, M., 2008. "Design and Implementation of a Hardware-in-the-loop Simulation System for Small-scale UAV Helicopters". , No. September, pp. 29–34.

Coombes, M., Mcaree, O., Chen, W.h. and Render, P., 2012. "Development of an autopilot system for rapid prototyping of high level control algorithms". , No. September, pp. 3–5.

Figueiredo, H.V. and Saotome, O., 2012. "Modelagem e simulação de veículo aéreo não tripulado (vant) do tipo quadricóptero usando o simulador x-plane e simulink". In *Anais do XIX Congresso Brasileiro de Automática, CBA 2012*. Campina Grande, Brazil.

Indriyanto, T. and Jenie, Y.I., 2010. "Modeling and simulation of a ducted fan unmanned aerial vehicle (uav) using x-plane simulation software". In *Regional Conference on Mechanical and Aerospace Technology*. Bali.

Pizeta I. H. B., Brandao A. S., F.M.S. and R., C., 2012. "Uma plataforma hardware in loop para vants de asas rotativas". , No. September, pp. 3–5.

Santana, P.H.R.Q.A. and Braga, M.A., 2008. "Concepção de um veículo aéreo não-tripulado do tipo quadrirrotor". *Trabalho de Graduação, Universidade de Brasília*.

Tarhan, M. and Altug, E., 2010. "Model predictive quadrotor indoor position control". *Journal of Intelligent and Robotic Systems*, Vol. 62, pp. 587–607.

Teo, R., Jang, J.S. and Tomlin, C.J., 2004. "Model predictive quadrotor indoor position control". *Proc 43rd IEEE Conf. Decision Control, Paradise Island, Bahamas*, Vol. 4, pp. 4268–4273.

Zufferey, J.C., Floreano, D., van Leeuwen, M. and Merenda, T., 2002. "Evolving vision-based flying robots". *Proceedings of the 2nd International Workshop on Biologically Motivated Computer Vision-Springer-Verlag Berlin Heidelberg*, pp. 592–600.