Building an Autonomous Scale Electrical Vehicle with LabVIEW and myRIO

Ariel Guerrero, Micaela Jara, Erid Pacheco, Ariel Bogado y Jesús Franco - Universidad Católica "Nuestra Señora de la Asunción" Campus Alto Paraná, Paraguay

THE CHALLENGE

Building an autonomous scale electrical vehicle to compete in the first brazilian autonomous scaled vehicle race (Robocar Race) held in São Paulo, Brazil.

THE SOLUTION

Using an inertial sensors (IMU) , and NI myRIO hardware as the in-car embedded controller to implement a path planner and a vehicle control with NI LabVIEW software.

Products: NI myRIO, NI LabVIEW, LabVIEW FPGA

Figure 1. Aguara'i at the Paulista Radiomodelism Society circuit.

Brazilian branch of National Instruments challenged LabVIEW Student Ambassadors (LSAs) to design and build an autonomous scale vehicle to compete at the Robocar Race, in return, they would give them a NI myRIO controller. Our ambassador for PTI-UCA accepted the challenge and assembled the first team from Paraguay to develop an autonomous scaled vehicle. Our goal was to win the competition that consisted in running the track in the shortest time. We entered the race with our car called Aguara'i (little fox in Guarani), an RC car that we modified to be autonomous. Although we finished in 4th place, our work inspired a lot of engineering students from our country to take part of this exciting and promising field.

System Configuration Autonomous Vehicle at 1: 8 scale: Modified chassis of a miniaturized electric RC car in which a myRIO 1900 is adapted as well as the sensors and actuators required for inertial navigation (IMU). In this platform the algorithms that allow determining the position and orientation information based on the sensors will be executed. It will also allow the acquisition of data from the sensors, execute the selected navigation algorithm and determine the control signals to the actuators. **Base Station:** For configuration of the navigation parameters (waypoints) and visualization of the status of the sensors. Human-machine interface with LabVIEW.

Using LabVIEW and myRIO

The first step is to acquire the signals of the sensors. The encoder was connected to one of the FPGA's inputs, and the IMU sensors was connected through a serial connection from a arduino microcontroller (ATMega2560) to the myRIO CPU.

In order to control the speed and direction of the car, we made use of the parallelism of the loops in the FPGA, these PWMs values to control the direction and the speed came from constantly updated values in the myRIO. To communicate this data from the RT-target to FPGA-target we use the global variables, since we needed that several processes running in parallel in labview, can communicate their data and in this way obtain the processing and control of the state variables of the car in real time. Next, the configuration of the system can be seen in figure 2

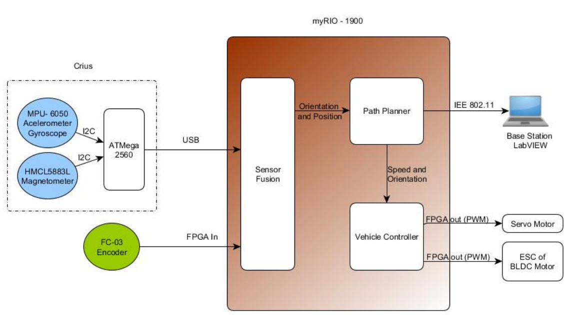


Figure 2 - System Settings

Sensor Fusion

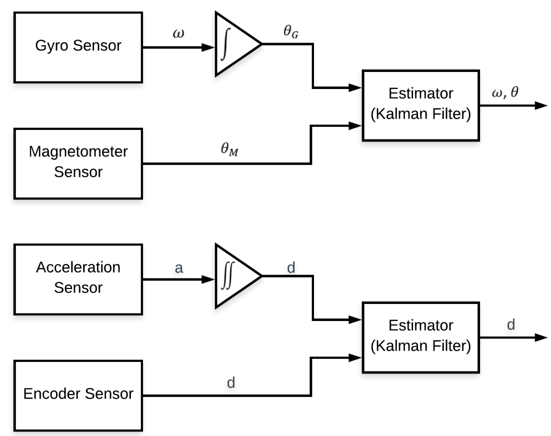


Figure 3 - Sensor Fusion

Sensor signals were used to improve and correct the position measurement of the 4-wheel autonomous vehicle to obtain a more reliable position estimate. From this, we calculated the estimation of the position and reduced the systematic and nonsystematic errors during the tests and we succeeded in estimating the deviation of the turn bias. The basic tool here is a Kalman filter.

In Figure 3, ω is the data of the angular velocity coming from the gyroscope, a is the acceleration of the accelerometer and θ is the angle referring to the magnetic north obtained from the magnetometer. From the estimators, the distance traveled and the orientation are obtained, to then obtain the coordinates (x, y).

Path Planner

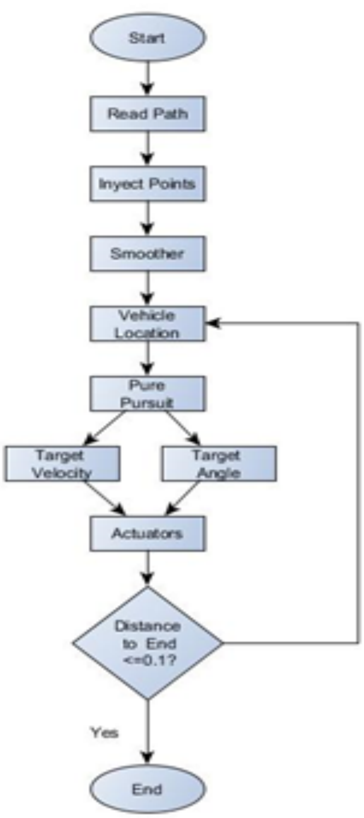
This module is responsible for taking the autonomous vehicle from an initial position to a final, following a trajectory. The algorithm used is the so-called pure pursuit algorithm. For practical purposes, the implementation of "Team 1712" has been used and modified according to the requirement.

With this algorithm it is possible to determine the target speed of the autonomous vehicle depending on the curvature of the segment of the trajectory in which the autonomous vehicle is located, as well as to establish the direction to which it should go knowing its current position and a target point called "Look Ahead Point". To obtain greater precision in the calculation of the speed and curvature, points were injected into the original trajectory obtaining in this way closer points, to then pass them through a smoothing stage and achieve continuity in the path.

Figure 3 - Path Planner

Conclusion

NI hardware and software allowed us to develop an autonomous scale car in a short time and with little previous knowledge.



If it wasn't for the NI platform,which made us achieve a very high level of development without entering into hardware programming and allowed us to make use of modularity even using third-party code, we wouldn't have been able to develop the car in time for the competition. After we accepted the challenge, we had just two months to design and build an autonomous car.

The project emphasizes the pedagogical value of teaching based on challenges, allowing students to acquire not only teamwork experience, in a multidisciplinary environment, but also the experience of reusing third-party codes. It too allowed students to implement concepts developed in different subjects of the university such as calibrating the sensors and improve the reading efficiency by minimizing the errors through the implementation of the Kalman filter.

The effectiveness of the navigation algorithm has been checked, as well as the estimation of the current position through the acquisition of data from the sensors and the fusion of them.

After a few test and readjustments, Aguara'i was able to run from an initial position point to a final, following a trajectory previously defined with a minimal error.