Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-6 (2016) 242–247

# Low-cost Hardware-in-the-loop Testbed of a Mobile Robot to Support Learning in Automatic Control and Robotics ⋆

**Inés Tejado, Juan Serrano, Emiliano Pérez, Daniel Torres and Blas M. Vinagre**

*Industrial Engineering School,*
*University of Extremadura, 06006 Badajoz, Spain*
*(e-mail: {itejbal,emilianoph,datglez,bvinagre}@unex.es,*
*jserranokc@alumnos.unex.es)*

**Abstract:** Hardware-in-the-loop (HIL) systems are recognized to be effective tools to support teaching in Controls, especially in laboratory courses, rather than only simulations, since a more realistic student experience is achieved with no need of the use of a real equipment. This naturally leads to a better use of the resources for both the students and the university facilities and a reduction of costs invested in laboratory equipment. This paper presents a low-cost HIL application for a mobile robot to support learning in Automatic Control and Robotics. It consists of a simulator of the robot, which is built by means of physical modeling tools in the MATLAB®/Simulink® environment, and an Arduino Yún board. The communication between the simulator and Arduino is done by TCP/IP. For illustration purposes, some examples of possible control coursewares to be carried out using the developed testbed are given.



*Keywords:* Education, Control, Testbed, Simulink, Arduino, Mobile robot.

## 1. INTRODUCTION

Laboratory teaching in Automatic Control and Robotics becomes a challenge for large groups of students (probably with more than 6 people). In that situation, one of the most effective and reasonable alternative to support teaching is to use hardware-in-the-loop (HIL) systems to support teaching in order to avoid students to queue to test their designs in a real platform, reducing consequently the wasted time. Furthermore, the use of real platforms needs a heavy budget in terms of resources, especially related to multiplicity of equipment. HIL systems are somehow embedded in the simulation loop aiming to get more accurate results, closer to the real case. In such systems, part of the simulation loop is composed of computer software, while the rest is hardware systems. Although the hardware in the loop can either be the controllers or the plant, in this work we focus on the former case. Luo et al. (2013); Lu et al. (2007); Sala and Bondia (2006); Martono et al. (2005) proposed some examples of HIL applications in education in engineering degrees.

Nowadays, Arduino is not only a low-cost solution for many engineering problems but also very portable, so it becomes a valuable resource for HIL applications. Coombes et al. (2014); Gambino et al. (2014); Vidal et al. (2014); Xue and Chen (2013) discussed how to integrate Arduino in HIL systems.

⋆ This work has been supported by the Spanish Ministry of Economy and Competitiveness under the project DPI2012-37062-C02-02 and the Junta de Extremadura under the Ayuda a Grupos with reference GR15178.

Given this context, the objective of this paper is to build a low-cost HIL testbed for a mobile robot to support laboratory teaching in Automatic Control and Robotics as a platform for the students to validate control architectures and algorithms in a more realistic way rather than only simulations. It consists of a simulator of the robot, which is built in the MATLAB®/Simulink® environment by means of the physical modeling tool Simscape™, and an Arduino Yún microcontroller, in which the students will implement the controllers designed for the proposed control problems. The communication between the simulator and Arduino will be done by TCP/IP. Some examples of possible coursewares to be carried out using the developed testbed are given.

The remainder of this paper is organized as follows. Section 2 gives the main characteristics of the mobile robot under study to understand its simulator later on. In Section 3, the testbed developed for the mobile robot is described. Section 4 proposes some experiments that may be carried out with the testbed. Finally, Section 5 draws the main conclusions of this paper.

## 2. THE MOBILE ROBOT

This section describes the mobile robot under study. As mentioned, the aim is to build a tested of this platform to support laboratory lectures in Automatic Control and Robotics. The details explained below are then given to understand the simulator described in Section 3. Figure 1 shows a picture of the mobile robot. As observed, it consists of:

- The cart is the National Instruments DaNI mobile robot, which is a four-wheel platform, two at each

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.184

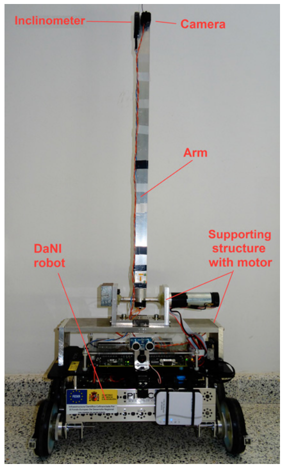


Fig. 1. Picture of the mobile robot with flexible arm

side, with traction of differential type that features sensors, motors and a single-board RIO (sbRIO) board for embedded control with a real-time processor, reconfigurable field-programmable gate array (FPGA) and analog/digital inputs/outputs. DaNI robot acquires data about obstacles from an ultrasonic transducer placed at the top of its front. We include a Wi-Fi router to add the robot the possibility to communicate with a computer. More details about this platform can be found in National Instruments (2014a,b).

- The flexible arm is actually an aluminium rectangular bar of 61 cm of length and 0.4 kg of mass. Taking into account the structural features, only single-mode vibration of the bar is possible, along the longitudinal direction of the cart. The angle of the bar is measured by an inclinometer placed at its tip. To perform supervision and inspection tasks, a minicamera is also located at the bar tip.
- The support of the arm is a structure which combines U- and inverted-U-shaped steel pieces, which is in charge of holding and fixing the bar and enables its positioning by a DC motor Maxon (model A-max 32 20 W). The bar is coupled to the motor shaft via a cylindrical piece. This motor is connected to the FPGA through the CAN port thanks to the National Instruments 9853 module, which allows to perform its control.

## 3. BUILDING THE TESTBED

This section describes the testbed built for the mobile robot with flexible arm. Figure 2 shows a connection scheme of the testbed. It can be observed that the simulator of the mobile robot will run on a computer, in the MATLAB®/Simulink® enviroment. The controllers will be embedded on an open hardware Arduino Yún microcontroller. Data exchange between the simulator and the Arduino board will be carried out through TCP/IP protocol via Wi-Fi, configured as follows: computer and Arduino IP addresses assigned by Windows operation system, BigEndian byte order, 32-bit data and IEEE Standard 754 floating point for data representation.

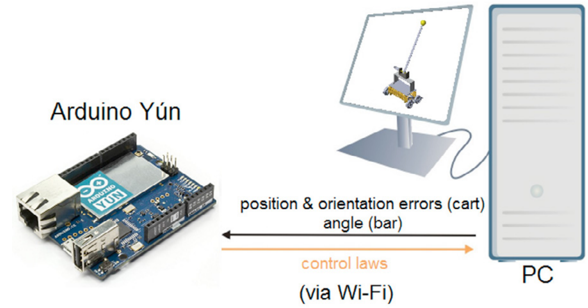


Fig. 2. Connection scheme of the testbed

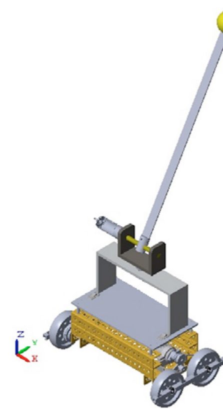

Fig. 3. Appearance of the simulator of the mobile robot

### 3.1 *Simulator of the mobile robot*

The simulator of the robot is built in Simscape™, which is a toolbox for physical modeling developed by the MathWorks for Simulink©. This type of modeling uses a physical network approach to model building: only requires to join the physical components with physical connections to define the underlying dynamic equations of the system to be modeled. A specialized toolbox for mechanical systems, included in the Simscape™ suite, is SimMechanics™, which extends Simscape's capability providing a multibody simulation environment in 3D: it is possible to import computer–aided design (CAD) models of the system components (`.stl` files), from which an automatically generated 3D animation lets visualize the system dynamics.

With SimMechanics™, the model of each component of the system is built connecting bodies by means of joints. The bodies will be defined given a number of parameters: (i) mass, (ii) matrix of inertias with respect to each axis, (iii) center of gravity, and (iv) points that belong to the body, which determine the body geometry if an external graphic file is provided. In the case of inserting a definition file of the body in `.stl` format, it is also necessary to establish a point of the body as the reference point on which the body is rendered. In particular, we are going to use the *Second Generation* library of SimMechanics™ to build the simulator as follows.

Figures 3 and 4 show the appearance and the whole model of the simulator of the mobile robot developed in Simscape™, respectively. The main components which take part in the model, from right to left in Fig. 4, are: