Report template Lab 4

* Group member 1 ID: 302407
* Group member 2 ID: 305208

# I/O interfaces

In the controller Simulink model used to generate the firmware binary file, we used two inputs and three outputs:

* The displacement of the car mass ()
* The displacement of the wheel mass ()
* The damping coefficient (*c*)
* The error signal (*Error*)
* The blinker

Since we get the two inputs from the plant, they are analogic signals that need to be converted

into digital signals, so that they can be compatible with Arduino.

The conversion formula that we used is:

Where:

* 10, is the number of bits (*N*)
* 5, is the maximum voltage the device can measure

Then we multiplied it by a range of values that we obtained from a previous harness’s simulation.

Therefore, we used the same range of max and min values for and :

The three outputs are digital outputs that represent:

* The damping coefficient that constantly changes in response to the road profile.
* The error that informs us when the sensors aren’t working properly, so when the inputs are stuck to any value for more than 20 ms.
* The blinker instead is used only as a check to make sure that the circuit is switched on.

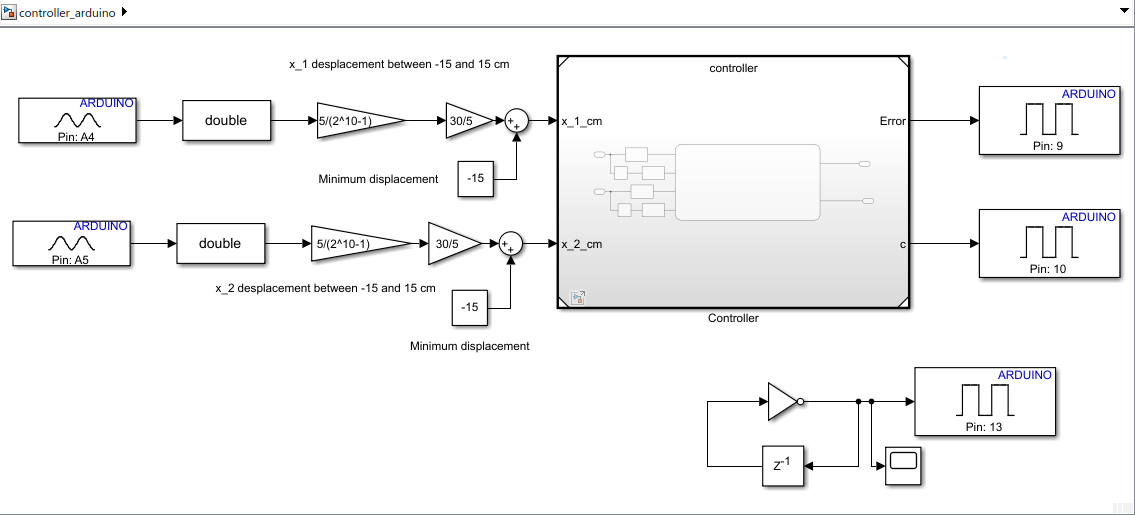


Figure 1: I/O interfaces

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Unit** | **Type[[1]](#footnote-1)** | **Conversion formulas** | **Min[[2]](#footnote-2)** | **Max** |
|  | Cm | AI |  | -0.0733 | 0.0733 |
|  | Cm | AI |  | -0.0733 | 0.0733 |
| Error | N.A | DO |  |  |  |
| Damping coefficient | N.A | DO |  |  |  |
| Blinker | N.A | DO |  |  |  |

# Code generation for Arduino

We started by setting the controller’s solver and in particular its type (Fixed-step) and solver (discrete) and finally the most important feature which is the fundamental sample time set to 0.001s.

Then we set the “code generation” features and we chose as system target file the “ert.tlc”.

Subsequentially, we went on “interface” and we select “code interface packaging” as “reusable function”.

Then on code placement we selected “compact” as file packaging format.

Finally, on “hardware implementation”, we selected as hardware board “Arduino uno”.

# Harness

# 

Figure 2: SimulIDE circuit

On SimulIDE we build the same interfaces that we had on Simulink, as inputs we have:

* The displacements and build as sinusoidal signal, through the wave generator.
* The error which is modeled as a led that lights up in red, whenever the sensors don’t work.
* The damping coefficient modeled as a led that lights up in green whenever .
* Finally, the blinker that lights up in yellow when the current is flowing.

# Test stimuli

In order to test the controller functionality, we have chosen as a stimuli test two wave generator connected to the displacements pin (X1 and X2). We have set the wave generators with two sine waves.

Immagine che contiene tavolo

Descrizione generata automaticamente

We know that the Damping is govern by the equations below:

Having as input, in the test of the controller, the displacements instead of the velocities we have doing a brief calculation in order to choose right values for the test.

We have that the displacements are describe as a function of time (t):

Where:

= amplitude of the wave

= 100 (frequency of the wave)

= 0 (phase of the wave)

Being the frequency equal to , we obtain:

If we compute the derivative respect to the time of the functions, we obtain the velocities as follow:

For a time , leave the frequency and the phase invariant and changing the value of the amplitude, we have checked the controller on SimulIDE.

## Case 1

In the figure below is reported the simulation of the case does on SimulIDE:

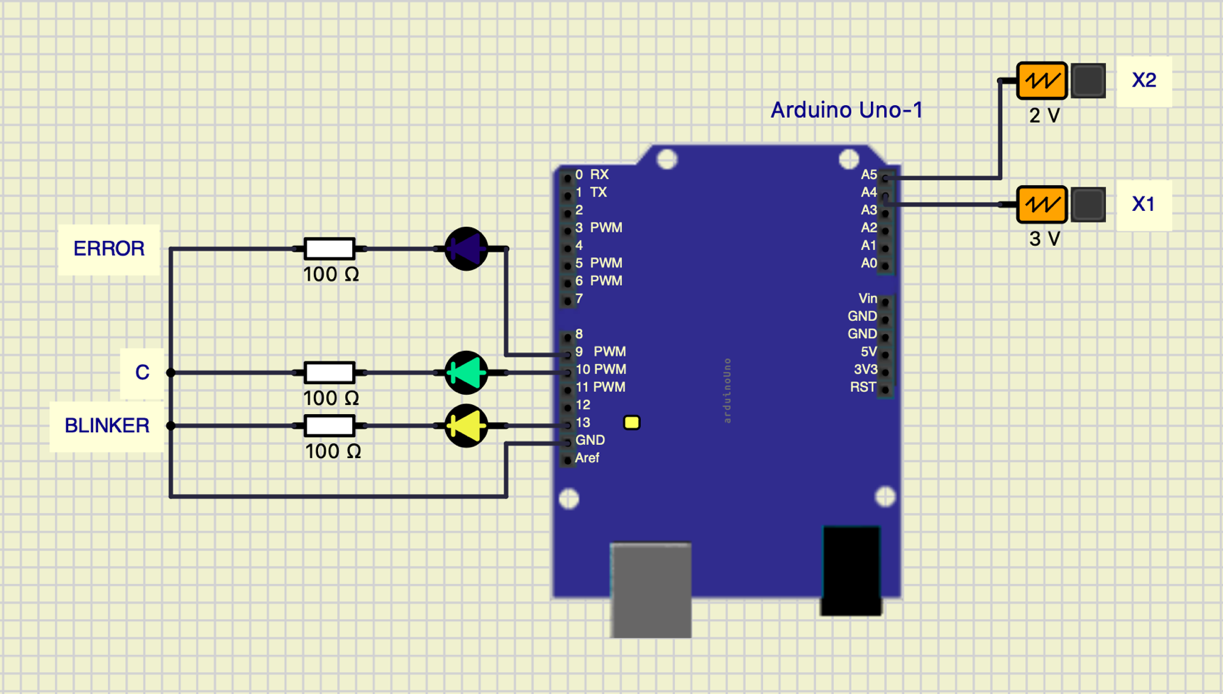


Figure 5 - case with >0

As we expected, we can see in the figure below that the led of the error is turn off because the system works properly and the led of the damping c is turn on.

## Case 2

As we can see from the simulation, in this case, we have the led of the error and of the damping are turn off but the blinker led is turns on, so the current flows and the system works properly.

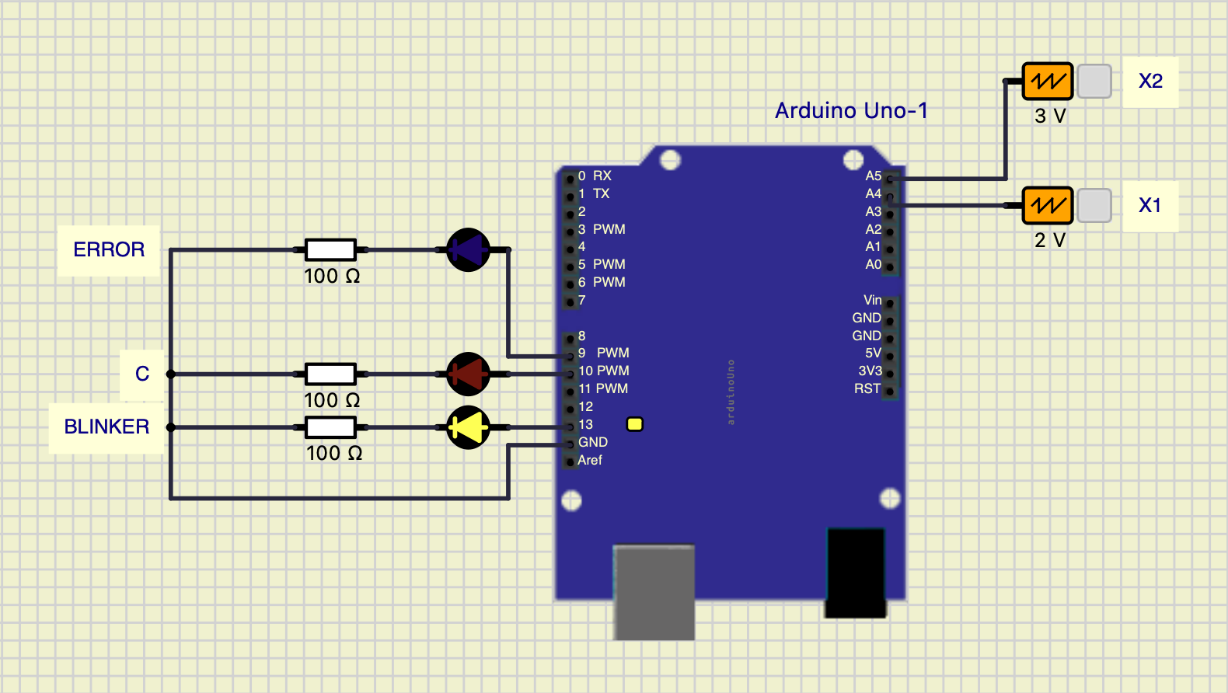


Figure 4 - case with <0

## Error case

In order to check the proper functioning of the error checker, we have run the simulation with the input of the displacements set as two square waves, with dutycycle set to 100%. We also repeated the test, setting as constant only one of the two inputs and the other one variable in time.

As we expected, in all these simulations, the error led lighted up and the damping coefficient entered the safe state, so it was set to .

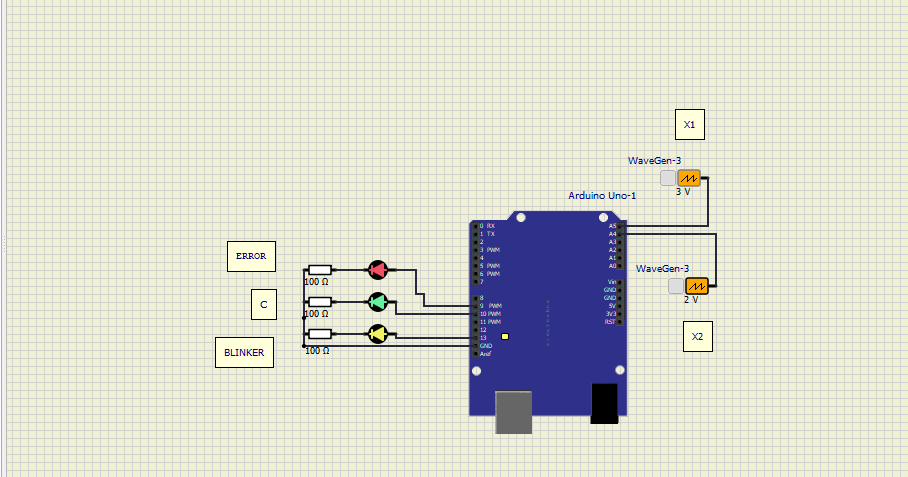


Figure 3 - Error case

1. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)