Senior Design
ENG EC 463

Memo

Team: Team 09
Date: 11/18/22
Subject: Testing Plan

Required Materials

Hardware:
- Arduino, recommend Uno or Nano (2)
- Breadboard and jumper wires
- Potentiometers (2)
- 9v power supply/USB power supply

Software
- Simulink Scripts
  - PI implementation and code-gen
- Arduino/C scripts
  - UART serial library
  - Peripheral interface

Setup

Simulink:

In Simulink we prepared a simple model based on a linear state function. There is a coefficient matrix A, which has parameters based on inputs from the environment, like tire stiffness and the distance between each tire and the center of mass. The model being linear requires some simplifications. The vehicle is modeled as if it were a two-wheeled vehicle, the only varying input from the driver being the steering angle. The angle between the fixed back wheel and the front wheel forms the steering angle δ. The coefficients $C_{y,r}$ and $C_{y,f}$ represent the cornering stiffness of the front and rear wheels, which will have to be determined for our specific car. A full set of definitions for the equations is below:

Term	Symbol	Value	Units
Yaw rate	$\dot{\psi}$	-	$[rads^{-1}]$
Longitudinal velocity	v_{x0}	[0,40]	$[ms^{-1}]$
Cornering stiffness at rear wheel	$C_{y,r}$	21429	$[Nrad^{-1}]$
Cornering stiffness at front wheel	$C_{y,f}$	15714	$[Nrad^{-1}]$
Inertia moment	I_{zz}	120	$[Kgm^2]$
Mass	m	356	$[Kg]$
Front wheelbase	l_f	0.873	$[m]$
Rear wheelbase	l_r	0.717	$[m]$
Steering angle	δ	[-3.3,3.3]	$[rad]$
Yaw moment	M_z	-	$[Nm]$
Lateral velocity	v_y	-	$[ms^{-1}]$

Figure 2: Variable names. Joao Antunes, Torque Vectoring pg. 24

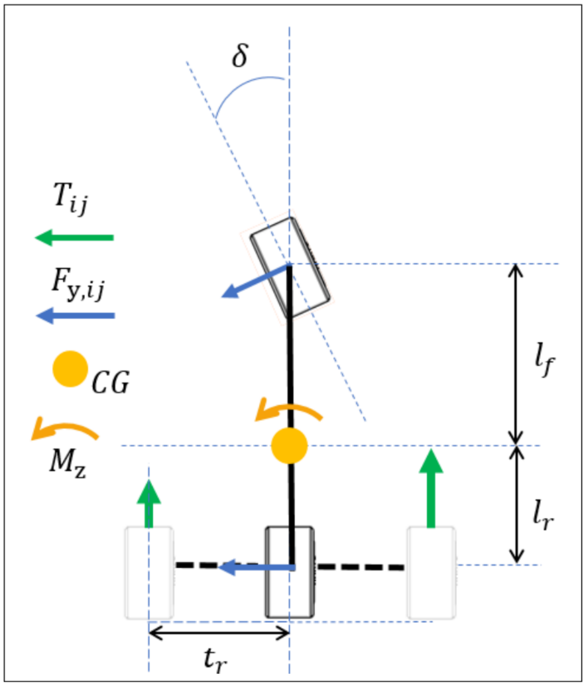

Figure 1: Bicycle model of a car. Joao Antunes, Torque Vectoring pg. 25

UART Interconnect:

We prepared a simple Arduino-based system that allows data to be read from the potentiometer sensors and transmitted over the UART communication protocol from one Arduino to the other. This emulates the connection that we will have between a remote microcontroller packaged at the front of the vehicle and the main control board towards the rear. The potentiometers represent a “throttle” and “steering” signal, which will also be coming from larger potentiometers on the real vehicle and output in the range of 0.5 to 4.5 volts.

Pre-testing setup procedure

Simulink:

Launch Simulink and run the control model.

UART interconnect:

On a breadboard, wire two potentiometers, with the output (middle) pins connected to A0 and A1 of the microcontroller. Program and upload one microcontroller with the signal transmitter code and the other microcontroller with the signal sender code. Connect the serial interfaces between the two microcontrollers once both programs are successfully compiled and uploaded, as this may interfere with programming the controller. Run three jumper wires, connecting Rx to Tx in both directions, and connecting each microcontroller’s ground to ensure signal integrity.

Testing Procedure

Simulink:

Demonstrate that Simulink CodeGen works properly and can output interpretable C-code.

UART interconnect

The inputs are the throttle and steering angle signals, which will be transmitted from the NANO to the UNO and displayed on the arduino serial monitor.

Measurable Criteria

Simulink:

C-code is generated.

UART interconnect

- Connectivity: microcontrollers should have a stable connection between them
- Noise/variation: If each potentiometer is not being touched, then the transmitted signal should not vary by more than 1%, or ~10 out of a range of 0-1023
- Throttle output to left and right is varied based on steering input