

**LOW-COST HARDWARE IN THE LOOP (HIL) TEST TOOL**

**Project Proposal**

Document ID: Low cost Hardware in The Loop Test Tool  
Origin Date: Sep 27, 2022  
Applicable to: ECE-574 Adv SW Techniques for Eng Applications Fall 2022

Student Name: Luis Castaneda-Trejo

E-mail: luisct@umich.edu

**Revision History:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version No.** | **Date** | **Details of Change** | **Modified by** |
| 0.1 | 12 Sep 2022 | Initial Version. | Luis Castaneda-Trejo |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[Introduction 4](#_Toc113913800)

[1.1 Concept 4](#_Toc113913801)

[1.2 Scope 4](#_Toc113913802)

[1.3 Learning Objectives 4](#_Toc113913803)

[Requirements 5](#_Toc113913804)

[2.1 Hardware 5](#_Toc113913807)

[2.2 Software 5](#_Toc113913808)

[Project Elements 7](#_Toc113913809)

[3.1 Hardware 7](#_Toc113913814)

[3.2 Software 8](#_Toc113913815)

# Introduction

Test and Validation teams across several automotive companies use COTS (Commercial Off-the-Shelf) technology for the design and development of Automated Test Equipment (ATE). System development programs budget high costs for professional development tools like Vector CANoe/CANalyzer, Vehicle Spy from ICS, etc.

While these tools are excellent to design and develop large simulations and tests scenarios, sometimes startup companies do not require or cannot afford these tools with high-cost licenses. On the other hand, when the test development team has finished the test automation and the rest of the R&D teams just need a simple tool to execute predefined test cases occasionally, they need to borrow at least a Runtime license of this expensive tools in order to access the test scenarios again to keep with OEM requirements for product maintenance.

## Concept

The purpose of this project is to emulate the functionality of an Automated Test Equipment (ATE) capable of running pre-defined test scenarios via CAN communication to special types of ECUs. For this academic project, a basic simulated Body Control Module will be simulated using Vector CANoe. The simulation will include vehicle engine status, Door locks, Lights and temperature sensor.

The HIL Test Tool will consist of an STM32 MCU (NUCLEO-H723ZG). The microcontroller will be running FreeRTOS. It will contain the required tasks to interact with the Device Under Test (DUT) in CANoe via CAN and to receive test and configuration instructions from a TCP client developed using NI LabVIEW.

## Scope

The HIL Test Tool will provide the ability to run predefined test scenarios to any ECU that has a CAN communication layer.

## Learning Objectives

* Apply the best software design practices for Software and Test Engineering.
* Have a state-of-the-art project template for the development of future engineering tools.
* Strengthen my ANSI C coding skills and use proper data structures.
* Strengthen my skills in CAN network simulation and knowledge with Vector tools (CANoe).
* Learn how to use the LWIP layer (Ethernet) for STM32 MCU.

.

# Requirements

This section describes the project requirements for software and hardware.



## Hardware

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Description | Comment | Status |
| HW-001 | Distance Sensor | Sensor to provide a voltage output/signal proportionally to the distance of an object. | HC-SR04 ultrasonic sensor will be used. | OK |
|  |  |  |  |  |

## Software

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Description | Comment | Status |
| SW-006 | CAN filter | System should only allow CAN id 0x726 to be processed. | A CAN id filter was set to only react to 0x726. See function Prepare\_CANFilter inside fdcan.c | OK |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Description | Comment | Status |
| SW-007 | Debounce Logic | Provide a debounce algorithm to prevent the LEDs from switching too fast when the distance falls in the middle of 2 categories. | Debounce algorithm was implemented in the controller\_handler task in app\_freertos.c | OK |
| SW-008 | Distance notification | Distance notification should be sent via CAN. | CAN message containing the distance is handled by CAN\_Tx\_Ctrlr\_handler task in app\_freertos.c | OK |
| SW-009 | Distance zones notification | Distance zone notification should be sent via CAN. | CAN message containing the distance zone is handled by CAN\_Tx\_Ctrlr\_handler task in app\_freertos.c | OK |
| SW-010 | Debug mode | System should have a debug mode reporting status of every task to a UART console | A debug macro was added to the main tasks in app\_freertos.c. | OK |

# Project Elements

This section describes the parts of the project that will be used both in hardware and software.

Chart

Description automatically generated with low confidence

**Fig 1. Main Project Elements**



## Hardware

* MCU: [STM32H7](https://www.st.com/en/evaluation-tools/nucleo-h723zg.html#overview&secondary=st_all-features_sec-nav-tab) (Cortex-M7) running FreeRTOS 10.3.1 with the following modules:
  + SPI module for TFT 1.8” screen.
  + CAN FD module configured for CAN HS.
  + Ethernet for TCP communication.
  + GPIO for user buttons and LEDs.
* CAN transceiver NXP TJA1441AT.
* Vector VN1640 CAN case to receive data from simulated vehicle sensors.

## 

## Software

* Vector CANoe 12.0
* National Instruments LabVIEW 2020.
* STMCubeIDE 1.10.1