Senior Design
ENG EC 464

Memo

To: Professors Pisano, Hirsch, Alshaykh
From: Team 9 (Giacomo Coraluppi, Alex Zhou, Will Krska, Jonathan Ye, Nick Marchuk)
Date: 3/16/2023
Subject: Second Prototype Test Report

1.0 Equipment and Setup

1.1 Equipment

Required Materials for both test bed and enclosure:

Hardware:

- 2 breadboards for controllers, steering, and misc, and their respective connections (jumper wires)
- 3 ESP32-S3 Microcontrollers
  - PID controller
  - Backup "watchdog" controller
  - Auxiliary controller
- 3 relay boards
- 8 4.7kΩ resistors
- DAC boards
- 3 DC-DC 12v to 5v converter
- 12v power supply
- Polycase enclosure
- 3D printed fittings for the enclosure
- Screws to attach backup parts to the printed fittings

Software

- Simulink Scripts
  - PID implementation on controller
- C code scripts for auxiliary, main, and watchdog controllers
  - ESP-IDF libraries
  - Configurations for UART and I2C communication

Test Bed:

The test bed is set up in a purposefully "exploded" view to demonstrate the separate functionality of each part. We have created a means of simulating the input a car would be experiencing as inputs to the system's controllers. We have also implemented a redundancy scheme, where separate hardware is constantly checking that the main controlling microcontrollers are not misbehaving.

The test bed is split into two parts, where the main controller and its relevant components are placed on one side of the board (including the PID and the watchdog controllers and relay modules), while the other side contains the auxiliary controller and inputs from the steering and throttle signals through potentiometers (we were unable to use our hall-effect rotary sensor due to previous manufacturing defect and backlog lead times for new parts).

Enclosure:

We used some of the backup materials for the hardware components to build a demonstration of the test box with the printed fittings. The fittings were all arranged in the box according to the planning assembly we created in Solidworks, and the boards were screwed into them. We verified that everything fit with the cover on, that there were no missing parts, and noted some improvements to be made for the next version.

1.2 Pre-Testing Setup Procedure

Test Bed:

Wire the auxiliary controller together by placing one of the ESP32-S3s on a breadboard and connecting the chip with two potentiometers for steering and throttle inputs and a relay module. The main controllers will consist of two ESP32-S3s connected to two relay modules and the MCP4725 DAC breakout board on another breadboard. Wire the auxiliary and main controllers together using the UART pins (in our setup, GPIO 17 and 18). Compile and upload the respective C-code for the appropriate controllers from the user's source computer to the microcontrollers. *Aux_controller.c* goes to the auxiliary controller, *main_controller.c* corresponds to the main controller, and *watchdog.c* will be for the watchdog controller. The code files should be located under "SrDes…" folders within our Github repository. Compiling and uploading the code will involve an installation of the ESP-IDF interface, followed by the following commands within the project directory folders:

(1) ". ./export.sh"

(2) “idf.py set-target esp32s3”
(3) “idf.py build”
(4) “idf.py -p [PORT] flash monitor”

Once no compilation issues are detected with the code uploading portion, enable power to the test bed microcontrollers and relay modules to verify that all components are connected and turned on properly.

Enclosure:

We loosened some of the fittings in the enclosure so that the cover could be removed for demonstration purposes. Otherwise, no setup was needed.

2.0 Measurements Taken

2.1 Testing Procedure

Test Bed:

Using a power supply machine, input voltage into the test bed system, where the microcontrollers and relay modules are turned on one by one. Use a multimeter to measure voltages at various points across the test bed. With the potentiometers on the auxiliary controller, turn them to vary the input signal and see how they affect the voltage output. At the end, increase the input signal magnitude to a point where it becomes larger than the desired design limits, triggering the watchdog controller and relay switches for system shutdown.

Enclosure:

No measurements of the enclosure were taken, barring the observation that all components fit as expected and did not compromise the watertight seal.

2.2 Measurable Criteria

Test Bed:

- Test bed should respond to user manual input from the steering and throttle potentiometers and transmit those input signals via UART from the auxiliary to the main controller.
- Steering inputs are fed through the PID controller and throttle signals go through the speed emulator logic within the main controller.

- The test bed does not overload or trigger the watchdog controller unexpectedly until an excessive amount of input voltage is applied by the user onto the system.
- When the watchdog is triggered, a series of “clicks” from the relay switches should be heard, followed by the power LEDs of the system turning off.
- All components work correctly with the proper voltage outputs/readings.

Enclosure:

- All available parts are secured in the enclosure in a reasonable fashion
- Objects in the enclosure can be removed easily.
- The enclosure is able to be shut completely.
- The printed fittings are able to be fastened to the electronic parts.

2.3 Results

Test Bed:

- All signals, voltages, inputs and outputs have proper readings upon booting up the system and manipulating the input signals.
- When the watchdog controller was triggered by excessive input voltage signals, the relay switches for each microcontroller activated with a “clicking” sound, followed by the power LEDs for each device turning off, signaling a successful deployment of the safety mechanism.

Enclosure:

- The printed fittings worked well, and fit in the box correctly.
- The enclosure was able to be disassembled readily to show the instructors what was inside and how it is being arranged.

3.0 Conclusions

Through our second prototype test, we were able to demonstrate that the testbed system worked as expected, with one big improvement being the established UART interface between the ESP32-S3 controllers, which we did not have in our first prototype test. We had three ESP32-S3 controllers (one as the main controller, one as the watchdog, and one as the auxiliary controller). Each of those is connected to a relay which shuts off power to the controller if it notices that there is a fault in the system because of excess power, for example. We were able to demonstrate how the potentiometers used for the steering and throttle signals would display the corresponding values on the screen and be fed to the other controllers via UART. When we increased the power supply to the system all the way, all three relays detected this change and shut off the controllers as expected. What was missing from our 2nd prototype test is the linear model integration into the hardware system, and the use of the accelerometers in the system, which we are still working on and aim to have completed by the end of this month for customer installation.

The enclosure met all of the criteria we were shooting for for this prototype, proving that it had the ability to contain the completed system if it was available. Now that we have a complete box with fittings we can iterate on its design and make it perfect in time for installation. We also have a wealth of experience on how everything will actually fit together, meaning the production of the second box will go smoother.