Experiment M12
Autonomous Vehicle
Procedure

Deliverables: Demonstration of your device, Design report

There is no procedure for this final lab exercise. It is an *independent* design project. The only requirement is that you begin by doing the pre-lab assignment, come to lab on Fridays, and complete the benchmark tests listed below. A final design report will be due following the final day of lab.

Project Overview

The overall goal of this project is to design and build a prototype of an autonomous vehicle or wheeled robot. The design is completely open-ended and your final design can take any form you like. However, you must use the materials and parts provided (listed on page 2), and your vehicle must complete a sequence of benchmark tests along the way (listed on page 3).

Project Rules and Guidelines

You adhere to the following guidelines.

1. Lithium-Polymer batteries must be treated with care. They can quickly start large fires if not handled properly.
    a. Students should not attempt to recharge the LiPo batteries. Leave it to the instructor.
    b. The LiPo battery must be connected to the autonomous vehicle via a 4 Amp inline fuse.
    c. Do not use a battery that appears damaged or 'puffy'. Give it to the lab instructor for proper disposal.
    d. Batteries must be stored in the fire-proof container.
2. The vehicle will be powered by a 3S LiPo battery connected in series with a 5A fuse and remote kill switch, as was done in the previous lab.
3. Vehicles must be built upon a chassis made of 2020 rail.
4. Any sharp corners or protrusions must be blunted by foam padding tape or some other method.
5. Before performing a benchmark test, you should test the design at your lab bench, with the vehicle chassis propped up on blocks so the wheel can spin freely in the air.

Parts and Materials

You must use the following materials for the project. Additional sensors, electronics, and actuators may be purchased upon request. You may also fabricate any reasonable part you wish in the EIH.

- 2020 T-slot rails of various length (for constructing chassis)
- M3, M4, and M5 T-nuts, bolts, and other hardware
- 12V DC motors, 6mm shaft, 200 RPM and 550 RPM (same ones used in M8 and M9 labs)
- Motor mounts
- BaneBots wheel hubs for 6mm motor shaft w/ retaining rings
- BaneBots wheels, black, 30A durometer (2", 2-7/8", 3-7/8", 4-7/8" diameters)
- 6mm shafts
- Pillow block bearings
- Shaft spacers
- 3S LiPo batteries
- Remote kill switches
- Cytron dual motor driver - SmartDriveDuo-10, MDDS10
- L298N dual H-bridge modules
- Arduino UNO microcontrollers
- Misc. wires and connectors
- In-line fuses
- Electrical power distribution terminals
- Masonite peg board (for mounting electronics to chassis)
- Servo motors
- Ultrasonic transducers
- BNO-055 9-axis IMU
- Foam padding tape
- 16' USB cables for testing

Project Benchmark Tests

Your wheeled robot must complete the following benchmarks in the order listed below. The TA or lab instructor will fill out your score sheet after you demonstrate each benchmark to them.

Importantly, **your design should evolve** as you move on to each new bench mark. Remember, you always should start with the simplest design and only add more complex subsystems if they are needed. For example, you may begin with just two driven wheels and a caster wheel, then add sensors as required.

The following maneuvers should be programmed using an Arduino UNO. Refer to the Cytron dual motor driver manual for information on how to interface the motor driver board with the Arduino.

Benchmark Tests:

1. The vehicle is able to drive 20 ft in a straight line going either forward or reverse.
   a. It should not veer off course by more than 1 foot.
   b. Define functions in your Arduino code driveForward(PWMvalue) and driveReverse(PWMvalue).
   c. **Optional** - Try using feedback from the BNO-055 IMU. Specifically, use the gyroscope to measure the yaw angle and yaw rate. Then, adjust the PWM signal to the left and right motors using proportional-derivative feedback. Instructions for setting up the IMU can be found in the subsequent section.
2. The vehicle is able to execute a 90 degree turn going either left or right.
   a. Define functions in your Arduino code turnLeft(angle) and turnRight(angle).
   b. Again, try using feedback from the BNO-055 IMU gyroscope.
3. The vehicle drives forward in a straight line for 4 seconds, stops for 2 second, drives in reverse for 4 seconds, stops for 2 second, and repeats.
   a. Utilize the functions in your Arduino code driveForward(PWMvalue) and driveReverse(PWMvalue);
   b. The forward and reverse motion should be as straight as possible. The vehicle should not veer off to the side.
4. The vehicle drives in a straight line and stops within 4 feet of a barrier without touching the barrier.
   a. The vehicle will begin an unknown distance between 10 – 25 feet away from the barrier.
   b. Use feedback from an ultrasonic transducer.
5. The vehicle drives in a straight line, stops within 4 feet of a barrier without touching the barrier, then turns away from the barrier and continuous moving forward.
   a. The vehicle will begin an unknown distance between 10 – 25 feet away from the barrier.
   b. Use feedback from an ultrasonic transducer.

6. The vehicle travels the entire length of the east-end hallway (outside B19) on the B-level of Fitzpatrick without touching anything.
   a. The vehicle must complete this maneuver in under 3 minutes.
7. The vehicle navigates the entire east side of the B-level of Fitzpatrick without touching anything, starting at the Lab 2 door of B14 Fitzpatrick and ending at the Lab 1 door of B14 Fitzpatrick.
   a. The vehicle must complete this maneuver in under 6 minutes.

Cytron Dual Motor Driver Board

The Cytron Dual Motor Driver Board will be used in PWM analog mode. Please note the following.

- You must adjust the DIP switches to have the correct code for PWM analog input. (See manual for more details.)
- The board must initially receive a PWM signal with 0 duty cycle to arm it. In the setup portion of your Arduino code, you should initially set the PWM signals to zero duty cycle for a second or two to arm the vehicle.

BNO-055 – Inertial Measurement Unit (IMU) – MEMS Gyroscope

The BNO-055 9-axis IMU is a digital I^2C sensor that can be used to measure the yaw angle ψ and yaw rate (angular speed) ω.

Procedure

1. Plug the BNO-055 IMU into a small breadboard. Dedicate two separate unused rows as +5V and GND bus lines. Use short jumper wires to connect the corresponding pins on the IMU to these bus lines.
2. Connect the SDA and SCL I^2C pins on the IMU to the corresponding pins on the Arduino. Use jumpers wires that are long enough to allow the BiCopter to rotate.
3. Go to the Adafruit webpage linked below, and follow the instructions for installing the Arduino driver and libraries for the BNO-055.

   https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/arduino-code
4. Download the BNO-055 9-axis IMU code from the M12 website. Connect the Arduino to the lab computer, select the appropriate COM port, and load the IMU code to the Arduino.
5. Open the "Serial Monitor" to see the output displayed as text. Make sure the Baud rate is set to 115200.
6. Slowly rotate the IMU senor about its yaw axis. You should see the angle and angular speed change for one of the three axes of rotation. Which of the six output parameters correspond to yaw and yaw rate?
7. In the main loop of the sketch, delete the IMU measurements and subsequent "serial.print()" functions for the two unused axes of rotation. Also, delete any "delay()"

functions in the main loop.

8. Modify the main loop of the code to print the time in ms, angle in degrees, and angular speed in deg/s. Save a copy of the code to your code library, and give it an intelligent file name.
    a. It should print in the format "time, angle, angular speed". This will make it easier to debug your controller.
    b. Use the millis() function to get the time and store it as a float point variable. Convert the units from milliseconds to seconds.

Clean-up

To receive full credit, you must disassemble your vehicle return the lab bench to its initial state:

- Disconnect all the wires and remove all electronics from the vehicle.
- Remove the wheels, wheel hubs, and motors. Return them to theirs appropriate bins.
- Remove the motor mounts and Masonite peg board.
- Return all screws and hardware to the appropriate compartment of the tackle box.
- Put anything that belongs in your tool kit back into the tote bin.

Deliverables

Demonstrations – You must demonstrate the various benchmark tests and have the instructor fill out you printed score sheet.

Written Report – A written design report should also be uploaded to Canvas. The report should be 4 – 8 pages in length.

1. A brief **Summary** section no longer than one page that explains the goal of the project, then lists any important performance metrics for your vehicle
2. A **Design Overview** section that explains the basic logic behind your vehicle design. It should include the following deliverables:
    a. A schematic illustrating the layout of the vehicle (similar to Fig. 1 in the M10 lab handout). It can be a simple illustration made in PowerPoint or a detailed CAD drawing. However you do it, it must be made using software, it must look professional, and it must contain labels indicating what the different parts are. It should show the where the motor, wheels, and sensors are mounted on the vehicle.
    b. A table containing a bill of materials (BOM) listing the various pieces of hardware and electronics used to construct your vehicle. Include important details, such as the wheel diameter or length of the 2020 rails for the chassis.
3. A **Results** section that discusses how well your vehicle performed. List the benchmark tests that it successfully completed.

4. A **Conclusions** section that summarizes the project, lists the shortcomings of the design, and proposes future improvements to be made.