Driving the BeyerBot

1 Collaboration Policy.

Collaboration is authorized.

Authorized Resources: You may use any electronic or hard copy resource at your disposal as long as 1) you cite your sources and 2) your use of the materials does not go against the intent of the assignment. For example, you can use a software library that you find online to help develop a project if you cite where you found it. However, you cannot complete your project by copying all of the source code, schematics, etc and simply running it on the required hardware.

2 Objectives.

1. Become familiar with the DFECBot driving functions.
2. Utilize the Arduino to drive the robot in multiple patterns.

3 Materials.

1. DFECBot
2. USB Programming Cable

4 Introduction.

4.1 Source Files

Browse to `K:\DF\DFEC\ECE210\Labs\Lab 9 - Driving the Robot` and copy the `robot_drive` folder to your computer. Open `robot_drive.ino`. Opening the *.ino* file should open 3 files in your Arduino IDE: `robot_drive.ino`, `drive.h`, and `TB6612FNG.h`. The two *.h* files are called header files and the *.ino* file is your Arduino sketch. Take a look at the `TB6612FNG.h` first.

4.1.1 TB6612FNG.h

This file uses five pins on the Arduino to communicate with the TB6612FNG Motor Driver chip on your PCB board. These pins are used as follows:

Pin 3 – Connects to the *APS_MA* pin on the PCB board. This pin sends a Pulse Width Modulation (PWM) signal to the motor driver to control the speed of left motor.

Pin 4 – Connects to the *BPS_CTRL* pin on the PCB board. This pin sends a signal to the motor driver to control the direction of the right motor.

Pin 6 – Connects to the *BPS_MA* pin on the PCB board. This pin sends a Pulse Width Modulation (PWM) signal to the motor driver to control the speed of right motor.

Pin 9 – Connects to the *APS_CTRL* pin on the PCB board. This pin sends a signal to the motor driver to control the direction of the left motor.

In the init function, all four of these pins are set to *OUTPUT* pins. Additionally, if a *reset_pin* is present it will setup the reset pin. We will not be using a *reset_pin* at this time.

The rest of the functions are used by the `drive.h` file to control the robot motors (speed and direction).

4.1.2 drive.h

This file uses the `TB6612FNG.h` header file to drive the robot. The `drive.h` file provides basic drive functions like *Motor_Forward(leftDuty, rightDuty)* and *Motor_Stop()*. All movement functions (*Motor_Forward*, *Motor_Left*, etc) will run until the next movement function is provided. For example, to get the robot to move forward for 5 seconds then stop, the *Motor_Forward(leftDuty, rightDuty)* function should be called followed by a 5 second delay and the *Moto_Stop()* function.

4.1.3 robot_drive.ino

This is the main file to place drive code. This file is reliant on both the `drive.h` and `TB6612FNG.h` header files. This file provides an example to move the robot forward for five seconds, turn right, and then stop. This code will repeat continuously until the DFECBot is powered off (See below code). Delete this example code before submission.

```
1  /*
2  *  Code to make the DFECBot go forward for five seconds, turn right, and stop.
3  *  This code will repeat continuously until the DFECBot is powered off
4  *  ***Delete this example code before submission***
5  */
6  robot.Motor_Forward(leftDuty, rightDuty);
7  delay(5000);
8  robot.Motor_Right(leftDuty, rightDuty);
9  delay(turnDelay);
10 robot.Motor_Stop();
11 delay(1000);
```

5 Procedure

Use the example code provided, `TB6612FNG.H`, and `drive.h` to code the DFECBot to drive the following patterns. The DFECBot should pause for 2 seconds between each pattern.

1. Drive the DFECBot forward.
   (a) Drive forward for 5 seconds.
2. Drive the DFECBot forward and reverse.
   (a) Drive forward for 5 seconds.
   (b) Pause for 1 second.
   (c) Drive in reverse for 5 seconds to return to the starting position.
3. Drive the DFECBot forward, turn around, and return to start.
   (a) Drive forward for 5 seconds.
   (b) Pause for 1 second.
   (c) Turn around.
   (d) Drive back to start and original orientation.
4. Drive the DFECBot in a square.
   (a) Drive in a square making right turns.
   (b) Return to the starting location and position.
   (c) Drive in a square making left turns.
   (d) Return to the starting location and position.
5. Drive the DFECBot in a small circle.
   (a) Drive in a clockwise circle keeping one wheel fixed.
   (b) Drive in a counter clockwise circle keeping one wheel fixed.
6. Drive the DFECBot in a large circle.
   (a) Drive in a clockwise circle with an approximate diameter of 2 feet.
   (b) Drive in a counter clockwise circle with an approximate diameter of 2 feet.
7. Drive the DFECBot in a pattern of choice.
   (a) Drive in a pattern of your choosing.