Experiment #3 – Advanced Motion

Objective:

In experiment #2 you learned, programmed, and calibrated the six basic motions. The objective of experiment #3 is to add advanced motion techniques to your basic motion library. You will write and test functions to perform 45 degree, arcing and pivot motions. Once the functions are written and tested, you will calibrate them so they may be called with meaningful numbers.

Advanced motion of mecanum wheel systems involves driving pairs of motors at different speeds or simply driving one pair at a time. The latter is the case for 45 degree motion and the former is the case for moving in arcs or pivots. Driving a single pair of motors is relatively simple. Driving two different pairs of motors simultaneously at different speeds is a more complex algorithm.

Preliminary setup

The motion of a single pair of motors is accomplished with the same PORTL toggle command with a different motion byte. Rather than toggle all four motors at once you only toggle two at a time. Expand your global variables by adding the following declarations before setup() {.

Using the results from experiment you must map the pairs. The are six sets of pairs, left motors, right motors, front motors, rear motors, left diagonal motors, right diagonal motors. In lab 2 you used –

```
byte motion = B01010101;
```

to turn all four motors simultaneously. You must now right the six motion bytes for motor pairs

```
byte leftmotion = B????????,
byte rightmotion = B????????,
byte frontmotion = B????????,
byte rearmotion = B????????,
byte motion45right = B????????,
byte motion45left = B????????.
```

Functions

In experiment #1 you found that you could turn the motors by using a loop to repetitively toggle the 4 motion pins of PORTL. To develop a 45 degree motion function copy the code for the forward motion function and simply change the PORTL toggle command. Ex.

```
PORTL = forward;
For (i=0; i< steps; i++){
PORTL ^= motion45Right;          //Port toggle command
                              }
```

You can change the direction from forward to reverse and the 45 to left or right to generate diagonal motion in each of the four directions. The angle of the diagonal can be controlled by driving the two sets of diagonal motors at different speeds. 45 degrees is the degenerate state of motion when one of the pairs is stopped.

There are exactly 3 ways that four motors can be divided into pairs. Each different pairing produces a unique motion. The simplest examples to study are the degenerate cases when one set of motors is off. We have already seen that running one diagonal pair of motors in a forward or reverse direction produces 45 degree motion. Similarly running the motors on one side in fwd/rev or rotate left or right produces an arc motion and running the front or rear pair in a rotating direction produces a pivoting motion.

Procedure

1. Program all preliminary global and setup variables
2. Write functions one at a time, start with forward motion
    a. 45 degree motion – fwd/rev right/left
    b. Arc motion – fwd/rev right/left
    c. Pivot motion – cw/ccw front/rear
3. Debug and test one at a time
4. Calibrate one at a time
5. Repeat steps 2-4 for each function

Calibration

No calibration required. Set 45 degree motion for 12 inch distance and verify. Set the arc and pivot motions for a 90 degree distance. Start arc and pivot tests with a 90 degree command and adjust the counts until the robot arcs/pivots 90 degrees. Measure the radius of curvature

Testing

Program, edit, compile, and download you program onto the robot. Place the robot on the calibration board under the blackboard. Align the robot to the painted gridlines and turn the robot on. The painted lines are on 12 inch centers squared at 90 degrees within 1/8th inch overall. You can align the chassis or wheels to the lines. For straight line motion distance measurement the chassis is the most reliable datum. For rotation the wheels are the most usable indicators of alignment because of their visibility.

First verify that a 12 inch commend will move the robot 1 full grid forward and sideways. Adjust if necessary. Next program the “arc’ motion and give it the turns for a 90 degree turn. Adjust the number of steps until a 90 degree arc is obtained and measure the radius of curvature of the motion. Repeat the arc process with the pivot command and measure the radius of curvature. You will need these numbers later.