

Vehicle of Revolution: How many turns will it take...?

A SMART project; Funded by the National Science Foundation

Polytechnic University – Mechatronics Department

Headed by Professor Vikram Kapila

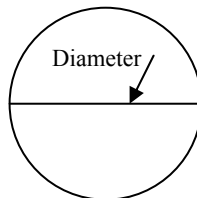
Group 2: Michael Fishenfeld, Henry Penna and Douraine Stewart

Question: *If a 6 inch wheel rolls down a 4 foot track, how many revolutions will the wheel make to get to the end of the track?*

This is a typical question that is asked on mathematics exams and Regents, yet many students get it wrong. Teachers take it for granted that since finding the circumference of a circle is an easy lesson to teach, no physical application is used as a visual aid to help explain the topic. This generation consists of young people that learn more when they're senses are engaged. Therefore, the Vehicle of Revolution will bring to life calculating distances in this manner.

Background

A circle is a locus of points equidistant from the center of the circle. The diameter of a wheel is the distance from one endpoint of the wheel to the very opposite endpoint through the center of the wheel.

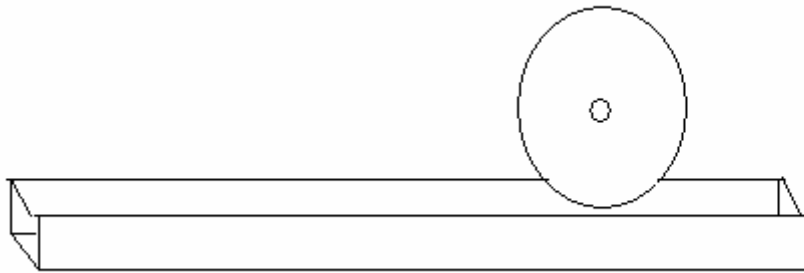


So a circle consists of an infinite number of diameters. The distance around the circle is the circumference. Let's make a connection here. If a circle is made up of diameters and the circumference is the distance around the circle, then the ratio of the circumference to

the diameter of a circle gives an irrational constant, π . Therefore, to calculate the circumference of a circle can be found using this formula:

$$C = d \cdot \pi \quad (\pi \approx 3.14)$$

Now the problem stated that there is a 6 inch wheel. Therefore, the circumference of the wheel is approximately $6 (3.14) = 18.84$ inches. That is like cutting a piece of string 18.84 inches long. The second part of the problem asks “how many revolutions will the wheel make to get down the track?”



If the circumference of a circle with a diameter of 6 inches is 18.84 inches, then the distance of the track needs to be converted from feet into inches.

$$4 \text{ feet} = 12 \text{ inches (per foot)} \cdot 4 = 48 \text{ inches}$$

$$48 \text{ inches} \div 18.84 \text{ inches} \approx 2.5 \text{ revolutions}$$

So we can conclude that a number of revolutions can be determined by:

$$\text{Revolutions} = \text{distance} \div \text{circumference}$$

It's a matter of proportions. If one revolution is equal to m units then n revolutions is equal to $n \cdot m$ units.

Purpose and Mission

Students tend to estimate linear distances correctly. It is a challenge, however to properly observe and estimate circular distances. They lack the intuitive understanding of the relationship between the perimeter of polygons and the circumference of circles. Engaging in experimenting with these variables while getting immediate responses (feedback) and correcting the variables to obtain satisfactory answers should help in developing their understanding of these relations.

The purpose of this project is to show that measuring circular distances is isomorphic to measuring linear distances. Our mission is to create an interactive project that will bring to life a concept that students take for granted as being simple yet make mistakes with quite often.

How Does This Project Work

The objective of the project is to build a preprogrammed model that will roll or drive a certain distance. The target at the end of the track is movable and can be set at a desired distance from the starting point. A student will choose a set of wheels to put on the back of the car. With the diameter of the wheel, the student must determine the number of revolutions that the car will make to get to the target. A student will enter that prediction into the model vehicle's display screen and then observe moving vehicle. If correct, then that is great. However if it is wrong then one is forced to backtrack and find out why. The vehicle is driven by a microcontroller that repeats the experiment continuously with different size wheels and different distances.

The Model Vehicle

The car has a simple design to avoid spending too much time on how it looks but to bring the focus on the effectiveness of the goal. The current vehicle represents the final evolution of the 3 different designs preceding it.

- The material finally decided upon was Lexan as opposed to Plexiglass, because Lexan is easier to machine and much more resistant to cracking and breaking. The downside of Lexan is, of course, that it is subject to scratching.
- The final design has a DC motor and an H-bridge to drive it as the two preceding motors (a pair of servo motors and a stepper motor) were found unsatisfactory. The servo motors have to be programmed in such way so that fractional turns could be recognized. However, the conversions would consist of numbers greater than 2^{16} which the Basic Stamp will not recognize. The stepper motor would have made the programming a breeze but the motor that we had did not have enough torque to move the vehicle.
- Because the vehicle is expected to move in a straight line, both rear wheels had to turn under power. This necessitated a driven axle, so two gears—one in the motor and one in the axle—were meshed together to achieve two-wheel drive. Later on, a third gear was added to drive the potentiometer of the DC motor.
- Because the experiment revolved around the different size drive wheels, the drive wheels had to come off and on easily. This resulted in the axles' being threaded

on each end so that the wheels could be changed. Rubber grommets were fashioned to hubs to make contact with the wheels to ensure both wheels' turning equally with the axle. A lack of flanges resulted in the machining of those pieces being out of stock. Set screws hold all flanges to their respective shafts.

- The front four wheel dolly was constructed so that the rear portion of the body could have different size drive wheels attached and the dolly portion of the car would always be parallel to the track.
- The final design has an extended body in order to hold an onboard battery.
- The track is made of plywood, reinforced with 1" x 2" strips, and lined with plastic to help keep the car on a straight line. A tape measure was secured to the side of the track so that measurements could be taken.

Programming

The program was written along while building the components of the model and experimenting with their performance. Many parts tested were not proper for the model or were not available from suppliers and had to be replaced. Rewriting the program and redesigning the main construction followed that. We constructed a working prototype of a vehicle to begin experimenting and programming. In the meanwhile the main model was reconstructed to fit available parts. Similarly, the circuitry was tested with new components and redesigned again.

The format of the program is to simply ask the student to enter their prediction, using the buttons they are allowed to do so and the vehicle will turn the wheels that

number of times entered. This is the final program that the Basic Stamp will use to control the vehicle:

```
' {$STAMP BS2}
' {$PBASIC 2.5}

`This is the list of variables and assignments

btnwrk VAR Word
btnwrk=0
btn CON 0
txpin CON 7
baud19200 CON 32
c1 VAR Byte
c2 VAR Byte
turns VAR Word
turns2 VAR Word
pot VAR Word

`This is the main program

GOTO home_pos

main:

`The screen gets ready to receive the input

SEROUT txpin, baud19200,[12]
PAUSE 5
SEROUT txpin, baud19200,[24]
SEROUT txpin, baud19200,["Enter the number of revolutions"]
PAUSE 1000
SEROUT txpin, baud19200,[12]
PAUSE 5
DEBUG HOME, "start"
PAUSE 500
DEBUG CLS
c1=0
c2=0

`Buttons here are programmed to allow one to enter their answer

DO
  DEBUG HOME
  BUTTON btn,0,200,20,btnwrk,0,no_press1
  IF c1>8 THEN c1=0
  c1 = c1 + 1
  SEROUT txpin, baud19200,[c1+48]
  SEROUT txpin, baud19200, [8]
  DEBUG DEC c1

  no_press1:
  IF IN1=1 THEN EXIT

LOOP
```

```
SEROUT txpin, baud19200,[9]
SEROUT txpin, baud19200, ["."]
```

DO

```
DEBUG HOME
BUTTON btn,0,200,20,btnwrk,0,no_press2
IF c2>9 THEN c2=0
c2=c2 + 1
SEROUT txpin,baud19200, [c2+47]
SEROUT txpin,baud19200, [8]
DEBUG DEC2 c1*10 + c2
no_press2:
IF IN2=1 THEN EXIT
```

LOOP

```
PAUSE 500
SEROUT txpin, baud19200,[13]
SEROUT txpin, baud19200, [12]
SEROUT txpin,baud19200, ["Is this number correct? ", c1+48,".",
c2+47]
DEBUG HOME,CLS, "Is this number correct? ", DEC c1,".", DEC c2
```

confirmation:

```
IF IN2=1 THEN GOTO no
IF IN1=1 THEN GOTO yes
GOTO confirmation
```

`These subroutines allow the person to decide to move on or go back and re-enter an answer

yes:

```
SEROUT txpin, baud19200, [" yes"]
DEBUG HOME, CLS, "yes"
PAUSE 1000
GOTO rotate
```

no:

```
SEROUT txpin, baud19200, [" no"]
PAUSE 500
GOTO main
```

`Conversion process starts here before the car moves. The "55" is the number of steps per turn. Also remember BS2 does not recognize decimals so turns is a formula to convert the answer into a whole number.

rotate:

```
turns= c1*10 + (c2-1)
'SEROUT txpin,baud19200, [12]
' SEROUT txpin,baud19200, [ DEC turns]
turns2= (turns * 55)/ 10
turns2 = pot-turns2
SEROUT txpin,baud19200, [12]
SEROUT txpin,baud19200,[DEC turns2]
```

```

    DEBUG HOME, CLS , "turns2 ", DEC turns2
    PAUSE 1000
GOTO move_motor

`This is the "driving" routine

move_motor:

    'motor set high b for forward
    'motor set high a for backward

    HIGH 6          'pin 0 is line 15 on breadboard
    PAUSE 1
    RCTIME 6,1,pot
    DEBUG CLS, DEC pot, CR
    'PAUSE 2000
    IF (pot > 527) OR (pot < 19) THEN stopmoving
    IF pot > (turns2 + 3) THEN forward
    IF pot < (turns2 - 3) THEN backward
    IF (turns2 - 3) < pot < (turns2 + 3) THEN stopmoving
    IF IN2=1 THEN home_pos
GOTO move_motor

forward:
HIGH 4
LOW 3
GOTO move_motor

backward:
LOW 4
HIGH 3
GOTO move_motor

stopmoving:
LOW 4
LOW 3
GOTO move_motor

home_pos:
HIGH 6          'pin 0 is line 15 on breadboard
PAUSE 1
RCTIME 6,1,pot
DEBUG HOME, DEC pot
IF pot >520 THEN stop2
IF pot <520 THEN backwards2
GOTO home_pos

backwards2:
LOW 4
HIGH 3
GOTO home_pos

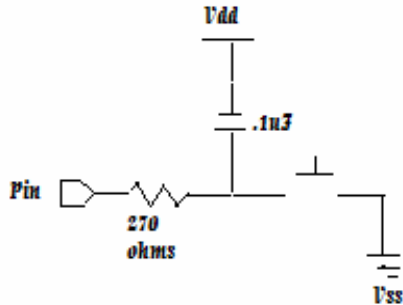
stop2:
LOW 4
LOW 3
GOTO main

```

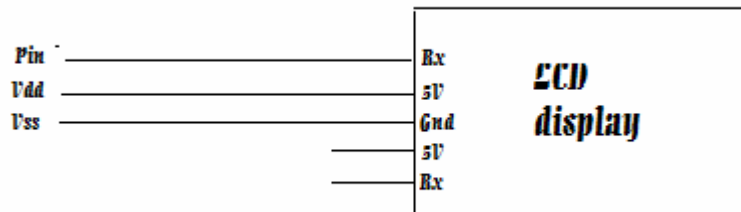

Circuitry

The following schematics were used to program the different parts of the vehicle.

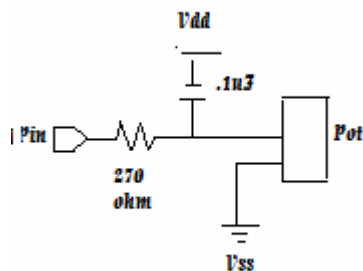
1. Buttons- There are three of them: one for increasing the number from 0 – 9, the other two are confirmation buttons, yes and no. They are pins 1, 2 and 3.



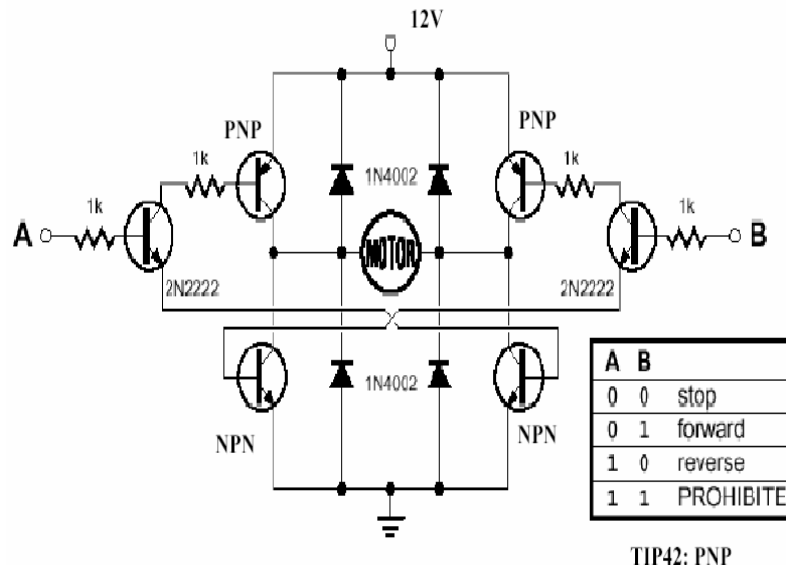
2. LCD Display- This is the interactive screen on the vehicle. This is pin 6.



3. The Potentiometer- This is the part that counts the number of turns. This is pin 0.



4. The H-bridge – This is the circuit that controls the driving of the vehicle. The chart shows how the circuit controls the vehicle based on programming commands. They are pins 4 and 5.



Observations

When building the project, our initial idea was that the vehicle would turn the number of revolutions that was entered with great precision, at least within $\pm 0.25\%$. Along the way, we encountered several obstacles which brought us to the model that we now have. First in calibrating the DC motor, we realized that the potentiometer, with a maximum of 10 full turns, did not have the same number of steps for the first turn and the last turn. All other turns had the same number of steps. So we needed an average to compensate for that which enlarges the percent error. However, we were able to come up with a number that gave us complete turns on average give or take a point or two. Also the H-bridge is to control the moving the motor but there is no way to control accurate breaking. It takes time for the motor, like a regular car, to slow down to a complete stop. So we needed some inequalities in the program so that the potentiometer can check its

position and adjust itself to the position closest to where it should stop. This gives the motor some time to slow down and come to a complete stop. Also there is a little backlash of the gears slipping. So after a few runs the gears have to be checked to make sure that they are tight and in place.

Future Enhancements

As of right now the vehicle is at its basic model stage. However it is built in such a way that we can go back to make upgrades. Later we will add LEDs, a piezospeaker and relay switches on the vehicle and track to let the student know that he/she is right or wrong. It will have the added component to calculate the right answer and correct the student. Also two markers can be set on the track and sensors on the car so that when the car passes the markers, it would recognize the distance traveled and display the circumference of the wheel as well as the diameter. Considering that a stepper motor would yield the best results, I could consider using that motor with the right controller to compare the accuracy of a DC motor to a stepper motor.