THE SMART RESONANCE TUBE

By
Robert Morris (Clarkstown High School North)
&
Joe Rodichok (Smithtown High School)

INTRODUCTION:

A common laboratory experience encountered by students in a first year physics courses in either high school or college, is an experiment which utilizes the concept of resonance to compute the speed of sound in air. Commonly, students take a tube, open at both ends, and submerge one end in water. After striking a tuning fork just above the tube, students slowly vary the length of the tube by raising both the tuning fork and the partially submerged tube, until a loud hum is heard emanating from the top of tube. When this occurs a standing wave pattern is set up inside of the tube, one quarter of a standing wave to be exact. With knowledge of the length which causes this phenomenon, and the frequency used, students can easily compute the speed of sound. However, this experiment includes one major limitation, an overwhelming amount of inaccuracy in recording the length. This is mostly due to movement of the tube after the resonance occurs, but before the measurement has been made. This is further compounded by parallax, that is students observing erroneous length values due to their position above the actual end point of the tube. This leads to a rather large percentage of error and often nullifies the worth of conducting such an experiment. The goal of our apparatus is to fully automate this experiment using a microprocessor, a DC Motor, position and sound sensors, a frequency generator, and a seven segment display. The hope is to eliminate human error, and allow the user to obtain a much more accurate number for the speed of sound in air.

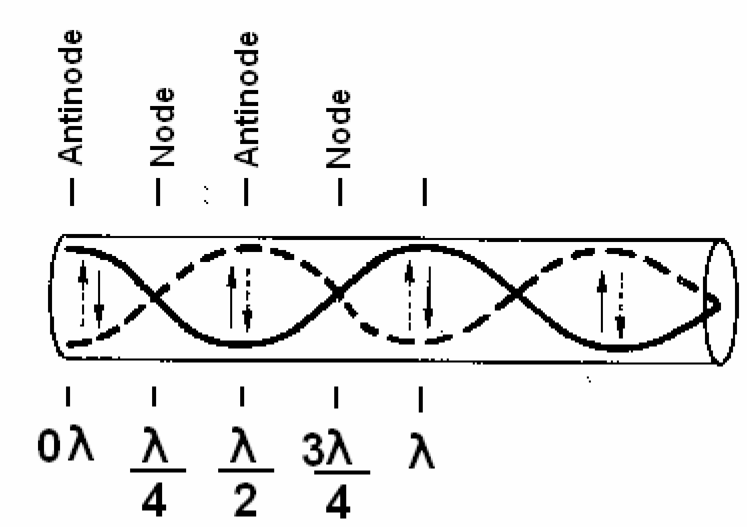
BACKGROUND:

What is resonance, and how can a closed tube be used in conjunction with resonance to compute the speed of sound in air.

All objects have a distinct set of natural frequencies of vibration. These natural frequencies of vibration can be varied by changing certain properties of these objects. When forces are applied to these objects at one of the natural frequencies of vibration, standing wave patterns are produced. In the case of a tube which is closed on one end, these natural frequencies of vibration can be changed by varying the tube's length. Since standing sound waves cannot be seen, one must use the hearing sense to determine when such a standing wave has been set up.

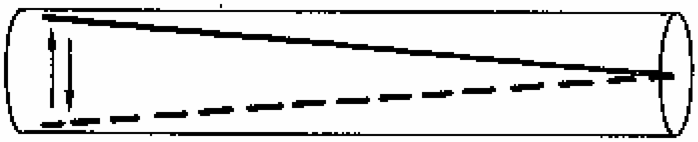
The following facts apply to standing waves, specifically standing waves set up in closed tubes:

- Standing waves are produced when two waves with the same amplitude and wavelength are passing through each other in opposite directions in the same medium. The picture below depicts a standing wave pattern, and fractions of standing waves noted:



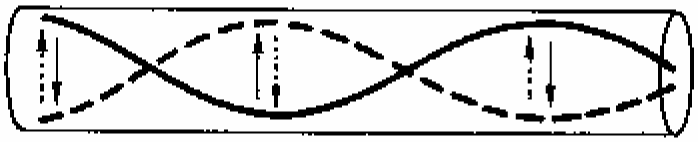
At the extreme left, “0λ” indicates the position of an antinode. This is the position where the wave moving right and the wave reflected by the right end of the tube and moving left constructively interfere. This is the location of a large volume of sound.

- The first harmonic is the frequency which must be applied to produce the first standing wave pattern. In closed tubes, the first standing wave pattern happens to be only one quarter of a standing wave. The pictures below note the standing wave patterns set up in a closed tube when the first harmonic, second harmonic, third harmonic, and fourth harmonic frequencies are applied to the tube.

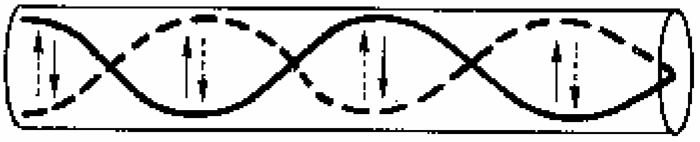


Standing wave pattern at first harmonic frequency (f_1)

Standing wave pattern at second harmonic frequency (f_3)



Standing wave pattern at third harmonic frequency (f_5)



Standing wave pattern at fourth harmonic frequency (f_7)

Note the first standing wave pattern is really ¼ of a standing wave. The second pattern shows ¾, the third shows 1 ¼, etc. In other words, closed resonance tubes produce standing wave patterns starting with ¼ and increasing by halves.

- Changing the length of the tube changes the natural frequencies of vibration of the tube including the first harmonic frequency. The equation below relates the length of the tube to the required first harmonic frequency and is explained further below.

$$f_1 = [V] / [4(L+0.4d)]$$

"f_1" represents the first harmonic frequency
"V" is the speed of sound in air at room temperature
"L" refers to the length of the tube that is required to support the first standing wave pattern
"d" refers to the diameter of the tube

If a specific frequency is sounded at the end of the closed tube, and the length slowly increased from zero until a loud volume of sound is heard emanating from the tube, the standing wave pattern associated with the first harmonic frequency has been achieved. This means inside of the tube of length "L" ¼ of a standing wave has been set up. Since L and λ are both lengths it is logical to equate:

$$L = \lambda/4$$

OR

$$\lambda = 4L$$

Since **V = fλ,** one can make the above substitution for λ and arrive at:

$$V = f(4L)$$

However, in reality, the true position of this antinode occurs at a distance above the top of the tube equal to 4/10 of its diameter. So really the effective length of the tube becomes L + 0.4d. After replacing the actual length with this, we arrive at the following equation:

$$V = f(4[L + 0.4d])$$

OR

$$f = V/(4[L + 0.4d])$$

If the length and diameter of the tube have been measured in meters, while knowing the frequency of the sound source used, one can use the equation above to compute the speed of sound in air.

EQUIPMENT LIST:

Diagram #1

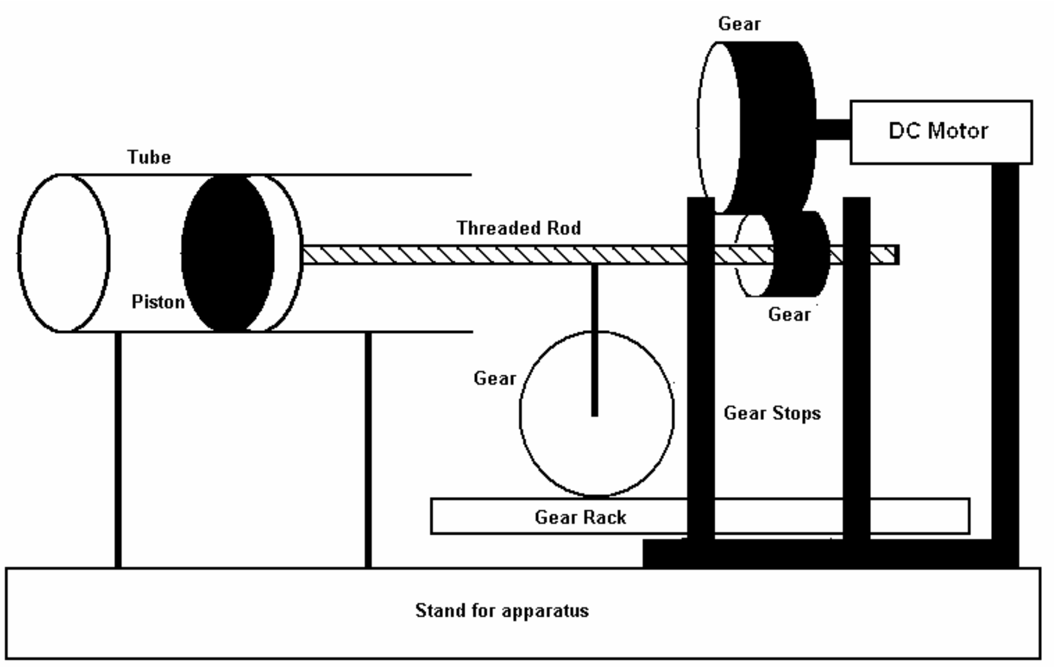


Diagram #2

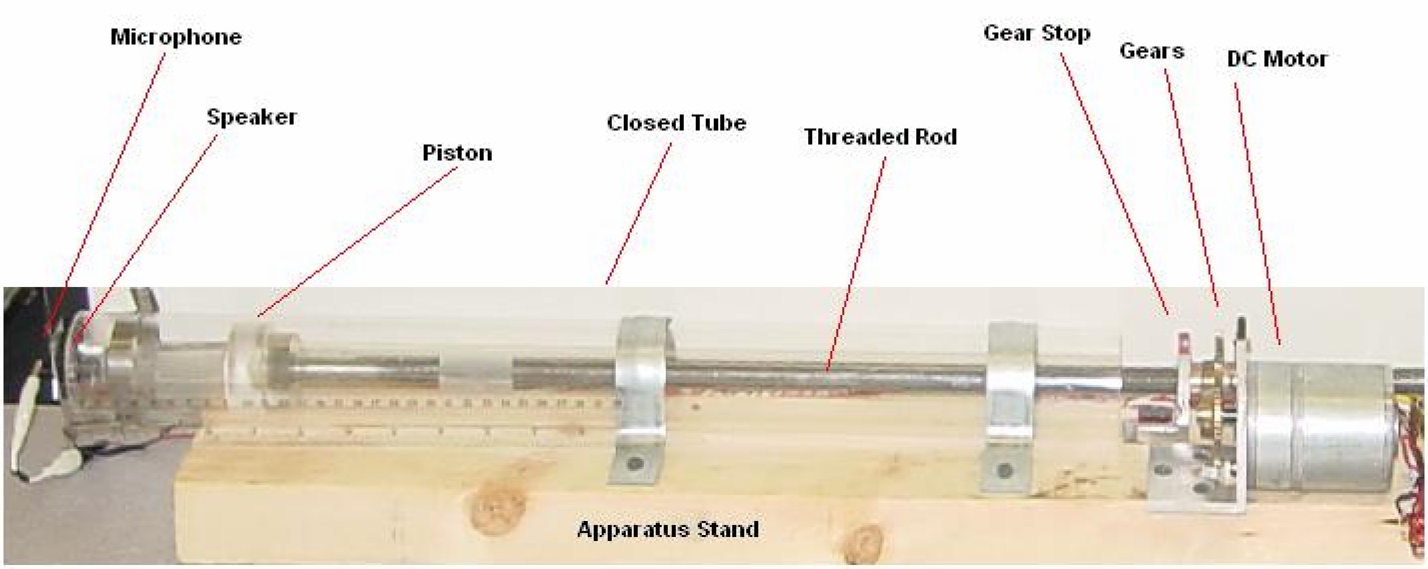


Diagram #3

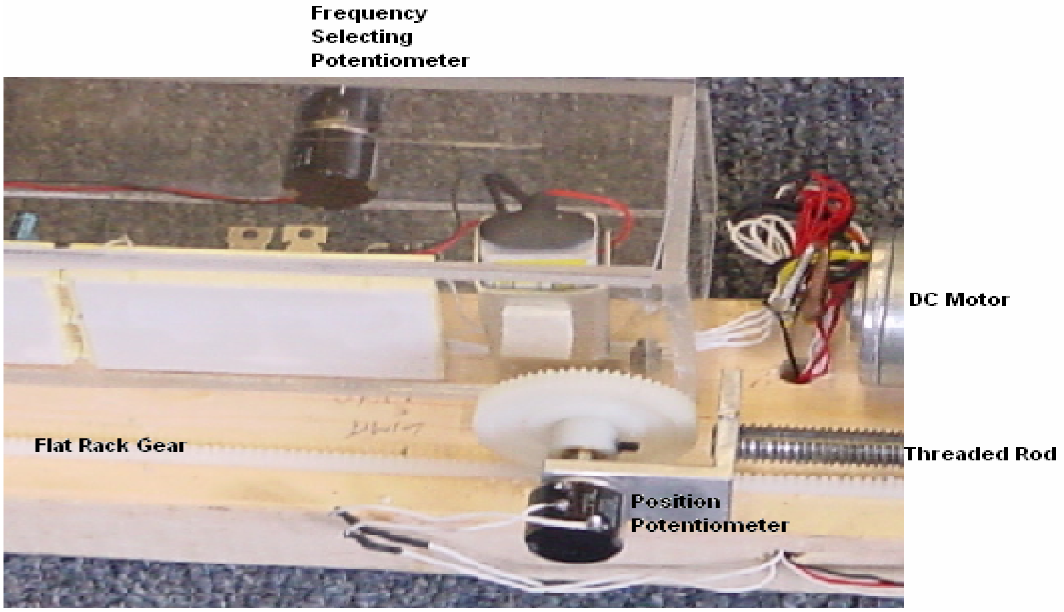
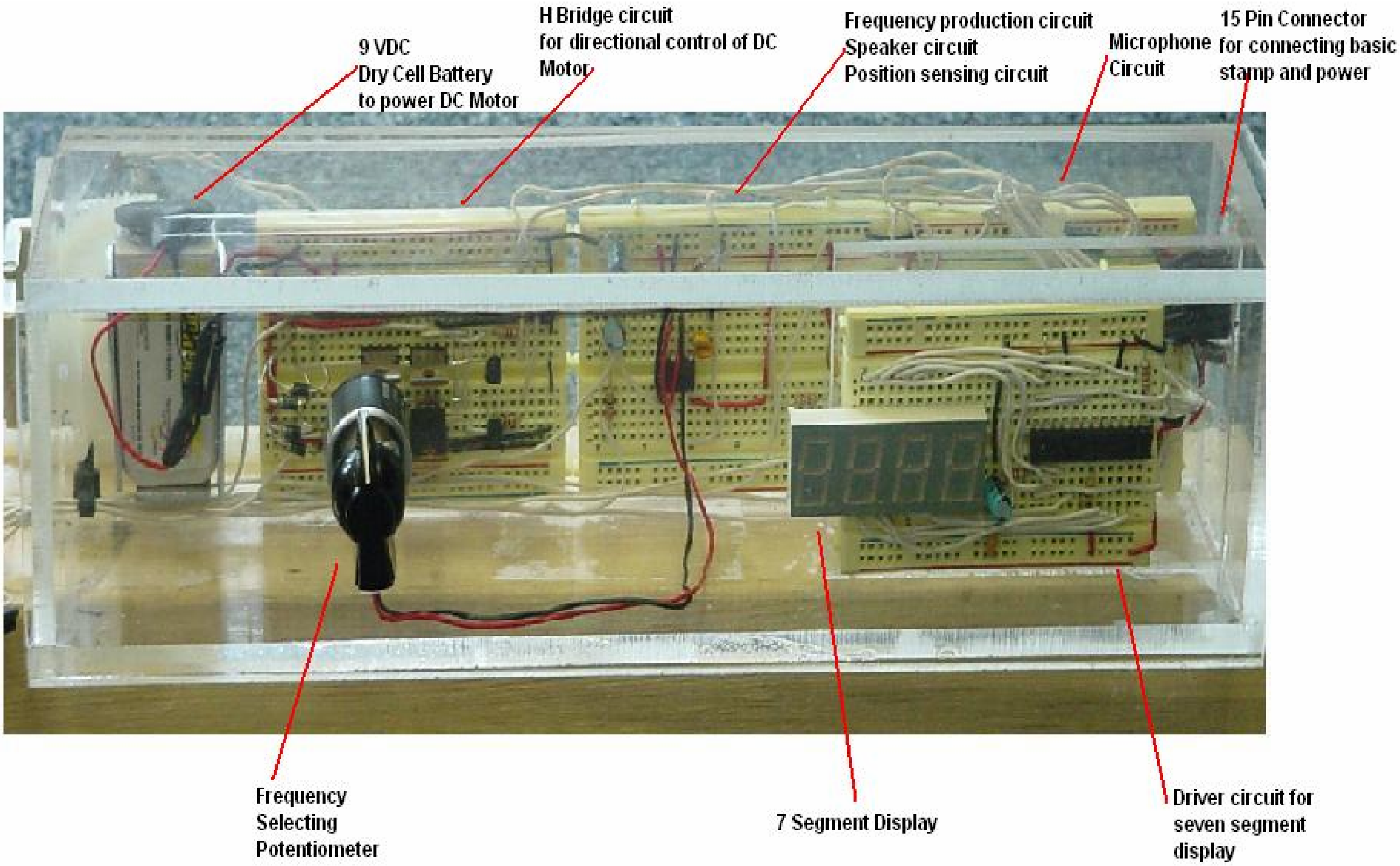


Diagram #4



EXPERIEMENTAL PROCEDURE:

Speed of Sound Experiment

Goals: To determine the speed of sound in air using resonance.

1-Connect the BOE, which contains the BS2, to the circuits on the apparatus using the 15 pin serial connector. Connect the BOE to the computer with a serial cable.

2-Download the Resonance.BS2 file to the BS2.

3-As soon as the DC Motor is heard, quickly dial any frequency from 614 Hz – 1500 Hz by using the dial and seven segment display located on the right hand side of the apparatus.

4-The BS2 will begin by resetting the experiment. This is done by first moving the closed end of the tube until a 0 meters tube length is presented to the speaker and microphone. Note, the closed end of the tube is the clear piston within the tube connected to the threaded rod.

5-Next, the BS2 will slowly increase the tube length. As the tube is lengthened, a rise in the volume will be observed. The BS2 will continue to increase the tube length until the volume just begins to decrease. At this point, the motor will reverse the direction of the piston until this maximum volume is located. At this point the BS2 has found the exact tube length which will support the first standing wave pattern. This means that the frequency being used corresponds to the first harmonic frequency of the closed tube.

6-Attached to the side of the tube is a ruler. If one measures the distance in meters from the piston to the open end of the tube, the variable of length can be numerically defined.

7- As suggested earlier, the true length is really the recorded length plus 0.4 x (tube diameter).

8-If this new length is multiplied by the frequency displayed on the 7 segment display, one can effectively arrive at the speed of sound in air.

9-One might further the experiment by deriving an accepted value for the speed of sound in air and finding the percent difference between this number and the number found in procedure step #8. The equation below finds the speed of sound using room temperature in degrees Celsius.

V accepted = 331m/s + 0.6T

Where:

V accepted is the speed of sound in air using temperature

T is room temperature in degrees Celsius

What follows is a suggested data chart, which could be used to compile the necessary data to complete the above experiment:

FREQUECY (Hertz)	**Tube Length (meters)**	**Diameter (meters)**	**Wavelength = 4[L + 0/4d] (meters)**	**Temperature (^oC)**	**Experimental Velocity ($V = f\lambda$) (m/s)**	**Accepted Velocity (331m/s + 0.6T) (m/s)**	**% Difference**

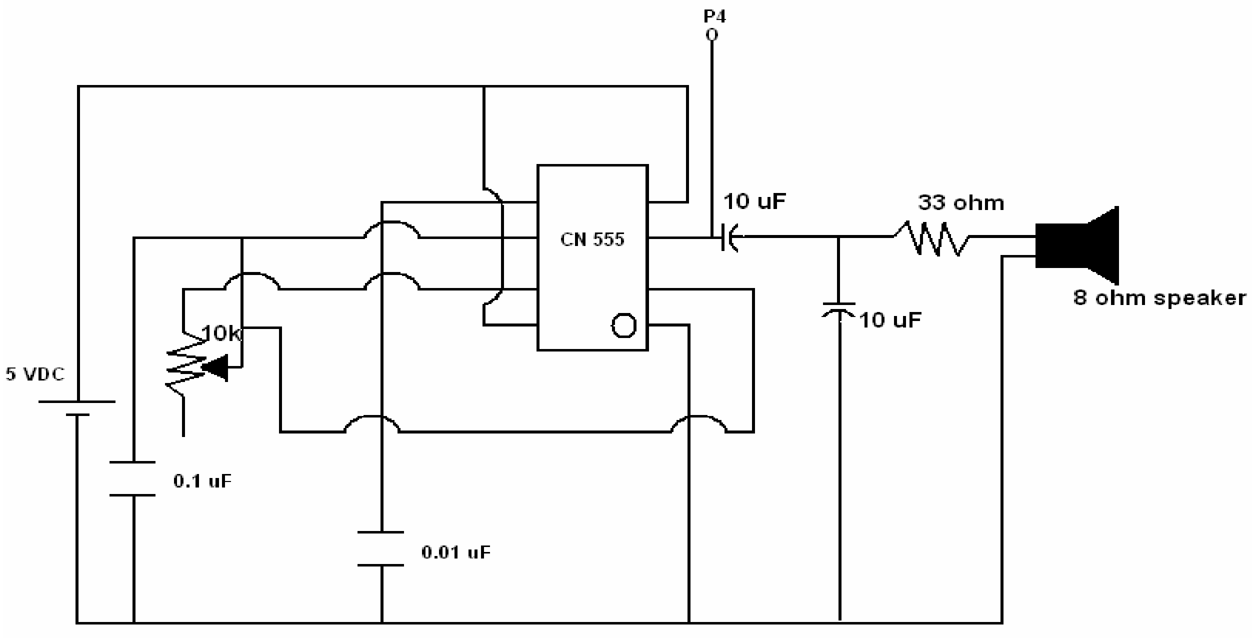
REFERENCES:

1- Online: Online: http://www.parallax.com, web site of Parallax, Inc. manufacturer and distributor of Basic Stamp microcontrollers.

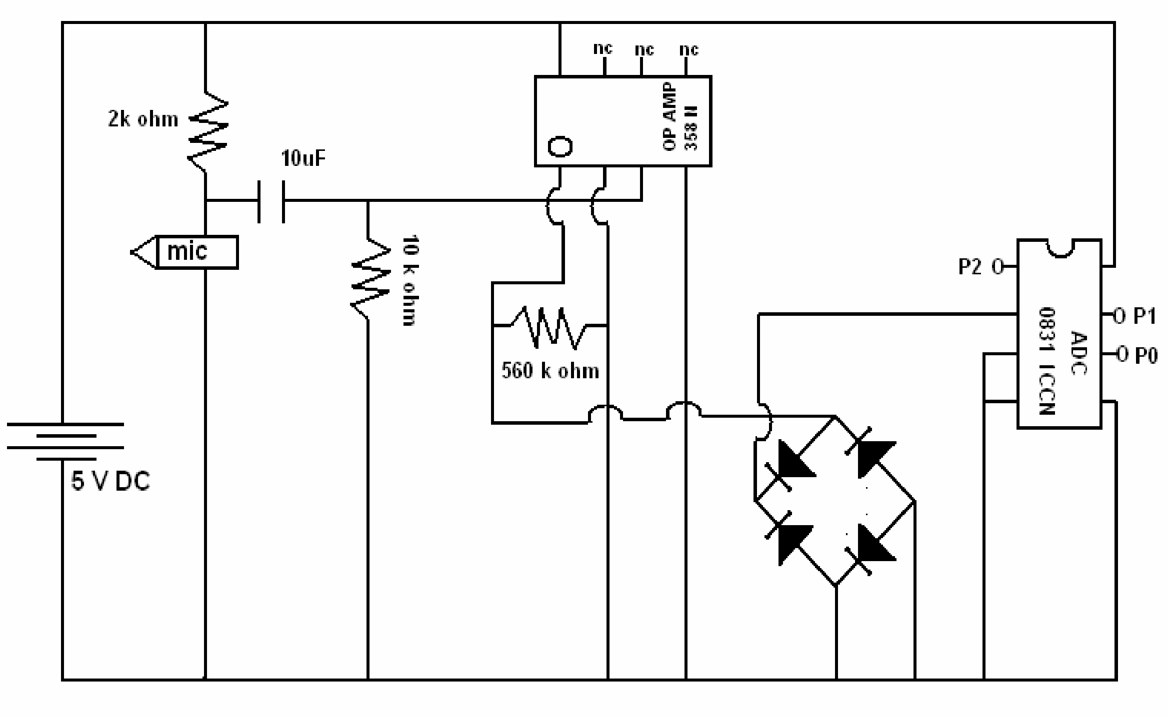
2- V. Kapila, *Course Notes for Mechatronics ME 3484*, Polytechnic University, Brooklyn, NY, (2003).

APPENDIX A: Circuit Diagrams

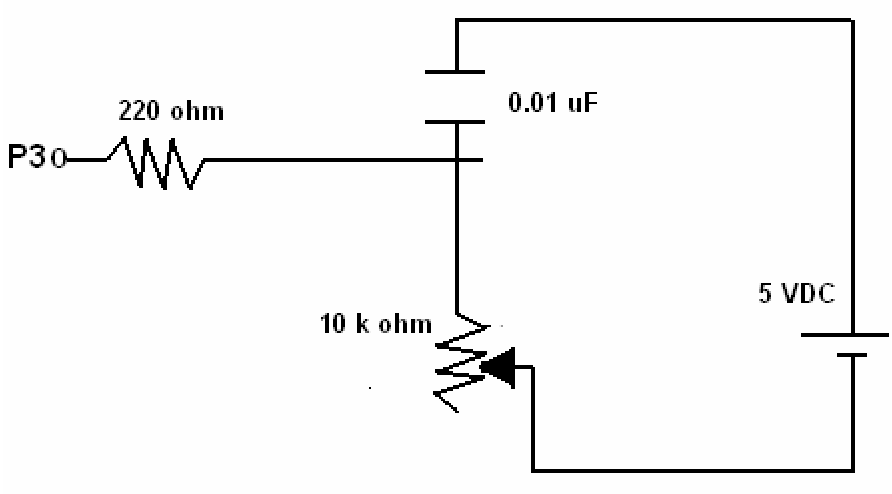
FREQUENCY GENERATOR AND SPEAKER CIRCUIT



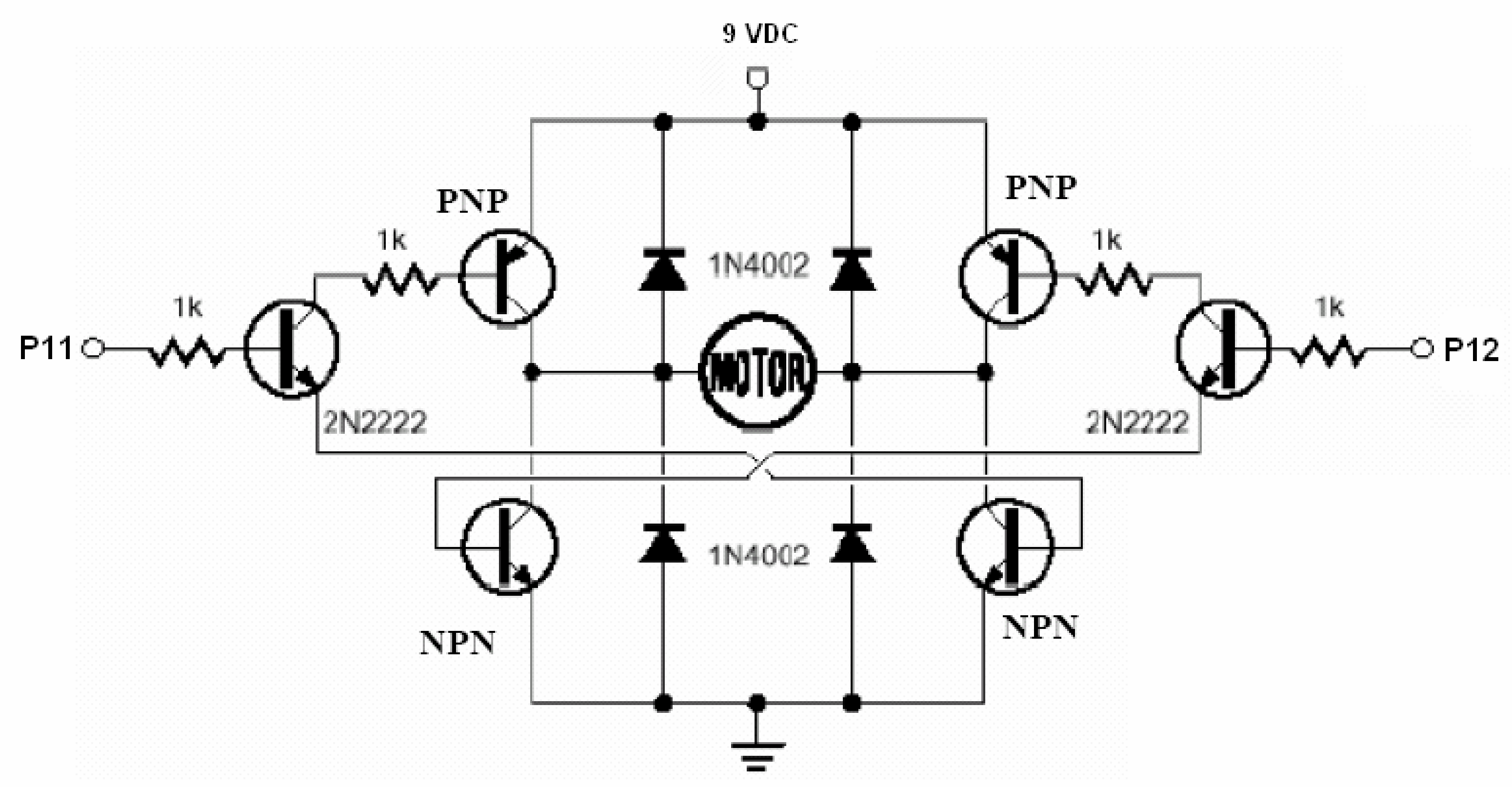
MICROPHONE CIRCUIT



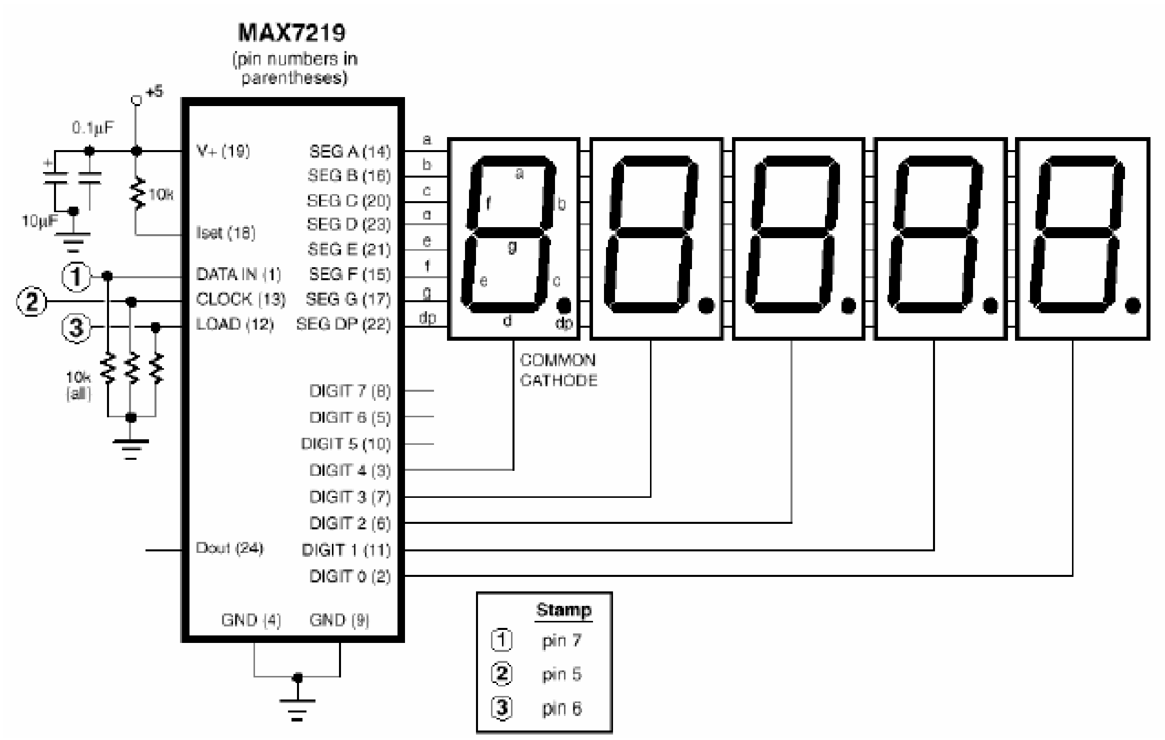
POSITION DETECTION CIRCUIT



H BRIDGE CIRCUIT AND DC MOTOR



7 SEGMENT DISPLAY AND CIRCUIT



APPENDIX B: Project Code (Written in PBASIC) “RESONANCE.BS2”

```
' ==============================================================================
' {$STAMP BS2}
'
' ==============================================================================
' Program Name
' THE SMART RESONANCE TUBE
'------------------------------------------------------------------------------
' Program Description
' ------------------------------------------------------------------------------
' This program stops piston due to volume change of resonance
' the piston has to overshoot by 1 cycle to find a lower value
' No piston position sensor
' LED displays frequency from 555 timer
' Microphone captures sound and converts it to mVolt signal
' mVolt comparison % increase
'003  initial RCT value for 0m length
'4.15  RCT = 1 cm length
' ------------------------------------------------------------------------------
' I/O Definitions
' ------------------------------------------------------------------------------
FreqPin CON 4 ' frequency input pin from 555 timer
DATA_n CON 7 'data to LED
CLK    CON 5 'clock for LED
Load   CON 6 '? for LED
A2Ddata         CON     0                       ' mic A/D data line
A2Dclock        CON     1                       ' mic A/D clock
A2Dcs           CON     2                       ' mic A/D chip select (low true)
HA CON 12       'H bridge motor control
HB CON 11       'H bridge motor control
POT    CON 3


' ------------------------------------------------------------------------------
' Constants
' ------------------------------------------------------------------------------

OneSec CON 1000 ' one second - for 555 timer
decode CON 9    'LED
brite  CON 10   'LED
scan   CON 11   'LED
switch CON 12   'LED


' ------------------------------------------------------------------------------
' Variables
' ------------------------------------------------------------------------------
freq VAR Word ' 555 frequency
freqold VAR Word ' 555 frequency comparison
setup  VAR Word    'LED
number VAR Word    'LED
index  VAR Nib     'LED
temp   VAR Byte    'LED
odd    VAR index.BIT0  'LED
result          VAR     Word                    'mic result of conversion
mVolts          VAR     Word                    'mic convert to millivolts
mVoltsold          VAR     Word                 'old mic convert TO millivolts
mVoltscomp         VAR     Word                 'mic convert comparison
avg          VAR     Word                    'average of X ADC outputs
X VAR Byte                                    '# of samples from ADC output
rct VAR Word                                  'pot position variable
pot_init VAR Nib                              ' position pt rct value at 0 cm
rct_comp VAR Word                             'pot position comparison

' ------------------------------------------------------------------------------
' Initialization
' ------------------------------------------------------------------------------

Initialize:
  HIGH A2Dcs  'mic
  LOW HA          'stop motor
  LOW HB          'stop motor
  mVoltsold = 0
  pot_init = 3   'rct value for pot at 0 cm
  GOSUB piston_reset
```

```
' ------------------------------------------------------------------------------
' Program Code
' ------------------------------------------------------------------------------
PreMain:
   LOW HA
   LOW HB

Main:
   DEBUG ">>>>>>>MAIN<<<<<<<"   , CR
   GOSUB freq_output
   GOSUB pot_comp

'Microphone
  avg = 0
  FOR x = 1 TO  75
  GOSUB Read_ADC_0831
  avg = avg + result
  NEXT
  avg = avg/x
  mVolts = avg */ $139C                      ' x 19.6 (mv / unit) new line
  DEBUG "ADC..... ", DEC avg, "   ", CR
  DEBUG "volts... ", DEC mVolts DIG 3, ".", DEC3 mVoltS, CR

'hbridge motor control
  HIGH HB         'start motor
  IF rct > 600 THEN reverse_piston

'sound amplitude comparison
  mVoltscomp = mVolts */ $0105           'take avg x 1.20 for comparison
  IF mVoltscomp < mVoltsold  THEN reverse_piston     'goto reverse pistion subroutine
  DEBUG "voltsold... ", DEC mVoltsold DIG 3, ".", DEC3 mVoltsold, CR
  DEBUG "voltscomp... ", DEC mVoltscomp DIG 3, ".", DEC3 mVoltscomp, CR
  GOSUB led_display
  mVoltsold = mVolts           'set IF THEN motor control variables =
  DEBUG "rct... ", DEC rct , CR
  DEBUG "rct_comp... ", DEC rct_comp , CR
  GOTO Main ' do it again

' ------------------------------------------------------------------------------
' Subroutines
' ------------------------------------------------------------------------------

'mic subroutine
Read_ADC_0831:
  LOW A2Dcs
  SHIFTIN A2Ddata, A2Dclock, MSBPOST, [result\9]
  HIGH A2Dcs
  RETURN

'reverse motor subroutine
Reverse_piston:
  DEBUG "<<<<<<<REVERSE PISTON>>>>>>>>>>>>" , CR
  GOSUB pot_rct
  IF rct < rct_comp THEN Main:
  mVoltsold = mVolts
  LOW HB
  HIGH HA
  PAUSE 200
  LOW HA
  'Microphone
  avg = 0
  FOR x = 1 TO  60
  GOSUB Read_ADC_0831
  avg = avg + result
  NEXT
  avg = avg/x
  mVolts = avg */ $139C                      ' x 19.6 (mv / unit) new line
  DEBUG "ADC..... ", DEC avg, "   ", CR
  DEBUG "volts... ", DEC mVolts DIG 3, ".", DEC3 mVoltS, CR
  'hbridge motor control
  mVoltscomp = mVolts */ $0101       'take avg x 1.01 for comparison
  DEBUG "voltsold... ", DEC mVoltsold DIG 3, ".", DEC3 mVoltsold, CR
  DEBUG "voltscomp... ", DEC mVoltscomp DIG 3, ".", DEC3 mVoltscomp, CR
  IF mVoltscomp < mVoltsold  THEN stop_piston     'goto stop pistion subroutine
  DEBUG "rct... ", DEC rct , CR
  DEBUG "rct_comp... ", DEC rct_comp , CR
  GOTO Reverse_piston

'stop piston subroutine
Stop_piston:
  DEBUG "<<<<<<<<<<<<<<<STOP PISTON>>>>>>>>>>>>>>", CR
```

```
  GOSUB pot_rct
  IF rct > rct_comp THEN reverse_piston:
  IF rct = rct_comp  THEN stop_motor:
  IF rct < rct_comp THEN main:
stop_motor:

  DEBUG "<<<<<<<<<<<<<<<STOP MOTOR>>>>>>>>>>>>>>>>>>>" ,CR
  DEBUG "rct... ", DEC rct , CR
  DEBUG "rct_comp... ", DEC rct_comp , CR
  LOW HA
  LOW HB
  END

'pot value for comparison to freq subroutine
pot_comp:
  rct_comp =   343 * 100             ' multiply vs to convert to cm
  rct_comp = rct_comp */ $0123       ' * 4.15 (rct conv to cm) / 4 formula
  rct_comp = rct_comp / freq         ' divide by freq to find wavelength
  rct_comp = rct_comp - 5            'd correction
  rct_comp = rct_comp - 3            ' Subtract inital pot value at 0
  DEBUG "rct_comp... ", DEC rct_comp , CR
  RETURN

'pistion reset subroutine
piston_reset:
  pot_init = pot_init + 5
  piston_reset1
  GOSUB pot_rct
  GOSUB freq_output
  GOSUB led_display
  IF RCT < pot_init THEN PreMain:
  HIGH HA
  IF rct > pot_init THEN piston_reset1:


'position pot rct value subroutine
Pot_rct:
  HIGH POT
  RCTIME POT,1,rct
  DEBUG DEC3 rct,CR
  RETURN

'led display subroutine
led_display:
  OUTPUT 5
  OUTPUT 6
  OUTPUT 7
  OUT5=0
  OUT6=0
  OUT7=0
  FOR index = 0 TO 7
  LOOKUP index,[scan,3,brite,10,decode,$1F,switch,1],setup
  SHIFTOUT DATA_n,CLK,MSBFIRST,[setup]
  IF odd = 0 THEN noLoad
  PULSOUT Load,1
  NoLoad:
  NEXT
  MaxDisplay:
  number = freq
  FOR index = 0 TO 3
  temp = (number DIG index)
  SHIFTOUT DATA_n,CLK,MSBFIRST,[4-index]
  SHIFTOUT DATA_n,CLK,MSBFIRST,[temp]
  PULSOUT Load,5
  NEXT
  RETURN

'freq output from 555 timer
freq_output:
  COUNT FreqPin, OneSec, freq ' collect pulses for 1 second
  DEBUG  "Frequency: ", DEC freq, " Hz",CR ' display on DEBUG screen
  RETURN
```