Chapter 11 Supplemental labs

Lab 4: Alarms

Now that we have found the resonant frequency for our piezo element, we can use that frequency to generate various alarm sound patterns. This lab uses the concepts for timers and arrays learned earlier in the chapter.

Parts Required:
1 Arduino
1 USB cable
1 Arduino Proto Shield and jumper wires
1 Piezo element

Estimated time for this lab: 30 minutes

☐ Use the piezo circuit shown in Lab 3.
☐ Either copy and paste, or type in the following program into the Arduino IDE. [Or open the A101_ch11_alarms.ino program - note this file is located in A101_ch11_supplemental.zip that you can find on the Nuts&Volts web site page for this article and on the www.arduinoclassroom.com Chapter 11 web page.]

```
// A101_ch11_alarms 9/30/14 Joe Pardue
#include <TimerOne.h> // Timer 1 library

#define highTone 4530
#define lowTone 3500

int speaker = 6;   // the number of the speaker driver pin
int alarmArray[100]; // array to hold the arlarm tones
int arrayCount = 0; // initialize the array count

void setup(){
  // Set up the serial port
  Serial.begin(57600);

  // identify yourself
  Serial.println(F("a101_ch11_alarm rev. 0.01"));

  // initialize timer1
  Timer1.initialize(125000); // interrupts 8 times a second
  Timer1.attachInterrupt(myTimer1);
}

void loop(){
  threeBeep();
  delay(3000);
  threeBeepWarbleUp();
  delay(3000);
  threeBeepWarbleDown();
  delay(3000);
```

```
  threeBeepWarbleUpDown();
  delay(3000);
  threeBeepWarbleDownUp();
  delay(3000);
  sos();
  delay(5000);
}

void sos(){
 int sos[] = {0, 0, highTone, 0, highTone, 0, highTone, 0, highTone,
highTone, highTone, 0, highTone, highTone, highTone, 0, highTone,
highTone, highTone, 0, highTone, 0, highTone, 0, highTone,0};
  arrayCount = 25;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = sos[i];
  }
}

void threeBeep(){
 int threeBeep[] = {0, 0, highTone, 0, highTone, 0, highTone};
  arrayCount = 6;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = threeBeep[i];
  }
}

void threeBeepWarbleUp(){
 int threeBeepWarble[] = {0, 0, highTone, lowTone, 0, highTone,
lowTone, 0, highTone, lowTone};
  arrayCount = 9;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = threeBeepWarble[i];
  }
}

void threeBeepWarbleUpDown(){
 int threeBeepWarble[] = {0, 0, highTone, lowTone, highTone, 0,
highTone, lowTone, highTone, 0, highTone, lowTone, highTone};
  arrayCount = 12;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = threeBeepWarble[i];
  }
}

void threeBeepWarbleDownUp(){
 int threeBeepWarble[] = {0, 0, lowTone, highTone, lowTone, 0, lowTone,
highTone, lowTone, 0, lowTone, highTone, lowTone};
  arrayCount = 12;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = threeBeepWarble[i];
  }
}
```

```
void threeBeepWarbleDown(){
 int threeBeepWarble[] = {0, 0, lowTone, highTone,  0, lowTone,
highTone, 0, lowTone, highTone};
  arrayCount = 9;
  for(int i = 0; i<=arrayCount; i++)
  {
    alarmArray[i] = threeBeepWarble[i];
  }
}

// Called 8 times per second
// Checks alarm array count is > 0
// if so it plays the tone in the alarm array
// and decrements the count.
void myTimer1()
{
  if((arrayCount > 0)){
   if(alarmArray[arrayCount] == 0) noTone(speaker);
   else tone(speaker,alarmArray[arrayCount]);
   arrayCount--;
  }
}
```

☐ Compile and run the program.
☐ Verify that the alarms play according to the program directions.

Lab 5: Light Theremin

A Theremin is an electronic audio instrument that was often used in early science fiction and horror films to make spooky UFO or ghostly sounds. Our version uses the light sensor to provide a way to input a continuously varying data from relative dark to relative light and to map those sensed variations into tones output on the piezo element.

Parts Required:
1 Arduino
1 USB cable
1 Arduino Proto Shield and jumper wires
1 Piezo element
1 CdS light sensor
1 10000 Ω resistor

Estimated time for this lab: 30 minutes

☐ Build a circuit that combines the light sensor circuit shown in Lab 1 and the piezo speaker circuit shown in Lab 3. We see these circuits combined in Figures 12 and 13

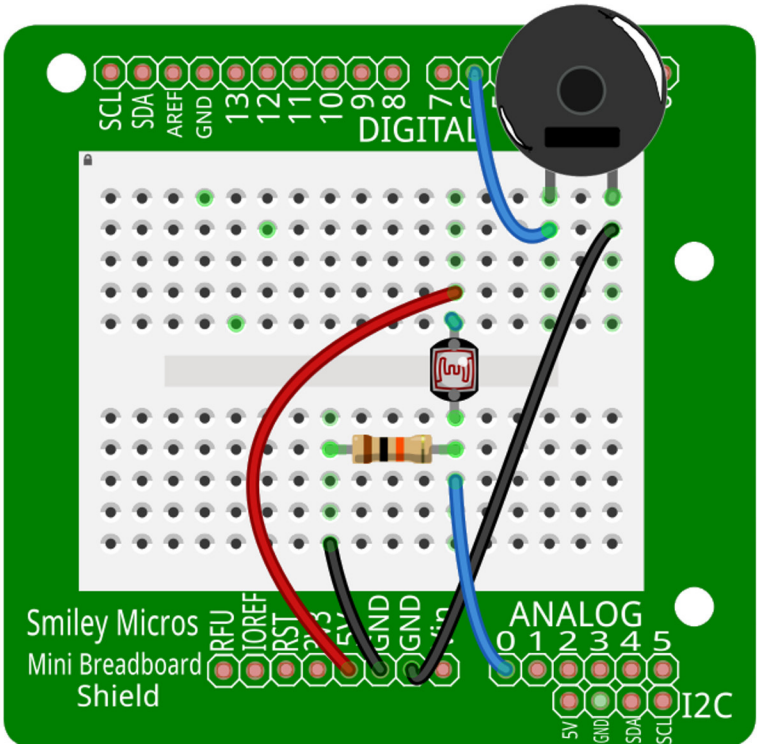

Figure 12: Theremin light sensor breadboard

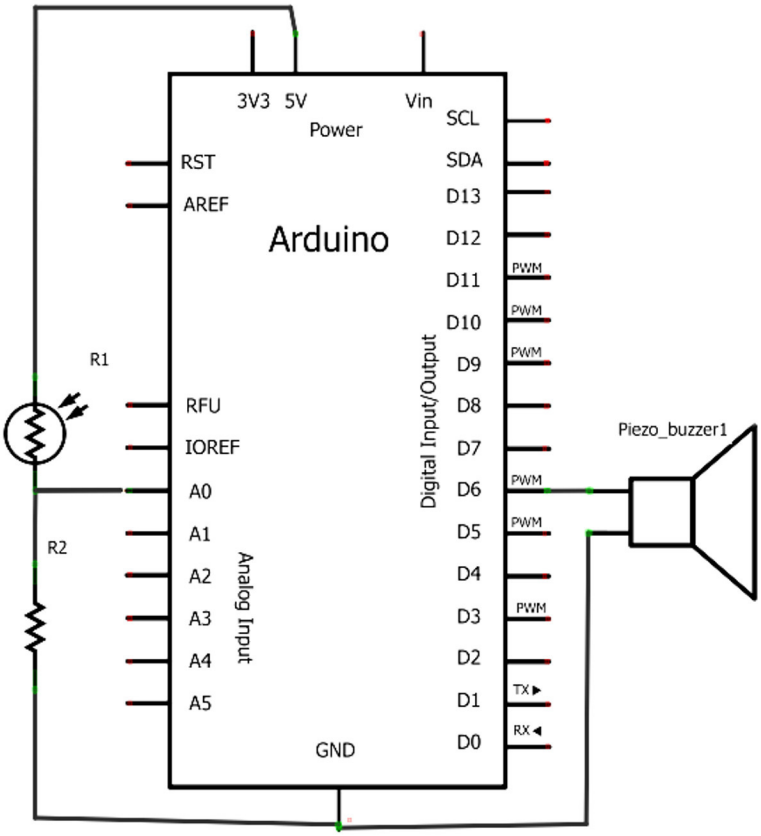

Figure 13: Theremin light sensor schematic

☐ Either copy and paste, or type in the following program into the Arduino IDE. [Or open the A101_ch11_light_sensor_theremin.ino program - note this file is located in A101_ch11_supplemental.zip that you can find on the Nuts&Volts web site page for this article and on the www.arduinoclassroom.com Chapter 11 web page.]

```
// A101_ch11_light_sensor_theremin 10/5/14 Joe Pardue

int sensorPin = A0;  // analog input pin
unsigned int sensorValue = 0;  // store the analog input value

int speaker = 6;
unsigned int freq = 0;;

void setup() {
  Serial.begin(57600);
  Serial.println("Light sensor LED Theremin rev 1.0");
}

void loop() {
  // read the value from the sensor:
  freq = analogRead(sensorPin);

  // map the light level to the frequency
  freq = map(freq, 1024, 0, 0, 5000);
  // play the tone
  tone(speaker,freq);
}
```

☐ Compile and run the program.
☐ Verify that the tone varies according to the light intensity.

Lab 6: Making Music

Making music with an Arduino requires a bit of understanding of how music is written and produced that is beyond the scope of this lab. Basically you produce musical tones for short periods of time according to a defined pattern. I transcribed a number of tunes so that there are two arrays for each tune, one for the tone and one for the time to play the tone. The program to read these arrays and produce the melodies is based on concepts we have already visited, but be forewarned - this is a very long program. Don't let that scare you though; you should understand each aspect of the program. We have one simplification. While there are a bunch of melodies available, the program only plays one. You select the melody you want to play and remove the comment symbols: /* and */ from the program you want to play and make sure that all the other programs are blocked in with those comment symbols. You could write some extra code that would allow you to have all the melodies available and let you select a specific melody using the Serial Monitor and what you've learned about sending commands from the PC to the Arduino.

Parts Required:
1 Arduino

1 USB cable
1 Arduino Proto Shield and jumper wires
1 Piezo element

Estimated time for this lab: 30 minutes

☐ Use the piezo circuit shown in Lab 3.
☐ Either copy and paste, or type in the following program into the Arduino IDE. [Or open the A101_ch11_tunes.ino program - note this file is located in A101_ch11_supplemental.zip that you can find on the Nuts&Volts web site page for this article and on the www.arduinoclassroom.com Chapter 11 web page.]

```
// A101_ch11_tunes 9/24/14 Joe Pardue

#include <TimerOne.h> // Timer 1 library
/*
//HBday.ardplay
int timerMS = 40;
int dataCount = 50;
int freq[] =
{587,0,587,0,659,0,587,0,784,0,740,0,587,0,587,0,659,0,587,0,440,0,784,
0,587,0,587,0,1175,0,988,0,1568,0,1480,0,1319,0,1047,0,1047,0,988,0,156
8,0,880,0,1568,0,};
int duration[] =
{3,1,2,1,8,1,8,1,8,1,16,1,3,1,2,1,8,1,8,1,8,1,16,1,3,1,2,1,8,1,8,1,8,1,
8,1,16,1,3,1,2,1,8,1,8,1,8,1,16,1};
*/

/*
//GoodBad.ardplay
int timerMS = 154;
int dataCount = 33;
int freq[] =
{0,466,1244,932,1244,932,1480,1660,1244,932,1244,932,1244,932,1480,1660
,1109,932,1244,932,1244,932,1480,1397,1244,1109,932,1244,932,1244,932,1
660,1244,};
int duration[] =
{1,1,1,1,1,3,2,2,8,1,1,1,1,3,2,2,8,1,1,1,1,3,2,0,1,8,1,1,1,1,3,2,8,};
*/

/*
//lullaby.ardplay
int timerMS = 60;
int dataCount = 109;
int freq[] =
{659,0,659,0,784,0,659,0,659,0,784,0,0,0,659,0,784,0,1047,0,494,0,440,0
,440,0,784,0,587,0,659,0,698,0,587,0,587,0,659,0,698,0,0,0,587,0,698,0,
494,0,440,0,784,0,494,0,1047,0,523,0,523,0,1047,0,440,0,698,0,784,0,659
,0,523,0,698,0,784,0,440,0,784,0,523,0,523,0,1047,0,440,0,698,0,784,0,6
59,0,523,0,698,0,784,0,698,0,659,0,587,0,523,};
int duration[] =
{4,1,4,1,5,1,4,1,8,1,8,1,8,1,4,1,4,1,8,1,5,1,4,1,8,1,8,1,4,1,4,1,8,1,8,
1,4,1,4,1,8,1,8,1,4,1,4,1,4,1,4,1,8,1,8,1,16,1,4,1,4,1,16,1,4,1,4,1,16,
```

```
1,4,1,4,1,8,1,8,1,8,1,16,1,4,1,4,1,16,1,4,1,4,1,16,1,4,1,4,1,4,1,2,1,2,
1,8,1,8,1,16,};
*/

/*
//BOLERO.ardplay
int timerMS = 94;
int dataCount = 47;
int freq[] =
{1047,1047,494,1047,1175,1047,988,880,1047,1047,880,1047,1047,988,1047,
880,1568,1319,1397,1568,1568,1397,1319,1175,1319,1397,1568,880,1568,156
8,1568,880,988,880,1568,1397,1319,1175,1319,1175,1047,1047,1047,1175,13
19,1397,1175,1568,};
int duration[] =
{8,4,2,2,2,2,2,2,4,2,2,8,4,2,2,2,2,2,2,16,2,2,2,2,2,2,2,2,8,8,2,2,2,2,2
,2,2,2,2,2,4,4,2,2,4,4,8,16,};
*/


/*
//ode.ardplay - note duration is off
int timerMS = 77;
int dataCount = 59;
int freq[] = {659,0, 659,0 ,698,0, 784,0, 784,0, 698,0, 659,0, 587,0,
523,0, 523,0, 659,0, 659,0, 659,0, 587,0, 587,0, 659,0, 659,0, 698,0,
784,0, 784,0, 698,0, 659,0, 587,0, 523,0, 523,0, 587,0, 659,0, 587,0,
523,0, 523,0,};
//int freq[] =
{659,659,698,784,784,698,659,587,523,523,659,659,659,587,587,659,659,69
8,784,784,698,659,587,523,523,587,659,587,523,523,0,};
int duration[] = {3,1, 3,1, 4,1 ,5,1, 5,1, 4,1, 3,1 ,2,1, 2,1, 2,1 ,2,1
,3,1, 3,1, 2,1, 2,1 ,3,1, 3,1, 4,1, 5,1, 5,1, 4,1 ,3,1 ,2,1, 1,1, 1,1,
2,1, 3,1, 2,1, 1,1,};
*/

/*
//jinglebells.ardplay
int timerMS = 34;
int dataCount = 96;
int freq[] =
{659,0,659,0,659,0,659,0,659,0,659,0,659,0,784,0,523,0,587,0,659,0,698,
0,698,0,698,0,698,0,698,0,659,0,659,0,659,0,659,0,587,0,587,0,659,0,587
,784,659,0,659,0,659,0,659,0,659,0,659,0,659,0,784,0,523,0,587,0,659,0,
698,0,698,0,698,0,698,0,698,0,659,0,659,0,659,0,784,0,784,0,698,0,587,0
,523,0,};
int duration[] =
{8,1,8,1,16,1,8,1,8,1,16,1,8,1,8,1,8,1,8,1,8,1,32,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1
,8,1,8,1,8,1,8,1,8,1,8,1,16,16,8,1,8,1,16,1,8,1,8,1,16,1,8,1,8,1,8,1,8,
1,32,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,32,8,};
*/

/*
//amazing.ardplay
int timerMS = 34;
int dataCount = 38;
int freq[] =
{391,261,329,261,329,293,261,2200,391,391,261,329,261,329,293,391,391,3
```

```
91,329,391,329,391,329,261,391,2200,261,2200,391,391,261,329,261,329,29
3,261,261,261,};
int duration[] =
{8,32,8,8,32,16,32,16,32,16,32,8,8,32,16,32,16,32,16,16,8,8,8,32,16,32,
8,8,32,16,16,8,8,32,16,32,16,32,};
*/

/*
//smokey.ardplay
int timerMS = 54;
int dataCount = 35;
int freq[] =
{523,0,523,659,784,1047,880,0,0,1397,1397,1568,880,1568,0,1568,0,0,1047
,1047,1319,1568,0,1568,1175,0,0,1319,1397,1175,1047,0,1047,0,1047,};
int duration[] =
{8,1,8,8,8,8,16,8,8,8,8,8,8,8,1,8,8,8,8,8,8,8,1,8,8,8,8,8,8,8,8,1,8,1,1
6,};
*/

/*
//america.ardplay
int timerMS = 55;
int dataCount = 111;
int freq[] =
{391,0,391,0,329,0,329,0,391,0,391,0,293,0,293,0,329,0,349,0,391,0,440,
0,493,0,391,0,391,0,391,0,329,0,329,0,391,0,391,0,293,0,293,0,587,0,554
,0,587,0,659,0,440,0,587,0,391,0,659,0,659,0,587,0,523,0,493,0,493,0,49
3,0,523,0,587,0,493,0,440,0,391,0,523,0,523,0,523,0,440,0,440,0,523,0,5
23,0,391,0,391,0,391,0,440,0,523,0,391,0,587,0,523,};
int duration[] =
{8,1,10,1,4,1,8,1,8,1,10,1,4,1,8,1,8,1,8,1,8,1,8,1,8,1,16,1,8,1,10,1,4,
1,8,1,8,1,10,1,4,1,8,1,8,1,8,1,8,1,8,1,8,1,16,1,8,1,10,1,4,1,8,1,8,1,8,
1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,16,1,8,1,10,1,4,1,8,1,8,1,8,1,8,1,8,1,8,1
,8,1,8,1,8,1,8,1,16,};
*/
/*
//furelise.ardplay
int timerMS = 28;
int dataCount = 73;
int freq[] =
{659,0,622,0,659,0,622,0,659,0,493,0,587,0,523,0,440,0,261,0,329,0,440,
0,493,0,0,329,0,415,0,493,0,523,0,0,329,0,0,659,0,622,0,659,0,622,0,659
,0,493,0,587,0,523,0,440,0,261,0,0,329,0,440,0,493,0,329,0,523,0,493,0,
440,};
int duration[] =
{8,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,16,1,8,1,8,1,8,1,16,1,1,8,1,8,1,8,1,16
,1,1,8,1,1,8,1,8,1,8,1,8,1,8,1,8,1,8,1,16,1,8,1,1,8,1,8,1,16,1,8,1,
8,1,8,1,16,};
*/
/*
//ode_to_joy.ardplay
int timerMS = 30;
int dataCount = 122;
int freq[] =
{329,0,329,0,349,0,391,0,391,0,349,0,329,0,293,0,261,0,261,0,293,0,329,
0,329,0,293,0,293,0,329,0,329,0,349,0,391,0,391,0,349,0,329,0,293,0,261
,0,261,0,293,0,329,0,293,0,261,0,261,0,293,0,293,0,329,0,261,0,293,0,32
```

```
9,349,0,329,0,261,0,293,0,329,349,0,329,0,293,0,261,0,293,0,196,0,329,0
,329,0,349,0,391,0,391,0,349,0,329,0,293,0,261,0,261,0,293,0,329,0,293,
0,261,0,261,0,};
int duration[] =
{8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,10,2,4,2,16,2,8,2,8,2,
8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,8,2,10,2,4,2,16,2,8,2,8,2,8,2,8,2,8
,2,4,4,2,8,2,8,2,8,2,4,4,2,8,2,8,2,8,2,8,2,16,2,8,2,8,2,8,2,8,2,8,2,8,2
,8,2,8,2,8,2,8,2,8,2,8,2,10,2,4,2,16,2,};
*/
/*
//indiana_jones_long.ardplay
int timerMS = 25;
int dataCount = 157;
int freq[] =
{329,0,349,0,391,0,523,0,293,0,329,0,349,0,391,0,440,0,493,0,698,0,440,
0,493,0,523,0,587,0,659,0,329,0,349,0,391,0,523,0,587,0,659,0,698,0,391
,0,440,0,659,0,587,0,391,0,659,0,587,0,391,0,659,0,587,0,391,0,698,0,65
9,0,587,0,523,0,329,0,391,0,349,0,293,0,349,0,329,0,391,0,523,0,523,0,3
29,0,391,0,349,0,293,0,391,0,329,0,293,0,261,0,261,0,329,0,391,0,349,0,
293,0,349,0,329,0,523,0,523,0,391,0,391,0,659,0,587,0,391,0,659,0,587,0
,391,0,659,0,587,0,391,0,698,0,659,0,587,0,523};
int duration[] =
{8,1,8,1,8,1,32,1,10,1,8,1,32,1,10,1,8,1,8,1,32,1,10,1,8,1,8,1,8,1,8,8,
10,1,8,1,8,1,16,1,10,1,8,1,32,1,10,1,8,1,8,1,10,1,8,1,8,1,10,1,8,1,8,1,
10,1,8,1,8,1,10,1,8,1,32,1,10,1,8,1,32,1,10,1,8,1,8,1,8,1,8,1,32,1,10,1
,8,1,32,1,10,1,8,1,8,1,8,1,8,1,32,1,10,1,8,1,32,1,10,1,8,1,8,1,8,1,32,1
,10,1,8,1,8,1,10,1,8,1,8,1,10,1,8,1,8,1,10,1,8,1,8,1,10,1,8,1,32};
*/

//indiana_short.ardplay
int timerMS = 26;
int dataCount = 76;
int freq[] =
{329,0,349,0,391,0,523,0,293,0,329,0,349,0,391,0,440,0,493,0,698,0,440,
0,493,0,523,0,587,0,659,0,329,0,349,0,391,0,523,0,587,0,659,0,698,0,391
,0,440,0,659,0,587,0,391,0,659,0,587,0,391,0,659,0,587,0,391,0,698,0,65
9,0,587,0,523,0,};
int duration[] =
{8,1,8,1,8,1,32,1,10,1,8,1,32,1,10,1,8,1,8,1,32,1,10,1,8,1,8,1,8,1,8,8,
10,1,8,1,8,1,16,1,10,1,8,1,32,1,10,1,8,1,8,1,10,1,8,1,8,1,10,1,8,1,8,1,
10,1,8,1,8,1,10,1,8,1,32,1,};

/*
//star_wars.ardplay
int timerMS = 18;
int dataCount = 74;
int freq[] =
{261,0,261,0,261,329,0,523,0,493,0,440,0,391,0,698,0,523,0,493,0,440,0,
391,0,698,0,523,0,493,0,440,0,493,0,391,0,261,0,261,0,261,0,329,0,523,0
,493,0,440,0,391,0,698,0,523,0,493,0,440,0,391,0,698,0,523,0,493,0,440,
0,493,0,391,0};
int duration[] =
{8,2,8,2,8,32,2,32,2,8,2,8,2,8,2,32,2,16,2,8,2,8,2,8,2,32,2,16,2,8,2,8,
2,8,2,32,2,8,2,8,2,8,2,32,2,32,2,8,2,8,2,8,2,32,2,16,2,8,2,8,2,8,2,32,2
,16,2,8,2,8,2,8,2,32};
*/
```

```
const int speaker = 6;   // the number of the speaker driver pin
int playIt = 0;
int i = 0;
int count = dataCount;
int myDuration = 0;
int finished = 1;

void setup(){
  // Set up the serial port
  Serial.begin(57600);

  // identify yourself
  Serial.println(F("a101_ch11_tunes rev. 0.01"));

  // initialize timer1 interrupt
  Timer1.initialize(timerMS*1000);
  Timer1.attachInterrupt(myTimer1);
}

void loop(){
  // if the tune is finished playing, play it again
  if(finished)
  {
    playTune();
  }
}

void playTune()
{
  playIt = 1;
  i = 0;
  count = dataCount;
  myDuration = 0;
  finished = 0;
}

void myTimer1()
{
  if(playIt)
  {
    // First see if you can get it to send out a single tone for the
duration in the array
    if(myDuration <= 0)
    {
      myDuration = duration[i];
      if(freq[i] == 0)noTone(speaker);
      else tone(speaker,freq[i]);
      i++;
    }
    if(myDuration >= 0)
    {
      myDuration--;
    }

    if(i > dataCount)
    {
     i = 0;
```

```
      playIt = 0;
      noTone(speaker);
      finished = 1;
     }
   }
}
```

☐ Compile and run the program.
☐ Verify that the Indiana Jones tune plays on the piezo element.
☐ Add comments to the Indiana Jones tune and remove comments from other tunes to play them.