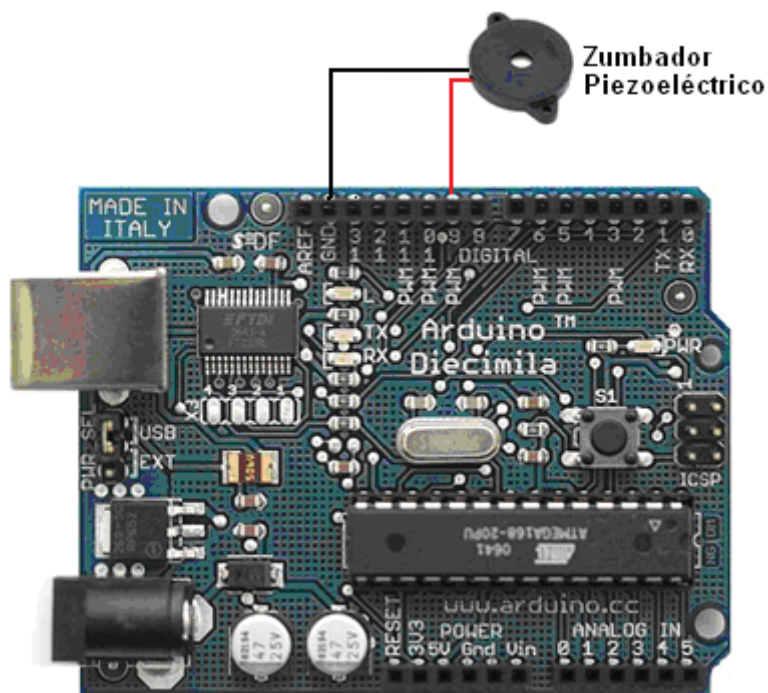

19. Toca tonos desde el puerto serial

En este ejercicio usaremos un Piezo eléctrico para reproducir tonos, aprovechando la capacidad que tiene el procesador para producir señales PWM y así reproducir música.

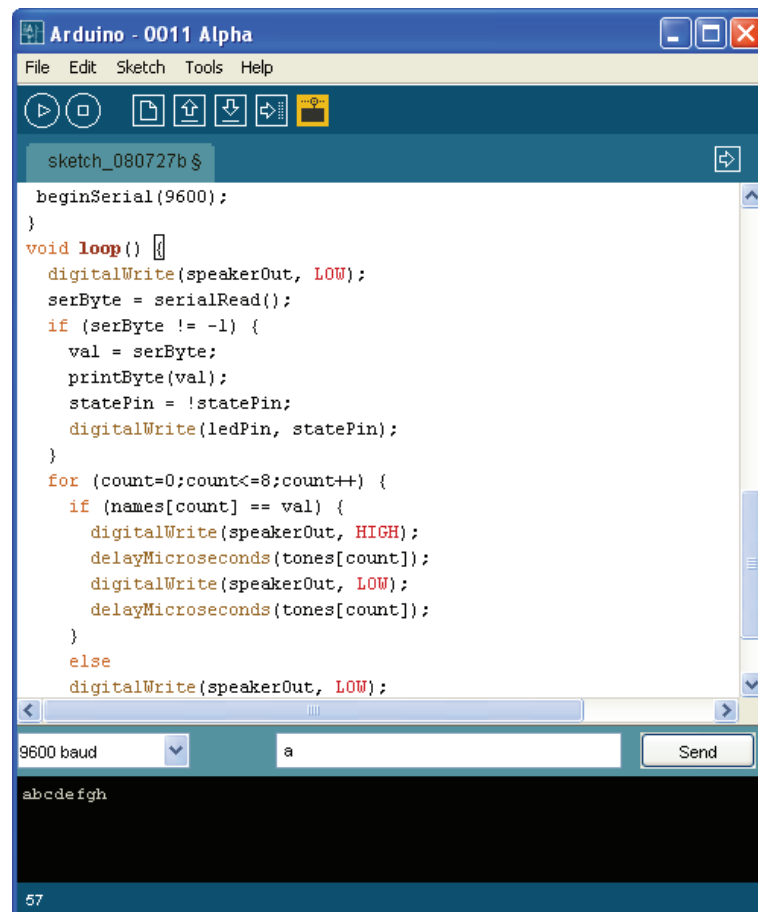
Un piezo eléctrico no es más que un dispositivo que puede ser usado tanto para reproducir o detectar tonos. En nuestro ejemplo estamos conectando el piezo en el pin número 9, que permite enviar una señal PWN .

Los tonos pueden ser generados a partir de cualquier programa capaz de enviar valores ASCII a través del puerto serial. Programas de Terminal, Processing, Pure Data o Director, pueden ser usados para generar los tonos. En nuestro caso los caracteres ASCII los enviaremos por la ventana de dialogo del IDE Arduino.



Ejemplo de la conexión de un piezo eléctrico al pin 9, y utilizaremos 8 tonos tal como se indica en el código del programa. Si pulsamos cualquier otra letra que no sea de las que tienen asignados tonos el zumbador deja de emitir tonos.

Aspecto del IDE Arduino enviando caracteres ASCII a la tarjeta para que estos se conviertan en tonos



Código:

```

/* Teclado Serial
 * -----
 *
 * Programa para reproducir tonos dependiendo de los datos que vengan del Puerto
 * serie. El cálculo de los tonos se realiza de acuerdo a la siguiente operación:
 *
 * 
$$\text{pulsoAlto} = 1/(2 * \text{frecuenciaTono}) = \text{periodo} / 2$$

 *
 * Donde los tonos son descritos como en la siguiente tabla:
 *
 * 

| tecla | frecuencia | periodo | PW (pulsoAlto) |
|-------|------------|---------|----------------|
| c     | 261 Hz     | 3830    | 1915           |
| d     | 294 Hz     | 3400    | 1700           |
| e     | 329 Hz     | 3038    | 1519           |


```

<i>* f</i>	<i>349 Hz</i>	<i>2864</i>	<i>1432</i>
<i>* g</i>	<i>392 Hz</i>	<i>2550</i>	<i>1275</i>
<i>* a</i>	<i>440 Hz</i>	<i>2272</i>	<i>1136</i>
<i>* b</i>	<i>493 Hz</i>	<i>2028</i>	<i>1014</i>
<i>* C</i>	<i>523 Hz</i>	<i>1912</i>	<i>956</i>

** Cualquier otra letra provoca el silencio del zumbador*
** (cleft) 2005 D. Cuartielles para K3*
** Trad. Juan C. Carvajal. Modificado J.M.Ruiz*
**/*

```

int ledPin = 13;
int speakerOut = 9;
byte names[] = {'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C'};
int tones[] = {1915, 1700, 1519, 1432, 1275, 1136, 1014, 956};
byte val = 0;
int serByte = -1;
int statePin = LOW;
int count = 0;

void setup() {
  pinMode(ledPin, OUTPUT);
  pinMode(speakerOut, OUTPUT);
  beginSerial(9600);
}

void loop() {
  digitalWrite(speakerOut, LOW);
  serByte = serialRead();
  if (serByte != -1) {
    val = serByte;
    printByte(val);
    statePin = !statePin;
    digitalWrite(ledPin, statePin);
  }
  for (count=0;count<=8;count++) {
    if (names[count] == val) {
      digitalWrite(speakerOut, HIGH);
      delayMicroseconds(tones[count]);
      digitalWrite(speakerOut, LOW);
      delayMicroseconds(tones[count]);
    }
    else
      digitalWrite(speakerOut, LOW);
  }
}

```