

# WOONEY TUNES

DEMO DAY!



# GOALS

- 
1. Create a string instrument that is controlled and played mechanically
  2. Capable of playing music that a traditional violin cannot (e.g. chords, large interval gaps)
  3. Connect to an engaging interface for the listener to choose songs
- 

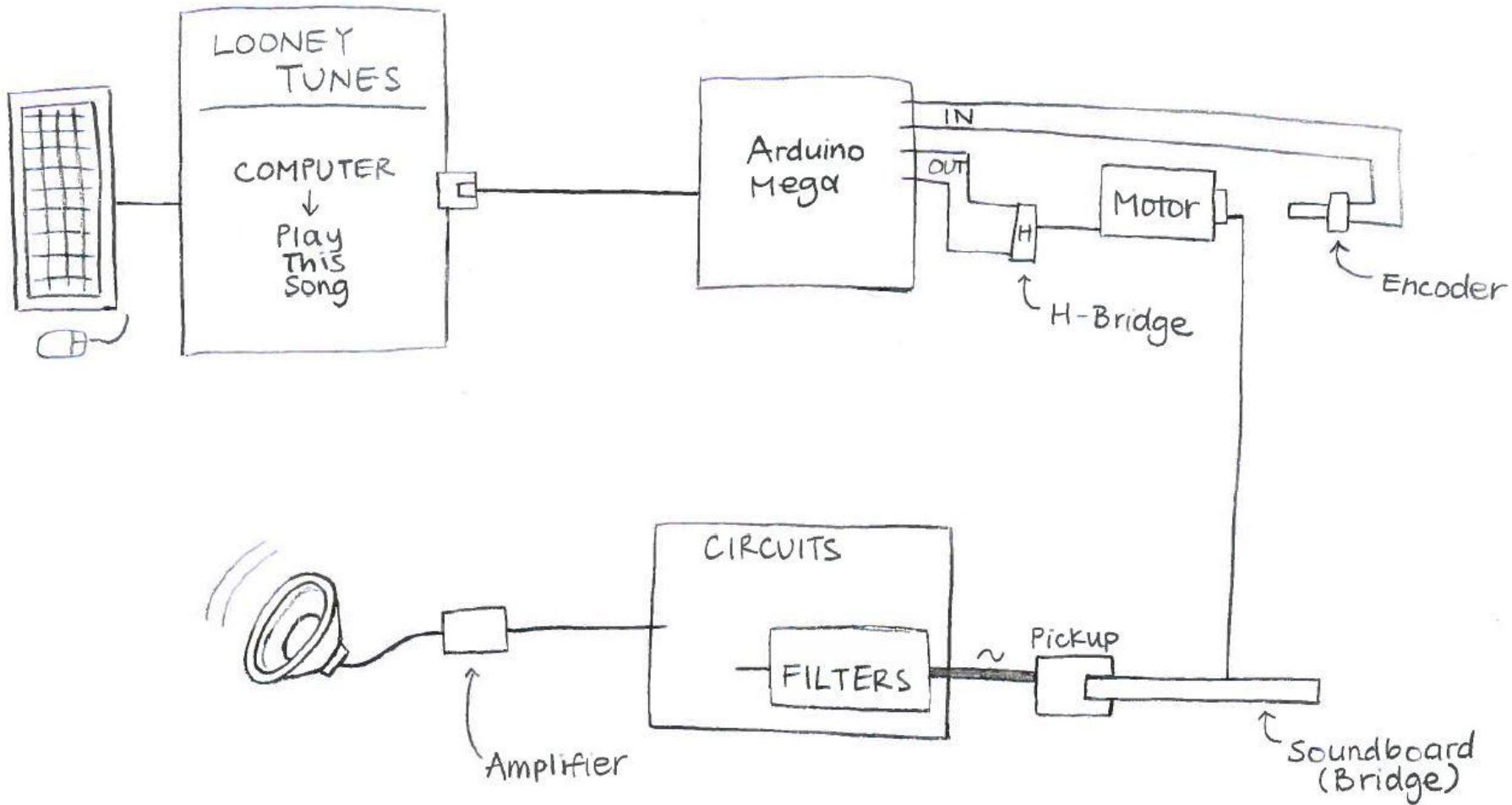
# STRUFTING OUR STUFF



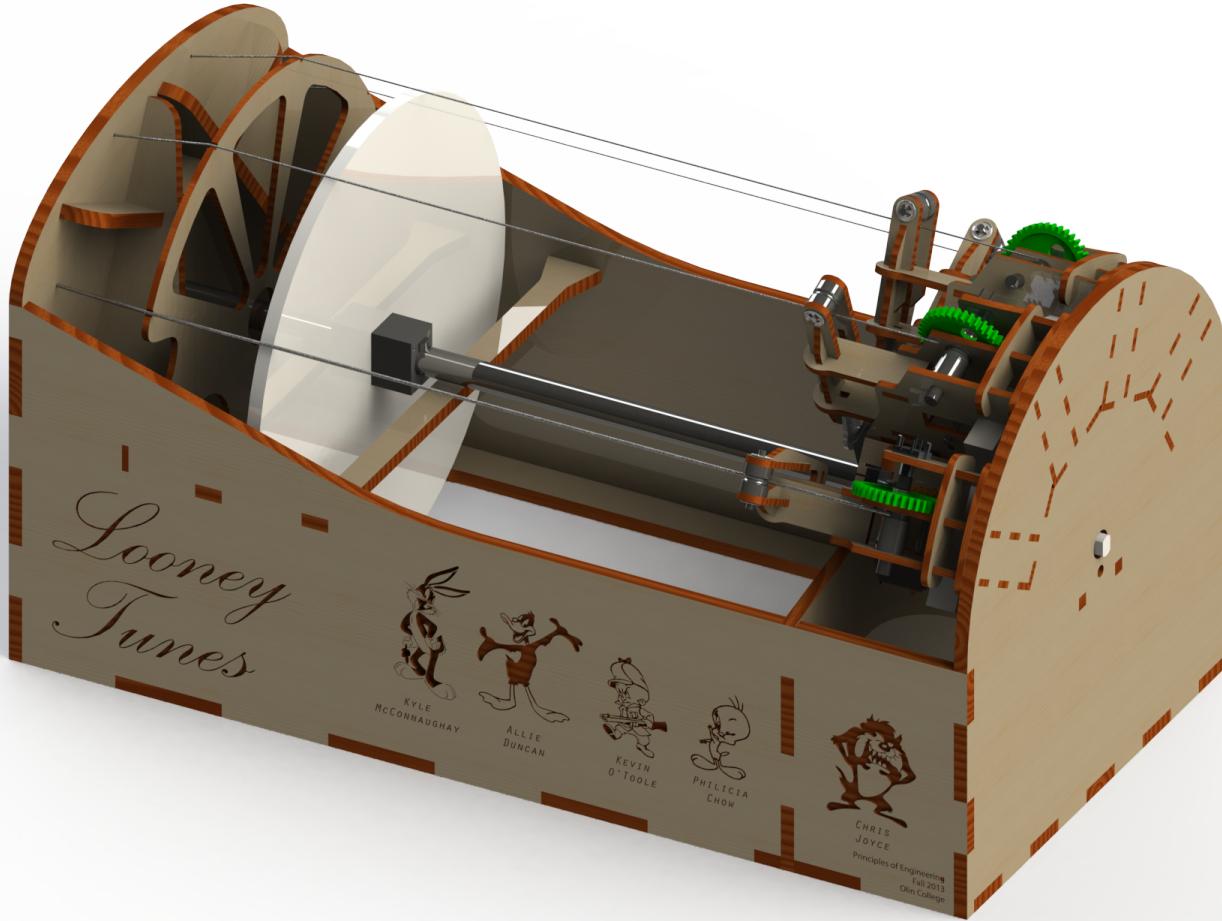
MORE DEMOS

WEBSITE

# SYSTEM DIAGRAM



# Mechanical overview

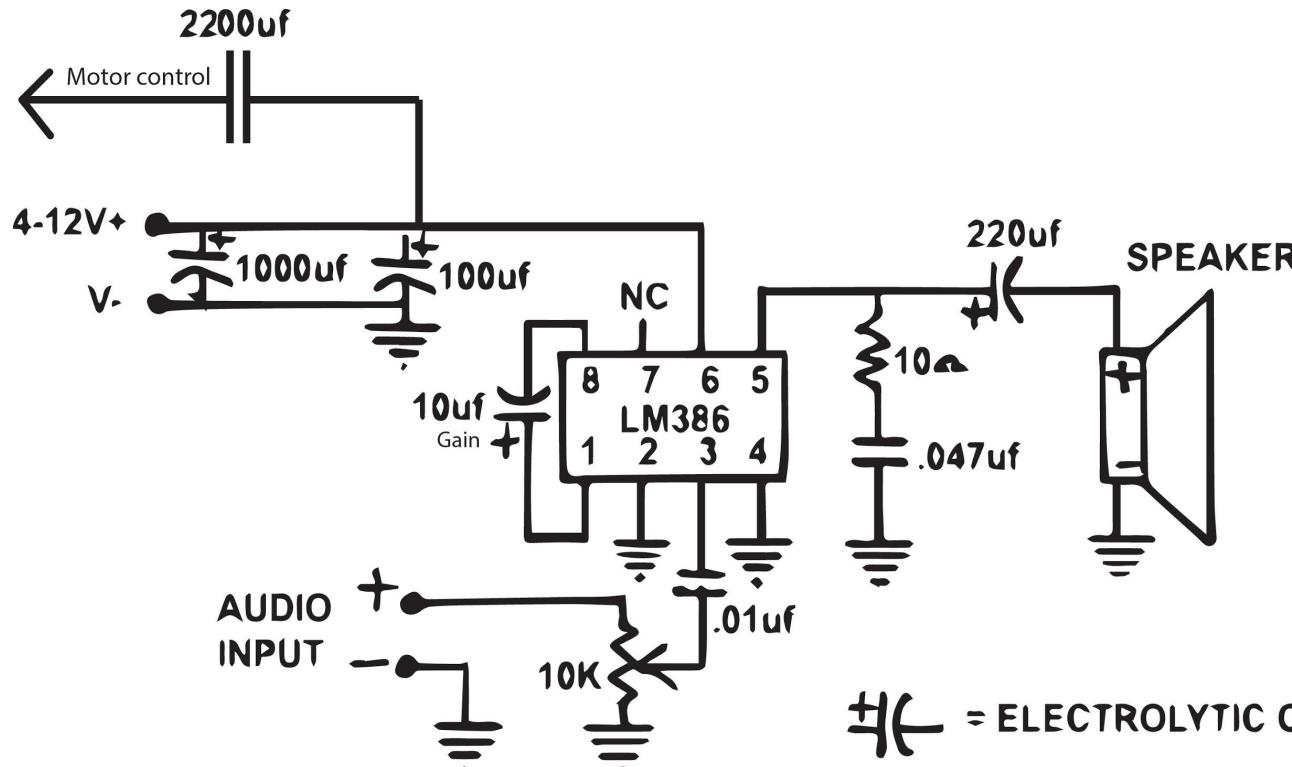


# ELECTRICAL overview

- Separation of logical power, audio signals, and motor power
  - Digital motor control through H-Bridges
  - Audio filtering and amplification
  - Servo control
  - Encoder feedback



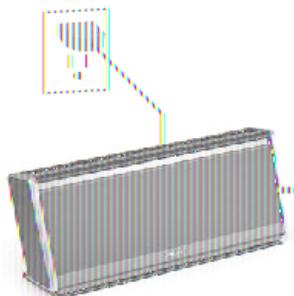
# AUDIO SCHEMATIC



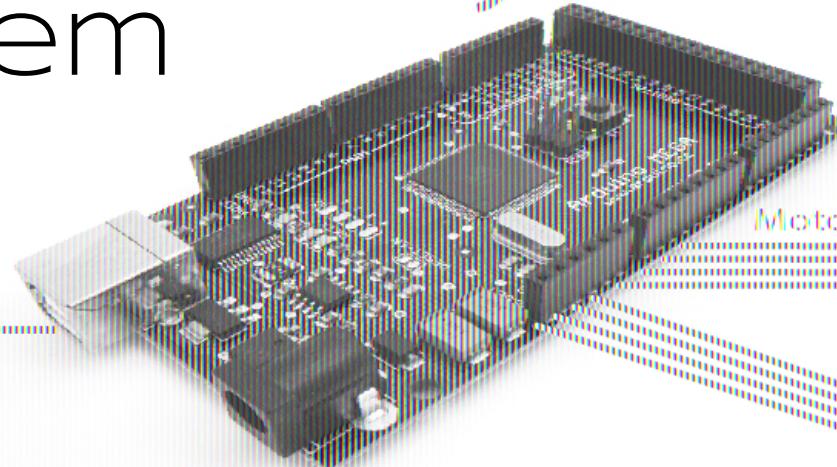
= ELECTROLYTIC CAP.  
 = CERAMIC CAP.

# ELECTRICAL SUBSYSTEM

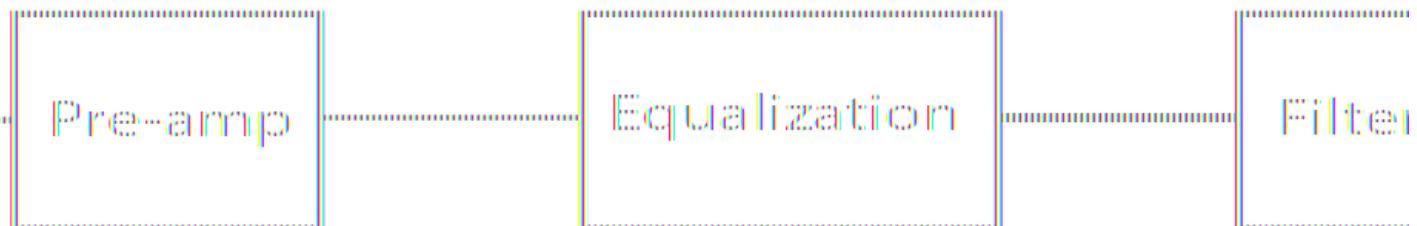
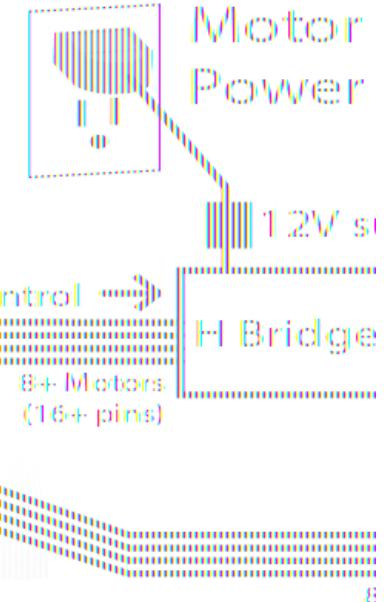
Logical Power



Amp and Speaker



System

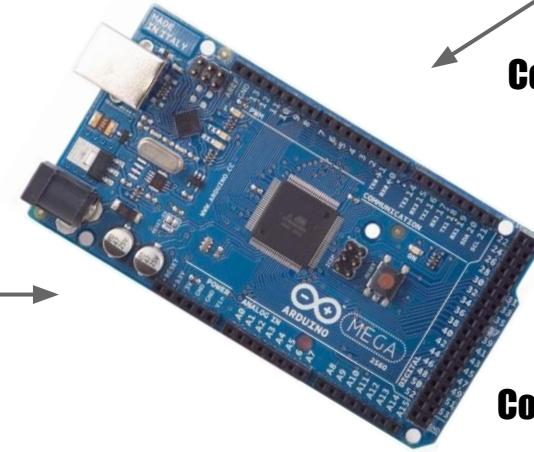


Equalization

Filter

# SOFTWARE & CONTROL

There are two control systems. The first is used before playing. The second is used while playing.



**Control System 1  
(manual)**



Feedback from the amplifier is used to “tune” the string before playing



**Control System 2  
(automatic)**



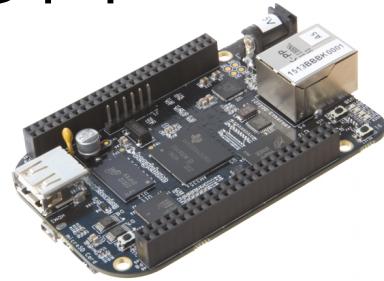
User input (dictating which song) is sent to the microcontroller

Microcontroller controls tension of the string by moving the motor

Encoder is used while playing to correct any error by the motor

# SOFTWare Transition

- BeagleBone ideal but unreliable
    - Chosen because it supports asynchronous, interrupt-driven development
    - Frequently froze or wouldn't accept pushes
  - Arduino Mega non-ideal but robust
    - Arduinos are unfortunately single-threaded
    - Never had any performance issues



# REFLECTIONS

- Rapid iteration
- Frontload technical risk in **all** disciplines
- Figure out the big picture first
- Take pictures/document every single step of the way
- Expect issues (such as our BeagleBone issue)



