Madam K Fortune Teller Construction

November 2014

Fortune Teller Construction

1   DISCLAIMER

This document and the attached drawings are not to be used for construction.

You are hereby authorized to view, copy, print, and distribute the materials from this document or the attached drawings subject to the following conditions:

- The materials are for your personal informational use only and not for commercial purposes.
- No liability in respect of any of the contents of such publications or use thereof is accepted and no warranties expressed or implied are made in relation thereto.

<u>Disclaimer</u>

ANY INFORMATION CONTAINED IN THIS DOCUMENT OR THE ATTACHED DRAWINGS ARE PROVIDED [AS IS] FOR YOUR INFORMATIONAL PURPOSES ONLY, WITHOUT WARRANTY OF ANY KIND, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. THE AUTHOR OF THIS DOCUMENT AND ATTACHED DRAWINGS WILL IN NO EVENT BE LIABLE FOR ANY DIRECT, INDIRECT OR PUNITIVE DAMAGE OF ANY KIND THAT RESULTS FROM THE USE OF OR INABILITY TO USE THIS DOCUMENT OR THE ATTACHED DRAWINGS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, GOODWILL, OR BUSINESS INTERRUPTION. THE AUTHOR OFFERS NO ASSURANCES, WARRANTIES OR GUARANTEES AS TO THE VALIDITY OR COMPLETENESS OF THIS INFORMATION. INFORMATION WILL BE CHANGED, UPDATED AND DELETED WITHOUT NOTICE.

2   Control

The fortune teller is controlled by two Arduino microcontrollers and a Pololu Maestro USB servo controller.

2.1   Arduino Uno R3 Controller (Coin Acceptor)

The system utilizes two Arduino Uno R3 controllers. The Coin Acceptor Arduino Uno R3 is programmed to monitor the status of the coin acceptor signal and determine when a valid coin was deposited. When a valid coin (nickel, dime, or quarter) is deposited, a show is requested. Since this is for home usage only, as long as any coin (no pennies) is deposited, a show will be played.

The Arduino requests the audio part of the show via a 4 digit binary code sent to the MP3 playback Arduino. The 4 bits are assigned values of 1, 2, 4, and 8. This technically gives you 15 combinations which leaves spares in the event I want to create more shows. I am not including all zeroes as a valid combination.

I created 3 different audio shows and the system will trigger them in the order of 1, 2, 3, 1, etc. The same show is never played back to back. The Arduino also requests the motion and lighting portion of the show by sending a serial request to the Pololu Maestro controller.

Programming Port

Arduino Uno

Power Jack
7 VDC – 12 VDC
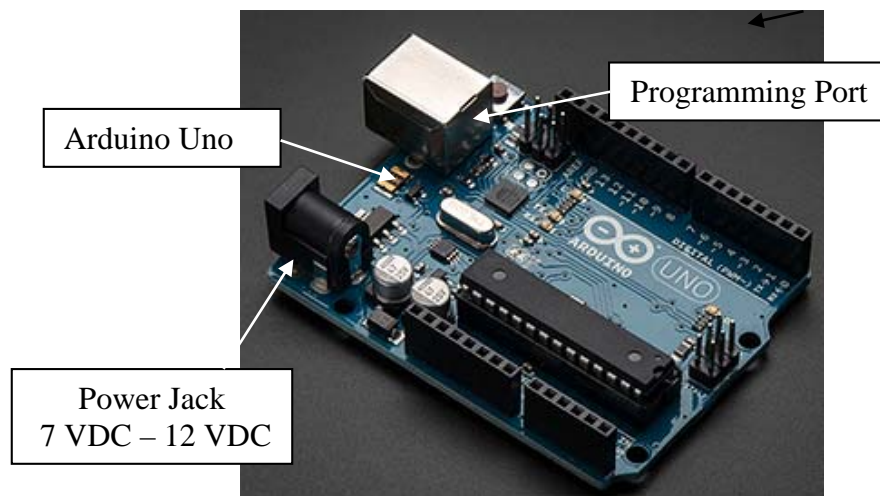
2.1.1 Audio Request

Based on the show number to be played, the Arduino sends the binary code via four pins configured as outputs to the MP3 player Arduino. The Arduino sends a show number playback audio show request. There are 3 shows programmed for audio. The show numbers increment as each show is played. When the system finishes playing show 3, the next show becomes show 1.

- Play Show 1 = Binary 1 0 0 0
- Play Show 2 = Binary 0 1 0 0
- Play Show 3 = Binary 1 1 0 0

2.1.2 Animation/Lighting Request

Based on the show number to be played, the Arduino sends a serial command to the Maestro controller to run a specified subroutine number. The Maestro controls RC relays and the elbow servo. The Arduino sends a show number playback animation/lighting request to the Maestro. There are 3 subroutines programmed for animation/lighting.

- Run Show 1 = Serial Write 0XA7 and 0X01 (Run Sub 1)
- Run Show 2 = Serial Write 0XA7 and 0X02 (Run Sub 2)
- Run Show 3 = Serial Write 0XA7 and 0X03 (Run Sub 3)

2.2 Arduino Uno R3 Controller (Audio Player)

The Arduino Uno that controls the audio has an Adafruit Music Maker MP3 player shield attached that is responsible for generating the audio for the system when requested to do so by the other Arduino. The MP3 file that is to be played is based on the Binary pattern received.

Since several of the pins are dedicated to the MP3 shield, I am using pins A0-A4 configured as inputs.
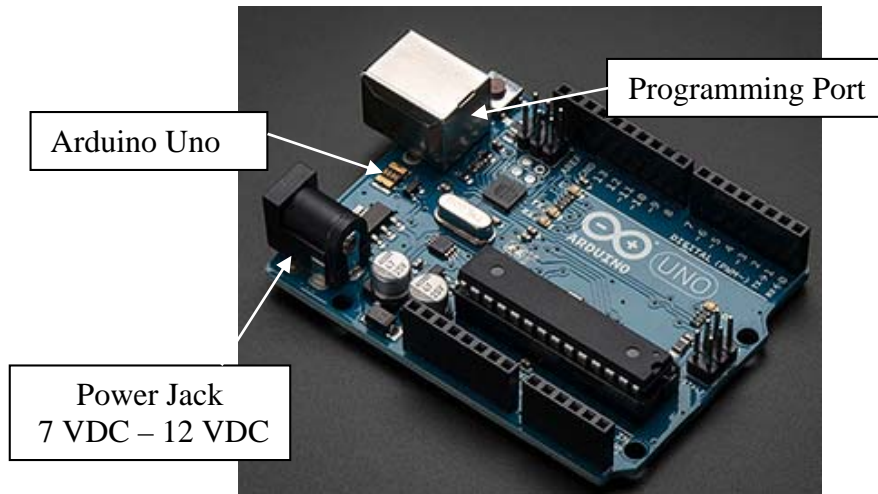
2.3 Proto-Screwshield R3 Kit for Arduino

I used a Proto-Screwshield for both Arduino controllers. The Proto-Screwshield kits were purchased from Adafruit.com and provides screw connections for all Arduino pins. This is handy while developing and provides a rugged connection for I/O pins. The Proto-Screwshield is shown below.
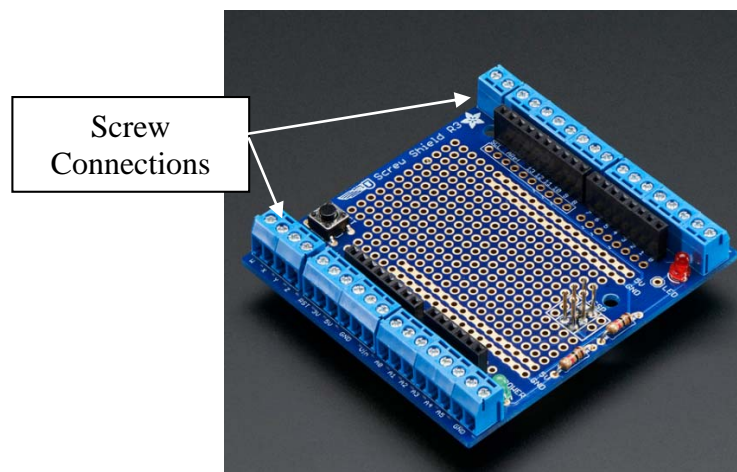
2.4 Arduino MP3 Music Maker Shield

The system utilizes an audio playback board for producing the audio. The audio board utilized for the fortune teller is an Adafruit Music Maker Shield. The audio board sits on top of the Arduino Uno R3 controller and interfaces with the controller via the .1" male stacking pins. The Music Maker Shield has an SD slot as shown below for storing the audio files.
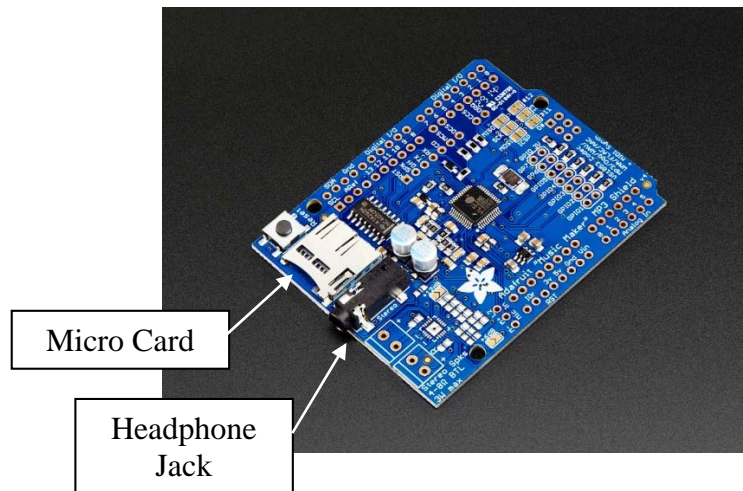

Micro Card

Headphone Jack

The audio files are in the MP3 format stored on the micro SD card. The file naming convention must adhere to 8.3 standard. The audio is routed through the right audio CD input jack on the Pyle PTA2 audio amplifier. The music maker is sold with a small amplifier section on the board. I blew one of these up by having the volume to high running larger speakers. The external amplifier design as shown here is your safest choice.

Audio Tracks

I created a script and had my wife record each of the scripts. I used Audacity software to modify the audio to my needs. I put the audio on the right track and some background music on the left track, the Audio is fed to both the main speaker amplifier and an inexpensive Lepai amplifier that has the voice track audio track routed to the PicoTalk controller so that the skeleton head will only react to spoken words. I used a second amplifier since I was getting a great deal of audio feedback and noise when splitting the signal from the Music Maker to the PicoTalk. The additional amplifier method has the advantage of allowing me to adjust the signal intensity going to the PicoTalk   as well as creating an isolated signal to avoid audio noise.

2.5 Pololu Maestro Servo Controller

The system utilizes a Pololu Maestro 18 channel servo controller. The Maestro servo controller allows control of the elbow servo via a scripting language located on the controller. The Maestro also allows you to run programmed subroutines based on serial commands sent from an external controller. The subroutine program is simple and allows you to have statements that perform delays, move the servo to a specific position, and set the overall velocity and acceleration for the servo. Since most of fortune teller requires digital contact closures for operation, I implemented the Pololu RC switch relays controlled by the Maestro to provide this feature. Since the Maestro programming language is flexible and easy, it made sense to have all of the animation and lighting control performed by the same controller.
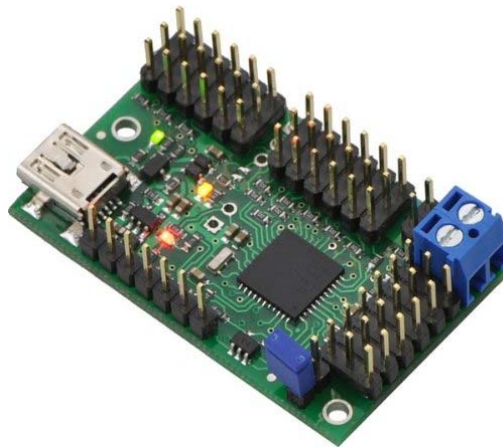
An Arduino Uno R3 transmits a show playback request via a serial line to run a specific show subroutine in the Pololu Maestro servo controller. The subroutine requested to run in the Maestro is based on the logic for the show to be played back.

The Pololu Maestro I am using is an 18 channel device. I only use 7 of the 18 channels for the fortune teller. Specifically, I am using the following channels.

Channel 0 – R1 Crystal Ball RC Relay
Channel 1 – R2 Relay (Spare)
Channel 2 – R3 Fortune Dispense RC Relay
Channel 3 – R4 Projector Relay
Channel 4 – R5 Relay (Spare)
Channel 5 – R6 PicoVolt Trigger Arm Sweep Relay
Channel 17 – Firgelli Forearm Actuator

2.6 Pololu RC Switch with Relay

The system utilizes Pololu RC switch devices. The devices incorporate a relay to disconnect power from the head servos, trigger the arm motion PicoVolt controller, provide power to the crystal ball and initiate the fortune dispenser. To maintain long servo life, I always kill power to servo motors when movement is not required. To perform this function, an RC switch is controlled by a channel on the maestro and the servo power wire is placed in series with the normally open contact of the relay. This provides a method to the servos after the show is complete and save wear and tear on the servo and eliminating potential noise from servo hum.

10 AMP Omron
G5LE-14-DC5
RC Controlled
Relay
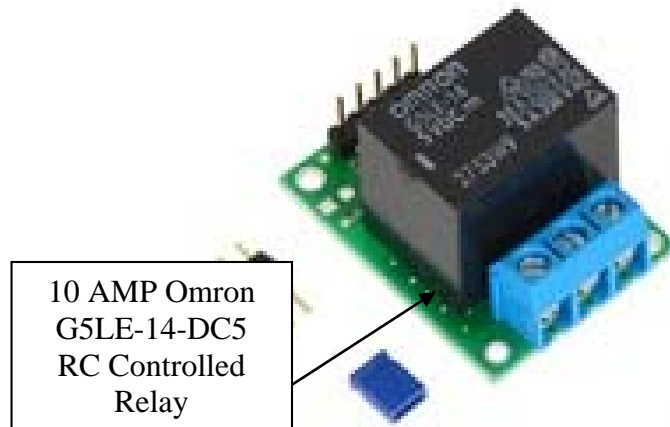
Since I am using several of the RC switches, I wanted to Din rail mount them to my control panel. The mounting holes accept 2-56 screws and standoffs. I installed standoffs on a piece of plain PCB board and the PCB board was attached to an Arduino Din rail mounting kit I purchased from rugercircuits.com. I purchased the stand offs and screws from Pololu.com. This method allowed me to snap a group of relays on the control panel Din rail. The RC switch with the standoffs installed is shown below.

2-56 standoffs

2-56 screws
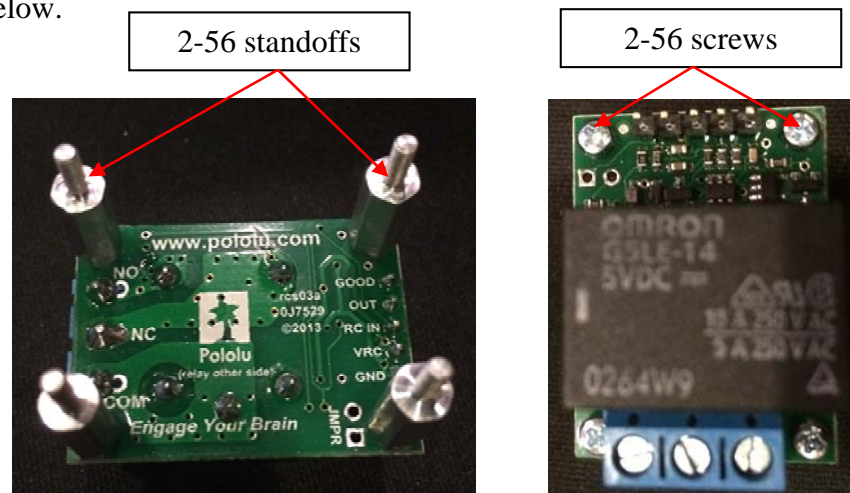
2.7 Coin Acceptor

The system utilizes a coin acceptor that was purchased at Adafruit.com. The coin acceptor can be programmed to recognize four different coin types. I programed it to recognize nickels, dimes, and quarters.

The acceptor requires 12 VDC to operate. A four wire harness attaches to the unit.

Red = +12VDC
White – Coin Pulse
Black – Ground
Gray – Not Used

Once programmed, the white wire puts out a pulse train that corresponds to a coin that was detected. I programmed the acceptor as follows:

Nickel = 6 pulses
Dime = 4 pulses
Quarter = 2 pulses

I have the switch located on the side of the acceptor set for slow pulses and normally open. Initially, I used the white coin pulse wire connected to input 2 (interrupt 0) and configured the input as a pull-up input. This worked well about 70% of the time but I would get erroneous pulses on the Arduino pin and a show would get triggered when a relay de-energized or some other noise induced interference would occur. I created the fortune teller for personal use and I wanted a show to play if a nickel, dime, or quarter were inserted. I used the coin pulse wire connected to pin 3 and brought this out to an Adafruit Art controller relay timer board. The relay of the Art Controller

board connects the ground to pin 3 of the Arduino. I also added some logic that must see the input high for at least 150 scans before determining a coin was detected.

Anytime a valid coin is detected, the coin pulse triggers the Art Controller board and energizes its relay for 1 second. The Arduino detects that input 3 went low for a time duration and a show is triggered.

The board shown below has excellent instructions and a highly recommended product.

Timing Duration

Relay
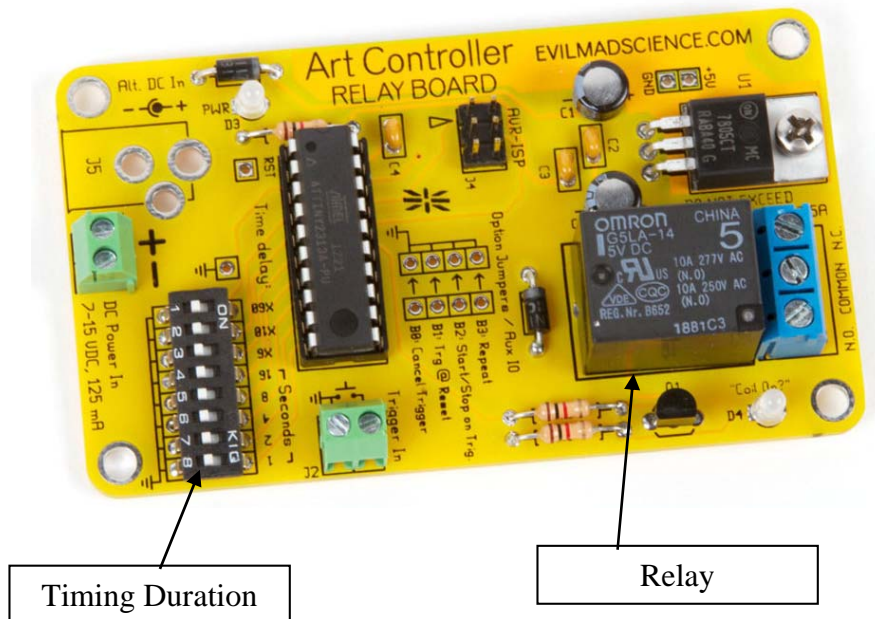
2.8 PicoVolt

The system utilizes a PicoVolt motor controller that can be purchased at
FrightProps.com. The controller allows up to 66 seconds of animation control to be
recorded for a motor via the speed/direction dial on the controller. Once you have the
desired motor motion recorded (variable speed and direction for 66 seconds), you can
replay the animation sequence by triggering the device. If you have a parking type
motor, the PicoVolt can be setup to return to the park position. The unit has several
modes of operation and for the money, it is an excellent device. If you need a motor,
FrightProps.com can supply this as well.

Direction/Speed
Pot

Program
Pushbutton

PicoVolt Motor
Controller
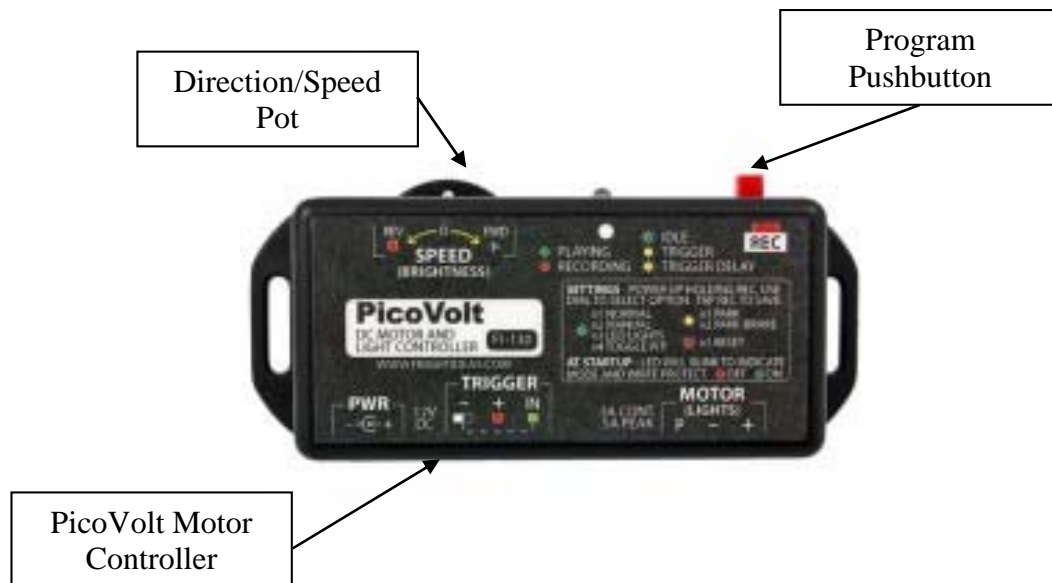
2.9 Arm Side to Side Motor

The system utilizes a parking type wiper motor for achieving the arm side to side
motion. The motor was purchased at FrightProps.com. The wiper motor has an
integrated gearbox to provide sufficient torque for the application.

Parking Motor
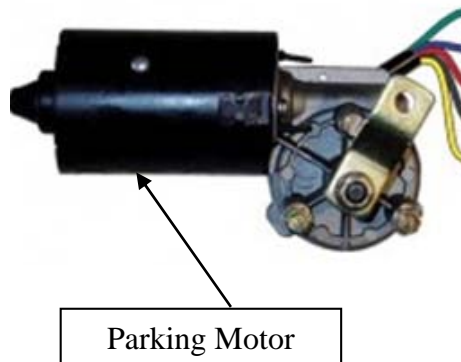
The motor is used to open and close a hinge in a reciprocating motion. The arm of the fortune teller is connected to the hinge. This motion provides the side to side effect of the fortune teller waving a hand over a crystal ball. An actuator is used to raise the forearm/hand so the arm does not collide with the crystal ball. The motor assembly needed to be mounted to a motor mounting plate. A mounting plate was fabricated using an electrical box cover. The cover that I used has a knockout hole for conduit. The conduit hole was used to allow the motor shaft to protrude through the plate. Three holes were drilled in the plate to accommodate the mounting of the motor assembly. The motor attached to the mounting plate is shown below.

Mounting Plate

Parking Motor
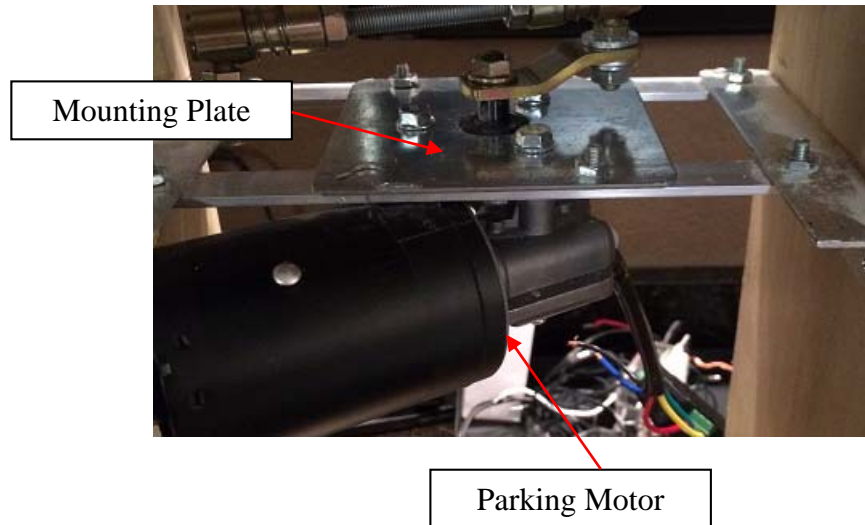
To provide the cam motion required, ball joint linkage was purchased from FrightProps.com to allow connection between the motor and hinge.
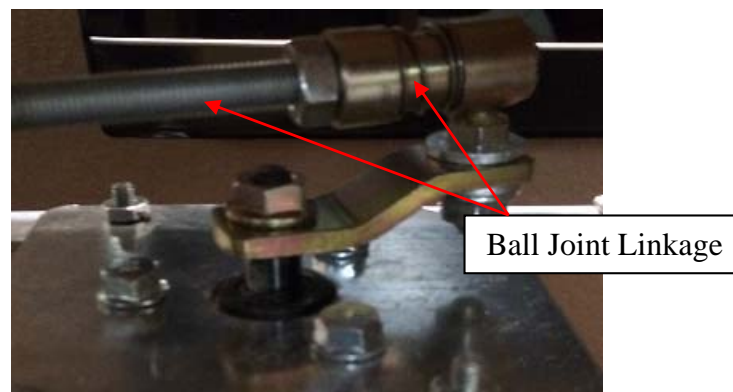
Ball Joint Linkage

An L bracket purchased from Home Depot was attached to the hinge to provide a surface for the other end of the ball joint linkage to connect to.

2.10 Forearm Actuator

The system utilizes an actuator purchased from Firgelli.com that raises the forearm/hand assembly so when the side to side motion is activated, it appears the arm is sweeping over the crystal ball. Raising the forearm/hand allows it to appear the fortune teller has raised the arm up so that the back and forth motion can commence.

The Firgelli actuator used is a L16-50-150-12-P. The actuator is connected to a Firgelli L.A.C. board. The Linear Actuator Control Board is a stand-alone closed-loop control board specifically designed for Firgelli "P" series actuators. The board allows a connection to the Maestro servo controller. The Maestro can control the linear actuator just like a hobby servo. The Linear Actuator Control Board has a USB connection that allows a 32 bit computer to run software to cycle and setup parameters for the actuator. The board can also be configured via the potentiometers on the board.

Linear Actuator

Linear Actuator
Control Board
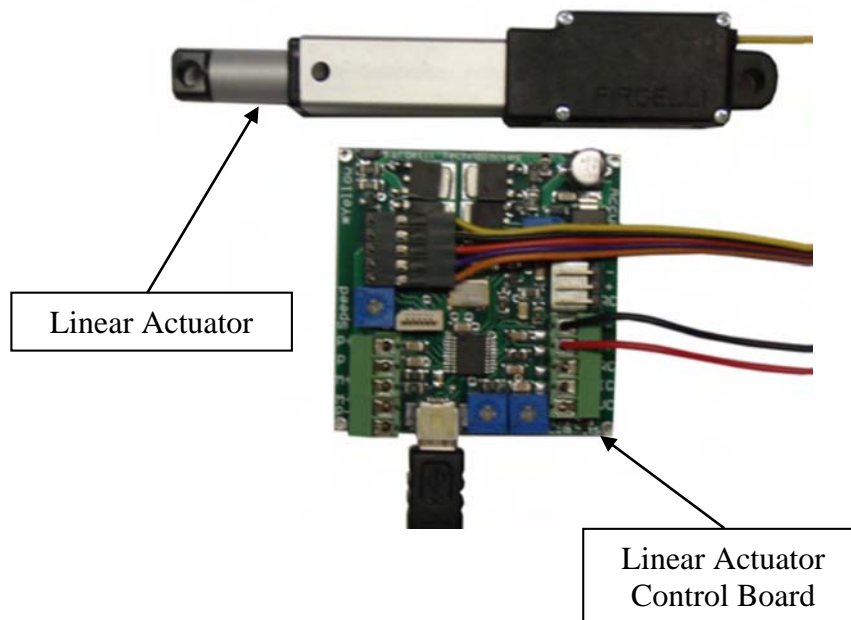
The Linear Actuator Control Board is housed in a LAC board case purchased from Firgelli.com. The LAC board case is shown below.

Control Board
Housing
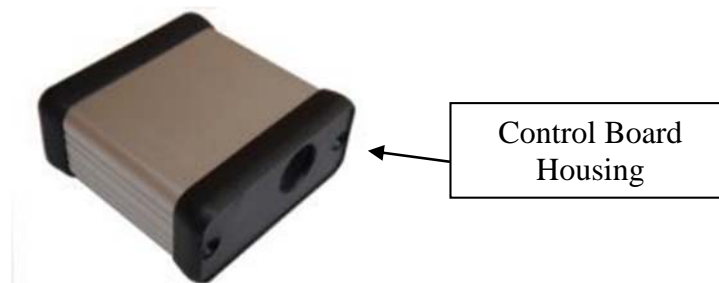
2.11 Audio Amplifier

The system utilizes a Pyle PTA2 audio amplifier to play the audio through the speakers. An amplifier with balance control was chosen since I wanted to have the voice track on the right channel and the ambient audio (creepy music) on the left channel. The left channel audio is set very low and only adds to the effect. The voice was also separated since the PicoTalk controller for the skeleton head required a signal that was only the voice. The amplifier was purchased from Amazon.

Speaker Audio
Amplifier

The Lepai amplifier shown below is used to drive the PicoTalk head servo controller.

PicoTalk Audio
Amplifier
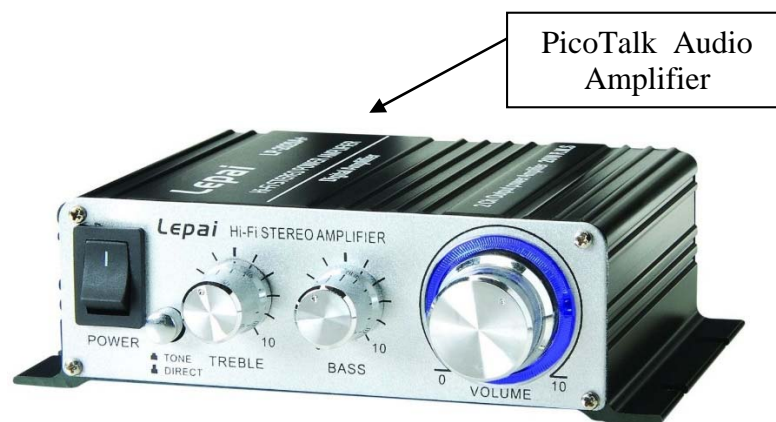
2.12 Speakers

The system utilizes two Rockford Fosgate R14X2 4-Inch full range coaxial speakers. The speakers are mounted to the front of the fortune teller. The speakers were purchased from Amazon.

2.13 Star Field

The ceiling above Madame K is comprised of a star field. This effect was created by utilizing a 10 watt LED fiber optic illuminator and Starpak 48 Strand Fiber Optic Cable. The items were purchased from Wiedamark.com. The fiber cable is pat number MK-Starpak48. I purchased a 14 foot quantity and cut it to length. The fiber optic illuminator is part number MK-LEE-1001-1.

The illuminator is shown below.

The fiber optic cable is shown below.
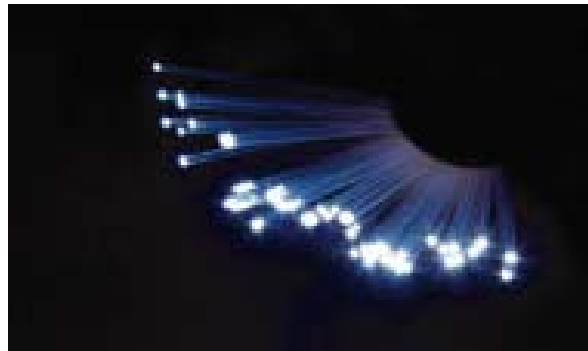
The star field was created by cutting a board the exact dimensions of the fortune teller booth. I used ceiling texture spray I got from Home Depot and sprayed the board with the texture on the side the guest will see. I made a grid on the other side of the board. In each grid (4 total) I drilled 12 random holes to accommodate the individual fiber strands. I used quick grip adhesive (home Depot) and glued the stands from the top side (not the interior of the booth). Make sure the strands protrude through the holes in an extra-long fashion since these are clipped after painting the ceiling and you will not need to worry about them when painting.

Here is a picture of the textured ceiling (un-painted) and the fiber poking through the ceiling. The fibers will be trimmed after painting.

3   Electrical Assembly

The electrical components are mounted on an aluminum sub-panel that is accessible by opening the speaker/coin acceptor door. The materials used for the electrical assembly are listed in the following section.

3.1 Electrical Component Mounting

The electrical components are mounted to the sub-panel utilizing Din rail. Din rail and wire duct is mounted to the sub-panel using rivet nuts and screws. The rivet nut is a simple way to install a threaded hole in steel or aluminum. Drill a hole, spin the rivet nut onto the threaded shaft of the tool, insert into the hole, squeeze the trigger, spin the knob on the end of the tool counter clockwise to remove the tool from the insert and you have a threaded insert. Much easier than tapping holes. I used a Marson 39302 poly nut thread setter kit I got on Amazon. The kit is shown below.

The kit comes with some rivet nuts but you an order additional ones on Amazon as well. I used 0-32 rivet nuts exclusively. I also used 10-32 rack screws I purchased from Amazon. I used Raxxess brand screws. With the tools for fabrication in place, it is now time to fabricate. The attached drawings illustrate the layout of the control panel so you will want to refer to the electrical drawings for details.

The picture below shows the early stages of the subpanel construction with the Din rail, wire, and components mounted.

Wire Duct

Din Rail
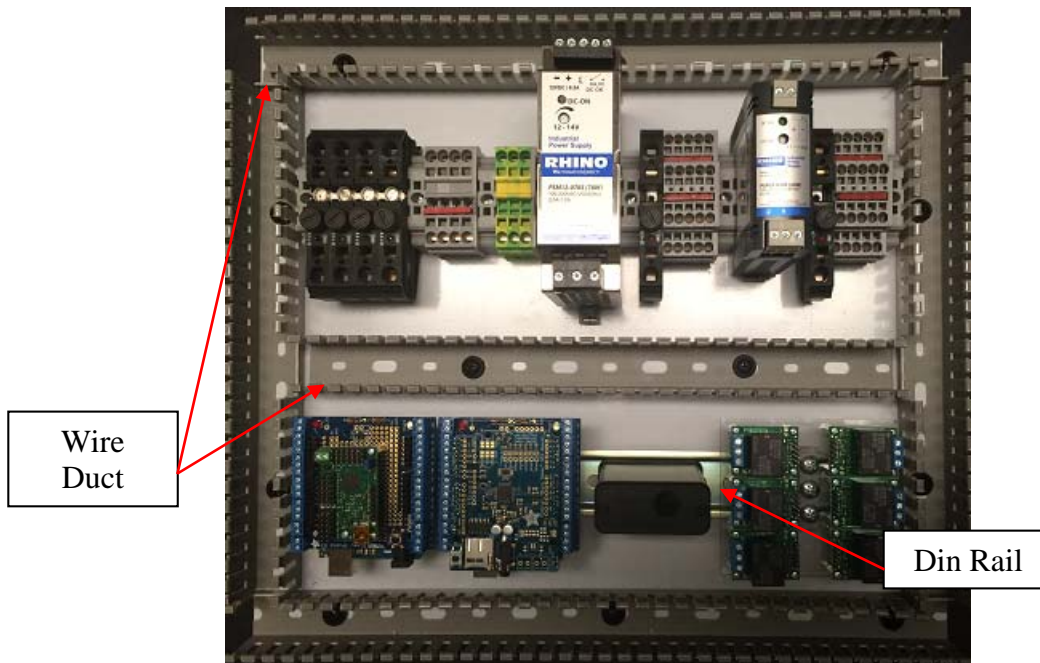
The picture shown below gives you an idea of what the rivet nut looks like on the back side of the panel. You can see the 10-32 screws protruding through the rivet nuts.

Rivet Nuts
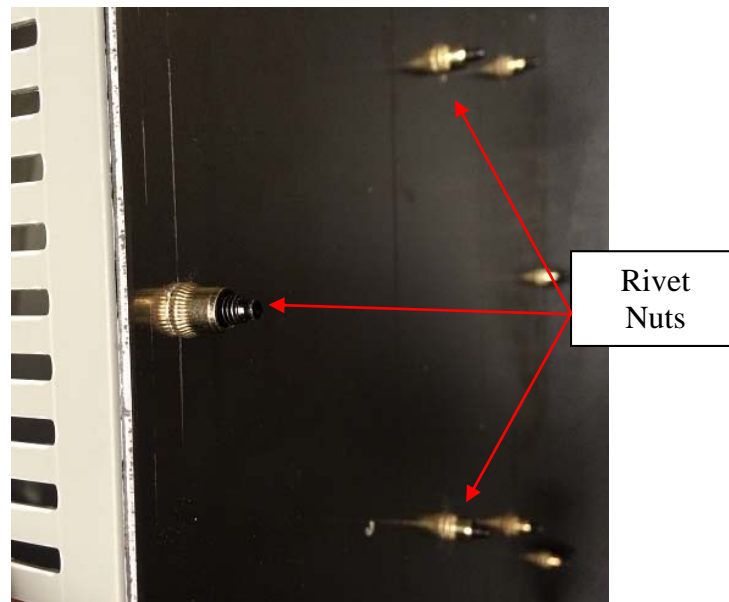
The picture below shows the subpanel with a majority of the wiring completed. As you can see, I always use wire ferrules for each wire and I label each wire using a heat shrink label. The labeling allows me to wire the system per the attached drawings.

120 VAC Fuses

12 VDC Power Supply

12 VDC Fuse

5 VDC Power Supply

5 VDC Fuse

Firgelli LAC Board in Enclosure

Arduino Unos and Screw Shields

Pololu RC Relays

Maestro

Music Maker Shield
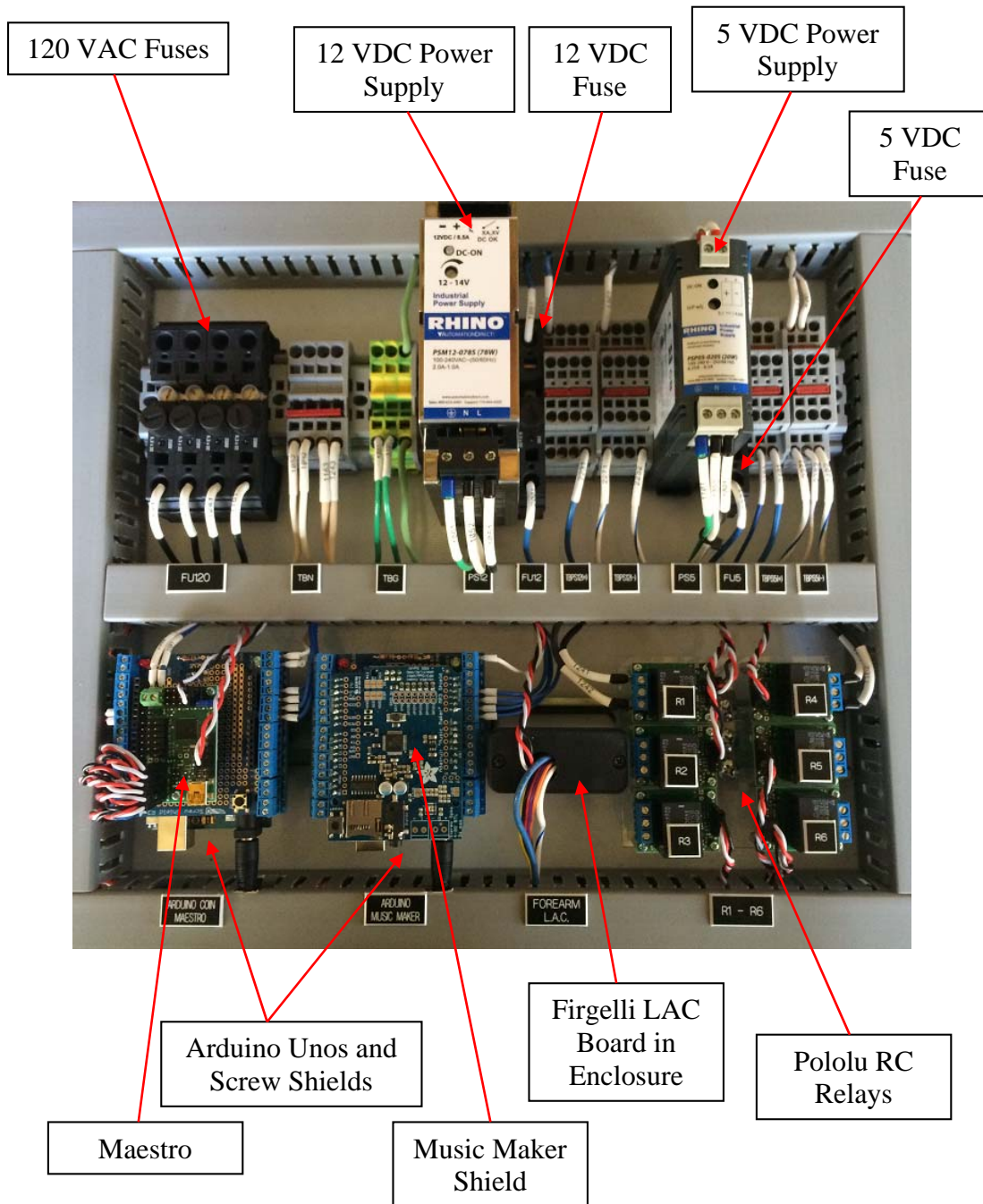
3.2 Electrical Wiring

As illustrated on the attached electrical drawings, a color code, wire awg, and wire number is associated with each wire. The drawing package will use the following terminology for a wire.
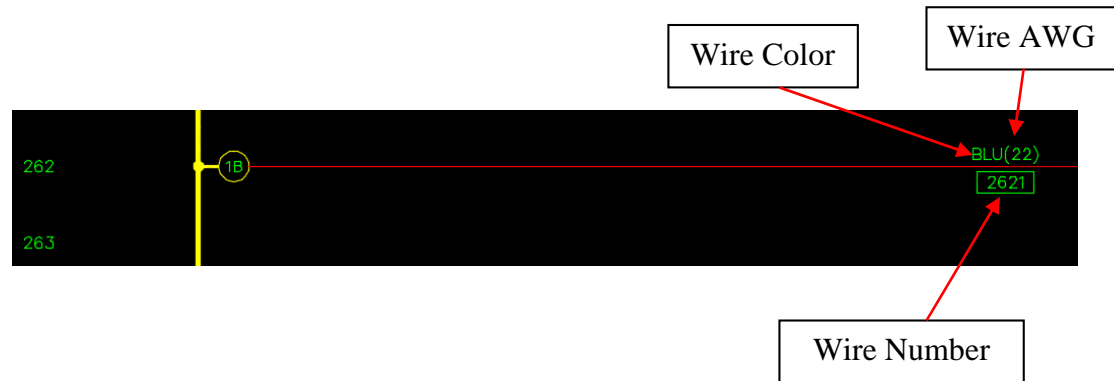
Each wire receives a ferrule. The wire ferrule is just a connector that is crimped on the wire which is used to terminate stranded wires. The wire used for the system is all MTW (machine tool wire) stranded and utilizes UL508A color codes where possible.

The ferrules, wire, and ferrule crimper can be purchased from automation direct or your favorite supplier.

The attached electrical drawings illustrate how the Madame K fortune teller was wired. The drawings will show all point to point connections.

3.3 Forearm Actuator

A Firgelli actuator was used to allow the right forearm to raise and lower. Raising the forearm is required so that the hand can be raise above the height of the crystal ball when the right arm sweeping motion is in effect. The picture below is shown with the actuator fully extended (arm fully down).

Forearm
Firgelli
Actuator
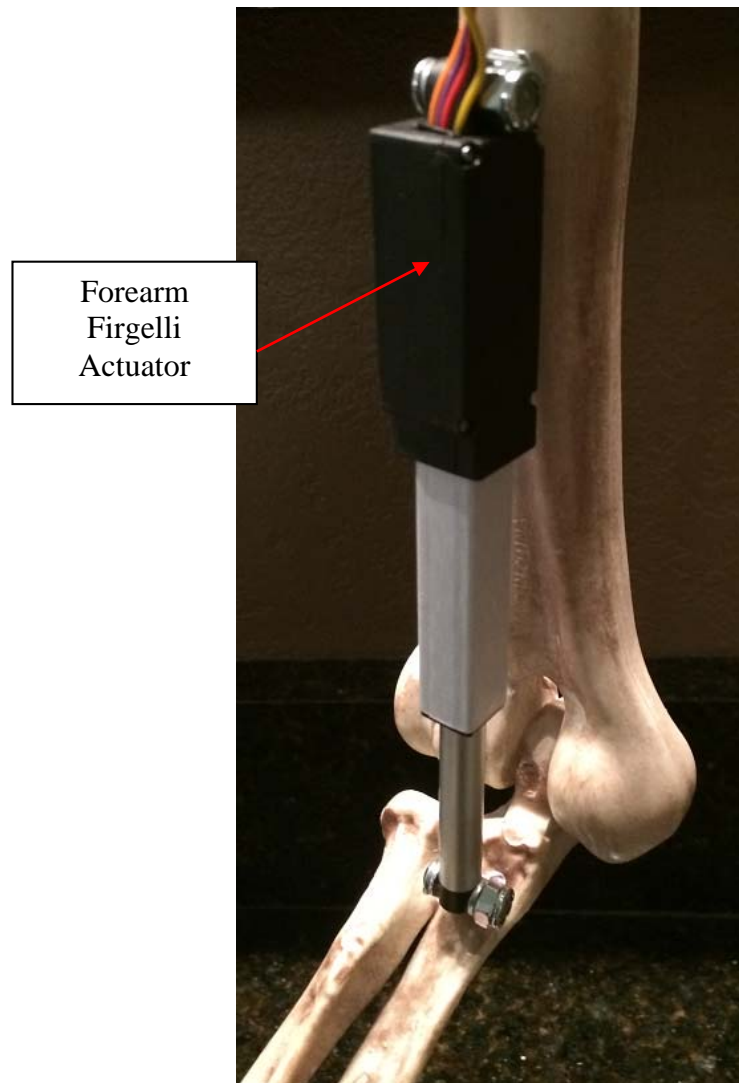
4   Fortune Teller Components

4.1   Skeleton Body Parts

The arms, hands, and neck vertebrae were scavenged from a Pose-n-Stay skeleton purchased from Costco. The Pose-n-Stay skeleton is shown below.

4.2   Skeleton Head

The skeleton head was purchased from FrightProps.com. The head was used since most of the mechanical animation components are included and already designed. The head has moving eyes, moving mouth, head turn, and LED eyes. The controller for these components is a PicoTalk form FrightProps and included with the animated head. Just feed audio to the PicoTalk and the PicoTalk will do the rest. Look for talking skulls on their website and that is what I used for this project.

4.3 Fortune Teller Accessories

I placed various fortune teller accessories around Madame K to add the theme of the fortune teller.

Tarot Cards

I purchased some very nice looking Tarot cards from Amazon. I purchased the "Easy Tarot: Learn to Read the Cards Once and For All!" By Ciro Marchetti and Josephine Ellershaw.

Aladdin Genie Lamp

I purchased a nice brass Aladdin Genie Lamp from Amazon. The lamp is small (5") but I wanted the lamp to be small since I do not have much real-estate inside the cabinet and it is only there for effect.

Genie Bottle

I purchased a replica "I Dream of Genie" bottle from JeannieBottles.com. The bottle is nestled in the left arm of Madam K.

4.4 Fortune Teller Clothing

I was dressing the skeleton for the Madame K clothing so the hands and face would remain as skeletal items.

Hat
I purchased an Arabian Princess Hat from Amazon. The hat seems to run small so it was a perfect fit for the skeleton head. The hat was purchased from Amazon Vendor "Jacobson Hat Company".

Arm Cuffs
The joint between the hands and the forearm needed to be hidden. Therefore, I purchased BellyLady Belly Dance Arm Cuffs from Amazon. These are perfect for hiding the hand/arm connection and adding to the theme. I used on as an arm cuff and the other as a hair braid decoration.

Blouse

I chose a blouse from Amazon to cover Madam K. The blouse is a "503 Rich Luxury Gypsy Medieval Renaissance Blouse Top".

Blouse Under Garment

My wife had an old black shirt that was cut to and placed under the upper section of the blouse.

4.5 Fortune Teller Jewelry

<u>Jewelry</u>
I used earrings and a gold coin necklace from Amazon as shown below. The item is a "BellyLady Belly Dancing Gypsy Jewelry – Gold Coin Necklace and Earrings, 3pc Set".

<u>Bracelet</u>
I used a bracelet I purchased from Amazon as shown below. The item is a "BellyLady Belly Dance Triangle Slave Bracelet With Gold Coins, Gypsy Jewelry".

5   Cabinet Construction

I wanted a nice looking cabinet to house Madam K. The cabinet has two doors on the front that are used for mounting the speakers, coin acceptor, and fortune dispenser. The electrical panel is also housed behind the lower door and mounted on drawer slides so the panel can be accessed easily. The material used to construct the cabinet was sand ply plywood, poplar 1 x material in 1x2, 1x3, and 1x4 widths. The crown molding on the top of the cabinet is 3'' crown molding and the glass area is not glass but polycarbonate AR sheets purchased from tapplastics.com. The entire cabinet is sitting on (4) rollers purchased from Harbor Freight Tools.

Front View   **Side View**

5.1 Construction Detail

I first made some plans regarding dimensions and cutouts required for the cabinet. I did this so I can be sure my fortune teller dimensions fit within the enclosure as well as having a plan for the build. A construction drawing is shown below and can be downloaded with this document. Dimensions of the cabinet and the lumber utilized are detailed in the drawing.
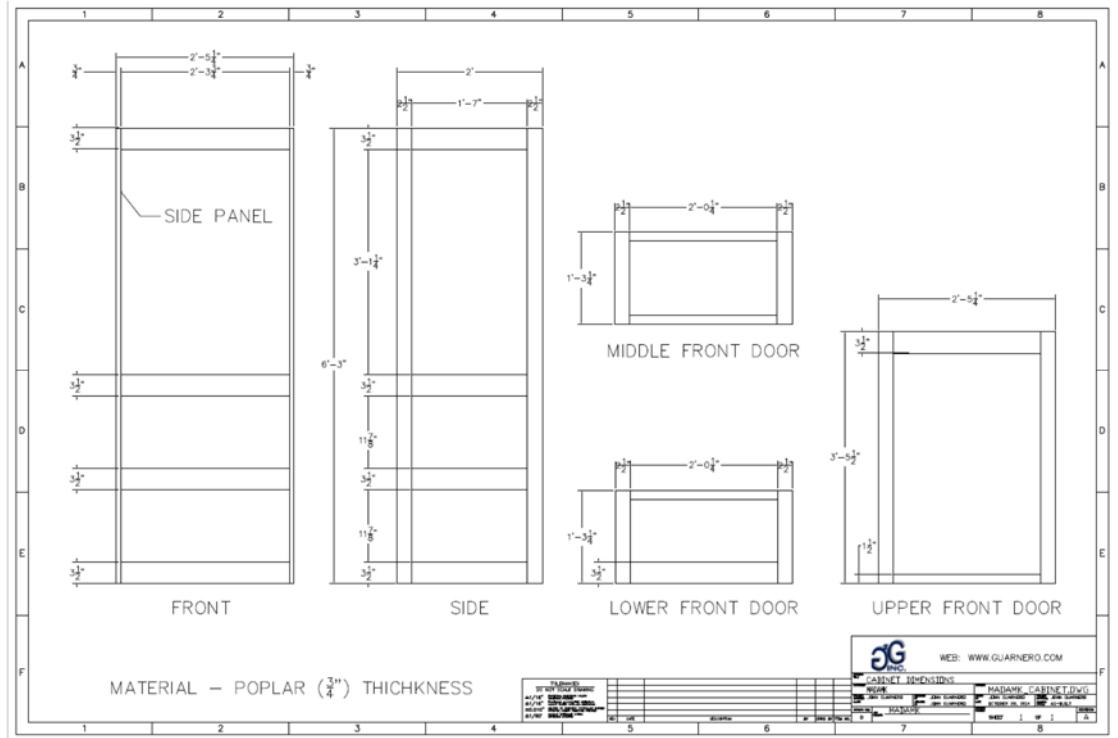
The cabinet is screwed and glued so it will not will not come apart and is very sturdy. I used pocket holes for the screws. I accomplish this with a Kreg jig. I use the Kreg jig for all wood working projects. It is easy to use and makes a rock solid hidden connection that is much easier than biscuit joining. You can get the Kreg jig, screws, square head bits, and everything you need at Lowes or Woodcraft.

5.2 Door Detail

The two lower doors have a solid panel in the center. This was accomplished by creating the door frame from the poplar 1 x material and using the Kreg Jig to join the rails and stiles. Once the frame was constructed, I used a rabbit bit in the router to create a rabbit for the 1/8" sand ply plywood panel to sit in the frame. The panel was glued and tacked to the frame.

⚠Be sure the Kreg Screws clear the rabbit bit before routing!!!

The large upper door that holds the Polycarbonate AR panel was constructed in the same manner as the lower doors.

On all doors. I routed the outside edge using a round over bit. On the inside area of the doors (rails and stiles), I trimmed the inner portion with a piece of molding to give the appearance that the doors were constructed in the traditional manner. The molding detail is shown below.

Molding applied to rail and stiles

5.3 Side Detail

The lower section of the side panels were fabricated to mimic the look of the front lower doors This was accomplished by using the Kreg Jig and a rabbit bit in the router to create a rabbit for the 1/8" sand ply plywood panel to sit in the frame. The panel was glue and tacked to the frame.

5.4 Front Detail

The front of the cabinet just joins the side with (4) pieces of 1" x 4" poplar. As always, the Kreg jig was used to join the rails (horizontal pieces) to the side sections.

5.5 Shelves and Madam K Base

The shelves, Madam K base, and bottom were made from ¾" sand ply Plywood. The Kreg Jig was used to create pocket holes on the lower sections of each panel. The panels were screwed to the sides and front rails

5.6 Speaker Cutout

The Rockford Fosgate R14X2 4-Inch full range coaxial speakers were mounted to the bottom door of the fortune teller. The picture below shows the door cutout for the speakers. The picture below shows what the door looks like with only one coat of paint.

The picture shown below illustrates what the finished door with speakers mounted looks like.

The speaker cutout was made using a MK Morse TA62 3-7/8-Inch Hole Saw with Attached Arbor purchased from Amazon.

5.7 Coin Acceptor Cutout

The coin acceptor did not have a cutout template so I created a bare minimum cutout to allow the coin acceptor to mount to the door. The picture below shows the cutout on the door.

Coin Acceptor Cutout
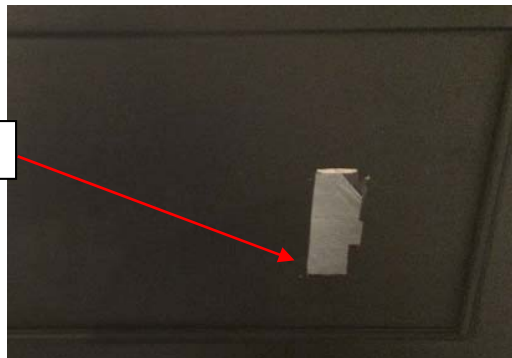
Coin Acceptor Mounted

5.8 Fortune Dispenser Cutout / Mounting

A rectangular hole was made in the door to accommodate the fortune being dispensed through the door. The hole was made after the fortune dispenser mount was fabricated. Once the mount was complete, the dispenser was attached to the mount and a pencil was used to make a mark on the back of the door where the fortune would come out. The dispenser and door was removed and a rectangular hole the size of the inner rectangle of a light switch plate was made. The cover for the dispenser was purchase at Lowes and is one of the fancy wall switch covers. I used this method instead of a small slot so that the alignment of the dispenser was not critical. The picture below shows the wall pate covering the dispenser hole in the front of the door.

A simple bracket was fabricated to mount the dispenser on the back of the door. The bracket allows the dispenser to be removed with no tools in a fast manner. A 3/4" wide by 1/8" thick piece of aluminum was cut to span the door width but small enough to allow the door to close normally. Two holes were drilled on each end of the aluminum piece and the aluminum piece was attached to the door with screws. I use two angle brackets that I attached to the aluminum about 1" apart to allow the dispenser to be easily hung on the brackets. The brackets needed to be in a "U" shape so I bent them using a pair of channel locks. I glued a small leftover piece of 1/8" sand ply to the lower part of the door where the bottom of the dispenser rests on the door to shim the dispenser out and keep it level. The pictures below should help clarify my configuration.

1/8" aluminum flat stock

Brackets Mounted to 1/8" aluminum flat stock

Dispenser

Art Relay Controller Board

1/8" aluminum flat stock

Brackets Mounted to 1/8" aluminum flat stock

Back of upper door

Dispenser
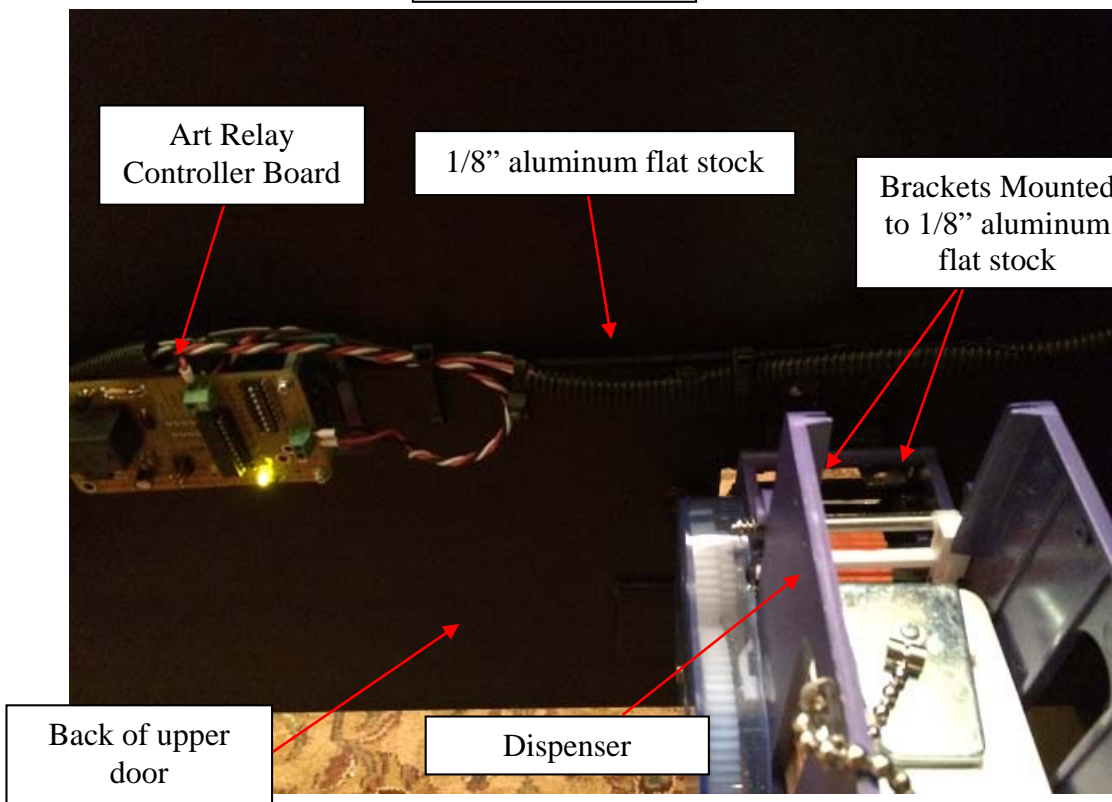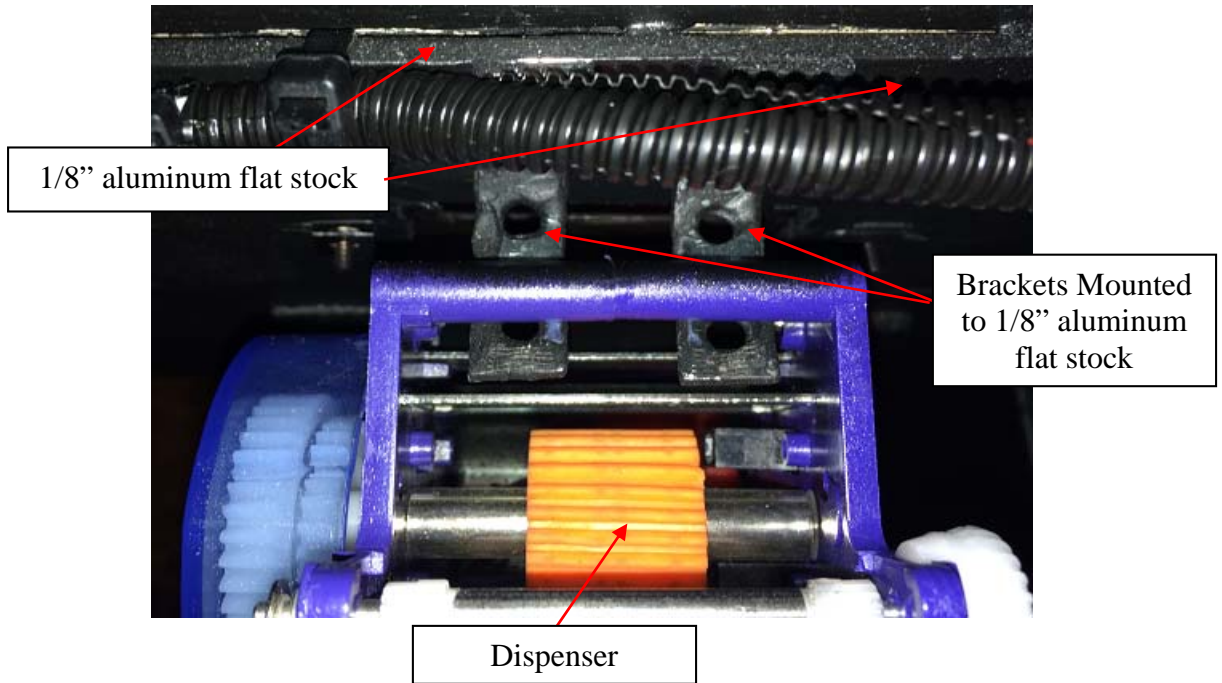
5.9 Polycarbonate AR Sheets

I used polycarbonate AR sheets instead of glass for the windowed areas of the cabinet. The polycarbonate AR sheets were cut to my exact dimension and purchased from TapPlastic.com. I chose polycarbonate AR since it has the same properties as polycarbonate with abrasion resistance. This means that I can drill the material easily and it will not scratch like other materials used to emulate glass. This stuff is great because it does not scratch if you use paper towel and glass cleaner. It is not advised to use paper towels but I have done this with no issue. Your mileage may vary.

I mounted the polycarbonate to the upper frame sections by drilling holes in the polycarbonate and using lose fitting screws. Polycarbonate will expand and contract with climate changes and you do not want to have cracking occur. I drilled the holes using the drill bits purchased at TapPlastic.com. They have various sizes and the bits are designed for drilling into this type of material which reduced the chances of cracking.

5.10 Stickers

As you can see, I used stickers on the cabinet shown below. The stickers were purchased from CarStickers.com and come in most any variety, font, color that you can imagine. Once you figure out the best method for putting the stickers on, it is easy. Order a spare one in case you screw one up.

Front   **Side**

Cost Sticker

The sticker shown above was not meant for a wood surface. Therefore, it took a great deal of patience to get the sticker applied. I applied the sticker a few hours after painting the door so the paint was dry but not cured. The sticker is now firmly affixed to the door.

5.11 Paint

The entire cabinet was painted using Behr Marquee Eggshell (MQ5-5) Limousine Leather paint (base 2453). The paint was purchased from Home Depot. This paint was the right color but the wrong sheen. We had Home Depot color match the paint so that we could get it in a satin finish. The color information is below.
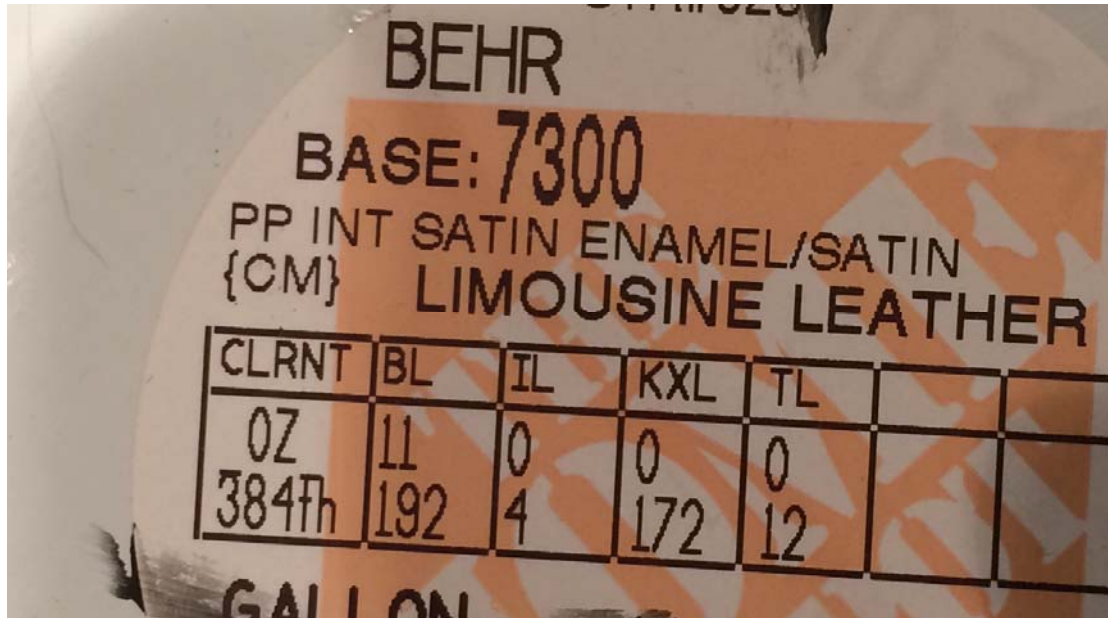
Three coats of paint were applied with foam rollers.

6   Fortunes

The machine will deliver a fortune after each reading. A portion of each fortune card that is dispensed is shown below.

<u>Fortune</u>
He who laughs last probably didn't get the joke.

<u>Joke Of The Day</u>
Q: What happens when it rains cat and dogs?
A: You can step in a poodle!

<u>Lucky Numbers</u>
3.1415 69 6.022 x 10^{23}

<u>Fortune</u>
When everything's coming your way, you're in the wrong lane.

<u>Joke Of The Day</u>
Q: What is the difference between an illegal immigrant and E.T.?
A: E.T. eventually went home!

<u>Lucky Numbers</u>
3.1415 69 6.022 x 10^{23}

<u>Fortune</u>
7/5th of all people do not understand fractions.

<u>Joke Of The Day</u>
Q: Do you know what the square root of 69 is?
A: Ate something.

<u>Lucky Numbers</u>
3.1415 69 6.022 x 10^{23}

<u>Fortune</u>
Two days from now, tomorrow will be yesterday.

<u>Joke Of The Day</u>
Q: How many ADHD kids does it take to screw in a lightbulb?
A: ...HEY LET'S RIDE BIKES!!

<u>Lucky Numbers</u>
3.1415 69 6.022 x 10^{23}

Fortune
If someone calls you fat, don't get angry... just turn the other chin.

Joke Of The Day
Q: Why does hamburger have lower energy than steak?
A: Because it's in the ground state.

Lucky Numbers
3.1415 69 6.022 x 10^{23}

Fortune
You're lucky numbers do not involve the digits 0-9.

Joke Of The Day
Two bytes meet. The first byte asks "Are you ill?" The second byte replies, "No, just feeling a bit off."

Lucky Numbers
3.1415 69 6.022 x 10^{23}

Fortune
Not everyone is perfect.

Joke Of The Day
Q: Why don't jokes work in octal?
A: Because 7 10 11.

Lucky Numbers
3.1415 69 6.022 x 10^{23}

Fortune
Download Error. Your fortune is temporarily unavailable.

Joke Of The Day
Q: Why are assembly programmers always soaking wet?
A: They work below C-level.

Lucky Numbers
3.1415 , everyone likes pi (∏)

Fortune
Life will be happy until the end when you pee yourself a lot.

Joke Of The Day
Q: How many hydrogen atoms does it take to make peroxide?
A: Oh, two.

Lucky Numbers
3.1415 , everyone likes pi (∏)

Fortune
The secret of staying young is to live honestly, and lie about your age.

Joke Of The Day
Q: Why do programmers always get Christmas and Halloween mixed up?
A: Because DEC 25 = OCT 31

Lucky Numbers
3.1415 , everyone likes pi (∏)

7   Program Listing

The code for the Arduino Mega, Arduino Uno, and the Maestro servo controller is shown in the following sections.

7.1   Arduino Uno R3 (Coin Acceptor)

The Arduino Uno R3 reads when the coin inserted input is connected to ground. The Arduino Uno R3 also determines the show animation and audio file to play and sends a 4 digit binary code to the Audio Player Arduino Uno R3 that represents the desired audio file to be played. In addition to sending the 4 digit binary code, it also requests a subroutine to be played to the Maestro. Refer to the electrical drawings for hookup information. Currently, I am only issuing three different shows but I have the ability to issue 15 different shows based on the binary code available. The Arduino code is listed below.

```
//serial to maestro config
#include <SoftwareSerial.h>
#define txPin 6
#define rxPin 7

SoftwareSerial mySerial(rxPin, txPin);

void setup()
{
// Serial.begin(9600); Enable For Local Serial Port Debugging

//Serial to maestro config

Serial.begin(9600);

//Delay on power up to avoid trigger of effect
delay(10000);

//Set pin 3 as the input from the ADAFruit Art Controller relay.  Relay triggered by
coin mech.
  pinMode(3, INPUT_PULLUP);

//define pins for software serial
pinMode(txPin, OUTPUT);
pinMode(rxPin, INPUT);

//define pins to send to music maker arduino
pinMode(14, OUTPUT);  //for binary, pin 22 is 1,pin 24 is 2,pin 28 is 4,pin 30 is 8
pinMode(15, OUTPUT);  //for binary, pin 22 is 1,pin 24 is 2,pin 28 is 4,pin 30 is 8
pinMode(16, OUTPUT);  //for binary, pin 22 is 1,pin 24 is 2,pin 28 is 4,pin 30 is 8
```

```
pinMode(17, OUTPUT);  //for binary, pin 22 is 1,pin 24 is 2,pin 28 is 4,pin 30 is 8

//set pins low
digitalWrite (14, LOW);  // 1 2 4 8
digitalWrite (15, LOW);  // 1 2 4 8
digitalWrite (16, LOW);  // 1 2 4 8
digitalWrite (17, LOW);  // 1 2 4 8

//Serial.println("John Guarnero Madam K Fortune Teller Coin Mech Arduno");


}
// show number to play on maestro and audio;
 int ShowNumber = 1;
 int Coin_Memory = 0;
 int Allow_Run = 1; //added alow run since when a relay de-energized, it would
sometimes initate a trigger like a coin was inserted.  This resolved any inductive kick
issues.
 int Coin_Detected_Debounce = 0;

void loop()
{

 int Coin_Detected = digitalRead(3);
   if (Coin_Detected == 0 )
   {
   Coin_Detected_Debounce = Coin_Detected_Debounce + 1;
   Serial.println(Coin_Detected_Debounce);
    }


   if (Coin_Detected == 1 )
   {
   Coin_Detected_Debounce = 0;
    }



  if (Coin_Detected_Debounce >= 150 && Allow_Run == 1)
{
 Coin_Memory = 1;

 delay (1000);
}

 {
```

```
if (ShowNumber == 1 && Coin_Memory == 1)
    {
    Allow_Run = 0;
    if (mySerial.available())
    Serial.write(mySerial.read());
    if (Serial.available())
    mySerial.write(Serial.read());
    mySerial.write(0xA7); //run subroutine 1
    mySerial.write((byte)0x00); //device id
      //Send Value of Binary 1 To Uno to Play WAV File S1
    digitalWrite (14, HIGH); // 1 2 4 8
    digitalWrite (15, LOW); // 1 2 4 8
    digitalWrite (16, LOW);  // 1 2 4 8
    digitalWrite (17, LOW);  // 1 2 4 8
    delay (500);
    digitalWrite (14, LOW);  // 1 2 4 8
    digitalWrite (15, LOW);  // 1 2 4 8
    digitalWrite (16, LOW);  // 1 2 4 8
    digitalWrite (17, LOW);  // 1 2 4 8
    delay (30000);  delay (30000);  delay (30000); delay (5000);
    Coin_Memory = 0;
    Allow_Run = 1;
    ShowNumber = ShowNumber + 1;


    }

 if (ShowNumber == 2 && Coin_Memory == 1)
    {
    Allow_Run = 0;
    if (mySerial.available())
    Serial.write(mySerial.read());
    if (Serial.available())
    mySerial.write(Serial.read());
    mySerial.write(0xA7); //run subroutine 2
    mySerial.write((byte)0x01); //device id
  //Send Value of Binary 2 To Uno to Play WAV File S2
    digitalWrite (14, LOW); // 1 2 4 8
    digitalWrite (15, HIGH); // 1 2 4 8
    digitalWrite (16, LOW); // 1 2 4 8
    digitalWrite (17, LOW); // 1 2 4 8
    delay (500);
    digitalWrite (14, LOW);  // 1 2 4 8
    digitalWrite (15, LOW);  // 1 2 4 8
```

```
       digitalWrite (16, LOW);  // 1 2 4 8
       digitalWrite (17, LOW);  // 1 2 4 8
       delay (30000);  delay (30000);  delay (30000); delay (5000);
       Coin_Memory = 0;
       Allow_Run = 1;
       ShowNumber = ShowNumber + 1;


       }


   if (ShowNumber == 3 && Coin_Memory == 1)
       {
       Allow_Run = 0;
       if (mySerial.available())
       Serial.write(mySerial.read());
       if (Serial.available())
       mySerial.write(Serial.read());
       mySerial.write(0xA7); //run subroutine 3
       mySerial.write((byte)0x02); //device id
     //Send Value of Binary 3 To Uno to Play WAV File S3
       digitalWrite (14, HIGH); // 1 2 4 8
       digitalWrite (15, HIGH); // 1 2 4 8
       digitalWrite (16, LOW);  // 1 2 4 8
       digitalWrite (17, LOW);  // 1 2 4 8
       delay (500);
       digitalWrite (14, LOW);  // 1 2 4 8
       digitalWrite (15, LOW);  // 1 2 4 8
       digitalWrite (16, LOW);  // 1 2 4 8
       digitalWrite (17, LOW);  // 1 2 4 8
       delay (30000);  delay (30000);  delay (30000); delay (5000);
       Coin_Memory = 0;
       Allow_Run = 1;
       ShowNumber = 1;
       }
  }
}
```

7.2 Arduino Uno R3 (Audio Player)

The Arduino Uno R3 audio player reads a 4 digit binary value from the Coin Acceptor Arduino Uno R3 and plays an audio file associated with the binary value.

```
#include <SPI.h>
#include <Adafruit_VS1053.h>
#include <SD.h>
#define BREAKOUT_RESET  9     // VS1053 reset pin (output)
```

```
#define BREAKOUT_CS    10    // VS1053 chip select pin (output)
#define BREAKOUT_DCS   8     // VS1053 Data/command select pin (output)
#define SHIELD_RESET  -1     // VS1053 reset pin (unused!)
#define SHIELD_CS     7      // VS1053 chip select pin (output)
#define SHIELD_DCS    6      // VS1053 Data/command select pin (output)
#define CARDCS 4             // Card chip select pin
#define DREQ 3               // VS1053 Data request, ideally an Interrupt pin

Adafruit_VS1053_FilePlayer musicPlayer =
Adafruit_VS1053_FilePlayer(SHIELD_RESET, SHIELD_CS, SHIELD_DCS,
DREQ, CARDCS);

//int val;
//Define weighted binary values used for input mapping
 int Binary1 = 0;
 int Binary2 = 0;
 int Binary4 = 0;
 int Binary8 = 0;




void setup()
{
//Define pins used for inputs sent from the other Arduino
 pinMode(14, INPUT_PULLUP);    // sets the analog pin A0 as input
 pinMode(15, INPUT_PULLUP);    // sets the analog pin A1 as input
 pinMode(16, INPUT_PULLUP);    // sets the analog pin A2 as input
 pinMode(17, INPUT_PULLUP);    // sets the analog pin A3 as input

 Serial.begin(9600);
 // initialise the music player
 if (! musicPlayer.begin())
 { // initialise the music player
   Serial.println(F("Couldn't find VS1053, do you have the right pins defined?"));
   while (1);
 }
 if (!SD.begin(CARDCS))
 {
 }

 // Set volume for left, right channels. Lower numbers == louder volume!
 //This is if you purchase unit with amplifier built in
 musicPlayer.setVolume(20,20);

 if (! musicPlayer.useInterrupt(VS1053_FILEPLAYER_PIN_INT))
```

```
      Serial.println(F("DREQ pin is not an interrupt pin"));
}

void loop()
{
 // Start playing a file, then we can do stuff while waiting for it to finish
 if (musicPlayer.stopped())
 {
 delay(1);
 }

 {

 Binary1 = digitalRead(14); // Binary 1
 Binary2 = digitalRead(15); // Binary 2
 Binary4 = digitalRead(16); // Binary 4
 Binary8 = digitalRead(17); // Binary 8

//Show 1 Detected - Binary 1
if (Binary1 == HIGH && Binary2 == LOW && Binary4 == LOW && Binary8 ==
LOW)// 1 2 4 8
{
Serial.println("Show 1");
  (musicPlayer.startPlayingFile("s1.mp3"));
  delay (1000);
}

//Show 2 Detected - Binary 2
if (Binary1 == LOW && Binary2 == HIGH && Binary4 == LOW && Binary8 ==
LOW)// 1 2 4 8
{
Serial.println("Show 2");
  (musicPlayer.startPlayingFile("s2.mp3"));
  delay (2000);
}

//Show 3 cents Detected - Binary 3
if (Binary1 == HIGH && Binary2 == HIGH && Binary4 == LOW && Binary8 ==
LOW)// 1 2 4 8
{
Serial.println("Show 3");
  (musicPlayer.startPlayingFile("s3.mp3"));
  delay (2000);
}
```

```
        }
      }
```

7.3 Pololu Maestro

The Pololu Maestro controls the RC relays for the Crystal Ball, Star Ceiling, PicoVolt trigger, and Fortune dispenser trigger. The Pololu Maestro also controls the Firgelli actuator.

The fortune teller is powered all of the time. To extend the life of the head servos, a RC relay is also controlled by the Maestro to only provide power to the relays when a show is active. Since a PicoTalk provides power to the servos and the PicoTalk is always powered on, a 12" extension for each servo was added and the power wire routed through the RC switch. The PicoTalk power is left on all of the time since it also controls the LED eyes and I wanted this to always be active.

Refer to the electrical drawings for hookup information.

```xml
<!--Pololu Maestro servo controller settings file,
http://www.pololu.com/catalog/product/1350-->
<UscSettings version="1">
  <NeverSuspend>false</NeverSuspend>
  <SerialMode>UART_FIXED_BAUD_RATE</SerialMode>
  <FixedBaudRate>9600</FixedBaudRate>
  <SerialTimeout>0</SerialTimeout>
  <EnableCrc>false</EnableCrc>
  <SerialDeviceNumber>12</SerialDeviceNumber>
  <SerialMiniSscOffset>0</SerialMiniSscOffset>
  <Channels MiniMaestroServoPeriod="80000" ServoMultiplier="1">
   <!--Period = 20 ms-->
   <!--Channel 0-->
   <Channel name="R1  Projector" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6003"
range="1905" />
   <!--Channel 1-->
   <Channel name="R2  Spare" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6000"
range="1905" />
   <!--Channel 2-->
   <Channel name="R3 Arm Sweep" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6000"
range="1905" />
   <!--Channel 3-->
   <Channel name="R4 Crystal Ball" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6000"
range="1905" />
   <!--Channel 4-->
```

```
<Channel name="R5 Dispenser" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6000"
range="1905" />
<!--Channel 5-->
<Channel name="R6 Head Servo" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="256" acceleration="255" neutral="6000"
range="1905" />
<!--Channel 6-->
<Channel name="" mode="Servo" min="3968" max="6976" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 7-->
<Channel name="" mode="Servo" min="3968" max="7040" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 8-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 9-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 10-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 11-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="6" acceleration="0" neutral="6000" range="1905" />
<!--Channel 12-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 13-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 14-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 15-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 16-->
<Channel name="" mode="Servo" min="3968" max="8000" homemode="Off"
home="3968" speed="0" acceleration="0" neutral="6000" range="1905" />
<!--Channel 17-->
<Channel name="Forearm" mode="Servo" min="3968" max="8000"
homemode="Off" home="3968" speed="0" acceleration="0" neutral="6000"
range="1905" />
</Channels>
```

```
 <Sequences />
 <Script ScriptDone="true">#
#John Guarnero
#November 2014
#Madam K Maestro Control For Servo Relays and Firgelli actuator
#Pololu RC Switch(6), Firgelli Actuator
#6000 center, 4000 min, 8000 max

4000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn Off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
8000 17 servo #Forearm (8000 is down)
500 delay

sub Subroutine_1

#Show 1 detected
4000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
8000 17 servo #Forearm (8000 is down)

500 delay
8000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
8000 5 servo #Head Servos

11000 delay
5500 17 servo #Forearm (5500 is up)
4000 delay

8000 5 servo #Head Servos
8000 0 servo #projector
4000 1 servo #Spare
8000 2 servo #Turn off PicoVolt Arm Sweep
8000 3 servo #Crystal Ball
```

```
500 delay
4000 2 servo #Turn off PicoVolt Arm Sweep
30000 delay
22500 delay
4000 3 servo #Crystal Ball
8000 17 servo #Forearm (8000 is down)
15000 delay
8000 4 servo #Dispense Fortune
1500 delay
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
4000 0 servo #projector
8000 17 servo #Forearm (8000 is down)


quit

sub Subroutine_2

#Show 2 detected
4000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
8000 17 servo #Forearm (8000 is down)

500 delay
8000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
8000 5 servo #Head Servos
8000 17 servo #Forearm (8000 is down)

23000 delay
5500 17 servo #Forearm (5500 is up)
4000 delay

8000 5 servo #Head Servos
8000 0 servo #projector
4000 1 servo #Spare
8000 2 servo #Turn on PicoVolt Arm Sweep
```

```
8000 3 servo #Crystal Ball
500 delay
4000 2 servo #Turn off PicoVolt Arm Sweep
30000 delay
22500 delay
4000 3 servo #Crystal Ball
8000 17 servo #Forearm (8000 is down)
10000 delay
8000 4 servo #Dispense Fortune
1500 delay
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
4000 0 servo #projector
8000 17 servo #Forearm (8000 is down)


quit

sub Subroutine_3

#Show 3 detected
4000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
8000 17 servo #Forearm (8000 is down)

500 delay
8000 0 servo #projector
4000 1 servo #Spare
4000 2 servo #Turn off PicoVolt Arm Sweep
4000 3 servo #Crystal Ball
4000 4 servo #Dispense Fortune
8000 5 servo #Head Servos

13000 delay
5500 17 servo #Forearm (5500 is up)
4000 delay

8000 5 servo #Head Servos
8000 0 servo #projector
4000 1 servo #Spare
8000 2 servo #Turn off PicoVolt Arm Sweep
```

```
8000 3 servo #Crystal Ball
500 delay
4000 2 servo #Turn Off PicoVolt Arm Sweep
30000 delay
22500 delay
4000 3 servo #Crystal Ball
8000 17 servo #Forearm (8000 is down)
15000 delay
8000 4 servo #Dispense Fortune
1500 delay
4000 4 servo #Dispense Fortune
4000 5 servo #Head Servos
4000 0 servo #projector
8000 17 servo #Forearm (8000 is down)

quit

</Script>
</UscSettings>
```

8   Electrical Parts Listing

The parts listed below were used to make the Madame K fortune teller machine.

Description	Purchased	Part #	Qty
Arduino Uno R3 Coin Acceptor	Adafruit.com	Product ID: 50	1
Arduino Uno R3 Audio Playback	Adafruit.com	Product ID: 50	1
Arduino Music Maker Shield	Adafruit.com	Product ID: 1790	1
Art Controller Relay Board Kit	Adafruit.com	Product ID: 1038	1
Lepai LP-2020A+ Audio Amplifier	Amazon.com		1
Maestro 18 Channel Servo Controller	Pololu.com	#1354	1
RC Switch with Relay	Pololu.com	#2804	6
Elbow Actuator & Board L16-P Linear Actuator 50 mm ,150:1,12 Volts	Firgelli.com	L16-50-150-12-P-LAC	1
Extension Cable -S	Firgelli.com	Cable-S	3
LAC Board Case	Firgelli.com	Case	1
PicoVolt	Frightprops.com	Item: 0975B	1
Motor	Frightprops.com	Item: 0955-MOT1-P	1
Ball Joint Linkage	Frightprops.com	Item: 0955-SALS	1
Ball Joint Linkage Connector	Frightprops.com	Item: 0955-SALS-BLCS	1
Talking Skull with Upgraded Blue LED Eyes	Frightprops.com	Item: 0911-2013	1
Talking Skull with Upgraded Blue LED Eyes	Frightprops.com	Item: 0911-2013	1
Card Dispenser	Vendapin.com	900527-CTD	1
Card Dispenser Power Supply (24VDC & 5 VDC)	Vendapin.com	300008-5/24	1
Sub Panel White Steel (12" x 10")	Automationdirect.com	HW-MP1210CS	1
Din Rail (35mm x 10mm x 1 meter)	Automationdirect.com	DN-R35SAL1-2	1
Wire Duct (1" x 1.5"x 6.5') Gray	Automationdirect.com	T1E-1015G-1	1
Power Supply 5VDC, 3A	Automationdirect.com	PSE05-115	1
Power Supply 12VDC, 1.25A	Automationdirect.com	PSE12-115	1

Description	Purchased	Part #	Qty
Din Rail Mounting Kit For Power Supply	Automationdirect.com	PSC-12-060	1
Skeleton	Costco.com	Pose-n-Stay	1
D-ST4 Terminal Block, End Cover	Mouser.com	3030420 (Phoenix)	2
ST-4 Jumper Bar, 10 Pole	Mouser.com	3030271 (Phoenix)	5
Terminal Block, ST-4	Mouser.com	3031364 (Phoenix)	5
STTB2.5, Endcover	Mouser.com	3030459 (Phoenix)	4
STTB2.5, Terminal Block	Mouser.com	3031270 (Phoenix)	14
ST-4 Ground Blocks	Mouser.com	3031380 (Phoenix)	3
End Blocks	Mouser.com	0800886 (Phoenix)	4
ZFK 6-DREHSILED 24 (6,3X32) Fuse Block	Mouser.com	3025587 (Phoenix)	2
ZFK 6-DREHSILED 250 (6,3X32) Fuse Block	Mouser.com	3025574 (Phoenix)	4