Junior Design II (Spring 2021)

Electronic Pet Door Developer Guide

Henok Techeste
Evan Cochran
Marcus Plumley

Table of Contents 1

System Overview

The electronic pet door was designed to lock, open, and monitor usage of the door autonomously. The system also allows for the user to program lock or unlock times for pet usage, and open the door manually with an emergency override option. Some other considerations for the design were material strength, means of data storage, and system clock accuracy. The system has one master arduino code file, and this is uploaded to the Arduino Uno to initialize the system. The user can adjust the lock and open times in a text file saved on the SD Module. Once initialized, the system has four input signals that are monitored: the two pressure plate signals and two buttons for the override signal. Output signals include save data to the SD module and signals to three relays that control door opening and locking. During general operation, an input from a pressure plate triggers a check against the lock and opens times, and if the signal is during a valid time, the door is unlocked, opened via motor, and the time is logged for usage monitoring. Otherwise, if a signal is received during a non-valid time, the door will remain closed unless the owner uses the emergency override.

Electrical Specifications

Parameter	Min	Typical	Max	Units
Supply Voltage	12V	12V	12V	Volts
Supply Current	50 mA	70 mA	3.5 A	Amps
Operating Temp.	-40	25	85	Celius

Table 1: Electrical Specifications for the system

User Guide

Setup:

1. Mark and Cut a 6x9 inch rectangle in a door or wall

2. Mount the pre-cut/assembled pieces of wood to the wall or door:
    a. One will be the pet door housing, and one way to align it to the door for mounting is to level and align it to one of the corners and screw through one of the pre-drilled holes. Once this screw is in, attach a second lower screw in such a way that the housing is level with the opening. Attach the remaining screws. Once this is done, the door sent with the kit will slide right in!
    b. The second will be the exterior of the wooden housing; this can be mounted to the left or right of the door, but when mounting, keep wiring in mind for connecting your devices. Once the location is determined, mount and attach the prebuilt door with the screws and hinges.Then mount the manual sliding latch for the door (there is a predrilled hole). There should be about 3 areas for wires to run out the door flap(the user will still be able to open the wiring enclosure).

3. Mount the Arduino Uno, RTC, SD adapter, power supply PCB, connector PCB, and the relays in the housing. This can be done by screwing through the predrilled holes in the PCBs.

4. To mount the motor, go to the point centered and about 10 inches above the top of the door housing and drill a 1-¼ hole. The motor can be screwed to the face of the wall or door. Finish this step by attaching string or cable to the door and motor arm to enable the motor to wind-up/down the door.

5. Mount the locking mechanism above the door housing. Wire the pressure plates and the emergency switches to the connector board, and wire the motor and locking mechanism to the relays.

6. Plug in the DC jacks to both the Arduino Uno and Power supply.(Drilling holes in the electrical housing might be required)

Operation:

1. Initialize the system by uploading the compiled code to the arduino uno.

2. Operating windows are adjusted in the Times.txt file on the SD card, which can be accessed by a computer.

3. Reload the system by using the reset button on the arduino so the new times are updated.

4. Once this is done, your pet activates the pressure plate sensor to open the door. This will unlock, open the door, and log the time. If during a Lock-time, the door will remain closed.

5. If the owner needs to let the pet out, there is a red button that can be pressed to open the door indefinitely. To end this state, press the green button which will lower the door and return it to original operation.

6. The logged times can be accessed from the SD card with a computer.

Electrical Schematic

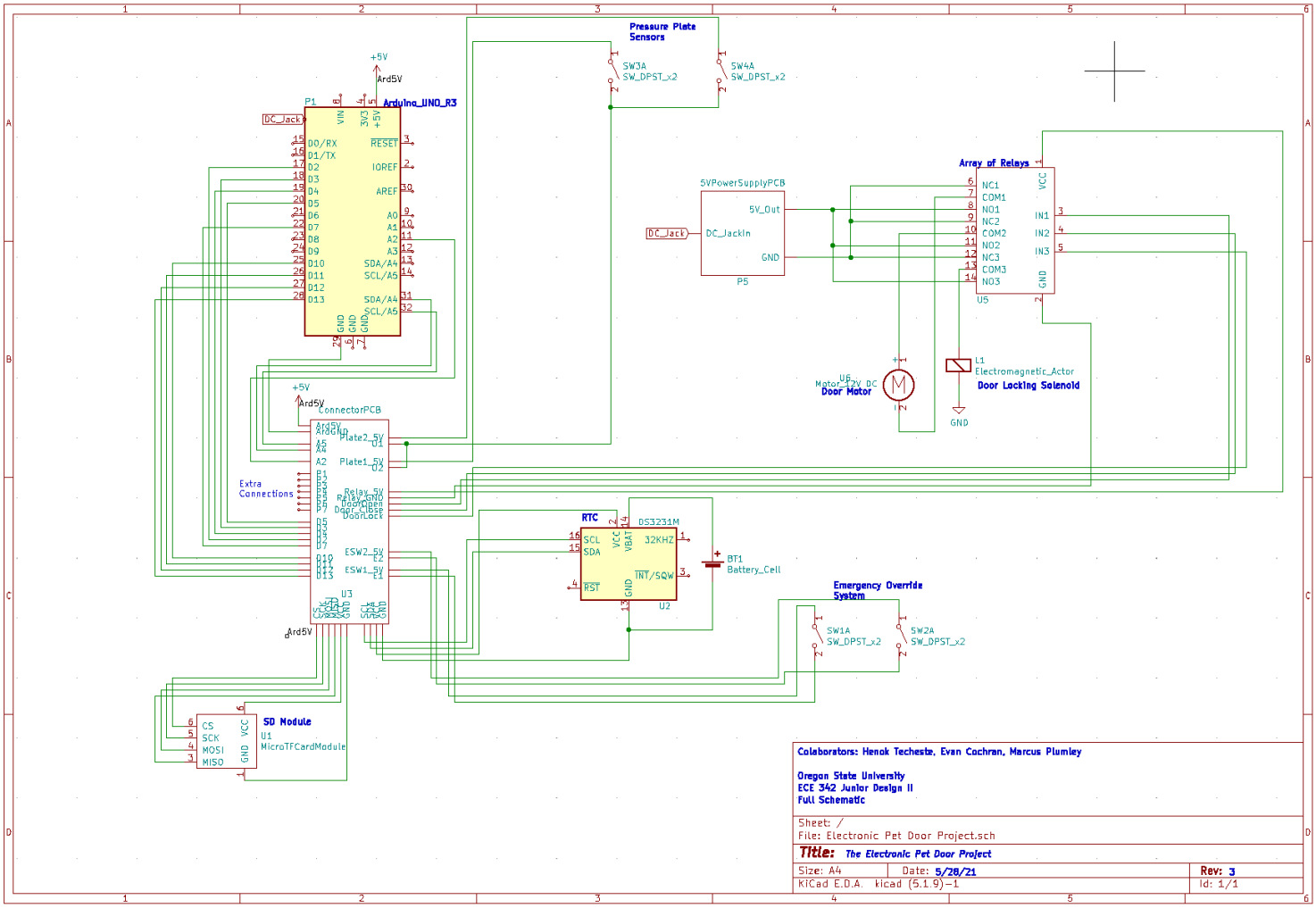

Figure 1: Full Design Schematic of Electronic Pet Door

The image above shows the complete system schematic with an Arduino Uno R3. The system includes the main programmer, two PCBs, three Relays, two pressure plates, emergency signal input, the SD Card, and RTC Module. The main PCB connector is utilized to connect the main parts of the system to the UNO. The voltage pins of the Arduino are connected to the main PCB and diverge to all the components' voltage inputs. The SD card Module gets power from the UNO and has the CS pins set to digital 10. The RTC module gets power through the UNO and has the pins set to digital pins as well. The emergency override buttons are connected to the main PCB and send the digital signal to the UNO. Two SPST relays are utilized to control the motor's movement and receive additional power from the power supply PCB. The power supply PCB is designed to receive 12VDC and regulate it to 5V to power the motor. The third

SPST relay is used to control the locking mechanism solenoid and receives power from the 5V power supply as well. A 12VDC motor is implemented into a pulley system to open/close the door. The pressure plates control the data usage of the pet and are powered through the Arduino UNO voltage and use two 10K Ohms pull-up resistors. The system is powered by 12VDC power through the wall.

3D Models & Mechanical Schematics

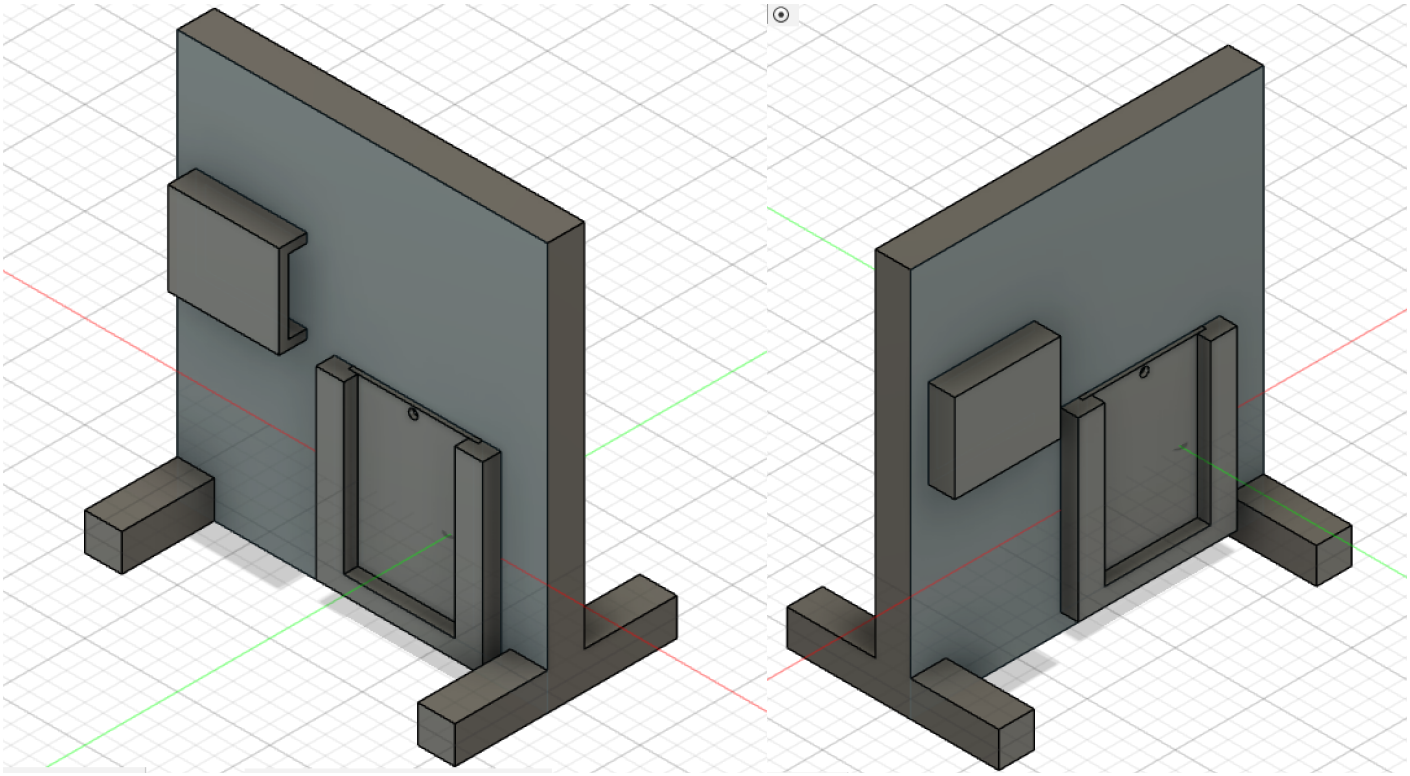

Figure 2: Main enclosure with door with electrical housing covered

The Main enclosure from Figure 2 consists of the prototype framing and the system enclosure. As described in the setup above, the door housing and electrical housing (the protruding enclosures) will be sent pre-drilled and preassembled. The door opening is cut out of the door in the setup, and the plexiglass door will also be sent preassembled.

In the physical build, the electrical housing is more enclosed and uses hinges as described in the setup section to be openable. This also allows access to the Arduino Uno for programming.

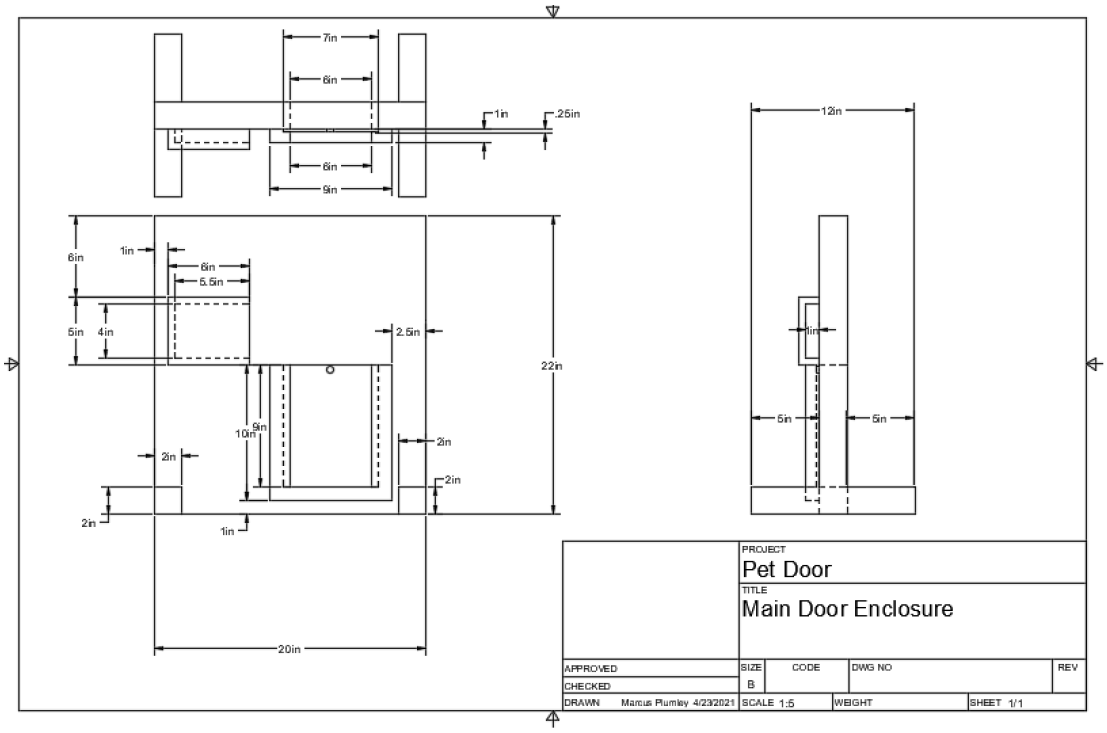

Figure 3: Main enclosure mechanical schematic

The dimensions of the prototype frame and the enclosures are shown above in Figure 3. The over footprint of the prototype frame was 20inchx22inchx12inch. The overall dimensions of the door enclosure are 10inchx9inchx1inch, and the overall electrical enclosure dimensions are 5inchx6inchx1.5inch.

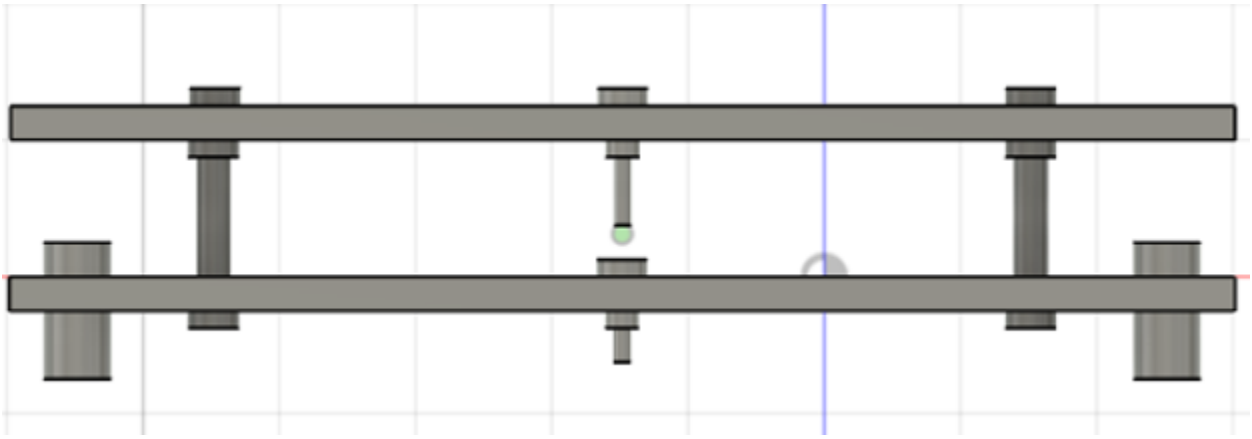

Figure 4: Pressure plate sensor model

This is the model of the pressure plate sensor which is the usage sensor. The overall design is like a spring-loaded button where the bolts connecting the two plates have springs holding the plates apart. The bolts in the middle create a short-circuit when the plate is pressed down, and the signal is sensed by the Arduino through the connector PCB. That PCB also includes pull-down resistors for each plate.

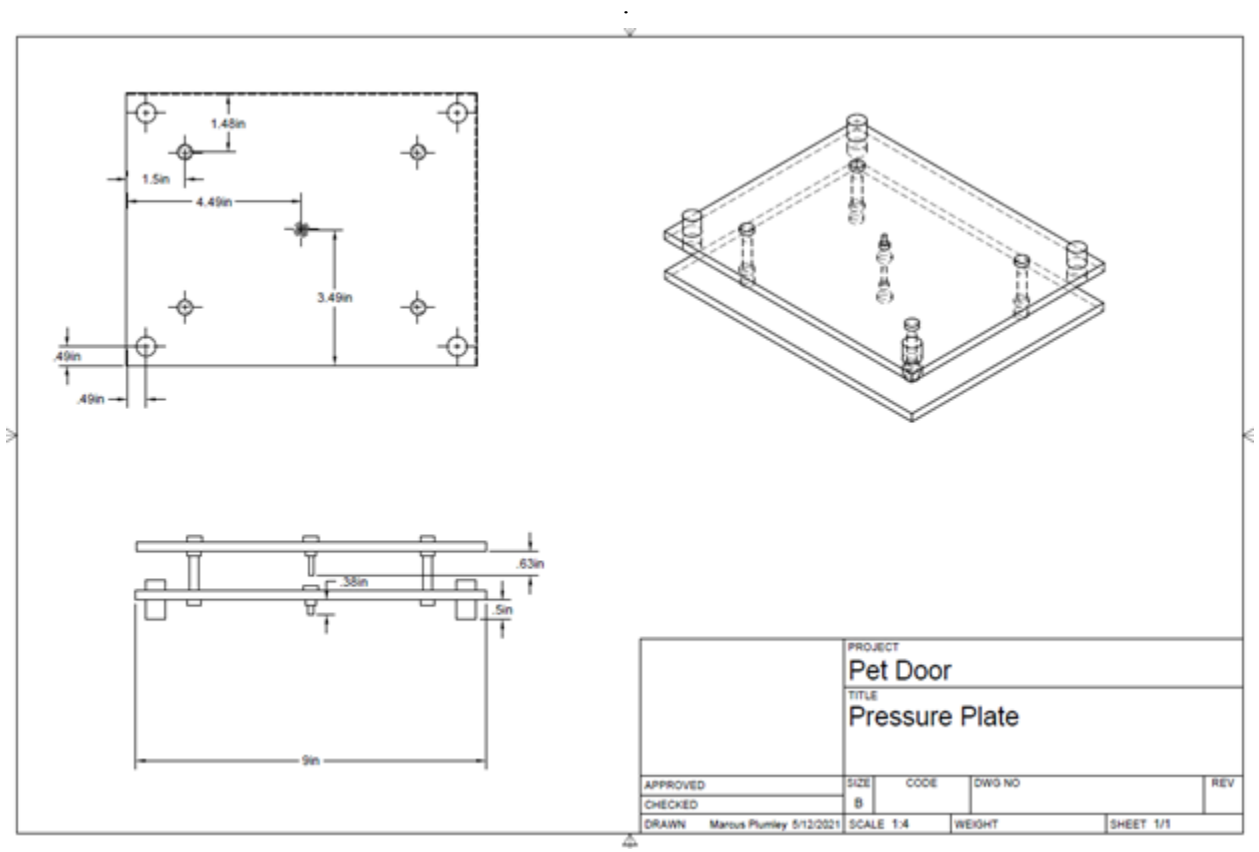

Figure 5: Pressure plate sensor mechanical schematic.

The overall dimensions of the pressure plate sensors built were 9inchx7inchx2inch. The top and side views of the plate show the dimensions well, but the bottom view shows the leg construction on the base which is important to allow the plates to be pressed.

Block Diagrams and Interface Definitions

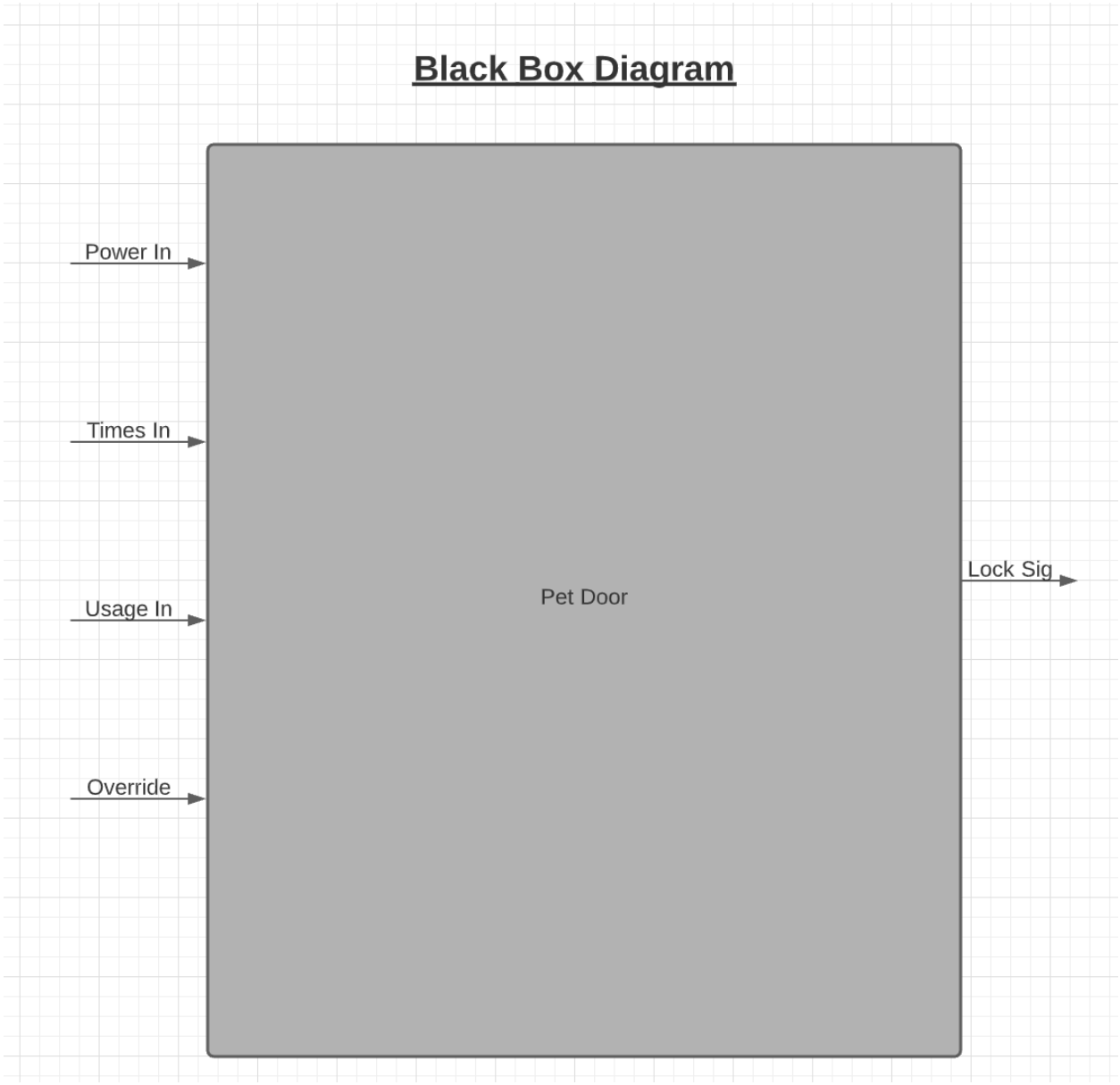

Figure 6: Black Box Diagram of Electronic Pet Door.

The electrical system takes in four inputs--power, time data, pet usage, and an override signal--and generates one output--a signal that locks or unlocks the door. For power, a 12V wall adapter is used. Time data is read off of a SD card, and can be configured on the owner's computer. The pet usage signal is triggered by the pressure plates, and tells the arduino that a pet wants to open the door. The override signal is generated when the owner uses the override button. The pet usage will only open the door during allowed times, but the override signal will always open the door. The lock signal output tells the door to unlock and the motor to open the door.

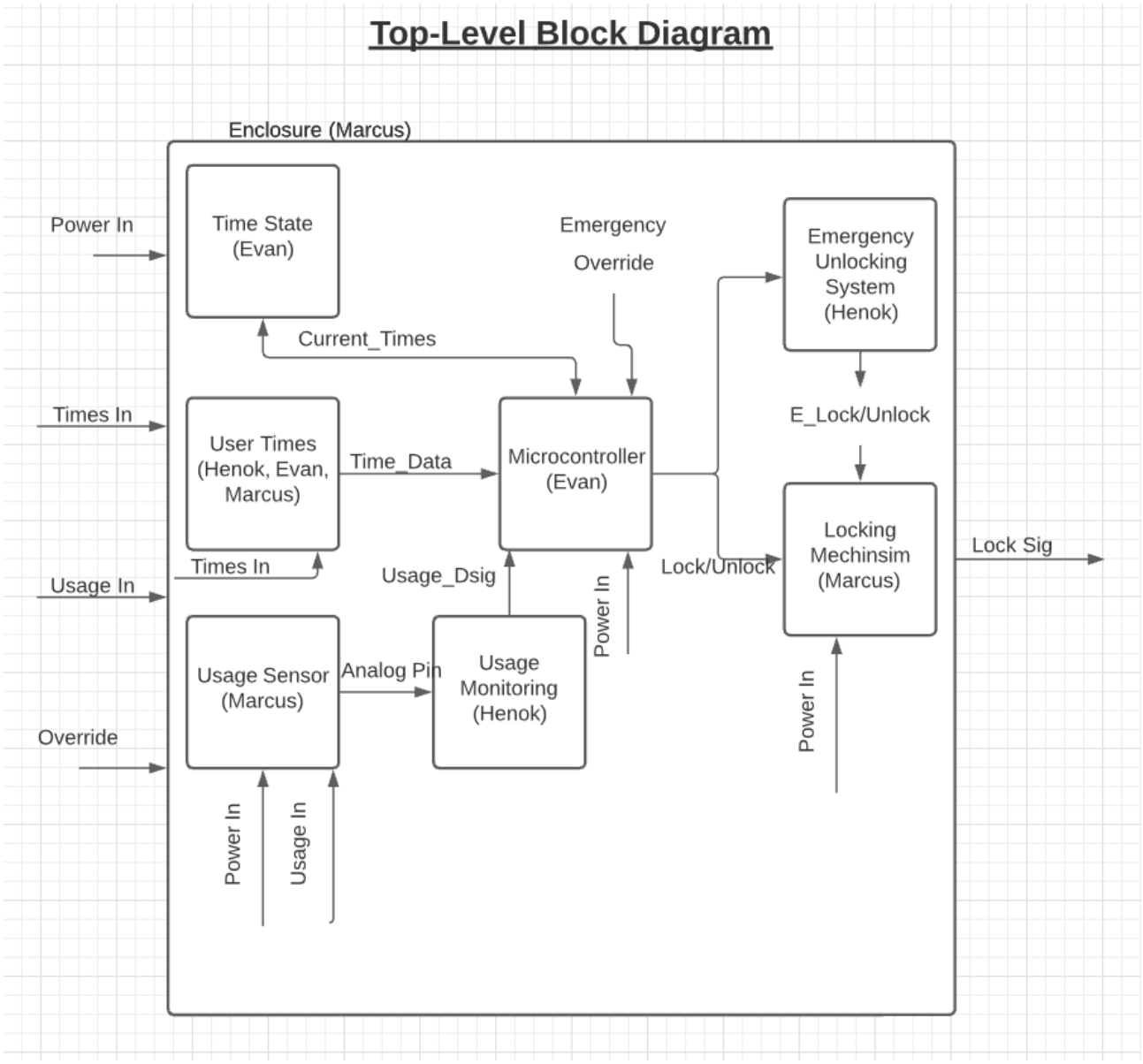

Figure 7: Top-level block diagram of Electronic Pet Door.

The top level block diagram shows how the arduino interacts with its peripherals. The time state block is the real time clock module, and accurately maintains the current time even when the system loses power. User times are generated on system reset, and are read off the SD card. The usage sensor is a pressure plate, functioning as a large mechanical button. When it is used, the current time is logged to the SD card. If the pet wants to use the door and is allowed to do so, the locking mechanism is retracted and the door is raised, remains open for 4 seconds, and then closes again. When the override signal is detected, the emergency unlocking system is used which unlocks and opens the door no matter what mode it is set to. The door remains open until it receives the signal again.

Interface Definitions

Table 2: Interface Definitions

Interface	Type	Properties
Power In	DC	DC 12V as a power source
		3.5A max
Times In	Data signal	Lock and unlock times are stored on the SD card, and are set in the same method as the user checking pet usage
Usage In	Analog	Signal from the mechanical pressure plate for usage
		DC 0 - 5 volts; we expect it to be the 0 volts in its natural state, and 5V when a pet wants to use the door. There are two plates that both use the same input pin
Override	Analog	Analog Signal, high voltage represents emergency incoming or emergency over, while low means button not pressed
		Voltage high of 5V, and 0V for low.
Lock Sig	DC	DC 5 volt Signal
		5 volts to unlock then 0 volts locks; this will be done by having a 5-volt relay activated when the lock needs to actuate
Usage_Dsig	Internal Digital	Arduino: Receives a 5V signal on analog pin 2. Opens the door if the defined times allow when this signal is received
		System should collect data for at least 7 days, and time recorded should only be +/- 1 min off; this will take times from RTC for times and SD card module for storage; will use the Arduino libraries for both
Current_ Times	Serial	The RTC module can be interfaced with time set, and time get functions. Time set sets the current time. Time get functions include functions to get the date-- .day(), .month(), and .year()-- as well as the current time-- .hour(), .minute(), and .second(). I2C is the protocol used to interface with the module

Interface	**Type**	**Properties**
Time_Data	Code	Timing window data is stored on the SD card. On reset, the system reads in the data and saves it in an array for convenient use
Analog Pin	Digital signal	Logic high 5v when signal present on analog 2, low otherwise
		When the signal is 0 volts, the door will be known to be closed; when the signal is 5 volt the door will need to be opened and the Usage Monitoring needs to note the time if it is allowed to
Emergency	Analog	Analog Signal: Emergency mode is activated by a separate external 5V input signal that overrides other inputs and immediately opens the door no matter the user set times
		A 5V signal on the emergency over pin closes the door and returns the system to normal operation
		0V on either is considered no input, and pressing either button multiple times in a row has no extra effect
Lock/ Unlock	DC	DC: Based on the user time set, lock and unlock will be actuated by the Arduino output pins and relays.
		DC 5v will unlock the solenoid while DC 0v will leave it locked. 5V DPDT will unlock upon user set times.
E_Lock/Unlock	DC	When the override digital signal is inputted into the system, the lock/unlock algorithm will break and immediately run the unlock sequence for the door

PCB Layout and Design

5V PCB Power Supply
Size: 39.04mm X 26.65mm

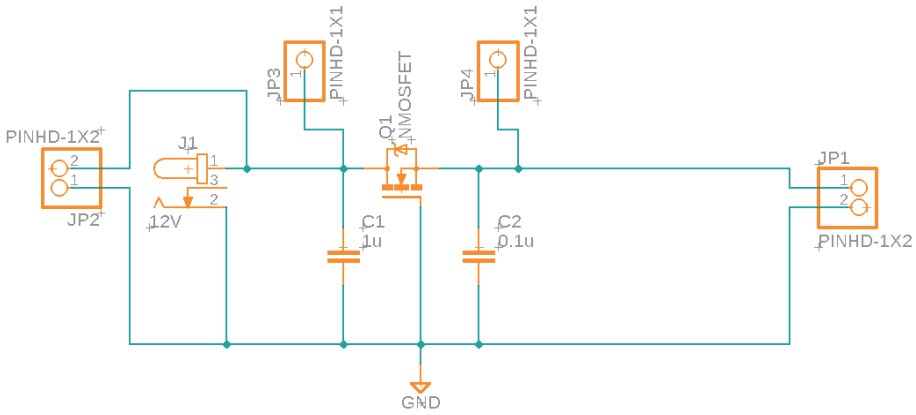

Figure 8: Electrical Schematic of Power Supply PCB

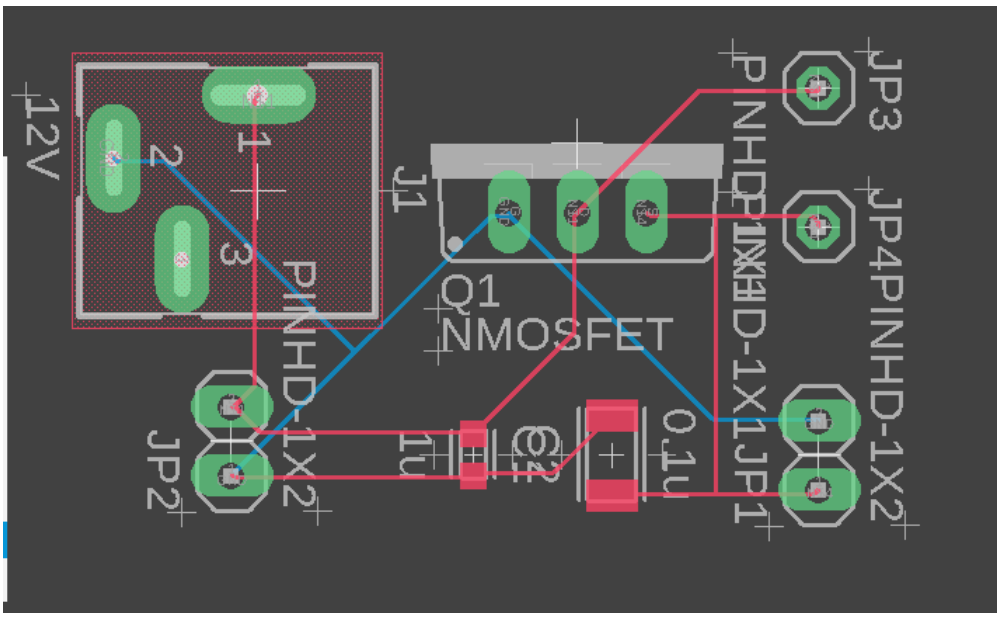

Figure 9: PCB Layout of Power Supply PCB

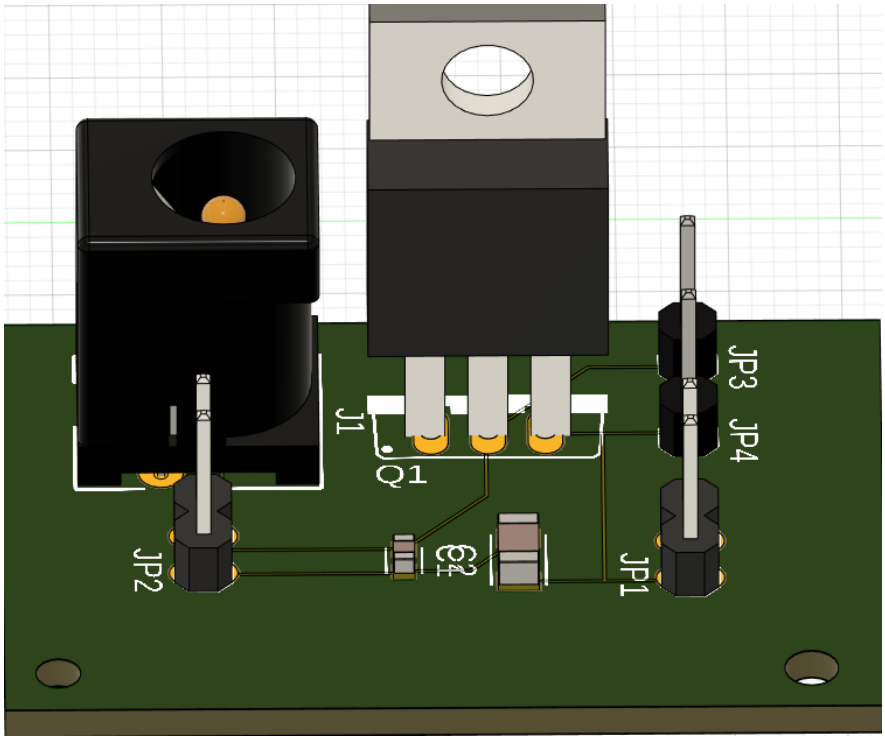

Figure 10: 3D model of Power Supply PCB

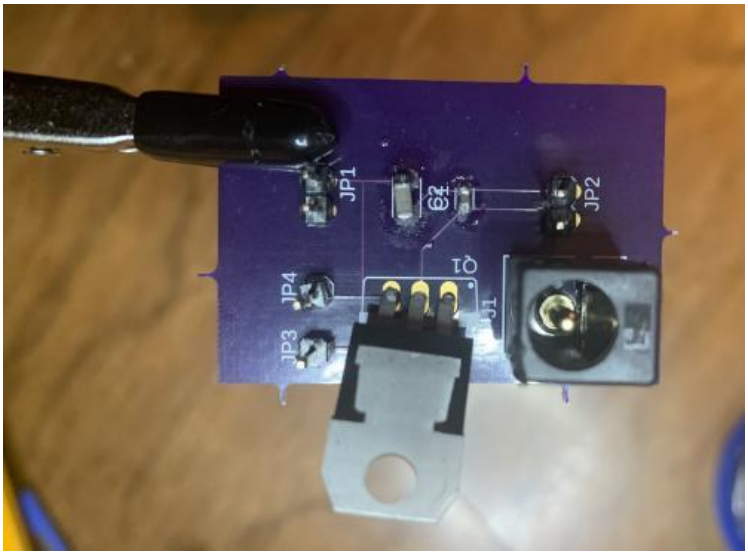

Figure 11: Physical model of Power Supply PCB

Connector PCB
Size: 78.72mm x 45.39mm

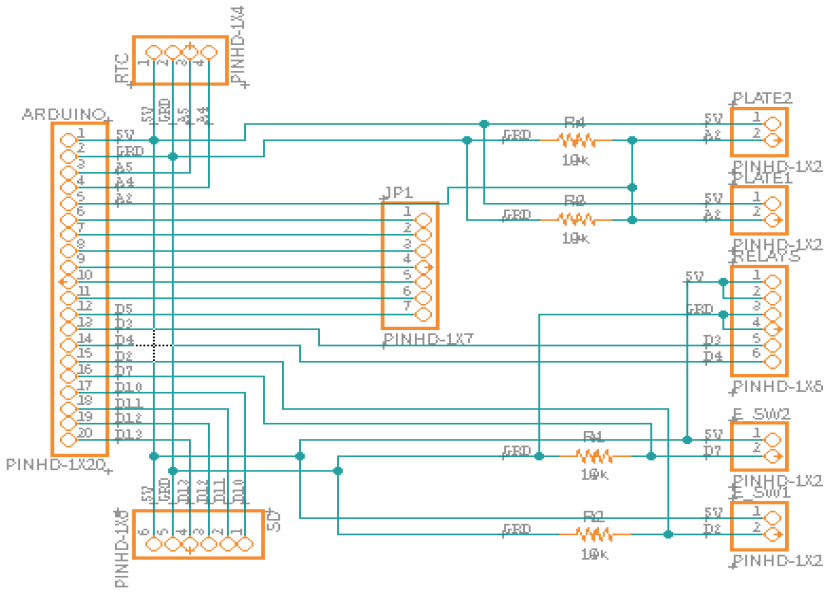

Figure 12: Electrical Schematic of Connector PCB

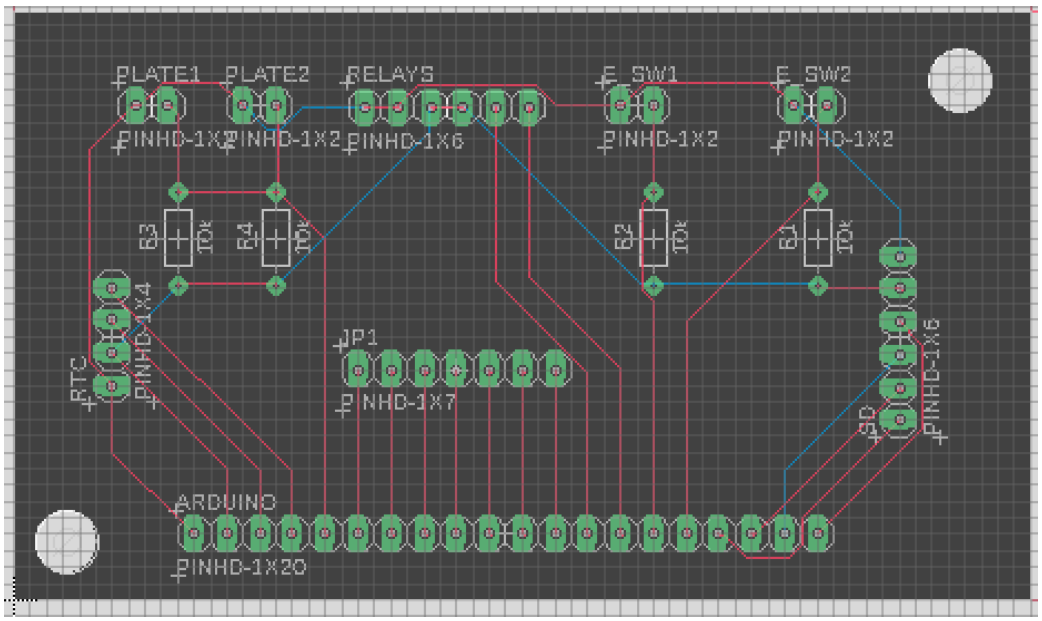

Figure 13: PCB Layout of Connector PCB

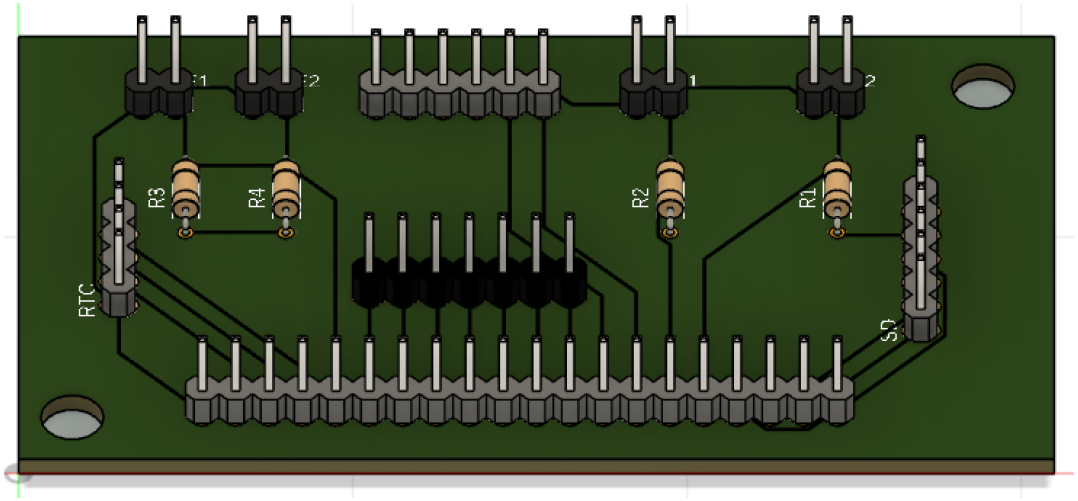

Figure 14: 3D model of Connector PCB

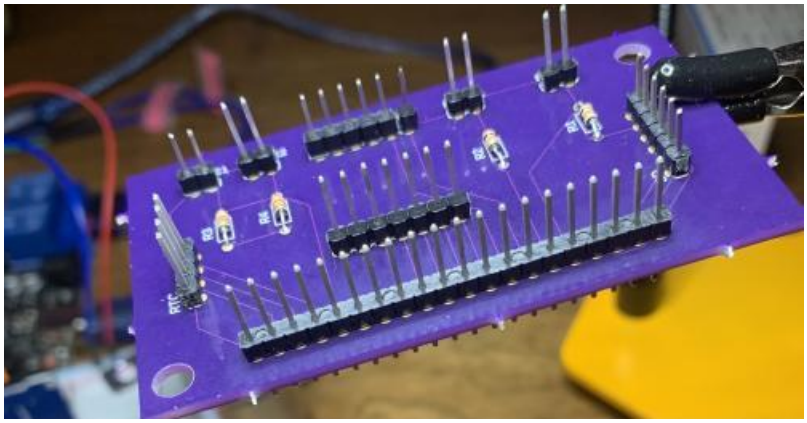

Figure 15: Physical model of Connector PCB

Parts Information

Table 3: Parts List with schematic designators when applicable.

Device	Type	Description	Quantity Used	Notes on Schem
Arduino Uno	R3	Main Microcontroller - Atmega328P	1	P1
RTC Module	DS3231	Real-Time Clock Module	1	U2
Micro TF Card Module	DAT-01-003	SPI Micro Storage Card Adapter Module	1	U1
Push-button switch	1825910-6	PCB push-button switch	2	SW1, SW2
Power Supply	DC 5V PCB power supply	5V Power Supply	1	P5
Resistor	805	10K Ohms resistors	2	R1, R2, R3, R4
SPST Relay	619-27115	5V SPST relays	3	U5
Motor Gear	No2998	12V DC 25 RPM gear	1	U6
Micro SD Card	SDCS2	32GB	1	SD1
Push- Pull Solenoid	2776	5v Solenoid	1	L1
Pressure plate	See 3D models	5v pressure plate switch	2	SW3. SW4
Connector PCB	Pin connector board	Integration and wiring	1	U3
AC Adapter	Netgear 007LF	12V DC adapter	1	

Hardware	**Description**	**Quantity**
9 inchx7inchx1/4inch wooden board	Thin Board to make base of pressure plates	4
Springs	Springs to keep plates separated	8
Copper Wire~2ft-long	Wiring for pressure plates	4
Insulated Ring Terminal	Connector for wire to bolt connection	4
3/16inch-Bolt~5/8-1 inch long	Short-Circuit bolts that create the switch connection	4
3/16inch-Nut	Nuts for the above bolts	4
1/4inch-Bolt~2 inch long	Guide bolts that keep plates aligned and guide the transition from pressed/not-pressed	8
1/4inch-Nut	Nuts for the above bolts	16
3/8inch-Bolt~1 inch long	Legs for bottom wooden plate for pressure plate model	8
3/8inch-Nut	Nuts for the above bolts	16
9inchx7inchx1/8inch Plexiglass	This will make up the door for the system	1
10inchx9inchx1inch Pet Door housing	This is the housing that the door slides up/down in	1
5inchx6inchx1inch Electrical Housing	This will be mounted to wall/door to contain components	1
Hinges	For electrical housing door that will contain PCB, microcontroller,etc	2
5inchx6.25inchx1/4inch Electrical housing Door	To be mounted with hinges to cover electrical housing	1

Electrical Housing Door Latch	To be mounted to latch the housing door	1

PCB BOMs

Part	Value	Device	Package	Description	CATEGORY	DESCRIPTION	DRAIN_CURRENT	MANUFACTURER	MPN
C1	1u	C_CHIP-0603(1608-METRIC)	CAPC1608X85	Capacitor - Generic	Capacitor				
C2	0.1u	C_CHIP-1206(3216-METRIC)	CAPC3216X135	Capacitor - Generic	Capacitor				
J1	12V	694106402002	694106402002	DC POWER JACK - VERTICAL - THT - 2.0 & 2.5Ø - WR-DC	Connector	DC POWER JACK - VERTICAL - THT - 2.0 & 2.5Ø - WR-DC		Würth Elektronik	694106402002
JP1	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR				
JP2	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR				
JP3	PINHD-1X1	PINHD-1X1	1X01	PIN HEADER	CONNECTOR				
JP4	PINHD-1X1	PINHD-1X1	1X01	PIN HEADER	CONNECTOR				
Q1	NMOSFET	NMOSFET_TO220	TO220BV	N-Channel MOSFET - Generic	Transistor				

Table 4: Power Supply PCB BOM.

Partlist exported from C:/Users/jntpl/AppData/Local/Temp/Neutron/ElectronFileOutput/13984/sch-3

Part	Value	Device	Package	Description	CATEGORY
ARDUINO	PINHD-1X20	PINHD-1X20	1X20	PIN HEADER	CONNECTOR
E_SW1	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR
E_SW2	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR
JP1	PINHD-1X7	PINHD-1X7	1X07	PIN HEADER	CONNECTOR
PLATE1	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR
PLATE2	PINHD-1X2	PINHD-1X2	1X02	PIN HEADER	CONNECTOR
R1	10k	R-US_AXIAL-7.2MM-PITCH	RESAD724W46L381D178B	Resistor Fixed - ANSI	Resistor
R2	10k	R-US_AXIAL-7.2MM-PITCH	RESAD724W46L381D178B	Resistor Fixed - ANSI	Resistor
R3	10k	R-US_AXIAL-7.2MM-PITCH	RESAD724W46L381D178B	Resistor Fixed - ANSI	Resistor
R4	10k	R-US_AXIAL-7.2MM-PITCH	RESAD724W46L381D178B	Resistor Fixed - ANSI	Resistor
RELAYS	PINHD-1X6	PINHD-1X6	1X06	PIN HEADER	CONNECTOR
RTC	PINHD-1X4	PINHD-1X4	1X04	PIN HEADER	CONNECTOR
SD	PINHD-1X6	PINHD-1X6	1X06	PIN HEADER	CONNECTOR

Table 5: Connector PCB BOM