

The Tesla Coils' Mystery

Nume: Șapcaliu Maria-Viorica

Grupa: FILS 1221EA

Introduction

My proposed concept entails a variety of applications based on the Tesla Coil phenomenon that can be used in everyday life. At the same time, I'd like to show how various systems can communicate with one another using the aforementioned phenomenon as well as other hardware and software IT infrastructures such as (arduino programming, serial communication, audio signal communication, switching, amplification and resonant circuits, data storage, nmos and bipolar npn technology).

The purpose is to apply the knowledge gathered in the laboratory. To demonstrate the phenomenon, the Tesla coil circuit will generate an electric arc with a high voltage and a high frequency of the electromagnetic field that displays the phenomenon's bolt of artificial lightning. This electric discharge will also be modified to produce musical tones by modulating its spark output via an arduino interface. Optionally, I will redo some experiments to demonstrate alternative methods of highlighting the coil's high-frequency electromagnetic field such as (lighting an incandescent light bulb without any direct electrical connection), also using the arduino interface.

The basic phenomena is the Tesla coil effect, and it uses this phenomenon to replicate sound frequencies. Tesla Coil is an electrical Resonant Transformer Circuit that generates exceptional levels of voltage and sparks, as well as the capacity to destroy or switch on electronics from a distance. To resonate, tune, and adjust the voltage, the Tesla coil uses capacitors, spark gaps, and multiple coils.



The project helps me to put the knowledge I gained in theoretical classes into practice. For people interested in this topic, the distribution of theoretical information and practical experience could serve as a model. Therefore, they can get practical skills, critical thinking abilities, and multidisciplinary knowledge from designing with microprocessors that will assist preparing them for professions in technology or other technological fields in the future.

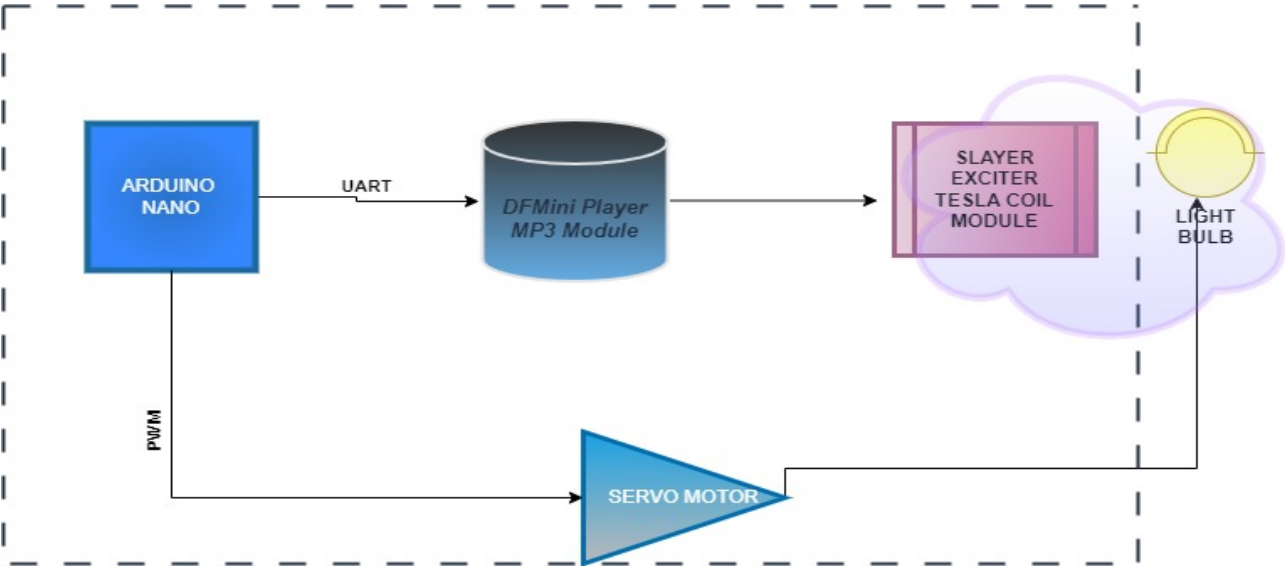
General Description

The following block diagram corresponds to the project:

First case:



Second case:



The project will be put together as follows:

The data storage module communicates with the Arduino via both the Lab 5 (SPI) and Lab 1 (UART) interfaces in this setup. Internal connection between the module and the microcontroller is enabled by SPI, whereas bidirectional communication between the Arduino and the module is enabled through UART. This enables efficient file storing and retrieval of musical notes.

For the second case, the microSD card in the DFMini Player Mp3 module is used to store and play mp3 music. Through UART, the Arduino interfaces with the DFMini MP3 Player module, allowing for control and data exchange. Furthermore, the module uses SPI internally to communicate with the microSD card. The DFRobotDFPlayerMini library, which provides a higher-level interface for smooth interaction with the module and microSD card, is used for this internal SPI connection.

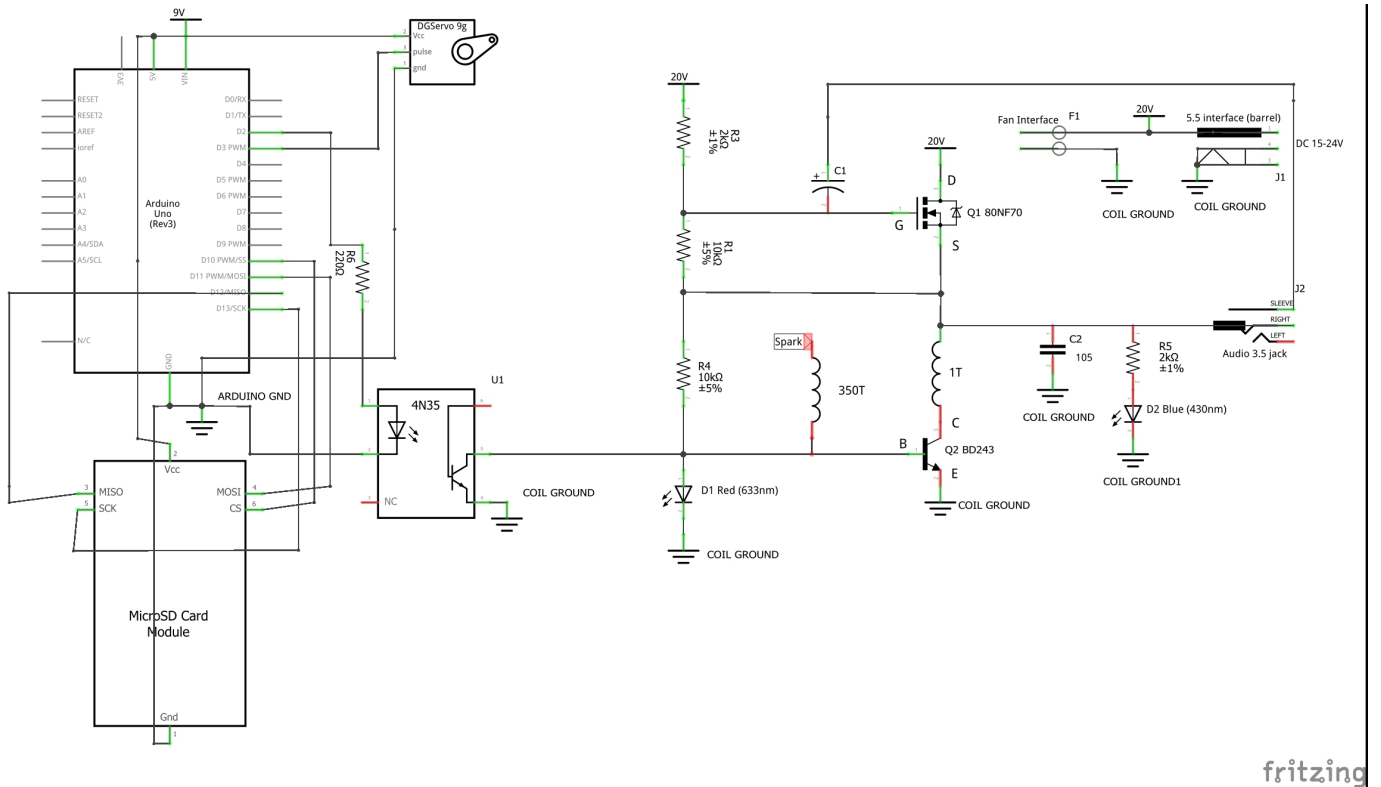
The Tesla coil's excitation module (circuit) will interpret the notes as signals of various frequency. The secondary coil will generate them in the auditory spectrum using a very high voltage electric arc. Optionally, based on the servo motor and a light bulb, I propose implementing PWM (L3) to emphasize the existence of the secondary coil's high frequency and high voltage electromagnetic field.

Hardware Design

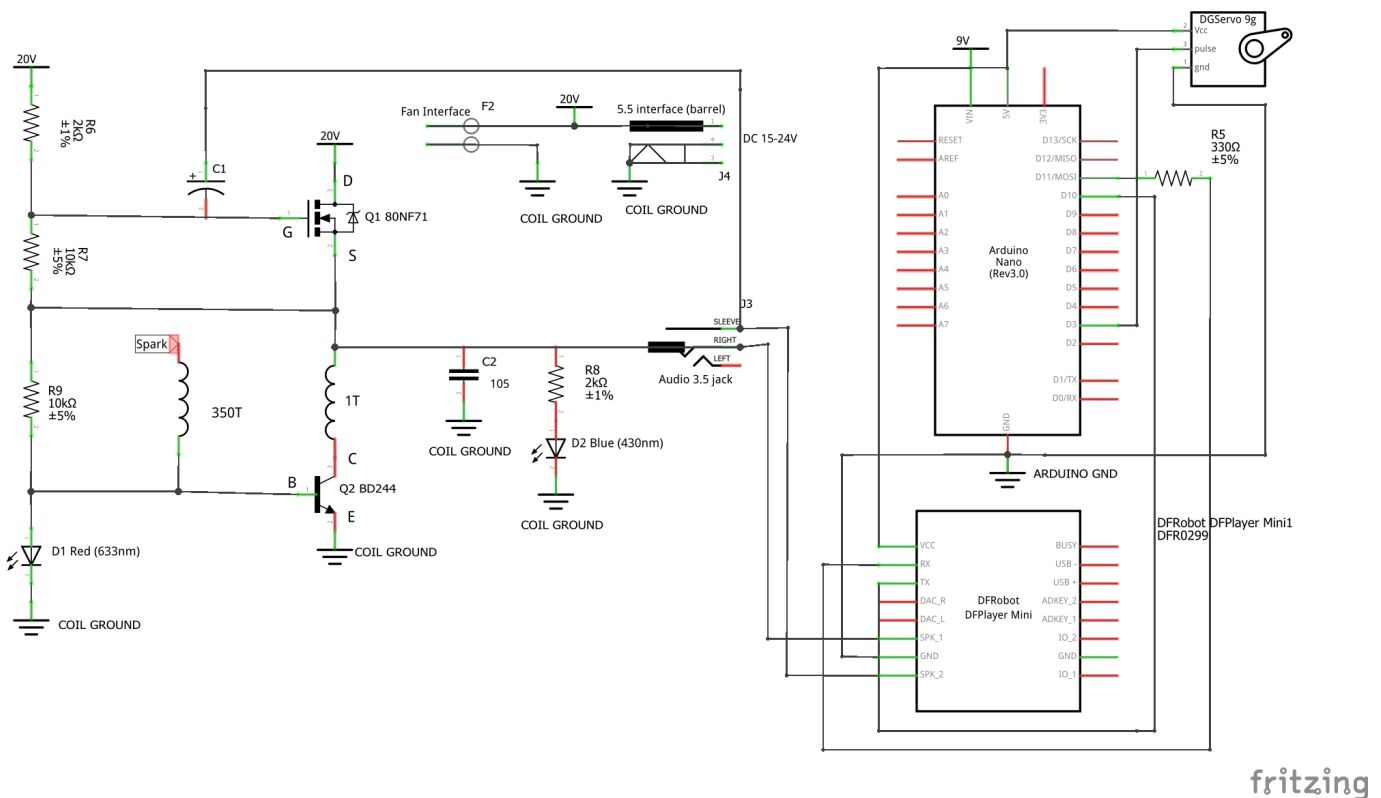
Component	Quantity
Arduino Uno R3	x1
Arduino Nano	x1
DFPlayer - A Mini MP3 Player	x1
Fan	x1
BD243C High Power NPN 100V	x1

80NF70 N-channel 68V Power MOSFET	x1
Transistor Heat Sinks	x2
10μF 50V electrolytic capacitor	x1
1μF 105J 100V polyester capacitor	x1
4N35 Optocoupler	x1
2kΩ Resistors	x2
10kΩ Resistors	x2
220Ω Resistor	x1
330Ω Resistor	x1
LED red	x1
LED blue	x1
pre-wound coil	x1
stereo audio socket	x1
power jack	x1
3.5mm stereo audio cable	x1
Dupont Jumper Wires	x30
wire	15cm
strip double-sided tape	5cm
screws	x6
brass standoffs	x4
Power Supply 9V for Arduino	x1
Power Supply 15-24V for Tesla Coil	x1
Universal PCB board	x1
Servo motor	x1
Micro- SD Card Memory Module	x1
Soldering iron	x1
Led/Bulb	x1
USB Cable for Arduino Uno	x1
USB Cable for Arduino Nano	x1
passive buzzer (optional)	x1
Micro SDHC card 32GB	x1

Digital Input Signal



Analog Input Signal



This project's circuit is a modified version of the "*slayer exciter*", a sort of Tesla transformer notable for its self-governing oscillation. The slayer exciter circuit, unlike classic Tesla coils, uses feedback from the secondary coil to enable self-tuning and self-resonating operation, avoiding the need for complex circuitry.

Some creative additions have been made to this version of the circuit. The current to the coil is modulated by an audio input via the 80NF70 NFET (Q2), while one LED (D2) is modulated by the audio input and another LED (D1) is modulated by the coil's oscillations. The audio-modulated coil can vibrate, and the plastic tube that acts as the coil's core resonates, effectively functioning as a speaker.

The “slayer exciter” self-governing oscillation is crucial to the circuit's operation.

Basic explanation of how it works with a single transistor:

1. The circuit is initially powered on, and the transistor (Q1) is in the “on” position. This means that current can pass through the Tesla coil's primary coil.
2. A magnetic field forms around the primary coil when current travels through it. This magnetic field causes a voltage to be generated in the secondary coil, where primary coil is wrapped around the secondary coil.
3. However, an unusual event occurs when the current in the secondary coil exceeds a particular threshold, often the breakdown current of an LED linked in series. The transistor's base is linked to ground, thus pulling it “down” and causing the transistor to reach a cutoff state.
4. When the transistor reaches cutoff mode, current is no longer conducted through the primary coil. This causes the magnetic field generated by the primary coil to collapse.
5. A high-voltage flyback voltage is induced in the primary coil as the magnetic field decreases. This flyback voltage discharges into the surrounding air, causing an electric arc or plasma to form.
6. The interruption of current flow in the primary coil has an effect on the secondary coil as well. A voltage spike in the secondary coil is caused by the sudden shift in magnetic field, which results in a high-voltage output.
7. The process is repeated indefinitely. When the flyback voltage dissipates and the current in the secondary coil falls below the breakdown level of the LED connected in series, the voltage falls below the breakdown level of the LED. This permits the transistor to return to saturation mode, turning it “on” and commencing the magnetic field accumulation.

The slayer exciter circuit can be controlled by modulating the circuit on and off with an interrupter, such as an Arduino combined with an optocoupler. This modulation can be done at the frequency of a certain musical note, and the ensuing electric arcs created by the Tesla coil can produce audible sounds at the modulated frequency.

Software Design

To interface the Musical Tesla Coil Slayer exciter circuit, which can accept analog or digital input signals, I conducted experiments to determine how the circuit can receive the signal and implemented the code accordingly. When an analog input is received, the circuit plays melodies, while digital input triggers the playback of musical notes.

I developed the programs using the Arduino IDE and included additional libraries and 3rd-party sources

For Analog input:

- SoftwareSerial.h
- DFRobotDFPlayerMini.h
- Servo.h

In the program, I used functions from these libraries to control the DFMini Player and servo motor. The DFMini Player receives commands via serial communication, with pins 10 and 11 serving as RX and TX. Melodies in integer-numbered files (e.g., 1.mp3, 2.mp3, etc.) should be stored on the memory card before specifying the commands.

Here is a summary of the program flow:

1. void setup():

- Initialized communication with the module and microSD card, as well as the Arduino serial communication.
- Set initial volume control, range control, equalization commands, and other features.
- Attached the servo object to pin 3.

2. void loop():

- Called the danceServo() function.
- Transmitted commands via serial communication to manage the DFPlayerMini.

3. void menu_opcoes():

- An interactive menu to select the command to be sent via serial communication.

4. void danceServo():

- This function accepts a pattern as input and controls the servo motor. Each pattern defines a sequence of servo positions and delays between them.

For Digital input:

- SPI.h
- SD.h
- Servo.h
- Arduino-songs Library by RobsonCouto

I developed two programs to accommodate to this scenario:

In the first program, communication with the microSD card module was established in SPI mode. A file was created to store musical notes, specifying their frequency and duration.

- The necessary Arduino pin for the SD card module's Chip Select (CS) pin was defined, and an object was created to handle file operations.
- The total number of notes was also calculated

1. void setup():

- Started serial communication and plugged in the microSD card.
- On the card, I created a "note.txt" file.
- Iterated through the musical notes array, writing to the file the total number of notes, frequency, and duration.
- Closed the file

The main objective of the second program is to play the musical notes read from the microSD card. I achieved this by utilizing a buzzer (for verification) and specifically through the Tesla coil.

- I defined pins for the servo motor, SD card module, and the buzzer/digital signal. Additionally, I created an object to handle file operations.

1. **void setup():**

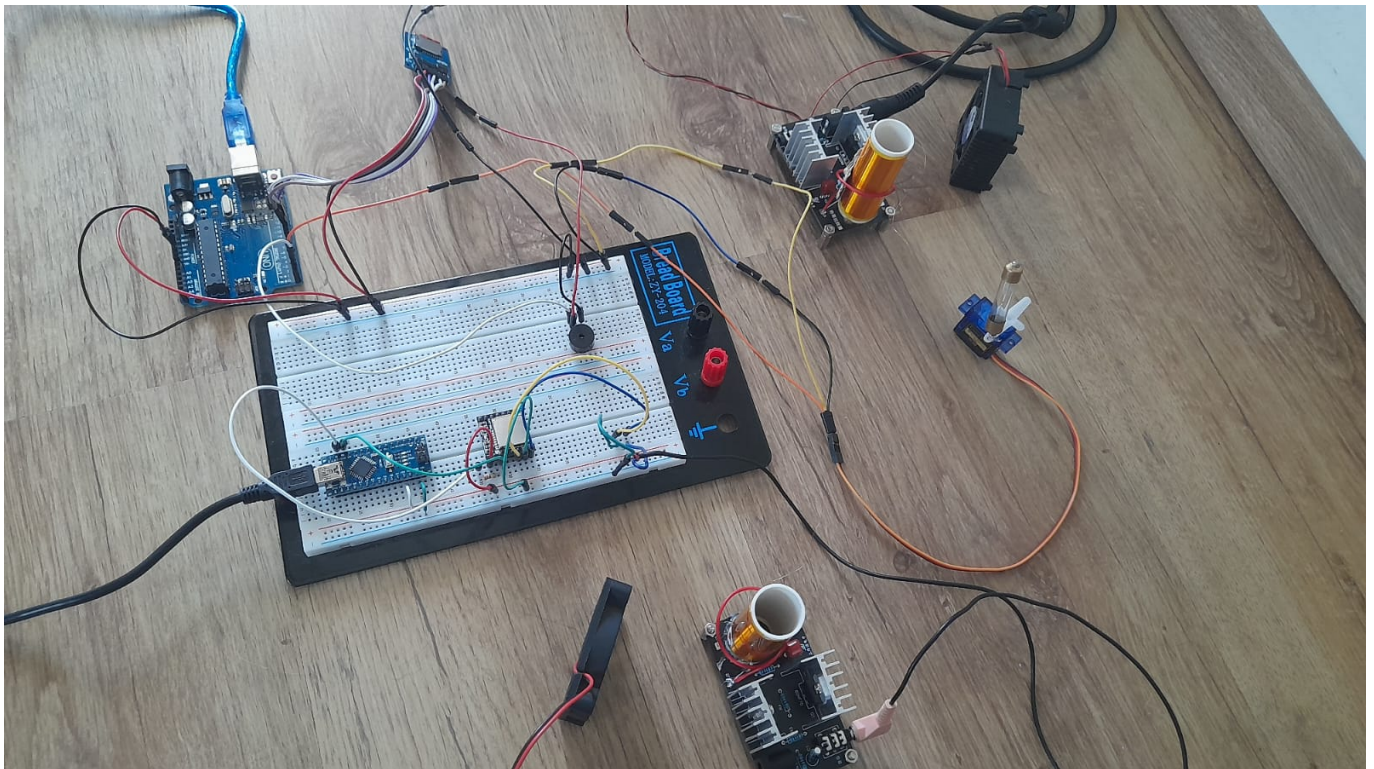
- Serial communication was initialized and connected to the microSD card and servo motor.
- The file was opened to retrieve the total number of notes, frequency, and duration.
- Frequency and duration values were stored in the `note[maxNotes][2]` matrix.
- Closing the file, the matrix was iterated to calculate the note durations.
- The corresponding sound was generated using the `tone()`, `delay()`, and `noTone()` functions.

2. **void loop():**

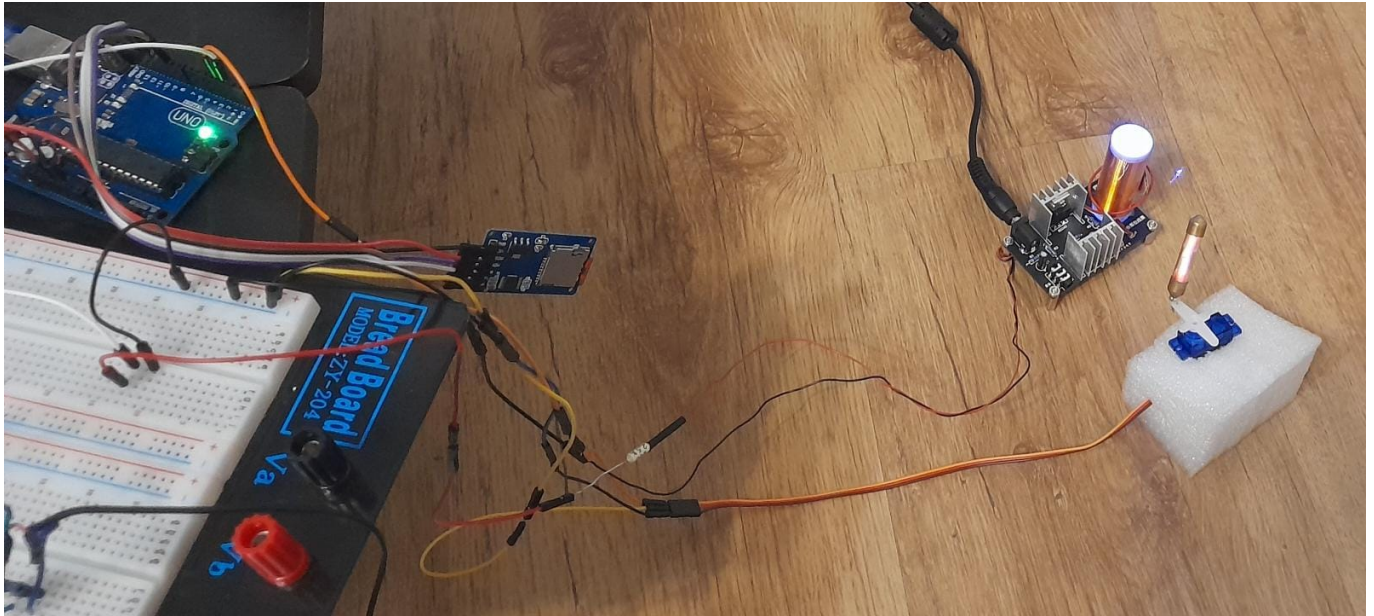
- Called the `danceServo()` function.

The `danceServo()` function serves the same purpose as it does for the analog signal.

Rezultate Obținute

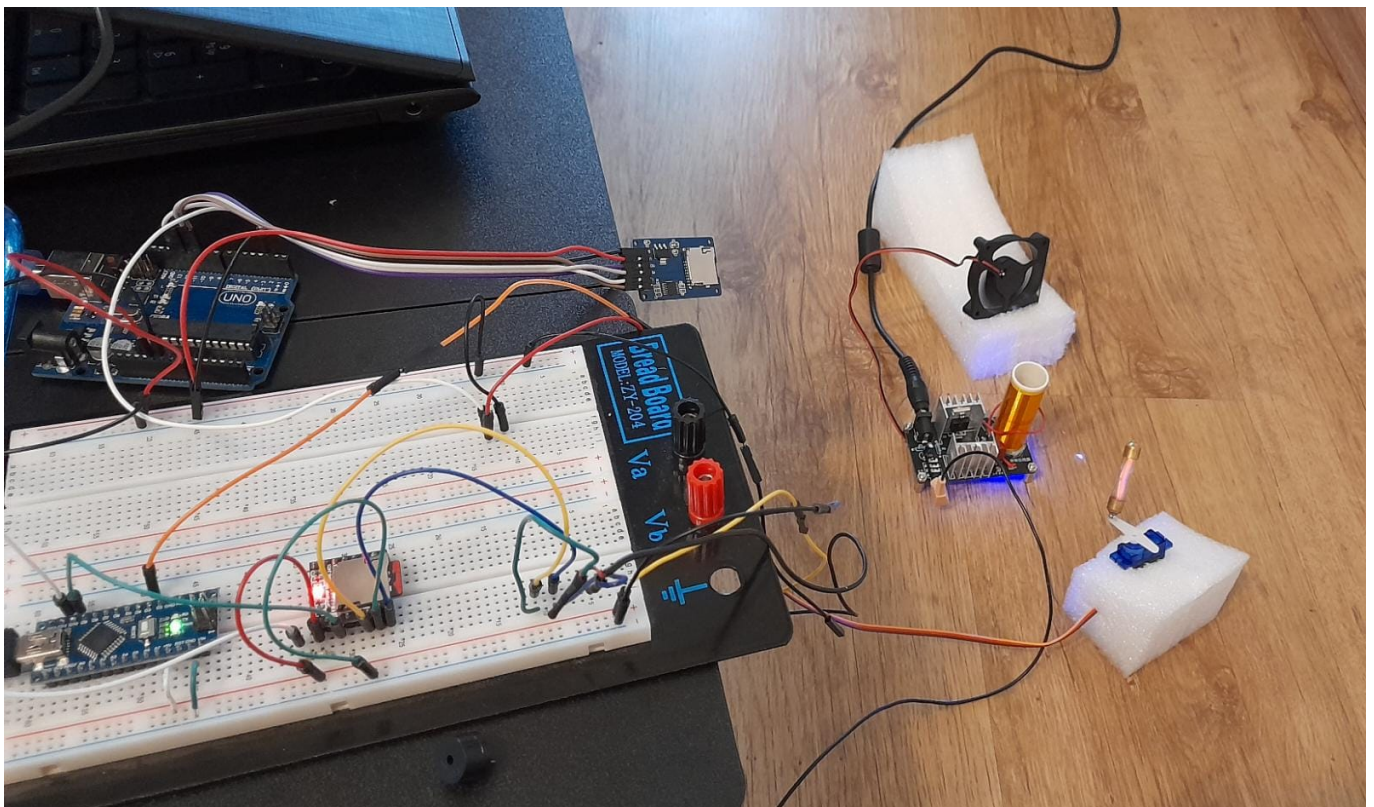


After numerous experimental attempts, I successfully achieved a high voltage electric arc in both cases and managed to transmit analog and digital signals to the Tesla Coil Slayer Exciter circuit. However, there is an unfortunate issue with the sound (musical notes) transmitted through the secondary coil using the digital path. The signal is muffled due to radio frequency interference that interferes with the digital signal from the Arduino. I used a passive buzzer to ensure that the sound was properly propagate in both situations.



In an attempt to address an issue, I added a fan for the BJT transistor to prevent overheating, but it had a negative impact on the sound quality due to altering the received voltage, so I decided to remove it. Due to the rapid overheating of the transistors, even with heat sinks and fans, I cannot afford to keep the Tesla Coil circuit connected to power for an extended period of time.

Furthermore, I encountered some inconvenience when transmitting melodies from the MP3 player module to the circuit via the audio jack cable. Occasionally, the connection between the audio jack plug and the stereo audio socket is unreliable, causing interruptions in the audio playback.



On a more interesting note, it's fun to watch as the gas discharge bulb modifies its intensity when it approaches an induced electromagnetic field, and I was able to demonstrate this effect using a servo

motor. 

Concluzii

The Tesla Coil project proved to be an incredibly enjoyable and rewarding experience for me. It provided a valuable opportunity to gain practical knowledge while exploring into the fascinating realm of electrical engineering and automation. By incorporating various technical concepts and utilizing Arduino, I was able to explore a wide range of subjects that I had learned.

The project was undoubtedly laborious and provided many challenges along the way. I had to adjust and work with the resources that were available. Despite obstacles, it was a rewarding accomplishment. Through rigorous research, careful component selection, system assembly, and code development, the long endeavor provided a deep understanding of both hardware design and software development, relying on the application of electrical and software engineering skills.

This comprehensive set of abilities gained from the project provided a solid foundation for my introduction into the realm of Arduino and the broader sphere of technology. I am grateful for the experience and knowledge that not only provided me joy but also broadened my horizons in this exciting field.

Download

[teslacoilsecrets.zip](#)

Journal

- 26.04.2023

I began to consider which project might pique my interest. I did some preliminary research on potential future projects.

- 03.05.2023

After a lot of thought, I chose to take on the Musical Tesla Coil project as a challenge, which would reproduce musical notes using high voltage sparks as a digital input signal from an Arduino. I selected this because I enjoy combining electrical and software engineering. In order to have a starting point, I began to conduct in-depth studies into the Tesla Coil phenomenon. Furthermore, I was looking for suggestions on the appropriate sort of Tesla Coil circuit to use for the requirements of my project.

- 04.05.2023

I decided to build my musical tesla coil slayer exciter circuit from scratch, so I went through the hardware components, looked at their datasheets, and calculated what materials I needed to make the primary and secondary coils (number of turns, gauge wires, and secondary coil output voltage). In

case that my circuit implementation fails, I have considered a backup plan (a little music tesla coil plasma speaker).

- 05.05.2023

I ordered some of the components I needed for my project and had others on hand. My progress will determine what I order later. I initially ordered -Arduino medium Kit from Robotlinking -TIP31C Transistor x2 -IRFP460 MOSFET N-Channel x2 -IRFP250 MOSFET N-Channel x2 -Transistor Heat Sinks x2 -Capacitors -4N35 Optocoupler x1 -10k/47k Resistor x1 -PVC Pipe (1 in. Diameter, 3.93 in. Length) x1 -30 Gauge Enamel Coated Wire -26 Gauge Rubber Coated Wire (Solid Core) -Jumper Wire -Power Supply (home) -Universal PCB board -Servo motor x1 -Micro- SD Card Module x1 -MP3 player DFPlayer Mini Module x1 -Soldering iron x1 (home)

- 09.05.2023

I acquired my components and began sketching my planned circuit on paper.

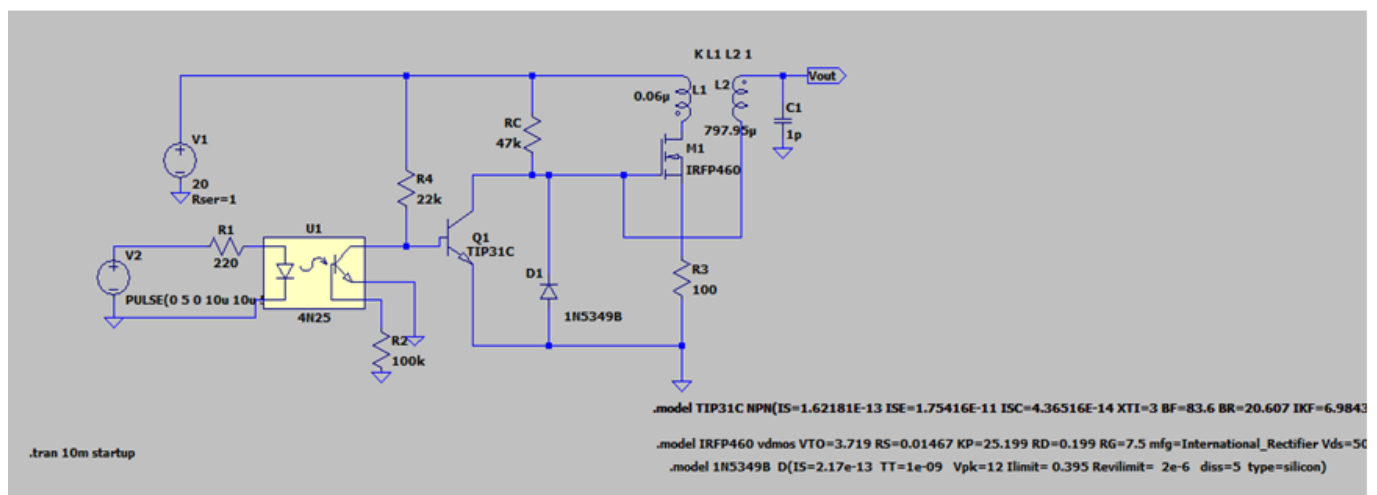
- 15.05.2023

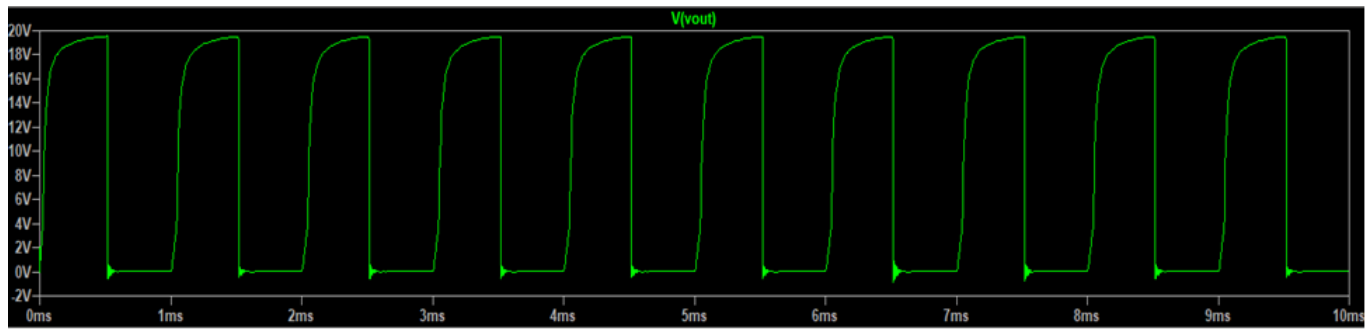
Because making and testing my circuit directly would have been too dangerous, I tried it on the free SPICE simulator program (LTSPICE).

- 18.05.2023

Unfortunately, after numerous experiments and simulations of various circuits in LTSPICE, something was not right and I was unable to carry out a definitive solution for my slayer exciter circuit. My proposed slayer exciter circuit used a npn bjt (TIP31C) to drive the Mosfet (IRFP460) gate.

Slayer Exciter circuit cu NMosfet Gate Driver via BJT(NPN)





- 19.05.2023

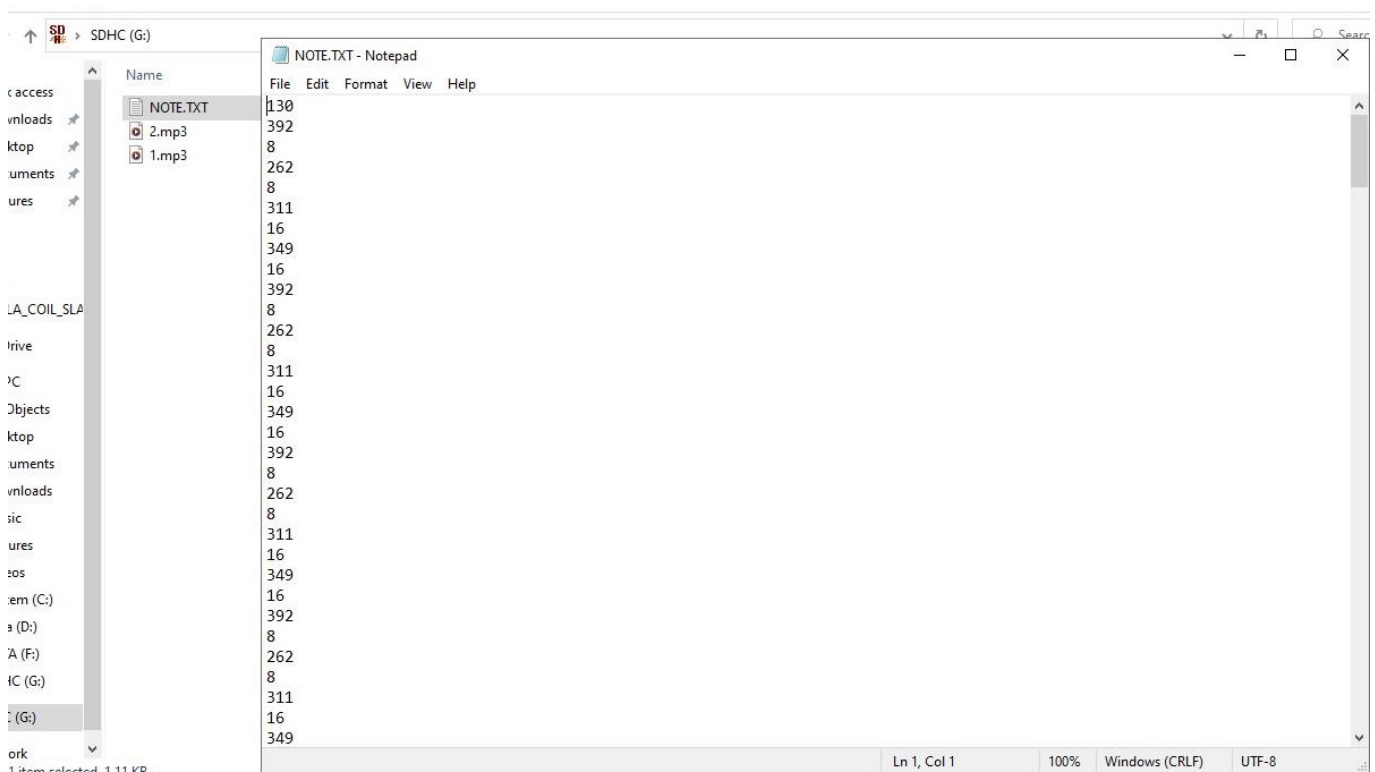
I decided to implement my backup plan for the circuit and also considered trying an alternate means of producing music via the spark, but this time the device would accept an analog signal input.

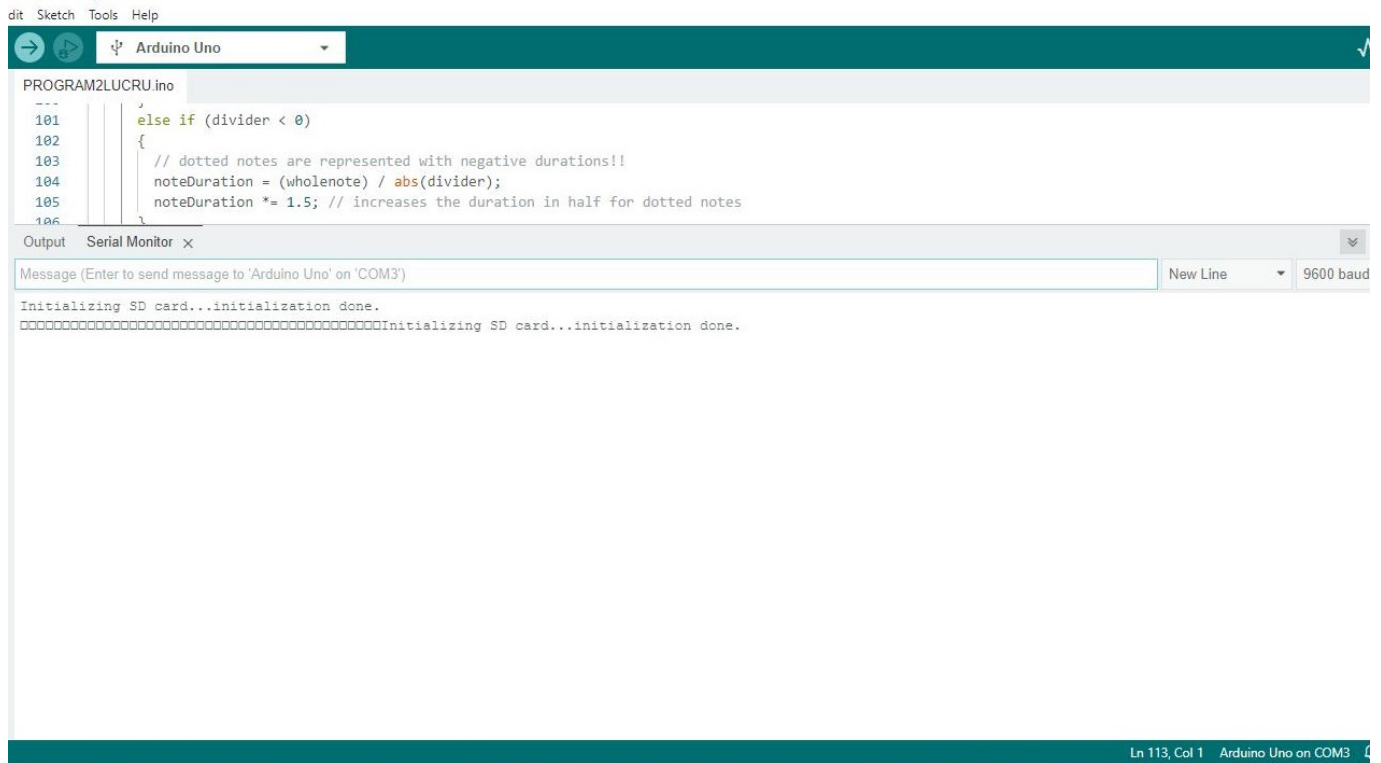
- 21.05.2023

I began connecting the components together on a breadboard and tested their functionality.

- 23.05.2023

I began putting the software design into practice.





```
PROGRAM2LUCRU.ino
101     else if (divider < 0)
102     {
103         // dotted notes are represented with negative durations!!
104         noteDuration = (wholenote) / abs(divider);
105         noteDuration *= 1.5; // increases the duration in half for dotted notes
106     }
```

Output Serial Monitor x

Message (Enter to send message to 'Arduino Uno' on 'COM3')

New Line 9600 baud

Initializing SD card...initialization done.
.....Initializing SD card...initialization done.

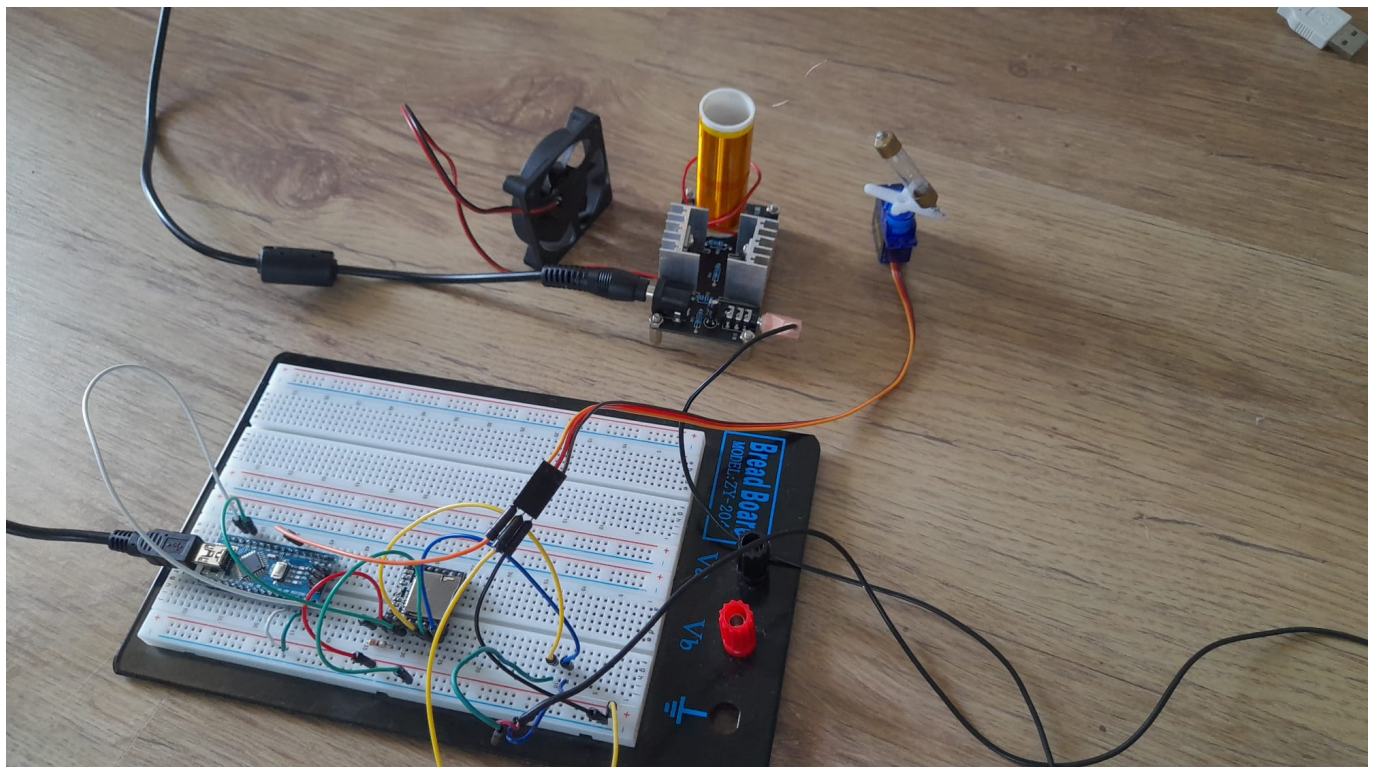
Ln 113, Col 1 Arduino Uno on COM3

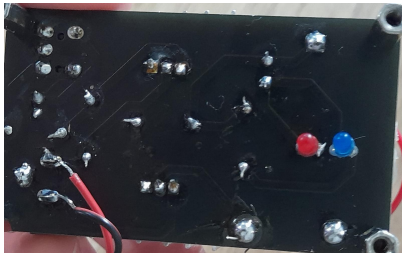
- 26.05.2023

I completed all of the project's code implementations.

- 27.05.2023

Begin assembling and testing the project for both situations with all necessary equipment (e.g., oscilloscope). They are effective!





- 28.04.2023

I'm putting the finishing touches on.

- 29.05.2023

The project is completed.

Bibliography/Resources

Tesla Coil Slayer Circuit

- <https://arcattack.com/resources/how-a-tesla-coil-works/>
- <https://members.tm.net/lapointe/HowItWorks.htm>
- <https://www.instructables.com/Slayer-Exciter-Circuit-Poor-Mans-Tesla-Coil/>
- <https://circuitdigest.com/electronic-circuits/how-to-make-a-mini-tesla-coil-9v>
- <https://www.electromaker.io/project/view/arduino-musical-tesla-coil>
- <https://www.electroboom.com/?p=521>
- <https://leap.tardate.com/electronics101/teslacoil/musicalkit/>
- <https://www.scienceofgadgets.com/post/how-does-the-tesla-coil-make-lightning-and-play-music>
- https://www.youtube.com/watch?v=a4zv7_wYWdg
- <https://www.youtube.com/watch?v=b877YL1f9zg>

Components

- <https://www.ee-diary.com/2023/01/tip31c-transistor-guide-for-arduino.html>
- <https://bestengineeringprojects.com/interfacing-optocoupler-with-arduino/>
- https://user.eng.umd.edu/~neil/enee313/BJTnotes_05_6_10.pdf
- <https://www.npntransistors.com/irfp460-mosfet/>
- <https://www.instructables.com/Mini-Musical-Tesla-Coil-Kit/>
- <https://www.allaboutcircuits.com/tools/coil-inductance-calculator/>
- <https://maker.pro/arduino/projects/how-to-use-the-dfmini-player-mp3-module-with-arduino+>
- <https://lastminuteengineers.com/arduino-micro-sd-card-module-tutorial/>
- <https://howtomechatronics.com/how-it-works/how-servo-motors-work-how-to-control-servos-using-ar>

[duino/?utm_content=cmp-true](#)

- <https://www.youtube.com/watch?v=QbgTl6VSA9Y>
- <https://www.optimusdigital.ro/ro/memorii/1516-modul-slot-card-microsd.html>
- <https://www.electronicwings.com/sensors-modules/servo-motor>
- <https://lastminuteengineers.com/servo-motor-arduino-tutorial/>

Arduino

- <https://www.arduino.cc/reference/en/libraries/sd/>
- <https://www.arduino.cc/reference/en/language/functions/communication/spi/>
- <https://docs.arduino.cc/learn/built-in-libraries/software-serial>
- <https://reference.arduino.cc/reference/en/libraries/dfplayer-mini-mp3-by-makuna/>
- <https://www.arduino.cc/reference/en/libraries/servo/>
- <https://docs.arduino.cc/built-in-examples/digital/toneMelody>
- <https://github.com/robsoncouto/arduino-songs>
- <https://ocw.cs.pub.ro/courses/pm/lab/>

Export to PDF

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/apredescu/teslacoilsecrets>

Last update: **2023/05/30 13:43**

