

# Musical Instrument Digital Interface Analyzer

## CPSC 499

By Christopher Villanueva

Project: <https://github.com/chrisvillan/CPSC499>

### What is a Musical Instrument Digital Interface and MIDI?

It's an interface that can convert noise from an instrument into digital data. It captures numerous musical data such as tempo, tone, pitch, length, and more.

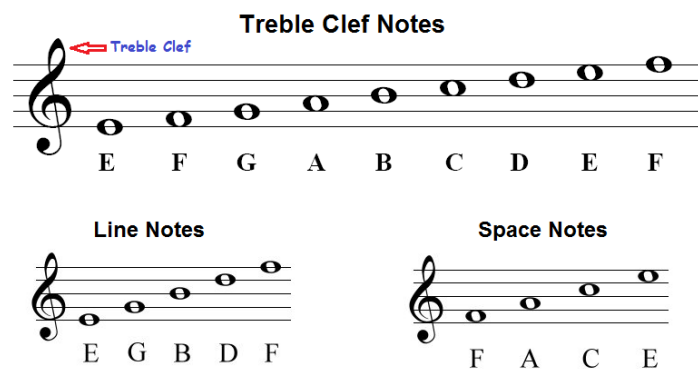
**Objective of Project:** to capture MIDI notes and develop an algorithm that determines the *mood* of the song.

### What makes a song happy or sad?

There really isn't a universal or standard rule that makes a song happy or sad. Since mood and how a song is interpreted can vary from one person to another. There are certain trends and theories that can help narrow it down.

### Music Notation

For the entire project, I'll be strictly using Treble Clef notes which is a set of notes that range within a certain pitch. This range usually carries the melodies of most popular songs. The picture below shows musical notation on music sheets and will be referring to this notation often.



### Major vs Minor Keys

Major and minor keys are different patterns in a scale. A musical scale is a set of musical notes. There are 12 different musical notes. The notes are incremented in what is called half steps. The various notes are A, A#, B, C, C#, D, D#, E, F, G, G#. The *key* is a preset combination of these notes that creates the basis of that song. For example, the most basic key, is C major, which uses C, D, E, F, G, A, B. An E major uses, E F# G# A B C# D#.

Generally, major keys are associated with “happy” and minor keys are associated with “sad”. A major key is defined by the intervals between the notes. There isn’t really a universal proof for this. Most songs that are considered “happy” tend to be in a major key. From another perspective, the unique intervals in the major and minor key really change the overall tone.

Going back to the key of C Major, a scale in that key would have the root note as C. Then a pattern follows it W-W-H-W-W-W-H where W is a whole step interval and H is a half-step interval.

Originally the algorithm was to differentiate between major and minor keys of a song while the notes were inputted. The algorithm would not know what key the song is in and would need to determine it based on the musical input. It first calculates if any of the combinations match a certain key, however a new problem came up.

Other than the key being identified in the beginning, the easiest way for a song’s key to be identified is its key signature which is determined by how many sharps or flats which are symbolized as (#) or (b). So, for a G major scale, there would only be one note (F) that is sharp and the rest are normal. It would have A, B, C, D, E, F#, G. However, this method isn’t reliable because of relative keys. A relative key is a key that is in the opposite category but has the same key signature. In other words, the G Major scale’s relative key is E Minor. They both have only F# but the difference is their root note. The root note in G Major scale is G and the for E Minor it is E. Usually the confusion is avoided by stating the major or minor key on the music sheet. It gets trickier when it’s being determined by the notes.

In many songs, even if the key sets the basis of what notes are sharp or flat, there could be additional alterations that aren’t defined by the key. A note can be altered by a half step anywhere in the song to create more complex melodies. With that being the case, we have to look at another analysis of determining the key.

Triad chords are another aspect within a song. A chord is a combination of notes played simultaneously to produce a unique sound. A triad chord is a set of three notes that is based on its root note. Two categories of a chord are major or minor. A major chord’s first out of the three notes is its root note. That root note determines that name of the chord and how the other notes are chosen. The second note is four half steps above the root note. The third note is three half steps above the second note. Altogether, major chords are defined as its root, major 3<sup>rd</sup>, and perfect fifth note. A minor chord has similar patterns, but instead of a major 3<sup>rd</sup>, it has a minor third which is three half steps above the root note. The perfect fifth of a root note is four half steps above the second note.

Identifying triad chords and combining them with basic rules of major and minor keys can help in determining the true key of a song by just analyzing the sequence of notes. I’m hypothesizing that a triad chord’s notes cannot be altered as any alteration would change the triad chord overall. A C major scale can correlate to a C major chord. That C major chord’s three notes are C, E, and G. In a C major scale, the C E G cannot be altered but some of the notes may be able to change.

Major Chord Root + Major 3 <sup>rd</sup> + Perfect Fifth Root + 4 Half Steps + 3 Half Steps		Minor Chord Root + Minor 3 <sup>rd</sup> + Perfect Fifth Root + 3 Half Steps + 4 Half Steps
<p>C Major [C – E – G]</p> <p><u>C - C# - D - D# - E - F - F# - G - G# - A - A# - B</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R M3 P5</p>	0 Sharps	<p>A Minor [A – C – E]</p> <p><u>A - A# - B - C - C# - D - D# - E - F - F# - G - G#</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R m3 P5</p>
<p>G Major [G – B – D]</p> <p><u>G - G# - A - A# - B - C - C# - D - D# - E - F - F#</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R M3 P5</p>	1 Sharp	<p>E Minor [E – G – B]</p> <p><u>E - F - F# - G - G# - A - A# - B - G# - A - A# - B</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R m3 P5</p>
<p>D Major [D – F# – A]</p> <p><u>G - G# - A - A# - B - C - C# - D - D# - E - F - F#</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R M3 P5</p>	2 Sharps	<p>B Minor [B – D – F#]</p> <p><u>B - C - C# - D - D# - E - F - F# - G - G# - A - A#</u></p> <p>1 2 3 4 5 6 7 8 9 10 11 12</p> <p>R m3 P5</p>

This logic can help in distinguishing the true key between relative keys. For example, in the song “Greensleeves” by Yasunori Mitsuda, the first line of notes are E, G, A, B, C, B, A, F#, D, E, F#, G, E.



When the notes are being analyzed, the 8<sup>th</sup> note will be an F#. This is the first clue to see to determine what key it is. A scale with an F# can either be an E Minor or G Major. The way to determine which the song is lies on the next line. The notes are E, D#, E, F#, D#, B, E, G, A, B, C, B, A, F#.



On the next line, the second note has a D# which isn't set in the original key. If we determine an E Minor chord and a G major chord, an E Minor chord has the notes E, G, and B. For the G major chord, the notes are G, B, and D. Since the G major chord has a note D in its triad form, that D is crucial for that chord

which translates to the key. Therefore, if the song was in the key of G major, the D can't turn into a D#. In conclusion, the song is in E Minor, which is the true key of the song.

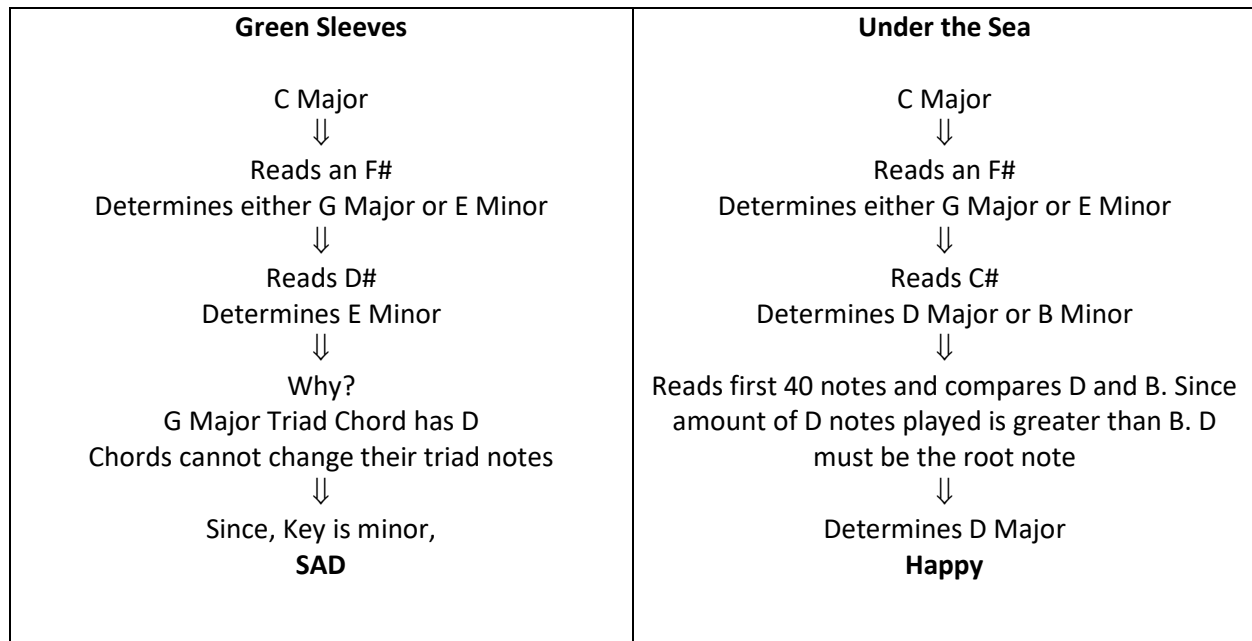
The flaw in the above algorithm is that it depends on a unique alteration to determine the key of a song. For the song "Under The Sea" in Walt Disney's Little Mermaid, there is no altered notes outside the given key. Therefore, the algorithm wouldn't be able to determine between the relative keys. There is another way of choosing between relative keys and that's understanding what the purpose of the root note is.

The root note sets the basis of the song and sets the tonal center. The tonal center is the note where other notes surround it, which usually is the root note. In most songs the root note of the scale it is in, is one of the most played notes in the song. That concept will be the solution that overcomes the previous problem. For the song "Under The Sea", we are able to identify the key signature and will need to choose between the relative keys.



By the 19<sup>th</sup> note we've identified an F# and C# which indicates it being either in D Major and B Minor. Throughout the first page there is no deviation from notes being normal with the F# and C#. Therefore, using the first algorithm won't be able to choose between the two. Using the tonal center concept, we can count D's against B's. Analyzing the first 40 notes, there is 16 D's and 0 B's. In this case, we can be confident that D has more of a tonal center than B, concluding that it is D Major, which is the true key of the song. Even though this can be applied in substitution of the first algorithm, the first algorithm can be a quicker indicator and better accuracy.

Overall the algorithm for the two songs:



### Program

The program language that is used is Python. It's the standard language for the Raspberry Pi programs. In the program, MIDIUtil Python library is used to create MIDI objects with values of the note, time, duration, tempo, and volume. Pygame is a platform in Python that has a MIDI player module. This will allow for MIDI files to play through the audio.

First, the program initializes the MIDI objects that contain note, time, duration, tempo, and volume values. Then notes are added by using the ".addNote" function and passing certain arguments. As each note is added into the object it's also appended into an array for the algorithm. Unfortunately, I couldn't figure out a way to read the exact notes from MIDI files. That is why an array is made to keep track of the notes. After all the notes are added the algorithm above is implemented to determine which key the song is in. The last steps are for the MIDI objects to convert into a ".mid" file and for that file to be played with the operating system's audio interface.

### Progression of Project

The hardest parts of the project was researching through the musical theory and attempting to use the a Raspberry Pi LED Matrix for an output. In terms of the musical theory, the difficulty wasn't understanding the musical terminology and concepts, it was narrowing down the factors in music and how that affects mood. Trying to find a balance between objectively analyzing songs but also considering subjective interpretations was tricky. The result of the project heavily relied on major and minor identification. However, there are exceptions where that identification won't work because of the infinite possibilities of music. Some songs in minor keys can deviate from being sad, if you add more

variables such as faster tempo, distorted sound and multiple tracks. I had to eliminate variables and figure out a way where the algorithm will be as consistent as possible.

The other hard part of the project was creating a visual output of the songs to further display the mood of a song. The original idea was using an LED Matrix to display various colors and shapes based on the song. This required a little bit more research in color psychology. Colors are heavily influenced by personal experience but there are some trends. According to ArtTherapyBlog, orange is associated with happy and energetic emotions and the color brown is associated with sadness. I wanted to display this interpretation on a 32x32 LED Matrix. At first, getting the connection between that and the raspberry pi took some work. There were some electrical failures and programming setup bugs. Unfortunately, even after that was fixed, I couldn't implement the LED Matrix display to the project, because it used a large program that would have required a lot more time to alter. Also, the only documentation I could find online was in a different programming language than Python. Overall, it would have been great to include that, but it wasn't the biggest priority on the project.

Overall, the result was mainly focusing on the analysis of the music and developing a way to measure and quantify the factors that influence the mood of the song.

### **The Big Picture**

I still think that this project has a big scope and can be further improved. If I had a lot more time and more knowledge, I believe incorporating artificial intelligence into the algorithm will really create a very accurate analysis. At a certain point, common trends and musical structure for moods can only be so general. It's because of the hard prediction of a person's interpretation of it. There also might be many factors that aren't currently investigated yet that might make this more accurate. That's where I think artificial intelligence can really take this to the next level. It may be able to interpret a song just like how a person naturally does it far better than any hard-coded algorithm.

### **References**

<http://www.piano-keyboard-guide.com/treble-clef.html>  
<https://www.musical-u.com/learn/major-minor-keys/>  
<https://www.shawnboucke.com/major-vs-minor.html>  
<http://www.indiana.edu/~emusic/361/midi.htm>  
<http://www.piano-keyboard-guide.com/major-chords.html>  
<http://www.piano-keyboard-guide.com/minor-chords.html>  
<http://www.simplifyingtheory.com/relative-minor-major/>  
<https://www.earmaster.com/wiki/music-memos/what-are-chords-in-music.html>  
<http://michaelkravchuk.com/wp-content/uploads/2017/02/Greensleeves-E-Minor.pdf>  
<http://hubguitar.com/music-theory/identifying-the-root-note>  
<https://www.aboutmusictheory.com/major-scales.html>  
<https://www.helpscout.net/blog/psychology-of-color/>

<http://www.arttherapyblog.com/online/color-psychology-psychologica-effects-of-colors/#.Wvp7Gogvzb0>  
<https://ledgernote.com/columns/music-theory/circle-of-fifths-explained/>