ASSISTIVE DIGITAL PIANO

ECE 445: SENIOR DESIGN
PROJECT PROPOSAL

Shruti Chanumolu (shrutic2), Anna Shewell (shewell2), Jae Kwak (jaekwak2)

---

1 Introduction

1.1 Objective

Our project aims to develop a digital piano (hereafter referred to as 'keyboard') for beginners. We want to make learning to play the keyboard fun and easy, thereby minimizing the requirement for professional instruction. To do this, our keyboard will implement many novel features: it will use LEDs and colored gloves to highlight correct fingering; after a song has been played, it will report the timing of the notes and the speed at which the keys were hit so that users can identify what they need to improve; and it will provide a strict "Learning Mode" where the keyboard will wait for the correct finger to press the correct key before it plays the note.

1.2 Background

With all the resources one can find on the internet nowadays, it's easier than ever to teach yourself new skills. Learning how to play the keyboard is no exception. However, many self-taught players often lack the technical skills that other players possess. For example, self-taught players might not play at the right tempo or strength or they might rely too much on their index fingers. Without a knowledgeable teacher there to correct them when they make a mistake, players can accidentally reinforce bad habits that will prevent them from playing at a high level.

However, taking lessons requires money and the availability of a good teacher. The average cost of piano lessons in the U.S. is between $15 and $40 for 30 minutes of instruction. If a student takes 1 lesson per week, that adds up to approximately $720 to $2160 per year. In addition, it's been found that many beginning students lose their motivation to play in part because they don't like their piano teacher [1]. Thus, we want to design a keyboard that would cost little more than a regular keyboard, but would allow the user to move at their own pace and improve their technical skills.

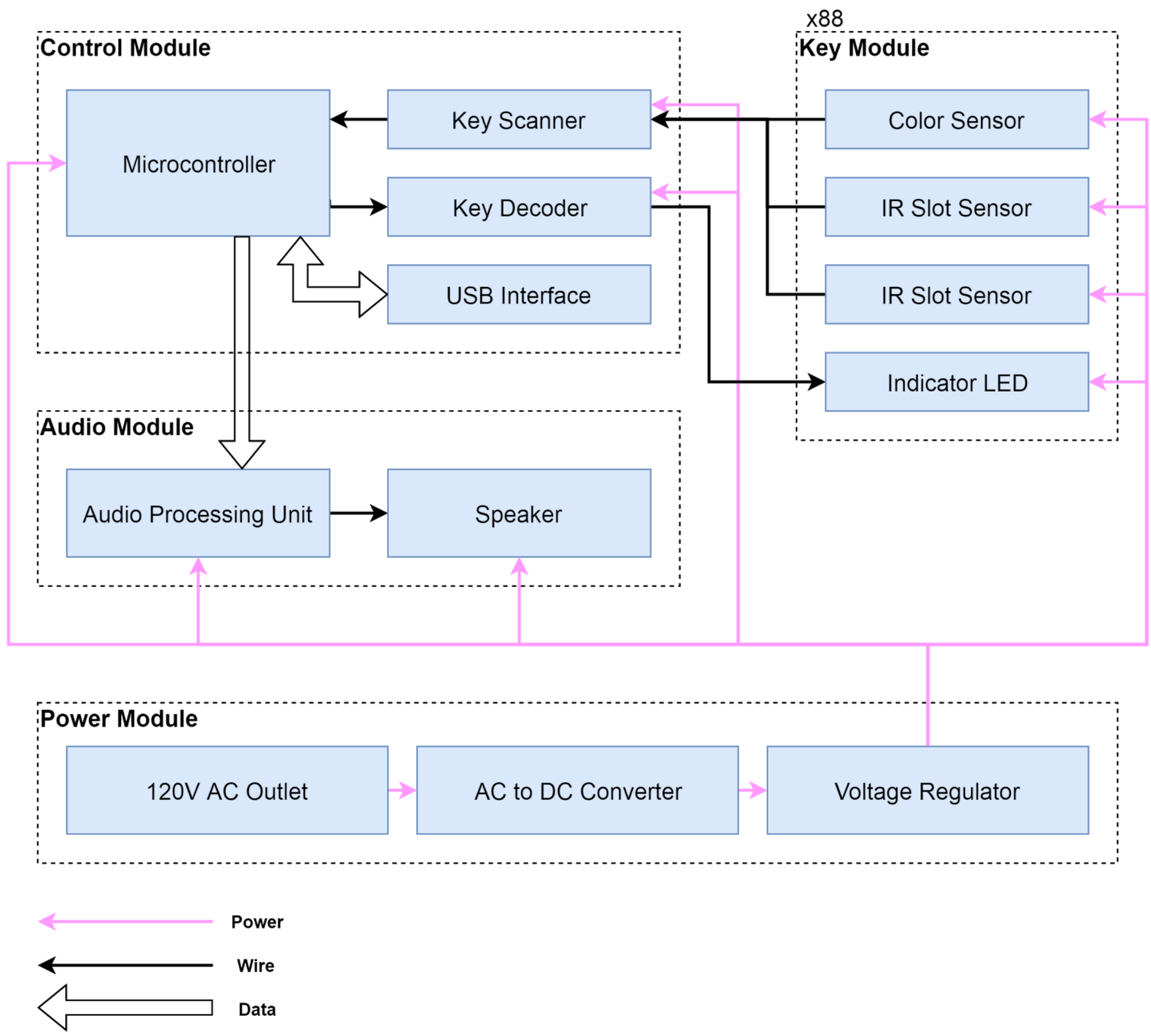
1.3 High-Level Requirements

- The keys must be able to identify which finger is used to press which key within 5 ms with 95% accuracy.
- The piano must be able to decode song files that we design, and not allow incorrect key/finger inputs to play sounds while in "Learning Mode".
- The project's user interface must be able to read rhythm and keypress data from the piano and then evaluate and display rhythmic accuracy and keypress accuracy to the user.

2 Design

2.1 Block Diagram

Figure 1: Block Diagram



2.2 High Level Design Description

Our piano will operate with four distinct modules: a key module, control module, audio module, and power module. The key module will gather input about which keys were pressed, how fast keys were pressed, and which finger was used to press which key. It will then send this information to the control module. The key module will also control LEDs in the keys to show the player which key to press next and with which finger. The control module will be responsible for scanning all inputs, processing the key press and song data, and interfacing with the USB port and audio module. The audio module is responsible for generating the sound output. Lastly, the power module is responsible for supplying all circuit components with adequate current and voltage.

2.3 Physical Design

Color sensors will be inside the keys to detect finger color. IR slot sensors will be placed beneath the keys, and will be triggered by a thin protrusion on the bottom of the key that will trigger them sequentially when the key is pressed. All measurements and locations at this stage of the design are approximated. Piano key measurements were taken from a real piano and online resources.

White key depth: 25mm, depression distance: 20mm
Black key depth: 10mm, depression distance: 8mm

Figure 2: Approximate Key Dimensions (not to scale)

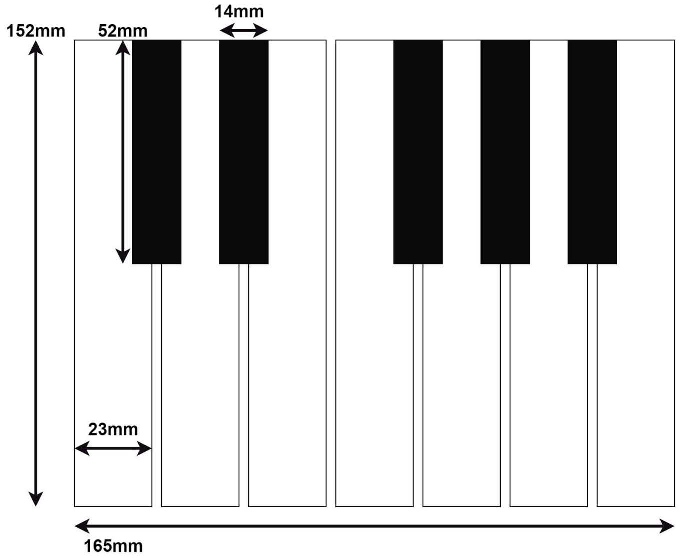
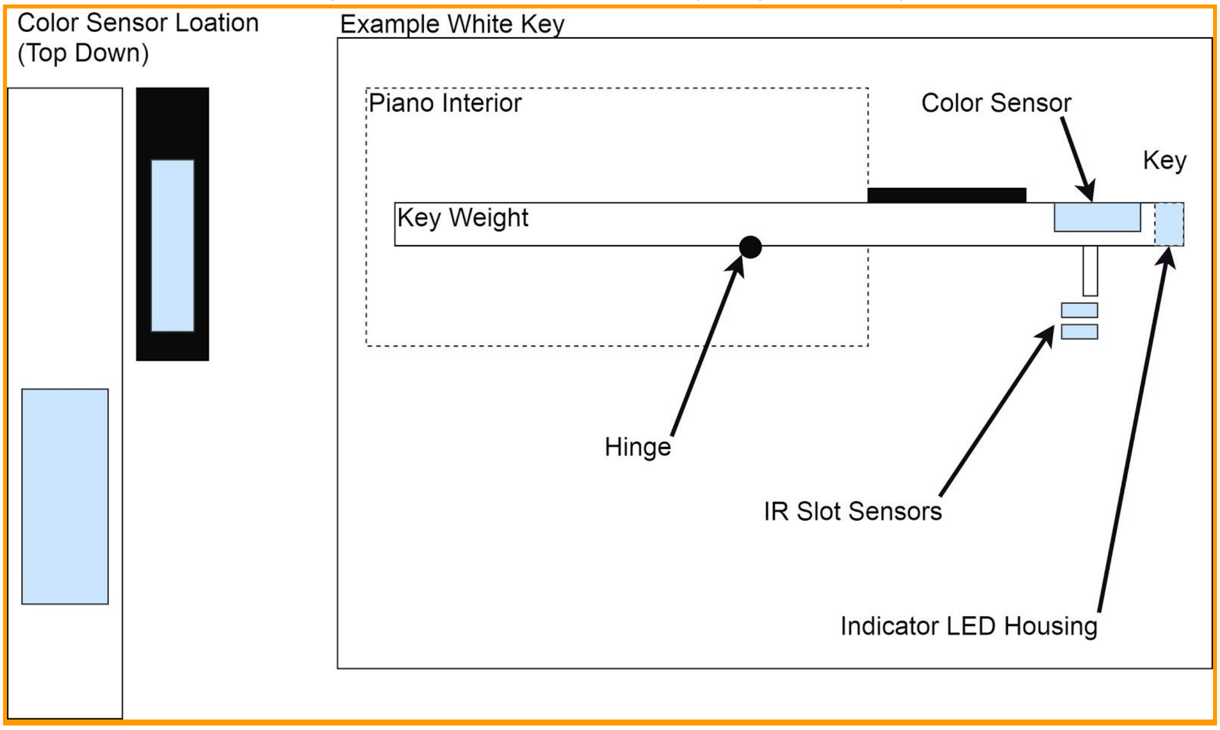


Figure 3: Approximate Sensor Layout (not to scale)



2.4 Functional Overview

Important Calculations and Estimates:

The fastest possible piano key press will be assumed to hit the IR slot sensors with less than 1 millisecond of delay from the first to the second. In the case that both sensors are triggered in the same round of a key scan, our microcontroller should still interpret this is a legitimate key press and play the note at maximum volume. A slow keypress will likely be around 20-100 milliseconds of delay. Our key scanning should be fast enough to detect the nuance in these presses, and therefore scan all the keys at intervals of less than .25 milliseconds. Note that all of these measurements are at best educated guesses in this point of the design. The delays were measured with an arduino push button prototype.

Temporal delay in the input vs sound output is also a consideration. Sound travels at approximately 1 meter per millisecond. Since the literature on human time perception is wide and varied and very specific to the situation, we will approximate that the maximum acceptable delay from piano input to sound output is 10 milliseconds.

The visible color spectrum is around 400nm to 700nm. Since there are 10 fingers, we will need to partition segments of the color spectrum for finger color detection.

2.4.1 Power Module

We plan to power the system with a 120 V AC power supply. We need a AC to DC converter to convert the 120 V AC to 12 V DC to power all components of our circuit. In addition voltage regulators are required to regulate the 12 V DC power supply as per the below requirements

Requirements:

- The power supply must provide a voltage in the range of 5+/0.25 V steady DC for a current load up to 20 +/0.25 mA for the LEDS, speakers and the components in the control module.

- The power supply must provide a voltage in the range of 3.3+/0.25V for a current load up to 10/+0.25 mA for the color sensors and IR sensors.

2.4.2 Key Module:

This module consists of all electrical components that will be place inside or on the keys of our digital piano and includes the IR sensors, color sensors and LED indicators.

IR Sensors

The IR sensors will be primarily used to identify if a key is being pressed. The voltage outputs of the IR sensors will be sent to the key scanner sub-module (of the control module) which will identify which key is being pressed. The IR sensors will also be used to measure the speed of each key press (hence why there are 2). Detection will begin a few millimeters from the initial position to a few millimeters before full key depression for adequate wiggle room.

Requirements:

- Top sensor must be able to detect key press when a white key is pressed 3 mm from its resting position, and when a black key is pressed 1 mm from its resting position with at least 99% accuracy.

- Bottom sensor must be able to detect key press when a white key is pressed 18 mm from its resting position, and when a black key is pressed 7 mm from its resting position with at least 99% accuracy.

Color Sensors

Color Sensors will be placed on the keys to detect if the correct finger is being used to press the key. This will be done by reading the colors on the fingers of a set of included gloves and sending this data to the control module. The control module will then compare the frequency of the signal sent by the color sensor to the range of frequencies corresponding to the color of the finger that was supposed to be used. Thus, it can determine if correct fingering is being used to press the key or not.

Requirements:

- Must be able to identify 10 distinct colors, each within /+10 nm of wavelength.

- Must be able to identify a color within a range of 0.5 mm/+0.25 above the key and within the full width of the key.

LEDs

LEDs are provided on each key as a guide for users. The keys that need to be played next will light up in 1 of 10 colors corresponding to the colors on the set of included gloves. Instead of using 10 separate LEDs, we plan to use 3 LEDS for each key: a red LED; a green LED; and a blue LED. We will use pulse width modulation to control the intensity of each of these LEDs to produce 10 distinct colors [3].

Requirements:

- Must be able to produce 10 distinct colors.

2.4.3 Control Module:

Key Scanner

The first purpose of this sub-module is to poll the keys and identify if a specific key has been pressed. This information will then be used by the microcontroller during key verification to determine if the key pressed is correct. If correct, the key information will be used by the audio module to produce the corresponding sound.

The second purpose of this sub-module is to identify which finger was used to press a key. The color sensors beneath each key will read the color on the finger of the glove as it descends towards the key. This information will then be used by the microcontroller during key verification to determine whether the correct finger was used. If the correct finger was not used and the keyboard is in “Learning Mode”, sound will not be produced and the song will not advance.

Requirements:

- Must be able to identify IR slot sensor inputs within 3 ms/+0.25 with at least 99% accuracy.
- Must be able to scan and output the frequency square wave with at least 90% accuracy.
- Must be able to identify at least 10 keys pressed within 1 ms of each other.

Key Decoder

The purpose of this sub-module is to light up the LEDs in the keys according to the song being played (decode the keys into colors). Information about which LEDs should be lit, the intensity of each LED, and the order in which they should be lit will be provided by the computer through the USB interface.

Requirements:

- Must be able to process and send digital signals to the LEDs within 3 ms/+0.25 with at least 99% accuracy.

Microcontroller [Key Verification and Key Press Speed]

The first purpose of this sub-module is to verify whether the key pressed is correct and whether the finger used to press the key is correct. If the key was not supposed to be pressed or was supposed to be pressed by a different finger and the keyboard is in “Learning Mode”, the audio module will not produce a sound and the key decoder module will not change the LEDs to reflect the next notes in the song. This also applies to situations where multiple keys need to be pressed at the same time (while in “Learning Mode”). For example, if a user misses a note in a chord, no sound will be produced and the song will not progress.

The second purpose of this sub-module is to determine the speed at which a key is pressed. This information will then be used by the audio module to determine the volume at which the corresponding sound should be played and will also be analyzed with the song data provided to the user.

Requirements:

- Must be able to decode the next keys in the song and scan the input from the key scanner at least 4 times per millisecond.
- Must be able to encode key input information and relay it to USB and audio controller at a rate of 100 kbps

USB Interface

The purpose of this sub-module is to provide an interface between our keyboard and the computer. This interface is necessary because the computer will be used to program song data (which keys need to be pressed in what order) into the keyboard, analyze performance data from the keyboard, and display that analysis to the user.

Requirements:

- Must be able to send and receive information via USB.

2.4.4 Audio Module:

Audio Processing Unit

The purpose of this module is to convert key presses into the corresponding waveforms that will produce the correct sounds for the speaker.

Requirements:

- Must be able to convert key press code (sequence of bits) into the corresponding waveform.

Speaker

The speakers will output the sound signals given by the audio processing unit.

2.4.5 Software:

Along with our physical keyboard, we will design software to send song data (which keys need to be pressed in what order) and collect performance information (speed of key press, tempo, etc.) and display that information via a user-friendly interface.

Requirements:

- Must be able to load song file into control module such that the LEDs are lit correctly and the verification of the notes works at least 95% of the time.
- Must be able to download key press data from the control module at a rate of 100 kbps.
- Must be able to process, analyze and communicate rhythm data such that a majority of beta testers are satisfied (would rank at least 7/10).

2.6 Risk Analysis:

The main components of the control module (key scanner, key decoder, microcontroller) pose the greatest risks to successful completion of the project. This module performs 4 integral tasks: it lights up the LEDs located on each key according to the song being played; it determines which keys, if any, have been pressed; it determines which finger was used to press each key; and it determines how fast each key was pressed.

Of all the things that could go wrong with these tasks, our biggest concern is the accuracy and precision of the color sensors. We are unsure of how the quality of lighting, finger position, and play style will affect the performance of the sensors, and whether our implementation of this design will be robust enough to account for these factors.

Our second biggest concern is calculating the speed of the key presses.This aspect of our design requires an additional IR slot sensor for every key, which is an additional measurement to scan in our key scanner. With such a large number of inputs, our key scanner gets more complex and reduces scanning speed, as more inputs need to be scanned and more data needs to be processed.

In terms of addressing these risks, should we run into serious complications with calculating the speed of key presses, we will abandon this aspect of our design as it's not particularly integral. To ensure that our color sensors will perform robustly, we will begin building and testing a prototype of this sub-module first. Hopefully we will discover any serious design flaws early enough to fix them.

3 Safety and Ethics

3.1 Safety Issues:

As with anything that connects to a wall outlet, there's a minimal chance of electrocution or fire if the plug/circuitry is tampered with, misused, or accidentally compromised. Should we or someone else choose to market our design, we would ensure that a safety warning is included in the packaging as per the IEEE Code of Ethics #1, "... to disclose promptly factors that might endanger the public or the environment," [2].

3.2 Ethical Issues:

The most likely ethical issue we'll face will be accurately reporting the capabilities of our project. It is difficult for us to tell at this stage how accurate and robust our design will be. Nevertheless, we will maintain the integrity of our work and be honest even if the final product falls below expectations in keeping with the IEEE Code of Ethics #3 [2].

Also of great importance to our project is the IEEE Code of Ethics #7 [2]. We will welcome criticism of our work and do our best to fix any errors in our project. In keeping both with the aforementioned IEEE policy and also with one of the University of Illinois's most important policies, we will not plagiarize anyone else's work and will give credit to outside sources whenever necessary.

Lastly, in keeping with the IEEE Code of Ethics #10, we will offer our aid to any of our fellow peers in their projects should they need it given we have the requisite knowledge and resources to do so [2].

References

[1] A. Mazzocchi, “Why Students Really Quit Their Musical Instrument (and How Parents Can Prevent It),” *The Music Parents' Guide*, 17-Feb-2015. [Online]. Available: http://www.musicparentsguide.com/2015/02/17/students-really-quit-musical-instrument-parents-can-preven. [Accessed: 08-Feb-2018].

[2] “IEEE Code of Ethics,” *IEEE*. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 08-Feb-2018].

[3] “How Colour Changing LEDs Work,” *Kitronik*. [Online]. Available: http://www.kitronik.co.uk/blog/how-colour-changing-leds-work. [Accessed: 08-Feb-2018].

[4] “Tempo,” *Wikipedia*. [Online]. Available: en.wikipedia.org/wiki/Tempo#Basic_tempo_markings. [Accessed: 08-Feb-2018].

[5] “Musical Keyboard,” *Wikipedia.* [Online]. Available: en.wikipedia.org/wiki/Musical_keyboard. [Accessed: 08-Feb-2018].