



Team 12: Auto-played Guitar

Proudly presented by Jiyu Hu, Qianlu Chen, and Peilin Rao



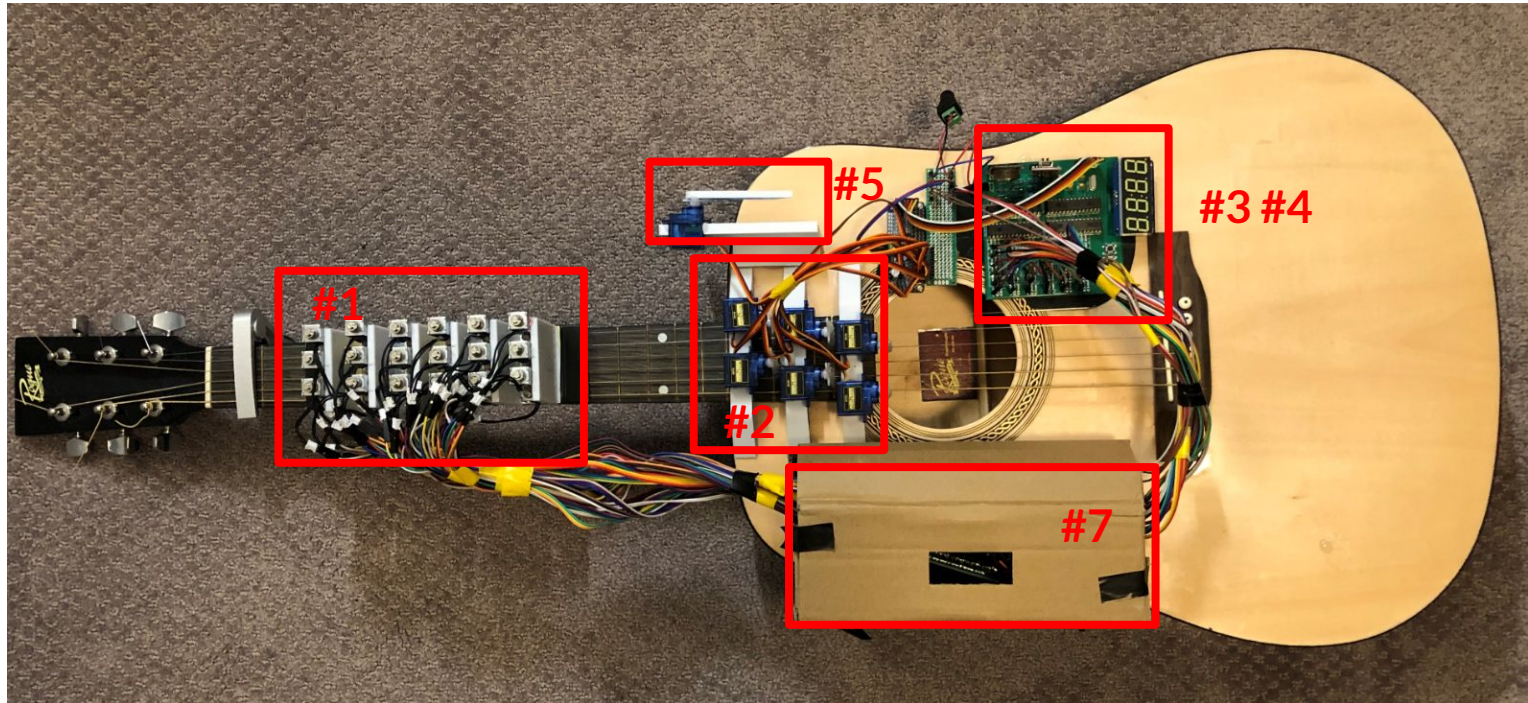
Introduction: Auto-played Guitar

- A device that can be installed on any type of guitars for automatic music playing.
- Features:
 - play almost any music on any guitars.
 - easy programming of music.
 - a metronome of adjustable tempo.
 - an alarm component.



Objective

- Bars: entertainment or creating atmosphere
- Guitar stores: the sound quality of the guitars
- Music creators: guitar music for remix
- Guitar learners: a demonstration of the musical piece
- Wake up alarm function
- etc



Systems Overview

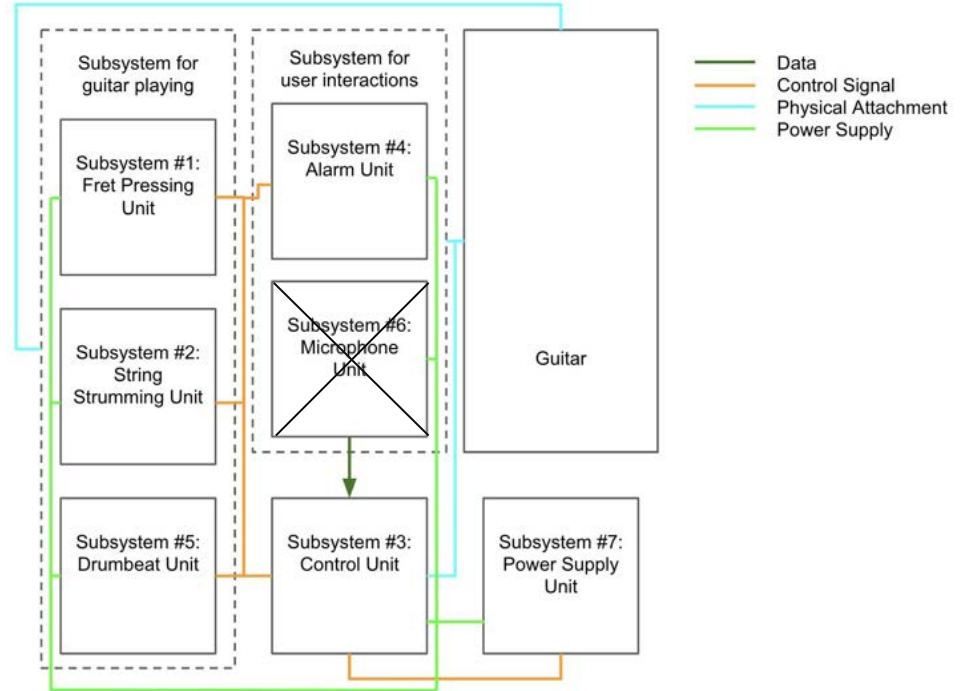
Units for Music Playing: #1 #2 #5

Units for User Interaction: #4

Control Unit: #3

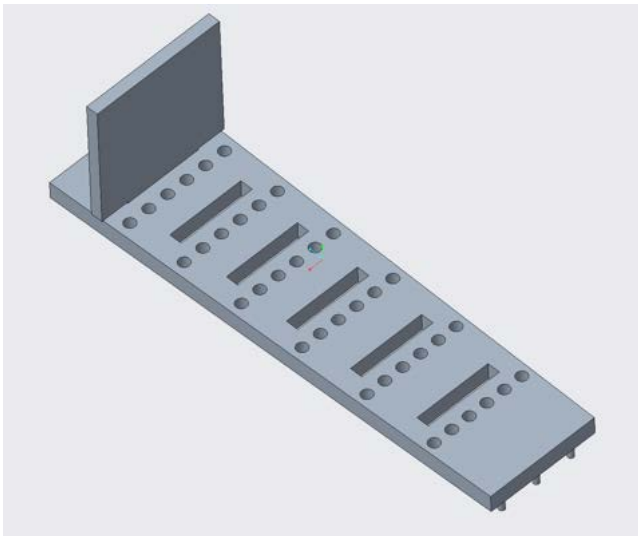
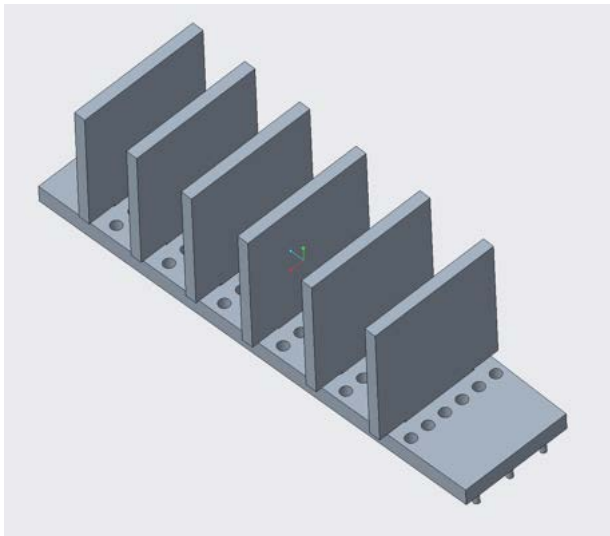
Power Supply Unit: #7

(#6 was labeled as optional in our design and we decided not to proceed with it)



Unit #1: Fret Pressing Unit -- Peilin Rao



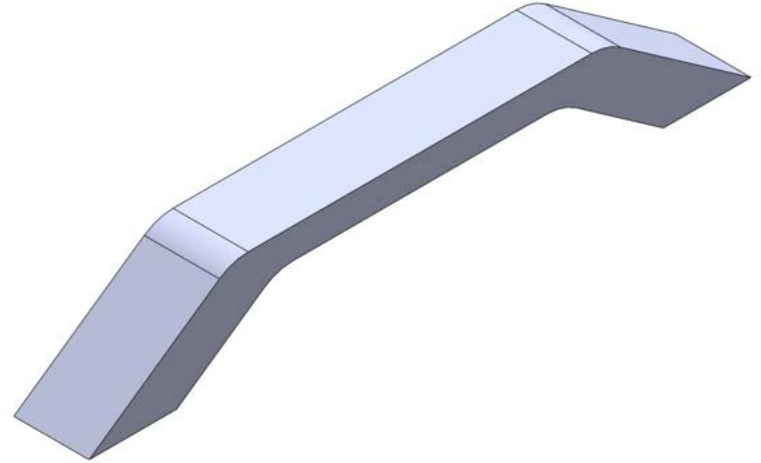




Requirements and Verifications for #1

- Requirements
 - Perform fast and accurate
 - At most six of actuators can be activated simultaneously
- Verifications
 - The notes are accurate verified by tuners within at most 3% error rate in Hz
 - Very fast reaction rate: software allows it to produce about 0.3 second per note without any flaws
 - Enough power supply for at most 8 actuators to work simultaneously

Unit #2 String Strumming Unit -- Jiyu Hu

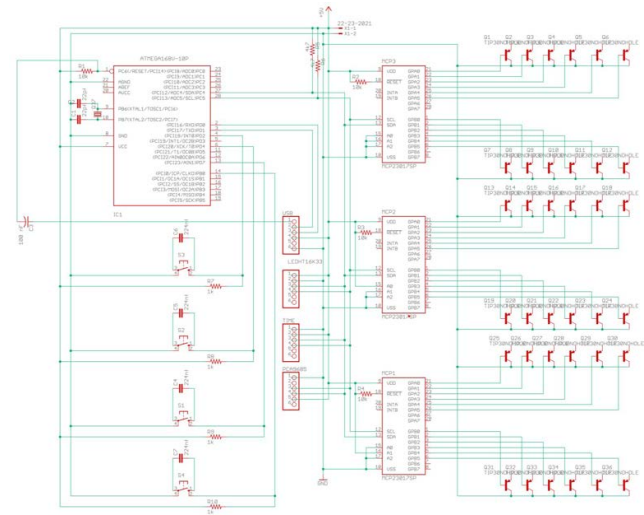
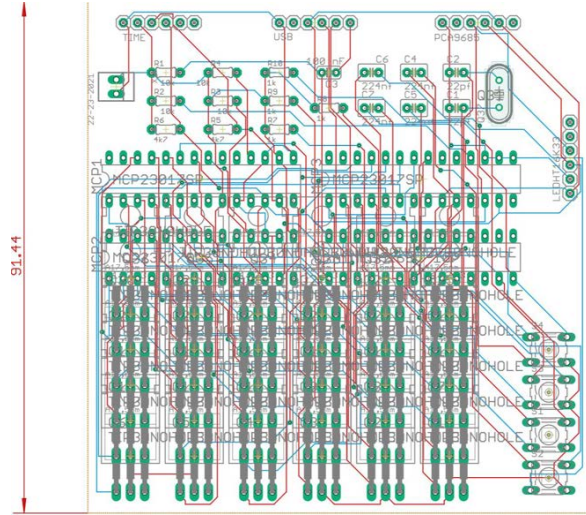
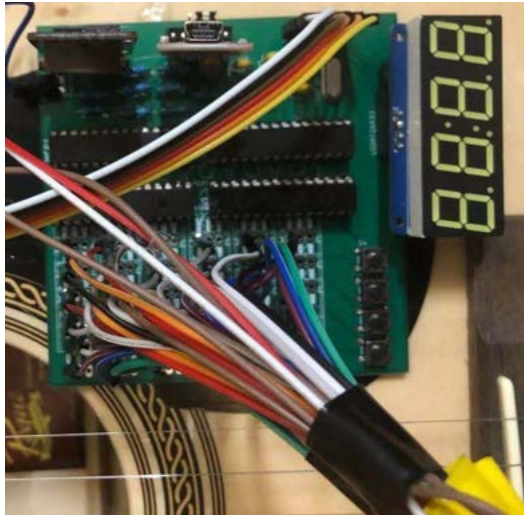




Requirements and Verifications for #2

- Requirements
 - The servos should be working and attached firmly above the strings on the guitar body
 - Each servo should respond fast and correctly
- Verifications
 - The servo tips just touch the string when placed at the correct position
 - The velocity of the servo is 60 degrees / 0.5 s

Unit #3 Control Unit -- Peilin Rao



Schematics Details

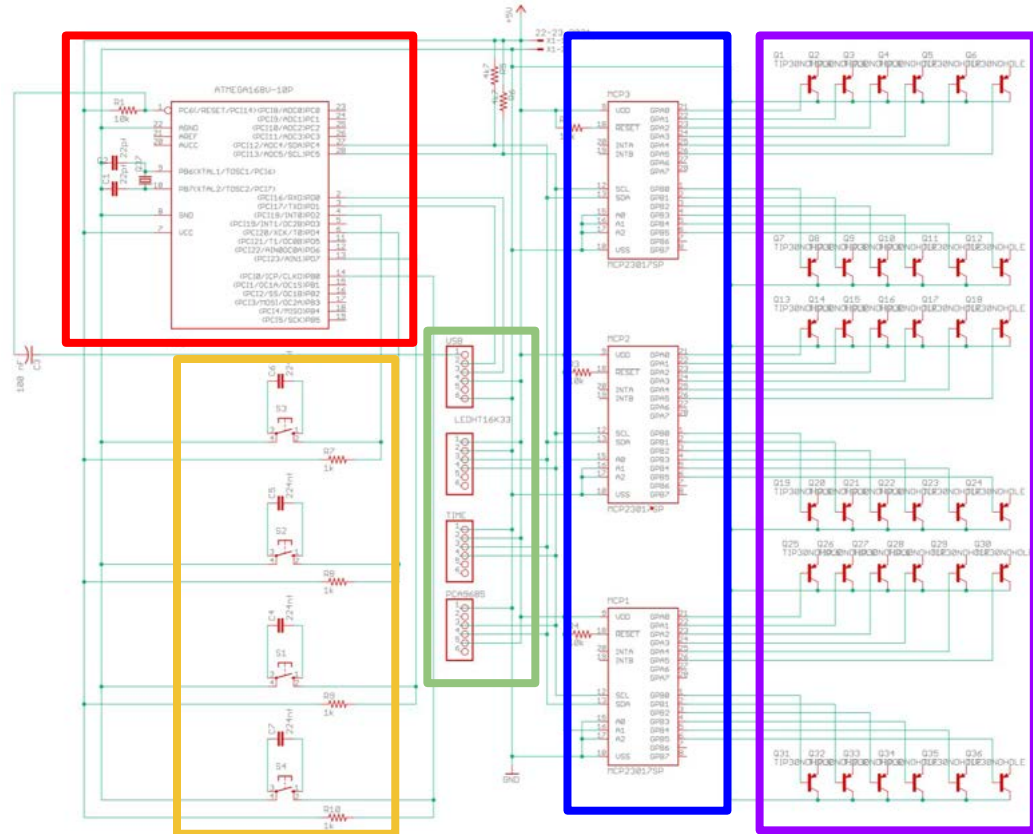
1 x Micro control chip: ATmega328

3 x I2C pin expanders: MCP23017

4 x buttons

LED port, USB port, Time port, Servo port

36 x actuator control pins



Board Details

1 x Micro control chip: ATmega328

3 x I2C pin expanders: MCP23017

4 x buttons

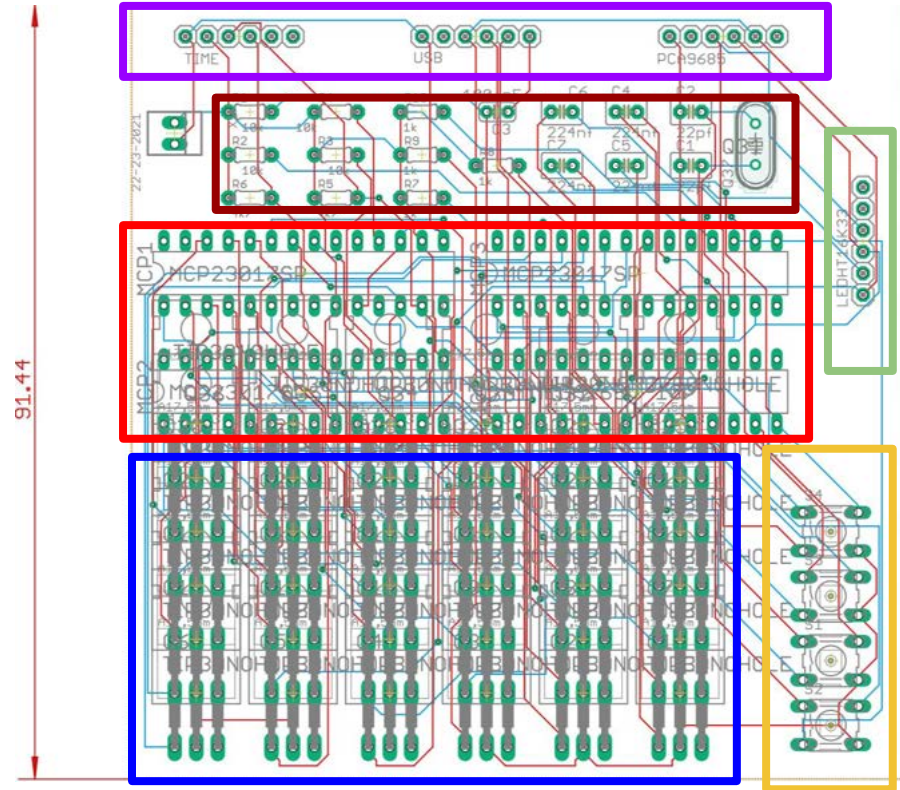
USB port, Time port, Servo port

LED port

36 x actuator control pins

Resistors, Capacitors and Oscillators

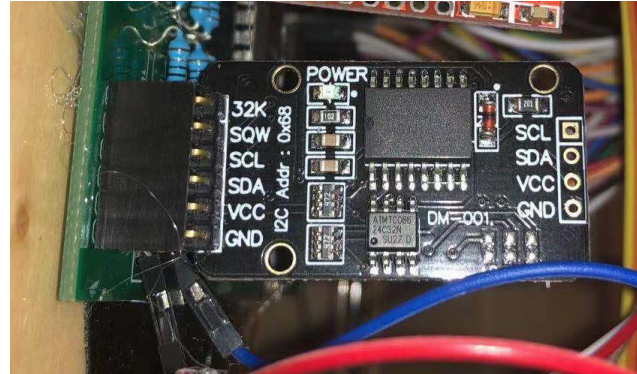
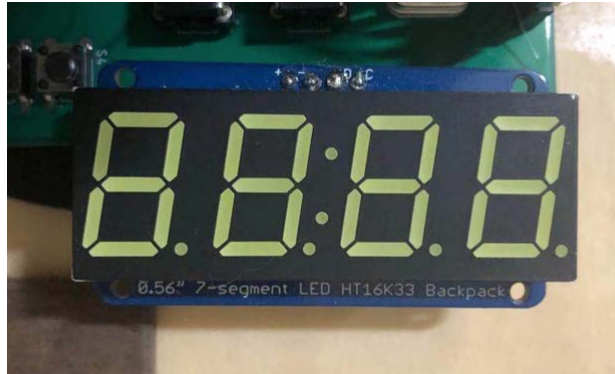
Design choice: clustering for convenience

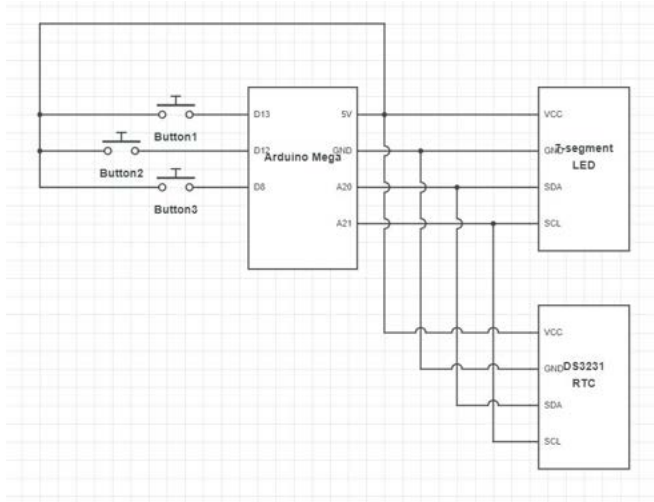


Unit #4 Alarm Unit -- Qianlu Chen

LED packet: Adafruit 7-segment

RTC packet: DS3231





Button 1: Mode switching

Button 2: Add

Button 3: Minus

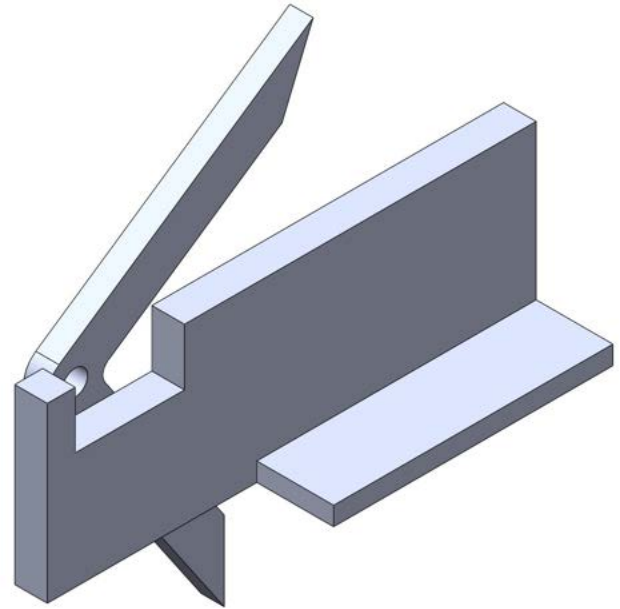
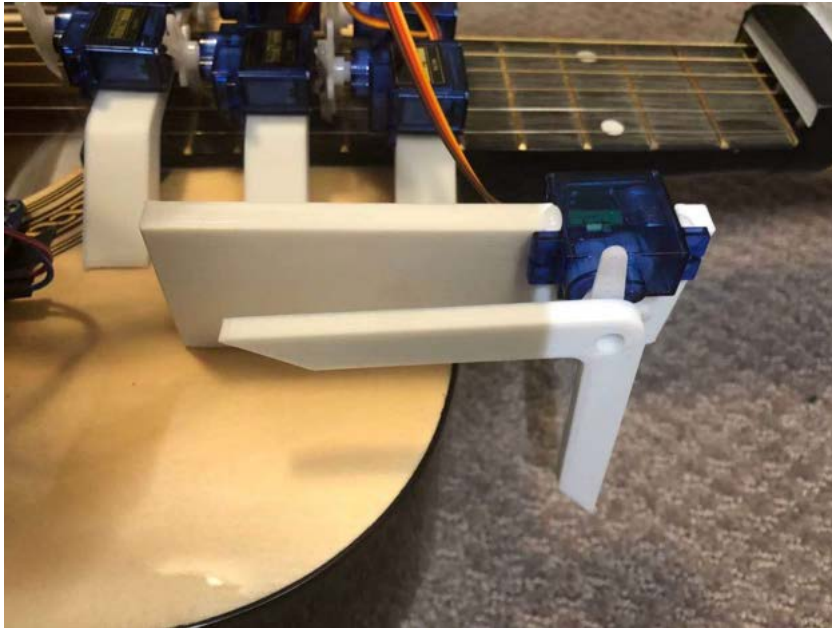
Button 4: Play Music



Requirements and Verifications for #4

- Requirements
 - Time should not differ from real time even after weeks
 - Pressing buttons should change mode
 - Alarm will go off as set
- Verifications
 - Use timestamp to verify the time difference
 - Buttons will change mode within 500ms
 - Alarm will go off at set time with 0s difference

Unit # 5 Drumbeat (Metronome) Unit -- Jiyu Hu



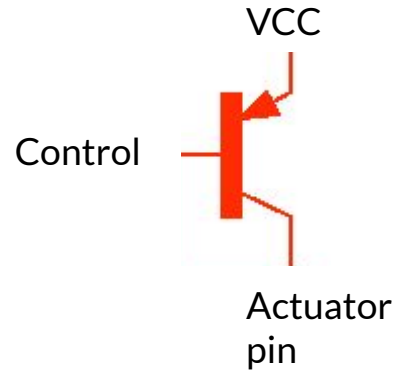


Requirements and Verifications for #5

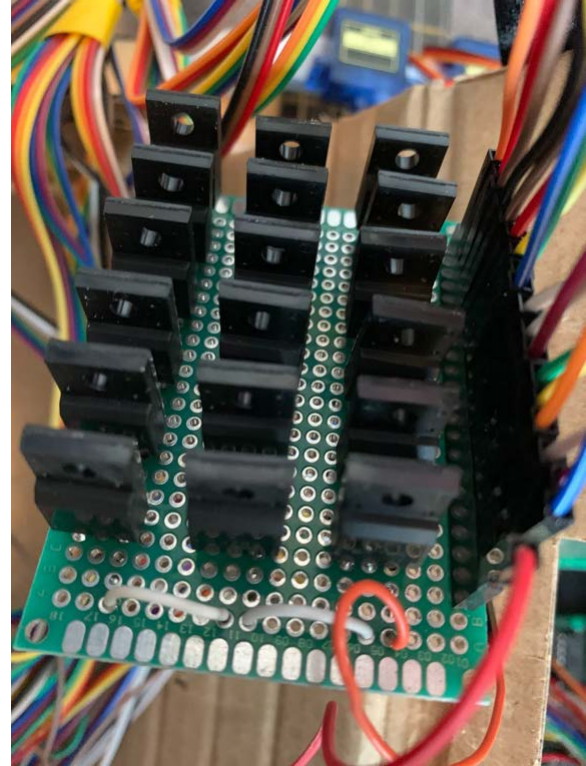
- Requirements
 - The tempo of the drum beats should be correct.
 - Should be able to provide two kinds of drumbeat (hit on the surface of the guitar & hit on the side of the guitar). The two kinds of drumbeat should be clear enough to tell from each other.
- Verifications
 - The tempo is verified with a stopwatch with 4% error rate.
 - Two kinds of drum beats create different sound.

Unit # 6 Power Unit -- Jiyu Hu

Actuators:	1.1A
Servo:	360mA
LED:	160mA
RTC:	160mA
Arduino:	10mA



Transistor matrix

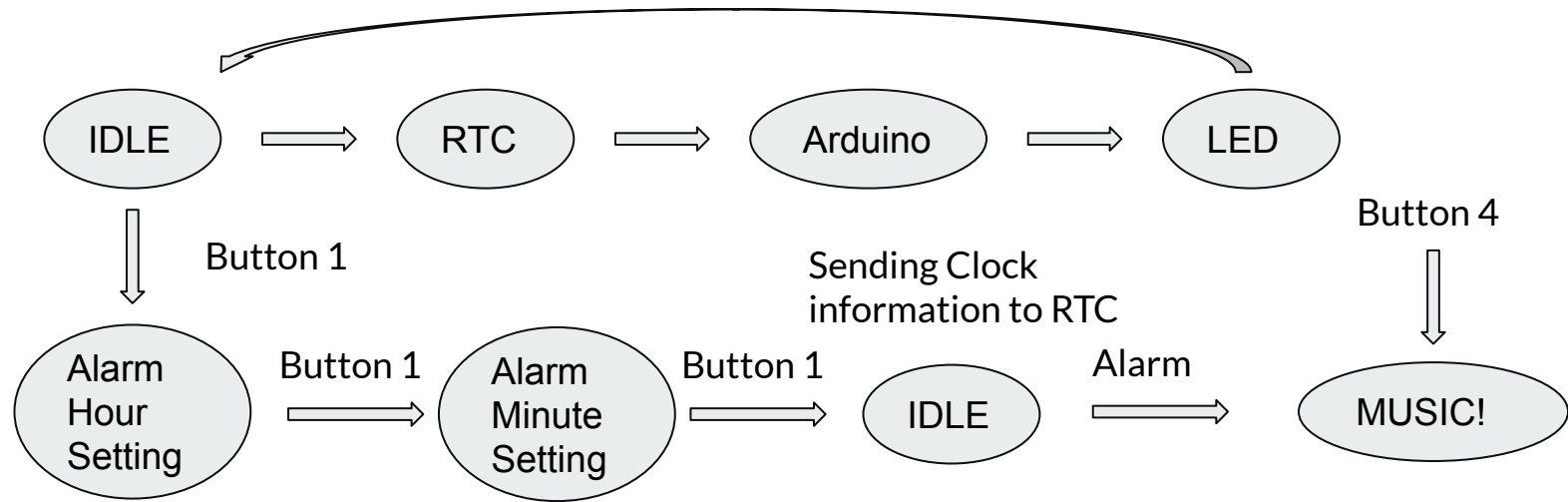




Requirements and Verifications for #6

- Requirements
 - This unit must be able to supply a voltage of $5V \pm 0.2V$ for a current load up to 10A.
 - This unit must be able to convert 120AV voltage to 5V DC.
- Verifications
 - All parts are put together and using multimeter we measure that the current and voltage passing by is within 5% of error.

Software Control Workflow -- Qianlu Chen





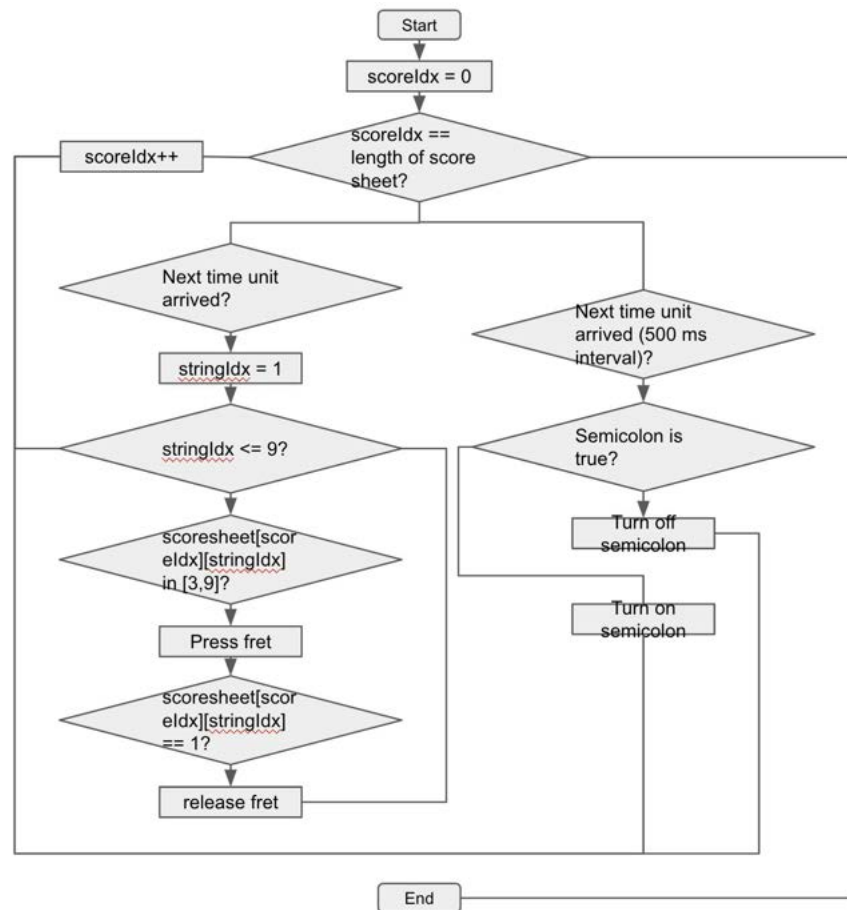
Score sheet data structure

- 0: do nothing
- 1: release actuator
- 2: Servo strike
- 3 - 8: Press corresponding fret on guitar

```
const uint8_t scoreSheet[][6]  
PROGMEM = {{3, 0, 0, 0, 0, 0},  
            {4, 0, 0, 0, 0, 0},  
            {5, 0, 0, 0, 0, 0},  
            {6, 0, 0, 0, 0, 0},  
            {7, 0, 0, 0, 0, 0},  
            {8, 0, 0, 0, 0, 0},  
            {1, 3, 0, 0, 0, 0},  
            {0, 4, 0, 0, 0, 0},  
            {0, 5, 0, 0, 0, 0},  
            {0, 6, 0, 0, 0, 0},  
            {0, 7, 0, 0, 0, 0},  
            {0, 8, 0, 0, 0, 0},  
            {0, 1, 3, 0, 0, 0}};
```

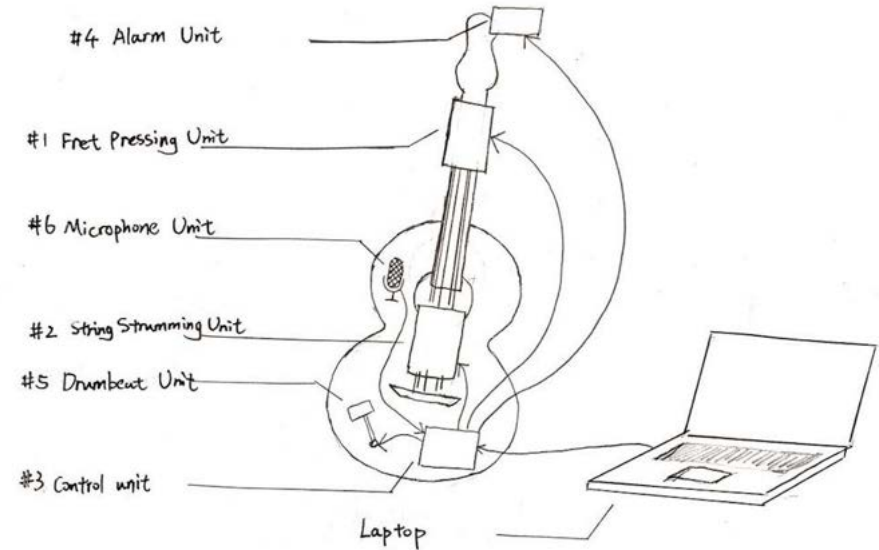
Multithread workflow

-- Jiyu Hu



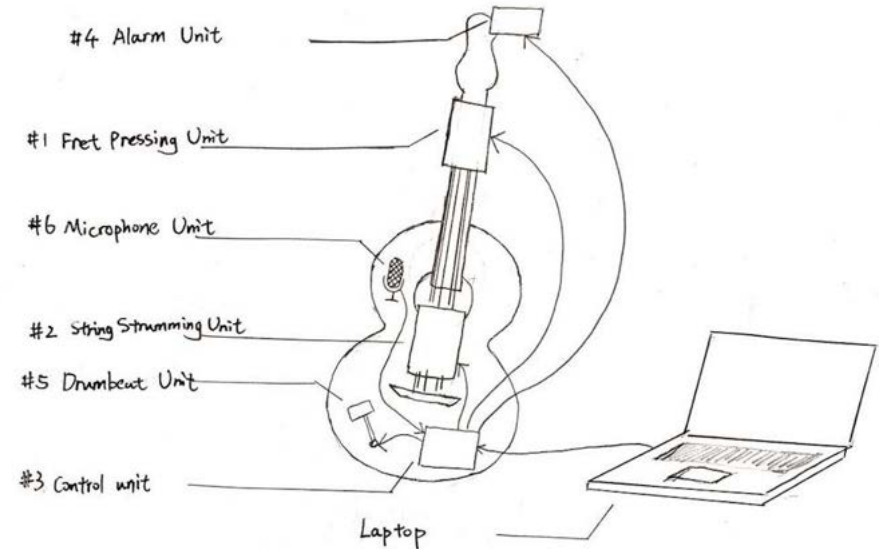
Potential Improvements

- Better packing and integration
- Better support for existing score sheet file types
- Add microphone



Conclusion

- All units pass unit tests.
- When combined, works correctly.
- Music produced is correct in tune and precise in tempo.
- Alarm works correctly.
- The drumbeat works correctly and indeed adds flavors to it.





**Thanks for
watching**