

# **LASER HARP**

*A Project Report*

*submitted by*

***Ajin Tom***

***Akshay Sanghat***



**TECHNITES - ENGINEER 2014**

**NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA**

**SURATHKAL, MANGALORE - 575025**

## ABSTRACT

The laser harp is light-based musical instrument. Like in the musical instrument harp, instead of strings there are lasers projected to corresponding targets whose path if interrupted gives the same response as plucking a string in the harp.

8 lasers (powered by adapter using 7805) and 8 LDR's are mounted and aligned on a mechanical structure as shown in the picture. The LDR's are connected in parallel with each having a resistor in series with it to get a sufficient drop when the laser is cut. This drop is detected by 8 corresponding digital pins of a microcontroller (here Arduino). Since the drop MIGHT not give a value low enough to be detected by the Arduino board, we introduce a comparator circuit so that the digital pins receive a healthy signal (high or low).

Whenever an interrupt occurs, the code sends a midi message\* via serial communication to a virtual bus on whose other end a virtual midi bus is configured. An external tool called hairless MIDI is used to establish communication between COM port and MIDI channel. This MIDI channel can be accessed from any music software like FL Studio, Cubase or Garage Band. Depending on which laser-LDR path was cut, corresponding pins send MIDI messages (in which the musical notes are assigned) to the MIDI channel accordingly. The instrument tone can be chose in the music software itself.

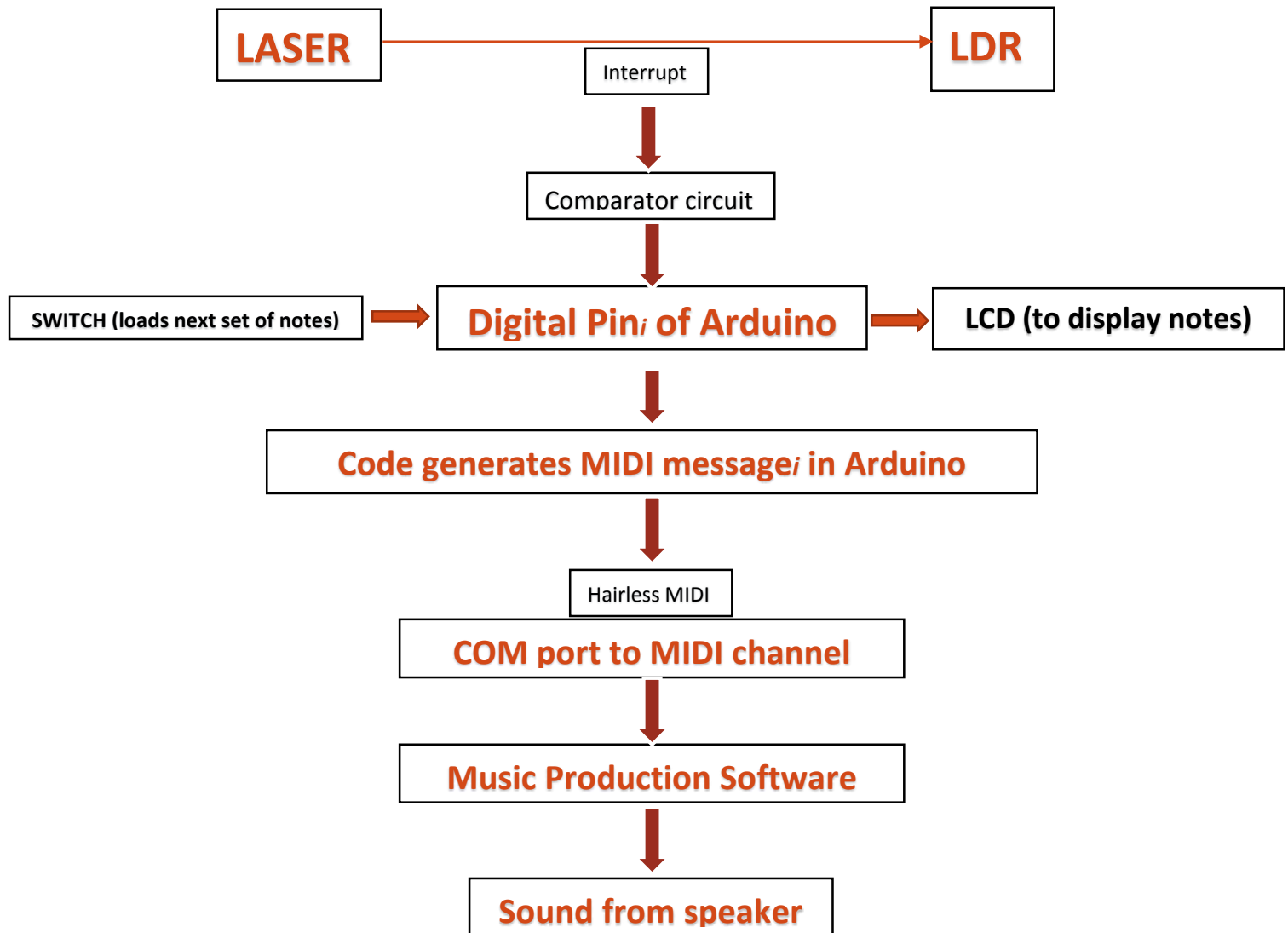
*\*a midi message consists of 3 parameters. **MIDImessageFUNCTION ( noteStatus , noteValue , velocity )***

- noteOn- note is pressed (0x90)
- noteOff- note is released (0x85)
- noteValue- hexadecimal value that maps to which note is being played[example C(0x3C) or D(0x3E)]
- velocity- intensity with which the note was pressed

**NOTE: This project hasn't used the Arduino MIDI library.**

---

## BLOCK DIAGRAM



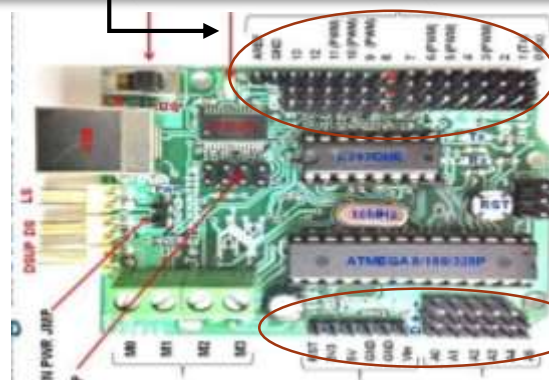
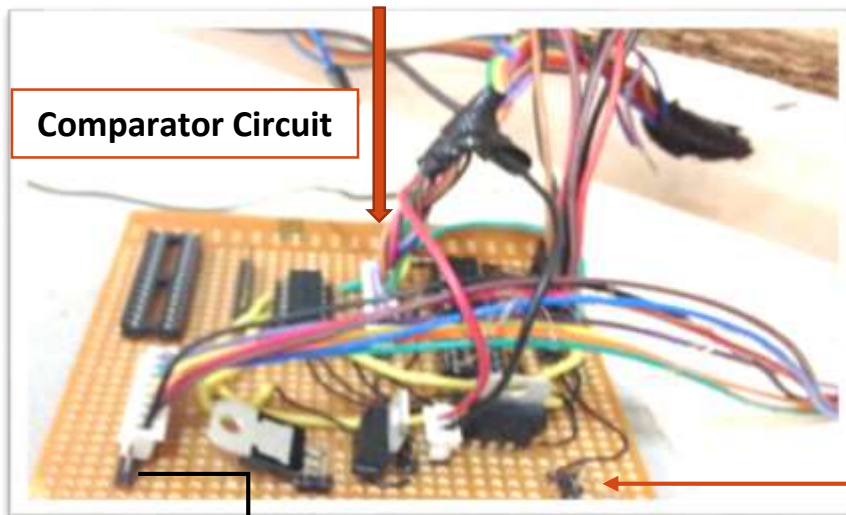
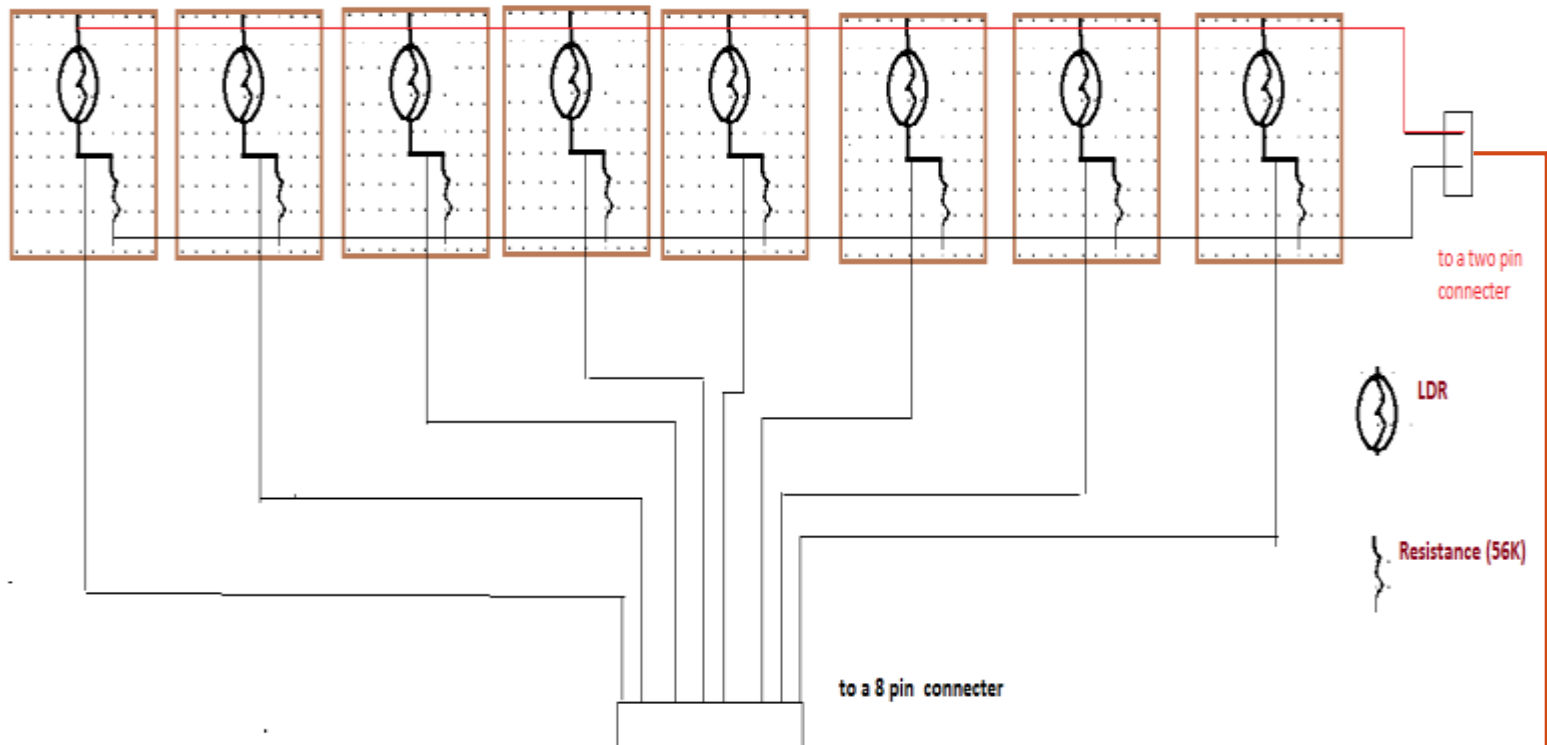
## COMPONENTS/SOFTWARES NEEDED

Lasers	LDR's	Resistors and pot
Microcontroller(Arduino)	7805's and 7833's	LM324
Mechanical Structure	L-clamps to mount lasers	Speakers
Arduino	Hairless-midi tool	FL Studio / Garage band

Link to Arduino code :

<https://drive.google.com/file/d/0B54QOBbu7CoXemNmbmR5UzNYVGs/view?usp=sharing>

## HARDWARE SETUP



LCD



## PROBLEMS FACED

- Using big LDR's thinking it would be easy to align the lasers was a mistake. Big LDR's never gave desired outputs since it gave a delay (it takes some time for the full drop to occur).
- Since the mechanical structure wasn't stable, the alignment of the lasers with LDR's changed often causing interrupts in all the LDR's thus leading to undesired outputs.
- The VCC and ground got interchanged in a few LDR's while making the parallel LDR circuit. This led to a big problem since the code detects HIGH to LOW and interchanging caused some LDR's to work in HIGH to LOW and others in LOW to HIGH.
- Ambient light gave undesired output. Basically, the room should be void of any ambient light.

## FUTURE WORK

- Mechanical structure can be more stabilized.
- The LDR's can be placed inside small boxes so that ambient light won't affect the harp. Making holes in these boxes can make alignment also easy.
- Good spacing should be given between the LDR's so that it becomes easier to play the harp.
- More coding can be done to allow change of instrument and other cool stuff (like looping).

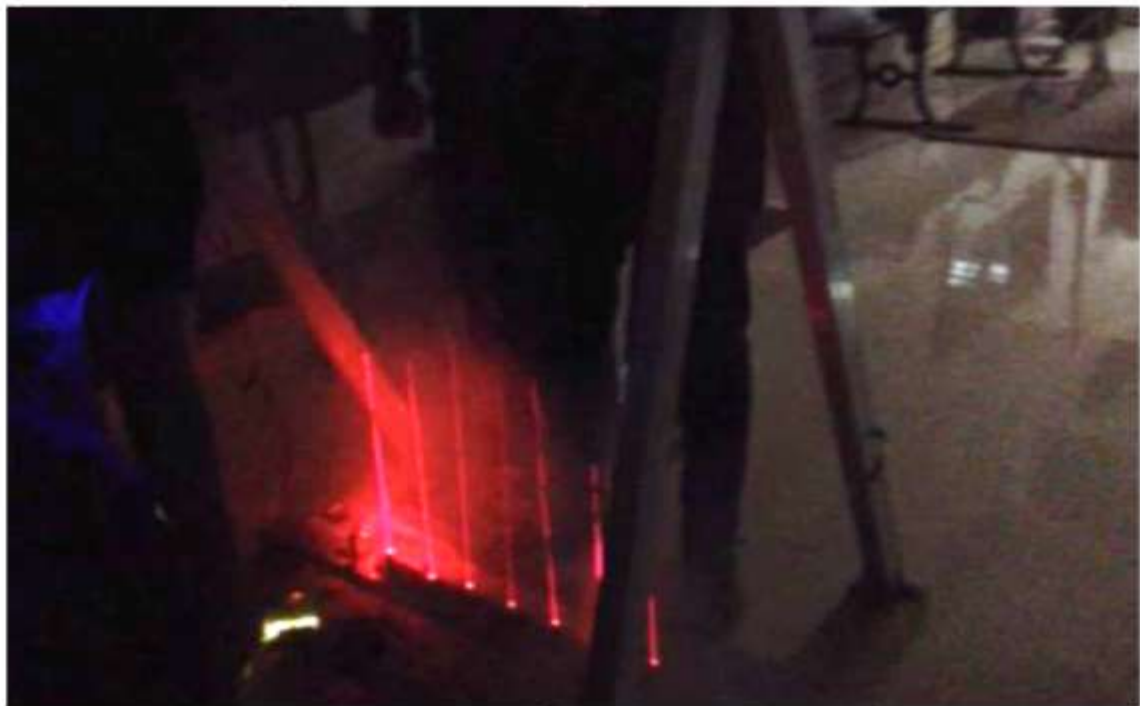
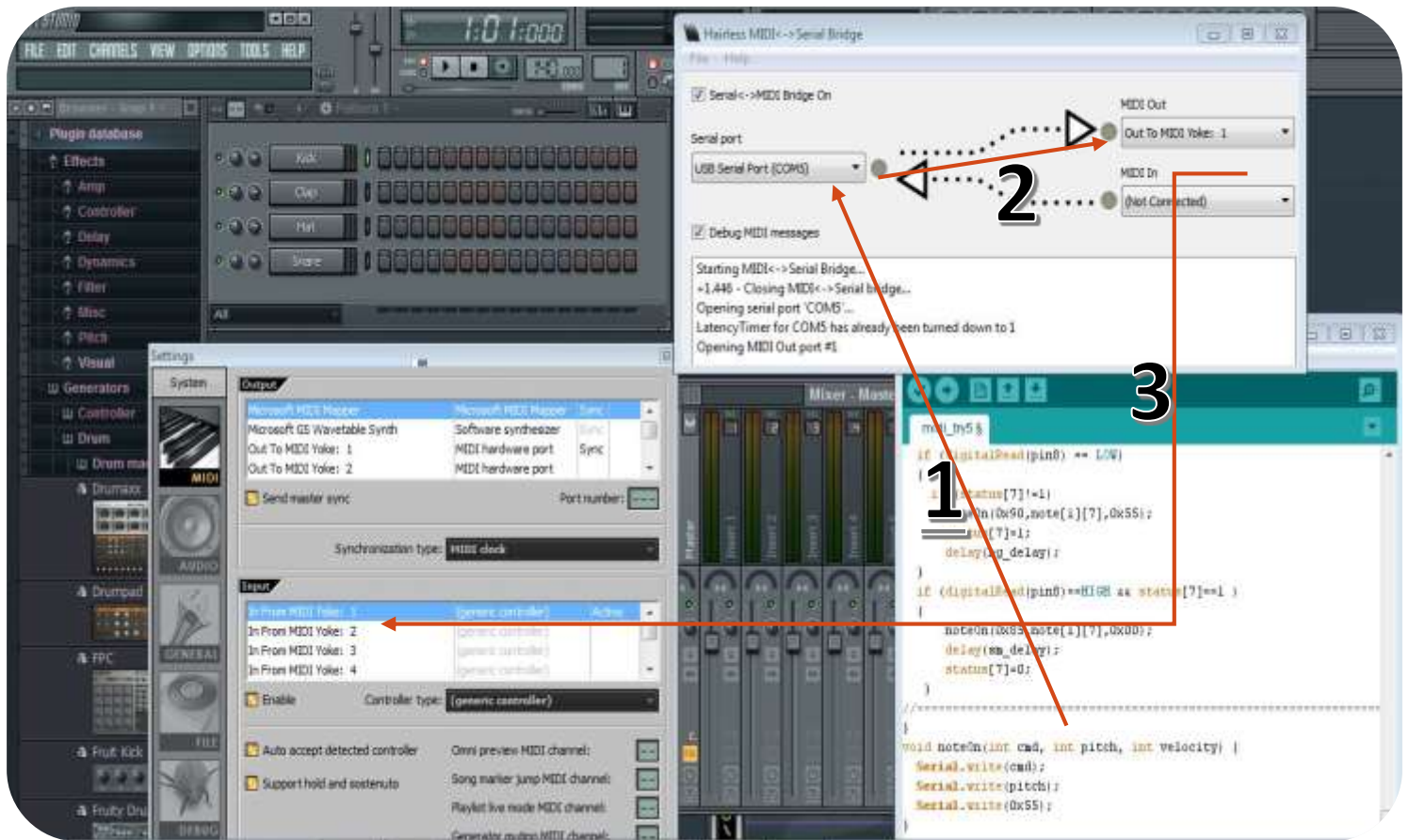
## CONCLUSION

This project will turn out to be successful if everything including the mechanical structure and circuit is done systematically and with perfection. The coding can be done using the built-in MIDI library in Arduino (in this project, we haven't used), but if one wishes to know exactly how MIDI works, refer to the code in this documentation.

---



## PICTURES & SCREENSHOT





**NITK Surathkal**



THINK. CREATE.  
**ENGINEER**  
2014