Multi-entertainment Tic Tac Toe Game

ECE 445 Design Document

Jiacheng Zhu
Minkang Yang
Nuochen Lyu
Group 30

TA: Xinrui Zhu
10/19/2017

Catalog 1

1 Introduction

1.1 Objective

Game is always a significant activity throughout the history of human being. Although it seems like it is sometimes purely for entertainment, games offer elements that are crucial for human development, both physically and mentally. The use of games includes self-improvement, encouraging positive lifestyle changes, and increasing motivation to complete work objectives.

An arcade game or coin-op is a coin-operated entertainment machine typically installed in public businesses such as restaurants, bars, and amusement arcades.This new style of game is emerging after the early 1970s because of the rapid development electronic hardware components. Graphic display, sensors and peripheral devices make this possible.

Therefore, we want to make a tic tac toe game that is fun to play together. For instance, when customers in the restaurant are waiting for dining, rather than checking their phones, a small game box can help customers get in the mood. Also, such a device can help people get to know each other without too much awkwardness. Moreover, it is portable and low-cost. For many popular restaurant, the waiting-to-be-served time can be very long. The customer can grow boredness during this time, which would affect their appetite, so it would be a very good idea for the boss of the restaurant to provide a few game devices to their customers to play. It would make customer willing to wait for a long-time other than leave for another restaurant. Another situation with long boring waiting time would be when people are waiting for the bus; the government could implant several game devices on the bus station for people to play, which could give them a good time while waiting.

1.2 Background

The history of the arcade game can be traced back to 1950s which are made by the electro-mechanical device. Different from video games nowadays, arcade games owns more hardware components. Make an arcade device is a very great way to get an idea how common hardware devices work and cooperate with each other. Moreover, the realization of arcade game also needs software codes to control and schedule the hardware. Usually, the software part is in much lower level comparing to modern video game running on a mature gaming engining. Build an arcade game device could be an appropriate project for ECE 445.

We are three Computer Engineering Students, and we hope through this project, we could learn how to make a portable game device. We would like to make the tic tac toe game box a high completeness, good-looking, and captivating product. After this class, we might donate some sample product to NGO, MTD bus, and primary school.

1.3 High-Level Requirements

1. Human could play tic tac toe with another human.
2. Human could play tic tac toe with AI.
3. LED blocks must support animation patterns “X”/“O”

2 Design

2.1 block diagram

The project is majorly composed of four sections: input, display, control, and power.The power section will deliver the corresponding voltage to every electronic element.
Display section ‘s light will be determined by the control section, which will continuously receive the input from the input section that receiving pressure input.

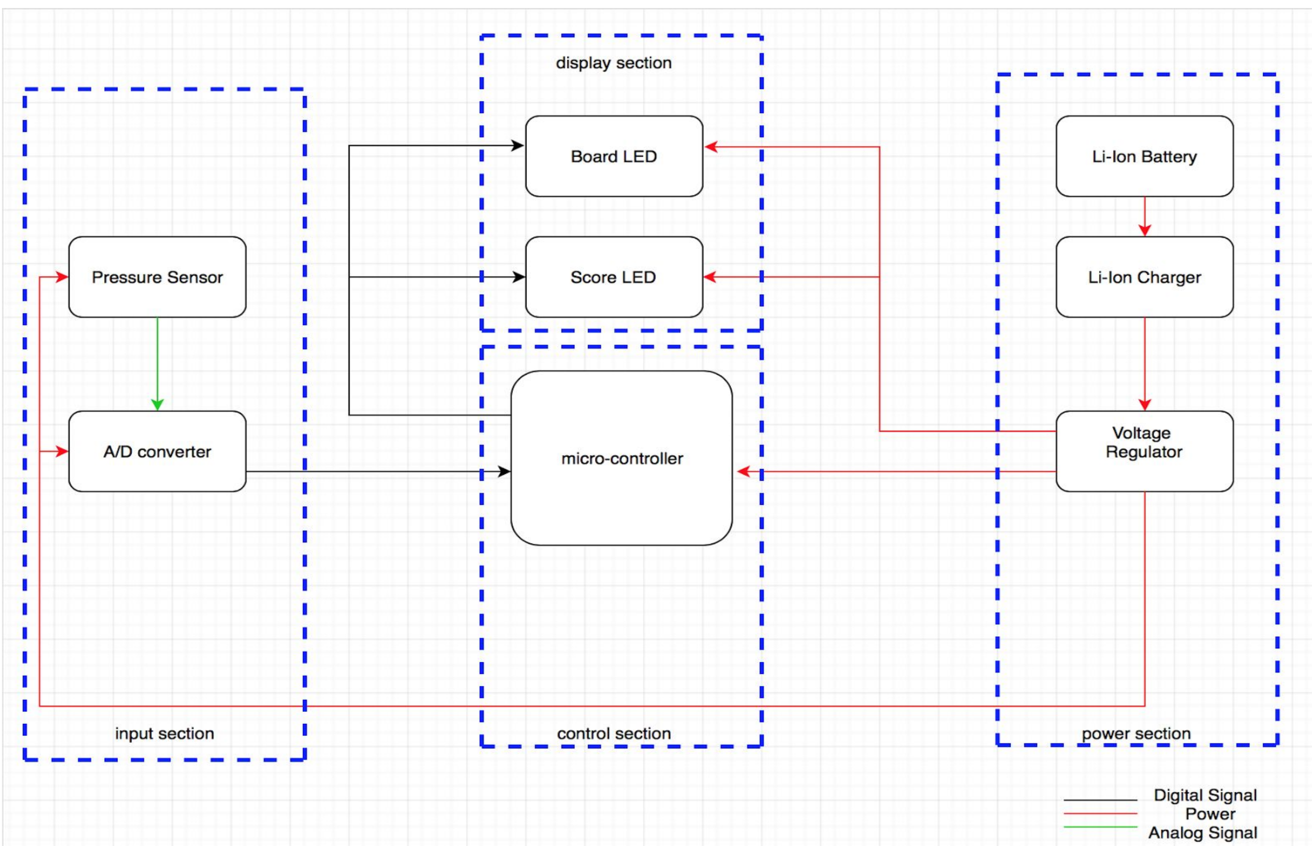


Fig. 1. Block Diagram

2.2 physical diagram

The final project would be similar to the below picture. The 3*3 LED tic-tac-toe board, 1*4 LED display on the top and two buttons on the right. The cube box would be made from wood or plastic and the hardware design is concealed inside. Underneath the 3*3 LED board lies 9 pressure sensors. The power supply is a battery that is plugged on the bottom of the cube box.

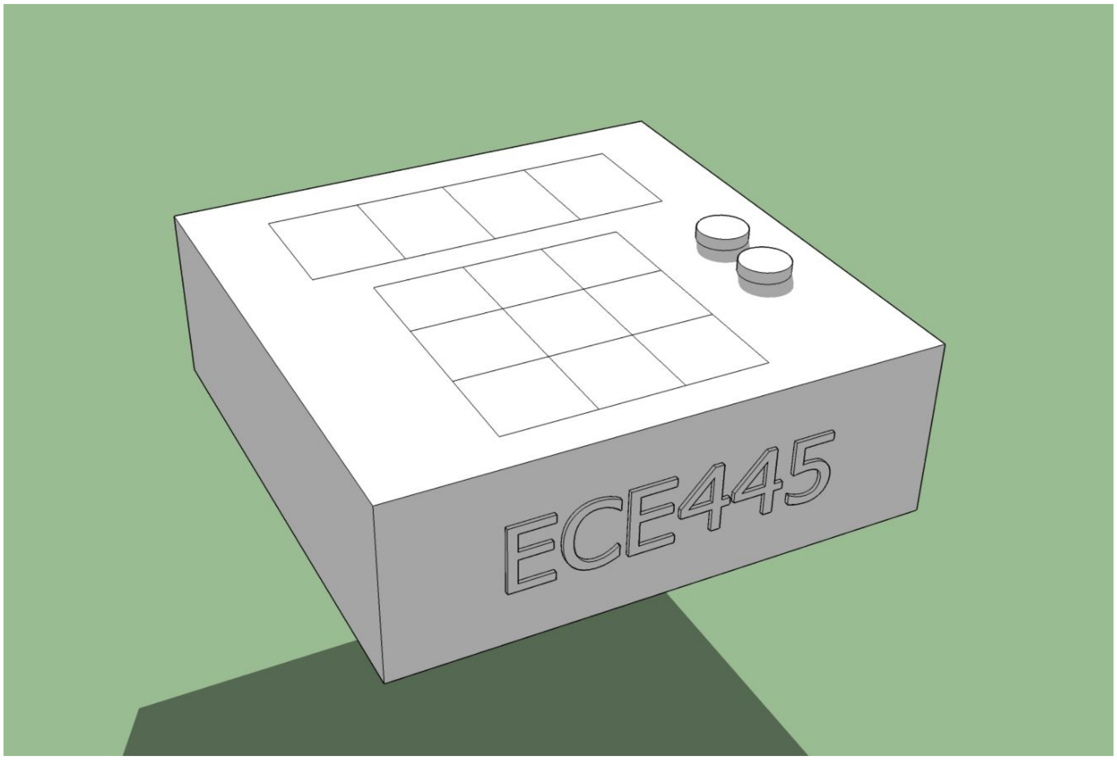


Fig. 2. Physical design sketch

2.3 Overall Functionality

2.3.1 Power Supply:

- A power supply is required to keep the whole system up continually. Power would be provided by a stable 5V fixed power source from our ECE 445 lab.

2.3.2 A/D converter module

Input: 9 analog signal from sensors

Output: 4 bit unsigned long

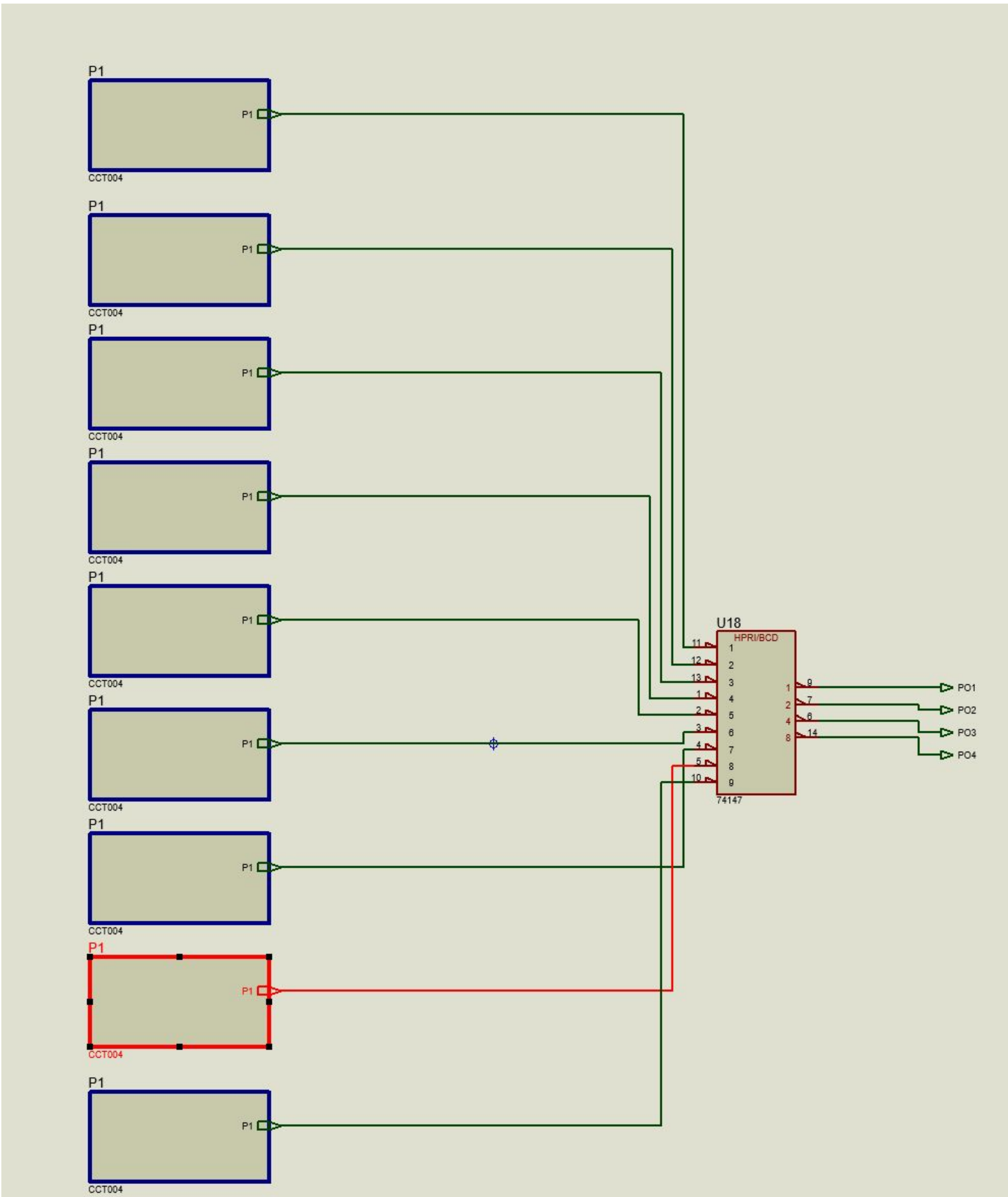
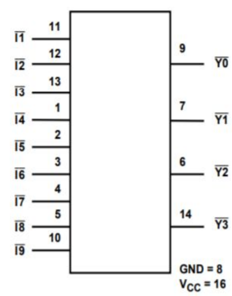


Fig. 3. A/D converter Schematics

- As it is descripted on the above Fig 3 description. They are essentially 9 pressure sensor which are on the left of the Fig 3. A analog to digital circuit is needed to convert the analog signal from pressure sensor to microcontroller. The 9 pressure sensor will give digital signal output. The digital signal output voltage would be depend on the design of our pressor sensor circuit. We would make the output voltage to be exactly zero or five voltage.
- The 9 output from 9 pressure sensor would deliver to the 10 to 4 Line Priority Encoder (CD54HC147). The encoder would receive the digital signal from 9 sensors, so one input pin would be idle. The bit encoding table is shown in the below figure. Outputs would have 4 bits and the output would be offered to the microprocessor. The 4 bit arrangement should be able to help microprocessor understand which pressure sensor is been pressed.

Functional Diagram



TRUTH TABLE

INPUTS									OUTPUTS			
$\overline{I1}$	$\overline{I2}$	$\overline{I3}$	$\overline{I4}$	$\overline{I5}$	$\overline{I6}$	$\overline{I7}$	$\overline{I8}$	$\overline{I9}$	$\overline{Y3}$	$\overline{Y2}$	$\overline{Y1}$	$\overline{Y0}$
H	H	H	H	H	H	H	H	H	H	H	H	H
X	X	X	X	X	X	X	X	L	L	H	H	L
X	X	X	X	X	X	X	L	H	L	H	H	H
X	X	X	X	X	X	L	H	H	H	L	L	L
X	X	X	X	X	L	H	H	H	H	L	L	H
X	X	X	X	L	H	H	H	H	H	L	H	L
X	X	X	L	H	H	H	H	H	H	L	H	H
X	X	L	H	H	H	H	H	H	H	H	L	L
X	L	H	H	H	H	H	H	H	H	H	L	H
L	H	H	H	H	H	H	H	H	H	H	H	L

H = High Logic Level, L = Low Logic Level, X = Don't Care

Fig. 4. The input/output table for encoder[1]

2.3.3 Pressure Sensor

Input: pressure force by human finger

Output: the analog signal/current

- Force Sensing Resistors (FSR), is a polymer thick film (PTF) device which exhibits a decrease in resistance with an increase in the force applied to the active surface. Shown in Fig 5 on the left. The standstill default resistance for FSR is 1M Ω. The force sensing resistor we use is made by Pololu with Part Number 1696, UNSPSC Code 32121600
- The pressure sensor circuit is shown in Fig 5. A 5V DC power would be provided to all Vcc in fig.5. The FSR(RT1) is connected with a 1.2M resistor in seriy. With such 1.2M resistor the voltage change across RT1(1M Ω) would be able to differentiated. The voltage change should be varied between 0.1 to 4.9 V. A op amp is connected to convert the changing analog signal from RT1 to digital signal. The op amp we use is made by Texas Instrument, with model number LM393. The voltage across RT1 will feed into the negative input of the op amp(pin 2). We use R14(2KΩ) and R15(3KΩ) to construct a voltage divider circuit and feed 3 V to the op amp's positive input pin.
- With the circuit we described above, if there is no pressure on the force densing resistor, the voltage feed to pin2 would be bigger than 3V, so the op amp will output 0V to our encoder, and 0V will be taken as digit 0. If there is enough pressure(can be easily done with one finger) on the force sensing resistor, the voltage feed to pin2 will be smaller than 3V, so op amp will output 5V(the Vcc to op amp is 5V) to our encoder, and 5V will be taken as digit 1.

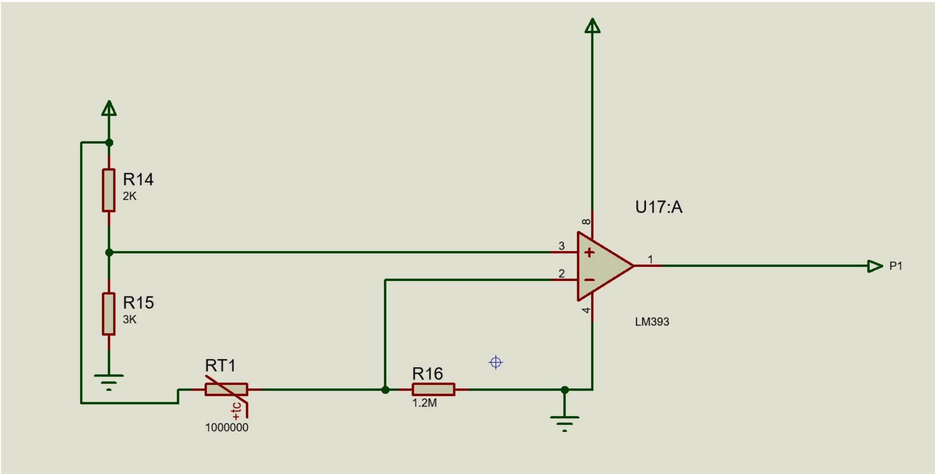


Fig. 5. The pressure sensor module

2.3.4 Microcontroller

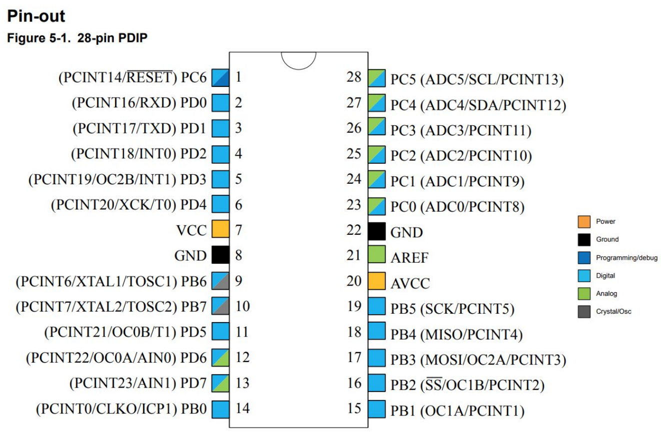


Fig. 6. Pins for microprocessor[2]

Input: 4-bit input digital signal, 5.5v power
Output: 2 bit CLK + 2 bit data translate + 6 bit location dispatch

- The microcontroller will be the decision maker for the 9*8*8 LEDs on the board since the 8*8 LED need clock and input, the microcontroller will generate a 4 -bit chip select signal, an overall input for all the LED light, and a CLK to ensure the data-transfer. Also, it will receive the input from the 4-bit mux, which is from the pressure sensor. Following on the left are the control flow on the controller

- Here, we design to use the MEGA-328p chip with a burner, with 1 bit on clock generation, 8 bit of location dispatcher, and 2 bit of datain , and 4 bit of digital input.

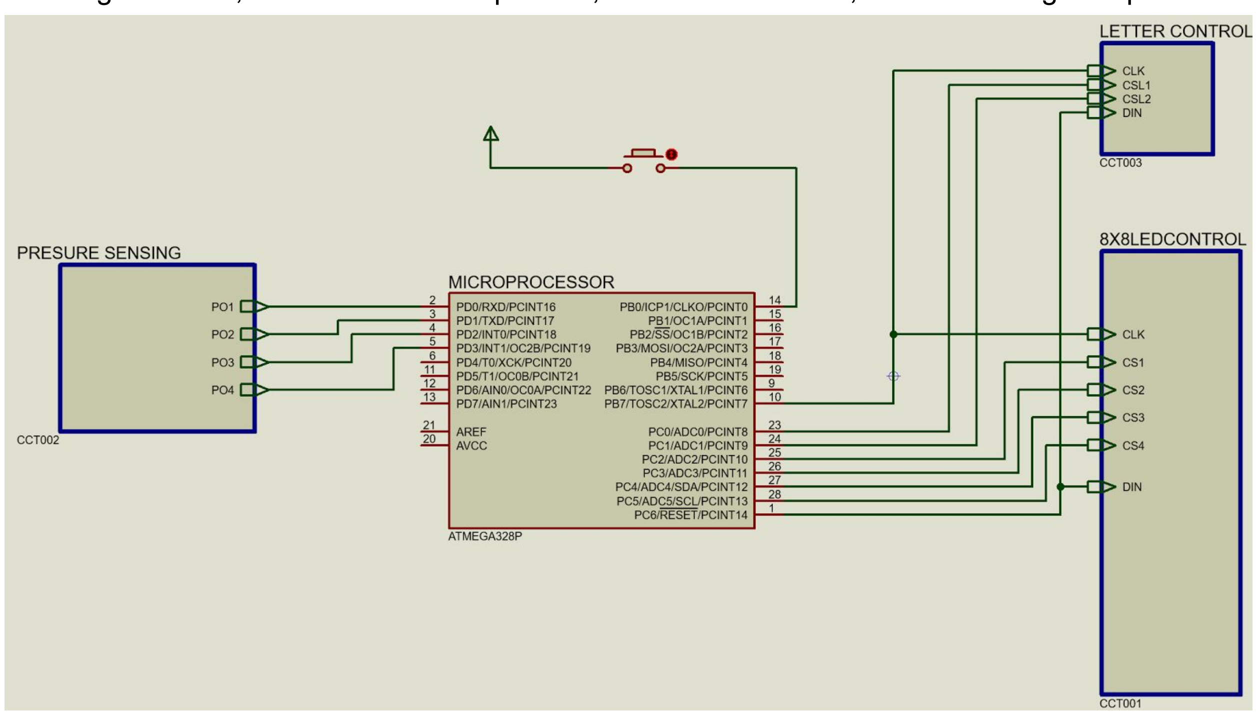


Fig. 6. The Microprocessor schema

2.3.5 Location Dispatcher

Input: 4 bit(No.1), 2 bit(No.2)

Output: 9bit (No.1).4 bit(No.2)

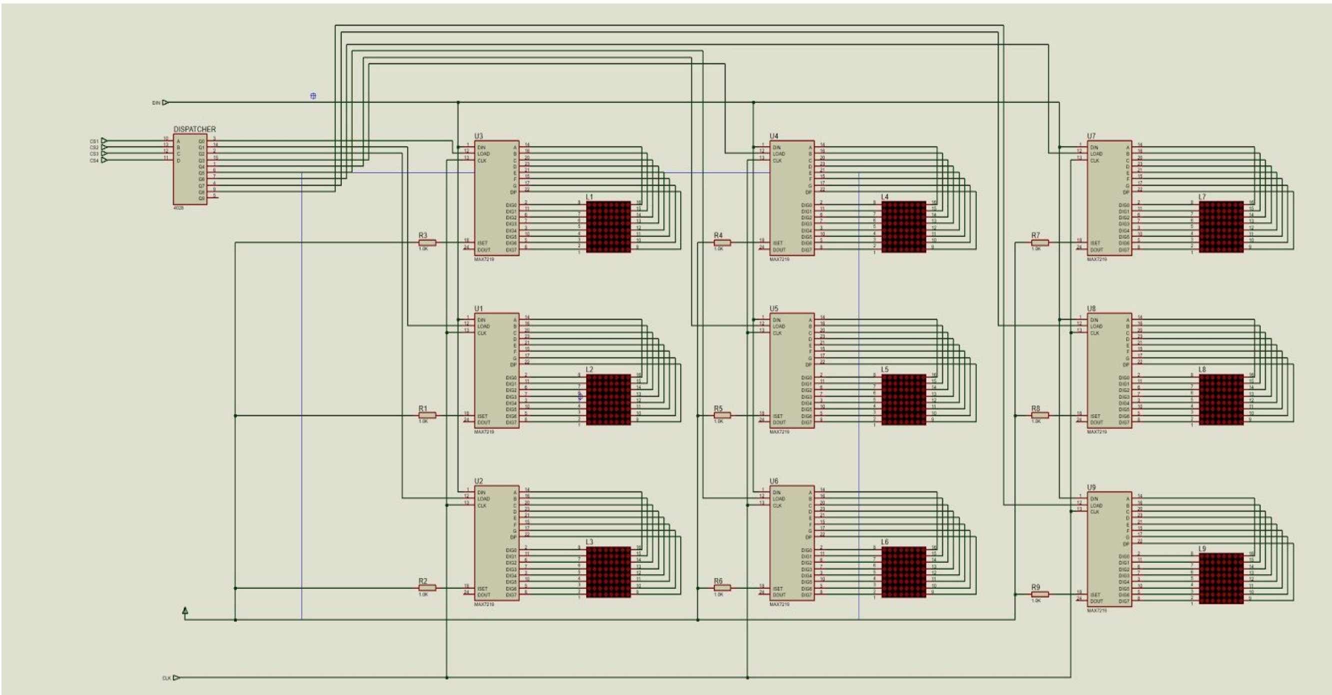


Fig. 7. The schema for LED module

- There are two separate dispatchers both are made of decoder, both will account for the chip select signal of the 8*8 LED, the first is 4 bit - 9-bit decoder, and will decide which board should receive the signal and display the tic tac toe. However in some conditions, we need to light up several sections of the board, for instance during the start the end games' graphics. In this case, since the led will remain its current lighting condition if the Vcc is not cut off, we will light each section one by one using the controller command. For the second location dispatcher, there is 2-bit input, and the decoder will select the according to LED screen to display text

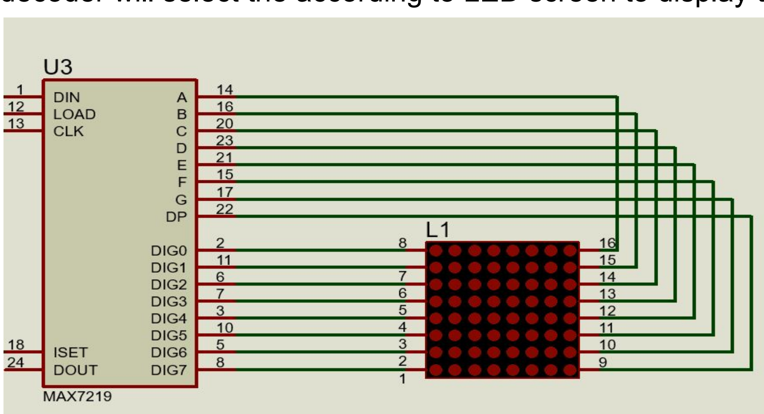


Fig 8 Detail schema for LED

2.3.6 Letter Display Screen

Input : Vcc, GND, CLK, Din, CS

Output: LED light

We plan to use four 8*8 screens ask the letter display on the top; also it receives chip-select input from location dispatcher, and 1-bit clock and data input from the controller

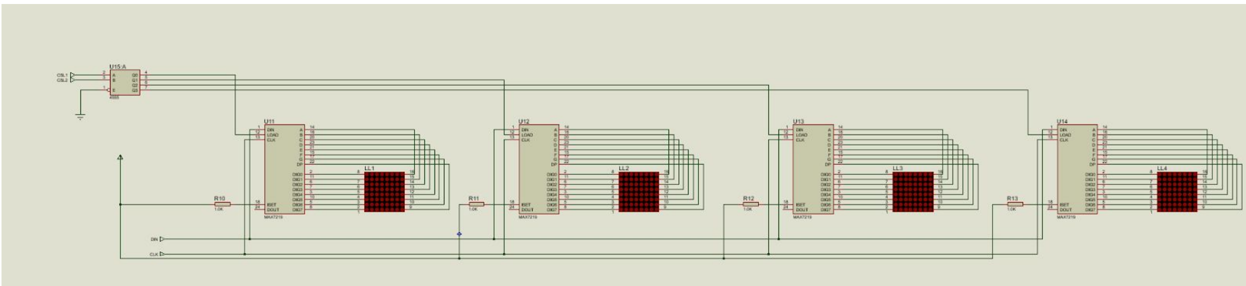


Fig. 9. The Schema for LED score display

2.3.7 Pressure Sensor Location Encoder

Input : 9 bit

Output :4 bit

This is a 9 -4-bit MUX, which will receive the input from pressure sensor and send the encoded signal to controller

2.3.8 9*8*8 LED Module

Input: 1 bit serialized

Output: 1 bit serialized

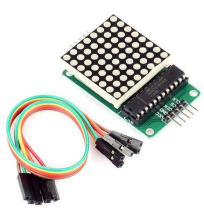


Fig. 10 LED

The 8*8 LED matrix we choose is SainSmart MAX7219 Red LED Dot Matrix Display Module, which is compatible with Arduino. The material is FR4 + electronic components; a single module can drive one 8*8 common cathode dot matrix, the working voltage of the MAX7219 chip is 5v.

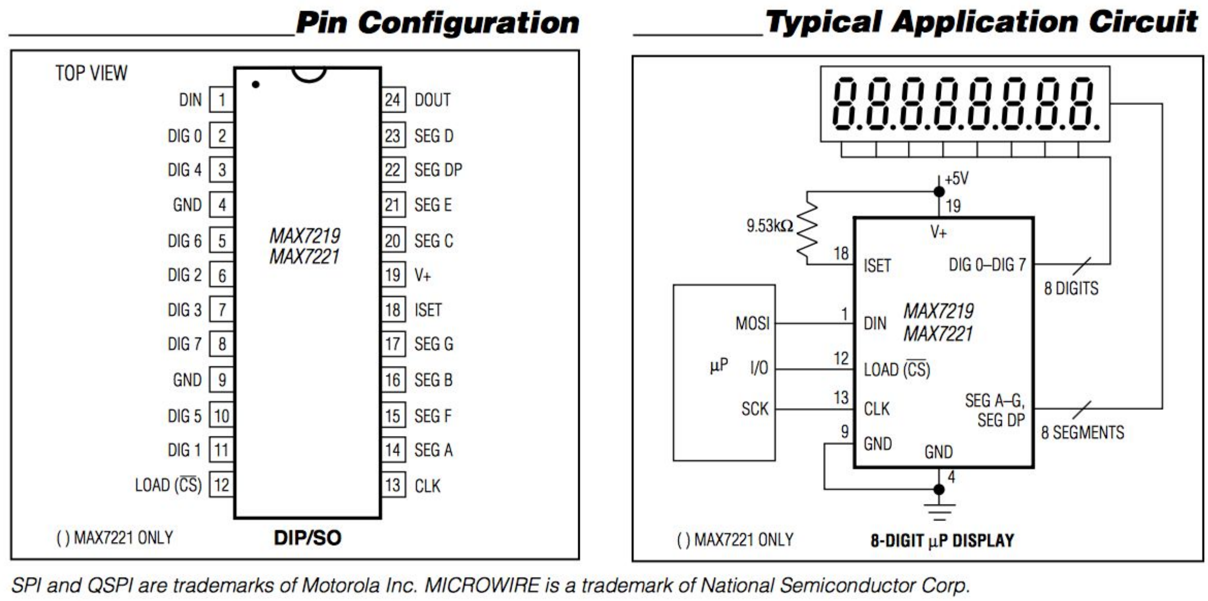


Fig. 11. The chip datasheet for LED chip[3]

The control chip is shown above and as long as the CS(chip select) is low, and the V+ is connected to the power, and CLK is connected to our Arduino CLK, our chip is ready to take input, DIN is for Serial-Data Input, Data is loaded into the internal 16-bit shift register on CLK's rising edge.

2.3.9 switch

This is a 6mm square SPST push button. We are going to use two of such device to act as two buttons. Model COM-12992.

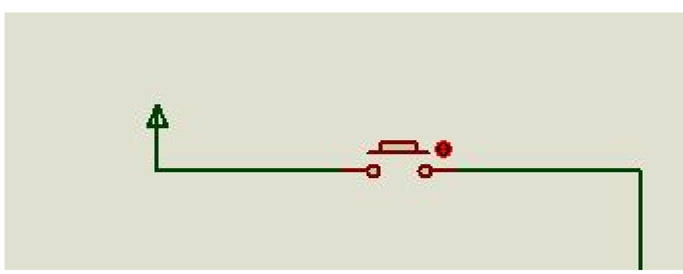


Fig. 12. The button schema

2.4 Risk Analysis

There are some parts in our hardware that poses risk to the successful completion of our project. In terms of the control module, its great complexity gives it a high failure rate. Our control module need to control the display of 9 LEDs, receive the input signals from 9 pressure sensors and communicate to another 4 LED for displaying score. Our microprocessor need to be powerful enough to handle such complicated demand. The

memory size and I/O port number have to be enough to meet our tic-tac-toe game requirement. For our game we need at least 6 outputs and 4 inputs.
Another possible failure is our LED module. During our test we found that the connection between our LED board, power, microprocessors is very unstable. For example when we use breadboard and cable to connect the Vcc, Din, Gnd and Clk in our LED board, there exist a high chance that the LED would die out or display only half of the screen. But when we plug the LED board directly into the breadboard, the LED works fine. Any little unstable connection would kill our LED display. We are worried that it would become very tricky in the future since we would have 9 LED display. To solve the problem we need to carefully design our PCB board and use proteus 8 to test our circuit output constantly to ensure there is no underlying problem. Furthermore, we need to design a firm and stable frame to hold our LED hardware since we are a portable device.
Finally our pressure sensors are another uncertainty. Although we brought 9 resistors with same model. There could possible be accumulated error for the instruments. So we should not just believe the number from datasheet. To avoid such error we need to test individual resistors to find their read standstill resistance. After that we would deploy corresponding resistance to adjust the error.

2.5 PCB Board Layout

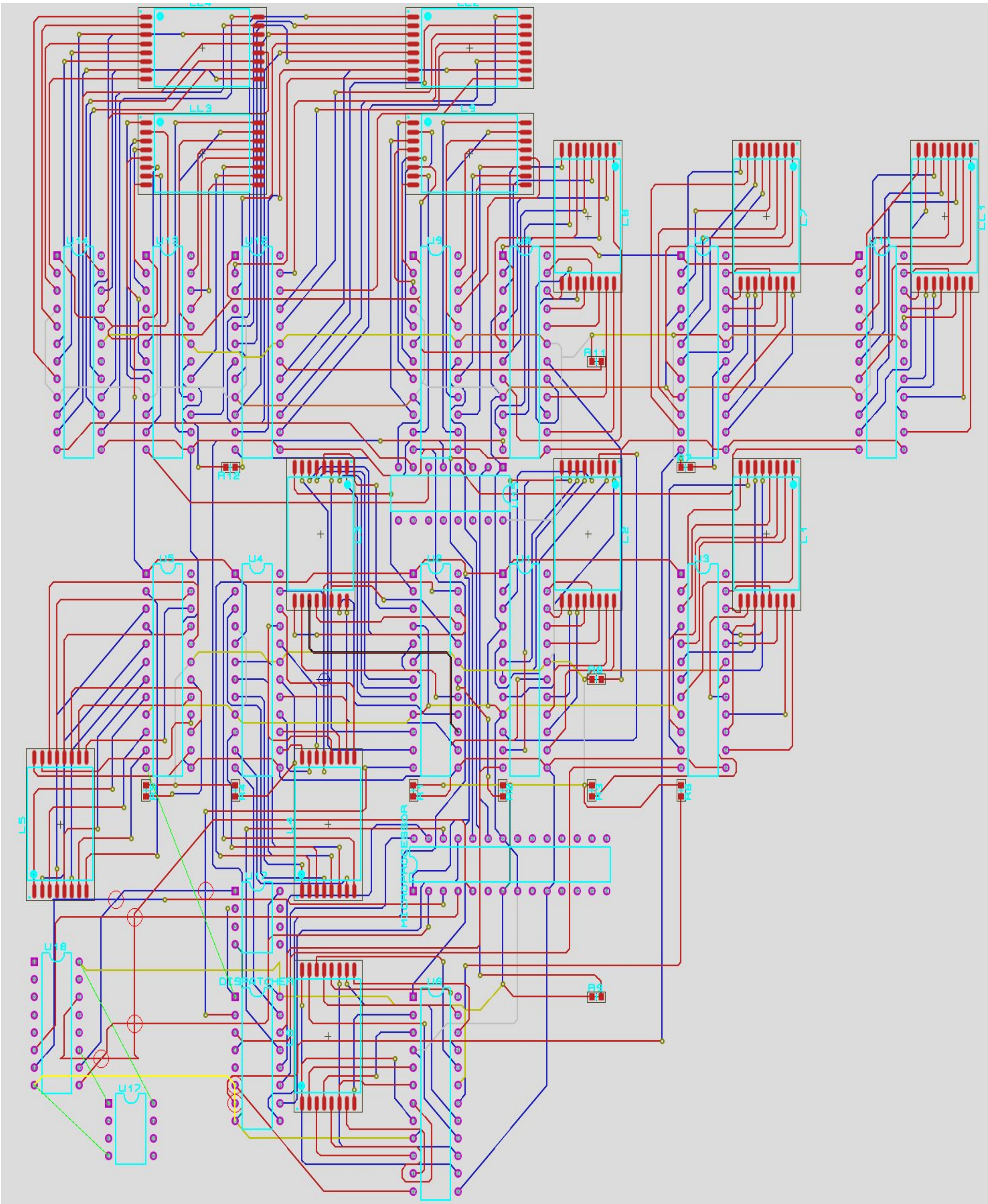


Fig. 13 PCB display

2.6 Software Overview

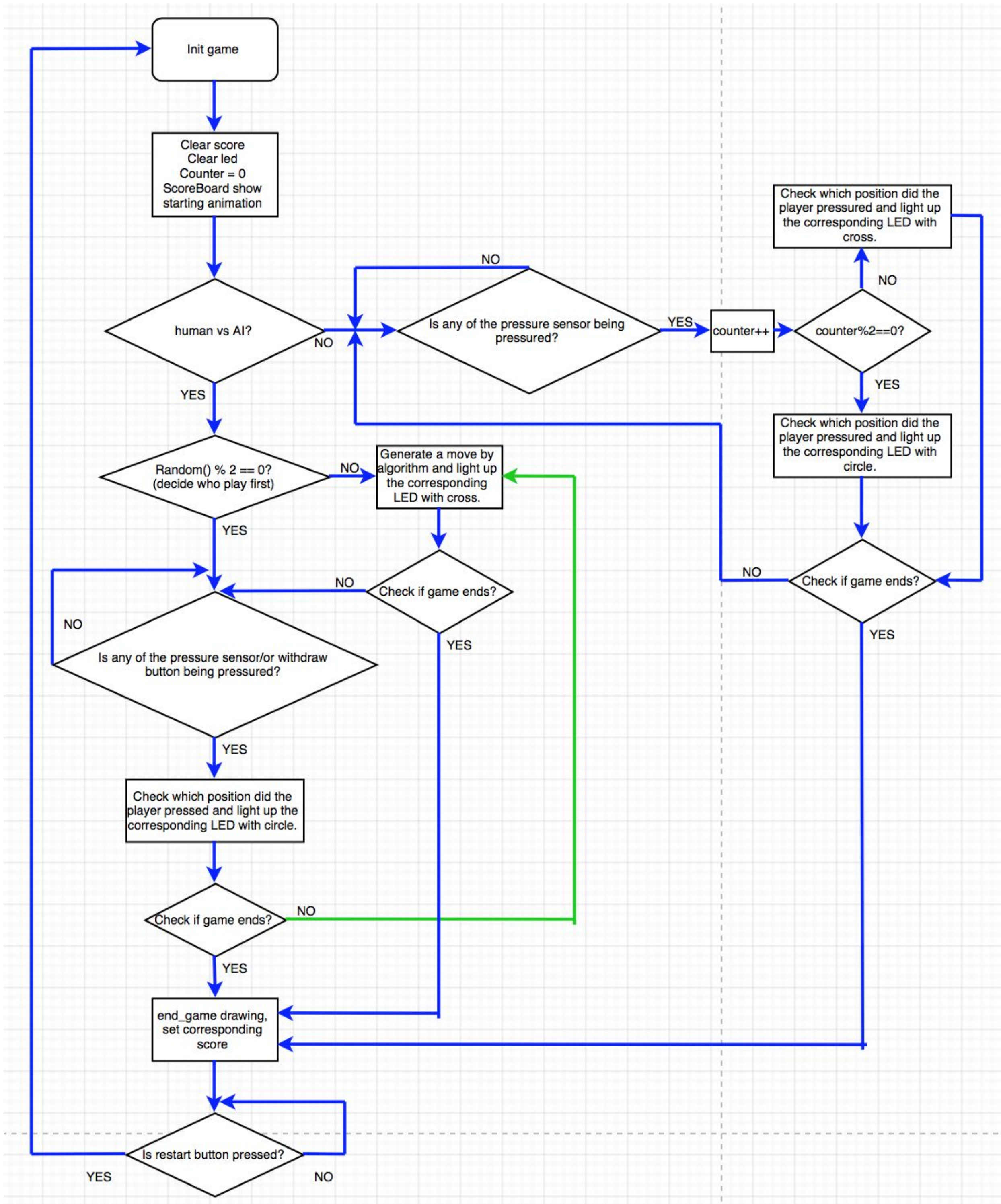


Fig. 14 Flowchart for software [11]

3 Requirement s and Verification

Requirements	Verification	Score
Power Supply (1) Need to continuously output an 5.0 voltage. The voltage varies within a 5% range. (2) Outputs current should be at least 10 mA	(1) Use multimeter to check the power source voltage. Should be 5.0 (+-5%) V (2)Use multimeter to check the current. Should be under 10mA.	5
A/D convertor module (1) The A/D converter must be able to give Digital signal as its output. When player press on the FSR, the A/D converter would give output voltage > 2.65V as digit 1; while the standby output voltage of the A/D converter would be < 1.0 mV as digit 0. (2) Encoder(multiplexer) should be able to read all nine inputs from A/D converters and give correct 4 bit output message. (3) Our A/D convertor module circuit should take the little deviation of the resistance in 9 resistors into account and correct the error.	(1) (a) Use multimeter to check the voltage for output1 pin in our comparator. Should be bigger than 2.65V when outputting digital1. Should be smaller than 1mV when outputting digital 0. (b) Use Arduino digitalread() to check output from A/D converter. Should read 1 when we press the pressure sensor. Should read 0 when we release the pressure sensor. (2) Use Arduino to test the output from Encoder. The 4 bit output should match which pressure sensor that we pressed. Only one sensor can be pressed and recognized by the microprocessor at one time. (3) There may be some little error and deviation for our pressure sensor. We should measure the resistance of each individual resistor one by one and adjust our circuit module according to the condition of the pressure sensor. If we successfully handle this case, we can read correct output from our Arduino so the verification is quite straightforward.	10

LED module (1) LED matrix should be able to light up all its pixels. (2) LED matrix should be able to show cross pattern and circle pattern. (3) 9 LED matrix should be able to show corresponding patterns collaboratively.	(1) When LED is powered. We should correctly configure the LED component (led intensity, display mode) and give enough power supply. All LED pixel should lighted up if correctly been initialized. (2) We would use Arduino to give data to LED’s Vin. We would let it display either “X” or “O”. “X” or “O” should be displayed on the LED. (3) After we connected all the 9 LED module. We should able to decided which one of the 9 LED module to display and what picture pattern is been display on this LED. We would use Arduino to give signal to our LED module. The LED module should able to understand the signal from Arduino and behave correctly. For example, our test case can be: (a) let the leftmost LED to display “X” (b) let the middle LED display “O” (c) let middle LED change from “O” to “X”	10

Microprocessor (1) Microprocessor need to give voltage > 3.0 V as digitalwrite 1 , and voltage < 0.8 V as digitalwrite 0 when it is giving data the LED matrix. (2) Microprocessor of choice(AT Mega-328) should have 4 ports for input; 5 ports output for LED matrix on the tic tac toe panel; another 3 ports output for LED score board. (3) Microprocessor should be able to communicate with LED modules and A/D Converter module within 2 seconds.	(1) Use multimeter to test the voltage and current of the PIN for microprocessor. The correct number should be > 3.0V and < 0.8V. (2) Count the input ports and output ports on our microprocessor, if the number matches our requirements, it is verified. (3) Connect the microprocessor to a pc and log out the state of the microprocessor. Press the forcing sensing resistor and check if the microprocessor can take that input and light up the corresponding LED within 2 seconds.	10
LED score display board (1) LED matrix should be able to light up all its pixels. (2) LED matrix should be able to show cross numbers. (3) 4 LED matrix should be able to show corresponding patterns collaboratively.	(1) When LED is powered. We should correctly configure the LED component (led intensity, display mode) and give enough power supply. All LED pixel should lighted up if correctly been initialized. (2) We would use Arduino to give data to LED's Vin. We would let it display number zero to nine. (3) After we connected all the 4 LED module. We should able to decided which one of the 4 LED module to display and what picture pattern is been display on this LED. We would use Arduino to give signal to our LED module. The LED module should able to understand the signal from Arduino and behave correctly. For example, our test case can be: (a) let the leftmost LED to display "9"	5

	(b) let the middle LED display “0” (c) let middle LED change from “0” to “2”	
Software (1) Give the most admissive tic-tac-toe strategy (2) AI VS human and human VS human	(1) Our AI should be smart in the tic-tac-toe game. AI should not play randomly. AI should either win or draw. (2) We can play in two modes. At the beginning of the game we are able to select which mode to play. The verification should be quite self explaining during the demo.	10

4 Costs Analysis

Our costs is composed by two parts: human labor fee and fixed manufacturing prototype costs. Regard our human labor, we estimated our salary to be $38/hour. Our project actually started at the fourth of this semester. So totally there are 16 weeks. In Saturday and Sunday the work is considered overtime, salary is 1.5x normal salary which is 38*1.5. Each week we will work 8 hours from Monday to Friday and another 8 hours in weekends. The calculation for our labor cost is:

$$3\ people\ *\ 16\ weeks\ *\ \left(\frac{\$38}{hr}\ *\ 8\ h\ +\ \frac{\$38}{hr} * 1.5\ *\ 8\ h\right)\ =\ \$\ 36480$$

Part	Cost(prototype)	Cost(bulk)
FORCE SENSING RESISTOR,0.5 INCH ,CIRCLE,1oz-22LB,FLEXIBLE * 9	$ 134.55	$58.41
SainSmart MAX7219 Red LED Dot Matrix * 13	$107.77	$107.77
Atmega328p-pu Chip	$4.37	$4.37
PCBs	$3.10	$1.95
Lithium-ion batteries ???		
Total	$249.79	$172.5

So the total cost for our project is 36902.29 dollars. We does not consider the other cost such as electricity, air conditioner and other expense. The real cost would be a little bit higher than what we calculate but the difference should be small.

5.Schedule

Week	**Lyu**	**Zhu**	**Yang**
9/25	Build the LED circuit for testing, record the testing data. Understand the Arduino I/O.	Build the LED circuit for testing LED, test the LED I/O, understand the LED datasheet.	Buy necessary hardware components, write code for LED testing.
10/2	Test the pressure sensors, record the testing data and diagram. Design the circuit schema and make the PCB eagle assignment for A/D converting circuit.	Build the A/D converting circuit on whiteboard. Design the circuit schema and make the PCB eagle assignment for A/D converting circuit. Write testing code for reading signal from the A/D converting circuit.	Write the notebook. Building the LED pattern dispatcher circuit for one LED. Given certain input to the dispatcher circuit, LED should display desired pattern.
10/9	Working on testing Arduino-A/D circuit-LED component. Writing codes in Arduino to test the interaction between pressure sensor and LED display. And build the tic-toc-toe decision making program.	Connect the A/D circuit to Arduino to LED circuit. Adjust and test the whole circuit according to the real data. Draw the schema.	Building dispatcher circuit for nine LEDs to Arduino. Record the testing data and notebook. Arranging the sequence and flow for the 9 LED display.
10/16	Complete the function for the tic-toc-toe gamebox. Functions includes AI VS Human, reset, withdraw, Human VS Human. Add animation effect.	Extract the microprocessor chip from the Arduino. Design the surrounding PINs and circuits for microprocessor. Replace the Arduino with microprocessor	Testing and adjusting the data sensing from each device. Record data and complete the lab notebook. Make LED score block circuit design. Add two buttons for reset and withdraw.

		module.	
10/23	Review the Whole PCB design and debug the circuit. Draw the PCB circuit from first round PCBway orders.	Integrate the whole Hardware circuit together. Working on integrated test and unit test. Testing the PCB design on software.	Embed the hardware design into the cube box ordered from ECE supply store. Recorded the data for final papers.
10/30	Get the PCB and debugging the whole project.	Get the PCB and debugging the whole project.	Get the PCB and debugging the whole project.
11/6	Check the software performance and see if we can improve the speed and lower the memory cost.	Adjust the PCB design to submit it to Final Round PCBway.	Conduct environmental testing. To check the working temperature, stability and moisture.
11/13	Add more software features to improve the visual effect.	Working on building the structure of the hardware to make it user friendly.	Paint the game box and write user manual for our project
11/20	Thanksgiving break	Thanksgiving break	Thanksgiving break
11/27	Working on final paper and prepare for the demo	Working on final paper and prepare for the demo	Working on final paper and prepare for the demo
12/4	Working on final paper and prepare for the demo	Working on final paper and prepare for the demo	Working on final paper and prepare for the demo
12/11	END	END	END

*Names are all surnames: Lyu->Nuochen Lyu   Zhu->Jiacheng Zhu   Yang->Minkang Yang

6.Tolerance Analysis

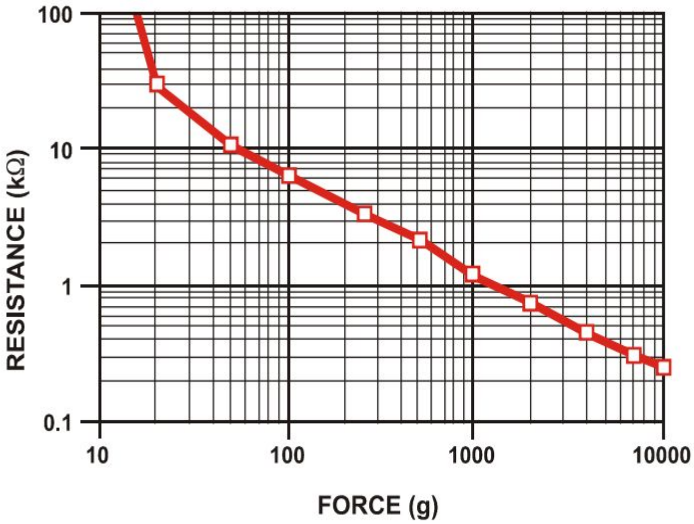


Fig. 14 the chart for force VS resistance[4]

The module with the most uncertainty is our design is our input section. The force sensing resistor is very sensitive, and can cause false alarm. In our physical design, we put the the force sensing sensor(FSR) under the LED matrix and when the player press on the LED matrix, the pressure will be added to the FSR under the LED matrix. However, the LED matrix has its own weight, which will give pressure on the FSR and cause the op amp to send signal(digit 1) by mistake(false alarm). The weight of the LED matrix is 28.35 gram. We use a voltage divider circuit configuration.

RFSR = resistance of the force sensing Resistor

RP = resistance of the paring Resistor that is in serial with FSR(75K ohm).

VOUT = (V+) * [RFSR / (RFSR + RM)].

With the weight of the LED matrix on the FSR, the voltage across the FSR will go down to 2 V(threshold is 3 V), which will trigger the op amp to output digit 1.
In order to avoid that situation, we will put several soft springs under the corner of the LED matrix to support its weight.

MAX7219 tolerance test:

According to the MAX7219 Datasheet, Iset from PIN 18 will decide the peak segment current , which is crucial for signal receiving

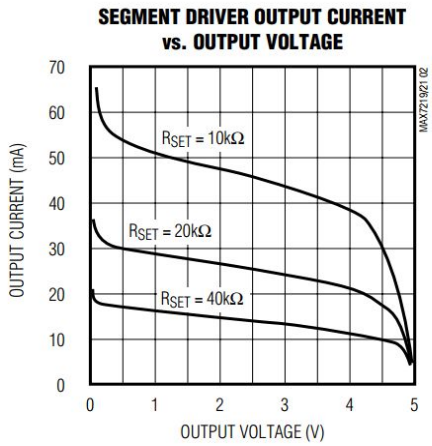


Fig.15 Max7214[5]

Since Rset is used for setting the peak current, as resistance increase the Output current is decreased according to the ohm's law

$$I = V / R$$

However, as the voltage increase, the current becomes lower ,such result will not affect overall testing since we did not consider output of the LED matrix

7.Ethics and Safety

According to IEEE code of Ethics #1[6], we have to aware of the potential safety hazards with our project. Since our tic-tac-toe game box is a portable device, a lithium-ion battery is applied for power supply. We need to avoid the battery overheated which would result in a fire. The combustible fire can be handled by a foam extinguisher which is installed in the senior project lab. All of the members of our team have completed the lab safety training, and we have understood the position and usage of the fire extinguisher. Moreover, we will constantly check the surrounding temperature of our circuit board to make sure it does not have a thermal runaway (between 32°F to 113°F)[3].

Another risk we need to take into concern is the use for Children. In 2012, the U.S. Consumer Product Safety Commission (CPSC) reported 11 toy-related deaths and an estimated 265,000 toy-related injuries treated in emergency rooms[9]. Our tic-tac-toe game box is an electronic device that could be dangerous for small children. We must have prominent warning labels so that kids will not access the dangerous circuit hardware of our game box. The cords and strings would be hidden inside the game box so that it would not pose a strangulation hazard for infants. We would set the age limit for our tic-tac-toe game for people older than eight years old according to the U.S. Consumer Product Safety Act[7].

Regarding environmental protection, since our game box is a box, we would need a cube box to encapsulate the whole inner circuit and leads. The panting and material for the cub box should be degradable and environment-friendly. The pigment for the box painting should be non-volatile and non-toxic[8].

We would finally make our detailed instruction book for this tic-tac-toe game box so that the customer would have enough knowledge to avoid the potential dangers.

According to Student Rights and Responsibilities part 4 Academic Integrity Infractions 1-402 Academic Integrity Infractions[10], we will not use or attempt to use any code, data, circuit design or idea without authorization and citation. We highly appreciate all professors, TAs, and any other people who give us helps during the project. According to IEEE code of ethics # 7[6], all sources of help and reference would be acknowledged and included in the reference section. We might seek inspiration from GitHub, sparkfun.com, arduino.com. Even if we figured that we might be able to reuse some open source code from the websites, I stated above or some other websites, but we will develop a good portion of our code and circuit by ourselves and keep our project a very original one. For devices and components, we purchased from amazon or Sparkfun or any other website, we would state the price, where we got them, and how are they implanted into our circuit and design.

The customer can only use our game box to play games. The input voltage of our game box would be only 5 V, which is very safe for play even if the coat of the wire is worn out, and the player touches the broken part. We would help each other during the whole process and make sure that everyone in the team is on the same page.

References

[1] High-Speed CMOS Logic 10-to-4 Line Priority Encoder, Texas Instrument, November 2003, http://www.ti.com/lit/ds/symlink/cd54hc147.pdf

[2] 8-bit AVR Microprocessor, Atmel Corporation, 11/2016, http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

[3] MAX7219 Serially Interfaced, 8-Digit LED Display Drivers, Maxim Integrated, 2003, https://datasheets.maximintegrated.com/en/ds/MAX7219-MAX7221.pdf

[4] Figure 2 force vs resistance, FSR force sensing resistor, Interlink Electronics, https://www.sparkfun.com/datasheets/Sensors/Pressure/fsrguide.pdf

[5] Serially Interfaced, 8-Digit LED Display Drivers, maxim integrated(TM), 2003, https://datasheets.maximintegrated.com/en/ds/MAX7219-MAX7221.pdf

[6] Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 29- Feb- 2016].

[7] United States Consumer Product Safety Commission, “Regulations, Mandatory Standards & Bans”, 2014, https://www.cpsc.gov/Regulations-Laws--Standards/Regulations-Mandatory-Standards-Bans

[8] Natura Veda Innovations from nature, “The Dangers of FD&C Color Pigments”, 2017 https://www.naturaveda.com/pages/the-dangers-of-fd-c-color-pigments

[9] BU-410: Charging at High and Low Temperatures:http://batteryuniversity.com/learn/article/charging_at_high_and_low_temperatures

[10] University of Illinois at Urbana-Champaign, article 1 - student rights and responsibilities, 2008, http://studentcode.illinois.edu/article1_part4_1-402.html

[11] Minkang Yang, University of illinois at Urbana-Champaign, ECE 445 Block Diagram, 2017, https://www.draw.io/#G0Bwtp5vDqfIdWbnEyWUtGOG5RTGc