



University of
**Central
Florida**

E-FROG GAME

SENIOR DESIGN I

EEL 4914C

GROUP #6

ASTRID CRUZ

CARLOS NEYRA

JAIME SALAZAR

SUMMER 2012

AUGUST 2ND/NOVEMBER 13TH, 2012

TABLE OF CONTENT

1.0 Executive Summary.....	1
2.0 Project Description.....	2
2.1 E-Frog Game History.....	2
2.2 E-Frog Motivation.....	3
2.2.1 Goals.....	4
2.3 E-Frog Game Description	4
2.4 Objectives	6
2.4.1 Main Control Unit Objectives	6
2.4.2 Sensor Objectives.....	7
2.4.3 Requirements and Specifications	8
2.5 Block Diagram	9
2.5.1 Function of the Project.....	10
2.5.2 Responsibility Chart	11
3.0 Research Related to Project Definition.....	12
3.1 Related Projects	12
3.2 Important Technologies.....	13
3.2.1 Central Control Unit Options	13
Main Controller Unit.....	13
Suitable Options for the Central Control Unit.....	14
T.I. Stellaris	14
Atmel 2560	15
PIC30F6012A.....	16
3.2.2 Independent MCU Options	17
Atmel ATmega128	18
Microchip PIC Family	19
T.I. MSP430	20
T.I. MSP430G2.....	21
Microcontroller Programming	21
Assembly Language.....	22
C-Language.....	22
3.2.3 User Control Panel System.....	23
Coins	23

Display.....	23
Digital Input.....	24
3.2.4 Hole Detection Sensory System (HDSS).....	24
HDSS Using Mechanical Micro switches.....	24
HDSS Using Distance Sensors	26
HDSS Using Infrared Barrier	28
3.2.5 Threshold Distance Sensory System (TDSS).....	32
TDSS Using a Threshold Strip Sensor	32
TDSS Using an Infrared Distance Sensor	33
TDSS Using a Transmitted Beam Sensor.....	33
3.2.6 Coin Counter Sensor System (CCSS)	35
One Sided System of Infrared Lights.....	36
Building an IR Curtain Barrier	37
Building a Customized System to Detect Coins.....	38
3.2.7 Enable / Disable Switches.....	40
DIP Switches	41
3.2.8 Display Screen Counter	42
LCD and LED Displays.....	43
Light Crystal Display (LCD).....	43
Light Emitter Diode Display (LED).....	44
3.2.9 Audio Effects System	45
3.2.10 Decorative Light System	47
LEDs	47
A-Series Light Bulbs	48
T-Series Light Bulbs.....	48
3.3 System Integration Options.....	48
3.3.1 Power System	49
Batteries	50
4.0 Project Design	52
4.1 Microcontroller Design.....	52
4.2 User Interface System Design.....	54

4.3 Hole Detection Sensory System Design	57
4.3.1 Light Emitters.....	61
4.3.2 Laser Diodes	61
4.3.3 The Silicone PIN Photodiode	62
4.3.4 Constructing the Demo	63
4.3.5 Matrix Type Hole-Detection System Array.....	65
4.4 Threshold Distance Sensory System	67
4.5 Coin Dispenser System	68
4.6 Display Screen Counter	72
4.7 Sound Effects System	75
4.9 LED Decorative System	78
5.0 Design Summary	81
6.0 Testing and Prototyping of the E-Frog Game	84
6.1 Testing Procedures.....	84
6.2 Testing of the Hole-Detection Sensory System.....	86
6.3 Testing the Coin Dispenser.....	91
6.4 Decorative Lighting System Testing.....	93
6.5 Threshold Sensor Testing	96
6.6 User Interface System Testing.....	97
6.7 Display Screen Counter Testing	99
6.8 Sound System Testing.....	101
7.0 Administrative Content	103
7.1 Project Milestones.....	103
7.1.1 Senior Design I Summer 2012.....	103
7.1.2 Senior Design II Fall 2012.....	104
7.2 Project Budget and Financing.....	105
7.2.1 Sponsors and Significant Contributors	106

1.0 Executive Summary

The group number six of Senior Design I Summer 2012 created this document in order to cover the project named Electronic Frog Game (E-FROG). The name of this project was chosen based on its physical appearance of a frog and on its main function of entertainment. This project idea was selected to provide an easy, fun, and interactive way to facilitate a team case coin throwing skill game. In effect of this game, group number six expects motivation of this coin-throwing game to obtain a very fun entertaining and unique experience with the E-FROG GAME rather than using regular manual tools.

The E-FROG GAME is composed of two parts; the hardware and the electronic part. Physically the game is made of a standing wood table box of 4ft height, and the top box of the game dimensions are 24"x 24" x 8". On the top of this table, we notice 20 holes evenly spaced out one from another in an array of 5 columns and 4 rows. Each hole has a diameter of 3". The main attraction of the game, the bronze frog, is located right at the middle of this array. Also, 3 flipping obstacles are located in the first row of holes. Each hole is going to represent a different value having of course the highest points where the frog and the 3 obstacles are located. A number of 10 bronze coins of 1'5" diameter and 1 ounce of weight are going to be used to throw them from a specific distance to the top of the wooden box where the frog and holes are located.

The electronic part of this project consists of quite a few improvements that we have added to this game. Such addition includes the infrared laser array emitters/sensors that are going to take care of the counting of the thrown coins that passed through the holes, the system of decorative LED lights that are going to be surrounding the wooden table box, the hidden speakers that will play the sound effects, the scan sensor that will detect all coins thrown around the table box, the interactive light system that will flash according to the situation of the game, a threshold sensor that will keep account of the proper distance from the game, a display unit showing up the current scores of the players, a system of selecting bottoms to configure the recent game, and all the proper electronic components such as wires, microcontrollers, resistors, inductors, capacitors, battery, chargers, diodes, and so on needed to interconnect the previously mentioned system.

The definition about this game that is given by the "Real Academia Española" (The Royal Spanish Academy is the official royal institution responsible for regulating the Spanish Language), says: "The Frog Game involves inserting a metal coin from a determined distance through the open mouth of a metal frog placed on a table, or other suitable arranged grooves". This means that the object of the game is to throw a few (usually 10) small metal chips, coins or rings, aiming them as precise as possible to any of the holes on the table or better through the mouth of the small metal frog located at the center of the array.

The purpose of the E-FROG GAME is to give its players and opportunity to look for a way out of their daily and stressful lives, to help them forget about their problems and give them a boost much need. The E-FROG GAME is played as follows, the maximum number of players is five (5), and the allowed distance to throw one coin is about 20 ft. from table. (A barrier sensor is used to tell when a player crosses the line. A buzzer sounds to alert the player so that he/she throws the coin from a valid distance. Each hole in the table is worth a predetermined amount of points. The maximum number of points per round, per player, is 4,000, with 4,000 points being the highest score per coin thrown (when throwing one coin through the frog's mouth). A scan sensor is placed on the floor/ground to alert the players if more than 10 coins have been thrown (10 is the maximum number of coins that can be thrown in a given round, per player).

2.0 Project Description

2.1 Historic Background of the "Sapo":

The ancient legend tells that in the Inca sacred lake, the Lake Titicaca, there evolved a mystical game that with the passage of time was made into a tradition. "El Sapu" (the frog) was the main character in the exciting game where the Inca, son of the sun, would throw the luck with his court. At that time, the royal family would throw gold pieces in the lake, with the hopes of catching some frogs' attention which were well known for their magical powers. If a frog came to the surface and took in his mouth a gold piece, instantly the player was awarded a wish and the frog turned into solid gold piece and was the price for the player to keep as award to his good luck.

Honoring so many wishes which became reality, the Inca ordered a great golden frog made, for all of royalty to enjoy. It was a game of suspense and dexterity, where dance and happiness would mix in one rite: PUKLLAY SAPU (Playing Sapo, or Frog), a Quechua expression.

Another historical trend of this game dates back three centuries where it was known in France, the game called "Le Tonneau" was not nothing but drums or barrels that served as the objective for the coins or elements used to be thrown inside of these barrels that were used for the structure of the game; then, such game begun to be called "La Grenouille" (frog) when they said they replaced the barrels by holes on a table and the main attraction the metal frog.

In Peru and Colombia, the games lack the element "Vieja", an element located at the back wall of the wooden table where the frog sits that was introduced with the passage of time in order Argentina becoming the most valuable objective in the game.

There are some others names used, such as:

- Inca : Sapu
- Quichua, Diaguita y Mapuche: Ampatu
- Guarani: Cururu
- Spanish: Sapo
- French: tonneau o Grenouille
- English: Toad
- German: Frosch.

There is another brief history of the game of the Frog in Asturias, Spain. According to Gerardo Ruiz, a Spanish historian says: "The Frog is a traditional game of accuracy and precision launch, which was developed in Asturias in the environment of winches and cider, sharing space with bowling game and the Key (another Spanish traditional game)". On the other hand, from a French background, the game has been playing in Asturias at least since the nineteenth century, as evidenced by some bibliographic documents in the newspaper El Comercio Gijón since 1891. Moreno Palos (another Spanish sportsman expert in traditional sports and games in Spain) note that the frog comes from a French play, "The direct antecedent can be found in the eighteenth century in the French game called Tonnueau, which consisted of throwing chips, coins or any objects to a table or a barrel that had different holes, arches and a windlass".

2.2 E-Frog Game Motivation

Today biggest cause of sickness and problems is the stress. People get too concern about their agitated lifestyle overloaded with too much work, family stress, lack of time for their enjoyment, and bad habits of sleep. We, Astrid, Jaime, and Carlos are part of this stressful society since we have to deal daily with the load of responsibilities such as full time jobs, time with the family, school classes and homework.

It's been a year since we study together and we found a way to relax in between study time, and one of the big sources of relaxation that we found was the "Sapo" game that Carlos has in his house and brought it from Peru, South America. We all loved this game since day one since it's a very traditional game played in Colombia and Peru. As the months went by, we thought... "Why we don't implement and create an electronic version of this fun game and make it easier to use and more attractive?"

Studying and working hard every day could be stressful and tiring for many people. In order to relax people enrolled in bad habits or addictions. Sometimes people quit jobs or school because of the society pressure. In addition, we noticed that student's daily routine can cause exhaustion, and might influence in the class performance and therefore have some consequences in your grade. Our group though about and innovative design that will resolve the issues of students and employees in means of stress situations.

The electronic Frog game will be an intelligent hi-tech product made with high quality components. It will be easy to use indoors. The most important thing is that you will be able to take it where you want to go and it could be easily played.

An Electronic powered game will resolve the issues of stress employees and students have from one place to another and to be able to bring it everywhere because of its portability.

This Electronic Game will reduce the issues of stress that students and people are experiencing today. We decided to call this innovative concept the E-FROG GAME, or in other words the Electronic Entertainment Coin Throwing Game.

GOALS

In this world where technology is all over the place, classical games have been put aside, and that is a shame since there are so many great games that have enlighten us since we were kids. The "SAPO GAME" is a simple and not so fancy game but very fun to play. The idea is to innovate it by adding new features and attractive technology. The goals of this project are to prove ourselves that we are capable to apply all the knowledge, hard work, and skills learned throughout the school years. Then, the challenging part is to practice and be able to create, design, and build a new fun product that has never been implemented to this scale before.

2.3 E-Frog Game Description

The E-FROG GAME is made based on its original predecessor, "The Sapo Game" which is a game that consists in a wooden table with holes at the top of it where there are some obstacles at the first row and the main attraction of the game further back at the middle of the table, the bronze frog. The coins slide down through a tunnel built in each one of the holes down to an incline that guides them to the front of the table to be collected at the end of each game and manually counted depending on which hole each one of the coins landed.

The concept of this traditional game is very simple and fun, but as we have said before, our vision to this project was to make it even more attractive and 'eye catching' by adding new features, lights, sounds and some limiting barriers that will help to the game to be conducted in a more standard way to all players,

leaving no allowances for cheating or miscounting. Since our game is going to have a system that will detect and count automatically the points in each one of the holes, we are going to describe this game as a two-part description game which is composed mainly into the following parts; the hardware and the electronic part.

The following picture [Figure 1] depicts a traditional Sapo Game played in South America; the picture was taken in one of the member's house.



Figure 1Traditional Sapo Game

Physically the game is made of a standing wood table box of 4ft height, and the top box of the game dimensions are 24"x 24" x 8". On the top of this table, we have 20 holes evenly spaced out one from another in an array of 5 columns and 4 rows. Each hole has a diameter of 3". The main attraction of the game, the bronze frog, is located right at the middle of this array. Also, 3 flipping obstacles are located in the first row of holes. Each hole is going to represent a different value having of course the highest points where the frog and the 3 obstacles are located. A number of 10 bronze coins of 1'5" diameter and 1 ounce of weight are going to be used to throw them from a specific distance to the top of the wooden box where the frog and holes are located. Since the game does not need the sliders to be built in each one of the holes because the counting part of the points

is going to be counted by our sensor unit, our game model is going to have a slight different construction from the original one. There is going to be only one big slider guiding all the coins to the front of the game so when the coins pass by any hole, the determined sensor will take account of the input sending the information to the microcontroller for later processing as the coin falls on top of this inclined base that will make it fall through a collecting opening where all the coins will be picked up at the end of each game.

The electronic part of this project consists of quite a few improvements that we have added to this game. Such addition include the infrared laser array emitters/sensors that are going to take care of the counting of the thrown coins that pass through the holes sending the information to its corresponding microcontroller for its proper handling; the system of decorative LED lights that are going to be surrounding the wooden table box that will serve as an attractive to the game, especially if the game is decided to be played indoors; the hidden speakers that will play the sound effects that will sound according to each specific situation of the game; the scan sensor that will detect all coins thrown around the table box and by doing this the game will automatically count the hits and the misses of the game; the interactive light system that will flash according to the situation of the game, a threshold sensor that will keep account of the proper distance from the game and it will avoid that some players will take advantage in getting closer to the unit for cheating during the game, this will work very similar like the barrier sensors used in bowling that penalties a player if it crosses the line of game; a display unit showing up the current scores of the players and the total score as well; a system of selecting bottoms to configure the game based on the number of players and requirements; all the proper electronic components such as wires, microcontrollers, resistors, inductors, capacitors, battery, chargers, diodes, and so on needed to interconnect the previously mentioned system.

2.4 Objectives

2.4.1 Main Control Unit Objectives

The main control unit is the principal system of the E-FROG Game, and it consists of three major parts: the ATMEGA 2560 microcontroller, the emitter diodes LEDs and the display monitor. This unit is the brain of the game that would receive data from inputs and control the rest systems to display the corresponding outputs.

ATMEGA 2560 Microcontroller Unit:

- Responsible for receiving and interpreting the data from the sensor units.
- Transfer of data to / from ATMEGA 2560 MCU and subsystems

- Be the interface media for the programmer to configure it to the desired specifications and functions of the game.
- Control the interaction and developing of the game at all times.
- Communication with sub-systems
- Send Signal to be display in all different outputs

LEDS:

- Send new signal from MCU to change the lighting pattern for a specific situation during the game
- Pregame light pattern
- Change of player
- Individual coin score
- Highest score display
- Winner sign
- Decorative light system
- Alerts if player is too close to the board

Display monitor:

- Display updated information in the display screen (LED,7-Segment)
- Player number
- Partial / final score
- Coins left
- Highest score
- Points Counter

2.4.2 Sensor Objectives

The hole-detection sensory system is going to be composed of an infrared barrier that will play a critical part of the E-Frog Game. It will allow the system to distinguish on which hole the coin has passed through and it will be responsible of “telling” the microcontroller such information for a proper handling of the data. To do this, two laser arrays and two emitter arrays will be strategically placed underneath the holed table and will work in together to detect the passing of the coin through the detection planes forming a matrix array. When the coin passes through any of the holes, the system will detect the passing of the coin by ‘sensing’ the temporary cut of the constant signal through a pair of laser/receiver and knowing which pair has got interrupted we will be able to determine which hole has been used. As well as the matrix array of IR sensors, the implementation and addition of a distance sensor to set up a threshold predetermined distance is also used and incorporated in this game.

HDSS (Hole-Detection Sensory System) Objectives:

- Laser array with infrared phototransistor.
- Detect when a coin passes through the sensor array.
- Detect the hole coordinates by which the coin passes.
- Infrared Barrier.
- Coin detector / array sensors detect the coin thrown.
- If only coin dispenser detects the last coin thrown.
- Distance Sensor.
- Sets minimum distance allowed for a player to throw a coin.
- Buzzer alerts a player when the line is crossed.

2.5 Requirements and Specifications

The following list shows the requirements and specifications that the E-Frog Game consists of:

E-Frog Requirements Table Box

- The number of holes is twenty (20) evenly spaced out.
- Maximum weight is thirty (45) pounds.
- Ten (10) bronze coins.
- Portable and ready to use.

Subsystems

- Central Control Unit.
- Able to handle fifty (250) consecutive games.
- Able to be driven by an ATMEGA2560 MCU.
- Able to be controlled by a user interface module.
- Aesthetic LED Array Matrix.
- Able to display up to five (5) different light patterns.
- Able to display different light patterns for each coin that goes through a hole.
- Able to display light patterns consecutively after any different input.

User Interface

- Able to handle multiple players.
- Give players the ability to select up to three (3) functions
- Select type buttons (4 in total: Reset, Threshold, Coin Dispenser, and Sound).
- Small display screen to visualize messages and points.

Infrared Barriers

- Able to detect a coin of 1.25" in diameter.
- Able to detect a coin traveling at a fast speeds.
- Able to cover an area 2 feet wide by 2 feet long.

Power

- Able to supply the E-Frog game with 350W of power.
- Able to run continually.

E-Frog Game Specifications

Subsystems

- Aesthetic LED Lighting System.
- LED voltage range: 1.5-3.3v.
- LEDs are 5mm in diameter
- LED viewable angle of <90 degrees
- Central Control Unit
- Coin Dispenser and Threshold as external devices added to the game.

User Interface

- Uses two (2) 8x8 LEDs matrices for game display.
- Uses three (3) slide-style buttons for enable/disable options.
- Uses one SPDT switch to turn ON/OFF the game.
- Uses one momentary-style button for resetting the game.
- Uses one (1) eight-digit seven-segment display for players' scores.

Infrared Barriers

- Infrared emitters are 5mm in diameter.
- Infrared detectors are 5mm in diameter.
- Nine (9) infrared barriers that cover an area of two feet long by two feet wide.

Power Supply

- Able to take in 120V.
- Deliver power at 3.3v, 5v, and 12v.
- Built-in current protection system.

2.6 BLOCK DIAGRAM

The following block diagram shows the parts (features and systems) that the E-FROG GAME will use and the overall interactive behavior between them to perform correctly. As the figure shown below, the system has four inputs: user control panel system, hole detection sensory system, threshold distance sensory system, coin dispenser system, enable-disable switches. These four inputs were designed and programmed to be connected to the central control unit (CCU) which receives all data from the inputs to control and lead the function of each sensor system in order to perform a total of four outputs: display screen counter, visual animation system, audio effects system, and panel light system.

Finally the power source consists in an ATX computer-type power supply that would power up the system for it to work. Also, the research of each main part is going to be divided as is shown in the block diagram. As well as what components are going to be acquired specifically are going to be chosen the member responsible in their parts. The design put it all together would be part of the entire team member for a complete work project design.

The following block diagram [Figure 2] shows all inputs and outputs systems that the microcontroller will manage.

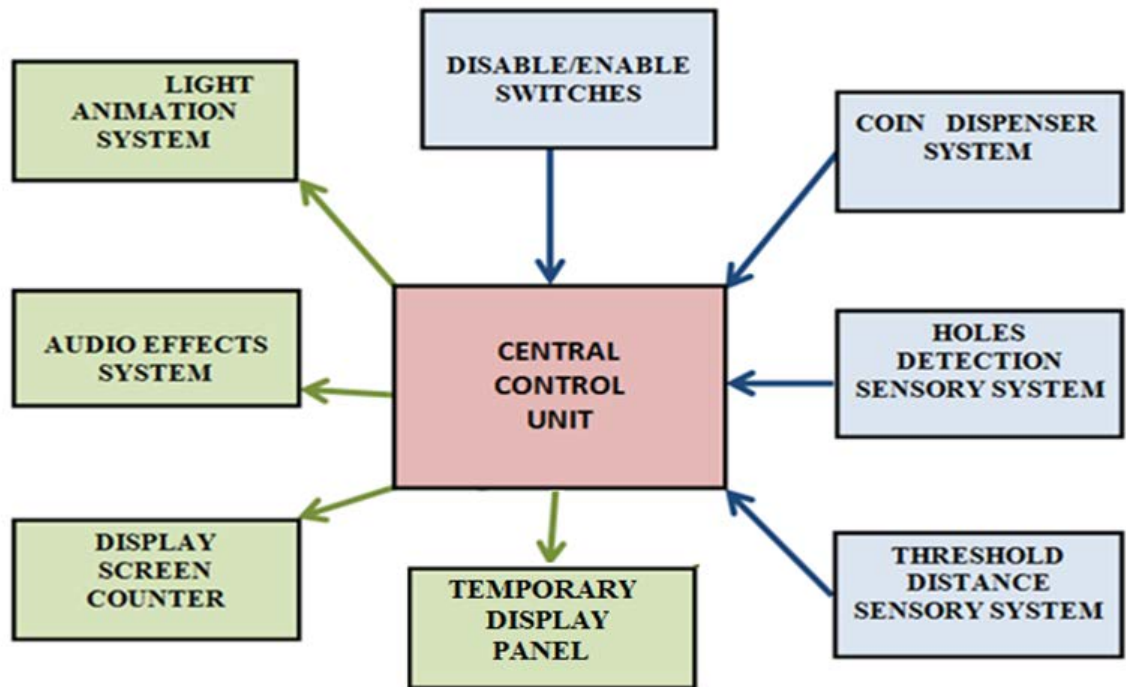


Figure 2- E-FROG Block Diagram

INPUT OUTPUT CENTRALCONTROLUNIT

2.6.1 FUNCTION OF THE PROJECT

The main function of the project is to diffuse entertainment and enjoyment to the users creating a relaxing environment with all the new gadgets that technology put in our hands to incorporate them into the game. By doing this, people will gather socially and relief daily stress. Sometimes human beings need an “excuse” to get together, and what a better excuse than playing a fun, interactive, and unique game.

2.6.2 RESPONSIBILITY CHART

The following chart [Table1] is a short summary of the group duties and responsibilities referring to all different systems inputs and outputs.

	Astrid Cruz	Carlos Neyra	Jaime Salazar
System Inputs	Enable/Disable Switches	Hole-Detection Sensory System	Threshold Distance Sensor System
		Coin Counter Sensor System	User Control Panel System
System Outputs	Display Screen Counter	Audio Effects System	Decorative Lights Systems
	Visual Animation System		

Table 1 Group #6 Responsibilities

The following description shows an approximation of how the amount of work and responsibilities will be divided within the group members. We are basically deciding to split up the load of this project into three main responsibilities:

Astrid Cruz is going to be the hardworking assembler. Her previous background and experience as an electrical assembler are going to be fundamental for the accomplishment of the wiring and routing of all the electrical components as well as her support in the research of information. Astrid will be concentrating and focusing her attention in the following areas of the project: enable/disable switches, display screen counter, and visual animation system.

Carlos Neyra is going to take care of all the research, process, and execution of the programming of all the subsystems of the project, making sure that operate properly and accurate as we all expect. He is also going to support the building and assembling portion of the project. The main objectives and responsibilities under his shoulders are going to be the portion of the hole-detection array system, the scan sensory system, and the audio effects system.

Jaime Salazar is going to take care of the aesthetic portion of the project. His talented skills as a great creator and designer of innovative ideas is going to be the key to the “wow” factor of the game, besides, of course, of his electrical skills when assembling all the components together. His specialty in this project is going to be the development and execution of the threshold distance sensor system, the user control panel system, and the decorative lights system.

3.0 Research Related to Project Definition

3.1 Related Projects and Inspiration for the E-Frog Game and Systems:

The idea of making the Frog Game into an electronic version was a challenging proposal for ourselves since we did not have any previous resources or projects similar to the final proposed product that we wanted to implement and also the lack of a common pattern or similar product made it harder for us to create the E-Frog Game. We began thinking about possible ways of making this project work and we started putting different ideas together to build up the different subsystems of our game. The concept of introducing an aiming game probably came from the very popular darts game available everywhere since bars to the comfort of our houses.

As we know, this game had its beginnings in a very old-fashion way of playing that consisted in throwing darts from a specific distance to a panel with circles enclosing one to another creating different areas where the hardest and most valuable ones are the areas closer to the center of the panel where the highest points are right in the middle of this array. Being a simple but enjoyable game, developers thought and envisioned to see this game in a fancier manner, with lights that decorate and animate the game and a detection and counting system that calculates the points of the thrown darts automatically helping the players to focus their concentration in the act of throwing the darts more than the sometimes complicated task of counting and adding repetitive times the points gained throughout the game.

Another example or guide that was used to implement our E-Frog Game was the bowling game where most of us had spent good time playing with. One more time, engineers have developed over the years evolution of the bowling game that went from the conversion of old-type of aiming game that consisted in just throwing a ball and hit pines to the top-of-the-line new bowling systems that count with many electronic features such as sensors located at each one of the pines to detect if they have been hit or not, the integration of animated sound system that will increase the lever of enjoyment, to the automatic screen counter of points and the important introduction of a threshold or limiting sensor that detects if a player crosses the edge of the lane.

Also, among all related projects and research that we have done we found the many implementations of beer pong games that are around. Students and engineer developers were creative enough to also turn this very American game into its electronic version that detects the hit and miss balls, implement some light effects according to the situation of the game and even including extra systems such as ball washers and ball dispensers making the game more pleasant and amusing.

From the previously mentioned games (darts, bowling, and beer pong) we tried to put together some of the innovative systems developed in each one of those games and merge them into our E-Frog Game to make it better and attractive. Systems such as the automatic counting system present in all of the games, or the light animation system present in the darts game, or the dispenser idea that comes from the beer pong or bowling games, or even the threshold sensor that limits and conditions the player to throw from a specific distance that comes from the bowling game are some examples that we implemented previously development systems into our project.

3.2 Important Technologies

3.2.1 Central Control Unit Options

MAIN MICROCONTROLLER UNIT

Thanks to the rapid growing of technology, we have found that there are many different microcontrollers units available on the market today varying in many different specifications important to consider in order for implementing it to this project. There is a vast variety of microcontroller families to choose from with different architectures, input/output ports, and internal memory. Deciding on a proper controller is a factor that will be important to the design process because it will drive the rest of the programming throughout an easier or harder path depending in how we choose to arrange the date to the chosen controller.

There are facts and needs to be balanced between cost and the performance. There are some microcontrollers that lack PWM pins, internal ADCs, or an internal NIC, while there are some others with excessive tools that are not necessary to applied them to this project. Some devices may cost less than their competitors, but adding those features with other modules may be time consuming and end up costing more money. Throughout doing the proper research we can take a responsible decision in choosing the right main controller for the E-Frog Game.

Some of the main functions of the central controller will be to:

- Communicate with the hole-detection sub-system that is located under the holed table that senses when a coin has entered through a hole.
- Communicate with the audible sub-system.
- Control movement of game LEDs animation system.
- Keep track of the score.

- Keep track of the number of coins remaining for each player from the coin counter sensory system.
- Keep track the proper positioning for throwing per person thanks to the implementation of the threshold sensor.

Display information to display monitor.

- Score.
- Current player turn indicator.
- Warns when last coin is being used for current player.
- Control the animation systems as desired per the programmers.

Suitable options for the Central Control Unit

In the research of different processor units that were considered as the central controller for the E-Frog Game we notice some similarities that each one of them had to offer. The PIC30F6012A processor had its 16-bit channel for its transmitting and receiving data. The ATMEL 2560 ARM processor had multiple communication methods it could use which made it flexible to choose devices that also work with those communication methods; it is an 8 bit AVR modified RISC architecture, with 32 8 bit general purpose registers.

The Stellaris ARM processor also has multiple communication methods. One of the processor that was taken in consideration to maneuver these tasks is the TI Stellaris EKS-LM3S8962 ARM processor. Some of the important reasons that will make us decide on choosing this product is the type of communication methods that uses such as the I2C and UART, making it to become the central controller choice for the E-Frog Game. Also, another great reason to prefer the last mentioned option (Stellaris) is its low power consumption and affordability since, as we know, Texas Instruments sell very affordable products for beginning developers.

In the next few lines we are shortly highlighting some important features and capabilities of the proposed options for the main controller unit of the E-Frog Game.

TI Stellaris

The Texas Instruments Stellaris is an ARM based microprocessor with an integrated networking controller and can contain up to 512KB of flash memory and 64KB of RAM. They have many different I2C and UART busses for communication and many different timers for various different functions. The chips operate at or above 25 MHz, which seems to be more powerful that what is

needed for the scope of this project. To program a Stellaris device you need to have access to the TI software library and have a license. You will also need a Stellaris flash programmer in order to connect to a computer to receive programs. You should be able to use GCC to compile C code to an ARM assembly file, which can be uploaded to the Stellaris chip through the programmer; the [Figure 3] shows the Stellaris Block Diagram.

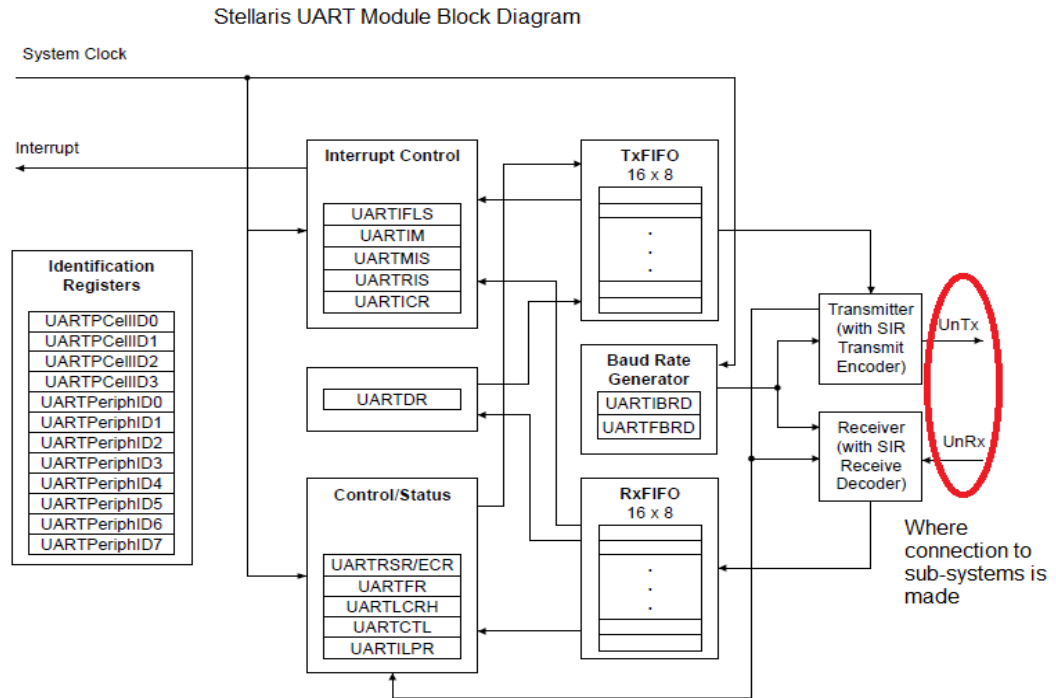


Figure 3 Stellaris Block Diagram

ATMEL 2560

The Atmel 2560 is a chip in the 1280 family of Atmel chips. It is an 8 bit AVR modified RISC architecture, with 32 8 bit general purpose registers. It has 54 digital IO pins, 16 analog input pins. It has 128KB of flash memory and 8KB of RAM. The clock speed of the chip is 16 MHz and CPU speed capable of 16 MIPS. There are a few different power modes for the chip for power saving. When active at 1 MHz it will draw 500 micro amps, but will only draw .1 micro amps in power down mode. Our system will not need to be active at all times so this feature will save power.

The 2560 is the chip in the Arduino Mega development board. This means that there are many open source libraries that work directly with the chip which will greatly reduce the amount of coding necessary for the project. There are also many examples of projects using this chip. Another perk of using this chip is that you can program them with another Arduino connected via USB to the computer with the source code, and you do not have to buy an expensive In System

Programming (ISP) device, the following diagram [Figure 4] represents the functional block diagram of ATMEL AT91SAM.

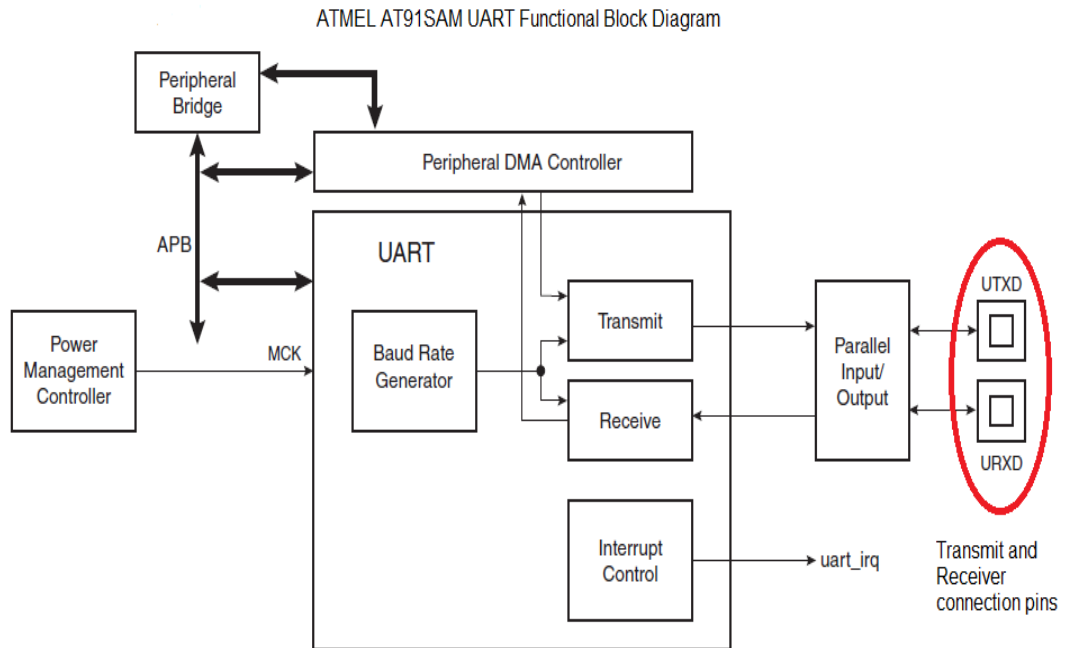


Figure 4 ATMEL Block Diagram

PIC30F6012A

The PIC30F6012A is a 16 bit microcontroller by PIC. It has 52 IO pins, a 16 channel 12 bit analog to digital converter along with 144 KB of storage space and 8KB of RAM along with 4KB of non-volatile EEPROM. The chip has a faster clock rate than the Atmel 2560 and the Stellaris 6100 at 30MHz. There is a C compiler optimized for use with the PIC30F6012A. Each pin is a sink and source of 25mA. The PIC30F6012A contains 5 16bit timers/counters which can be combined to create a 32bit counter. It also has an on-chip watchdog timer with built in low-power oscillator.

There are 8 pins that can be used for PWM, only 2 of which will be needed for the ALARM system though. [Figure 5]

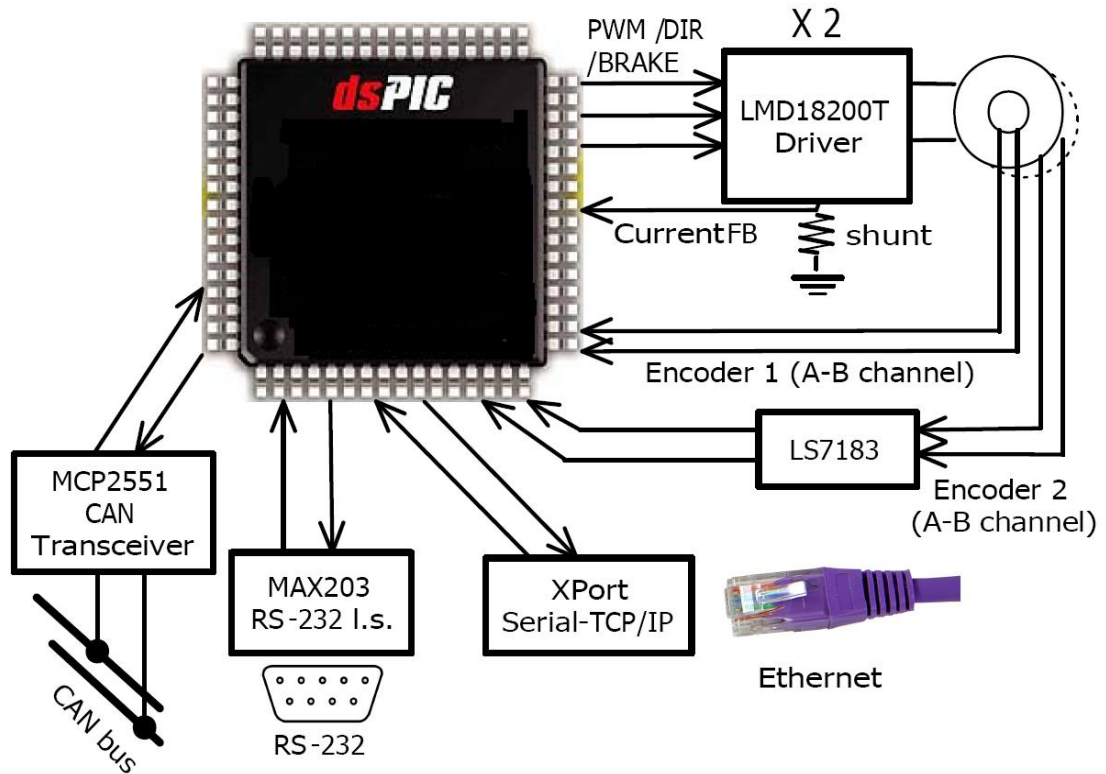


Figure 5 PIC pins

3.2.2 Independent MCU Options

In the research for appropriate microcontrollers suitable for our subsystems such as the counting system, threshold system, or audio system, among others, we started looking for options that will satisfy our simple and basic requirements for the tasks that we wanted them to get accomplished. There are numerous companies in the market that offer a wide range of microcontrollers and processors and choosing the appropriate one was not an easy task for our group. In the quest and research of appropriate microcontrollers and components to operate the E-Frog Game we considered a few requirements to take in consideration a specific company for the usage of its microcontroller.

Some of the requirements were reliability of the component, availability in the market, easy to use technology, with enough I/O pins to implement to its fullest each one of the proposed systems without falling in the case of choosing overdeveloped microcontrollers that will only increase the cost of the materials of the project without needing to do so, to be a low power microcontroller to be able to operate it for longer time and create a less impact to the environment, to be well known by the community so we can get some help and guidance from other projects and people from a big and well developed library of contents and sample codes, and as already mentioned, and one of the main requirements of the desired microcontroller is the cost of it.

Atmel ATmega128

In our search for a good and reliable microcontroller, we found that the ATMEL_ATmega128 has many of the capabilities needed to achieve this project. This microcontroller is widely used in the industry could serve as an important tool in the development of the E-Frog game. The ATMEL-ATmega128 microcontroller is used with the popular Arduino Uno board that comes with the new 8-bit technology, and has many similarities with the FPGA board. Even though the Arduino Uno is smaller in size compared to other known microcontrollers, it is still a very practical and powerful tool.

The ATMEL_ATmega128 uses a low power AVR 8-bit processor with a 16 MHz clock cycle. It has 32KB of Flash memory for storing a program, 1KB of EEPROM to store non-volatile data, and 2KB of RAM. This microcontroller also features 6 channels of 10-Bit Analog-to-Digital Converter (ADC), most use 3.3 volts in order to operate, and 23 programmable I/O lines. Since we are looking to have small subsystems, each charged with achieving a specific task, this microcontroller can easily do the work for us. What caught our attention is how the programming is done. The Arduino boards use their own type of programming. This type of programming is very similar to the C and C++. They are two of the most used programming languages, and for people with a basic background in C there should be no concerns about the programming of the microcontroller. [[Figure 6](#)]



Figure 6 Arduino UNO that uses Atmel ATmega 128

Microchip PIC Family

Microchip is a very well-known company that has been in the market for more than 15 years and is one of the leading companies in the world in electronics devices. Microchip possess a wide variety of microcontroller units that range from the 8-bit to the 32-bit microcontroller units such as the dsPIC[®] Digital Signal Controllers provide designers with an easy upgrade path from 8-bit PIC[®] microcontrollers and a cost effective option to 32-bit MCUs, having a 8-bit, PIC16F57 MCUs that is very suitable and compatible with the requirements and needs of our E-Frog Project. The broad product line includes everything from extreme Low Power microcontrollers to high performance digital signal controllers. Combined with hardware and free software these 8-bit products make the design of mTouch[™] Sensing, Graphics, Connectivity, Digital Power Conversion, Smart Energy and Motor Control solutions easy to implement.

Many hobbyists use this line of microcontroller for different projects, with many using this family of microcontrollers for the projects similar to ours. If this type of microcontroller was to be chosen for our group we would choose the PIC16F57. The reason for considering this microcontroller is the high use of this microcontroller which makes troubleshooting any issues, fairly easy. This microcontroller comes with 20 I/O pins which in case of the both the data and the motor microcontroller is more than enough. Programming on this type of microcontrollers can be done with the MPLAB Integrated Development Environment software. With this development software the group can program the microcontroller with the C-Language.

PIC 16F57 Specifications:

- Clock Speed – 20 MHz clock speed.
- On-Chip Flash program memory 2048 x 12-Bit.
- General Purpose Register (SRAM) 72 x 8-Bit.
- Operating Current: 170 μ A @ 2V, 4 MHz's.
- 20 I/O Pins with 28-pin DIP.
- Operating Voltage: 2.0V to 5.5V.
- Temperature Range: -40°C to 85°C.
- It has over 40 year retention period.
- Power-On-Reset.
- Watchdog Timer.
- Selectable Oscillator between RC, crystal, High Speed Crystal, and Low frequency Crystal.

These are the some of the basic specifications for the PIC 16F57 microcontroller. The price of this microcontroller from digikey.com is \$2.48. Looking at this microcontroller we then considered another Microchip controller afterwards that seems to provide a better alternative to the PIC 16F57. [Figure 7]

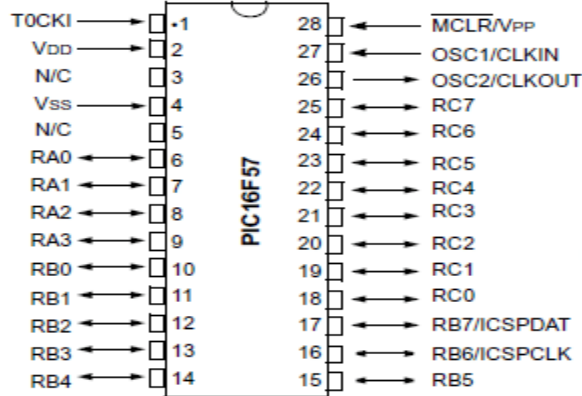


Figure 7 PIC1657 Pin locations

Texas Instrument MSP 430:

MSP 430

The E-FROG game requires some microcontrollers in order to develop independent tasks from inputs and outputs that are controlled from a main central unit. In order to create these subsystems we, group number six, have found a great option of microcontroller which is the Mixed Signal Processor MSP430. The MSP 430 microcontroller is a 16 bit CPU microcontroller designed by Texas Instrument. This is a very inexpensive device and it is designed in particular for embedded applications and low power consumption. The MSP430 is implemented especially for battery power uses. [Figure 8]

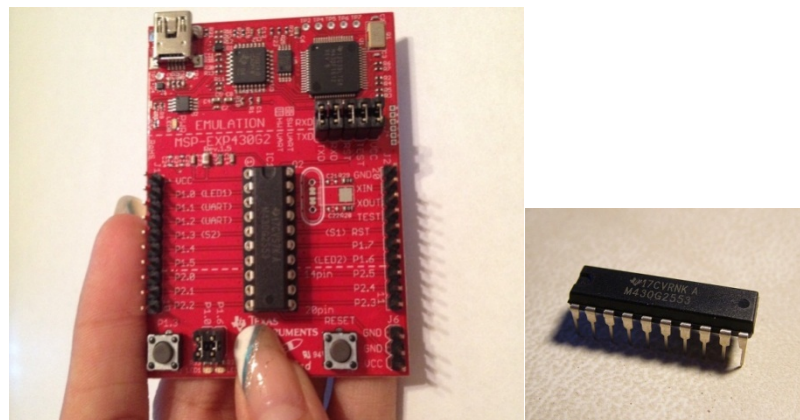


Figure 8 Development Board (TI)

MSP430G2 Microcontroller

MSP430G2

The Microcontroller MSP430G2 is a small chip that has two important characteristics that were eye catching to group number six. These are the very low cost and power consumption. It is designed with program space of 521B-16kB with the option of three whole eight bit Input / Output ports if surface mount Integrated Circuits, speed of 16 MHz, RAM to 256KB, flash memory of 0.5 to 16kB. Its power consumption at 2.2 v has RAM retention of 0.1uA, 0.4uA standby mode (VLO), 220uA million instructions per second active, 0.7uA of actual clock mode, and very fast wake-up from standby mode in less than 1us. The MSP430G2 comes with an inexpensive board named Launchpad made by TI.

Talking about program this device, usually the software is made on a PC and the move to flash ram of the MCU thru the USB. The common programming languages utilized are C and Assembly. In order to program the microcontroller unit it is required knowledge about hardware job as well the software leading several components of the MCU. Particular memory locations (registers) with especial functions would configure, turn on or off the peripherals MCUs. All of these bit switches are addressed by name since there are so many of them.

Microcontroller Programming

In order for our E-FROG GAME system works, we, group number six have to program the microcontroller chosen; we need to load a program before the microcontroller is useful. We found a massive diversity of microcontroller families to select from with different architectures, input/output ports, and internal memory. Determining a proper controller is a factor that will be important to the design process because it will drive the rest of the programming throughout an easier or harder path depending in how we choose to arrange the data to the chosen controller. This decision is the most important in our project since the MCU will be in charge of many important duties such as communicate with each system, keep track of scores and number of coins, control LEDS animation, and display data on the display monitor. Most of the microcontrollers are programmed through their own language, similar to the Assembly language. There are development kits for all the microcontrollers, which can make programming easier with the C Language. Apart from the immense majority of hardware resources required to implement the project functional requirements it is imperative that a suitable programming architecture be chosen in order to execute the desired functionalities defined by the project scope. Not only must the programming environment enable the implementation of the desired functionalities, it must also limit on the amount of code for which solutions already exist. As thus, this section delves into programming languages, code libraries, and other resources that will enable the designers to extract the necessary implementation from the available hardware. Prior to writing the program that will control the device, it is important to have the ability to transfer the program itself to the controller that will act as the mind of the device. As such,

a programmer will enable the team to implement the desired functionality for the available hardware. The device features a USB connection to a computer with the ability to power the target microcontroller when connected.

Assembly Language

One of the languages used to program microcontrollers is Assembly Language. Programming the microcontrollers can be done by the microcontroller's native language. Programming through assembly was done previously by all group six members in a class prerequisite for Senior Design, which was Embedded Systems. Having this in mind, basic program commands are easy to decode, and it would be easy to follow if one group member were to troubleshoot another group member's code. Programming through Assembly would require more time, as each line of code in the C-language would require anywhere from 3 to 10 lines of code in Assembly, and it receives the signal at one of its ports. It seems that programming will be a lot of time consuming for programming the data microcontroller. This is because the data microcontroller will need to run the LED display and control the data sent to the LED display and also make many duties. This method does not seem to be very practical when a lot of data needs to be processed.

C Language

Another common high language used to program microcontrollers is the C Language. The C language is a very popular language for programming microcontrollers. C Language was done previously by group number six members in the intro to C class, but as this is an intro class, it is not as fresh as Assembly, which we used in Computer Organization and Embedded systems classes. Programming with the C Language is based on the development kit that comes with the microcontroller, but most all of them have the option to program the microcontroller with C. Programming with C would also require less time, as recurring code is done utilizing loops. The data microcontroller would definitely require to be programmed in C or some other higher-level programming language.

There is also programming examples on the datasheets of the microcontroller in the C Language that can support the group number six to troubleshoot any errors in code. Programming in C is also made at ease with the header files that are included with the development kits, where many pins of the microcontrollers are defined, which decreases the need for the group number six members to declare these. Very related language to C is the C++ Language which is an object-oriented language. This is not required much in the microcontroller as the use of objects is more used in other higher level projects. Then C++ can still be used but it is not necessary for our E-FROG GAME project.

3.2.3 User Control Panel System Options

COINS

The E-Frog game needs to have a user interface system, which is going to interact with the player and the digital components of the game. This is a very important part of the project because, for instance, materials used for the “coins” or “tokens” have to be compatible with sensors and other parts of the components. The first topic to take into consideration is the material used for the coins. The group can choose between a wooden, a plastic, or a metallic copper-made coin. The fact that an array of sensors is going into the flat surface of the game makes it very difficult for a wooden or a plastic coin to be read.

Also, when the game is played, the coins that miss the target will probably fall down to the ground, therefore damaging the coin. Another concern is that the wooden or plastic coin does not meet the requirements established by the group (the coin has to travel at least 7 m/s). The plastic coin, for instance, being a very light coin only could travel a few feet in distance depending on the environment where the game is played. On the other hand, heavier more compact copper coins travel faster, are better read by the sensors, and will not be damaged as much by the constant hitting against the body of the game or the ground.

DISPLAY

The E-Frog, being an arcade type game, has to have some type of display that helps the players throughout the game by showing the updated information for each player after each throw. The first option the group had was not having a display unit. It is the easiest and cheapest way to implement the project, but without a display unit, the burden rests with the players to try to either memorize their score or use pencil and paper after many coins have been thrown. Another option is of course having a display unit. A LED screen could be used to account for every score the player obtains.

Also, the screen could show the number of players, the score as it is adding up, and the total amount of points gotten by each player. Even though, the fact that having a display unit means more work in the design and actual construction of the game, it is a very valuable addition to the project because in a simple way our microcontrollers can be programmed to carry out all these operations. One more option for the display unit of the game is to have a LED unit with a seven segment. The game will have at least one screen for every player (maximum of 5). Although this is a more challenging idea, it is probably the one that best suits the E-Frog since every screen shows every stage of the game independently during any given game.

DIGITAL INPUT

The digital inputs of the game have to be easy to use, and straightforward to navigate. Several options have been taken into account for this part of the project. The first option is to use relay switches. For our menu, we would have to have either one switch or a combination of switches to perform a desired task. For instance, there would be a switch for the main menu and then two more switches would allow the user to choose from the “next” or “previous” part of the menu. Also, another combination would allow the user to change to another player.

Even though relay switches work, they are a little difficult to use because of the various functions of the project. The next option the group has discussed is the use of momentary switches or button switches. These switches are simple to use, and for the purposes of the project, implementing them would make the menu easier to navigate. For instance, a momentary switch is used in a doorbell. The doorbell only rings while the button is being pressed. There are a variety of arcade type buttons that the group could use. For the project only 4 buttons would be needed: one menu button, one “up” or “down” button, and a player selection button. One last button would be used for resetting the game each time all rounds are finished.

3.2.4 Hole-Detection Sensory System

A) Hole-Detection System Using Mechanical Micro Switches:

The idea of having a system that detects the inputs in each one of the holes of our project was one of the most important tasks that we needed to research in order to decide the best one or the most appropriate one that will work better with the environment and the mechanical parts that compose this project. Throughout the research that we did, we found many options available to utilize and accommodate with the E-Frog Game, some of those options fancier than others, some more expensive, and others no so useful. Among the options that we found, we considered to have as an option the usage of mechanical limit switches.

Basic Micro Switches have a very small contact gap and operate at the specified movement and force guided by an arm that pushes the pin that sticks out from the switch and is the one that clicks the inner contacts using a snap action mechanism. Mechanical micro switches are available in different models and designs with split contacts, maintained operation, drip-proof specifications, high capacity, and DC current specifications.

A basic micro switch is composed of three leads: COM, N.C., and N.O. leads. The first one is called COM or common lead, which is the one that is connected to the other two but this connection is conditioned to the position of the mechanical arm of the device, here is where we usually connect the constant source of power or ground that comes from the emitter part of the system and “waits” at the lead to be transferred to any of the other two depending again, to the position of the mechanical arm.

The second lead is known as the N.C. lead which stands for normally closed lead, this means that under normal conditions the contacts between the common lead and this N.C. lead remains closed or in “touch”, letting the signal pass from one lead to the other until the mechanical arm is pressed and this passage is interrupted. The third lead that is called by its initials N.O. stands for normally open lead and this means that under normal operation, without pressing the mechanical arm, the inside contacts between the common lead and this N.O. lead remains open without letting the signal passing between this two leads.

The two figures below taken by us show the two positions that the micro switch can be operated, at its normal (not pressed) position on the left, and at its pressed position on the right. [Figure9-10]



Figure 9 Normal Conditions



Figure10 Pressed Micro Switch

After analyzing the properties and usage of these mechanical micro switches, one of the ways to incorporate it into the project is by adding a flap or a pallet at the end of its mechanical arm that will be hit every time a coin will touch it in a manner that an individual switch will be mounted underneath each one of the twenty holes. The micro switch will have to be adapted on an inclined, custom-made mounting surface that will allow the coin to hit the pallet if it passes through the hole, a small tunnel will have to be constructed to guide the coin and maneuver its fall into the hole in a way that it will always hit the arm of the micro switch only one time and then continue falling into a common drop where all the coins will fall and to be collected at the end of the game.

The microcontroller unit corresponding to this hole detection sensory system will emit a constant signal to the common lead of all the twenty custom-made micro switches mounted right under each one of the holes, this signal will be passively waiting to be transferred back into the microcontroller unit through one of the twenty normally open leads from the micro switches when one coin will hit one of the pallets and will send a temporary pulse of signal back to the controller. In order to handle this information, we are going to assign one specific input at the microcontroller for each of the twenty possible options that will emit the temporary pulse. By properly program the microcontroller, our program will assign the desired values to each one of the different inputs received from the micro switches sensory system for later execution and usage of these values, such as storage, addition, visual, and audio responses.

B) Hole-Detection System Using Distance Sensors:

Another option that we had available to use it and adapt it in our E-Frog Game was the introduction of infrared distance sensors. Nowadays there are many people interested in the usage of infrared technology and researchers and developers have handled the concept and knowledge learned from this new technology to increase their interest in the introduction of new robotics systems. Many manufacturers have created and produced new sensor arrays and systems to facilitate creation of their new robots. Making use of infrared light is a common element of some of these sensor arrays. Among all the new designs created from the infrared technology is the design of these infrared sensors using an infrared emitter, paired with a charge coupled device detector. By choosing this configuration, these new sensors are capable of measuring the angle of incidence of the infrared light reflected off an object and detected by the detector.

This measurement then provides enough and accurate information about the distance of the object from the sensor, because the distance between the emitter and detector is known by the device. In other words, the emitter will send constantly a beam of infrared light out from the sensor while the receiver will be "waiting" for a response to read it back, staying alert for a reflection of this beam. When an object gets on the way with the beam of light emitted by the emitter, this object will create a reflection back to different directions but the receiver will pick up the reflection that is transmitted straight back to the sensing unit but there will be a small angle created between the reflected beam and the receiver.

Then, the sensor is so accurate to measure this angle that will find the distance of the obstructing object by knowing the angle of incidence of the reflected beam and the distance between the emitter and receiver. The sensor will do an operation such as the inverse tangent of the known angle and side of the created triangulation and by doing that, it is able to determine the distance of the object reflected. This distance is sent out from the sensor as a voltage signal. The greater the voltage, the closer the object is from the sensor with some restrictions

such as the first few centimeters such reading is not accurate until the sensor stays steady for proper reading, also the signal gets faded and erratic after certain distance.

A device of this nature has been invented and created by Sharp. Each of the sensors in the GP2D family of sensors is a unit packing together the emitter and the charge coupled device detector into one unit; and is capable of doing this measurement of the angle of incidence of the reflected object. The figure below may be used as a visual aid to better understand this functionality. In the figure, the measured angles are θ_1 and θ_2 . One will notice that the detected angle for an object which is closer (θ_2) is much smaller than the angle detected when the object is further away (θ_1). [Figure11]

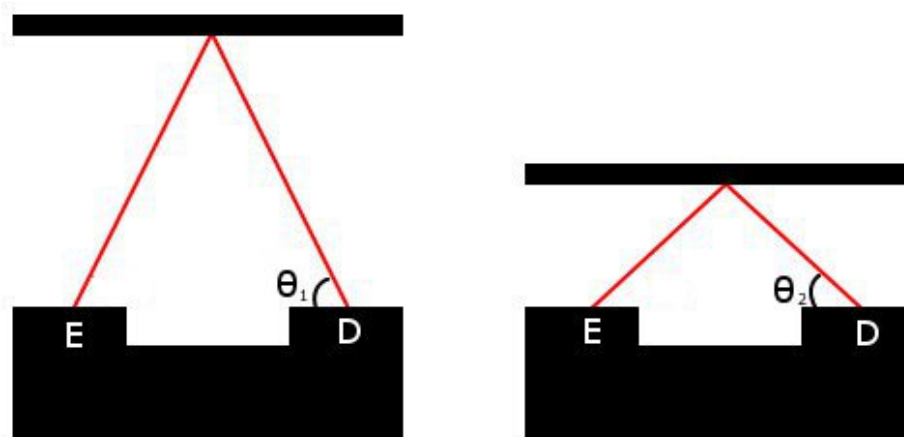


Figure11 Example of Sharp GP2D functionality

By knowing and being aware on how this new technology works, we can also adapt it into our E-Frog Game by using only four of these distance sensors and place them along the left side (designer preference) of the game table aligning them with the four rows of holes underneath the holes entrance (below hole level) hidden from the end user to prevent unwanted maneuverability that may cause damage to the sensors. Another reason to place them under the table is to protect them from excessive light that may distort the proper angle of reflection. As a part of this design proposal, the four rows will have to be isolated from under the table to prevent overlapping of areas of detection of the distance sensors since their coverage area varies from spectrum coverage to an elliptical-shape coverage area like the Sharp distance sensors.

Once the sensors have been properly placed and wired, the next step consists in programming each one of them with the chosen microcontroller to set the right distance of detection in each one of them. For example, if using the Sharp GP2D-type distance sensor we will have to disregard the first 10 cm measured from the sensor since it is an inaccurate area to be measured since it is the setting up region where the sensor gets ready to start putting out scaled voltages inversely proportional with the distance from the object.

After those 10 cm have been taken in consideration, we divided the regions corresponding to each one of the five holes per row to range it with a specific reading from the distance sensor. By doing this, the microcontroller will recognize the input from the coin by accurately reading the voltage output coming from one of the four distance sensors, convert this analog output (in a voltage form) into a digital input reading for the microcontroller by dividing the available amount of bits in the already chosen regions for the holes. Then the program will assign desired game point values to each one of these inputs for later execution and processing to display it out, add it, and put out some audio-visual effects.

The figure below is a representation on the desired system using the Sharp distance measuring sensor device [Figure12].

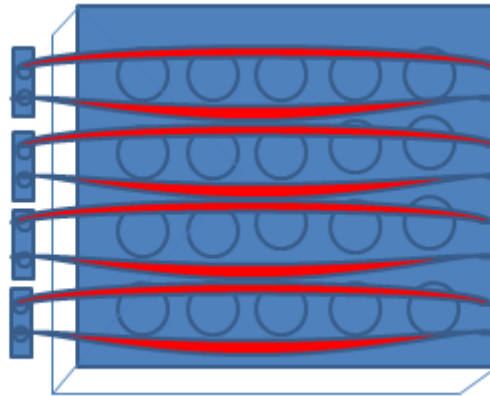


Figure12 Distance Sensor and Lasers diagram

C) Hole Detection System Using Infrared Barrier

Advances in technology have demonstrated and shown that light exists in a spectrum. An individual can easily produce a spectrum from a prism by hold it up to a beam of light, and see the spectrum reflecting out from the prism. There is something that people don't notice and it is that visible light is simply a small portion of the spectrum known as the electromagnetic spectrum. This electromagnetic spectrum contains all physically possible frequencies and wavelengths of electromagnetic fields, ranging from extremely low frequencies which are also known as radio waves, all the way up to high frequency or high energy waves known as Gamma Rays.

Thanks to the advancements in modern technology, people have gained the ability to tie together the powers of these wavelengths which are outside of the range visibility to the human's eye perception. For example, humans utilize this new approach of technology for X-Ray radiation to view bone structure, also, we use the knowledge learned to handle low frequencies for data and voice communication. Nowadays it looks like there's no part of the electromagnetic

spectrum that is outside of human control anymore. As a result of this vast knowledge and handling of light spectrums, one of the very common ranges of wavelengths that humans have been able to adjust and take advantage of it is known as infrared radiation.

Our eyes are detectors which are designed to detect visible light waves (or visible radiation). Visible light is one of the few types of radiation that can penetrate our atmosphere and be detected on the Earth's surface. There are forms of light (or radiation) which we cannot see. Actually we can only see a very small part of the entire range of radiation called the electromagnetic spectrum.

Infrared radiation is that portion of the electromagnetic spectrum that extends from the long wavelength, or red, end of the visible-light range to the microwave range. Invisible to the eye, it can be detected as a sensation of warmth on the skin. Most of the radiation emitted by a moderately heated surface is infrared; it forms a continuous spectrum. Molecular excitation also produces copious infrared radiation but in a discrete spectrum of lines or bands.

Infrared radiation is not only invisible to humans; they are low enough power to be nearly harmless, not only to biological organisms, but also to integrated circuitry. On the other hand, the infrared range of electromagnetic waves exists at a high enough frequency to not be able to pass through solid objects, unlike their higher wavelength counterparts, radio waves.

Because infrared waves cannot pass through solid object such as walls, they have been introduced and utilized into technological applications which require short range remote communication. Possibly the most common application of these infrared waves is the television remote control. This form of remote control uses pulses of infrared light emitted by the remote control, and detected by a photo receiver embedded in the television, in order to communicate over the short distance.

One common application of infrared light is its use as a barrier sensor. For example, when someone goes to play bowling we all know that it is prohibited to throw the ball after the limiting line where the lane starts because it will be considering as a foul for the game. Many bowling systems use the infrared technology to incorporate a barrier sensor that will detect when the player obstructs the light beam and it will send the corresponding broken signal to the 'brain' of the game to let it know that an obstruction through the barrier has happened.

Another good example of the usage of this infrared technology will be an automatic garage door that possesses a set of this sensor and when they are obstructed someone had probably noticed the door becomes inoperable if an object is in the threshold of the door. This is because automatic garage doors make use of an infrared emitter/detector pair in order to sense if an object is in

the way of the door, were it to close. Instead of using pulses of light however, this application continuously emits infrared light on one side of the doorway, which is detected on the other side of the doorway by a photo detector. If the beam of light is interrupted, say by a child walking through the doorway, the system disables downward motion of the door.

Also, there is another common application of infrared radiation and is called passive infrared sensing or PIR. Passive infrared sensors are most commonly used for motion detection like the ones used for security systems in houses. In this form, the sensor determines a “normal” state.

The normal state can be described as the amount of infrared radiation detected the majority of the time; for example, the amount of light detected when a room is empty. Then, the way this sensor works will be when a mobile object enters the view space of the sensor (in this case the room), the amount of infrared light detected by the sensor increases. This is because everything that exists above absolute zero temperature emits radiation.

An emitter/detector pair is relatively simple to implement thanks to today's technology. Using a program like Multisim we were able to construct the following screenshot that shows a configuration which operates in a way which combines the applications described earlier.

In the schematic below, an infrared LED continuously emits light. When there is nothing within the working range of the emitter, the detector, in this case another infrared LED, will have a constant voltage drop. This voltage drop depends on the operating specifications of the LED. When an object moves within the range of the sensor, infrared light will be reflected off of the object and be detected by the receiver LED. This will increase the voltage drop across the LED, which is detected using the comparator. [Figure13-14]

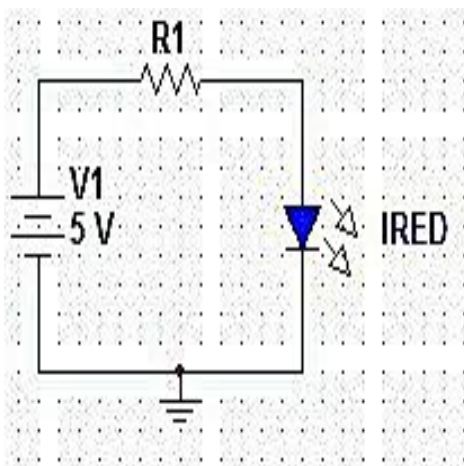


Figure13 Simplified emitter

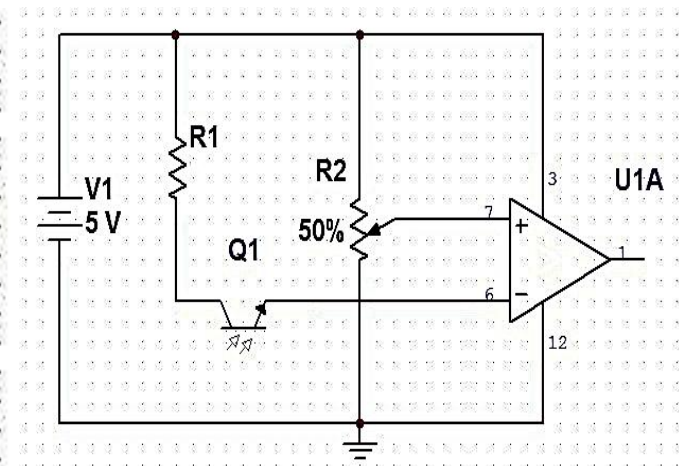


Figure14 Simplified detector

Infrared emitters are available in the form of common LEDs, as well as infrared laser diodes. In general, infrared light generated by a regular LED device diffuses quicker, as LEDs have a much larger viewing angle than lasers do. Also, LEDs can emit multiple wavelengths of light. In the case of infrared LEDs, these wavelengths could range from Near Infrared (NIR), which are the wavelengths closer to visible light, all the way to Far Infrared (FIR), which are wavelengths on the radio wave side of the infrared range. Infrared lasers and all lasers in general, emit a very specific wavelength of light.

If we would use this technology to implement it into our game we will have to install the emitters and receivers under the holed table top, right after the passage of the coins so these can be detected by our LED array system. Since the plan is to have the LED system installed under the table, we will not have to worry so much about compensating for the effect of ambient light on the infrared detectors because these will not be easily exposed to the ambient light.

Since the diameter of each one of the holes is about 2 inches and the coin diameter is about 1.25 inches, the chance that the coin will enter the hole in a flat manner is not always expected because the coin may be flipping at the time it enters the hole, so the proper way to detect the coin getting into the hole will be by putting two laser diodes shooting in the "X" direction of the hole, spaced about 0.5 inches from each other and another two laser diodes shooting in the "Y" direction, again, spaced apart 0.5 inches from each other. The "X" and "Y" directions cannot be on the same exact plane because they will interfere with each other, so one of them will have to be slightly in a lower plane than the other. By having these four beams crossing the underside of each of the rounded holes we will be covering pretty much all possible detection areas of each one of the holes. The microcontroller unit will have to be properly programmed to recognize all different combinations of possibilities to pick up correctly the inputs from these laser beams broken at the time a coin passes through their shooting range.

The sketch shown below is a rough idea of what is desired to do with the infrared laser diode/receiver and the hole dimensions of the table [Figure15]

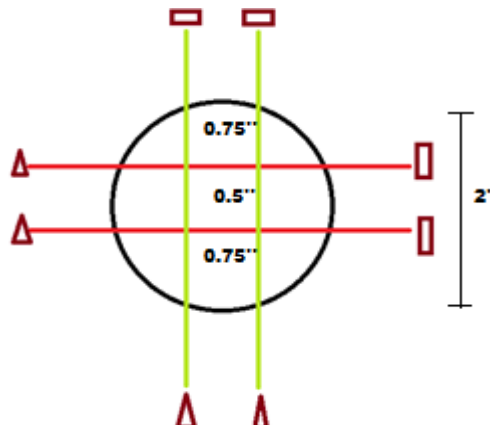


Figure15 Hole Dimensions

3.2.5 Threshold Distance Sensory System Using a Threshold-Strip Sensor 31221A

Another type of sensor that the group could be tempted to use is called “Touch-Sensor” or “Position Sensor”. This type of sensor is used in a variety of applications including gaming machines. A strip sensor has a rectangular shape and it detects the position of an object (in this case the player) by the use of small and flexible potentiometers. Two copper strips lie inside the actual strip, and thus when they make contact, the sensor sends a signal. This force-sensing resistor is a passive component that decreases in resistance when there is an increase in the force applied to the active area, detecting a change when under pressure. The strip sensor is generally designed to sense a force ranging from a few milligrams to over 10 kilograms.

To obtain better and accurate results in the readings, the strip sensor has to be calibrated first by exerting a force evenly applied across the surface area of the strip. This has to be done with a flat surfaced object; otherwise the readings sent by the sensor will be wrong. These types of sensors are low powered and have a linear output. For our project, the strip sensor is located about 20 feet from the E-Frog Game and whenever a player steps on it, the sensor changes from its normal state and sends a signal to the microcontroller and a buzzer sounds.

This is a simple sensor to implement since the strip goes on the floor and it can be easily moved around. The graph displayed below shows the corresponding force vs. the resistance of a 100 pound sensor. From the graph we can see that the resistance (in blue) decreases as the force applied increases. It is important to note that the conductance curve is linear therefore; it is useful when calibrating the strip-sensor. [Figure16]

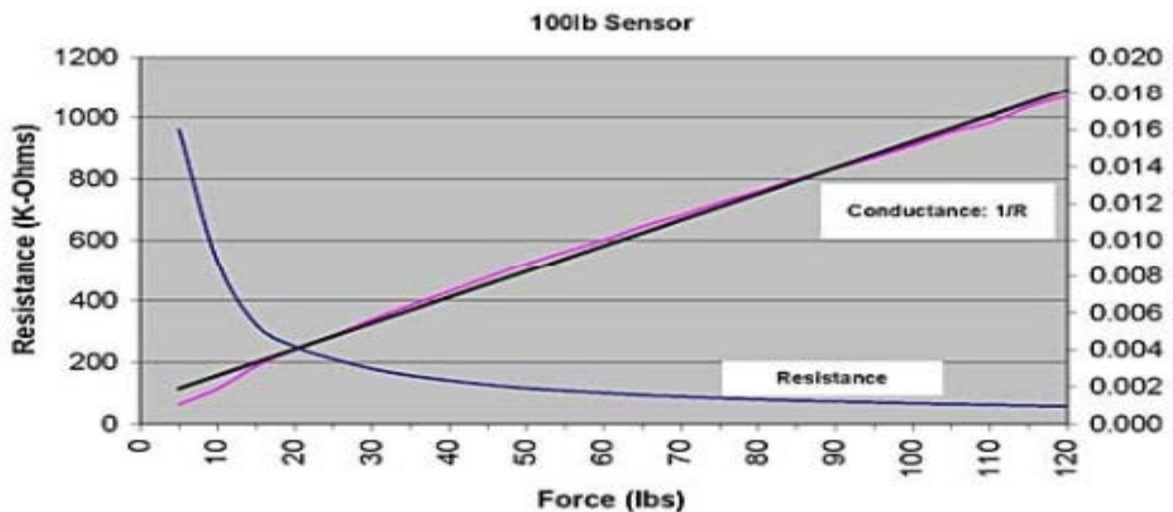


Figure16 Force vs. Resistance Plot

Threshold System Using an Infrared Distance Sensor

One of the options our group had for the use of a distance sensor was an infrared distance sensor. Infrared sensors are widely used today, from television remote controls to motion sensors that can detect when a person is waiting at you step door, for instance. Infrared sensors are manufactured by the millions and normally they are made to work at close range because sensors can't go through walls. There are several types of sensors: capacitive sensors, distance or proximity sensors, inductive sensors, and motion sensors, just to name a few. Infrared sensors work by sending out light beams made up of an invisible light. It consists of an emitter on one end, and a photoelectric detector on the other.

The sensor works with a system known as "triangulation"; this means that an infrared LED is continuously emitting light when no object is obstructing this beam of light. As soon as the desired target cuts off the light beam sent by the emitter, the light beam is reflected back to the receiver LED off the target at an angle after the contact is made. Therefore, determining the distance at which the target or object is located at. For our project, we need to know when a player crosses the threshold line at which the coin is to be thrown. If the player crosses the line, then a buzzer sounds alerting the player that the throw does not count as a valid throw.

As our group is looking for a cheap and easy way of implementing this system, the Sharp GP2Y0A21YK optoelectric device is certainly a good option. The way the sensor works is by sending out a specific amount of voltage depending on the distance where the target cuts off the light beam. The closer the target is to the light beam (excluding the first 10 cm or so, where the voltage readings are unstable) the stronger the signal, and the higher the voltage. As the distance from the target increases, the voltage dissipates and the signal is low. Then, the voltage that is picked up by the sensor is then transmitted to one of our microcontrollers, which in turn is coupled with our audio system which produces the sound. Also, it is of great importance that the target or object to be measured is white colored so that more accurate results are obtained (90% accuracy). The sensor readings for a gray or black colored object are less exact (around 18% and 10% respectively). For our purposes, this sensor could potentially suit our project since our concern for this part is focused only on the fact that a player cannot throw the coins from an undesignated distance.

Threshold Sensor Using a Transmitted Beam (Thru-Beam) Sensor

Photoelectric sensors are being used in a variety of ways because of their adaptability and high accuracy (these sensors can detect a target of 1 mm in diameter). Unlike the infrared photoelectric sensor mentioned in the past section

this sensor is composed of an emitter and a receiver, which are totally independent of one another; this technology, called “thru-beam”, separates the emitter and receiver into two separate housings. This sensor has many applications and is widely used in the industry. Some of its applications are: monitoring of production and packing lines, automotive parts handling, and door operation.

For instance, this type of mechanism is seen in garage doors. The emitter is placed at one end of the wall and the receiver is completely aligned to the emitter in other end in a way that the greatest possible amount of pulsed light from the transmitter reaches the receiver so that when any interruption of its light beam occurs, the state of the receiver changes and the garage door closes. There are two different operating modes for this type of sensor: dark operate mode and light operate. Dark operate mode consists in the energization of the load whenever the light from the emitter is absent from the receiver. Light operate mode consists in the energization of the load whenever the light from the emitter reaches the receiver. The desired operation mode can be configured before the sensor is used for the first time. For our E-Frog project this is an interesting idea because our group can use this sensor which has a high degree of accuracy and a long range compared to the infrared sensor from Sharp (Up to 12 feet) to create a limiting distance between the table and the player. The long range capabilities and the high limit of ambient light in this type of sensor catches our attention since it is likely that the players throwing the coins will not be throwing them from the same exact spot every time.

Every player may step more to the left or more to the right every time he / she plays the game. Also, the game is likely to me moved around from place to place, therefore changing ambient conditions. Additionally, the fact that this sensor is not controlled by voltage makes it probably easier to implement. A single beam of light cut off by the object gives a more accurate reading every time with a very low percentage of error. The figure below shows how the sensor works, first without a target, and then when a target cuts the beam off. [Figure17]

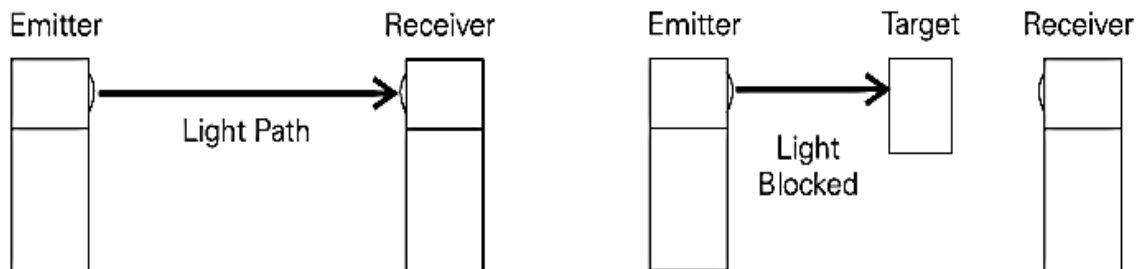


Figure17 Thru-Beam Sensor

3.2.6 Coin Counter Sensory System

Playing a game with friends is always a good method to seek relax and enjoyment from our daily routine. Convectional games have been always part in our life but laziness and the availability of new resources that made our life easier have changed our election in choosing the game to be played. Today's technology and advancement of new interactive have changed the way how we choose things, even the ones that helps us relaxing. If there's a reason why people changed from analog televisions to remote controlled TVs', or from chess and monopoly to interactive applications in cell phones is the convenience that these devices offer to the end-user; convenience that summarizes in "less work" for the person who uses them. Such reduction of work or effort can be, for example, the automatic counting operation that bowling places have in their screens.

The idea of incorporate a sensor that detects or counts the number of coins being used in the E-Frog Game started when we wanted to control the automatic passing of turns among the players. When Player 1 finishes his 10 attempts with the coins, the game is supposed to be smart enough to detect that the Player has finished all the available coins to be thrown, and to prevent cheating or miscounting, the microcontroller stops the addition of points when the Player is over with his turn. The E-Frog Game will enter in a minimum waiting period to allow the next player to collect the coins and get to position to start his turn without having to bother in pressing any buttons to jump to his turn. In order to complete this task, a sensory system needs to be implemented to detect the coins when they are being used by the current player. As we can understand, it becomes a little difficult to think about an accurate system that can do this task. We would like to add some kind of GPS sensor to each one of the coins and detect its position but it will not be practical at all since we will be overcomplicating the project and it will turn it into not useful and very expensive as well.

The objective in this portion of the project is create and incorporate a sensory system that can detect the number of coins thrown so it can jump automatically to the next player when there is no more to throw. To implement a system like this, we considered a few options that we will describe them in the following pages.

Due to the complexity of the specifications and requirements of the desired infrared barrier, many design approaches have been invented and modified, in an attempt to overcome as many of the problems presented by the design. Prototyping and early testing will determine which design is chosen for the final version of the Counting Detection System of coins.

One limiting factor in the design of the infrared barrier is known as the Nyquist Sampling Theorem. The Nyquist Theorem states that an analog signal may be fully reconstructed from sampled data if and only if the sampling frequency is at

least twice the frequency of the signal frequency. This theorem can be extrapolated to the infrared barrier application. In order to be sure a measured interruption of the barrier is valid, and not simply some form of noise or interference, at least two emitter/detector pairs must lie within a distance equal to the diameter of the bronze coin (for the purpose of this document, the diameter of a coin is 1.25"), as this will likely be the smallest diameter object to pass through the barrier in any circumstance.

A) One Sided System of Infrared lights

The single sided plane design was the first theoretically designed which satisfied all the requirements. In this design, the emitter/detector pairs will lie within the same unit. Emitters and detectors will be arranged within the unit in alternating pairs, in order to maximize the chances of detecting an object. The emitters must have a long enough range to reach from the surface of the floor to the standard height of a room ceiling (between eight to ten feet); with the emitted light remaining intense enough to be reflected back down to the detectors embedded in the unit by any object which enters the plane.

As stated before, theoretically, there must be at least two emitter/detector pairs within a distance of at most 1.25 inches to cover two beams of infrared light. This means there must fall at least 10 pairs per foot, but this approach is relative and inaccurate since the coins are not always going to be thrown in a flat position. Each player could throw the coins in any manner such as flat, rotating, flipping, and free style, therefore there will be no accurate way to detect the passing of these coins unless we will construct a single sided system of infrared lights that will be spaced apart as much as the thickness of each coin, which will be 0.25 inches. Such proposed array will involve the placement of at least 48 pairs of emitters/detectors per foot to cover all possible options to detect the passing of a coin through this barrier. Such barrier will have to be placed in front of the player around covering the area in which the player will throw the coins towards the game; the proposed width for this area will be 4 feet, which will be a total of at least 192 emitter/detector pairs. This approach will be possible to be implemented but not convenient for the amount of components involved in this system.

See [Figure18] below for a visual aid:

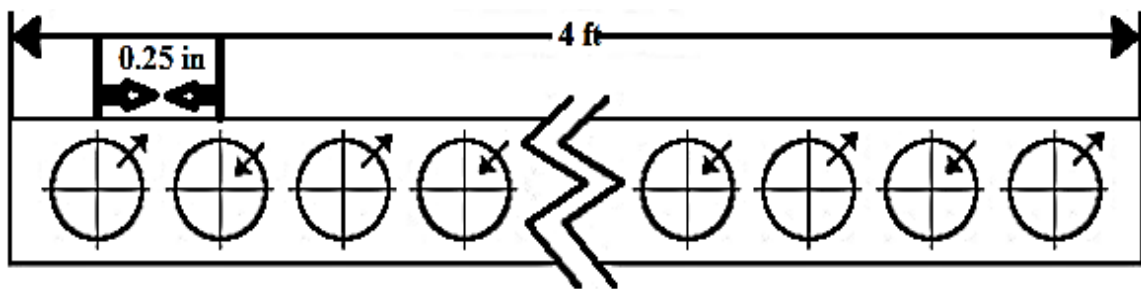


Figure18 Layout of emitters and detectors in single sided design

B) Building an Infrared Curtain Barrier to Detect Coins:

From the previously described approach, we started researching for more available options that will satisfy the desired purpose of detecting the coins when they are being thrown. The first idea right after we looked for more options was based on the security alarm systems that many houses have. The passive infrared sensor that security systems use to detect moving objects was the chosen sensor to be researched. These sensors operate based on the previously explained infrared technology, sending constant beams of infrared lights to the exterior up to certain distance and the system first scans the surrounding area, recording the localization of the present objects to don't get a false alarm that may activate the sensor. Then, the procedure or mode of operation of this security sensor gets activated when the moving object/individual reflects the infrared emitted light back to the detector unit, this detector will interpret this reflection as a positive input to the covered area.

One of the drawbacks from this sensor implementation is that the covered area is, most of the times, a conical coverage that will detect movement in a tridimensional area such as a room. This will represent a problem since we only need and want to detect the coins in a single plane. Researching through other options we found another infrared sensor that detects in curtain coverage, this will be like having an invisible curtain where the coins will pass by towards the table. Every time the coin passes the invisible curtain, the system will emit a signal to the microcontroller who's going to decode the signal interpreting it as a count of coin. Programming the microcontroller will come in a different section of this document.

The following [Figure19] illustrates a rough representation of the Sharp Wide Angle Measuring Sensor Unit implemented in our Project:

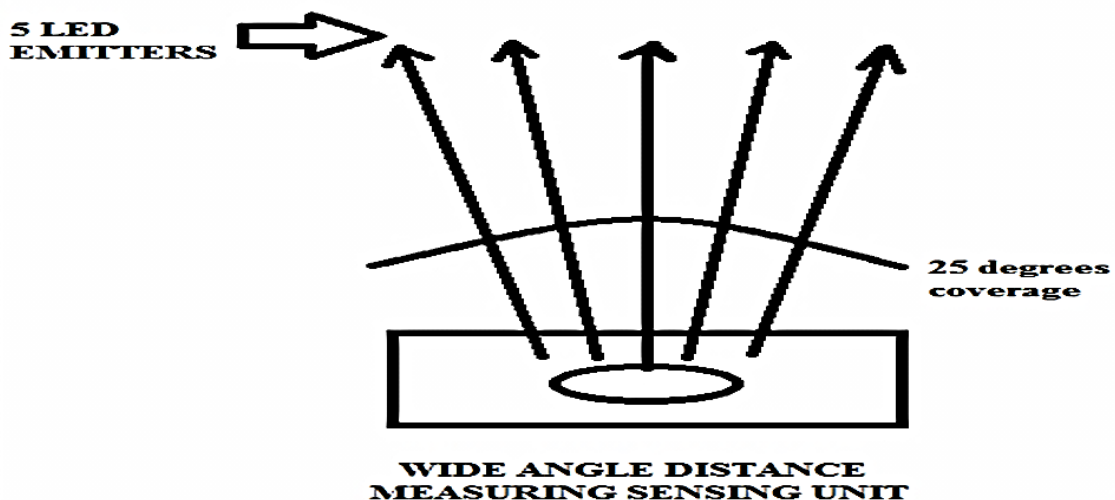


Figure19 Wide Angle

One of the proposed options that we found that complies with the idea of the curtain-type sensor is the wide angle measuring sensor unit made by Sharp. This sensor consists in the precise placement of 5 LED emitters within a sensory unit and a detector that is able to detect the reflection from any object that will lie within the spectral area emitted from the 5 LED emitters. The sensor unit will intercalate the operation of each one of the 5 lights turning them one by one at a very fast speed controlled by a built in oscillation circuit and LED drive circuit, together with other needed components such as the voltage regulator and the signal processing circuit.

C) Building a Customized System to Detect Coins:

In all the hardware designs detailed, functionality is roughly the same. Each design relies on detection of an object passing through multiple barriers or through the same barrier multiple times. Due to limitations in number of pins available on some microcontrollers, as well as other limitations, we thought in a different option that can do the same two main purposes of blocking the game from undesired inputs after the last coin has been thrown and, to automatically jump to the next player.

The concept of implementing a counting system for the coins, as mentioned before, is to prevent the following scenario: when Player 1 finish throwing his last coin, the other players will help picking up the coins that fell on the surroundings of the game and from the top of the table; but what if by mistake or purposely someone pushes a coin in a hole while is picking them up? If we wouldn't have the counting system, our game will count this input and will add it to the Player 1 counted points. Another reason to include this system to the project is simply to let the 'game' know when Player 2 is ready to start without the need of having to press a button to change the player every time.

One of the solutions that we came up with is that instead of having a system like the ones explained before with the infrared barriers sensors that will detect the coins as they go towards the game, we will have a system that will detect the coins as they are being taken out from a dispenser. In other words, we will create a dispenser that will count or detect the presence of the coins inside of it and it will send a signal when the last coin has been dispensed out so a timer in the microcontroller starts counting for the last coin before it blocks the game until the next player. Adding this timer feature will increase the level of emotion and entertainment to the game since the current player will only have a determined amount of time to throw the last coin before the program stops adding points.

One of the designs that we thought was to build up a pedestal from the floor to about three feet tall and it will end up in the coin dispenser body composed by an attractive 1.5 inches diameter pvc tube of about 8 inches long that it will have a

bigger open end on top that will serve as a funnel to put in the coins and will be placed or mounted in an incline position so the coins can go down by gravity.

At the lower end of the dispenser tube we will cut a semicircle slit where the body of the coins will be exposed to be easily picked up by the player. The trick or creative part of this dispenser will be that it will contain a pair of infrared emitter/detector led light having at the top end the photoelectric detector and at the bottom the infrared emitter diode laser. This pair will make fully contact when the last coin is removed from the dispenser, letting the light beam travel through the center body of the tubular dispenser and activating the signal of the photoelectric detector. This signal will alert the main game controller that the last coin has been removed and it will start a countdown timer of 45 seconds before it blocks any more inputs to the current game. By blocking the inputs we eliminate the chance of dropping by mistake or intentionally any illegal coins to a hole creating a false reading of points. After the coins have been collected for the next player, these will have to be put inside the funnel of the dispenser so the coins will start sliding down to the desired starting position and they will also block the light beam again. This new blockage of light will be interpreted by the microcontroller as the signal to jump to the next player without the need of having to press any mechanical button to change players.

In the following [Figure20] we can see the shape and functionality of the coin dispenser, a simple idea for a smart solution:

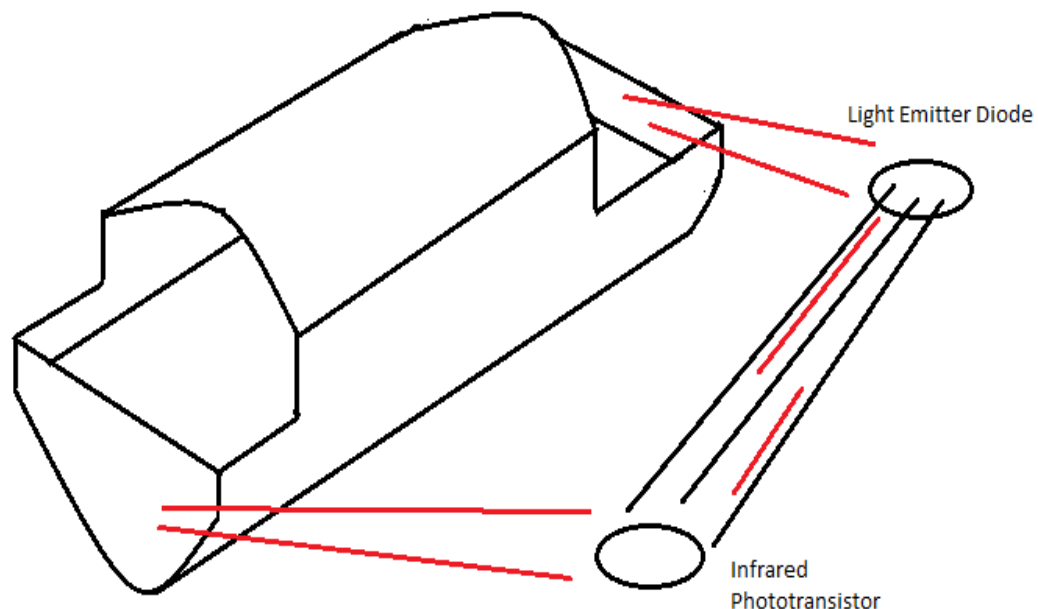


Figure20 Coin Dispenser design

3.2.7 Enable / Disable Switches

Enable/Disable Switches

The E-Frog Game consists of four inputs and four outputs, some of these have the option of being turn on or off since the main idea of this game is to make all players comfortable using the features they want depending of the situation or of the kind of player. For this reason, our group decided to offer the users an input compounded of a group of switches that active or inactive an input or output.

There are inputs that had this option; such as the threshold distance sensory system; this sensor can be turned on or off based on the decision of the user. This sensor consists in a limit line where the player is allowed to throw the tokens to the table box, if the player passes this line getting closer to the table box (easier to throw coins inside the holes and frog's mouth), an alarm will be activated, so the person will have to step back and throw the coins from the allowed point. Some people will not like the idea to hear an alarm in the middle of the game since they are just having fun or practicing, or they are probably not playing a tournament or a competition. For this kind of situation or player, we decided to have this feature optional for on or off.

On the other hand, there are also some users that get more motivated in to the game when features of sounds are enrolled in it since it creates a more challenging game. For this kind of player, the best choice will be to turn on this threshold distance sensor and enjoy this feature of distance limit to throw the coins to the table box from the specific point (closest distance from table is 15ft).The second input is the coin counter sensor; basically this sensor detects the number of coins that have been thrown to the table box. In order to control that each player is throwing the same amount of coins and to keep consistence in the game, this sensor had to be activated. Sometimes, the game can be used just for practicing, so the user may turn on or off this sensor and will have to manually select the next player to throw since the automatic function has been disabled.

There are also some outputs that have this benefit of being use or not. There are three of them: the visual light animation output; this feature consists of having some animated flash lights that function during a specific situation. For example, if a person score the highest shot points 400 (coin in the frog's mouth) there is a light that corresponds with this situation showing the word "FROG" in the screen making the game more exciting and fun. Also, if a player tries to cheat using more coins or throwing a coin so close, or maybe changing position of the coin for higher score, there is an alarm or a bad noise that activates to tell the players that something was not correctly used during that turn.

This feature makes the game fair and consistent. Other users do not want to have this feature ON since there are just two or three players, and they can tell if someone cheats easily, so they may choose to turn this sensor off to save battery and to do not get annoyed with those noises.

The second output that has this optional on off switch is the Audio Effects system. This output consists in different sounds that activate corresponding to the situation of the game. For example, when a player throws the token inside the frog's mouth (maximum points per turn), the person gets 400 points that will be announced with a very loud voice saying "SAPO" accompanied with lights and sounds and signs like "GREAT JOB". This feature makes the game more exciting and challenging for the other players to score also the highest turn points 400. There are users that will prefer to just see lights or signs of the amount of points obtained, so they can choose to have this switch off and do not worry about hearing those announcing voices.

Finally, the third output that has this advantage of being able or disable is the decorative light system. This system is based in some lights that are ON during the game making it more attractive to the player's eyes such as lights in each hole of the box table and maybe a green light around the bronze frog. All these lights just decorated our game and make it look prettier.

This option of using these two inputs and three outputs as preference or need, allows the game to save battery since for instance; the players may play outdoor and they want the game last longer or there is not possible battery charge at that time.

We have decided to incorporate these great features in the E-Frog Game having the opportunity to able or disable them using on/off switches. Our group has searched about different kind of switches to be compared and finally select the best option according to our project goals. The following are the possible options of switches we can implement in the coin throwing game.

DIP SWITCHES

Dual in-line package (DIP) is a mechanic-electric switch that is composed in a group of small switches integrated in a circuit board with equal shape like a chip. These switches has the two options of ON (1) or OFF (0), and let the user configure a circuit board for a specific kind of application.

Some of the types of dip switches are the slide, rocker and rotary. The rocker and slide are arrays of (SPST) single pole, single throw links that have the choice of ON or OFF and allows selecting a one-bit binary value. The switches values can be taken as one number; for instance, one byte is equivalent to 256 combinations that eight switches offer. On the other hand, 128 combinations are offered by the seven switches which allow picking an ASCII digit. We could implement these kinds of switches in our E-FROG game because of their specific

design easy to incorporate into circuits, inexpensive price, and also because they have the option to be checked at a brief look with the system power off. Also, another advantage is that they can be easy and fast to change with not parts to lose. For instance, the rotary dip switch is a possible option for our project since allows us to select one of the inputs or outputs we want to turn on or off depending of the user. [Figure21, 22]

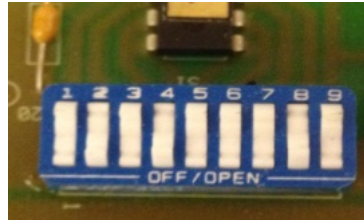


Figure21 Dip switch sample

This switch SPDT has three ends that may be linked one and to the other. For instance, in figure below the switch is connecting the white terminal with the brown one since it is located diagonally to the lever.



Figure22 SPST switch

This switch SPST just switches one wire because it has just a pole. It utilized to connect and disconnected the wire ON/OFF or one pole with just one wire.

The Double Pole Double Throw switch has two switches that function on the same time. They have a common linker and a normally open and close contact. These contacts rout in several manners corresponding to the position. The temporary push ON switch is a button that works temporarily when it is push on. This means that it works just when is being pressed down.

3.2.8 Display Screen Counter

1. Display Screen Counter

One of the E-FROG game outputs is the display counter, so group number 6 has decided to do research in two inexpensive and common displays; the LCD and the LED display.

LCD AND LED DISPLAYS

Like most of the games, the E-Frog Game is based in points obtained from the coin throwing. This electronic game selects the winner depending of his/her points based of the addition of the points per round. In order to display those scores, we need a screen with a maximum of 400 points per turn and 4000 per round. For this reason, our screen must be able to represent this maximum score; which enrolled a total of four digits. Also, the E-Frog Game requires a screen in order to display the number of players besides their corresponding scores per turn and the total points per round. Our group found information about different kind of displays to select the best option to be incorporated in the system of the game. The following are some of the ones we searched about.

LIGHT CRYSTAL DISPLAY (LCD)

LCD is a very thin screen most used for small computers like screens of laptops, notebooks, and flat panel monitors. This electronic technology consists in reflection of liquid crystal segments that changes depending of the applied amount of volts. This display is created by a display grid of passive or active matrix (TFT), and its image is made by enlacing a substance reactive electrically between two electrodes. Adding or reducing the current the color of the substance varies. This display uses low power because the screen is based on blocking light and not emitting it.

Monochrome LCD's

This type of technology is the cheap and the data transmission is simple with this type of technology and connections are maintained to a minimum. Monochrome LCD's work by using a small amount of data sent from the microcontroller to the LCD to display information. There are different types of monochromes screens that can be used and as well the configuration used for these types of screens. Some of the options considered are the use of a character LCD and the use of a numerical LCD. Character LCD's has more flexibility for use in this type of application but it also uses more data. A numerical LCD only shows numbers and thus uses less data.

Monochrome Seven Segment LCD Displays

These screens are also known as panel displays and they go from two to eight digits. Because of the low power and cost of the seven segment display, it is excellent to display text on mobile objects. Also, they are designed for high temperature and humidity scenarios

Color LCD's

The color LCDs function by having the capacity to show a large range of colors. Different LCD's give different bitrates for data that could be as minimum as 8-bits to as maximum as 24 bits. The bit rate is determines the number of colors the LCD is able to display by an exponential equation, which is like the binary equations for computers. An 8-bit LCD will have the ability to show 256 distinct colors with a 24-bit LCD giving 16777216 different colors. Modern color LCD's do not use the 8-bit configuration as this does not provide pleasurable viewing experience. With this in mind the least amount of bits used by a color LCD that will still provide a pleasurable viewing experience is 16-bit. The order of bits in a color LCD is to find what color each pixel will be. The pixel has three different sub pixels that make up the pixel.

The sub pixels are of color red, green, and blue which will in the industry is called RGB display. In a 16-bit configuration, 5-bits are used for red, 6-bits for green, and 5-bits for blue. This results in different shades of the three colors, with red and blue having 32 shades, and green having 64 shades of color; it results in about 65K colors. This configuration will have an easier load for the microcontroller. The advantages of Monochrome LCDs are the less power consumption the need of less components and logic, easier design, less likely to get programming wrong, and cheap price. The disadvantages are not as good to look at and contrast in sunlight will not be as great as the Color LCD. On the other hand, the advantages of color LCD are better viewing experience, can implement more, and the disadvantages is battery consumption.

LIGHT EMITER DIODE DISPLAY (LED):

This is a screen that consists in a group of diodes that together create many luminous dots named pixel which are made by the main colors, red, blue, and red which combined can make white color. This display emits light depends of the electrons flow from cathode to anode in the LED. They are two kinds of LED displays; conventional and surface mounted device panels. LED, gas plasma, and LCD screens are thinner than CRT technology and LCD uses less power than gas displays and LED since the block the light. The [Figure23] below is an example of an LED 7-segment display unit:



Figure23 7 Segment LED Display

There are two kinds of LED displays; the conventional screen which utilizes discrete LEDs and the surface-mounted device screen. The majority of indoor and outdoor displays are constructed with discrete LEDs called individually mounted LEDs. It consists in a collection of blue, red, and green diodes that make a full color pixel commonly in shape of square. The pixels are separated uniformly and measured from middle to middle for a total resolution.

LED or LCD display?

LCD screens utilize fluorescent lights which have short life duration. Around five years the display will start deteriorate since they are not built for replacement of missing lights. For instance, a fluorescent bulb can last around 10,000 hours when a LED bulb can last longer than 10000 hours after it has been used.

The LEDs screens can be lighter and thinner than LCDs screens since LED displays do not utilizes fluorescent bulbs. Also fluorescents bulbs are heavier, larger, and make the LCD displays bigger than LED displays. The side lit LED screens are thinner and lighter than LED backlit screens, so they are easier to achieve when it need to be move from place to place.

LED screens leaves LCD screens behind talking about picture quality in motion blurring and color intensity. LCD displays sometimes have problems of blurring image in times of extreme motions. About color quality, another benefit of LED's screens is that they use red color (diodes blue and green which combine the light in precise ratios to the actual color. This is the reason why LED screens have larger color range and contrast ratio than LCD screens. In addition, because LCD displays just try to block the light coming by closing pixels, the contrast for dark colors as well blacks is more true and crunchy in LED displays, so they reduce the correct diodes to develop dark pictures.

After a research about LED and LCD displays, group number six has decided to implement the LED display in the E-FROG game for several reasons. The main reasons of selecting LED screens are the long life duration lights, the thinner presentation, and the light weight that these screens offer. Since our project design is a portable game, these convenience characteristics make the E-FROG game a very durable and comfortable portable game. Another important reason of choosing this kind of display is the high imaging quality; LED screens have a large color scale and have no issues with color intensity and motion blurring like LCD ones.

3.2.10 Audio Effects System

Sound is an important part of perception and an excellent way for a microcontroller to interact with humans (and other animals). The audio feature was an attractive idea that came to the project to make it more interesting and attractive for the players since it will play an important role to grab the players'

attention throughout the whole game. It can be simple and inexpensive to implement. A microcontroller could do anything from generate tones, play music, or even output sampled sounds such as a dog bark. The goals are to generate good quality sound which can be used to play simple tones, stored music, sampled sound, and even act as a MIDI synthesizer. The idea is to implement a microcontroller that yells “SAPO” when someone hits the coin inside the frog’s mouth as well as other tones and sounds as the game takes place in different scenarios and situations.

Many embedded microcontroller applications require generation of analog signals. Sometimes an integrated or stand-alone digital-to-analog converter (DAC) is used for this purpose. However, Pulse Width Modulation (PWM) signals can often be used for generating the required analog signals. PWM signals can be used to create both dc and ac analog signals.

Enabling new functionalities in microcontroller applications, including crystal clear audio with playback and recording capabilities, Texas Instruments Incorporated introduced the C5000™ ultra-low-power digital signal processor (DSP)-based Audio Capacitive Touch Booster Pack. This new Audio Capacitive Touch Booster Pack is a plug-in board for the \$4.30 MSP430™ Launchpad development kit, and it is the first solution from TI where a DSP is solely controlled by a microcontroller which allows designers to add audio and other real-time features to their system without DSP programming experience. The Booster Pack is ideal for low-power applications that use record and playback audio functions such as MP3 players, home automation and industrial applications. The following [Figure24] shows a block diagram of this capacitive.

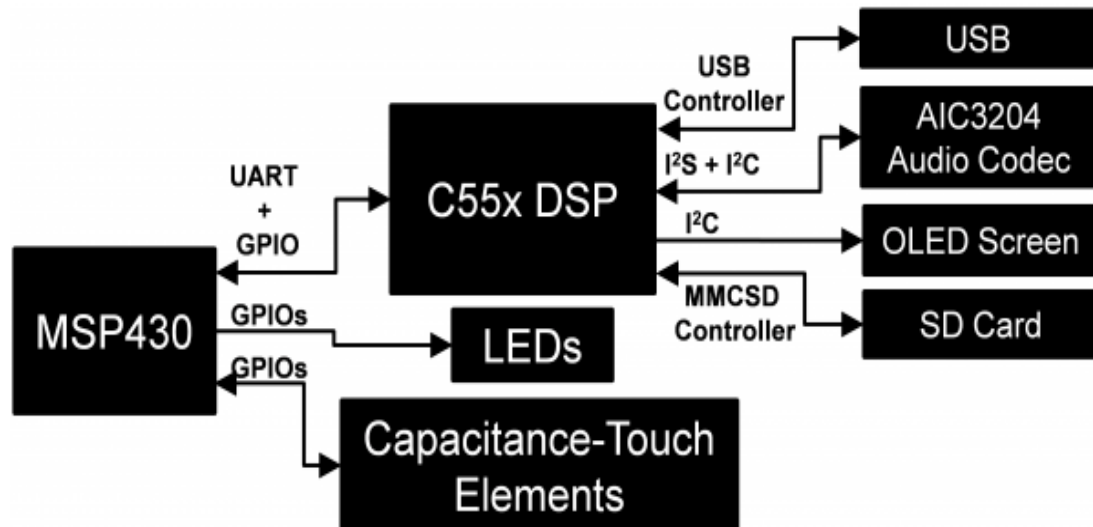


Figure24 Audio Capacitive Touch Booster Pack Block Diagram

3.2.11 Decorative Light System

Decorative Light System

The E-Frog game will have a light system to give the game a more attractive look. Also, because the game can be played under any conditions, a lighting system is very helpful so that the players can interact even more with the game. First, the table would have a set of lights in front of it to create a more pleasurable environment to play the game. Also, each hole will have a set of lights embedded within a panel of temporary score light, allowing the player to see the hole scored in his/her current throw in case of playing in the dark. Additionally, the game would have a set of lights by the display screen where the score and player number are located. The group has agreed that the light system must have a set of animations for a given situation during the game. The first animation would occur when the game is turned on together with a message displayed at the seven-segment display. The light system matrices will power off and on and will run random sequences clockwise or counterclockwise and as we will decide to get them programmed.

LEDS

The first option the group has to build the lighting system is using LED's. They are used in a variety of applications in the industry and since the beginning of the 1980's have been the type most used. A LED (Light Emitting Diode) uses what is known as the SSL (Solid-State Lighting) technology. Instead of emitting light from a vacuum like a regular light bulb, it emits light from a piece of solid matter. This matter is called a semiconductor. As we know, a semiconductor has a positively charged component (holes), and a negatively charged component (free electrons). When an electric charge strikes the semiconductor, it activates the flow of electrons from positive to negative. When electrons move around this semiconductor then light is emitted. With this brief definition in mind, there are advantages of using LED's over other technologies. For the E-Frog game the main advantage of using LED's is that they are energy efficient. In addition, a LED light lasts much more than any conventional light bulb.

This option is very attractive for our project since saving in energy translates to saving in power, which would make our lighting system very economic and of excellent quality. Since a 24" by 24" table is going to be used where the coins land, and each LED is going to be placed at a distance of 2" from one another, 36 LED bulbs are needed to cover this area. The next area is around the seven segment display. This area is 12" by 4" and will be covered by 12 LED light bulbs. Also, there will be LED's embedded into the table. The E-Frog will have 20 holes and each hole will have 4 light bulbs around it, therefore 80 LED's will be needed. In total the E-Frog decorative light system will have around 128 LED's. In addition, the group could use flexible "strip LED's" to implement the light system. They are widely used today mainly in the automotive industry and can be

found in different presentations. Even though they could be of great help for our project since only a few connections and wiring would have to be made, they are very expensive.

A-Series Incandescent Light Bulbs

Another option the group has discussed is using regular light bulbs like the ones used at our homes. The idea is to have several light bulbs on top of the table so that they illuminate the holes. For the project, there will be the need to try to customize them somehow in order for them to fit in our table. Two hooks would be used to attach several wires to the base of the game, so that the light bulbs can be placed at the top looking down to light the panel. For the lights at the edges of the holes, the same procedure will be done as stated above. Incandescent light bulbs have been used for many years, but as time goes on, their use has been decreasing. They are very expensive and consume a lot of energy. Most of the energy consumed by the bulb is given off as heat, causing its performance to be low.

T-Series Light Bulbs

We have the option of using the T-series light bulbs, which are relatively small and can fit in our game table easily. The T series comes in various sizes and colors, and they could reduce the number of light bulbs and wiring needed (only a few bulbs would be needed for each edge of the table). The light bulbs that we might need are the following: The T-12, T-8, T5, and T-2. The T-12 is a 1.5 inch wide light bulb (about 38.1 mm). Even though this bulb is relatively inexpensive, it has a poor lumen per watt ratio. One solution could be to use many of these bulbs to balance out the lack of output. The T-8 is a very similar to the T-12 bulb, but it is smaller at only 1 inch wide (25.4mm) with better efficiency, thus making it the preferred bulb in many projects. The T-5 bulb is yet smaller (13 mm in diameter) and uses less Watts per Lumen. The T-2 series bulb is the smallest bulb, and it is the best option. Its diameter (7 mm.) makes it very easy to allocate in our table. The disadvantage of the T-2 bulbs is their unavailability because they are relatively new. Even though the T-series bulbs are very good in saving power and costs, there have been documented defects that we would have to take into account if these bulbs are used in the project.

3.3 System Integration Options

Before determining the correct system integration for the game, an evaluation of the E-Frog Game must be made. The different systems of the project must be analyzed. The E-Frog Game will detect the coins when they enter the holes and will do different responses such as displaying the points-outputs, sounds, and lights. If a coin enters a hole, the corresponding LED laser/receiver pair will be shut off and a specific point would be awarded to the player who has thrown the token. Once the number of throws is complete the state of the game changes

and the next player now becomes the thrower changing automatically once the coin counter system detects that there are no more coins to be thrown. These actions continue until all players have finished certain pre-arranged amount of games.

When the systems of the E-Frog Game are generally defined for the purpose of the game there are only appear to be two integration systems that could potentially handle those requirements. The two integration system considered for this project are the distributed load of tasks architecture and the event-driven architecture. Distributed load of tasks integration allows for the computing of different processes concurrently. This would be great because there would be no need for a central controller unit to subordinate all the functions. Each microcontroller can be programmed for its specific task. But we found out that there is an issue with this method. The issue is a lack of control for systems that aren't needed at a specific time which leads to unnecessary power consumption and errors that would occur in terms of coin detection occurring at infrared systems and others sensors. This would not be a problem if the systems of the players were controlled by a central unit.

The alternative architecture for the E-Frog Game is the event-driven model. This model was considered because the game the E-Frog itself is a turn-based game. Due to this realization it would only make sense to have an integration that would utilize a method that would coincide with the mechanics of the game. Event-driven integration of course is triggered by an event. In the case of the E-Frog Game the event is the coin being detected by the infrared system when entering through the holes. If the infrared system detects that the coin thrown has got it into a hole, it will communicate to the central processor and will again wait for another event to occur, either another coin entering though another hole or simply by timing out after the last coin has been use so it can 'decide' to jump to the next player.

When comparing these two interacting systems, the distributed load of tasks architecture and the event-driven architecture, it is clear that the event-driven interacting system is the one best suited for the E-Frog Game. Distributed the tasks allows for each individual system to operate itself, hence the functions of the E-Frog Game can be equally distributed amongst the different processes. Its lack of control for the integration of the sub-systems is the issue with the using the distributed computing architecture. The event-driven interacting system better suits the E-Frog Game because of the turn-based game play. With the event-driven architecture system it allows for a single control unit to keep a certain order with the different sub-systems when it comes to their processes.

Power System

Power distribution is important for all of the components to work properly and give accurate data. There are components that need power as well various mechanisms with which to supply them. The idea is to power the components off

of power supply of 120. All of the components run on different supply voltages. Therefore, once a power supply is chosen there needs to be circuitry to supply the parts with the correct supply voltages. There are a couple ways to do this. One way is to use linear regulators. They will step down the voltage to a nominal supply voltage. Another option is to use operational amplifiers and resistors to step down the gain. This will achieve the same thing.

Since efficiency is important in power supplies an efficiency of 85% or higher is preferred. As long as the maximum the voltage can be supplied from each of the sources then the rest of the mechanisms can be powered. The power supply system will consist of the main power supply unit and all buses, cables, and wires responsible for delivering power to the different subsystems involved inside of the entire design of the E-Frog Game. The main power supply will be the one of the unit receiving power from an outside source delivering power at various voltages. The specifications for the main power supply unit are as follows:

- Able to run continually for a long period of time under AC operation.
- Equipped with functional unlimited use surge protection system and ground fault interruption.
- Intake power from a 120VAC 60Hz.
- Power delivered at 12VDC, 5VDC, and 3.3VDC.

The main power supply is one of the responsible for taking in power from a main source of 120VAC 60Hz and safely converting it into many rails of other voltages such as recharging the battery at 12V, or 5V and 3.3V for the system circuitry. Other circuitry can be incorporated to create any other forms of power used for subsystems by using voltage regulators and other similar devices to control the source power present.

Batteries

A variety of considerations need to be taken into account when selecting a battery solution for the E-Frog Game. A research of the different technologies will help us decide what the best option that we can choose for the design of the game. In consideration are the lithium-ion batteries, for their ample range usage in consumer electronics such as laptops, wet-cell lead acid batteries for their uses in off-grid reliability power backup systems, and valve regulated lead acid batteries, or sealed batteries for their convenience and use in motor sport vehicles and domestic emergency power applications.

Lead-acid Batteries

There are two types of lead-acid batteries. The first type is a cranking or starting battery and it is used primarily in motor vehicles for delivering high amperage levels for short periods of time at the vehicle's start. Physically, this battery contains a higher density of thin lead plates that assist it in performing this

function. The second kind is a deep-cycle lead acid battery. The deep cycle variety has thicker plates, which allow it to withstand the physical demands of a charge-discharge cycle. This type of battery is used in marine and electric vehicle applications. Each kind of lead acid-battery has three categories.

Wet cell lead acid batteries have lead plates impregnated with a sulfuric acid electrolyte solution. As the battery discharges, the sulfuric acid in the solution bonds to the lead plates. As it recharges, the sulfuric acid bonds are pry apart and the acid returns to the solution. This type of lead acid battery has been around since 1859, and is still very popular. It is also the least expensive of the lead acid technologies.

The second kind of lead acid technology is the gel cell. This type of battery would be ideal for the E-Frog game because it requires little to no maintenance. Because of the gel interior, this type of battery is more resistant to vibration-heavy applications. The tradeoff is that this battery is more sensitive to overcharge situations, and has a lower charging voltage than most lead acid batteries.

The third subcategory is the AGM (Absorbent Glass Mat). Depending on the orientation and geometry of the cells internal to the battery, it can be a traditional AGM or a —spiral wound. Good for deep-cycle applications, these batteries can be discharged and recharged quickly and feature the highest energy density of the lead acid batteries. The AGMs are also the most expensive.

Another consideration is whether the battery is a deep-cycle battery or a starter battery. Starter batteries are common and are used in vehicles. Deep cycle batteries are used in backup power systems and in RVs and boats as power so the most ideal for the E-Frog Game will be this type of battery.

The power supply unit will be mounted underneath the center of the table box where proper ventilation can be obtained from the surroundings but without interfering with the normal operation and aesthetic of the game. The power supply chosen will be of the type of a regular power supply of a desktop computer since it already has all different ranges of power needed. The dimensions of the power supply unit being used are standard ATX computer power supply unit dimensions. Those dimensions are 6 inches wide, 3.5 inches tall, 5.5 inches deep. The power system will supply with power to all the main systems. The main systems are, the infrared matrix barrier system (hole-detection sensor system) located under-table, the emitter laser infrared diodes, LED display system, light decoration and animation system, coin detector sensor system, the sound animation system, and the main microcontroller unit. The power supply will feature one and only one 120 volts alternating current, in case of running the system under battery power, the 12V battery added to the system will take care of the power supply system of the E-Frog Game.

4.0 Design

Microcontroller Design

In order to reach this modular design, most of the subsystems of the E-Frog Game operate on their own microcontroller unit. The microcontroller of choice for most subsystems of the project is the ATMEGA 2560. This controller was chosen because of its excellent capabilities and competitiveness that has when compared to others microcontroller units. The 2560 is the chip in the Arduino Mega development board. This means that there are many open source libraries that work directly with the chip which greatly reduced the amount of coding we needed for the project. There are also many examples of projects using this chip. Another perk of using this chip is that we programmed them with another Arduino connected via USB to the computer with the source code, and we did not have to buy an expensive In System Programming (ISP) device. As we researched in the previous section, we had a few options to consider to our project among the vast variety of microcontrollers units available in the market.

One of the three options that we considered to choose and use in our project was the MSP430. Mixed Signal Processor microcontroller is a 16 bit CPU microcontroller designed by Texas Instrument. This is a very inexpensive device and it is designed in particular for embedded applications and low power consumption. The MSP430 is implemented especially for battery power uses. Another microcontroller option that group number six had was the Atmel ATMEGA 128 that is the microcontroller unit that Arduino uses in his very well-known and popular product Arduino ONE which is the development board that for years has been chosen and preferred by millions of developers around the world for many electronics projects and other similar applications.

Arduino has a very wide variety of Library codes to share online and it served as a great help for our project. One of the reasons that made us to choose this microcontroller was the software program that uses that it is very similar to C and C++. Since Arduino products use their own system of software, we have had to learn and practice a lot about this language.

We also considered the option of choosing another microcontroller unit made by the company Microchip, its PIC16F57 was another great candidate to accomplish the requirements that we set for our project. With all the attributes that the PIC microcontroller has such as the low-power consumption feature, i2c communication system with others microcontrollers and the low price of this product made it a big competitor to the chosen microcontroller. Unfortunately, we had to decide not to use this Microchip MCU for personal preferences, lack of a rich library to use it as a guide in our project, and for not being familiarized with the product as much as we are with the ATMEGA2560.

Some of the reasons of choosing this microcontroller were the high performance, low power, and the number of pins since we just used 45 of the 86. The clock speed and voltage were compatible with what the project required. About the ATMEGA2560 programming, this microcontroller was programmed with the Arduino software (The ATmega2560 on the Arduino Mega comes with a boot loader that allowed us to upload new code to it without the use of an external hardware programmer. We could also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header. The main reason why we selected this microcontroller was the programming language that this chip requires. Since we are not computer engineer people, and we just had an introductory class in programming, we had to do our research about the programming language of each microcontroller in order to select the more suitable one for our previous knowledge. Arduino programming language was the closest to C language then we decided to use the ATMEGA2560 that uses the Arduino platform.

The Arduino Mega2560 has a resettable poly fuse that protected the USB of our computer ports from over current and shorts. The maximum length and width of the Mega2560 PCB are 4.0 and 2.1 inches correspondingly, with the USB connector and power jack prolonging away from the former dimension. The ARDUINO UNO was initially chosen to control the functions of the subsystems for the previously mentioned reasons and for its affordability, the ATMEGA2560 controller was chosen because it was needed by the group to connect all the subsystems into one brain. The ATMEGA2560 was made as based processor, and it functions to keep track of the overall game state, as well as coordinating events of each subsystem.

To control all the different subsystems that we have in the E-Frog Game we had to research and choose a main brain to handle them efficiently that is why we chose to have a single central controller unit. This unit, unlike the rest of the controllers in the system, is the ATMEGA2560 controller, a more powerful controller that can receive, process, and handle more inputs and outputs simultaneously to make it operate as we would like it to be. The ATMEGA2560 has 54/86 general purpose I/O lines. We had used 45 pins of this 86 pins as follow: 20 pins for our light panel, 8 pins for our sound chip, 3 pins for our display and the two 8x8 led matrices, 9 pins for our nine NPN phototransistors (receivers), 1 pin for our coin dispenser, and 1 pin for a threshold sensor. This is how we combined our four inputs as well our four outputs in the game.

In general, the ATMEGA2560 officially unrestricted by ATMEL, is an eight bit integrated circuit chip of low power CMOS established on the AVR architecture improved RISC architecture. By executing powerful instructions in a single clock cycle, the atmega2560 achieves throughputs approaching 1 MIPS per MHz allowing the designer of the system to improve processing speed and power consumption.

User Interface System Design

The user interface system is a very important subsystem of the E-Frog game because it enables the players to interact with the game in an easy way. The goal of this system is to control the different subsystems of the game by the use of SPST and slide switches. In case the player wants to disable some of the features of the game he / she may able to do so. This system will be made of five switches. The switches will be charged with the following tasks: a switch to turn on or off the game, a reset switch to start a new game, a switch to disable / enable the sound system, a switch to disable / enable the threshold system, and a switch to disable / enable the coin dispenser system. The on / off button will turn on the game. This switch will be located on the low back of the table box; not visible when playing to avoid hitting it with the coins thrown. We have found this part in the market for very low price just \$3.15, and its specifications are 10A for 277 VAC and 20A for 125 VAC. This is the 3900 Metal Lever Switch single and double pole with options of three positions circuits. The dimensions of this switch are 58.6mm by 21.5mm and 31mm. The following picture [Figure 25] shows the SPST switch and its dimensions.

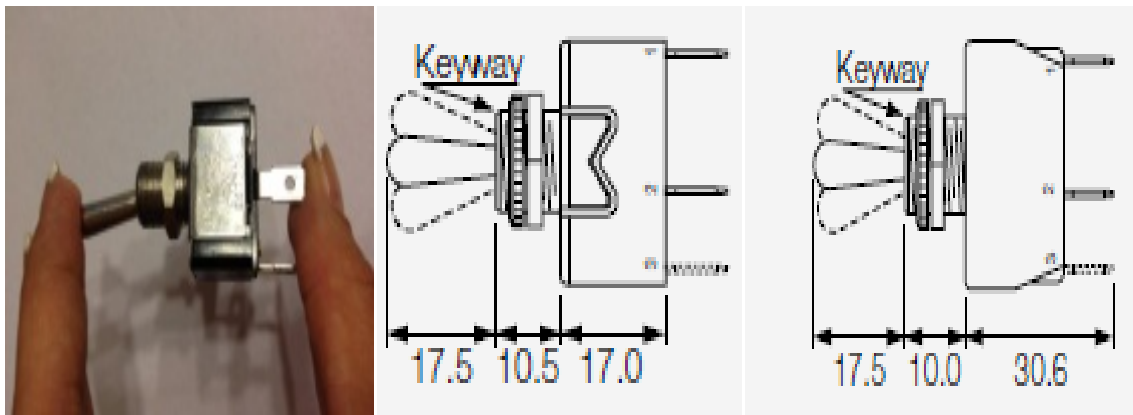


Figure 25 ON/OFF switch and its dimensions.

Once the game is turned on, a reset button will be pressed to let the game begin. One round arcade switch will be used as the reset button. The button will be located on the side of the table box; visible to the players, but they also will be hidden to avoid being hit by any coin. This button will be labeled as RESET (red) and will reset the game. The cost of this switch was \$3.95. The limit switch that this type of button uses has switching currents of 10A to 21A at 250 VAC. Its length is 2.125 inches and its width is 0.875 inches.

The following picture [Figure 26] shows the chosen arcade buttons for the user control panel and its dimensions taken from the datasheet:

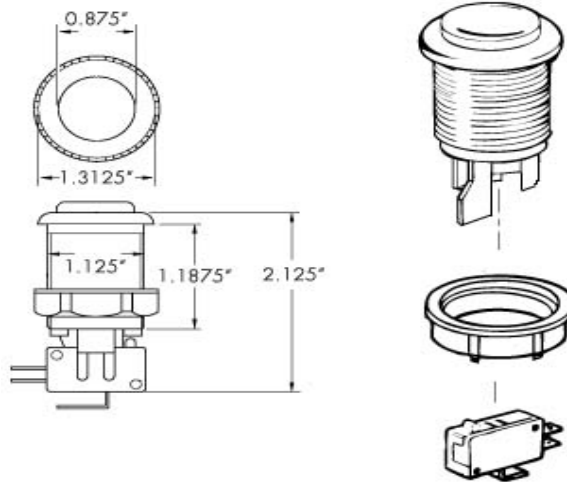


Figure 26 Arcade button and dimensions

If the players want to disable one or all of the systems mentioned above he / she will have the capabilities to do it. . This is a really a great choice this game offers since some people would prefer to play with not all the features. For instance, if a player does not want to play with the audio system ON because he or she cannot focus throwing coins, the person has the great option of turning it OFF. Also, a player who does not like flashing lights while playing has the option of having this feature ON or OFF. Group number six has found a good slide switch part in DigiKey market and manufactured by CW Industries for low price of \$0.67 each, so it would be a total of \$2.68 for four switches. Its part manufacturer number is GF-1123-0009, and it is a SPST ON/OFF switching current of 11A EYELET at 125 VAC. This part has a length of 1.33 inches and a width of 0.55 inches, and the height of the button is 0.344 inches.

The following picture [Figure 27] shows the chosen slide buttons for the user control panel and its dimensions taken from the datasheet:

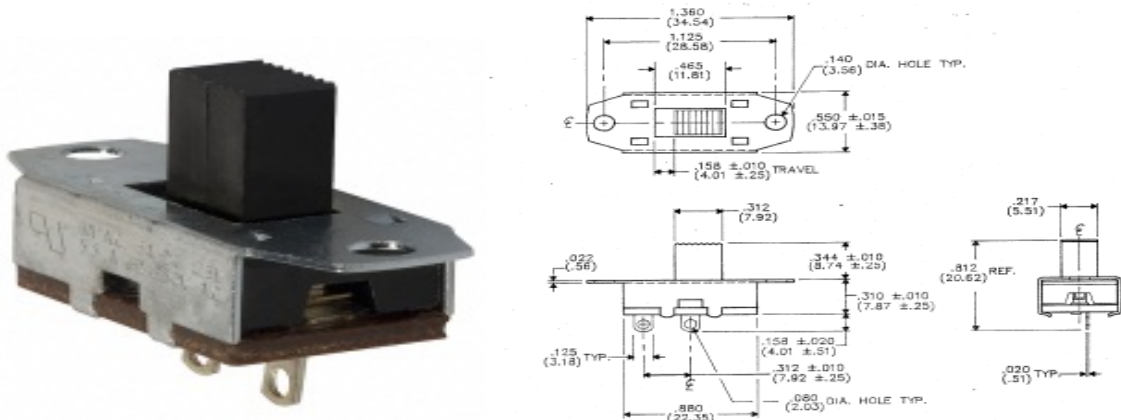


Figure 27 Slide Switch SPST and dimensions

By accomplishing these tasks the game will flow smoothly and it will be less prone to error. The switches have been tested more than 300 times to ensure the capabilities of the slide switch. Its 20 gr. sensitivity weight is of great importance because we have to be certain that the microcontroller always receives the correct signal. The wiring of the button is relatively simple. The wiring for the addition of the switches is relatively simple. After the systems are wired, a cut in the power line and in the signal line or ground line will be enough to attach a switch. The switch will be wired to one of the microcontroller pins and to the power supply. Since we are going to have a 12V power supply and the microcontroller only takes in about 5.0 V, we will need a voltage regulator to account for the change in voltage. The following [Figure 28] is a depiction of the push button when turning on / off the game.

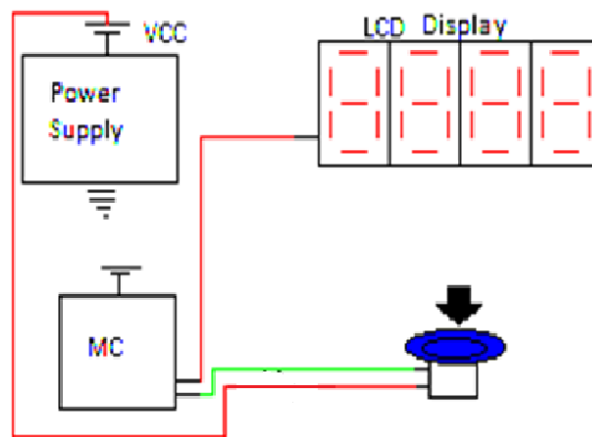


Figure28 ON/OFF Push Button

When power is given to the system, voltage from the power supply will go through the switch. The switch will serve as a bridge between the power supplied and the microcontroller. As figure above shows when the button is pressed once, the game and display turn on. Below, the next [Figure 29] shows the seven segment display after the button is pushed for a second time to turn off the game.

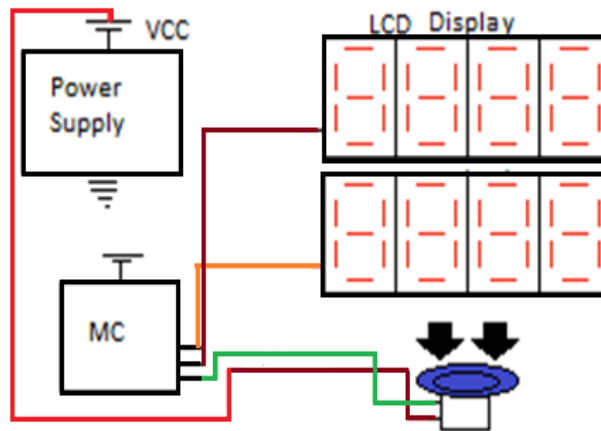


Figure29 LED Display button pushed

To enable / disable the systems of the game we will use a Single Pole Single Throw (SPST) and slide switches. In this part of the design the on / off switch will turn on and off the desired system before, after, or during the game. Most of the SPST and slide switches available work only on two states, on and off. The SPST switch chosen to accomplish this part is the ADG751. This switch works in only two different positions, on and off. It is used only to change the supplied power to our circuit. Both the dip and limit switches have been tested over 100 times to ensure its reliability and capabilities. The SPST switch has the following specifications: 3db signal bandwidth 300MHZ, it is low on resistance; about 15Ω , and it is low on power consumption, typically less than $0.01 \mu W$. Its voltage range, from 1.2V to 5.5V makes it ideal for our project.

The "start" button will allow the players to begin the game. After the game is turned on and the described components above are also on (coin dispenser stacked of coins, sensors ready, and threshold distance sensor, among others) each player can start throwing the coins. At this stage it is very important to check the power supplied because any abrupt changes in power could damage the components of the game. Finally, a PCL-L button will handle the resetting of the game. This button will be in charge of resetting the game after it ends. This part will reset the number of players and the amount of points to zero. As mentioned before, the user interface system is a very important system that will make the game flow smoothly, therefore making it more enjoyable for everyone playing it.

As seen in previous drawings of the game, the table has twenty holes and a frog. For this system, each button has a 3.65cm diameter, but the body of the button within thread is 3.50 cm. So, holes of 3.49 cm in diameter need to be drilled at one side of the table, preferably at the side edge of the table. The four switches and the button will be wired together allowing the line of power to go through each one at the same time and making them available as soon as the game is turned on. This wired button and switches will go directly to the microcontroller pins and from there to their respective seven segment display depending on the situation of the game. In addition, a de-bounce circuit will be implemented in the system. A de-bounce input means that the system will check twice for any given push to make sure the push is valid and to make sure that the microcontroller does not read a push as multiple presses.

Hole-Detection Sensory System Design:

The E-Frog Game was our Senior Design final project but in order to be approved we had to contemplate different aspects to make it challenging enough to keep us entertained and busy until the end of the two semester period of this final class. As far as the system to detect the coins when they fall into one of the twenty holes we had, as previously mentioned before, different proposed ideas and solutions.

The mechanical contact limit switch approach was one of the systems proposed and partially developed to implement it to the project. We constructed a sample of this approach by using a mechanical micro switch made by Omron Electronics Inc. (Manufacturer part #: V-10G2-1C24-K) that we had in our tool box as a spare part from previous projects and we added to the lever a piece of plastic of 3 inches in diameter and we glue it to the lever with JB Weld 5-minute compound. After we added the lever, we connected two wires to the terminals, one to the common terminal, and another one to the normally open terminal. After getting the micro switch ready, we constructed a wooden box to simulate one of the holes from the game; this box was used in general to test each one of the proposed system of coin detection. Since the mechanical micro switch needs to be mounted on an inclined platform constructed under the hole, we added a piece of wood that fitted this requirement, then we applied a 12 volts source to the common end of the mechanical micro switch and connected the normally open end of the switch to our Fluke 73 III multimeter to record the passage of voltage when the switch is temporary hit by the bronze coin. Following, we started the experiment and threw coins inside our single-hole box and we noticed that the passing of voltage was successful as expected, even though the limit switch was hit temporary and for a very short period of time by the coin, the Fluke multimeter was able to record and measure the voltage while the coin hit the constructed flap that was positioned in an angle to allow the coin to slide down to the bottom of the box.

Unfortunately, we found a couple of glitches with this proposed approach. Out of the approximately 50 times that we tested this micro switch option, the coin did not hit the flap hard enough as expected one time to create a pulse signal, and also we had the scenario when the coin bounced on the flap creating a double pulsation or signal from a single throw and this happened three times. These two problem scenarios concerned us in our design because the goal of this Senior Design Project was to construct a game where there is no allowance for failure or misreading. We as new engineers getting ready to change and improve our nation, strive to get perfect results in every job that we performed and these four failing throws were not acceptable to what we expected to get for the hole-detection system. Also, we had to spend some extra building time getting all the parts customized to create the 'perfect' fall for the coins by building the inclined ramp, screwing the micro switch to the ramp, and running a lot of more wires to implement this approach to each one of the 20 holes that our game has. Of course, the cost of materials was higher since we need more parts.

On the other hand, we also considered to have a hole-detection sensory system build with a fairly new electronic device called a Distance Measuring Sensor Unit, made by Sharp Electronics (Part #: GP2Y0A21YK0F). This piece of equipment has a valuable and interesting importance to nowadays technology because thanks to the infrared emission light system that uses and its built-in infrared photo detector, together with the circuitry required for the operation of this sensor (signal processing unit, LED drive circuit, voltage regulator, oscillation circuit, and output circuit) made possible that a single infrared emitter diode and a receiver

diode work in a way that by triangulation of the infrared beam with the obstructing object that reflects back the infrared light to the receiver diode and this one picks up this response and precisely measures the angle of reflection between the reflected beam and the position of the receiver diode. Once the angle has been detected, the signal processing unit takes place from angular conversion to the voltage regulation; in other words, it puts out a signal in a voltage form that has an inversely proportional relation to the distance from the object. When the device was tested at 5 volts for VCC, the output signal at 10 cm from the sensor device put out a signal of approximate 2.2 volts, and as the interfering object was moved away from the sensor, the output voltage was decreasing.

Knowing the maximum limit of reading of the sensor is at 80 cm, we also measured the output voltage at this range, obtaining a reading slightly below 0.5 volts. We recorded different measurements at other different distances between the valid range of reading and we confirmed the functionality of this device. Once we tested the proper operation of this device, we mounted and adapted to the sample box that we previously built for our previous micro switch system. The only modification that we had to do to test this device in our testing holed-box was the addition of an extra hole for a better testing and be able to record different distances and values. We placed the distance measuring sensor unit about two inches below the holes, inside the box pointing in a direction where all the holes are sensed by the laser spectrum. When we tested the sensor with the coins to detect them as they fall down to the hole, we used again the digital multimeter Fluke 73 III to record the voltage data outputted from the sensor.

When we manually moved the object slowly down to the hole, the multimeter was able to read the outputs from the sensor but, unfortunately when we let the coin fall through the hole by gravity, the pulse signal was so fast to the ability of the multimeter to capture an accurate reading that we had to find a solution or an alternative way to read the output from the sensor. One of the solutions thought to notice the reading of this sensor was the usage of an oscilloscope but, since we did the testing at home, we had to find an alternate way to measure or at least confirm the operation of this sensor when the coins pass through the holes, so we found an analog voltmeter where the needle marks the measured voltage level. Every time the coin passed through the any of the holes while connected to the analog voltmeter, the marking needle jumped or flicker indicating an output recorded; since the microcontroller reads information accurately thousands of times faster than an analog voltmeter, the distance sensor approach works great with the microcontroller.

After a couple of hours testing this option, we noticed that the readings recorded by our digital multimeter decreased in voltage when measured at the same distance, this brought up a potential problem to our game. Knowing that each hole measures three inches, we had to configure or program the microcontroller to assign values to a certain range of readings captured from the distance sensor. For example, the edge of the first hole was located at 5 inches from the location of the sensor, then we know that the range of reading corresponding to

that hole takes as many values as needed throughout the diameter of such hole so, our program is set to recognize the hole by the detection of the voltage readings from that hole. The problem arises at the time when the battery starts draining down from its initial value and varies the output of the sensor decreasing it and unfortunately, our program that was initially set up to recognize a hole by the range of voltage conditioned at the time of the encoding will be off from the pre-set value. Having this problem presented, we were not be able to rely in this system if the output reading is not constant from beginning to end. At this point we found out that the operation and proper application of this device is precise and useful to detect objects at certain range of distance and/or for short periods of time while the supply voltage remains fairly constant but not to be used for longer periods of time if we want to rely in a very consistent reading at the same distance.

Continuing with our research for the proper equipment that we used in our hole-detection sensor system, we found another type of technology to apply it to our E-Frog Game, and this is the approach of the emitter and receiver diodes. Using infrared technology was a challenging field to step onto it, but the expected results seemed to be very trustful, so we took the defiance of doing it. Initially we thought in putting a couple of emitter and receiver diodes per hole to detect the passage of a coin though it.

The way we thought it was going to work was by having the led on all times and the receiver diodes detecting the light constantly so when a coin passes through a hole it temporary cuts the emission of light emitted to the receiver interpreting this interception as a valid input. But, as we started developing the model for our first hole, we discovered that by placing many infrared emitters under the holed top we created a tremendous diffusion of infrared signals going in all directions and it did not matter if one spectrum was going to be temporary blocked by the coin, the corresponding paired receiver still pick up a light from another emitting diode.

One of the solutions thought for this upcoming problem was the idea of building separators in between holes to allow the spectrum and reflectance of an emitter diode was directed to only one receiver diode but at the same time this solution resulted in the increase of labor, materials, and effort to develop it. It was our job to look for a simple and more effective way of doing things and as engineers we have to always look for better ways to improve existing solutions. Based on the same approach of infrared technology, we thought in the usage of infrared laser as a source of emission light. By doing this we had to place fewer amounts of construction materials and electrical components because we set up the lasers in a way that they shoot their beams from side to side covering more than one hole at the same time.

Light Emitters

Unlike the limited number of useable light detectors, there is a wide variety of light emitters that we could use for optical through-the-air communications. The specific designed system depends much more on the type of light source used than on the light detector. We chose the light source based on the type of information that needs to be transmitted and the distance we wished cover to reach the optical receiver. In all cases the light source must be modulated (usually turned on and off or varied in intensity) to transmit information. In the development of the E-Frog Game we decided that the maximum range or distance to be needed to use this type of technology was at most the width of the constructed table which is 2 feet. At such a short distance like this we did not really need to have a powerful light emitter because pretty much any infrared light source is capable to reach this desired distance.

Laser Diode

A laser diode, also known as an injection laser or diode laser, is a semiconductor device that produces coherent radiation (in which the waves are all at the same frequency and phase) in the visible or infrared (IR) spectrum when current passes through it. Laser diodes are used in optical fiber systems, compact disc (CD) players, laser printers, remote-control devices, and intrusion detection systems. Laser diodes differ from conventional lasers, such as the helium-neon (He-Ne), ruby, and gas types, in several ways.

What Does a Light Detector Do?

Usually the transmitter simply turns the light source on and off. To decode the information from the light pulses, some type of light detector was employed. The detector's job is to convert the light signals, collected at the receiver, into electrical signals. In the E-Frog Game the role that the desired light detector has is to send an active-low signal to the microcontroller unit when an obstruction occurs in the beam path, this obstruction passes the thrown coin. The electrical signals produced by the detector's optical energy to electrical energy conversion were much easier to demodulate than pure light signals.

Although light is a form of energy, it is the intensity or power of the light that determines its strength. Therefore, the real job of the light detector is to convert light power into electrical power, independent of the energy of the transmitted light pulses. This relationship also implies that the conversion is independent of the duration of the light pulses used. This is an important concept and is taken advantaged of in many of the systems that follow.

The Silicon PIN Photo Diode

We may be aware of many kinds of light detectors, such as a "photo transistor", "photo cells" and "photo resistors" but most PIN photodiodes are made from silicon and as shown on the figure below, they have specific response curves. Look carefully at the curve. Note that the device is most sensitive to the near infrared wavelengths at about 900 nanometers. Also notice that the device's response falls off sharply beyond 1000 nanometers, but has a more gradual slope toward the shorter wavelengths, including the entire visible portion of the spectrum. In addition, note that the device's response drops to about $\frac{1}{2}$ its peak at the visible red wavelength (640 nanometers). Fortunately, most IR light emitting diodes (LEDs) and infrared lasers do indeed emit light at or near the 900nm peak, making them ideal optical transmitters of information. But since our goal was not to create a wide spectrum or combination of signals to emit any kind of communication signal, a lower emission rate and at the same time cheaper technology was utilized in our project because we were only interested in a constant infrared beam that activates a photo receiver diode all times until the coin will interrupt a couple of beams. [Figure 30]

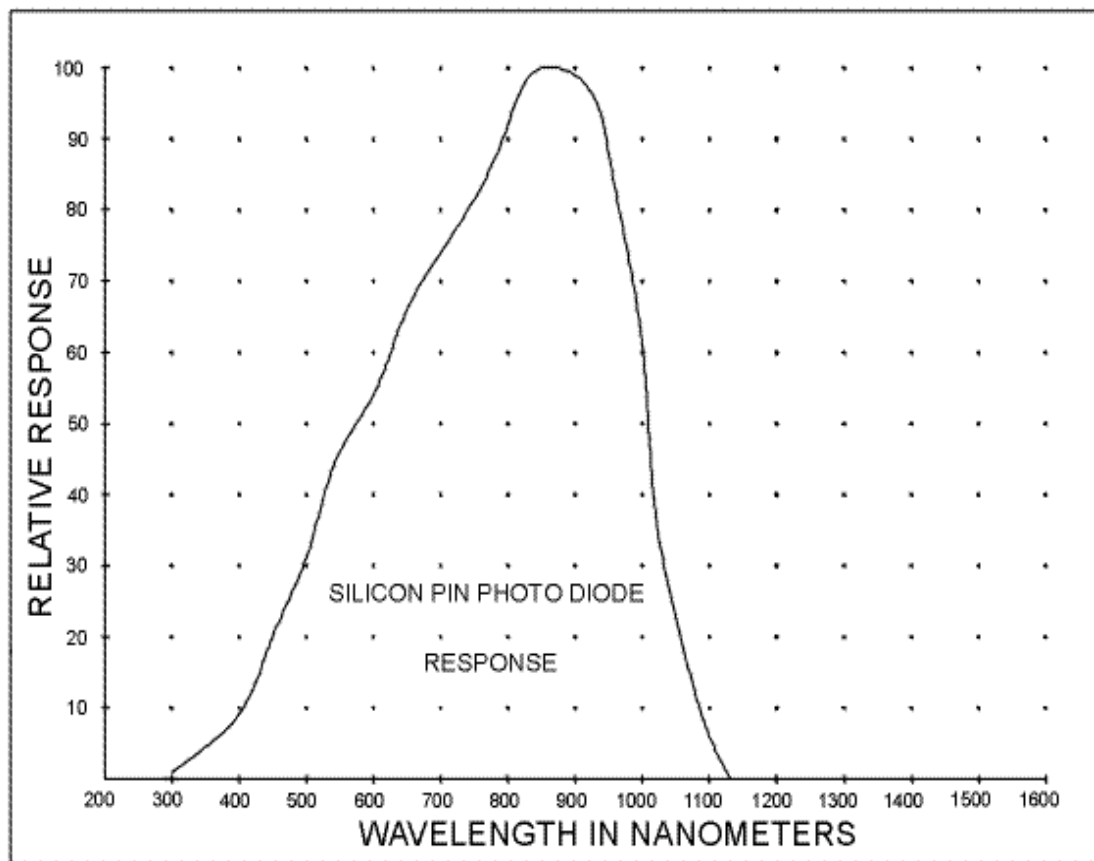


Figure 30 Wavelength (nm)

The PIN photo detector behaves very much like a small solar cell or solar battery that converts light energy into electrical energy. Like solar cells, the PIN photodiode produces a voltage (about 0.5v) in response to light and also generates a current proportional to the intensity of the light striking it.

Constructing the Demo:

Lasers are definitely not an inexpensive device to find available in the electronics market and the prices of these can go up to thousands of dollars depending on the precision and application that we may need them for. Our group had to do an extensive research to find the proper and most affordable laser to use in our project and we found on the eBay page a package of 10 diodes of 5mWatts and emitting an infrared laser beam at 650 nanometer wavelength at the affordable price of \$ 24.99 with shipping included from a Chinese vendor.

Another important part that we had to obtain to do the simulation was the photodiode receiver and we decided to go to the local Radio Shack store and get a couple of pairs of photodiodes emitters and receivers with a part number: 276-142 at a price of \$3.69. This pair worked perfectly fine when bench tested on our breadboard with a supply voltage of 3 Volts, a step-down resistor of 100 ohms to lower the voltage to the desired 1.5 volts to operate the emitter diode at its optimal conditions and on the other diode, on the receiver diode we connected the collector lead to the positive side of our testing power supply and we connected the emitter lead of the receiver diode to the digital multimeter to detect the reading that the receiver diode emits at the time that the infrared signal is activated. Using the same receiver diode we prepared a bench test simulation using this time the laser infrared pointer to be the source of emitting light that took care of the activation of the receiver diode and we discovered that the signal emitted by a laser was good enough to activate the circuit built.

Having the bench test successfully developed and tested we started implementing and building the laser/receiver system in our previously used testing holed box. We placed one laser at the beginning of the first hole and pointing towards the other end of the box, subsequently, we installed the receiving diode at the end of the laser beam, aligning it to the center of emission to be precised when actively shoot the light beam. One of the testing devices that we had to use again to be able to pick up on the light signal when broken by an obstruction at normal operating speeds (about normal falling by gravity) was the analog voltmeter, similar to the previous test that we did with the distance sensor, we proceeded to do the same test; we connected the voltmeter to the receiver diode and read the output with the needle indicating the voltage when light was pointing to it and when the light was cut by the passing of the coin, the needle flickered indicating a change in the reading. Analyzing this approach and testing it for hours we noticed the great amount of reliability that this system offered to our game but we just had to implement it and make it better and better with new ideas and a better placement of the components.

When we used a single beam to cover the detection of the hole from the perspective of an 'X' axis we notice that there were some possible areas where the coin fell and was not detected by the laser beam. A coin measures about 2 inches in diameter but that does not necessarily mean that the coin will always lay out flat through the hole. When the coin was tossed to the hole, it entered in a flipping movement or vertically and passed through the hole without being detected by the laser. As engineers our task is to always overcome any obstacles when they are present in our lives, so we decided to shoot two beams per row, spaced out enough to be able to cover all possible positions within the holes plane. Following the same concept of the two lasers in the x-axis, we repeated the same concept in the y-axis and this was done to detect the coins as well, with the difference that by doing this procedure we were able to write a code where it interpreted the laser breakage in a manner that was convenient for us. Writing an instruction where we conditioned the program that if there were at least two infrared receiver detectors that temporary had their beams cut by a coin it would tell the program exactly on to which of the twenty holes the coin entered. The following [Figure 31] is a diagram that represents the position of the sensors on the rails, some dimensions and the Cartesian plane that the matrix array of lasers will form in the E-Frog Game:

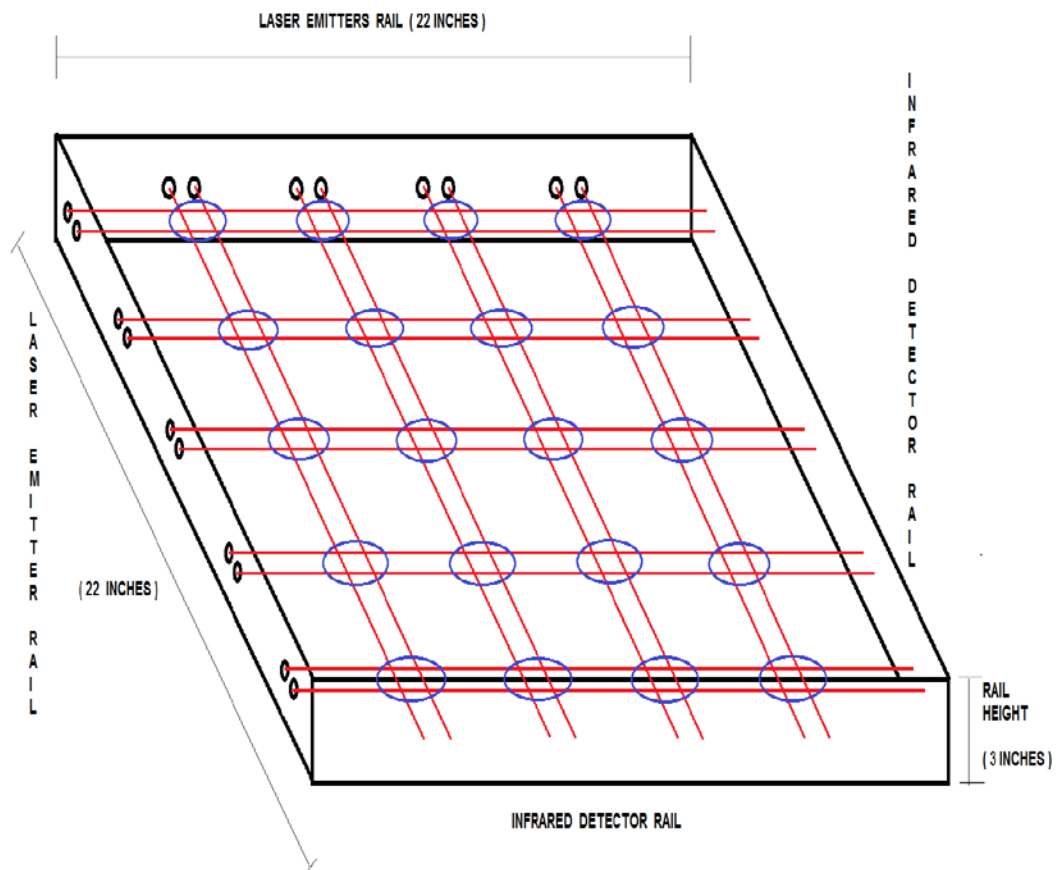


Figure31 Matrix table design of the Laser Hole-Detection System illustration

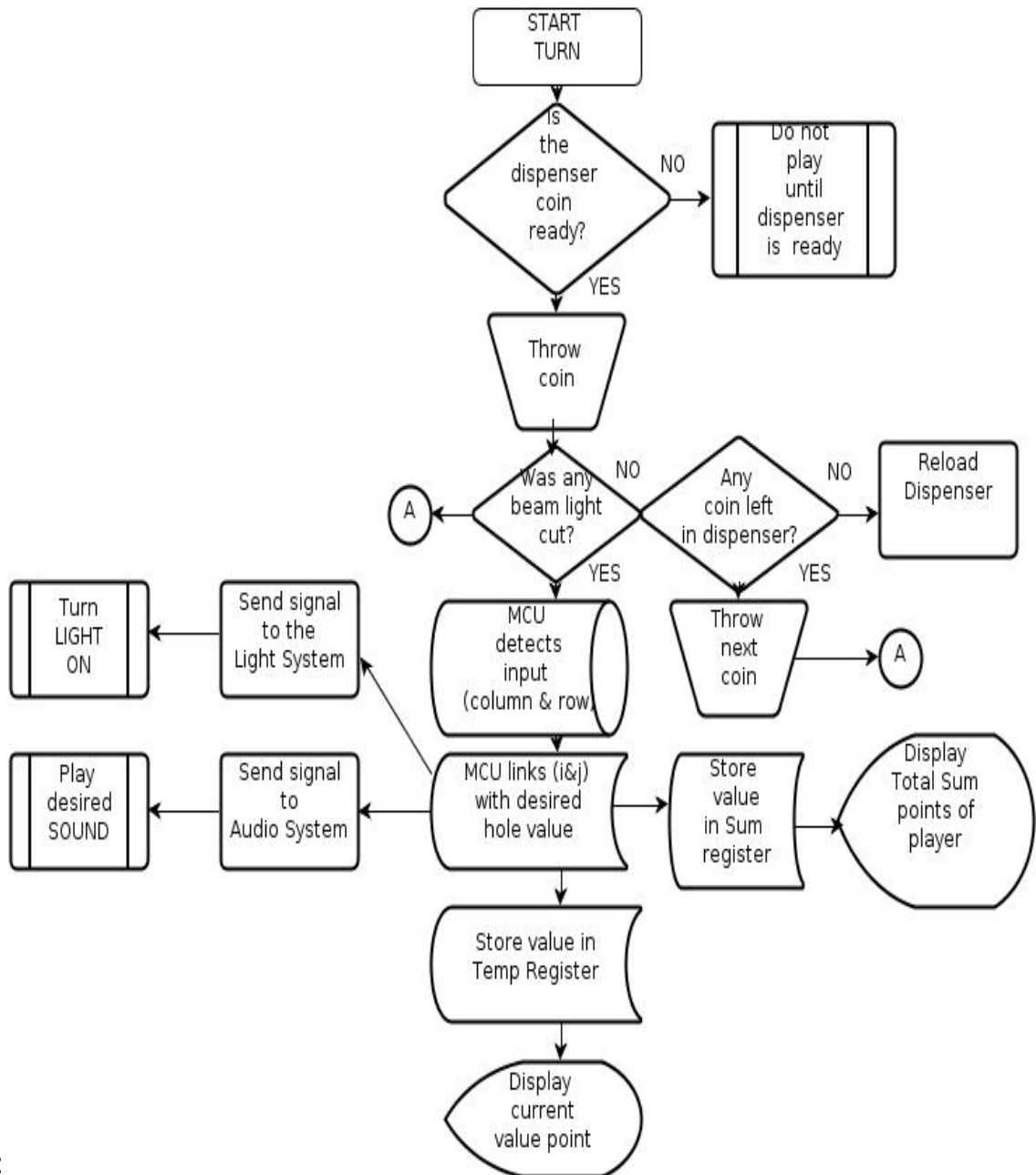
Matrix-type Hole Detection System Array

The final design as we were explaining along these lines that we chose to integrate it to our E-Frog was a combination of the previously presented infrared technologies already described above. In the matrix-type hole-detection system array, there are four aluminum rails running along the edges of the E-Frog Game underneath the table. These rails are hidden from the players view because they are going to be located after the coins pass the top holes of the table, approximately 2 inches from the bottom surface of the table; this way the barrier exists in a plane just below the entrance of the coins used to play the game. The rails have a width of 2x2" which was determined during prototyping and was minimized depending on the size of the parts embedded into the rail but we can roughly give an estimate that the length of each one of these four of these rails was slightly less than the length of the game table, since the table is a squared table, approximately the length of each one of these four rails 22 inches but, as far as width we determined the correct dimension during prototyping to hold the lasers and receiver diodes.

Each of these rails were set up much in the same way as the emitter/detector rail pairs described in the testing box experiment that we did. For each dimension of the table (length and width), the rails running parallel to each other work exactly as the rail pairs from above do; emitter lasers are always shining on the detectors, and the system detecting objects passing through the barrier by the lack of illumination of the detectors. Using this configuration, the holed-table under surface was calculated because for its design matrix into measurable units, very similar to a Cartesian plane. When a coin is thrown by a player, as long as it actually goes through any table hole, the system is able to determine the exact position on the table where the coin landed. This is achieved by knowing the location of the detector(s) which measured the presence of an object, in both dimensions (x and y). The intersection of the rays which can be imaginarily drawn across each pair of emitter lasers and detectors are the position in which the coin crossed the hole and subsequently the barrier. Distinguishing a coin entering through the matrix table design worked in a way that the system tries to detect where it crossed the plane.

It attempts to detect the object passing through the plane again within a period of time. This design does have one advantage over the others in that the exact location of the object passing through the barrier, in both dimensions, is able to be determined. Each one of the infrared lasers emitter diodes is positive powered with its corresponding voltage required which is 3.3 volts but since our common power supply is going to be about 5 volts, we needed need to add a voltage regulator to the system. One of the advantages from powering the lasers is that since they need to be ON all times, there is no need to split up the voltage source to each one of the eighteen laser diodes. We just needed to make sure that the voltage regulator chosen to drive all the lasers was powerful enough to handle the current required to keep them activated without failing or fading.

Taking in consideration that the maximum operating current that each one of the laser emitter diodes utilize is 60 mA, we will need to find a voltage regulator that can handle at least $60 \text{ mA} * 18 \text{ sensors} = 1080 \text{ mA} + 20\% \text{ more (just to be safe)} = 1.3 \text{ A}$. After doing the corresponding research we found that the Voltage Regulator made by Texas Instruments P#: LM317TFS-ND it will satisfy with the required conditions. The following flowchart is a representation of the hardware and software interpretation of the operation of this Hole-Sensor Detection System: [Flow Chart 1]



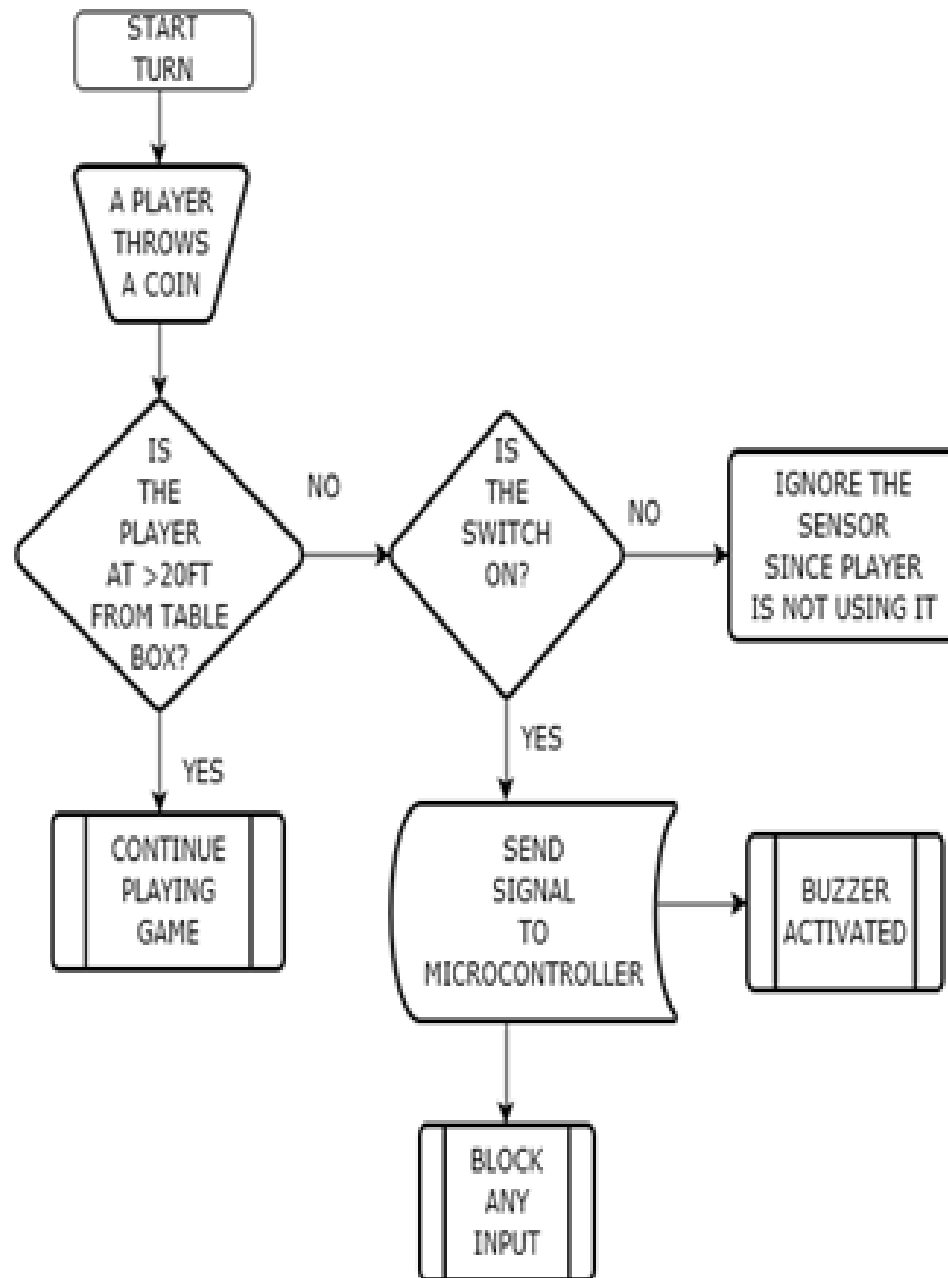
Flow Chart 1 Hole Sensor System

Threshold Sensor System

As it was discussed in the previous pages, the group needed a threshold sensor in order to keep a player at a minimum distance from the table, from which he / she could throw the ten coins. The E-Frog game had what is called a “touching” or “strip” sensor which is widely used in the industry. In the need for a reliable sensor, the design incorporated the strip sensor BRAUN 31221A. One of the main reasons this sensor was chosen was because of its simplicity, and its rather easy implementation into our game. Unlike a photo-detector sensor which has an emitter and a receiver, this sensor has a flat surface which can be placed on the floor just like a floor mat. This is a very smart solution to our problem because we did not want players playing the game walking close by and tripping with big sensors lying on the ground.

The way the sensor works is by having a decrease in resistance when there is an applied force. The sensor can detect a force ranging from a few grams to over 150kg. Also, the sensor has a range of ± 15 to ± 25 tolerance depending on the force applied. The actuator or switch which turns on the strip is the player’s foot. Since this is a very sensitive sensor, we did not need to be so accurate with the actual force; as a result as soon as the player touched the strip, we got a reading that went to our microcontroller. In addition, once the strip sensor was pressed, an alarm-like or buzzer sounded to alert the player that the threshold line had been crossed. In the case that the buzzer is activated, any input to the game will be automatically blocked. When the strip actuator or “switch” is pressed, the microcontroller will sense a voltage in its pin, and as result it will respond in a desired manner.

The dimensions of the sensor are 24” by 0.7”. We want the sensor to be at least 15 feet away from the table and we want to couple the wiring with the coin dispenser wiring towards the microcontroller. The coin dispenser and the threshold system work together in the sense that for the sensor to work we always need to have at least one coin placed in the coin dispenser. In the case when no coins are stacked, the strip sensor will not work. As soon as the microcontroller picks up this voltage, the buzzer will sound. The system will have a de-bouncing circuit included because we want to make sure that we have valid pushes. For instance, if the strip sensor is interrupted by a person who is not playing the game and steps on it by accident, the buzzer will not sound. On the other hand, when a player steps on the sensor, we will use programming routines to detect when a certain amount of current is sent, thus determining if the push is valid. Figure [Flowchart 2] below shows a flow chart with the description of the game with its respective functions.



Flowchart 2 Threshold Distance System

Coin Dispenser System:

As we have previously discussed in the previous sections, the E-Frog Game was composed by quite a few different systems that integrating them all together into the same brain make up a unique and innovative game for all the players who have enjoyed this game before. Among the systems that are present in the design, we wanted to include a system that keeps track of the coins tossed

throughout the game per player so the selection or moving to the next player happens automatically without the need of selecting the change of players every time the current player ends with his turn.

Before taking a final decision about which proposed system was going to be more efficient we tried a couple of testing and prototyping in a smaller scale before trying to choose which one we implemented and developed in our design. When we tried the first approach of using a couple of distance sensors (GP2Y0A21YK0F) that we already had from previous shopping that we did because we were going to use them for our hole detection sensory system, and put them strategically one next to the other just enough to allow their elliptical paths slightly overlap and create an area with some specific width and scanning out a range of up to 80cm height. The idea of this approach was to design a system where the infrared sensors detect the passage of the coins as they are being tossed to the table but in reality we notice that it was not a good approach by using this kind of device as a curtain sensor. We even found out a proximity sensor that its spectrum worked in the same manner as if we put five distance sensors beams one next to the other, then the beams will be refreshed rapidly creating an illusion that will form a 25 degree curtain spectrum. The problems that we found in our small scale design of these systems were the accuracy and reliability of the readings from these sensors was not as desired or expected.

When the infrared detectors were exposed to the outside environment, any object or insect passing through the spectral range of these devices will trigger a signal that will be interpreted as a valid input to our microcontroller. When tested the two distance sensors we had some inaccurate readings when we place them outdoors; we could not conclude if some fail readings were because of the ambient light reflected to the sensors or because of insects flying close by the spectral covering range of the sensors. Another reason of for the rejection of this system was that since the sensors had to go on the floor shooting out their beams upwards, there was a great risk that bouncing coins or even players walking by may destroy them.

The other system proposed for the detection of coins was similarly to the infrared technology already tested with the distance sensors but in this case, we had using two rails and some emitter laser diodes on one end and photoelectric receiver diode on the other end and place them as an imaginary wall 3 feet in front of the player line of throwing. After consideration of this system, we decided not to even test it because we were going to need too many lasers and receiver diodes, many wires and too much labor involved in the setup of these double sided panel of emitter/detectors.

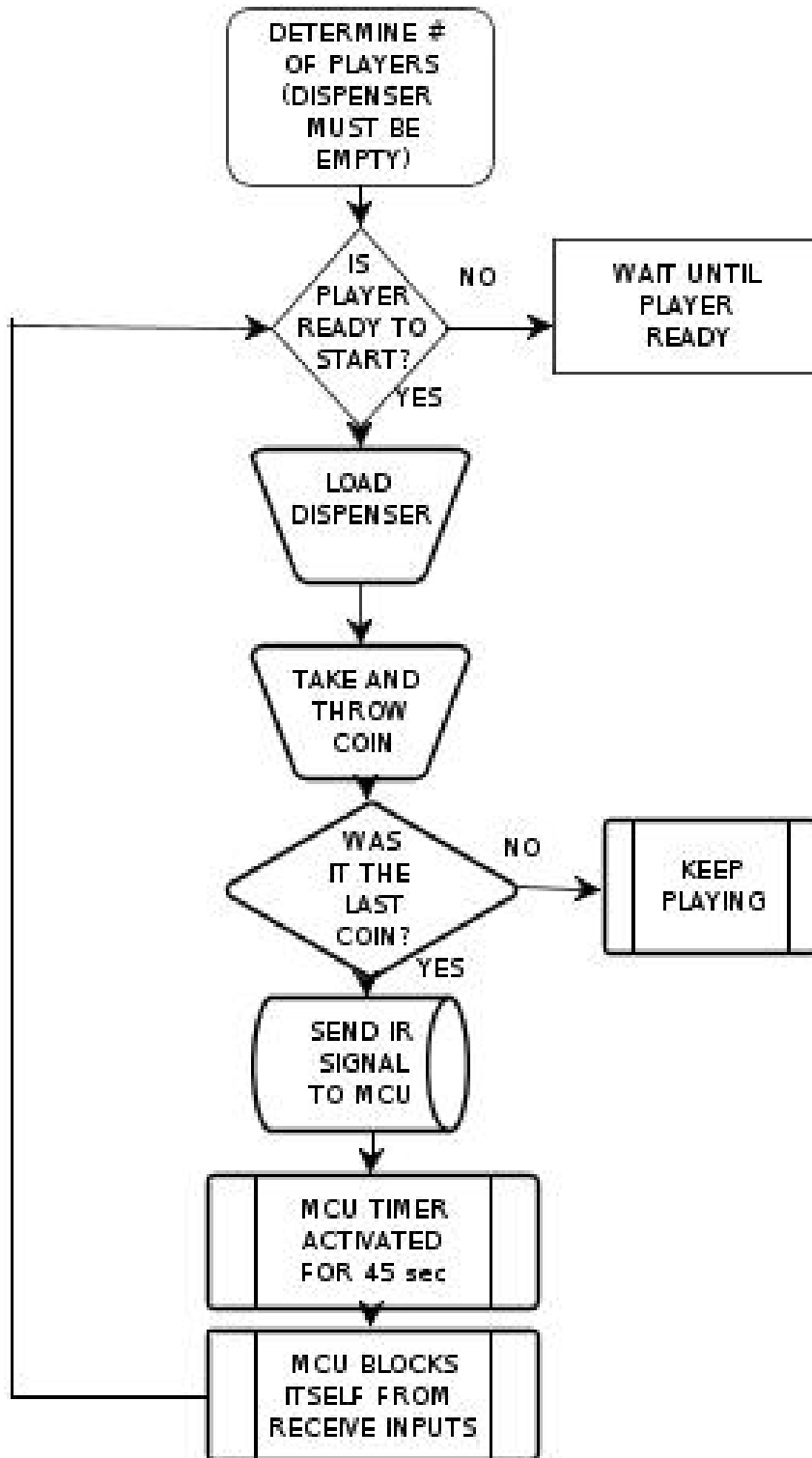
Finally, we started reorganizing our ideas and goals for this specific system and we clearly define what we wanted to obtain from this coin detection system and how we were going to add it to the whole game. We concluded that the main two reasons for this system were the automatic blockage or pause of the game after the last coin was thrown and the jump to the next player after the current one

finished his 10 attempts. To achieve such goals we did not need to implement or create a complicated or expensive system that can count each one of the coins. We needed to create a system to detect the last coin to be thrown and maneuver such information in a way that the microcontroller knows that such signal represents the end of the current player game and automatically blocks itself from receiving new inputs to prevent accidental dropping of coins inside any holes while collecting the coins for the next player. Also, it automatically shifts to the next player when everything is ready to start all over again. With the simple creation of the Coin Dispenser Sensory System we created a simple input device that satisfies both requirements and was relatively inexpensive to implement it to the project.

To construct this innovative system we needed to get some hardware material from our local warehouse selling store. Such parts were 10 inches in length by 1.5 inches in diameter of pvc pipe, a wye connector, a support pedestal, a pair of infrared emitter/receiver diode, wires and some shop tools to customize the parts previously mentioned. Building a plastic tube in an inclined plane where it has a bigger open end at the top to collect the coins and a customized open end at the bottom where the coins are collected one by one to be thrown was the solution for our design approach. The electrical part comes into play when we added an infrared emitter/receiver pair connected in opposite directions facing each other inside the tube and always energized. The concept of this design is to have the emitter source always shining infrared light towards the receiver creating a closed loop of power and when the coins are placed inside the PVC tube it will cut the closed loop formed between the light and the receiver.

This initial cut of beam was inversely interpreted by our microcontroller as a valid input that allowed the game to begin the round. Then we programmed the microcontroller to operate as long as it does not receive any more inputs from this sensor, meaning that it doesn't let the E-Frog Game to be fully functional while there is no signal being received from the coin dispenser. The trick or important part of the game comes into play at the time the current player takes the last coin from the dispenser to throw it in his last attempt to score but this time, being the last coin, we conditioned the system to operate for another 15 seconds more before going into sleep or pause mode again until reloaded. By setting up this system in this manner we achieved the two goals proposed for this system without having to count each one of the coins but only count the last coin taken out from the dispenser and reload game to the next player when the coins are inserted back into the dispenser and cut the light beam again, some basic programming conditions and timers will have to be encoded to the microcontroller for this system to operate as desired.

The following [Flowchart 3] is a representation of the desired operation of the Coin Dispenser System of our E-Frog Game:



Flowchart 3 Coin Dispenser System Flowchart

Display Screen Counter

DISPLAY

The E-Frog game is a project designed to display temporary and total scores per player, so in order to fulfill this requirement, a good display was implemented. After some research and tests, we, group number six found that the 7 segment LED display is a good choice to incorporate visual light scores in the game. On the other hand, we also discarded the LCD displays since they did not accomplish our project goals such as: long life duration, low weight (thin and light), picture quality, and a large color range. There are many advantages that LED displays include in their specifications. A single 7 segment display consist of eight LEDS including the dot, this screen makes a digit using the position segment corresponding to each digit. Each segment has a letter assigned from a to g as the figure below shows. For instance, if we wanted to represent number two, then segments a, b, g, e, and g were activated. The same way each digit from 0 to 9 can be represented activating (1) or deactivating (0) each segment. Table [Table2] below shows all possible combinations of segment LEDs that the system could include binary inputs and decode outputs.

	REGISTER DATA							
	D7	D6	D5	D4	D3	D2	D1	D0
Corresponding Segment Line	DP	A	B	C	D	E	F	G

Binary Inputs				Decoder Outputs							7-Segment Display Outputs
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	1	0	1	1	9

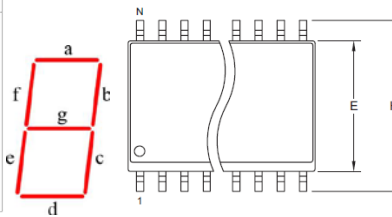


Table 2 LEDs combinations

The Table 2 above is the top view of one seven segment display that shows us the number of pins (N), the dimensions range of E distance is 0.291 to 0.299 in or 7.40 to 7.60mm, and the dimensions of H distance is from 0.394 to 0.419 in or from 10.00 to 10.65 mm. Group number six analyzed and tested different ways to design and build a good screen for the E-Frog Game, and one of the ideas we had was to build a four digit seven segment display since we know that a led is a light emitter diode that has a cathode leg that goes to ground and an anode leg that goes to a resistor (controls current in each segment) which modulates the voltage coming from a source connected to the resistor, and to find the resistor corresponding to that LED, we know that we subtract the voltage needed (From LED specification)from the source voltage an then we divide this voltage result by the current used (from LED specifications). Using this basic, we tried to connect four seven segment LED displays to make a four digit number, but each seven

segment LED requires eight independent resistors and eight independent power sources. This means that in order to make a four digit number display, we would need around 32 different resistors and 32 different power sources, also the current is too high when connecting the four seven segment displays. When we tried this possible option, we also found out that we would have to use decoders in this kind of connection. Using decoders, we would have to connect all segment position a together, all b segments together and the same for each of the segments from a-g including the dot led (h). These decoders would minimize this circuit choice because each of these can extend from 3 to 8 pins. Besides the resistors, sources, and decoders, the following design [Figure 32] option requires more devices like drivers, cables, and others.

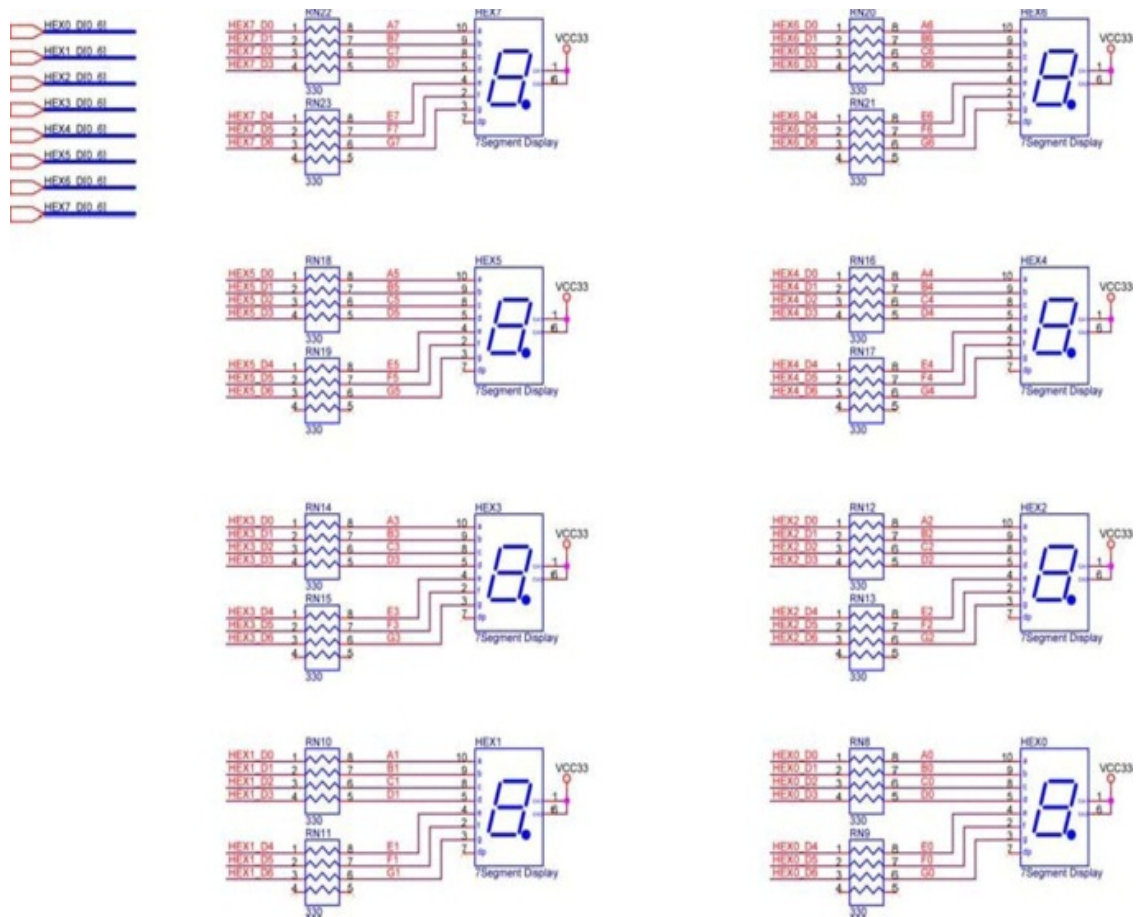


Figure 32 Multiple Decoders design

After we tried this option, we realized that making every four seven segment requires a lot of connections and electric devices as shown above, and it would take many pins from the microcontroller. Since the E-FROG game requires a total of 4 four digit seven segment display, and this implies many devices, circuitry, and another extra devices like decoders, this option was not the best to choose, so we decided to go deeper in research to save money and time making this part of our project.

Thanks to our current technology and after a lot of investigation and testing, we found a package part that includes all of these devices. This is the Serial Seven Segment Display Board MAX 7219. This display is a 24 pin DIP and consists of two boards 4 digits display that includes a binary code decimal BCD decoder to control each digit intensity, 64 single LEDs, character and segment drivers, scan multiplex circuitry, and a static RAM of 8 by 8. This is a serial I/O common-cathode screen that communicates with the MCU thru Serial Peripheral Interphase (SPI). Also, another reason why we selected this device is because the linking between the board and the system is thru a cable that ends with an IDC10 female header connector and a 2 by 5 connector on one of the ports which is compatible with most of the common boards.

The devices content a 150uA low power mode (data reserved), a test mode to turn all LEDs on, an analog and digital control of bright, a register scan limit to show digits 1-8, 10 MHz serial interface, specific LED segment control, decode or not decode digit selection, a display blanked on power up, a 24 pin DIP and SO packages, an interface of serial 3 wire 10Mbit, font of code B, voltage supply minimum of 4v and maximum 5.5 v. Some electrical characteristics like voltage of 5V \pm 10, resistor set of 9.53K Ω \pm 1%, TA= Tmin to Tmax. This board is only for 5 volts systems, and it is in the market for \$21.90 which was a good price since includes all the devices required to communicate with the microcontroller. The on-board MAX7219 driver allows us to easy add sixteen 7-segment LED displays to our project by using only 6 I/O pins of the microcontroller. The main benefit of using this device is the time-division multiplexing operations required for continuous refreshing of the display digits are performed by the MAX7219, in that way holding the microcontroller able for performance other pressing tasks. The MAX7219 outputs involve 16 bits, the high eight being the register and the low eight being the data. The 8 registers include the digit and control, decode mode, scan limit, display test, and No-Op register [Figure 33].

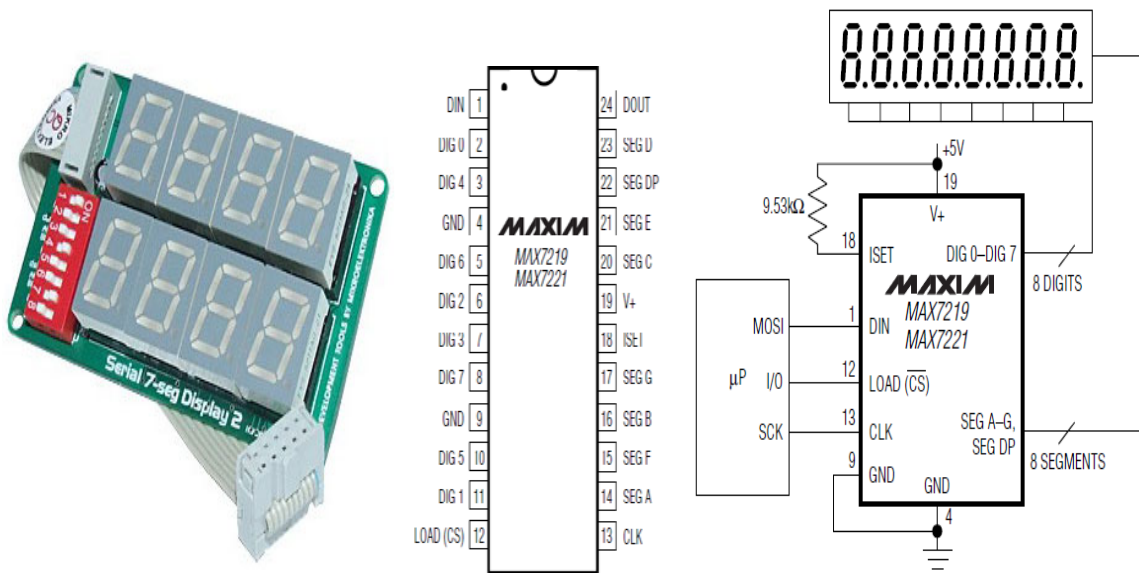


Figure 33 7-Segments LED Display

This 24 pin device has the following configuration pin functions: the pin 1 is named DIN, and it is the serial data input where the data is moved into the 16 bit shift register on CLK's rising edge. Pin numbers 2, 3, 5 to 8, 10, and 11 represent digit 0 to digit 7, and their function is to drive lines that drop current from the display shared cathode. Pin numbers 4 and 9 are the ground, and they must both be connected. Pin number 12 is the load (load data input) and the CS (Chip select input). Pin number 13 is the serial clock input (CLK) at a highest 10 MHz rate. Pin numbers 14 to 17 and 20 to 23 are the segments from A to G including the dot DP (Decimal Point) which source current to the display. Pin number 18 is named ISET, and it is connected to VDD thru a resistor to set the peak current to the display. Pin number 19 is the positive voltage V+ which is connected to +5 volts. And finally, the last pin number 24 DOUT is the serial data output.

Sound Effects System:

How joyful is life when we get pleasure from it with our senses. What about when we find relax and enjoyment in activities or games to get out from the monotony of our boring lifestyle? Even better, what if we can integrate both concepts together and get a better result in relieving stress and find enjoyment. The E-Frog Game is a project that seeks to fulfill as many requirements as possible to become a complete game and source of relaxation for its users and spectators.

This game encourages and promotes our body senses in a good way to create a pleasant environment in different ways. It promotes our tactile sense by the interaction of the user and the coins, by feeling them and improves the ability of maneuver the tossing on each one of the throws. It also entertains the user with the already developed light effect system and the interaction and combination of its led lights around the table box. Our body is designed to respond with emotions when we stimulate it throughout our senses and this is the main objective of this sound effects system.

Among all the other stimulations that we have already added to the E-Frog Game, we also have added one more system that made it super attractive and exciting to the user and this is the implementation of the audio effects system. How fun and amusing it is when we play a game and we hear some feedback from it. The objective of the audio effects system is to provide a response to the user when certain situations or conditions are met while the game is in operation.

These responses can be supportive, exciting, or to emit a warning if the condition applies. The goal was to generate an introductory sounding signal or mode to animate the beginning of the game, another sound to respond when a coin is inserted to a regular low-value hole, a more exciting sound when the coins are inserted into a higher-value hole, a warning message or annoying sound when an invalid input occurs such as stepping onto the threshold, and the most

important and attractive sound occurs when the player hits the main attraction of the game, the bronze frog's mouth.

In order to implement and add this interactive sound system, our group performed a research about the different types of audio system available to add it into our project and we found out a few options that we used. The first question we asked ourselves was: How can we add audio to a project to be controlled with a microcontroller? It is relatively simple in concept. We had to find an integrated circuit that consists of some latches.

In first step we added an Analog to Digital converter that transforms the sound waves captured or recorded from the emitter voice or source to a digital form to be handled by our program and this is done through a device called A/D converter. Once we accomplished the task of having the desired analogue data into a digital source (such as the voice changed to the digital value), we saved this information with latches (some type of memory device such as an EEPROM for example) as well. Then, as needed we handled this stored information in different ways as we need them, and one of the ways that we got them and played them was done through the Digital to Analog converter.

One of the options that we found was The Audio Capacitive Touch Booster Pack that includes a pre-programmed micro SD card with a directory of music files to experience crystal-clear audio and simple music player-like commands using the capacitive touch interface. This T.I. product was a very tempting option to add it to our system since it has the capability of record sound waves and sound files, convert them into a digital form and store it in a built in memory to be handled as needed at a later time.

One of the drawbacks from this design is the dependability that this Audio Capacitive Touch Pack has with the MSP430 microcontroller which means that we could have add a MSP430 Launchpad to control this audio board. This audio pack has more features than what we need and it could not be as cost effective as we would like it to be since we used it at its 100 %.

Looking for solutions to our dilemma in choosing the appropriate part to implement it to our E-Frog Design we had to choose between building the subsystem from scratch zero or getting a board from a third party that is compatible with the board that we already built to our project. Having said so, we considered the option of building a small board with an integrated circuit, analog to digital converters and vice versa, incorporate an EEPROM or another kind of memory device, a communicating port to configure the chip, among other electrical components such as resistors, capacitors, voltage regulators, crystals and inductors. But since the main objective of the E- Frog Game was not to build a specific system such as the audio system, we decided not spend so much time and effort in a secondary subsystem when the main goal was to identify and assign values to each one of the holes of the game.

The following diagram [Figure 34] shows a schematic of the block functionality of the chosen device:

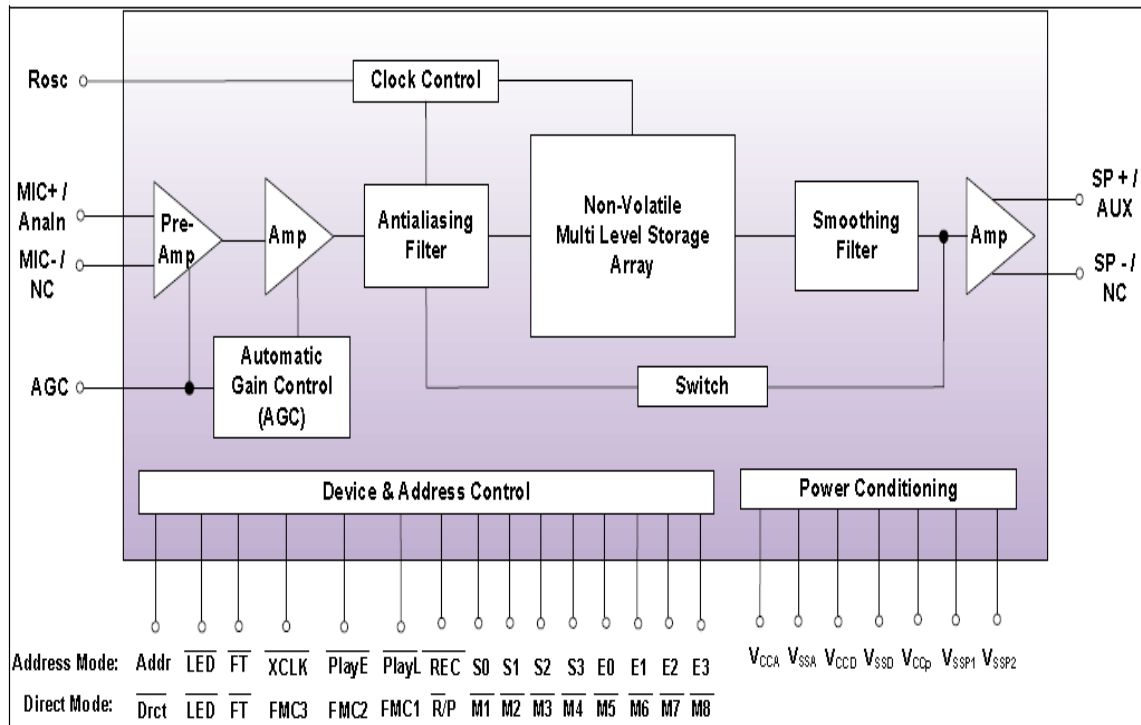


Figure 34 Block Diagram of Single Chip, Multiple-Message Voice Record/Playback Device

The Nuvoton's ISD1900 ChipCorder Series is a single-chip multiple-message record/playback device with dual operating modes and wide operating voltage ranging from 2.4V to 5.5V. The sampling frequency was selected from 4 to 12 kHz via an external resistor, which also determined the duration. The device is designated for mostly standalone applications, and of course, it can be manipulated by a microcontroller, if necessary. This is the most interesting part of this module, that it can be easily adaptable to the main board that we built, this saved us some time to finish up some other details concerning about the E-Frog Game.

The two operating modes are Address Mode and Direct Mode. While in Address Mode, both record and playback operations are manipulated according to the start address and end address specified through the start address and end address pins. In Direct Mode, the device can configure the memory up to as many as eight similar duration messages, pending upon the fixed message configuration settings. Since we wanted to keep the messages length uniformly, we decided to use the direct mode of the chip with 8 messages of 4 second each.

The following picture [Figure 35] shows the chosen sound chip module for the audio system and its dimensions taken from the datasheet:

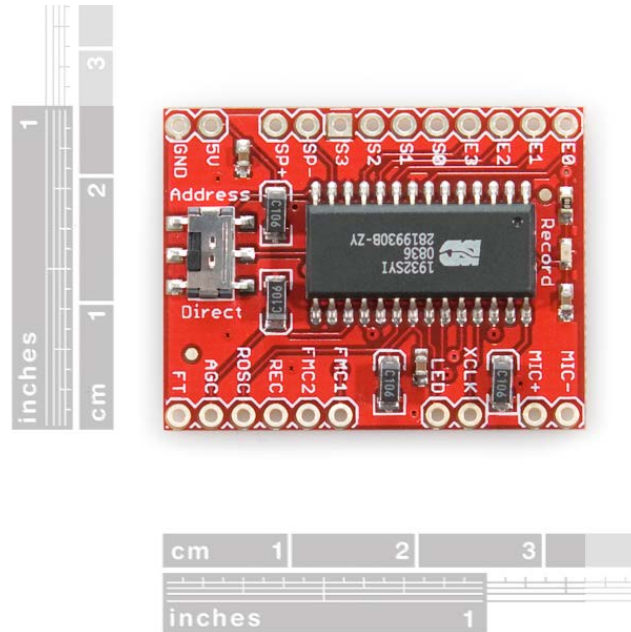


Figure 35 Sound Chip dimensions

With the record or playback feature being selected, each message can be randomly accessed via its message control pin. The device has a selectable differential microphone input with AGC feature or single-ended analog input, under feed-through mode. The audio output is either a differential Class-D PWM direct-drive or a single-ended voltage output (AUX out), depending on the derivative selected. After choosing this helpful device and incorporating it to the E-Frog Game, we made a proper placement of it inside the holed-table box and integrate it with the already chosen game components. Since this is a small board of 1.3in x 1in it was easily located but for convenience and less trouble with wiring, we placed it right next to the main microcontroller unit board. We required to have at least five I/O designated to control the different audio inputs to our chosen sound module, also we needed an enable line to control the operation of this sound controller and a voltage source and ground.

LED Decorative Light System

According to the number of LED's that are going to be needed, we will implement the following design. Our lighting system will have a total of 128 LED's. To avoid placing a resistor in each of the LED's and as a result having a workload twice as arduous, we decided to use a LED driver. The driver is a very important tool because it saves work and allows for a safer and more efficient design. The most important advantage of using this driver is that there is the possibility of regulating the current that goes into the LED's. The method of regulating LED current is to drive the LED with a constant-current source. The constant-current source eliminates changes in current due to variations in forward voltage, which translates into a constant LED brightness. Generating a constant-current source is fairly simple. This is done by regulating the input power supply with the voltage

across a current resistor. The lighting system will be made up of two 8 by 8 matrices which will be driven by two MAX7219 chips manufactured by Maxim Integrated. One of these chips is able to control one 8 by 8 matrix with only three pins from the microcontroller. The MAX7219 is a driver with the following characteristics: supply voltage from 4.0 to 5.5 Volts, operation current is 330 mA maximum, scanning rate at 800 Hz (if all LED's are one at the same time), and the Load, Data-in, and Clock pins to activate the chip. The brightness of the MAX7219 can also be controlled with an external resistor (RSET) connected between V+ and the current regulating resistor (ISET). The peak current sourced from the segment drivers is about 100 times the current entering ISET. This resistor can be adjustable or fixed. The MAX7219 already comes with the resistor (which cannot exceed 9.53KΩ), so this will set the current for each LED at 40mA. To implement the lighting system we will use two 2.3" 8 by 8 LED matrices (ELM-2881SURWA/S530-A3). The 2.3" 8 X 8 matrix to be used has the following specifications: forward current 25mA, reverse voltage 5 Volts, forward voltage 2 Volts, operating temperature -45°C to +85°C, and soldering temperature at 260°C, and it is low power / high brightness. Below is the configuration of one 8 by 8 matrix. It shows the wiring required in order for the matrix to work. To wire the matrix, the 8 registers from the MAX7219 are the Dig0-7 pins (ground coming from the matrix), and the columns are the Seg Dp. A-G pins (supply voltage to the matrix). [Figure 36]

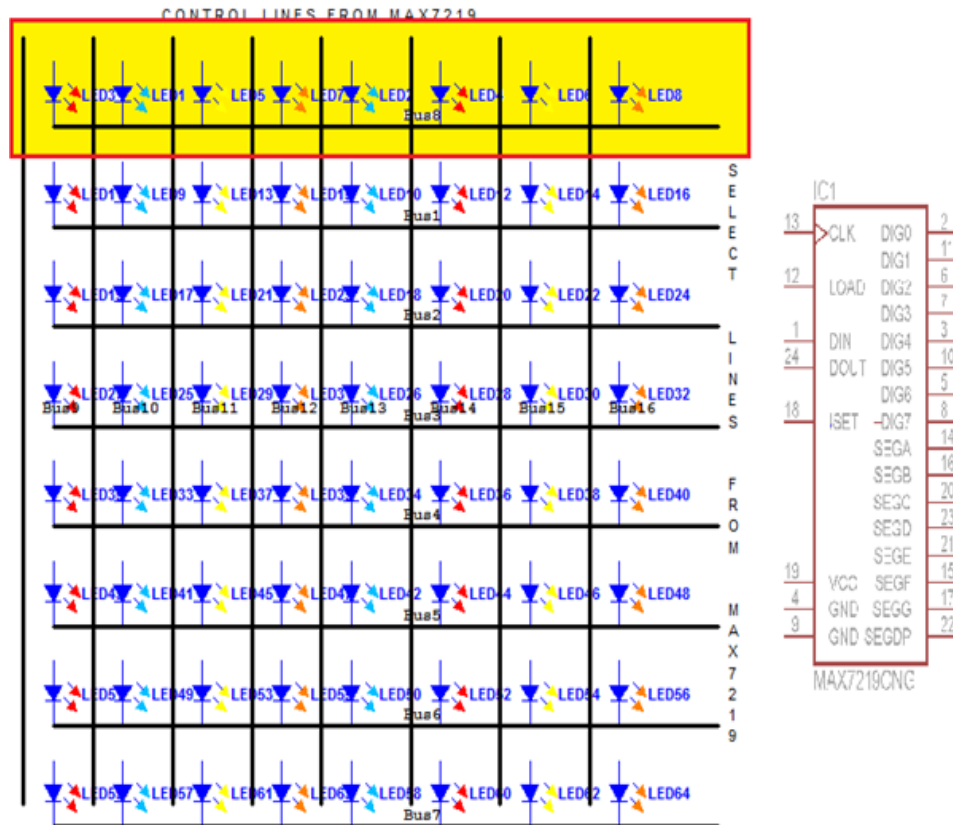


Figure 36 8 by 8 Matrix Configuration

As the figure above shows, the matrix to be used in this project is a common cathode type matrix. The cathodes (short lead of the LED's) or are connected to the control lines of the MAX7219 while the select lines of the chip are connected to the anodes (long lead of the LED). Another advantage of using the MAX7219 driver is that the chip can be cascaded to be used with more 8 by 8 matrices. For the E-Frog Game lighting system we will cascade two chips. This approach can save us a lot time and also several pins of the microcontroller unit that can then be used for other tasks. Only three pins will be used and they will provide clock signal (mark between every bit), the Load signal (mark between every command), and the DataIn signal (information / command coming from microcontroller).

In addition, there are two more pins, one for the voltage and one for the ground. Also, when using two or more MAX7219's a 10uF electrolytic and a 0.1uF ceramic capacitor should be connected between V+ and GND as close as possible to each chip. The wires should be should also be kept between the MAX7219 and the LED matrix as short as possible. Also, when using two or more chips the signal from the load and clock has to be amplified for better results. The design will be implemented using EAGLE Software Design which is an easy to use and reliable tool. From the point of view of wiring and building the system into the E-Frog Game, we will have holes drilled at the edges of the front part of the game where we are going to secure the two 2.3" 8 by 8 matrices with the two MAX7219 LED drivers that have already been secured to a custom made perforated board. The printed circuit board (PCB) will be attached just underneath the table where the holes are located to make the connections a little bit easier.

The E-frog game was designed with the goal of being an attractive and fun game. To accomplish this goal we decided the game was set to have four different animations according to the situation of the game. This animation system will be located in the front of the main board so that the lights can be noticed by all players when they are throwing the coins. In addition, the game will have this set of lights around the display screen where the score and player number are located. The animations that the game will have are as follows: The first animation would occur when a player inserts a coin into one of the twenty holes available. When this happens, an arrow pointing down and blinking will indicate that the player has scored. The second animation will take place when a coin is inserted in the frog's mouth. In this case, the word "frog" and two happy faces will be displayed for about ten seconds. The third animation will be displayed when the threshold sensor is pressed. When this occurs, a vertical line made up of dots will shift from left to right and backwards. The final animation will be displayed when the coin dispenser does not have any coins left and the player has thrown the last coin. At this time, while there is a change of player, we will see the light in the form of a dot going from left to right and backwards

5.0 Design Summary

The design of the Electronic Frog Game turned out to be not only a challenging problem, but also the fact of making the project became a perfect challenge for our relative lack of experience as engineering students. The group decided to split up the load of the systems, research, and writing of the report to be spread to a modular design, in order to get around of many of the challenges a less modular design would involve, such as combining code written by multiple people into one coherent code that can be compiled later to form a whole code of the complete game. By doing this split up of functions each one of us would be able to work on their designated system from beginning to end and implemented to the hardware part and building of the project.

This modular design also allowed for an equal division of labor, with each member of the group being in full control of the design of their designated subsystem. The design for the E-Frog Game was divided into several subsystems that all coordinated with each other to achieve its function. The subsystems that the E-Frog Game utilizes are the LEDs, sound systems, lighting systems, the threshold sensors as well as the infrared sensors, and the display unit. Each of these subsystems is designed to enhance and entertain the players throughout the game. They achieve this function by allowing the user to select the desired systems before the game even starts and manipulate the functions of the outputs or even inputs for a better playing experience.

The design of the E-Frog game was divided into several subsystems that were programmed to work with a central microcontroller unit, in our case the ATMEGA 2560 from Atmel, to achieve the desired functions of the game. Any event or change that took place during the game was detected by the microcontroller, and this information or change in state was immediately sent to the subsystem so that they could do their tasks. Another decision that we had to make when designing the E-Frog Game was to choose the right system of hole-detection. We had available the mechanical limit switch technology, the infrared distance sensor technology, and the laser array table technology. After a deep discussion and meditation about the pros and cons of the different proposed systems, we decided that the most appropriate one to choose was the laser array table system.

We decided not to use the mechanical micro switches system because of its complexity in building it and implementing it to our holed-table top. Had we chosen this system we would have had to use 20 micro switches, 40 wires (2 per switch), 20 flaps, extra construction of designated tunnels to guide the coins through each one of the holes, the parts list could have been more expensive since the cost of each one of these micro switches was about \$ 2.82 (DigiKey P#: SW1068-ND, also, the need to designate 20 inputs from our microcontroller unit (this means that more than one ATMEGA 2560 was needed to accomplish this task) and this alone meant that the programming part could become more

and more complicated. Overall, the option of the mechanical micro switches was a possible option to be used for our project, but not the best one.

The usage of independent units of distance sensor such as the Sharp Distance Sensor P#: GP2Y0A21YK was another option considered to the implementation of this technology to the E-Frog Game. In the beginning, the availability of this sensor seemed to be a very exceptional choice since it was a device that could do different and complicated measuring readings and converted them into a voltage expression inversely proportional to the distance that the object was approached to the sensor. Unfortunately, one of the main drawbacks from this option is that the voltage outputs related to the distance of the object was not linearly related to the distance.

The first 10 cm were not to be considered for the distance reading, then the next 30 cm presented a voltage reading scaled enough to be able to handle it into the analog inputs from the microcontroller and then convert them into bits, but after the first 40 cm, the voltage outputs were so minimal as the object got away from the sensor that the set up and arrangement of the bits became more complicated and with the tendency of overlapping when the proper ranges were predefined. Also, since the path of covering of these distance sensors was elliptical (2 cm at the beginning of the sensor and 16 cm at its widest distance, at 40 cm from the sensor), the overlapping of the light beams under the holed table was inevitable and that was something that we did not want to happen. To prevent or overcome the overlapping, we needed to redesign the dimensions of the holed table and that was something that we did not want to do since the initial proposed measurements could be affected.

The first subsystem is the sensory system. For the hole-detection system, the sensor was programmed to interact with the coins that entered a specific hole. For instance, the main table or the table where the holes are located had several pairs of infrared emitters and detectors. These sensors were placed at the edges and under the table box so that their range covered a row of holes. When a coin passed through one pair of these detectors, a low signal was generated by interruption and it was sent to the microcontroller, which recorded the amount of points a player obtained, and the points were displayed in the screen. In addition, the MCU was in charge of sending out a signal which did another specific task. For instance, once a coin went through one of the holes, the MCU received the signal from the ATMEGA 2560 and sent a new signal to the audio effects subsystem.

When we had to discuss and analyze the threshold sensor system to be chosen, we also had to investigate and decide for the best option available for our E-Frog Game. As we found different options as well, we decided to use the threshold sensor strip due to its simplicity, portability, and effectiveness in its design. The chosen threshold sensor strip is a strip that is commonly utilized in the handicapped industry equipment such as vehicle lifts that are used to put the wheelchair user inside the vehicle.

These sensor strips are located at the base of the lift, between the support posts of the equipment so it will detect if there is somebody near the lift while in operation and as a safety mode, it will stop the lift from keep working until the obstructing individual moves away from the lift.

The audio effects subsystem was charged with making a sound given a situation during the game. Again, this subsystem worked together with the microcontroller. The ISP 1900 chip is for us, engineering students; was a solution of great help because the DSP system was to be controlled only by a microcontroller. For example, when a coin went through a hole a sound or buzzer was heard at the same time. The microcontroller was programmed with predetermined sounds in order to have different sounds or voices for each hole a coin went into a hole.

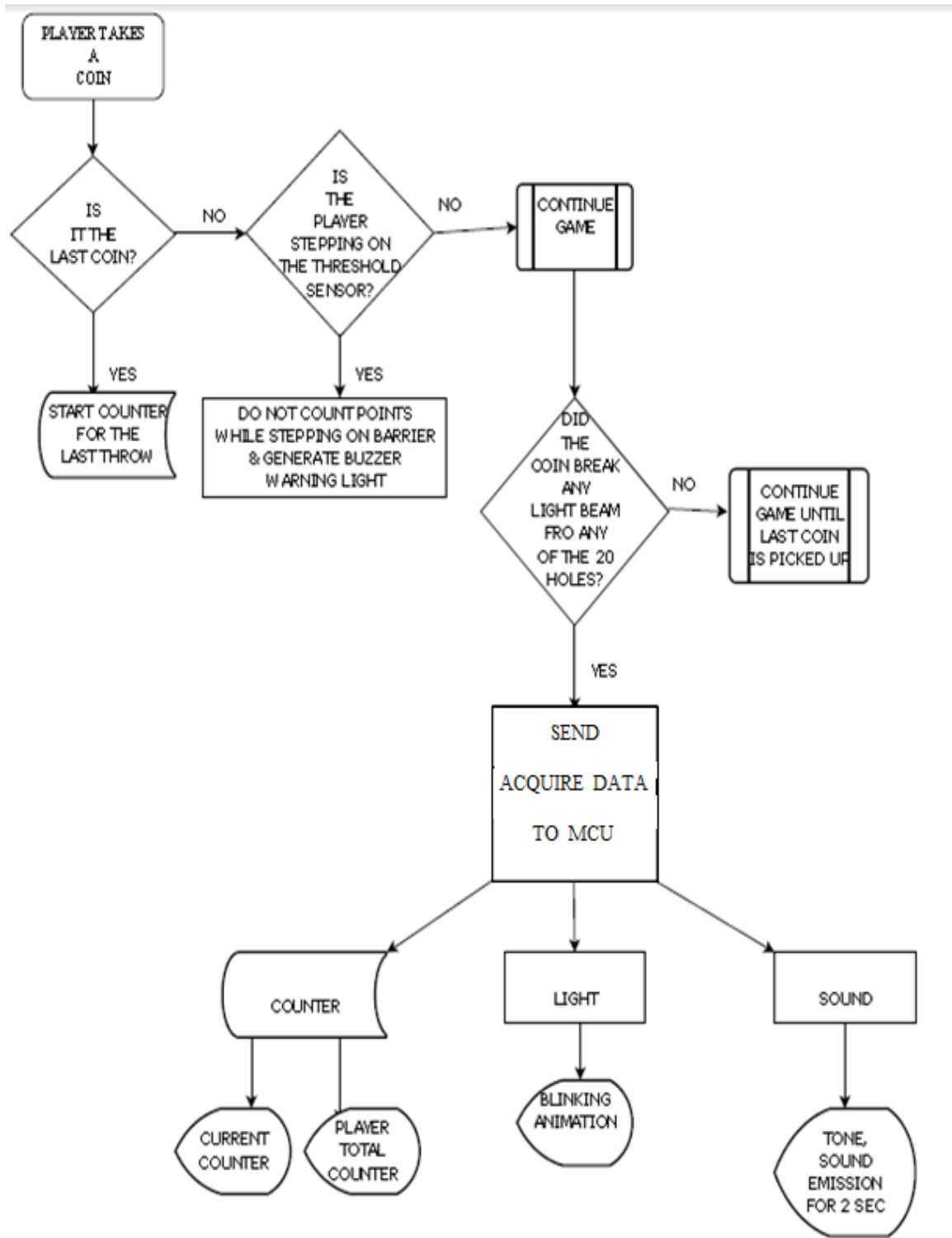
The decorative LED lighting is the next subsystem. It is a very important subsystem because it will give the game a more attractive look. As far as the design goes, a variety of several LED's of different colors were used. For this subsystem, several LED's were placed around the perimeter of the table where the holes were located and around the seven segment display which held the current information of the game. The LED's were connected in series. The main advantage of wiring them in series is that the total voltage from the power source is distributed evenly throughout each LED.

The E-Frog Game lighting system had different sequences and animations depending on the state of the game. For instance, it had a dot moving from left to right and backwards when the coin dispenses was empty. These LED's were controlled by a driver that kept current constant to each of the LED's without concerns of changes in voltages or temperatures, case in which the diodes could be damaged. Additionally, this driver served as a tool to dim the LED's and to change their chasing sequences. The sequences were programmed so that there was a different sequence once an event during the game triggered one of the subsystems. Also, a user interface system was created to allow the players an easier handling of the game before it began. The user interface was made up of one button (momentary switch) three slides switches, and one SPST switch that allowed the user to reset the game, disable / enable the systems of the game, and power on / off the game respectively.

The user interface worked hand to hand with the microcontroller unit and with our seven segment display unit. For instance, when a player disabled one of the systems, this information was sent directly to the MCU and from there to the programmed LCD seven segment unit, which showed each player's number of points and the number of players playing the game at any given time during the game.

The following [Flowchart 4] is the general description of the overall operation of the E-Frog Game. This diagram summarizes all systems including the five inputs (Disable and enable switches, hole-detection sensor, threshold distance sensor, and user control panel) and the four outputs (temporary panel, audio effects,

display screen counter, and decorative lights) of the entire system. This is how to play the game from when player takes a coin to be thrown to display his points, lights, and sound corresponding to his score.



Flowchart 4 E-Frog Game General Flow Chart

6.0 Testing and Prototyping of the E-Frog Game

Testing Procedures

In order to be certain that all components of the system functioned properly, we tested certain blocks and sub-blocks or systems of the E-Frog Game throughout the design and fabrication process. The project was broken down into different parts for proper testing of all its systems. As we already mentioned before, most of the systems were tested as we were developing on the prototype that we built. Such small prototype was of great help to give us the ability of testing each one of the game component in a minor scale, making sure that our proposed inputs and outputs worked fine and to adjust them as needed in the software coding. When we receive all the necessary hardware, we began by testing the signal of the sensor systems. We connected the analog-to-digital circuitry to the digital-to-analog circuitry to verify that the systems functions nominally. Once we verified the results, we initially integrated the prototype with the microcontroller and to the system. We then tested to confirm that each one of the sub systems of the E-Frog Game works and operates as needed.

During hardware acquisition and circuit board design we started of bench testing all the components as we get them from the manufacturers; for example, when we obtained the LED display matrix, this device came without any kind of instructions or datasheet, so we had to test it with the multimeter to find out its proper pin configuration. Since it is a common-cathode display matrix, we first started testing them to find out the proper pins for the cathodes, then we located all the 8 anodes respectively and tested each one of the LED lights on the display to make sure each one of the worked as desired; after every pin was located, we amp-tested to ensure that the current used by the LED's was within the desired and available current to be supplied by the MAX7219 driver that was going to control them. Following this procedure, we tested each one of all the components of the E-Frog Game as they arrived in the mail. In the following pages it will be explained more detailed each testing procedure applied to every component. We also verified all the concepts and fixed any initial design problems. Of course, nothing is always as expected and we bumped onto many inconveniences that we were not aware of but successfully overcome from it after the proper testing was done.

Testing the E-Frog systems took us quite a few weeks since this procedure involved many changes in structure of coding, soldering components, repairing damaged parts, integration with other systems working in parallel, adjusting different options and procedures following datasheets and recommended parameters, and making sure that it worked in our testing prototype for later implementation into the final game.

Testing of the Hole-Detection Sensor System:

In the same fashion than every other system of the E-Frog Game, the infrared laser array system barrier was tested in a special way, by testing different situations. As we already stated, the hole-detection sensor system is composed by four wooden rails with all nine infrared lasers pointers and its respective infrared phototransistor pair incrustrated onto the surface of these rails. The objective was to have them all aligned for proper operation and signal transfer.

Before any of the four infrared barrier rails was manufactured, the normal operating conditions of the emitter laser diodes and detector pairs was determined by independent testing. In order to achieve this goal, a single emitter laser was chosen to bench test it for proper operation at 3.3V as recommended at its respective datasheet. The test was repeated with each one of the lasers to ensure they emitted the proper radiation at the proper intensity and at a proper beam diameter. Since the laser pointers were adjustable on the beam spot, we adjust them all at an average of 3mm in diameter to cover the detection area of the infrared NPN phototransistors.

To test the NPN phototransistors we had to build up a special setting on a breadboard to test its operation on a development board and turning on a single LED light when the proper response was achieved. The receiver units were bench test them individually to ensure the detection of the infrared light emitted by the laser unit. To accomplish this testing procedure we had to ground the emitter leg of the NPN phototransistor, and then connect the collector leg to a node that was shared with a 10 k ohm resistor and towards one of the input pins of the development board used for testing (Arduino UNO Rev3). Such resistor previously mentioned was attached to a 5V source and of course, we connected everything in a closed-circuit configuration with the development board forming the previously explained "pull-up design" to detect any beam cut or interference with the phototransistor.

After the lasers were tested and confirmed that they all worked at the proper radiation level, we kept one ON all times taking special care with the handling of the laser beam (remember that it is a class IIIa laser and it could damage the retina if it is exposed for more than 2 minutes) and directed the beam towards the NPN receiver already setup in the breadboard with its pull-up configuration and already connected to the development board. On the coding part of the board, we set up an external LED to be turned ON every time the receiver lacks of the infrared laser beam, in other words, every time an obstruction cuts the radiation of light to the beam.

In the following picture [Figure 37] shown below we are displaying how the set up of the laser, receiver, LED, pull-up resistor, breadboard, and development board were placed for proper testing.

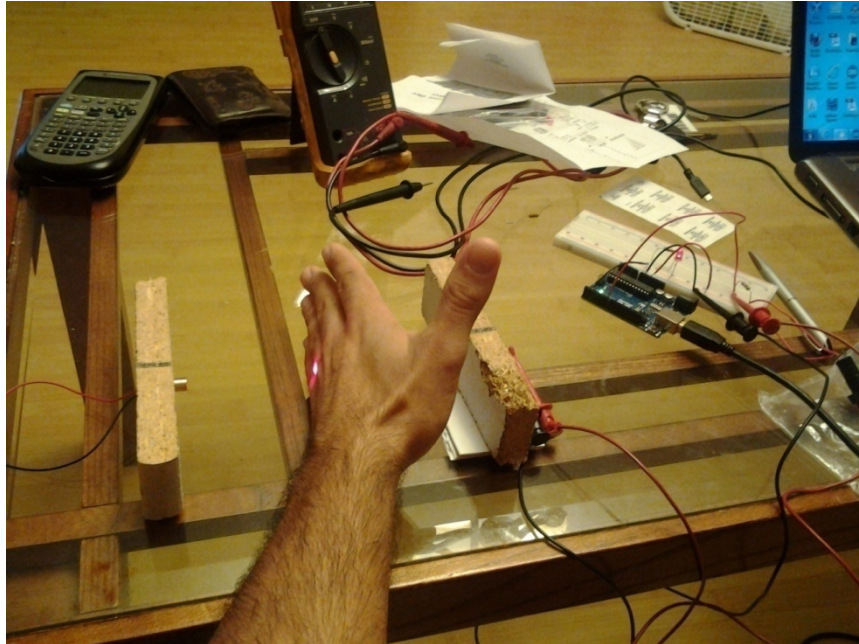


Figure 37 Bench testing a single pair of emitter laser and NPN phototransistor

The piece of code used to make this testing is displayed in the following lines:

```
void loop(){  
  // read the state of the receiver value:  
  receiver1State = digitalRead(receiver1);  
  
  // check if the receiver is lighted by the laser.  
  // if it is, the receiver is:  
  if (receiver1State == HIGH) {  
  
    // turn LED on:  
    digitalWrite(led1, HIGH);  
  }  
  
  else {  
    // turn LED off:  
    digitalWrite(led1, LOW);  
  }  
}
```

The emitter and detector circuits from the infrared laser array system research section of this document were constructed using the proper voltage, resistance, and other parameters already described before. By measuring the current and voltage of these units during testing, the uniformity of the other emitters and detectors can be determined. These measurements were recorded in the following tables [Tables 3-5]:

Laser #	Low Input Testing Voltage	High Input Testing Voltage	Desired Input Testing Voltage	Measured Laser Beam Diameter (mm)	Adjusted Laser Beam Diameter (mm)	Did it pass the desired requirement ???
1	2.5V	3.7V	3.3V	2.0 mm	3.0 mm	YES
2	2.6V	3.6V	3.3V	2.5 mm	3.0 mm	YES
3	2.4V	3.6V	3.3V	4.0 mm	3.0 mm	YES
4	2.5V	3.8V	3.3V	2.0 mm	3.0 mm	YES
5	2.4V	3.6V	3.3V	3.0 mm	3.0 mm	YES
6	2.7V	3.7V	3.3V	4.0 mm	3.0 mm	YES
7	2.5V	3.9V	3.3V	3.5 mm	3.0 mm	YES
8	2.6V	3.8V	3.3V	2.5 mm	3.0 mm	YES
9	2.7V	3.7V	3.3V	1.5 mm	3.0 mm	YES

Table 3 To be used to record emitter laser diode parameters

Infrared NPN Detector #	Lower Threshold Input Testing Voltage	Ideal Input Voltage	Measured Output Voltage at 5V Input	Number of times tested	Effectiveness Response Percentage	Satisfy Requirements ???
1	4.5V	5.0V	4.98V	25	96%	YES
2	4.4V	5.0V	4.99V	25	92%	YES
3	4.4V	5.0V	4.97V	25	96%	YES
4	4.6V	5.0V	4.98V	25	100%	YES
5	4.4V	5.0V	4.97V	25	96%	YES
6	4.5V	5.0V	4.96V	25	100%	YES
7	4.5V	5.0V	4.98V	25	100%	YES
8	4.6V	5.0V	4.97V	25	96%	YES
9	4.5V	5.0V	4.98V	25	100%	YES

Table 4 To be used to record detector output parameters

Once each emitter and detector has been tested for uniformity and complies with its ideal parameters, the most uniform and closer to the ideal specifications of the lasers available were selected to be embedded into the infrared array system units. Each coordinate laser array unit was then constructed and tested individually; in this case, each one of the twenty holes was tested individually after the rails have been put in place for proper operation. Each unit was connected in a configuration similar to that which it was in when embedded into the E-Frog Game, but was not connected to the microcontroller yet since we want to test its alignment and operation again without connecting it to the main E-Frog Board.

Once we had these rails aligned with its corresponding hole placement, in this configuration, the baseline measurements and independent testing for each unit was checked. This was a measurement of the reflection detected when there was nothing passing through the barrier and all the values from the receiver diodes should be ON since the laser beams were powering them. Then a coin was attached on a stick or on a piece of string to be used to test the reflection measured from the coin at each one of all the different holes and in all possible angles and positions of the coins, as well as the different speeds at which they can land inside the holes and through the infrared barrier matrix unit. During this process, a Digital Multi Meter was used to measure the voltage response and changes across the detectors when the coin is detected. These will be logged in a table of the following form when a source of 5V is supplied to the receivers NPN phototransistors.

Detector/Laser Pair #	Measured Voltage (No Object)	Measured Voltage when interrupted	Distance Between Coin and Receiver	Did it work as expected ???
A	4.99V	4.98V	21"	YES
B	4.98V	4.97V	16.5"	YES
C	4.98V	4.96V	12"	YES
D	4.99V	4.98V	7.5"	YES
E	5.01V	4.98V	3"	YES
W	4.99V	4.97V	20.5"	YES
X	4.98V	4.97V	14.5"	YES
Y	5.00V	4.98V	8.5"	YES
Z	4.98V	4.96V	2.5"	YES

Table 5 To be used to record receiver diode detection parameters

Using this test, the required voltage setting of the receivers was determined and it was not needed to manipulate or reduce the voltage on the receivers if needed to adapt it to a microcontroller unit without any problems. After this, the unit was connected to the available ports of the microcontroller. The controller was programmed with a variation of the final code specifically designed for testing. It will take the input from the matrix array unit, and will process this information by assigning its corresponding value to each one of the holes when the object is detected by the interruption. The bronze coin on a stick or string was one of the objects used in order to test that the controller is properly distinguishing the coins presence in the barrier plane and then, once the speed and accuracy has been adjusted, the test was also done with the bronze coin alone in free fall.

Finally, once each unit has passed the individual testing phase, we will start testing the Cartesian array of lasers at normal operation speed and conditions, simulating different scenarios like vertical falling coins, double falling coins, and bouncing falling coins to make sure the system responds at different than normal scenarios.

The following pictures [Figure 38-39] depict the moment in which the holes has been tested with an object to detect it when the barrier has been broken and turned a LED while the object was interfering with it.

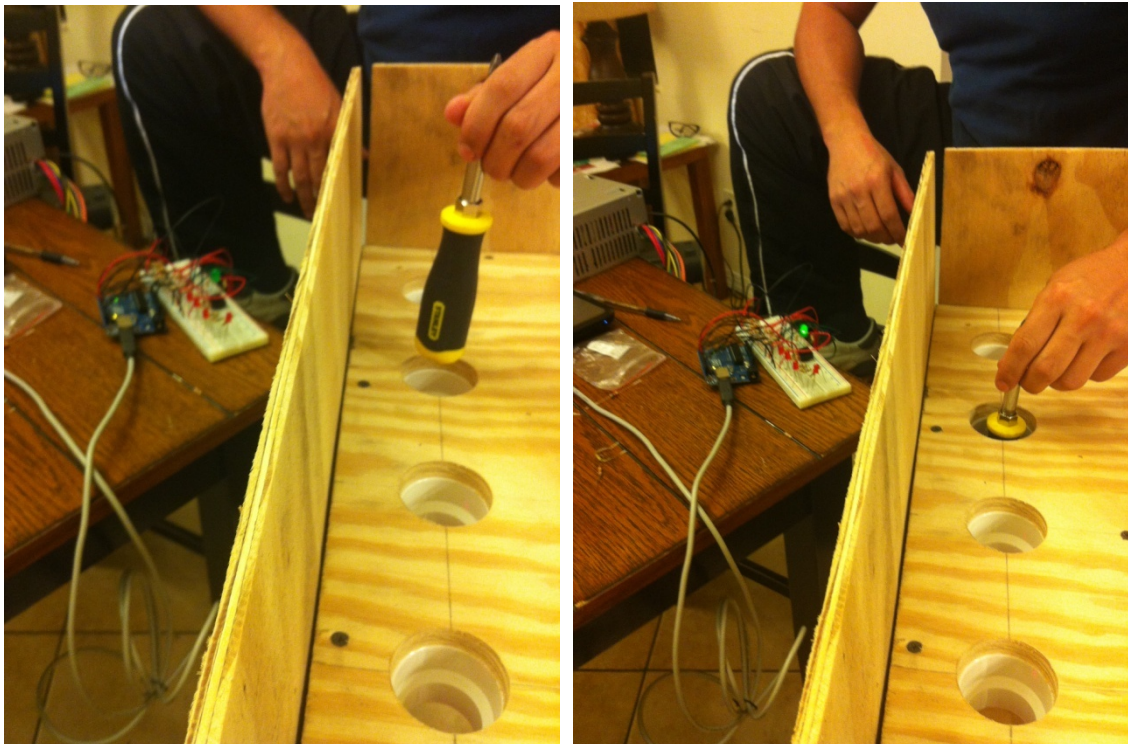


Figure 38 Testing holes without obstruction **Figure 39** Testing with obstruction (see LED)

Testing the Coin Dispenser:

As we previously described in the design and research part of this report, the coin dispenser is a system introduced to the E-Frog Game as a complement for its proper functionality and automatic operation and an anti-cheater mechanism to prevent the game from keep adding points to a player right after the player game has been finished. Once we had built the Coin Dispenser as desired, with the PVC tube, the emitter laser diode on one end and the receiver diode on the other one and connecting these signal wires toward the microcontroller but without attaching it to the MCU we will start testing the coin dispenser.

Following the same testing concept as the laser array system used to detect the passage of coins through the holes on the table, the testing procedure to pursue in this system will have to be similar but its core operation and functionality was inversely as the one used in the hole-detection system. Instead of interpreting a valid input to the microcontroller when the laser beam is interrupted like in the previous system, in the coin dispenser system a valid input was interpreted by the MCU every time the laser beam hits and activates the receiver diode.

One of the tests that we did to the Coin Dispenser device was to set the proper angle of the incline to allow a free fall of the coins through the device but at the same time, avoid tipping it too much because the coins may fall out from the bottom opening. After different positions tested, we determined that the most proper one for our goals was at around 45 degrees so we had to build up the pedestal or base that supported the dispenser at the proper level desired. In the following pictures [Figure 40-41] we can observe the sequence of testing of the coins entering the Coin Dispenser and the removal of the last one as well.



Figure 40 Coin Dispenser being loaded



Figure 41 Removing last coin

The electrical test was combined with the drop test of the coins to check for proper signal pick up when the dispenser is loaded with the ten coins and removing them one by one until the last one is removed to see if the beam properly turned on the signal for the empty dispenser and start the desired code from that point.

By accomplishing this operation, the coin dispenser sensor will send the signal to the E-Frog board's microcontroller to control the operation of the game as desired. Before connecting the signal wire to the microcontroller, we had to test the proper operation of this sensor manually so we connected the power and ground signals to the laser emitter and the receiver and connect the digital multimeter to the collector lead on the receiver transistor diode. Every time the transistor receives the infrared beam allows the passage of voltage to the multimeter. The goal of this manual testing is to ensure that the voltage collected by the multimeter was under the predetermined parameters and the operation of the detector is acceptable and accurate because the MCU is so sensitive that it will burn out if there is any wrong signal being sent by one of the systems. After the manual test of the coin dispenser laser/receiver system has been accomplished, it was safe to connect it to the MCU to do the programming at a later time.

The Coin Dispenser was connected to the development board to test its proper operation, to do this we used the pull-up configuration already described in the previous section. We connected power and ground supply to the emitter laser beam and we supplied with ground on the emitter leg of the NPN phototransistor while the other leg (the collector) was connected to a node that contains the 10 k ohm pull-up resistor and it goes to the board input signal pin.

One of the great challenges on this system was to utilize the signal sent from the NPN phototransistor to the PCB to do or serve as more than one input. In other words, we needed to split this signal in two to be able to manipulate the initial countdown timer that we wanted to add to the game and the other signal to be used to keep the game blocked while the Coin Dispenser was empty. Such task was a little complicated since we only had one sensory sending unit (the receiver) and we needed two inputs to the E-Frog game. There were three possible solutions to solve this problem that we thought at that time; the first solution was adding another pair of sending/receiving unit, the second option was to add a piece of hardware such as a reset chip board, and the third one was maneuver and modify the software to accomplish the needed task.

By adding another pair of laser/receiver to the Coin Dispenser unit may have solve the problem but, the solution was a little unpractical and also, due to the proximity of the receivers and the limiting space available inside the dispenser such option became rapidly discarded.

The other option, a little bit more complex but smarter, was the option of adding an external reset chip that will serve both purposes, to send an initial pulsation

when the laser first hits the receiver, and to keep sending a continuous signal to the microcontroller while the same signal is being emitted from the only NPN phototransistor. Everything seemed to be ok when we planned to take this option, even though we did not like the fact of adding more electronic devices to the game, we thought it was the solution chosen.

Finally, we did not wanted to accept the option above until we have utilized all possible options from the three previously mentioned before, so the last option was to research, create, and modify a new code into the software to be able to handle a single continuous signal and interpret it or make the MCU 'think' that are two different signals. One that it will be just a pulse to 'kick on' the final countdown timer for the last coin to be thrown, and a second continuous signal that will 'keep' the MCU busy while this signal is applied. By keeping the MCU busy we create a blockage in the input system and no more coins will be detected after the timer has ended and the continuous signal is applied.

Due to the fact that none of us was a pure programmer by nature, this option was a little more complicated than the others to implement but, at the same time was the simplest and smartest to choose if it was doable. Using this option was going to eliminate the need to add more hardware electronics to the game and overcomplicate the design since everything could be solve in a few lines of code.

We took the challenge of researching more about the problem and after a good amount of hours of study, trials and errors; we were successfully able to encode it just the way we wanted to get it done. To accomplish this, we split the Coin Dispenser code into three parts. The first one takes care of the first pulse of signal just at the time the receiver is being lit by the laser and inside this part is when it sends the command to the microcontroller to start counting down the last 15 seconds to throw the last coin and raises a flag to enable the next line of code after the countdown has started. In the next part of code, once enabled by the first pulse and its flag, it turns off the first part, and then sends the MCU to check an external function for the dot animation and this operation keeps the MCU from receiving more inputs from the hole-detection sensor system and consequently blocks the game and also enables a flag for the next part of the code. Finally in the last part of this code takes care of monitoring the Coin Dispenser status, as soon as it is reloaded, the program gets out from blockage, turns off previous flags, and restarts the game to the next player by enabling the first part of the Code Dispenser code.

Decorative Lighting System Testing

As with other components of the E-Game, individual parts of the lighting system were tested first, and then one by one they were added to the entire game to confirm their functionality. First, the MAX7219 was tested with single seven-

segment digits to make sure we had an understanding of the wiring and the code written. Once we did this, we moved to test the chip with one of the LED matrices. The advantage of implementing this system with the MAX7219 was that we only needed three pins plus VCC and GND from our microcontroller unit. The first obstacle found was that the chip was very sensitive to heat. The first time we soldered the chip to the matrix we did not get any response from it. So we decided to use a second MAX7219 to solder the second 8 by 8 matrix and the result was negative. After going over various documents we found out that the chips overheated and the most obvious reason was because the soldering was done without using a socket to place the chip. A third attempt was carried out this time using a new chip and a socket, and it worked. Even though it worked, we were very careful when performing this task with other components of the game.

Another important factor when testing the matrices was the distance required from the chip to the components. As it was mentioned before, the wires needed to be at a distance of at least ten centimeters from the matrices in order for the chip to work. To test this assumption, we first cut the cables about one foot from the chip and connected them to the matrix. The first time we did not get a response from the chip. The second time we tried the cables at twenty centimeters of length and the chip did not respond. In the third attempt the wires were cut to a length of about eleven centimeters and the matrix worked. At this time we realized that when we used two or more MAX7219's the load and clock signals weakened and lack of power affected the functionality of the chip. In our final test with the points display and the two matrices (three MAX7219's used), each set of wires belonging to each of the chips were cut to about 9 centimeters to ensure a positive response from the system as a whole. In addition, while testing the chip we found out that a 0.1uF capacitor needed to be added to the circuit because eliminating noise from the power supply was fundamental in order for the matrices to function. We tested the one matrix with the wires already shortened but without the capacitor and the outcome was negative. But, once we attempted to use the chip with the capacitor in place the MAX7219 responded perfectly for the rest of the testing sequence.

Also, a 28K Ω resistor was chosen to set the current to the LED's. The value of this resistor was selected in conformity with the MAX7219 datasheet. In addition to the matrices, the game has a temporary lighting panel which shows the position at which a player inserts the coin. The first step in testing this panel was to test all the twenty LED's located within the panel for functionality. This testing was done easily by using a 3 V battery and checking the response of each LED. This process was implemented over 10 times and the effectiveness was one hundred percent. The second step to test the temporary lighting panel was to test it along with the twenty holes of the game. To do this we needed to test the lasers / receivers at the same time with the LED's of the temporary panel. The idea was to insert a coin in any hole and have the position of that hole lit up in the panel. Once all connections were made, we started inserting coins into the holes. This process was done for every single hole of the game, 15 times per hole. The

response of the light panel was about 96%. The small error was due to small problems with the wiring and the alignment of the lasers.

As far as programming, both matrices were tested with simple routines to ensure that each dot in the matrix functioned. As the game became more complex, more routines were used to satisfy the requirements of the animations. To accomplish the tasks of the animations, we set each bit of the LED we wanted to light up to one, and the LED we wanted off to zero. So by using 8 by 8 matrices we had the opportunity of writing messages on them or drawing small figures. For instance the happy face was done by setting bits to one in the positions [1, 3], [1, 6], [2, 3], [2, 6], [3, 3], [3, 6], [4, 0], [5, 1], [7,6], and [8, 4]. All other bits were set to zero. Also, each sequence of lights was tested for duration and frequency. Using small delays we set both matrices to work synchronously. The code was run twenty times to ensure efficiency. [Figure 42-46] show the animations of the lighting system.

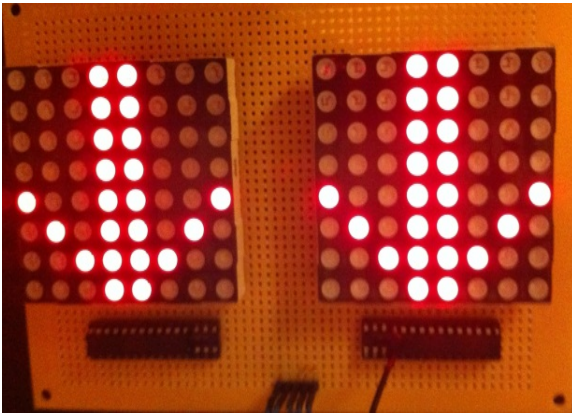


Figure 42 Arrows pointing down

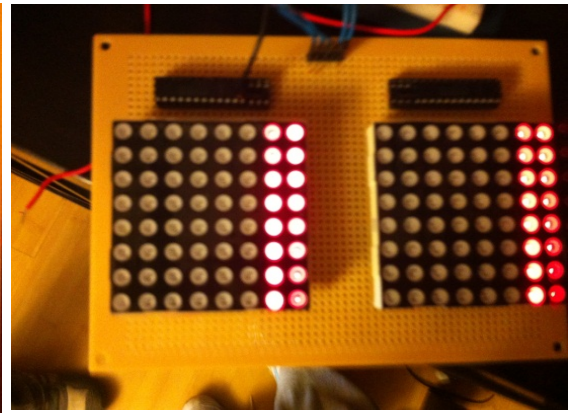


Figure 43 Threshold System Animation

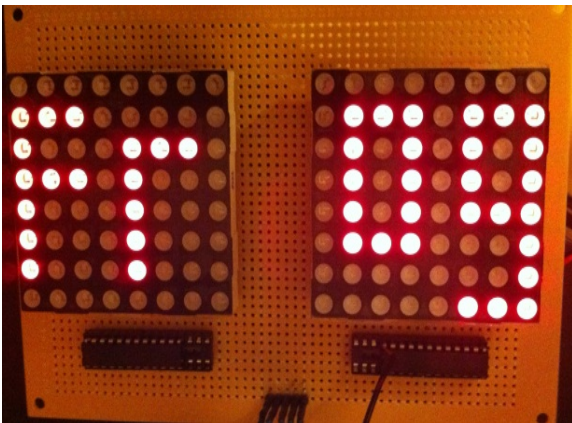


Figure 44 "Frog"

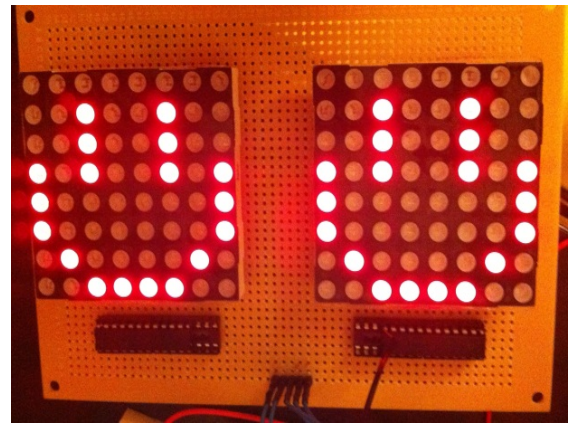


Figure 45 "Happy Face" animations

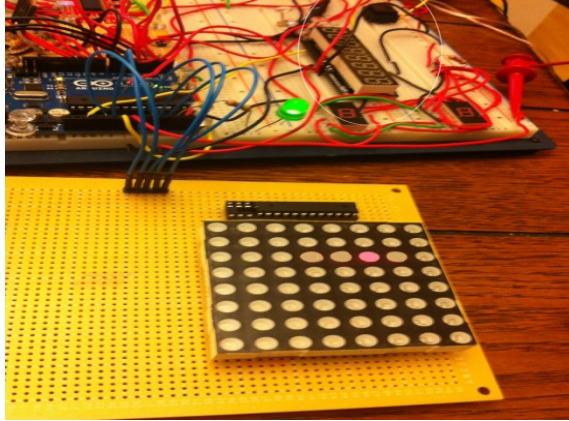


Figure 46 Coin Dispenser Animation

Threshold Sensor Testing

The threshold sensor system was tested by first connecting the sensor by itself with the microcontroller. We obtained a reading every time a player stepped on the strip, even if the player stepped for a very short period of time. As with the other systems, a de-bouncer circuit was implemented to avoid obtaining wrong readings. First, we tested the strip sensor in each point of its length to make sure that we got a valid reading if a player stepped anywhere on the sensor. The sensor was physically checked and a multimeter was used throughout its length to test for voltages when a force was applied. Since the game is to be played under different weather conditions and environments, the sensor was tested on different surfaces.

We tested it in solid ground conditions first to calibrate the resistance needed for the sensor to work. Also, we tested the sensor for occasions when the game is to be played outdoors. Since the possibility of playing the game in an uneven surface could arise, we tested the strip sensor 50 times in order to get a percentage of valid readings. It is important to say that when the sensor was placed on curved surfaces, the accuracy of the readings diminished and the response of the sensor was less effective. These valid readings were in the range of 95 to 98 percent. When the sensor was calibrated and we got a steady number of readings and then it was tested with the audio system. This procedure was carefully monitored to make sure that every time someone stepped on the strip, the sound system reacted along with it. In addition, the strip sensor was tested when there were no coins available in the coin dispenser. We wanted the sensor not to work when the last coin was taken and when the game was updating the information on the next player.

As mentioned before, when the threshold sensor was pressed, future inputs were blocked and not recognized. Also, we tested the accuracy of the signal sent by the microcontroller to both the coin dispenser system and the strip sensor. When they were fully synchronized, then the game was ready to be played. The following images [Figure 47-48] show the sequence of functioning.



Figure 47 Before pressing sensor



Figure 48 While pressing sensor

User Interface Testing

The user interface of the E-Frog is an important system. Even though some of its parts are not immediately related to the actual circuits, they are needed for the continuity of the game. The coins, for instance, are the objects with which the players interact with the most. They were tested to check compatibility with the phototransistor of the hole-detection system. To a higher level, of course, the press buttons were tested, along with the seven-segment LCD display which showed the amount of points and the current information of the game.

The coins were tested to make sure that their weight, diameter, and material worked perfectly with the systems. Initially, the coins were tested with each hole to make sure that the lasers responded to the change of state produced by the coin. Weight was a very important factor because we wanted the coins traveling at certain speed. The material of the coin was tested for deformation because many of them were hitting the ground repeatedly either when a coin missed the table or bounced off of it; therefore, making the coin loses its rounded shape.

The game was played at different weather conditions to test the coin's durability. After taking these considerations, a round of 50 throws was done both indoors and outdoors, and every result was recorded.

The push button of the user interface was tested independently to make sure the whole system worked. The switches were tested to make sure they operated correctly. So, using an ohmmeter with the switch closed, we got approximately zero ohms. With the switch open we got the voltage of the power supplied. The same steps were taken to test the voltage with a multimeter. Also, the button switch was tested with de de-bouncing circuit already implemented to make sure we got the desired results. The most important button of the system was the "on / off" button switch because it was the one that turned on the game for the first time. It was tested along with the power to make sure that once the switch was moved to the on position; the game gained enough power for all the systems to work correctly. In the specifications given, the switch used had a range from 0A to 20A, which is plenty of amperage to run the game. The slide switches were tested 50 times each to check for malfunctions or bad connections. Even though the slide switches are very delicate and sensitive, about 97% of the times they worked. In addition, as a last step in testing the switches, the entire game was connected together to check the systems as a whole. When disabling or enabling the systems, we found that the switches worked every time.

Overall, we tested the button and switches of the game at least 400 times to make sure that we had a high percentage of efficiency when disabling or enabling the switches. Finally, when a given game is finished, the "reset" button is pressed to delete permanently the scores obtained by the players. As soon as this button is pushed the LCD display showing the scores goes back to its original position and turns off. The procedure, from turning on the game to resetting the game, was run at least 50 times to make sure that the system worked altogether with minimum error. [Figure 49] shows the switches.



Figure 49 User Control Button and Slide Switches

Testing of Display Screen Counter (7 segment LED displays)

Another way of testing the switches system is to get verification with a screen response, so testing the five LED displays helped us to confirm that buttons were working correctly. First part of the user interface we tested was the current game status demonstration on the 7 segment LED displays. The display must show the sum of the scores of each person. The first four digits were used to display the player number, and the second half of four digits was used to display the points corresponding to that particular player. Then, every time there is a change of player the points add to the corresponding current player, so at the end we know who the winner is.

To test each common cathode seven segment LED display on the board, we used some experimental tools such as an orange and a black hookup wire. We selected an appropriate length for each jumper, as well as make the proper +5 volt and ground connections. Then we set all eight logic switches to logic 0, and turned on power to our testing circuit. By that point, each LED segment was off.

We had to make sure that all seven segments: **a** to **g** (LEDS) work in order to display numbers from 0 to 9. It was not difficult to test all ten digits of each display. For instance, digit 2 requires the corresponding segments **a**, **b**, **d**, **e** and **g**. On the same way digit 3, just removes the segment **e** of the digit 2 and add segment **c**. The numbers 6 and 9 have the option of using segment **a** for number 6 and segment **d** for number 9. Testing the display was not easy since the maximum distance from the MAX7219 driver to the display was just 10 inches, so we tried many times and we had to cut the cables many times until we had good signal. Also, this driver is very sensitive to heat, so we had to solder it using sockets and very precise.

The following picture [Figure50] shows the soldered 7-segment LED display:

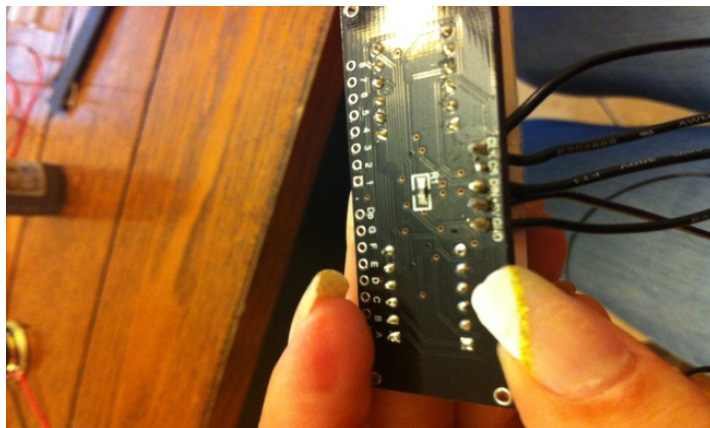


Figure50 7-segment LED display connected to MAX7219

The following pictures [Figure51] show the testing of the LED display with the microcontroller and the result of the points counter of a player.

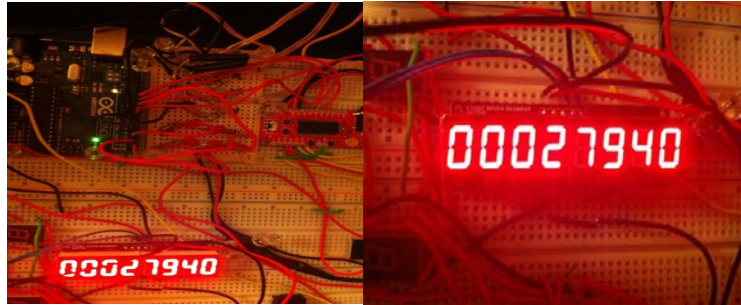


Figure 51 LED display test

Once we have completed this part of the testing, we made sure that the power to our experimental circuit was turned off. Then, we removed all of the jumpers from our board socket. When each of those conditions was satisfied, testing moved on. Now group number six was able to throw some coins to check if the correct points show in the corresponding screen. For example, if the game starts and there are two players; the coin dispenser needs to be loaded with the 10 coins once it is full, the display shows on the first half digits the characters “PL1=” meaning player 1 and the second four digits half displays the points corresponding to the sum of the scores in that round. Then, after the coins were used the coin dispenser is ready to be re-loaded once it is full with the 10 coins then the display changes to the digits “PL2=” meaning player 2 is ready to start. After player two finishes that round then coin dispenser is reloaded and the display shows the accumulated points corresponding to player 1, then player one continues the second round and each next rounds the same.

The following pictures [Figure52-53] are the demonstration of our final physical E-Frog Game with the display showing one of the member’s name and that the “PL1=” (player1) is ready to start.



Figure 52 PL1= displayed



Figure 53 Developer's name

Testing of the Sound System:

As the others systems, the last system of the E-Frog Game the audio system, was also tested. The Sound Chip ISD 1932 allowed us to record 8 messages. In order to test them, we have to play them back placing the REC pin in high (5V) and the ground jumper in low (Ground). Since we selected the direct mode as operational mode which let us have all the messages of the same length, we tested it at different sampling frequencies depending of the resistor chosen. For instance, for a 100k Ω the sampling frequency was 6.4 kHz and the duration me 40 seconds as the table [Table 6] below shows:

Sampling Frequency	12 kHz	8 kHz	6.4 kHz	5.3 kHz	4 kHz
Rosc	53.3 K Ω	80 K Ω	100 K Ω	120 K Ω	160 K Ω
ISD1932	21.3 sec	32 sec	40 sec	48 sec	64 sec

Table 6 Sampling Frequency vs. Resistance

After testing with different resistor, we deiced to use the 80K Ω because the frequency and duration were suitable of what the E-Frog Game requires. For this resistor value we had a frequency of 8 kHz and a duration time of 32 seconds which allowed us to implement 8 messages of 4 seconds each. This resistor is connected to the ROSC pin. Also a capacitor of 4.7uF was connected to the AGC (audio gain control) pin in order to reduce noise followed it by ground. The following table [Table 7] represents all the possible connection options corresponding to the number of messages, the highlighted row show the one we selected, the E0 to E3 are the four first pin messages and S0 to S3 are the corresponding pins for the last four pin messages.

Table 7 Message Bit Combinations

	FMC1	FMC2	X (clock)	# Messages
E0	0	0	0	1
E1	0	0	1	2
E2	0	1	0	3
E3	0	1	1	4

	FMC1	FMC2	X (clock)	# Messages
S0	1	0	0	5
S1	1	0	1	6
S2	1	1	0	7
S3	1	1	1	7

We have first tested each of our systems with the prototype which includes just four holes then also four messages in order to later maximize it into 20 holes. This prototype was really helpful in each system in hardware as well in software matters. Testing the electrical part of the chip was just changing jumpers from low to high and high to low as corresponds. To record we put the jumper on low as well the REC pin.

To play back we change the REC pin to high and we place the ground jumper on each message pin, it was just need to touch the pin in order to listen to each message. We tested many times and re-recorded different messages, and we decided to incorporate the messages depending of the level of points earned. For instance, for 30, 40, and 50 points we have recorded the message “You scored, but you can do better” for scores of 60, 70, and 80 we recorded the message “Good job, try higher”, for scores of 90, 100, and 200 points we decided to record the message “Great job, keep high”, and for the maximum points of 400 (Frog’s mouth) we assigned de sound “FROG you made it” (claps). Also, the other five messages we have implemented in the game were: “Welcome to the E-Frog Game” followed by the names of the members, when threshold sensor is steeped also there is a sound of “Move back, you are too close”, when coin dispenser is reloaded indicating that the game is ready for next player, there is a message recorded of “last coin, next player” the following player can start throwing. Finally, the last message that we, group number have created for our audio system was the end of the round for the current player with a message of “Game over, next player”.

This electrical test was combined with the drop test of the coins to check for corresponding message of each hole. In the following pictures [Figures 54-55] we can observe the sequence of testing just with a touch of each pin, we were able to listen to each message, and sometimes re-record on top of each message several times until we all agree.

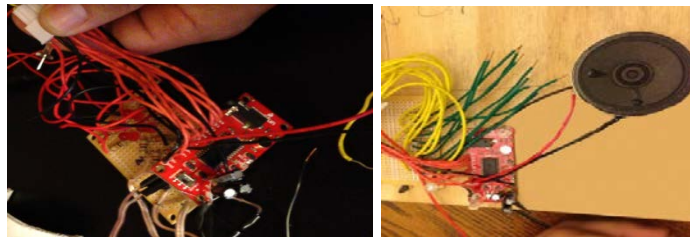


Figure 54 Prototype test with 4 holes and 4 messages

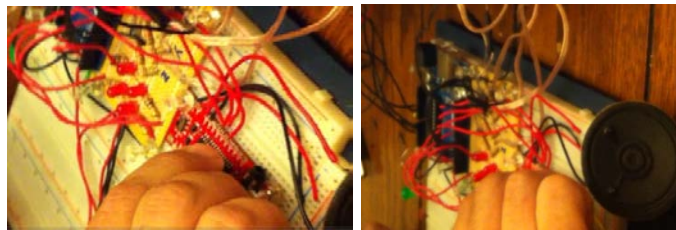


Figure 55 Testing msgs (Grounded on each pin msg)

Finally, our sound chip ISD 1932 placed next to our print circuit board and connected to the 8 pins of our microcontroller ATMEGA 2560 in the pins numbers 53 to 60. Now E-Frog Game is ready to sound every time the game is reset, a coin is inserted in a hole, next player is ready to start or the game is over.

7.0 Administrative Content

PROJECT MILESTONE (SUMMER AND FALL 2012)

Senior Design 1 (Summer 2012)

This section shows the plan our group has established up as a reference in order to keep track with all the responsibilities we have to accomplish during this months. This is an approximate milestone we followed to build, test and document our project.

05/14/12	First week of class. The challenge begins...
05/21/12	Understanding the Process of the Final Project.
05/28/12	Group formation and discussion on Project Idea.
05/31/12	Initial project outline and document completed.
06/04/12	Decided subsystems to be included and researched.
06/11/12	Begin work on project research and documents.
06/18/12	Research and analysis for subsystems and operation.
06/25/12	Research and analysis for subsystems and operation.
07/02/12	Finalization of proposed subsystems.
07/09/12	Integration of all information obtained for the final paper.
07/16/12	Project discussed to get it ready for public presentation.
07/23/12	Project information and documentation finalized.
07/30/12	Project Paper is due and ready to be turned in.
08/06/12	Project has successfully passed the required specs and it gets ready for prototyping it next semester.

The following chart [Table 8] is a tentative schedule that we have created to accomplish the previously mentioned milestones that we have established:

SUMMER 2012 (MAY 20TH TO AUG 3RD)

TIME:	MON	TUE	WED	THU	FRI	SAT	SUN
Morning		10-11:50		10-11:50		10-12m	
Night	8:00-10:00	12:30-3:30		12-1:00pm	3-7pm	1:00-3:00	

Table 8 Project work plan

Senior Design 2 (Fall 2012)

In order to keep to a timely schedule for designing, building and testing the Electronic Frog Game, the group has agreed upon milestones. These milestones helped us keep the group on task for working toward the end of Senior Design 2.

08/20/12	Look for providers to buy the parts
08/27/12	Purchase hardware and electronic materials
08/29/12	Begin building box
09/02/12	Frog's box built
09/05/12	Start building infrared barriers
09/07/12	Configure and set it up with its corresponding controller
09/09/12	Start building threshold barrier sensor
09/15/12	Connect and set it up with main controller
09/17/12	Start building the user's control panel system
09/25/12	Configure and set it up with its corresponding controller
09/28/12	Integrate disable/enable switches
09/30/12	Configure and set it up with the Main Control Unit
10/02/12	Begin building Aesthetic LED array
10/04/12	Configure and set it up with its corresponding controller
10/06/12	Testing of each subsystem individually
10/09/12	Connect the input subsystems to the Main Control Unit
10/10/12	Begin programming Main Control Unit
10/12/12	Build the display screen
10/15/12	Build light animation system
10/17/12	Build audio effects
10/19/12	Build aesthetic decorative light system
10/22/12	Integrate outputs to the Main Control Unit
10/24/12	Begin modular testing on Subsystems
10/25/12	Aesthetic LED array completed
10/27/12	Central Control Unit programming completed
10/31/12	Begin Designing the PCB in Eagle
10/30/12	Begin testing of related Subsystems
11/01/12	Complete related Subsystems testing
11/03/12	Begin final presentation preparation
11/04/12	Manufacturing of E-Frog PCB
11/05/12	Begin full integration testing of Subsystems
11/06/12	Complete full integration testing of Subsystems
11/07/12	Committee Crew Search
11/10/12	Testing game
11/12/12	Prepare for final presentation
11/17/12	Prepare final Power Point
11/21/12	Final Presentation
11/23/12	Website Format
11/27/12	Paper Senior Design II
12/03/12	Turn in of Final documentation and Website

PROJECT BUDGET

Since we did not decide to get a sponsored project, our group provided all the money required to overcome our projected budget. The final amount of money needed for the project was divided into three equal amounts. The rough expectation about how much we spent was the amount of \$800, but we are definitely positive to decrease this budget as much as possible for future production since our goal is to make at least two more for us to keep and also, low cost production was one of our targeted areas that we were focused on. We split the money spent into three categories:

- Foreign parts such as the bronze frog, coins, and flipping obstacles bought from South America.
- Electrical/electronic components such as wiring, power supply, sensors, microcontrollers, lighting system, and more that we bought from here.
- Hardware and miscellaneous materials such as wood, tools, blades, glue, and more that we bought it locally as well.

The following table [Table 9] shows the total budget spent:

DATE	NAME	DEVICE	QTY	COST	TOTAL
Jun 7	CARLOS NEYRA	MSP430, LASERS	2,10	\$33.60	33.60
July17	JAIME SALAZAR	LEDS, 7SEGLED DISP	6,6	\$3.21	36.81
July17	ASTRID CRUZ	EMMITER-RECEIVER, transistor	1,1	\$5.20	42.01
July27	ASTRID CRUZ	ARCADE BUTTON,SOLDER IRON	1,1	\$19.67	61.68
Aug23	CARLOS NEYRA	SOUND CHIP, WOOD, PVC	1	\$28.62	90.30
Aug23	ASTRID CRUZ	ARDUINO UNO & RECEIVERS	1,3	\$39.90	130.20
Aug26	JAIME SALAZAR	CABLE DOUBLE CORD,RECEIVERS	1,1,4	\$22.54	152.74
Aug27	CARLOS NEYRA	RECEIVERS & BREADBOARD	3,1	\$6.39	159.13
Aug30	JAIME SALAZAR	RECEIVERS, LASERS & DISPLAY	10,1	\$43.46	202.59
Sep5	CARLOS NEYRA	MAX 7219, LEDES, RED LIGHTS	2,200	\$20.57	223.16
Sep27	JAIME SALZAR	WOOD AND TOOLS	1,1	\$58.00	281.16
Sep30	ASTRID CRUZ	LABELS, TAPE, SCORING KNIFE	1,1,1	13.50	294.66
Oct14	ASTRID CRUZ	SPEAKER, PC BOARD,NPN	1,2,3	\$21.99	316.65
Oct18	CARLOS NEYRA	MAX7219 LED DISPLAY & DRIVER	1,1	\$20.19	336.84
Oct18	CARLOS NEYRA	NEW 1X8X8 Dot Matrix 5mm	2	\$13.85	350.69
Oct25	JAIME SALAZAR	7 SEGMENT LED DISPLAY, SOCKET	2,1	\$5.72	356.41
Oct26	ASTRID CRUZ	Breakout Board FT232RL USB-Serial	1	\$18.59	375.00
Oct26	CARLOS NEYRA	USB Tiny ISP programmer	1	\$10.99	385.99
Nov1	ASTRID CRUZ	PCB EAGLE &Extra for 3 designs	1	\$70.00	455.99
Nov1	CARLOS NEYRA	PCB EAGLE (part of it)	1	\$31.53	487.52
Nov1	ASTRID CRUZ	DIGI KEY (PCB components)	+100	\$100.0	587.52
Nov1	CARLOS NEYRA	ATMEL-ATMEGA 2560	1	\$45.00	632.52
Nov3	CARLOS NEYRA	PC BOARD, PIN IC SKT, SOLID UL	1,1,1	16.50	649.02
Nov3	JAIME SALAZAR	JUMPER, GREEN CARPET, PVC	1,1,1	\$28.84	677.86
Nov14	CARLOS NEYRA	GREEN CARPET, PC BOARD	1,1	\$19.89	697.75
Nov18	CARLOS NEYRA	PVC	1	\$4.00	701.75
Nov19	CARLOS NEYRA	PUNCHED BOARD	1	\$2.65	704.40
Nov21	CARLOS NEYRA	Van gas	1	\$14.00	718.40

Table 9 Table of Expenses

TOTAL BUDGET SPENT: \$718.40

EACH MEMBER: \$239.46

Sponsor/ Significant Contributors

Since this is a project that caught our personal attention at first, we decided to self fund it to keep the interest and to control and be conscious of each component chosen and penny spent. Also, others reasons we chose to sponsor ourselves are because the project total cost is low, and we would like to keep rights on this innovative and unique fun game.



APPENDIX

[Figure3] [<http://www.ti.com/product/lm3s6c11j>]

[Figure4] [http://www.atmel.com/dyn/resources/prod_documents/doc2549.pdf]

[Figure5] [<http://www.alldatasheet.com/view.jsp?Searchword=PIC30F6012A-30>]

[Figure16] [<http://www.tekscan.com/flexible-force-sensors#specifications>]

<http://www.digikey.com/scripts/dksearch/dksus.dll?vendor=0&keywords=gf-1123>

<http://www.doctrionics.co.uk/4511.htm>

http://microcontrollershop.com/product_info.php?products_id=4205

http://pub.ucpros.com/download/serial7seg2_manual.pdf

<http://www.digikey.com/scripts/dksearch/dksus.dll>

<http://www.kpsec.freeuk.com/components/led.htm>

<http://www.purdyelectronics.com/pdf/FE0502.pdf>

<http://fortytwoandnow.blogspot.com/2012/06/mps430g2-tutorial-microcontrollers.html>

<http://www.robotroom.com/DPDT-Bidirectional-Motor-Switch.html>

<http://www.diagram.ly/>

http://www.phoenixdisplay.com/color_display.php

<http://www.purdyelectronics.com/products/displays/segment.cfm>