An-Najah National University

Faculty of Engineering & Information Technology

Computer Engineering Department

Beautiful Times Machine

Shihnaz Nasser **Horia Mahmoud**

Project supervisor:

Dr. Sufyan Samara

Presented in partial fulfillment of the requirements for

Bachelor degree in Computer Engineering - Hardware Project

December, 2021

Table of contents:

Table Of Figures:

Dedication

To the amazing man to my father ...
To the great woman to my mother...
To the lovely brothers and sisters
To my teachers and my friends ...
To Everyone supported me to reach here..

Shihnaz

To my legends in life , my mother and father.
To my family, friends and teachers.

Horia

Abstract

Beautiful Times Machine is a machine for catching toys, using a moving robotic crane controlled by Joystick , that controls the claws movement in three dimensions up down left or right, to help the player to catch the toy he/she chose. Or he/she can play wirelessly using a Bluetooth application on a mobile phone. We used Arduino-Mega to control claws movements.

The player will play for a specified time depending on the coin that he/she entered.For example ,if a player enters 1 Shekel then he/she plays for 1.5 minutes,and if he/she enters 2 Shekels then he will play for 2.5 minutes, ,the more money the player enter the more time that he/she plays,once his/her time is up then the he/she will lose the control ability.

Beautiful Times Machine is an entertainment game that can be used in parks ,moles ,or places where there are children; not only children, adults,and the elderly can play and have fun.It gives you a feeling of a challenge to have a toy or candy, feeling of happiness, and keep trying; that is why we named it "***Beautiful Times Machine***" as it gives us beautiful times that reminds us of our childhood .

Ch1:

Introduction

Since we were kids, we liked to play with claw machines or toys catcher games, in the hope of catching a toy or having candy, giving us beautiful moments of happiness , teaching us to keep trying until achieving and getting what we want. As this amazing machine used to give us beautiful times, we decided to make this beautiful machine to remind us of these beautiful moments in a modern way .

Beautiful Times Machine is a machine for catching toys, using a moving robotic crane ,It is an entertainment machine gives the player fun ,teaching him/her not to give up ,to be patient and keep trying , the player can play using JoyStick or with Bluetooth application for a specified time depending on which coin he/she entered .Our machine supports 4 coins (1 Shekel,2 Shekels,5 Shekels,10 Shekels) and each coin has its own time for example if the player enters 5 Shekels he/she will play for 5 minutes and 30 seconds .

Beautiful Times Machine is useful for both the player and the owner .To put it simply, if you want to buy a traditional claw machine it will cost you 3995$ or more ,but our machine cost one third or less ,the player will have fun and giving him/her the flexibility to choose the playing way either using mobile or using JoyStick.

This report contains the process of developing our project. Starting with the main constraints that faced us and how we handled it , Moving to talk about the similar projects and what differentiates our project . Then in the methodology chapter, we will go deeply into how we implemented our project and what components we used. After that in the results chapter the results of our project are discussed then the whole process is concluded briefly in the conclusion chapter . Finally, The future work chapter is focused on how we will improve our project.

Ch2: Constraints, Standards/ Codes and Earlier course work

Constraints

Z axis Constraints:

As the player can control the movement in the three dimensions ,we faced a problem of how we could implement the movement of the Griper up and down with equal steps. To handle this problem we used a double shaft stepper motor holding the griper from the two sides .

In addition ,when the user press the JoyStick the griper will go down for a specified height opening to catch then closing and go up with the same height ,the problem was how could we know the specified height ,to handle this we used ultrasonic sensor that finds out the distance that the gripper should move up or down .

Playing Time constraint:

Since each coin has its own time ,we faced that the actual time is not the same of the wanted time due to the delays that used to operate the componentes ,so to handle this we added this time to the timing of the actual time to have approximately precise playing time .

Design Constraint:

To make the gaming space clear ,therefore the player can play easily and see clearly, we we made the design nearly big with size of (50*50*70) ,So it was not easy to move it from one place to another.

Standards/Codes:

For the coding:

We used c++ language with arduino ide, achieving the concept of readability using inline comments and choosing suitable variables names ,to make it more understandable .We tried to seperate each functionality in a function to make it easy to maintain.In other words ,we tried to write well documented code.

Earlier coursework :

Since c++ language is used which we learned in the c course ,to control the arduino mega controller which is familiar to the microcontroller course .We tried to achieve the coding principle which we learned from the software course.

Ch3:Literature Review

If you take a look around you ,in the parks ,or in the moles ,or other places you will see different claw machines which are similar to our machine but if you take a deep look , you will find that our project differs from them in different ways .

Our machine provides two ways of playing: using mobile via bluetooth application or using a joystick. Furthermore, our machine has a timer for each coin unlike the traditional ones that do not have a timer that depends on how much money the player puts into it ;As in our machine, in which the player will lose the control ability once his/her time is up ,and if you enter a coin when you are playing you will not lose it ,it will add more time to your playing time .In addition the cost of the traditional claw machine is triple the price of our machine.

Ch4:Methodology

The basic platform that we used to control all modules in our project is Arduino mega 2560 based on easy to use hardware and software. We control the Arduino mega board by uploading different instructions on it .This is what we will explain in this section.

In this section we will show the block diagram that represents the whole project in a simple way. In the first part(Hardware Part) we will talk about each component ,and discuss how we built the circuit for each component too. We finished this part by showing the schematic diagram for the whole project. In the second part(Software Part) we will discuss how we controlled each component and made it works as good as possible

Ch 4.1: The Hardware Part

In the hardware part, we will talk about each component and how we connected each one with the arduino mega controller , and how everything is connected until we reach the final design.

About the components :We used the arduino mega 2560 as a microcontroller to control every component in the project. We also used the arduino uno to deal with the bluetooth component.

And because we need a three-dimensional movement we used 3 stepper motors each motor controls one dimension. And to drive the stepper motors to rotate we used a motor driver for each stepper motor.

We controlled the hand that would catch the toys by using a servo motor. We also used a joystick to operate the motors according to its direction. And we used a coin acceptor that allows the user to enter a coin and then the time for playing starts according to the entered coin. And we used the LCD to display some messages on it.

Arduino Mega 2560

Arduino mega 2560 is a board based on the Atmega2560 microcontroller that contains many powerful features that help us to build complex projects such as serial communication ,crystal oscillator, external interrupts , LEDs …etc.

The arduino mega 2560 has large number of pins compared with other arduino types,such as:

- 54 pins for digital input/output .
- 15 of the 54 pins can be used as PWM outputs.
- 16 analog inputs.

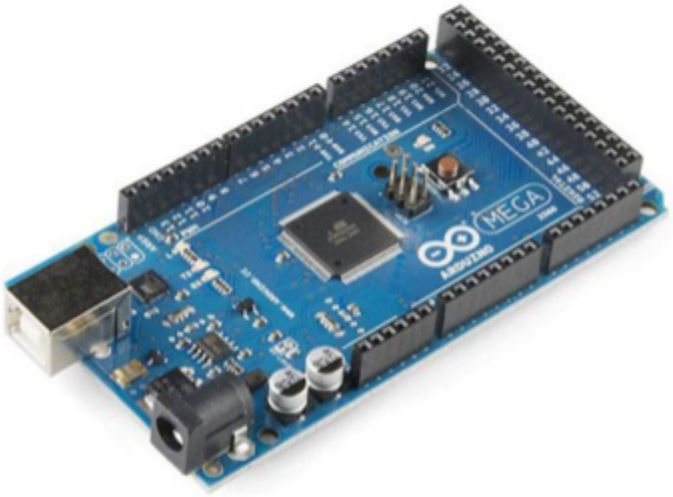

Figure 4.1.1 : Arduino Mega board.

Bluetooth Module HC-05

The HC-05 provides wireless communication with small devices like mobile phones by working on serial protocol (RX & TX) to receive and transmit data.

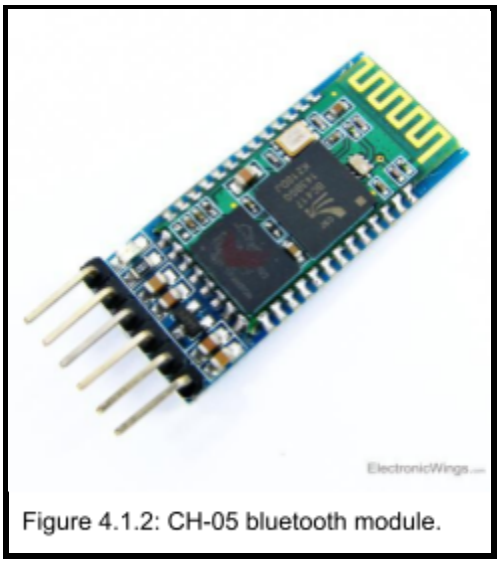

Figure 4.1.2: CH-05 bluetooth module.

And we used it in our project to give the user additional choice to play with our machine by using a mobile phone instead of using the joystick. We used the Arduino Uno to interface the HC-05 using the RX & TX pins in the arduino board with baud rate equal to 9600,Then we connected the arduino uno with arduino mega to get the values from HC-05 and use them to control the motors according to the entered coin. The connection circuit for the HC-05 as shown below:

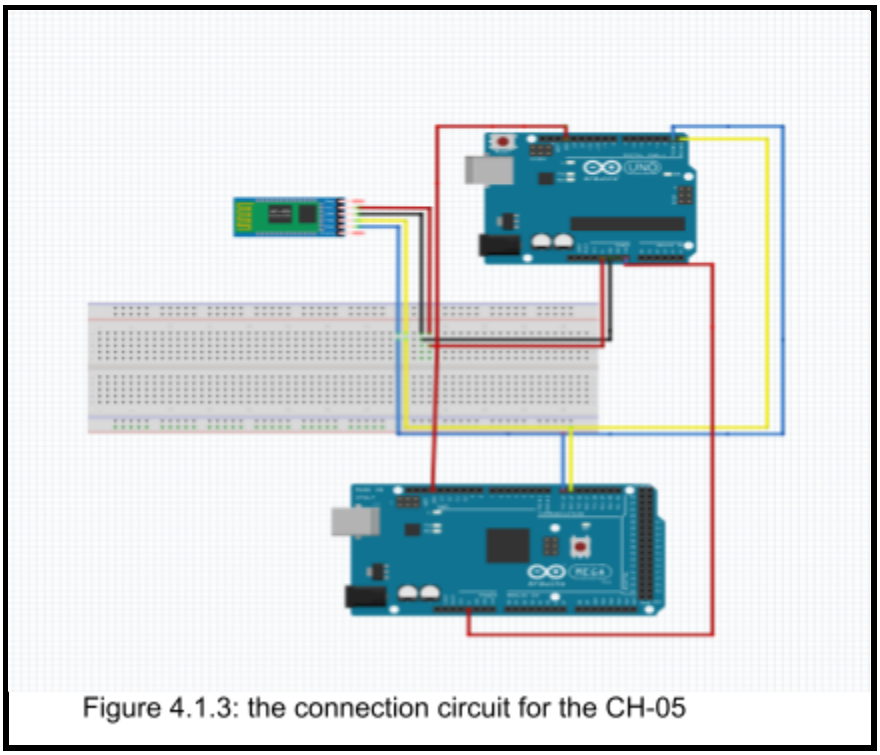

Figure 4.1.3: the connection circuit for the CH-05

A4988 Motor Driver Module

The A4988 Motor Driver used to drive motors by controlling the stepper motor step and direction ,and it has five different step resolutions such as full step, half step …etc.

Figure 4.1.4: the A4988 driver.

Figure 4.1.5: the A4988 driver.

We used in our project three stepper drivers for three steppers

Stepper Motor

The stepper is an electric motor that rotates by moving a fixed amount of degrees after performing the value of steps .

Figure 4.1.6: The stepper motor.

We used in our project three stepper motors to move in three-dimensions ,each stepper controls the movement in a different direction (for example the 1st stepper motor controls the movement in the x-axis). The connection circuit for each stepper as shown below:

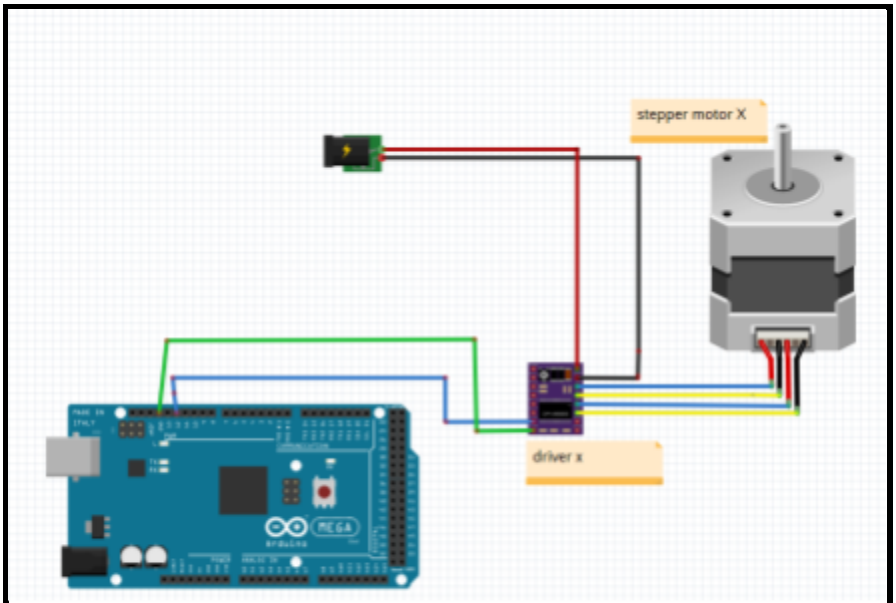

Figure 4.1.7 : the connection circuit for x stepper motor.

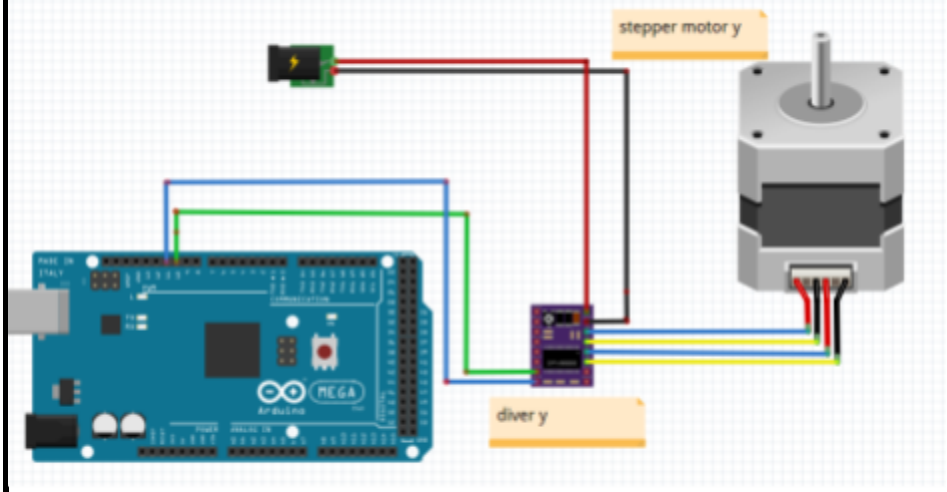

Figure 4.1.8 : the connection circuit for y stepper motor.

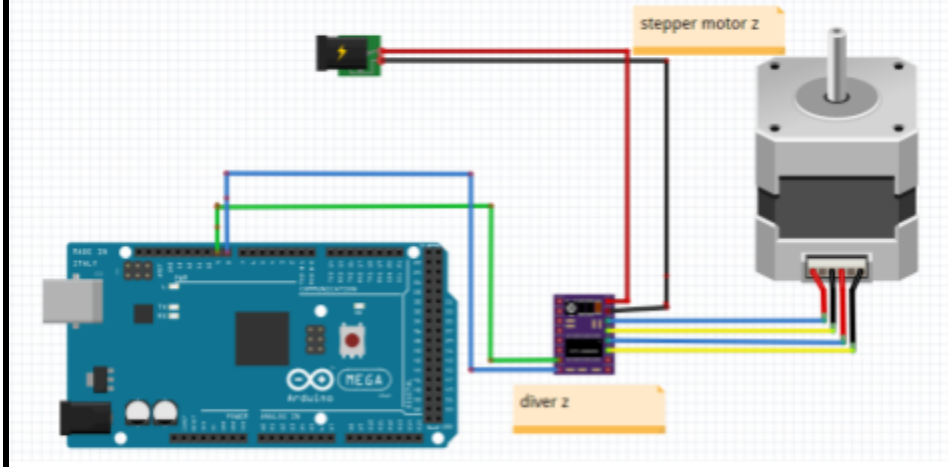

Figure 4.1.9 : the connection circuit for x stepper motor.

SG90 Servo Motor

Servo motor is a small and easy-to-use motor. it can rotate 90 degrees in both directions (180 in total).

Figure 4.1.10 : the servo motor.

We used the servo in our project to control the opening and closing operation for the hand when the user tries to catch a toy . The connection circuit for the servo as shown below:

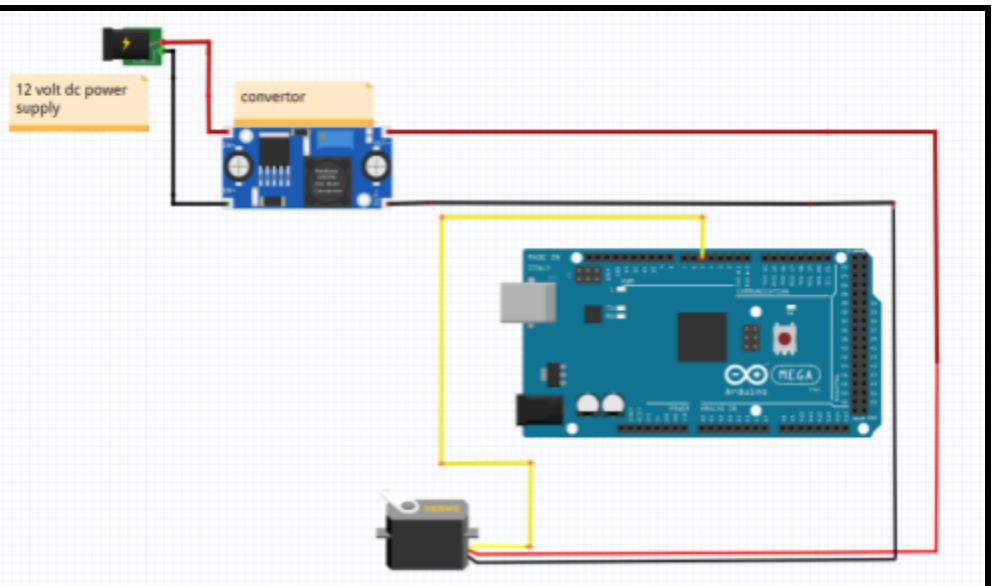

Figure 4.1.11 : the connection circuit for servo motor.

JoyStick

A joystick is an input device which is used to control gaming applications.

Figure 4.1.12 : the joystick..

We used the joystick in our project to determine the direction range for each stepper motor to work on , if the user moved the joystick to the right or left (x-axis) the stepper x will work only , and if the user moved the joystick up or down (y-axis) the stepper y will work only ,and if the user pressed the joystick button the stepper z and the servo motor will work. the connection circuit for the joystick as shown below:

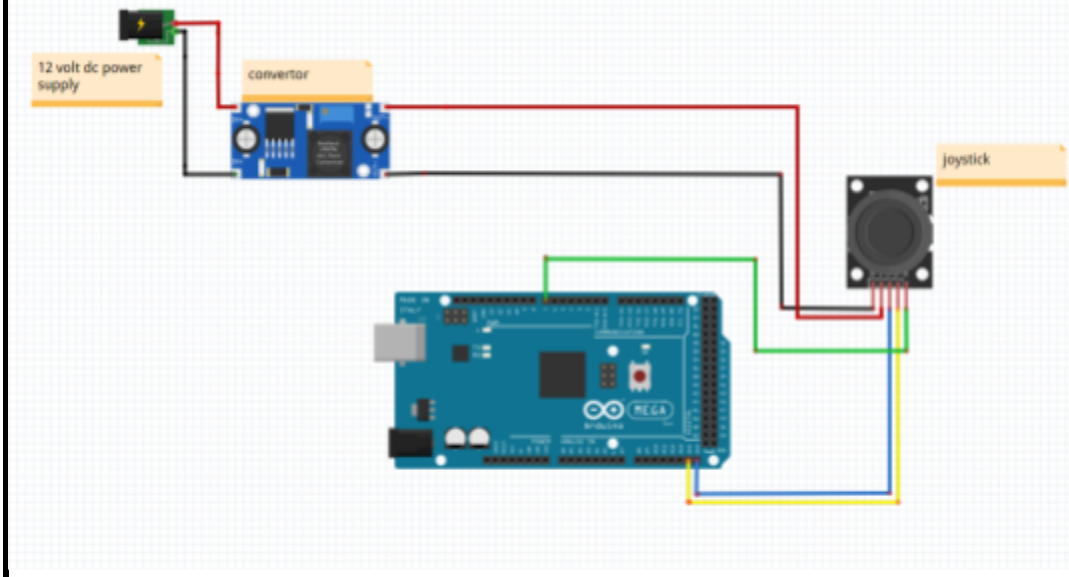

Figure 4.1.13 : the connection circuit for joystick.

Coin Acceptor CH-926

The CH-926 accepts 1-6 types of different coins at the same time by setting the number of samples and pulses for each coin type during configuration; it also accepts any world coin.

Figure 4.1.14 : the coin acceptor CH-926.

The coin acceptor in our machine accepts 4 coins (1 Shekel,2 Shekels,5 Shekels,10 Shekels) and each coin has its own time. For example if the player enters 2 Shekels he/she will play for 3 minutes. If the user entered an accepted coin the playing time starts and the player can play until the time ends. And if the player enters a coin that is not accepted, the machine will reject this coin.the connection circuit for the machine as shown below:

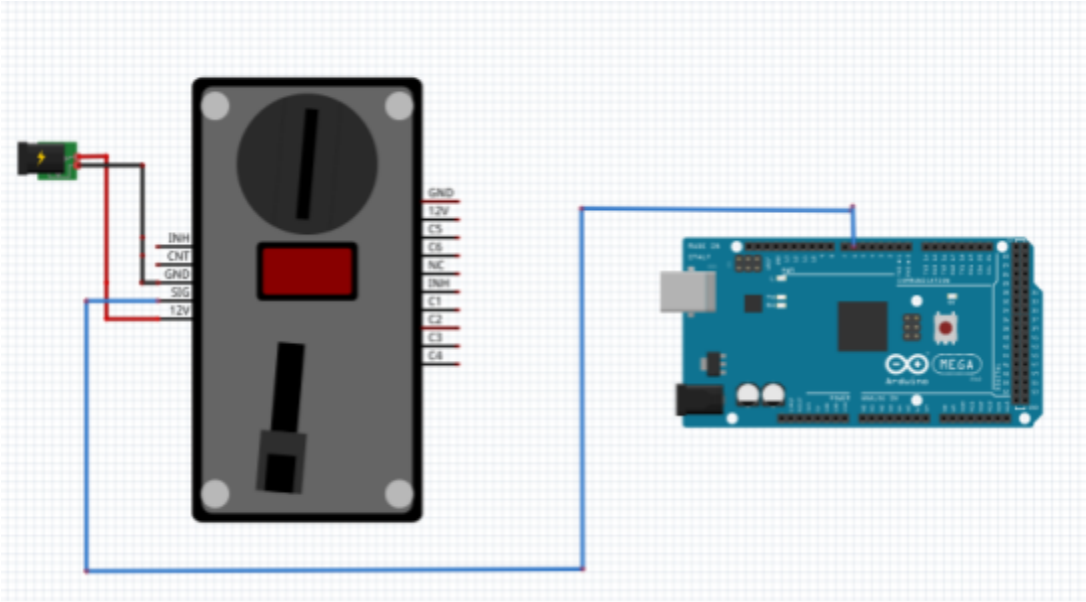

Figure 4.1.15 : the connection circuit for CH-926.

LCD 16* 2

The LCD is an output module which has sixteen columns and two rows.

Figure 4.1.16 : the LCD 2*16..

In our project we used the LCD module to display some messages for the user .It displays “Hello everyone” by default ,when the playing time starts the LCD shows “hi your playing time starts now” , and when the playing time ends the LCD shows”Bye your playing time ends”.

Figure 4.1.17 : the messages on LCD.

The circuit for the LCD in our project as shown below:

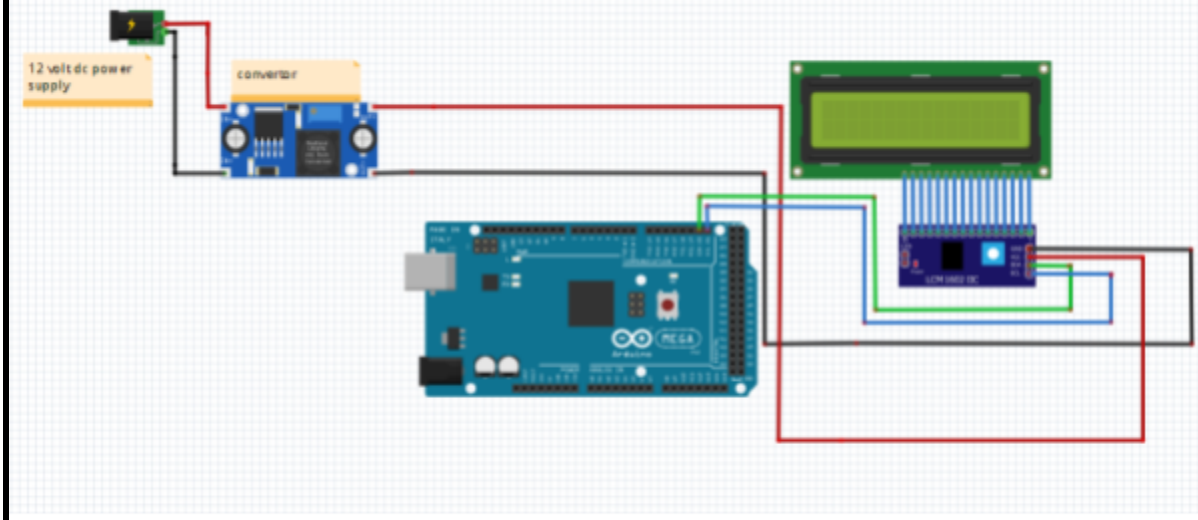

Figure 4.1.18 : the connection circuit for LCD 2*16.

HC-SR04 Ultrasonic Sensor

The ultrasonic is a sensor which is mainly used to calculate the distance between the sensor and the target object with high accuercy.

Figure 4.1.19 : the HC-SR04.

We used this sensor in our project to determine the distance between the hand and the toys in the machine ,so when the user starts playing the z-axis will move up and down

according to the distance measured by the sensor.in order to guarantee that the player can catch a toy whatever the height of the toys in the machine. The connection circuit for this sensor as shown below:

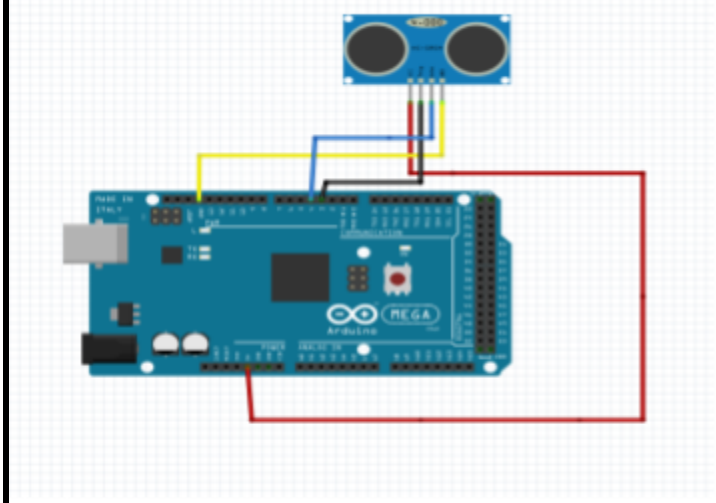

Figure 4.1.20: the connection circuit for ultrasonic sensor..

After discussing how we connecting and using each component in the project , we are going to show the **Schematic Diagram for the whole project as shown below :**

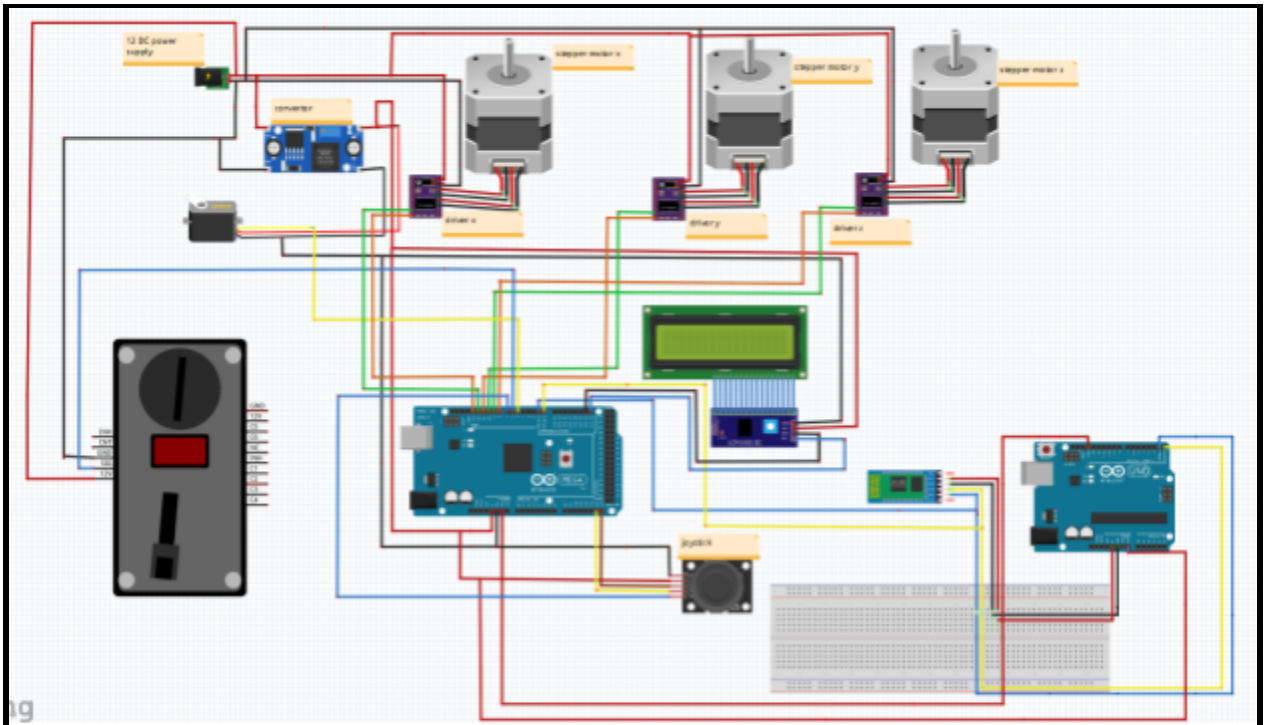

Figure 4.1.21 : the schematic diagram.

This diagram explains what the components that we used and how all the components are connected and which pins in each module used to make the connections . This helps us to take a general view about the circuit on a small scale.

Ch 4.2: The Software Part

As we explained above how we connected each component in our project . In this section we will explain some of the functionalities in our code .

Reading from JoyStick:

```
void stepper(){

int val = analogRead(VRx);// read analog x value from the  pin A15 in arduino
//     Serial.print("X: ");
//     Serial.print(val);
//     delay(1000);

//y
int val_y = analogRead(VRy);// read analog y value from the  pin A14 in arduino
//     Serial.print("y: ");
//     Serial.print(val_y);
//     delay(1000);

////z
int val_z = digitalRead(SW);// read digtal z value from the  pin 7 in arduino
//    delay(1000);
//    Serial.print("z: ");
//    Serial.print(val_z);
//
```

Figure 4.2.1: the code for reading from the joystick.

Depending on these values we determined which stepper should be on and in which direction . For example for the x-axis we determined when to turn it off ,and when to turn it on cw and ccw depending on the x value as shown in the figure below

BeatifulTimesMachine §

```
//
if((val >= 540) && (val <= 680) )// if the joystick is in the middle ===> stop the motor

{
  digitalWrite(stepPin, LOW);// stop
}
else
{
while ((val >= 0) && (val <= 400) )//if the values are in this range then move the steper motor of the Xaxis  clockwise --- high ////left
{
      digitalWrite(dirPin, HIGH);

      //digitalWrite(stepPin, HIGH);
for(int x = 0; x < stepsPerRevolution; x++)
  {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(100);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(1000);
  }
val = analogRead(joystick);///read the value again to check if it is still pressed
}
while ((val >= 700) && (val <=  1023))// move the motor in the other direction //if the values are in this range then move the steper motor of the Xaxis --- low////ri
{
      digitalWrite(dirPin,LOW);

        //digitalWrite(stepPin, HIGH);
for(int x = 0; x < stepsPerRevolution; x++)
  {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(100);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(1000);
  }

val = analogRead(joystick);//read the value again to check if it is still pressed
}
```

Figure 4.2.2: the code for determining which stepper should be on.

With The same concept we determined when to turn it off, and when to turn it on cw and ccw depending on the y value as shown in the figure below

```
BeatifulTimesMachine §
if((val_y >= 570) && (val_y <= 700) )///// stop the motor if joystick im middle

{
  digitalWrite(stepPin_y, LOW);}// stop
else
{
while ((val_y >= 900) && (val_y <=  1023))//// move the  steper motor y in the other direction // cclockwise --- low
{
      digitalWrite(dirPin_y,LOW);

        //digitalWrite(stepPin, HIGH);
for(int x = 0; x < stepsPerRevolution_y; x++)
  {
    digitalWrite(stepPin_y, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin_y, LOW);
    delayMicroseconds(1000);
  }
val_y = analogRead(VRy);/////read the value to check if it is pressed again
}

while ((val_y >= 0) && (val_y <= 400) )//// move the steper motor in the first direction // clockwise --- high
{
      digitalWrite(dirPin_y, HIGH);

      //digitalWrite(stepPin, HIGH);
for(int x = 0; x < stepsPerRevolution_y; x++)
  {
    digitalWrite(stepPin_y, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin_y, LOW);
    delayMicroseconds(1000);
  }


val_y = analogRead(VRy);
}
```

Figure 4.2.3: the code to determine when to turn the stepper off, and when to turn it on cw and ccw

Then if the player press the sw button in joystick the servo will be opened then the stepper motor of the Z-Axis goes down for a specified height depending on Ultrasonic

reading then catch by opening servo motor and close it again and goes up as shown below :

```
 for(pos=180;pos>=0;pos--){/////open the servo motor again completly
Myservo.write(pos);
delay(15);
}
delay(2000);//open it for two seconds
    while(countup <= ultraResult())
{
       digitalWrite(dirPin_z,HIGH);

for(int x = 0; x < stepsPerRevolution_z; x++)
  {
    digitalWrite(stepPin_z, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin_z, LOW);
    delayMicroseconds(1000);
    Serial.print(x);

  }
 countup++;// increament counter to go back to the specified height
}  // Serial.print("pressed");
           val_z = digitalRead(SW);// read the value again to check if it is pressed

    }

}
`   }
```

Figure 4.2.4:The code for controlling the catch operation using the servo motor.

```
 for(pos=180;pos>=0;pos--){/////open the servo motor again completly
Myservo.write(pos);
delay(15);
}
delay(2000);//open it for two seconds
    while(countup <= ultraResult())
{
       digitalWrite(dirPin_z,HIGH);
for(int x = 0; x < stepsPerRevolution_z; x++)
  {
    digitalWrite(stepPin_z, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin_z, LOW);
    delayMicroseconds(1000);
    Serial.print(x);

  }
 countup++;// increament counter to go back to the specified height
}  // Serial.print("pressed");
           val_z = digitalRead(SW);// read the value again to check if it is pressed

    }

}
  }
```

Figure 4.2.5: The code for controlling the movement of stepper motors with Joystick.

With the code above we controlled the movement of stepper motors with Joystick inside the“**stepper()”**function which will not be called if you don't enter money , or your time ends . moving to Coin acceptor that accepts only 4 coins (1,2,5,10 Shekels)and reject any other coins this programing was isolated from the code “manually”. Then we used interrupts to check if the player enters money while playing so he/she will not lose them adding them to his /her time .As shown in the code below :

```
Stepper stepper(stepsPerRevolution, stepPin,  dirPin);
//y
Stepper stepper_y(stepsPerRevolution_y, stepPin_y,  dirPin_y);
//z
Stepper stepper_z(stepsPerRevolution_z, stepPin_z,  dirPin_z);

void setup() {

 Serial.begin(9600);

 attachInterrupt(digitalPinToInterrupt(coinIntPin),incomingImpuls, FALLING);
  pinMode(joystick, INPUT);
  pinMode(VRy, INPUT);
  pinMode(SW, INPUT_PULLUP);
  pinMode(stepPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
  pinMode(stepPin_y, OUTPUT);
  pinMode(dirPin_y, OUTPUT);
  pinMode(stepPin_z, OUTPUT);
  pinMode(dirPin_z, OUTPUT);
  digitalWrite(SW,HIGH);
  Myservo.attach(3);
```

```
  void incomingImpuls()
  {
    impulsCount=impulsCount+1;
    Serial.println(impulsCount);
    time=0;
    newCoin = true;
    i=0;
  }
  int prevalue=0;
  int pretotal,total=0;

  void loop() {
     i=i+1;
     if(i>=1){//to ensure it enterd the interrupts function
    if (  abs( prevalue-impulsCount)==1){////check if the enterd impulse is 1 impulse as 1 shekel is on
      Serial.println(" 1 sh");
      total=total+1;//increase the total money by 1
      impulsCount=0;

    }
     if ( abs(prevalue-impulsCount)==2){////check if the enterd impulse is 2 impulse as 2 shekel is two
      Serial.println(" 2 sh");
      total=total+2;//increase the total money by 2
      impulsCount=0;
    }

     if ( abs(prevalue- impulsCount)==3){////check if the enterd impulse is 3 impulse as 5 shekel is th
      Serial.println(" 5 sh");
      total=total+5;//increase the total money by 5
      impulsCount=0;

    }
     if ( abs(prevalue- impulsCount)==4){////check if the enterd impulse is  4 impulse as 10 shekel is
      total=total+10;
      impulsCount=0;
              Serial.println(" 10 sh");
    }
```

Figure 4.2.6: The code for controlling the coin acceptor.

After finding which coin is entered, the timer starts for a specified time depending on the entered value ,calling one of the functions Steppers that are controlled by joystick or controlled by bluetooth .

The bluetooth mode using android application that each button in it represents a number sending it to HC05 that sends it to arduino then depending on this number we controlled the movement of the steppers as shown in the following code :

```
void    bluetooth() {
     if (Serial3.available() > 0) { // Checks whether data is coming from the serial port
    state = Serial3.read(); // Reads the data from the serial port
        }

while(state=='1')  // if the player pressed on  right
{      t+=1; /// turn the steper motor x to move left for full iteraition
       digitalWrite(dirPin,LOW);
       for(int x = 0; x <stepsPerRevolution; x++)
  {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(100);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(1000);
    Serial.println("right1");
  }
 if ((t==2 )||(Serial3.available() == 0)){ state=0;t=0;// if the iteration ends or the conection lost turn it off and show lost connection message
break;
}}

while(state == '2' ) {// if the player pressed on left
      digitalWrite(dirPin, HIGH);
       t2+=1;/// turn the steper motor x to move right for full iteraition
     for(int x = 0; x < stepsPerRevolution; x++)
  {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(100);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(1000);
    Serial.println("left 2");}
  if (t2==2||(Serial3.available() == 0)){ state=0;t2=0;// if the iteration ends or the conection lost turn it off and show lost connection message
break;
}|
```

Figure 4.2.7: The code for controlling the steppers according to bluetooth value .

With the same concept we checked the other buttons on the application controlling the steppers movement . For Lcd part it will print “Hello everybody” when no one is playing,and it will welcome the player again when his/her time starts printing “Hi,Your Playing Time Starts from Now” and telling him/her when his/her time finished printing “Bey ,Your Playing Time ends”.

Ch 5:Results and Discussion:

With all of the steps we did above regardless of the challenges that we faced as working with different components for the first time,and to the constraints that faced us and how we solved it , , finally we finished developing our project “Beautiful Times Machine “ , the entertainment machine with two playing methods .

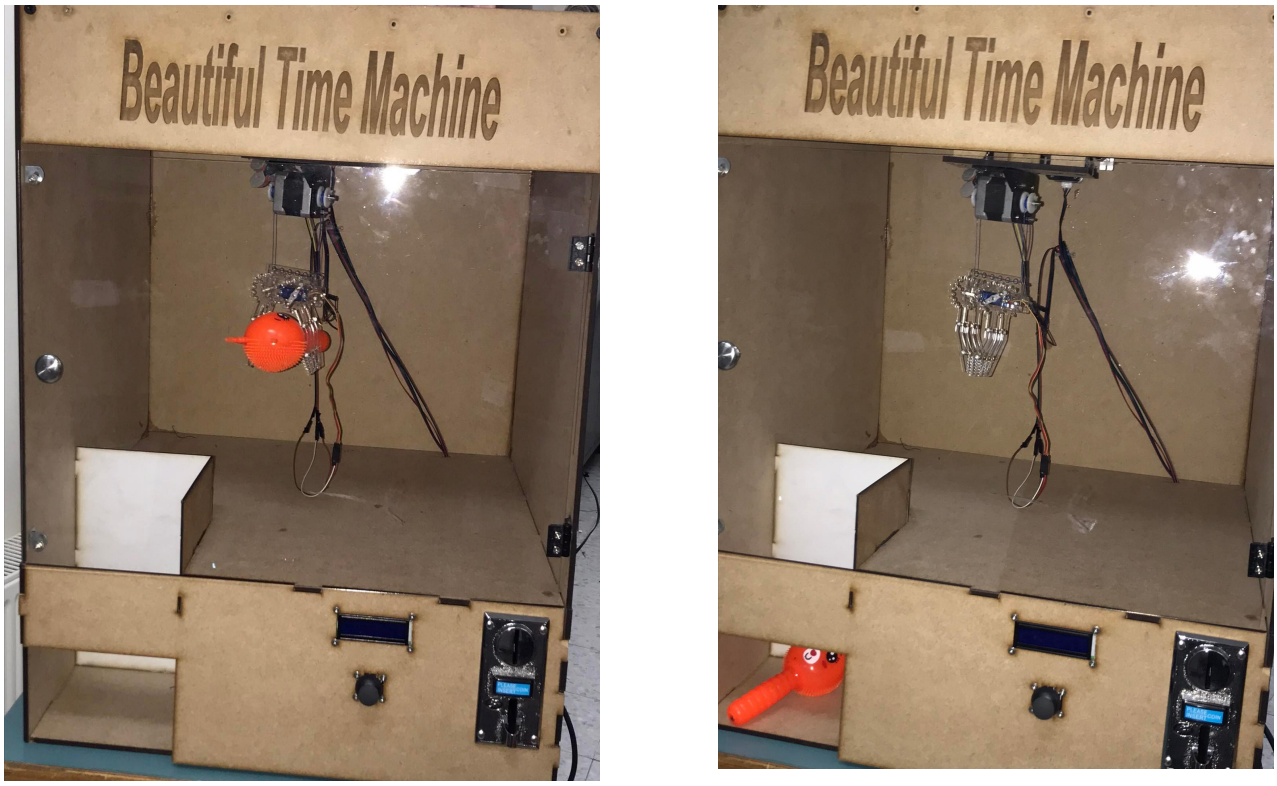

Figure 5.1: The final design of our machine.

Ch 6:Conclusion

Conclusion:

To sum up all of what we mentioned above , we designed a toy catcher machine with two playing modes: with a joystick or with a bluetooth application ,in which the user plays for specified time having a mixture of feelings of happiness and challenge.
Beautiful Times Machine is entertainment machine ,which gives the player beautiful time , having fun and teaching him/her to not give up .These beautiful times remind him/her of the beautiful moments of childhood ,doing this we achieve our hidden goal which is making others happy with what we did as someone said ***"Happiness is felt by making other people happy*** ".

Future Work :

Unfortunately we didn't have enough time to do everything ,some of the features are delayed to the near future ,one of them is the music while the player is playing a motivation music will be turned on during the playing time to give the player more enjoyment .
Another thing is the payment method. We would like to make it more flexible in the future, for example when the player wants to play with one Shekel and enters 5 Shekels to give him back 4 Shekels or further more to have different ways of payment such as using a credit card.
In addition, we would like to expand the machine field from entertainment to include other fields like educational fields, for example teaching kids to sort the alphabets in boxes , or in chemical purposes as puting chemical material in specified places as some of them are dangerous to hold in hand .Finally ,Beautiful Times Machine is not only atoy catcher it can be used in many different fields making them more fun and easier .

Attachment A:

Disclaimer

This report was written by students (Shihnaz Nasser and Horia Haji) at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.