

B Blueprinted

'HEART MUSIC BOX'
TECHNOLOGY REPORT

Blueprinted Technology Company

DIRECTOR

Prof. Yanhui Chen

Group 4

B39VS System Project



Name`	Post
Haoxuan Du	Project Manager Electronic Engineer
Zeyu Xia	Chief Digital Systems Engineer
Ruoxi Gao	Chief Public Relations Software Engineer
Yuze Sun	Secretary Mechanical Design Engineer
Senpu Yuan	Web Designer Mechanical Assembly Engineer
Yutong Xu	Product Publicist Workshop Engineer

Contents

1. INTRODUCTION.....	1
1.1 Project Introduction.....	1
1.2 Achievement.....	1
1.2.1 Meeting customer needs.....	1
1.2.2 Mechanical design and fabricate.....	1
1.2.3 Software design and implement.....	1
1.2.4 Joint debugging.....	1
1.2.5 Low cost.....	2
1.3 Acknowledgements.....	2
1.4 Document Overview.....	2
2. MECHANICAL DESIGN.....	3
2.1 Design Process.....	3
2.1.1 Design Requirement.....	3
2.1.2 The fist attempt.....	3
2.1.3 Problems of new design.....	3
2.1.4 Feedback.....	4
2.1.5 Design concept.....	4
2.2 Newl Design.....	10
2.2.1 Concept model.....	10
2.2.2 Assembly Introduction.....	11
2.3 Entities operation.....	17
2.3.1 Shell	13
2.3.2 Supply disc	13
2.3.3 Shipment disc.....	13
2.3.4 Lifting system	13
2.3.5 Card Reader.....	13
2.4 Final Mechanical drawing.....	23
2.5 Finished Product.....	25
2.6 Operation instructions.....	25
3. Digital Design.....	27
3.1 Introduction.....	27
3.2 Basic design concept	27
3.3 Designing process.....	27
3.3.1 Set states for each motor.....	27
3.3.2 Define output variables.....	28
3.3.3 Use if statement to select different motor.....	28
3.3.4 Select hte state of the motor.....	30
3.3.5 Set the pin.....	31
3.3.6 Load the code to The FPGA.....	31
3.3.7 Lcate the data lines.....	31
3.3 ASM Chart.....	32
4. Software Design.....	33

System Project Technology Report
Group 4

4.1 MBED.....	37
4.1.1 MBED Introduction.....	33
4.1.2 Functions of MBED.....	33
4.1.3 Code Structure.....	34
4.1.4 Functions in Program.....	37
4.1.5 Communication protocols.....	37
4.2 PC.....	37
4.2.1 PC Introduction.....	37
4.2.2 Maintenance Mode.....	38
4.2.3 Operation Mode.....	40
4.2.4 Database.....	46
4.2.5 Communication.....	46
4.2.6 Mechanism of Error Processing.....	46
4.2.7 Other functions.....	46
5. Sensors.....	55
5.1 Color Sensor.....	55
5.1.1 Measuring and Modulation.....	55
5.1.2 Algorithm.....	56
5.2 TOF sensor.....	57
5.2.1 Modulation.....	57
5.2.2 algorithm.....	58
5.3 Infrared light sensors.....	59
5.3.1 simple introduction.....	58
5.3.2 modulation.....	58
5.3.3 algorithm.....	58
5.4 Circuit Diagram.....	59
6. Marketing and Web.....	61
6.1 Market analysis.....	61
6.1.1 Overview of Product Market Developemnt.....	46
6.1.2 Analysis of Current Product Status.....	46
6.2 Product advantage.....	61
6.3 Product publicity.....	62
6.3.1 Poster.....	46
6.3.2 Promotional Video.....	46
6.3.3 Website.....	46
6.3.4 Exhibition Stand.....	46
6.4 Team Member Biography.....	62
7. Project Management.....	65
7.1 Task Routine.....	65
7.2 Project Plan.....	66
7.3 Project risk.....	67
7.4 Team work.....	68
7.5 Meetings.....	69
7.6 Experience and Lessons.....	69
8. Appendix.....	71

System Project Technology Report
Group 4

8.1 Meeting Minutes.....	71
8.2 Risk Management.....	83
8.3 Author of this Document.....	89

System Project Technology Report
Group 4

1. INTRODUCTION

1.1 Project introduction

This product is called Psychic Music Box. It is intended to give players a gift of blessings by playing the music selected by the user and putting forward questions and answers according to the corresponding music, so as to ease their emotions.

During the holidays to play in the playground, some popular entertainment items inevitably have to queue up for a long time, which often makes people feel anxious and bored. This music game box is a good medicine for healing your mind and alleviating your troubles in the queuing process. You can choose two kinds of emotions, happy or anxious, and then choose the music style you like, whether it is R&B, Bruce, Jazz or tranquil folk songs. It can meet your needs here. Finally, it has our carefully prepared heart. Ling Jin Bao, a small note carrying the warm "soul chicken soup" and blessing words, will surely bring you unique happiness and understanding in your entertainment life.

1.2 Achievement

1.2.1 Meeting customer needs

We held many meetings with the client after the project started. Report our progress to client in real time and show our results. The development of the meeting can let us have a clearer understanding of the client's needs. During this period, the client put forward a variety of ideas. After comprehensive consideration, we finally finalized the current plan.

As long as the customer is dissatisfied, we will modify the scheme to meet the customer's requirements. Try to reach a final result satisfactory to both parties.



(1)



(2)

Figure 1 Meeting (1), (2)

1.2.2 Mechanical design and fabricate

Our chief mechanical designer used CAD, C4D and other software to produce our mechanical design drawings. We use two disks to realize the function of delivery and replenishment. This not only saves a lot of space, but also ensures the speed of delivery and the accuracy of replenishment. Because of the accuracy of the drawing, there was no problem in assembly. The assembly progress was very smooth, and the final assembly of the

physical object reached the standard on time.



Figure 2 Double disk structure

1.2.3 Software design and implement

Our team designed MBED with c language and FPGA with C#. MBED as a “CPU” to receive information from motors, sensors, and PC. Then MBED issues instructions to control the operation of the entire system. FPGA is used to control the state of the motor. Adjust the speed and angle of the motor. The PC terminal is written by C #. The PC terminal receives the information input by the user, such as the choice of emotions and the choice of questions' answers. After that, the PC terminal sends the received information to FPGA, and FPGA processes the information.



Figure 3 MBED

1.2.4 Joint debugging

Joint debugging is the most time and effort consuming step. We met all kinds of problems, such as ground fault, wiring interference and so on. We went through hundreds of modulations and finally ensured the stability and user experience of the whole system.

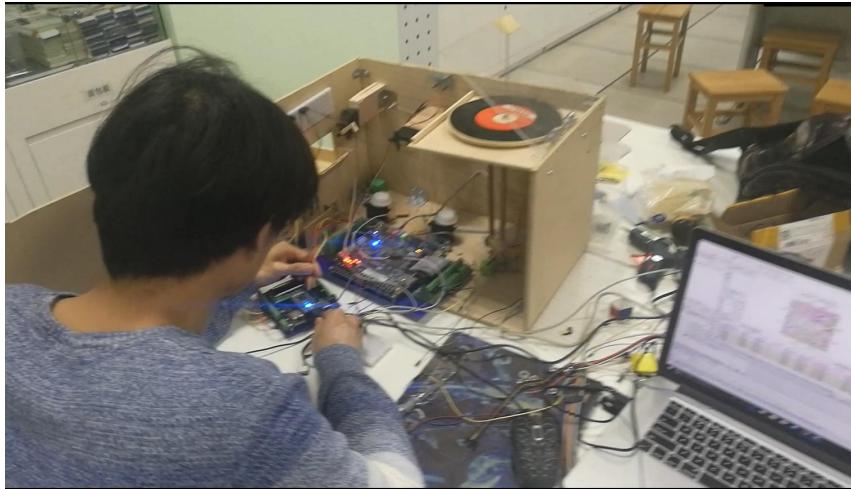


Figure 4 Joint debugging

1.2.5 Low cost

Apart from the FPGA and MBED, the product only needs the wood-based body, the internal dual-disk structure, the motor and the decorative structure. The overall cost is about 370 RMB, which is very economical.

1.3 Acknowledgments

Thank our three teachers here. They raised many questions that we didn't expect and helped us to improve various schemes. Without their guidance, it will be very difficult for us to finish the project on time.

Prof. Yanhui Chen	Project Manager
Weidong Xu	Project Management Lecturer and support staff
Huanfeng Liu	C#, MBED, digital system, sensor Lecturer and support staff

1.4 Document Overview

This introduction explains the composition of our project. It includes mechanical composition, software compilation, modulation process, publicity, market research, and competitive advantage.

2. MECHANICAL DESIGN:

2.1 Design Progress

2.1.1 Design Requirement

The followings are the requirements on mechanics structure:

- The size of the machine is required to be 300mmX300mmX500mm.

- The size of the ID card has some special requirements.
 - All electronic systems should be contained in this structure: FPGA, MBED, Color sensor, Distance sensor, infrared sensor and servo motors, DC motors.
 - The machine should be intelligent enough so that when a player release a instruction, it could make a series of response.

2.1.2 The first attempt

At the beginning, our team decides to produce an self-service shooting machine which can let customers shoot to get corresponding points and then get different gifts. The design drawings are like these:

2.1.2.1 Concept drawing

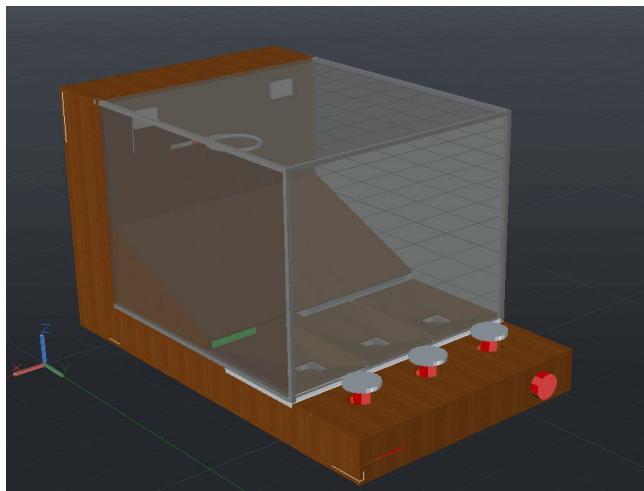


Figure 5 Concept model of shooting machine

2.1.2.2 dimensional drawing

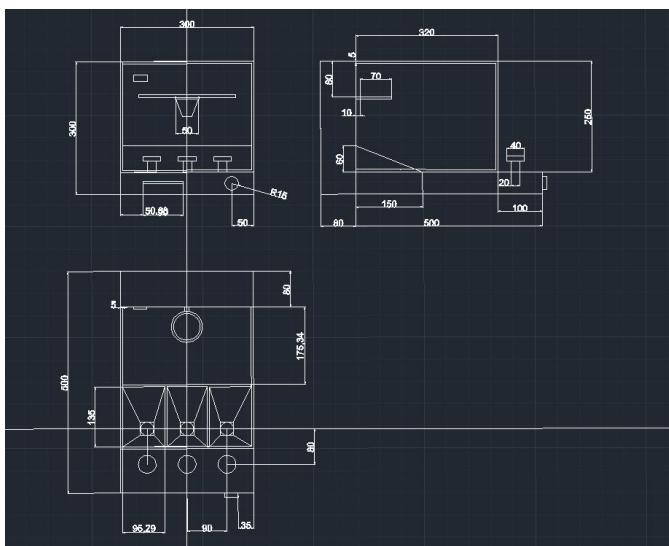
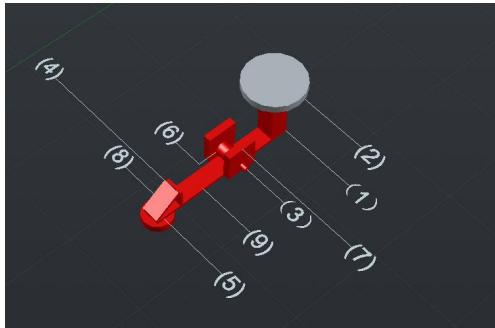
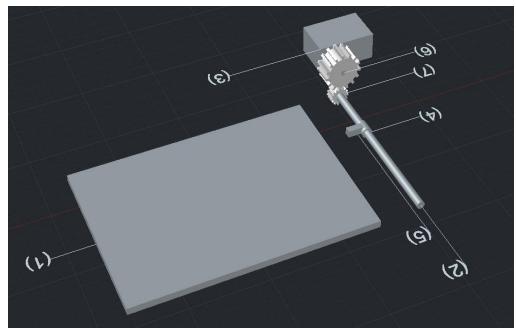


Figure 6 Dimensional drawing of shooting machine

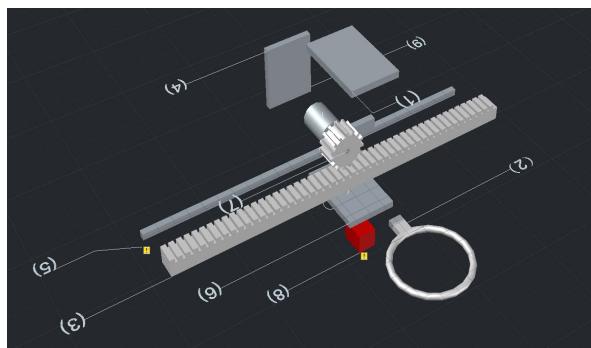
2.1.2.3 Part drawing



(1) .ejection lever



(2). moving flap



(3).moving basket

Figure 7 Parts model of shooting machine

However, in the Q&A session, customers give new requirement. The machine should be a robot with some recreational function and players do not have to do a lot of work. If received an instruction, it should do a series of work to display intelligence. Players must have physical interaction with the machine but not too much. After receiving this message and talking with our teacher, we have to give up our original thought and find a new solving method.

2.1.3 Problems of new design

During discussion, every member has his/her own opinion about the robot from appearance to function. And we find that this discussion manner is of no significance as we have met many issues during discussion. So we have to ask the lecturers some problems so that our design can meet with the requirement of this course. The problems are as follow:

1. the condition limitation of using color sensors;
2. In addition to the motors provided in the course, can we use more servo motors and can we choose motors of different version and power.
3. Can we design the structure to be 300mm high, 300mm width and 500mm length?
4. What is the real function that this machine should have, display a performance automatically or just to be a recreational machine.

2.1.4 Feedback

Through consulting in the project Q&A session, we have received the answers we want.

1. the color sensors must be used in a random event, which means that the outcomes can not be determined by people. The specific manner depends on your own project.
2. We could use more servo motors of different types if we include them in our total cost.

3. We can design our machine in any shape no matter horizontal or vertical.
4. This machine must be intelligent enough, making a series of operations after receiving instructions.

2.1.5 Design Concepts

Basing on the requirements, the design concepts were determined as following.

1. Saving Space

It means that the machine structure should fully use space and avoid overcrowding.

2. Simplicity

The simpler structure means low cost, easy maintenance and low fault rate.

3. Entertainment

It should be full of enjoy and can help customers relax themselves.

4. Appearance:

Nobody want to stop in front of an ugly or strange robot for a long time. So decorate it with heart to make it beautiful and brief.

2.2 New Design

According to the new requirement, we decided to design a robot named "Heart Music Box" which is intended to give players a gift of blessings by playing the music selected by the user and putting forward questions and answers according to the corresponding music, so as to ease their emotions. In terms of the physical structure, it mainly has three parts: induction part, performing part, shipment and supply part. In the process, they work together to show the best result so that customers can get what they want.

2.2.1 Concept Model

2.2.1.1 Overall Diagram

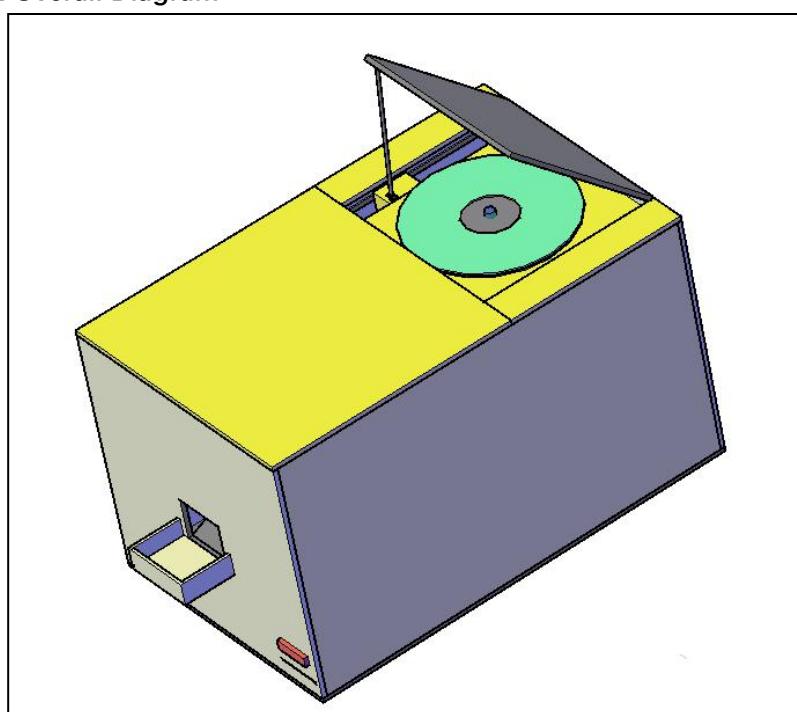


Figure 8 Concept model of 'Music Box'

From the figures, you can clearly see the basic structure of our product, it is an closed machine with exquisite internal structure. For the purpose of creating a simple and mystic sense, we design such closed space and the lid can be lifted automatically during playback, making players feel surprised. The shipment port is just 8cm wide and 6cm high, so you do not know what the gifts are and have more interest to try the game.

2.2.1.2 inner diagram

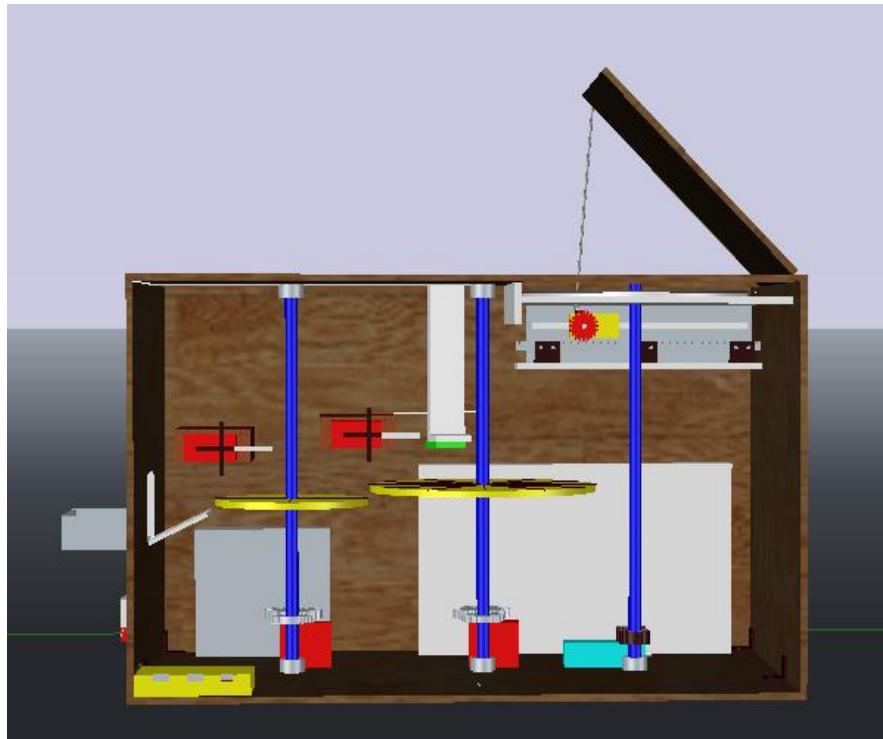


Figure 9 Inner structure of original design

This is the inner structure of the machine. As discussed before, we divide it into three different parts: induction part, performing part, shipment and supply part. Induction part includes distance sensor, infrared sensors(card reader) and color sensor. Performing installation includes rotating turntable and lifting device which can lift up at the show. As for the gift device, we design this part with great care and have done many test for its stability. As shown in the figure, there are two disks with different size, the diameter of small disk is 12 cm and the diameter of large disk is 18 cm. The former is used for shipment and the later is designed to realize automatically sorting function and supply deficient goods. The rotating turntable simply acquires energy from DC motor. Before performing, the lifting device will operate. Later we will discuss the details of each part.

2.2.1.3 dimensional drawing

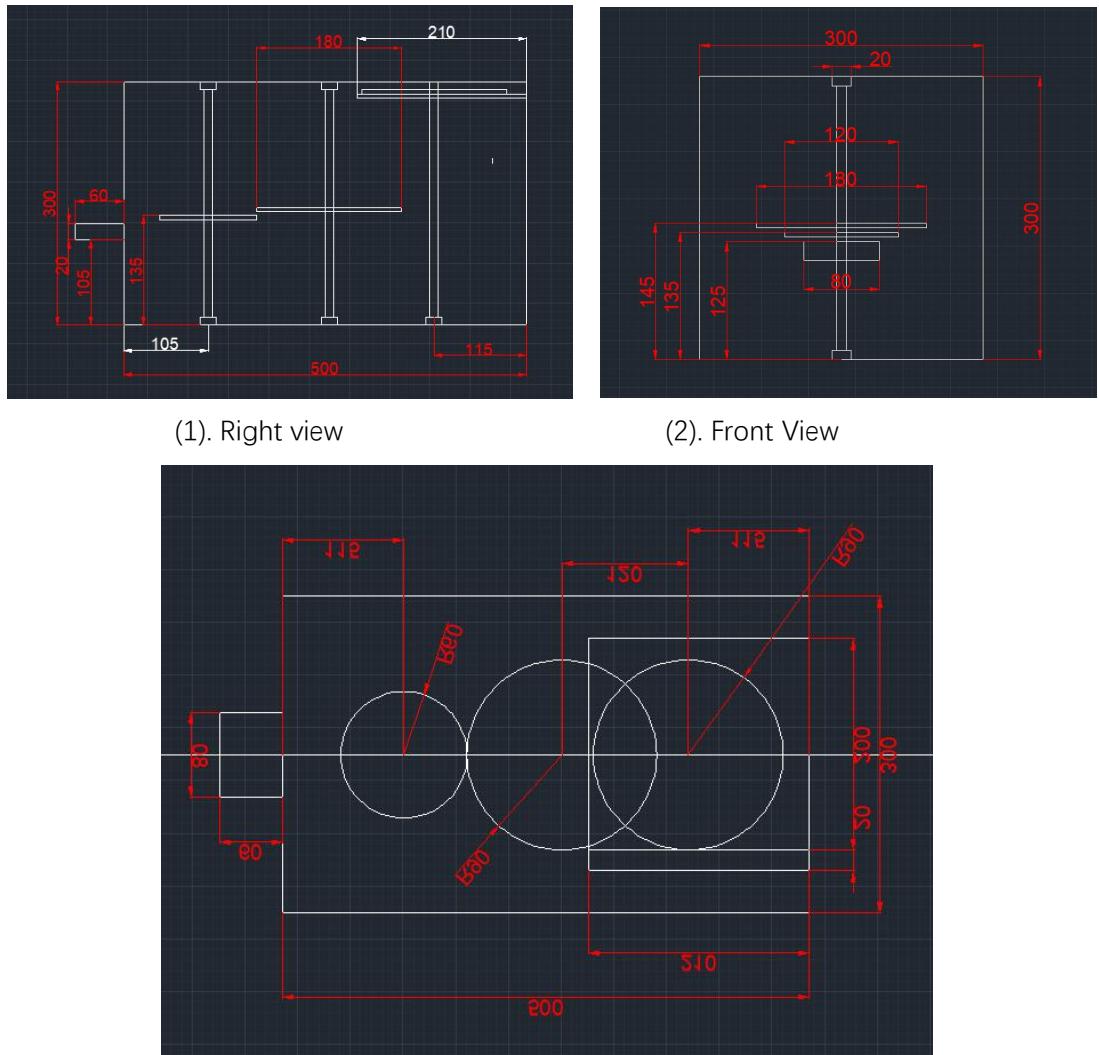


Figure 10 Dimensional drawing of 'Music Box'

These figures just demonstrate the rough size of our machine. During the process of assemble and debugging, a lot of adjustments have been made. So in our original design, many parts have been modulated and we do not discussed them more here as it has no significance.

2.2.2 Assembly Introduction

In this part, we will discuss how we design these different portions and how we assemble them together.

2.2.2.1 Overall assembly drawing

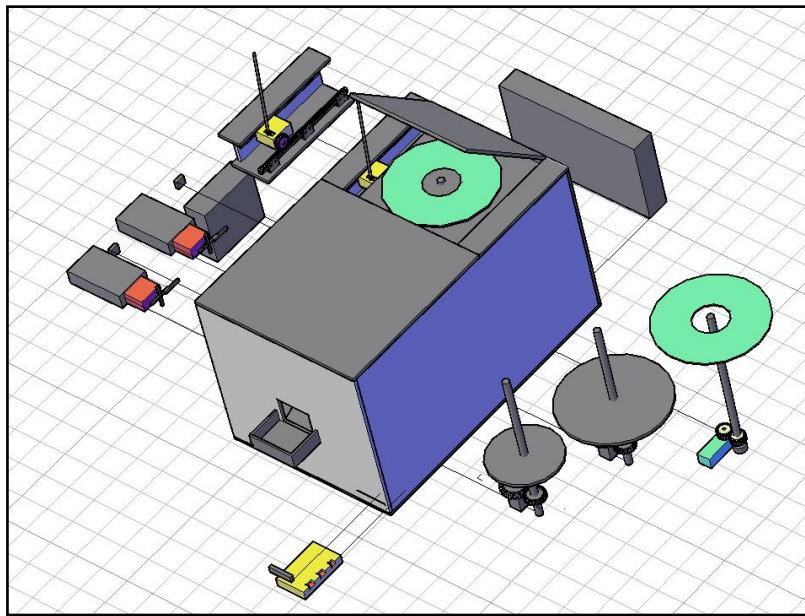


Figure 11 Assembly drawing

2.2.2.2 slide rail

This structure is used to lift the lid during performances. It uses some simple elements such as gear, rack, slide, servo motor and some fixing board to produce this mechanical structure. At first, we ignored the problem of the power of provided motor. In the test, we fought that it couldn't drive the gear to move on the rack. So we had to buy a servo motor with stronger power and then, this vital problem has been solved. And we chose a gear with an inner diameter of 6 mm and an outer diameter of 40 mm because when it turns 180 degrees from the original location, it can travel for a distance of: $3.1415926 \times 40 = 125.66\text{mm}$. In the physical model test, when it travel such distance from original point on the rack, the lid can be lifted to a idea height.

Model structure:

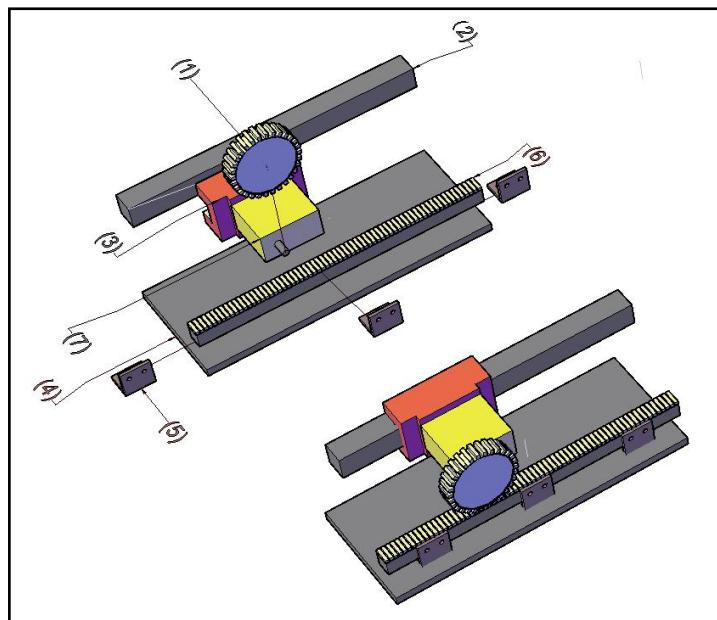


Figure 12 Lifting system

Size of each element:

Parts lists:			
ITEM	QUANTITY	SIZE(LxWxHmm)	DESCRIPTION
1	1	Standard	gear
2	1	200x5x5	Rack
3	1	2x5x180	slide
4	3	standard	Cutting ferrule
5	1	200x70x5	Fixing board
6	1	60x190x5	Fixing board
7	1	180mm length Diameter 2mm	Lifting rod

2.2.2.3 supply disc

This is the bigger disk used to supply goods. The design of this part has been modified several times. After the discussion between mechanical engineer and digital circuit engineer, we decided to divide it into 10 parts, one good takes up one part. The modulus ratio of driven gear and driving gear is 1:2 so that when servo motor turns 180 degrees, the disk can turn 360 degrees. Thus, no matter where the goods are, it can be rotated to the location we want. As for the fixation of rotating shaft, mechanical designer has thought many solutions. Comparing all of them, he fought that bearing block is the best. It can fix the shaft firmly in the vertical direction and the shaft can rotate freely in the horizontal direction, guaranteeing the accuracy of supply and avoiding the effect of friction force.

Model drawing:

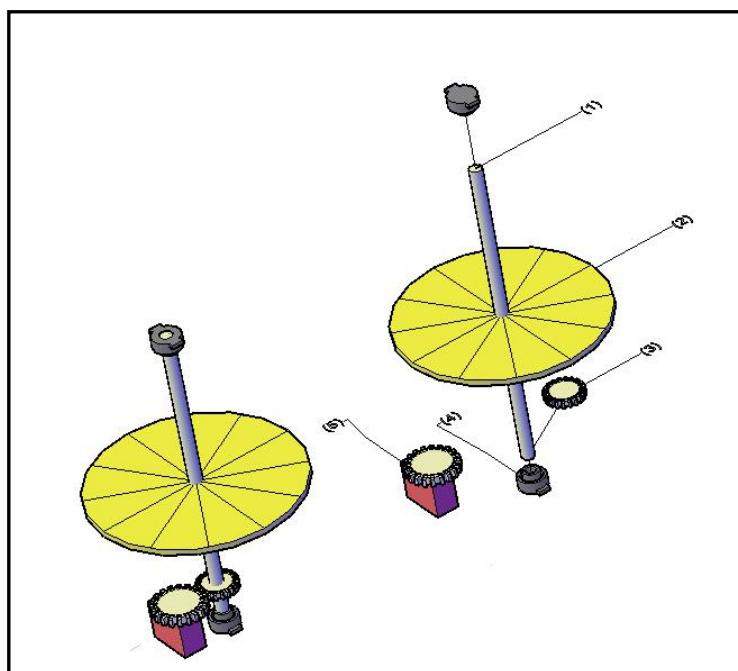


Figure 13 Supply disc

Size of each element:

Parts lists:			
ITEM	QUANTITY	SIZE(LxWxHmm)	DESCRIPTION
1	1	Diameter: 10mm Length: 290mm	stick
2	1	Diameter: 180mm Width: 5mm	disk
3	1	Inner diameter:10mm Modulus: 40 Width:10mm	Driven gear
4	2	Inner diameter: 10mm Whole size: standard	bearing block
5	1	Inner diameter:10mm Modulus :unknown	Driving gear

2.2.2.4 delivery

This part is similar to the supply disc. Here we chose a disc with a diameter of 12 cm and divided it into 4 parts. Each good was placed in one location. It was placed near the supply disk on the same horizontal line but 1cm lower than supply disc in the vertical direction. Fixation and transmission mode were all same as supply disc.

Model drawing:

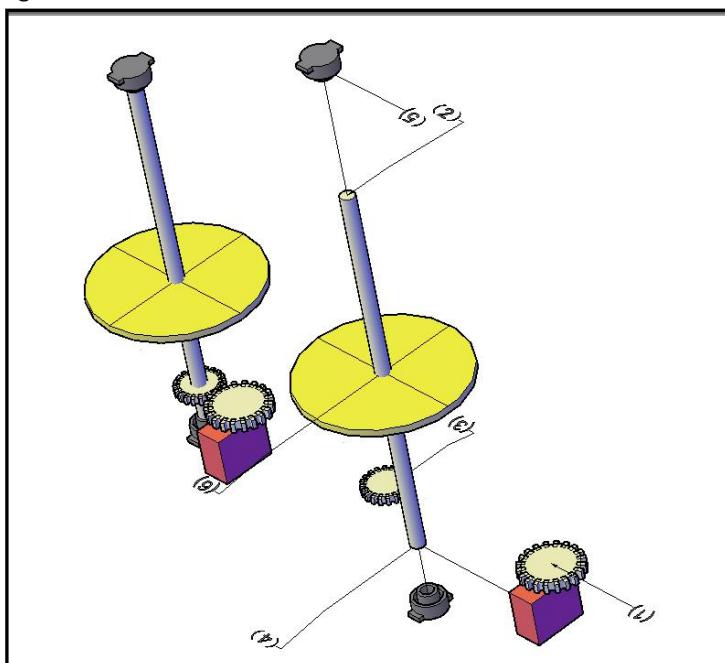


Figure 14 Shipment disc

Size of each part:

Parts lists:			
ITEM	QUANTITY	SIZE(LxWxHmm)	DESCRIPTION
1	2	Inner diameter: 10mm Whole size: standard	bearing block
2	1	200x5x5	rack
3	1	Inner diameter:10mm Modulus: 40 Width:10mm	Driven gear
4	2	Inner diameter: 10mm Whole size: standard	bearing block
5	1	Inner diameter:10mm Modulus :unknown	Driving gear

2.2.2.5 Conduction device

Totally, we have two devices like this. The one is used to deliver the goods on the supply disk to the smaller plate, and another one is used for shipment. There is a sweeping film on the motor wing, FPGA board controls its motion so it can rotate a specific degree to deliver the goods to the shipment plate. Several adjustments in the test can finally decide the proper degree and original point of rotation.

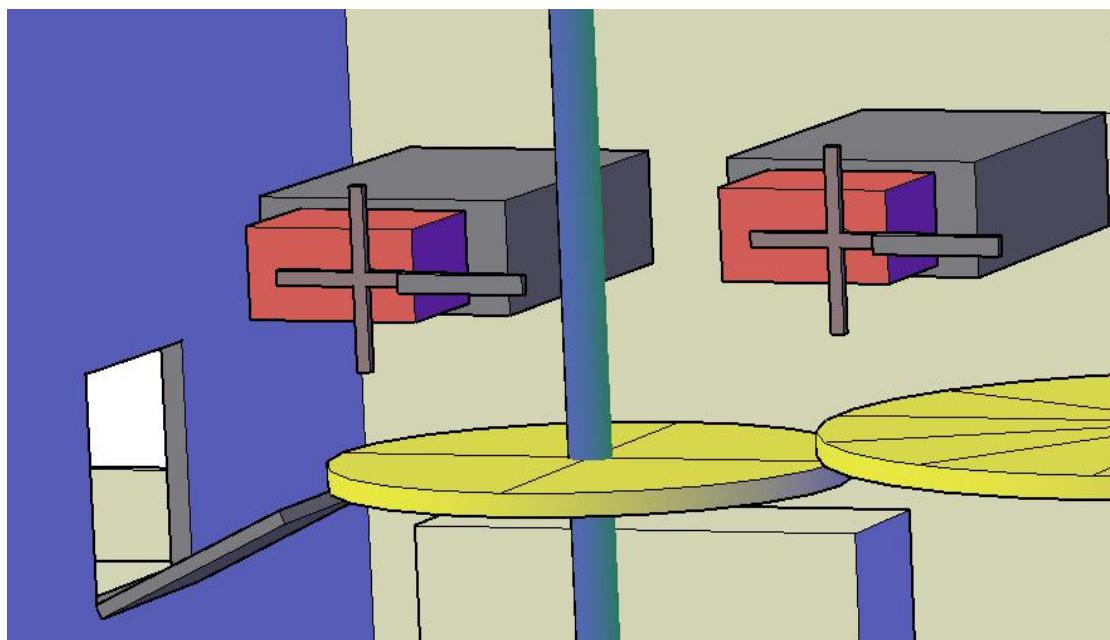


Figure 15 Conduction device for shipment and supply

2.2.2.6 card reader(infrared sensors)

At first, we placed it at the left corner in the bottom. It consisted of 4 infrared sensors which

were placed side by side. The placing manner depends on the card. Its structure is like this:

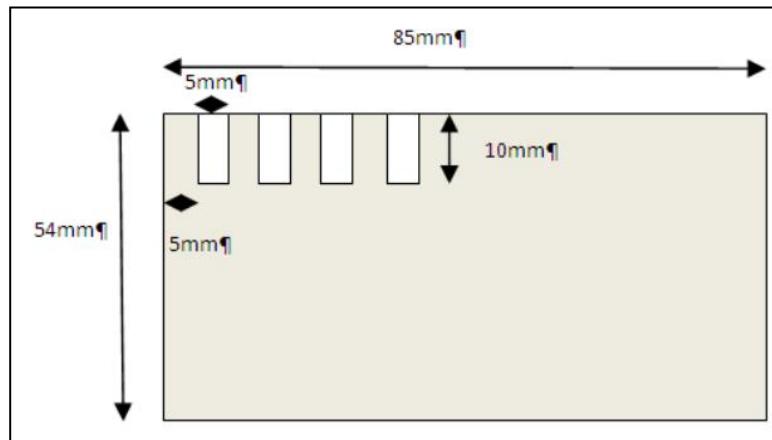


Figure 16 The structure of card

According to the card size, our first attempt:

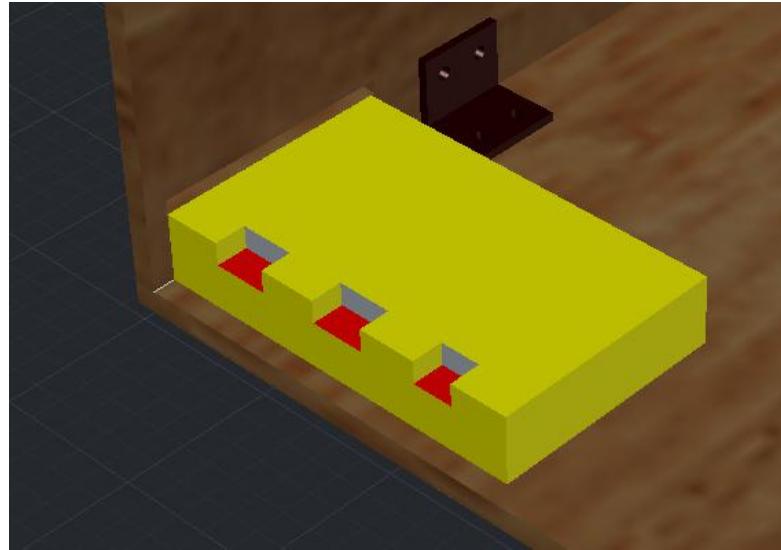


Figure 12 original design of card reader

It has 4 infrared sensors and can detect totally $2^4 = 16$ species of card. Players insert cards horizontally into holes and are then detected as customers to begin the game.

2.2.2.7 color sensor:

Since we divide the supply disk into 10 parts which means that one part occupies: $360/10=36$ degrees. For the purpose of rising accuracy and preventing it from being hit by film and rotating goods, we put it at an angle of 72 degrees from the horizontal line and glue it to the top of box. The height of color sensor has been modified several times in the test so that it can accurately detect the goods on the premise of not touching goods.

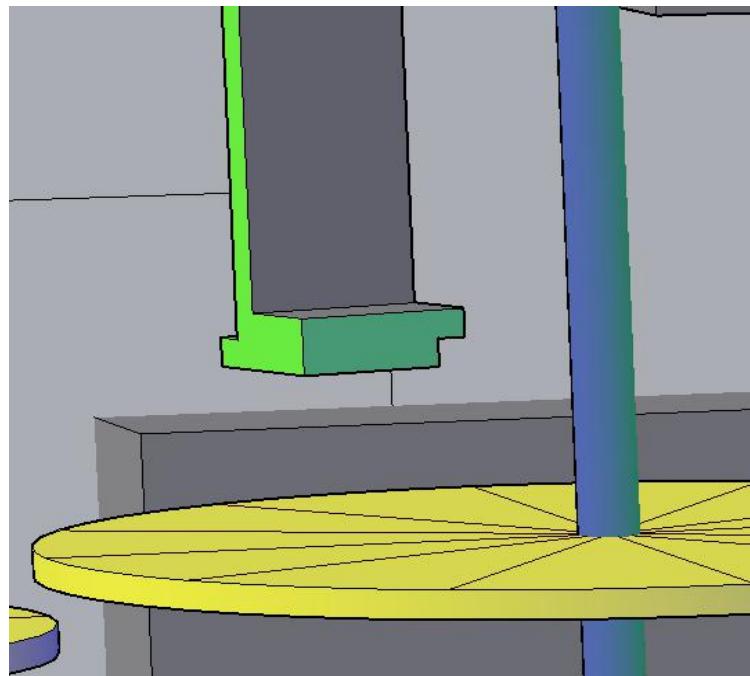


Figure 17 color sensor

2.3 Entities operation

2.3.1 Shell

We chose high density board as our shell material since its hardness is high and surface is smooth. What's more, it cost us 50 yuan, dramatically reducing the production cost. When we were choosing the HDB, thickness was a significance point. Too thick would make drilling inconvenient and would compress the internal space, too thin would lead to insufficient hardness and deformation. Finally, we chose the board with the thickness of 5mm and divided it into the size we want on the spot.

Then how did we fix them and connected the board with other elements?

Mainly four methods:

- Connected density boards with corner connectors, screws and nuts. This way agreed with engineering design requirements and easy to disassemble and reload.
- Used ruler and pencil to demarcate special positions for each component. Then drilled holes with bits of different sizes. Then we installed the corresponding components on the board with different screws and nuts.
- For some components like motors that were hard to be connected by screws, we carefully considered gravity effect and used glue to connect them.
- After all connecting works have been done, we dressed the board with a sharpener and eventually got a smooth and beautiful machine.



Figure 18 Connection between door and board



Figure 19 Construction in process

2.3.2 Supply disc part

After many tests, we finally put it at a height of 160cm above the bottom. In terms of how to fix goods to avoid being threw away, we used those wood strips in the figure. The gap between adjacent strips is also an issue to consider. Too loose did not make a difference, but too strict would increase friction force then it would be harder for motor to deliver goods to another plate. We found that when we set the gap between adjacent strips to be 2mm more than the width of cube, the issue could be solved appropriately.

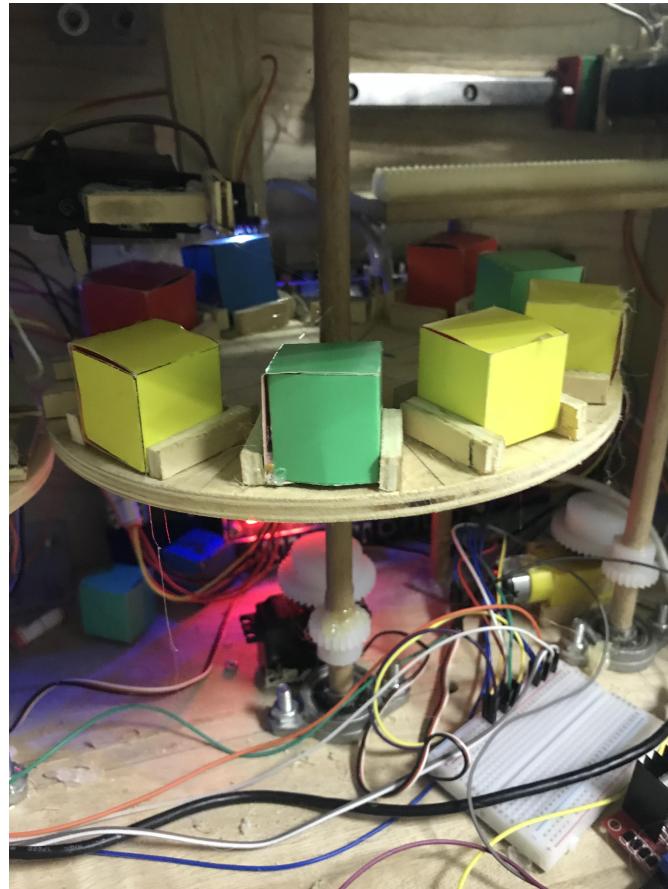


Figure 20 Physical supply plate

2.3.3 Shipment disc part

In practical operation, the situation of shipping plate was much more complex than we thought. On the one hand, it had the responsibility to catch cubes. On the other hand, it should deliver goods accurately to the customer. During experiments, we stuck some hard sticky paper on the plate to prevent the goods from flying out due to physical factors in the process of handover. What's more, in order to avoid being jammed during transportation, we installed some low baffles in the rear of each direction, which could effectively reduced mechanical jamming. We combined the two plates to test whether they could do automatic sorting and shipment accurately. After some adjustment, the accuracy rate could be more than 90 percent.

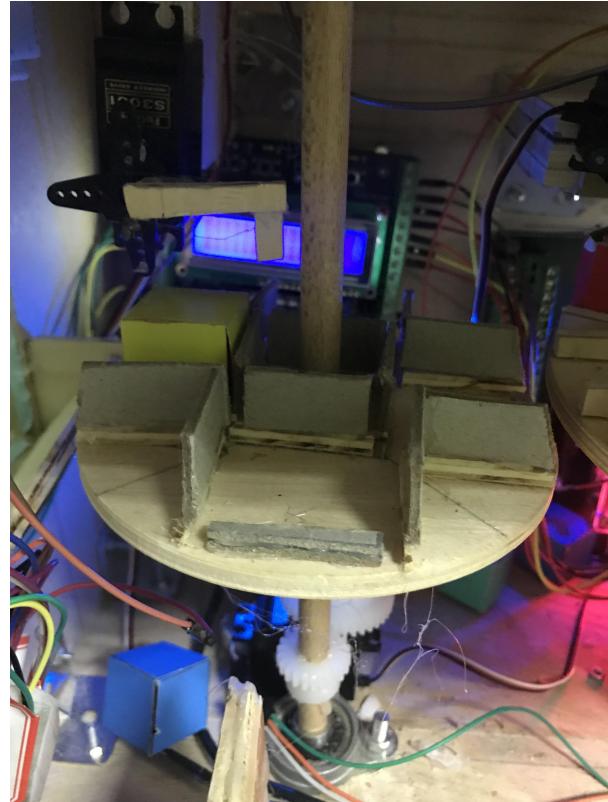


Figure 21 Physical shipping plate

2.3.4 Lifting system

Mechanical engineer wanted to make this part the best. He had thought a lot of method to achieve the lifting of lid, but they were all negated in the team meetings. This part related to many elements: gravity, friction force, the size of gear, connection between servo motor and slider, the choose of slider and rail...They constituted a whole and were also affected by motor horsepower and physical factors. Through a period of discussion and design, we had a complete solution og this question:

- Use a gear with an outer diameter of 40 mm and a rack of 200mm with the same modulus.
- Use flange slider to increase contact area and achieve better fixing effect.
- Add a small piece of wood on the flange slider to increase the friction between the motor, then connect wood and motor by glue.
- Construct a device consisting of screw, nut and wood. on the motor and string a hard wire of the right length. Do the same thing on the upper right corner of the lid and connect the two parts with wire.

In this way, a complete lifting system had been made and it has undertook many times of experiment so the stability and reliability were good enough.

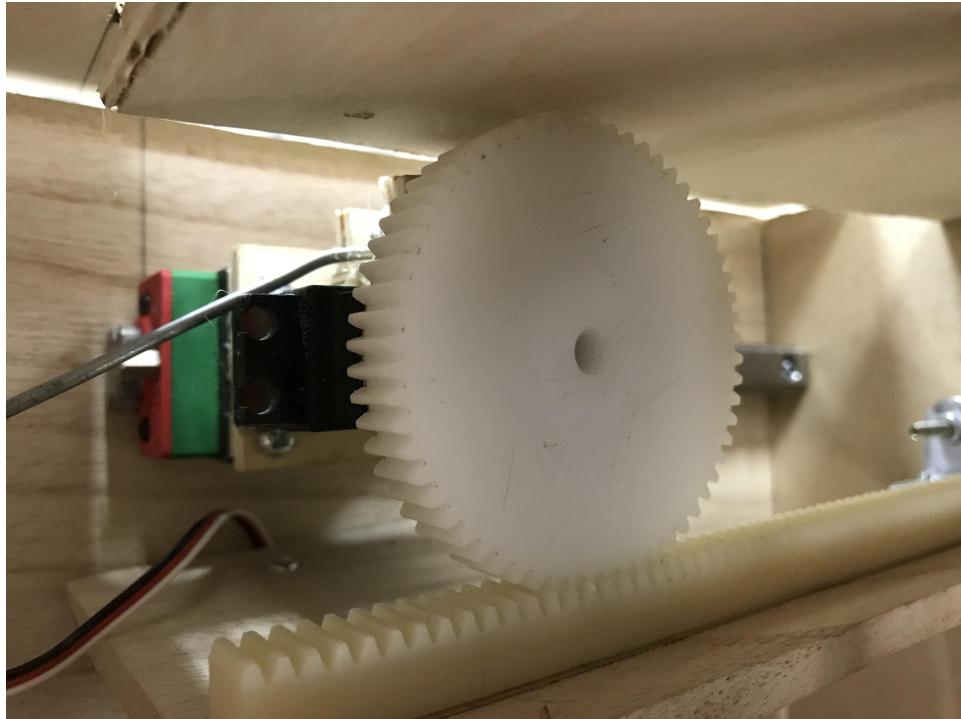


Figure 22 Physical Lifting system



Figure 23 Top view of lifting system

2.3.5 Card Reader

In the process of model design, we did not consider about the sensitivity and card size. Not like photoelectric gates, their sensitivity are unsatisfactory. In our test, we have tried many method to reduce disruption and increase recognition rate and in the end, when we placed in this form, we could get a high precision:

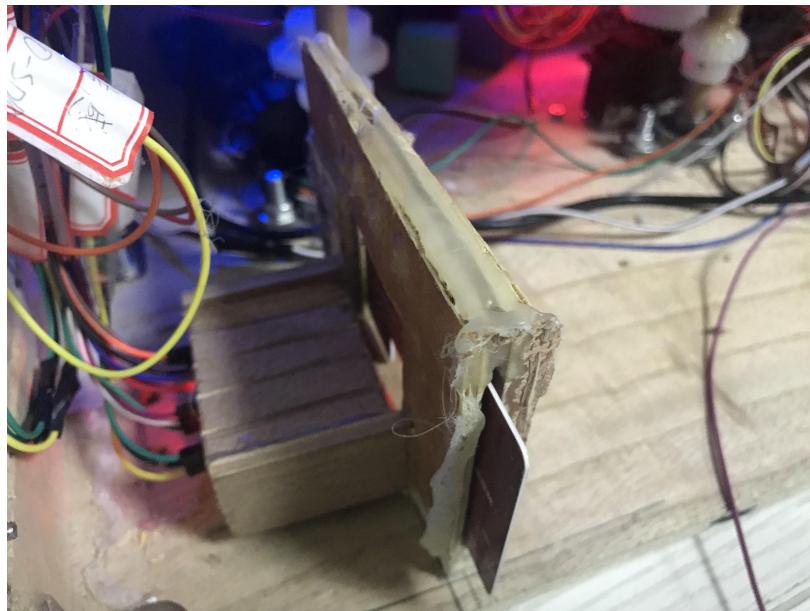


Figure 24 Actual card reader(1)

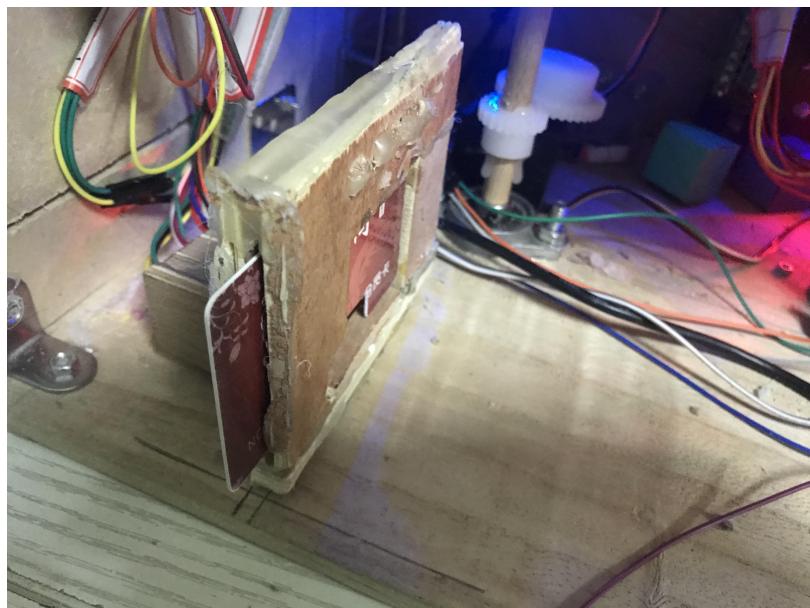


Figure 25 Actual card reader(2)

2.4 Final mechanical drawing

Combing our design with the real situation, we made some adjustment of the mechanical drawing. Here we use series of figures to show our final design. They present the product from different angles.

2.4.1 Northeast isometric View (without the inner structure)

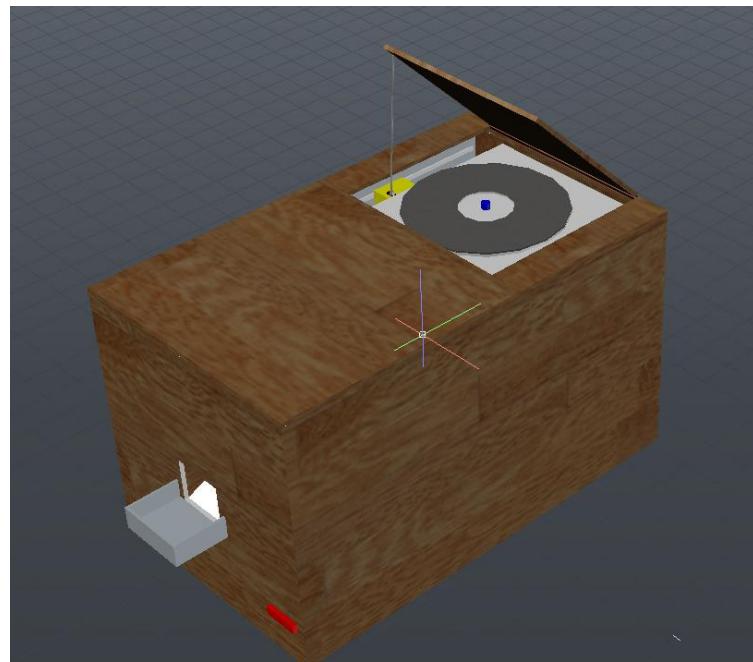


Figure 26 Northeast isometric View

2.4.2 Right View

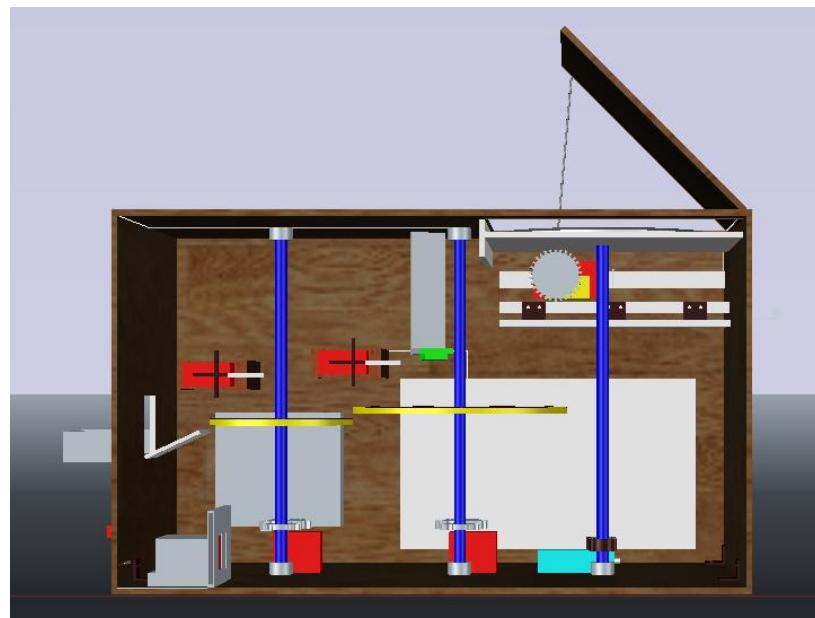


Figure 27 Right View

2.4.3 Top View

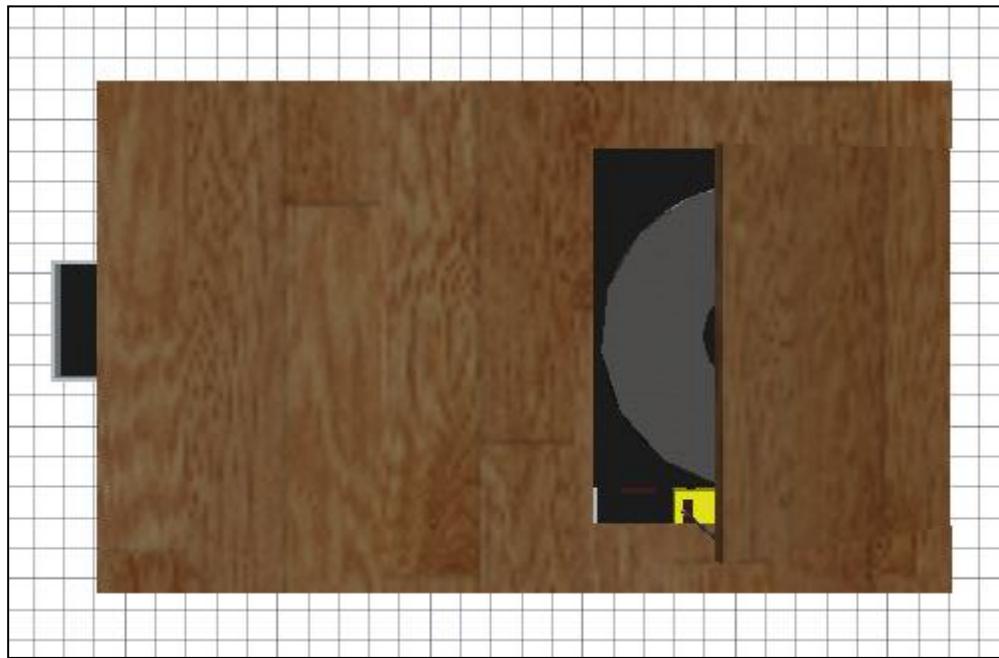


Figure 28 Top View

2.4.4 Final assembly drawing

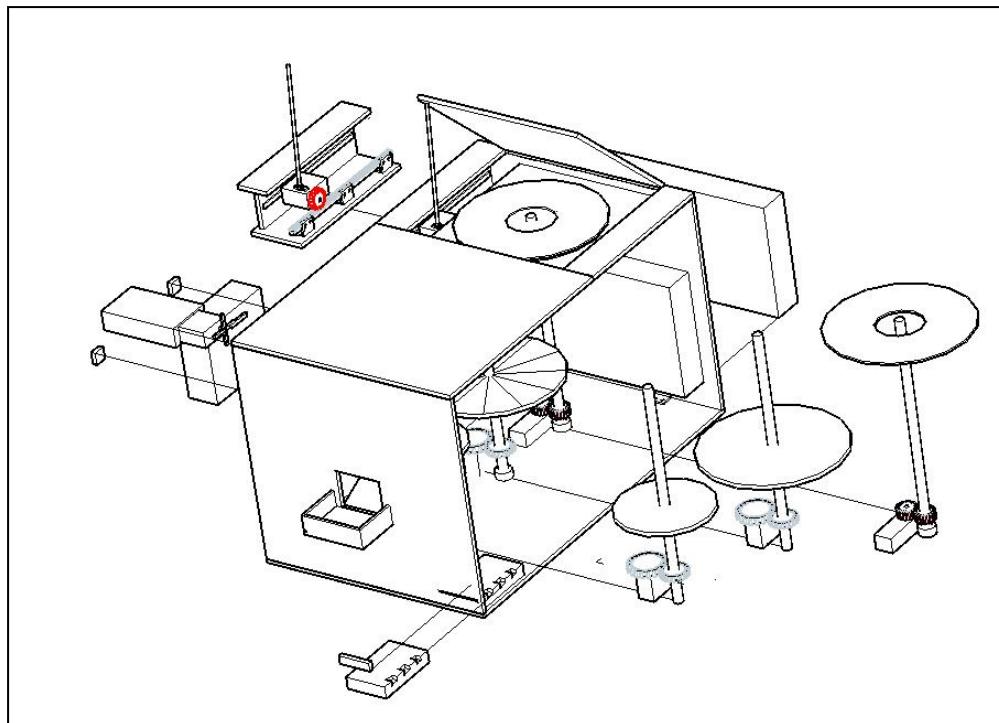


Figure 29 Final assembly drawing:

2.5 Finished Product

Here we demonstrate the overall appearance and inner structure of the machine:

2.5.1 Inner structure of product

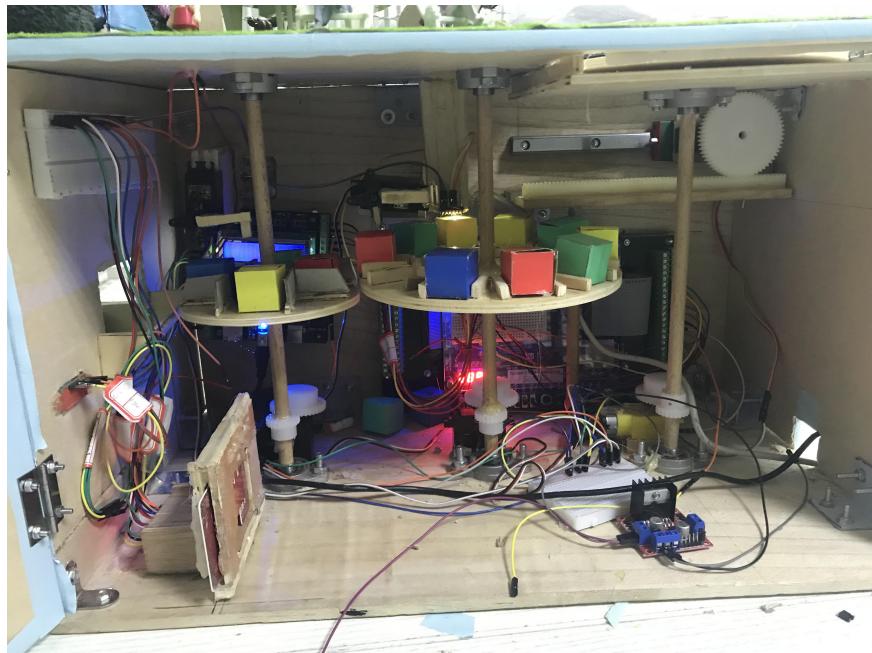


Figure 30 Inner structure of product

Clearly, we can see each part of our 'Music Box' and understand what they are working for.

2.5.2 Overall appearance

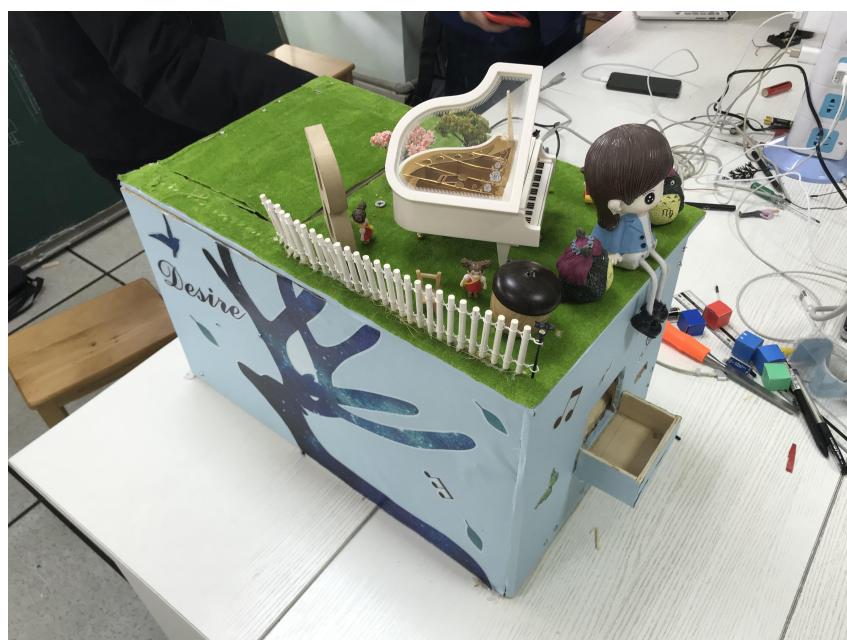


Figure 31 Beautiful Music Box



Figure 32 Top view of the box

2.6 Operation instructions

2.6.1. Entertainment

Everything starts with card insertion detection. After the infrared sensor detects the card type, the distance sensor detects whether the player is near or not. When everything is ready, the process begins. The electronic equipment controls the motor and the lifting device to lift the lid. At the same time, the DC motor controls the CD disc to start turning. The player stops turning after the PC chooses to end the music then entertainment part ends here.

2.6.2 Supply and delivery

After the player has answered the question, the PC transmits the information to mbed. The small disc motor receives instructions to rotate with an appropriate angle, and the delivery motor will type out the corresponding gifts. Then the delivery tray rotates 180 degrees to the place where it is handed over to the supply tray. The color sensor receives mbed instructions and begins to detect the color of gifts missing from the tray. The supply plate rotates 36 degrees at a time until the color is detected. The motor drives the gear to rotate the gift to the junction, and the transfer motor is responsible for delivering it to the pallet. Thus, it has realized the automatic replenishment.

3. Digital design

3.1 Introduction

FPGA (field-programmable Gate Array), or Field Programmable Gate Array, is a product of further development on the basis of Programmable devices such as PAL, GAL and CPLD. As a kind of semi-customized circuit in the field of ASIC, it not only solves the shortage of

customized circuit, but also overcomes the shortcoming of limited gate circuit number of original programmable devices

3.2 Basic design concept

Control five servo motors through the FPGA. Five counters are designed using the ‘clock’ output from the FPGA itself. The crossover technology allows the servo motor to reach different positions. Use the ‘if’ statement to select a different motor. After selecting a motor, the ‘if’ statement is still used in the subroutine to select different positions of the motor.

A DC motor is driven by a regulated voltage of 6V. The switch of the DC motor is controlled by a mechanical structure. When the upper cover is open, the DC motor works. When the cover is closed, the DC motor stops working.

Our project uses six motors, so a three-bit data selection port is required to select different motors. One of the motors has ten states, so four data selection ports are needed to select different states of the motor. The communication data lines of MBED and FPGA have a total of seven-bit ports.

3.3 Designing process

3.3.1 Set states for each motor

The processing of clock signals is one of the features of FPGAs, so the frequency divider is also one of the basic designs used in FPGA design.

The effective working range of the servo motor is 25000-125000bps. The frequency can be equally or unevenly distributed through different requirements in order to bring the motor to the specified position.

For example: the small plate motor has five positions:

```
1. parameter pulse_times=1000000;
2. parameter output_high_times1=25000;/////
3. parameter output_high_times2=50000;
4. parameter output_high_times3=75000;//small plate data
5. parameter output_high_times4=100000;
6. parameter output_high_times5=125000;///
```



Figure 33 motor device

Through continuous testing, we finally get the best and reasonable data to ensure the normal operation of the machine.

3.3.2 Define output variables and different counter names.

Counters can be used to implement the divider function. The number of counters corresponding to one motor is the same as the number of states owned by one motor.

```
1. input clk,case_select0,case_select1,case_select2,case_select3,a,b,c,d;
2. output wave1,wave2,wave3,wave4,wave5;
3. reg wave1;//small plate output
4. reg wave2;//big plate output
5. reg wave3;//small push plate output
6. reg wave4;//big push plate output
7. reg wave5;//upper motor output
8. reg [31:0] count0 = 32'd0;
9. reg [31:0] count1 = 32'd0;
10. /*partial omission*/
11. reg [31:0] count60 = 32'd0;
12. reg [31:0] count61 = 32'd0;
```

3.3.3 Use 'if' statement to select different motor

```
if(a==0 & b==0 & c==1)//small plate
  if(a==0 & b==1 & c==0)//big plate
    if(a==0 & b==1 & c==1)//small push plate
    if(a==1 & b==0 & c==0)//big push plate
      if(a==1 & b==0 & c==1)// upper cover open
```

3.3.4 Select the state of the motor

For example: we select the upper cover controller motor and its open state.

```
1. if(a==1 & b==0 & c==1)// upper cover open(select the motor)
2. begin
3.   if(case_select0==0 & case_select1==0 & case_select2==0 & case_select3==1)
4. // select the open state
5.   begin
6.     if(count50>=0 & count50<output_high_times50)
7.     begin
8.       wave5<=1; // output voltage level is high
9.       count50<=count50+1;
10.    end
11.    if(count50>=output_high_times50 & count50<=pulse_times-1)
12.    begin
13.      wave5<=0; // output voltage level is high
14.      count50<=count50+1;
15.    end
16.  if(count50==pulse_times)
```

```

17. begin
18.   wave5<=1;
19.   count50<=1;
20. end
21. end
22. end

```

The function of driving the specified motor to the specified position can be realized by superposition of a plurality of 'if' statements.

3.3.5 Set the pin

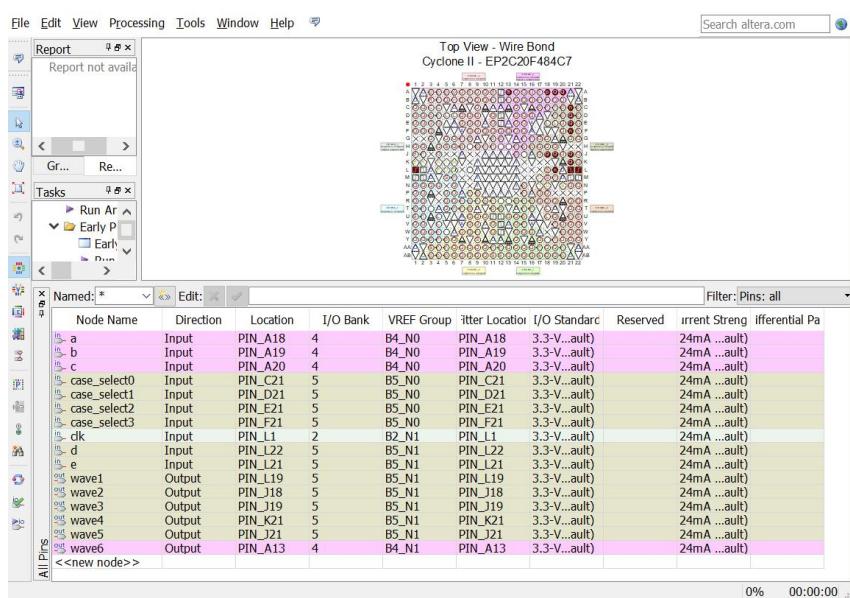


Figure 34 Pin Planning Result

3.3.6 Load the code to the FPGA and connect to theMBED

Select the correct FPGA model and burn the code into it.

Connect the communication pins that have been set on the FPGA to theMBED communication pins.

3.3.7 Locate the data lines

There are various factors to consider when placing different data lines:

- Whether it is rational to use the space.

- Whether it is easy to repair.

- Whether the connection is stable.

- Whether it is beautiful.

Combining these considerations, we have designed a unique way of placing data lines.

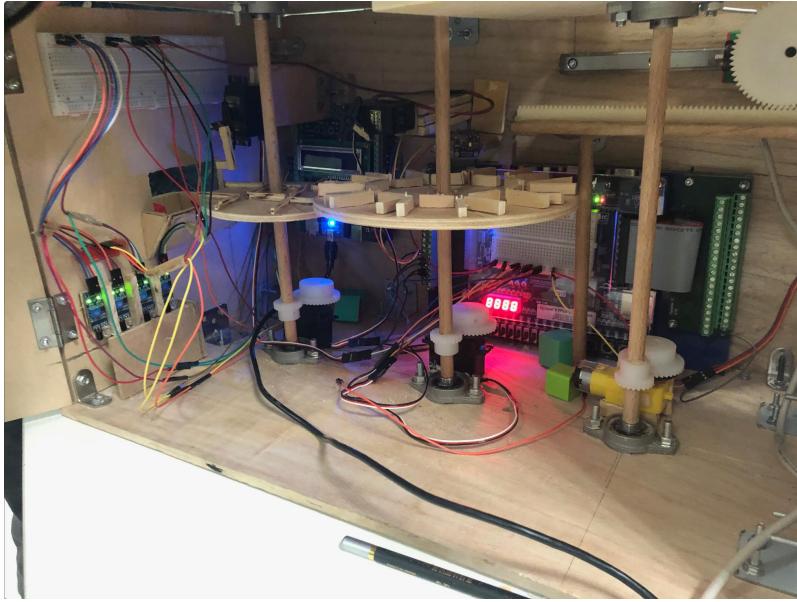


Figure 35 digital inner design

3.4 ASM Chart

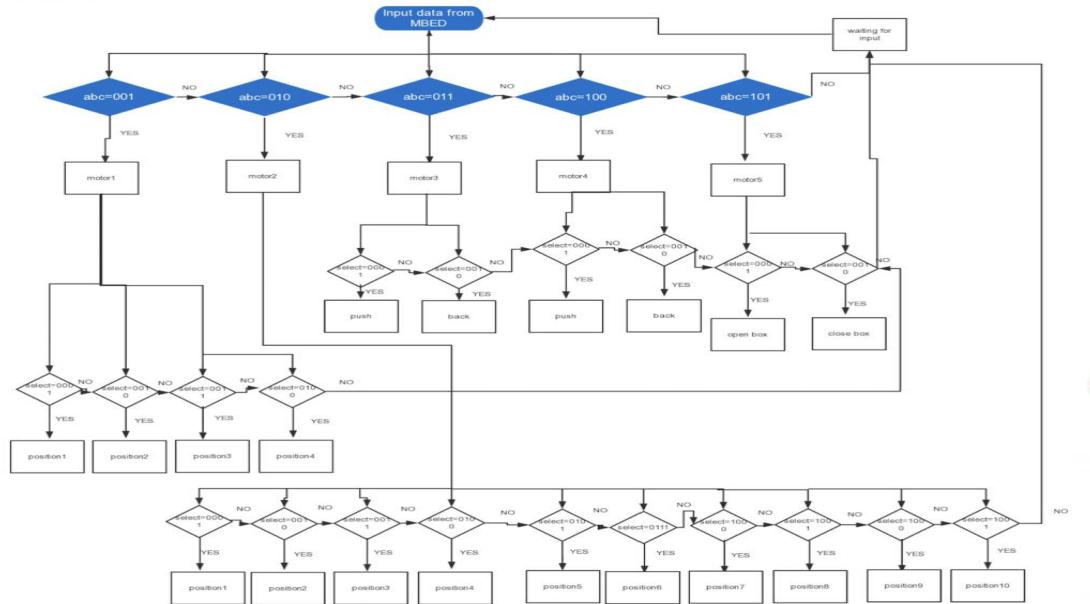


Figure 36 ASM Chart

4. Software Design

4.1 MBED

4.1.1 MBED Introduction

Our microcontroller is LPC1768 is one of NXP's LPC17XX family of microcontroller which is based on the ARM-Cortex-M3 core. The board can perform a small amount of processing

and data storage as well as communication. It can support MBED OS.

The MBED operating system is an internal operating system that controls hardware and interacts with the cloud in a simple, consistent manner. We can program on the cloud using C or C++ language rather than the Assembly language. What's more, Our codes can be widely used in the MBED machine without being specific to the board.

In conclusion, we can use the MBED IOT platform to program and control the microcontroller in a simpler way. We are able to pay more attentions to the algorithm.

4.1.2 Functions of MBED

In our program. MBED is just like a “brain”, it not only connects the “face”—C# interface window and the “legs” —FPGA motor section, but also it analysis the inner logic and make the proper judgement.

Basic Functions:

- ✓ Communicating with PC, Reading the commands from user and maintainers, giving the feedback and sensor information to PC.
- ✓ Communicating with FPGA board, giving the motor command information to the FPGA board.
- ✓ Controlling the color sensor, reading the three basic color values and background color.
- ✓ Controlling the Photoelectric sensor, reading the card information.
- ✓ Controlling the distant sensor, reading the distance information.
- ✓ Our system has two modes: Operating Mode and Maintenance Mode. Operating Mode is for players to play the game and distribute the different award. The maintenance mode is for maintenance personnel to test various components such as the condition of the sensors and control the motors to move.

4.1.3 Code Structure

4.1.3.1 Structure diagram

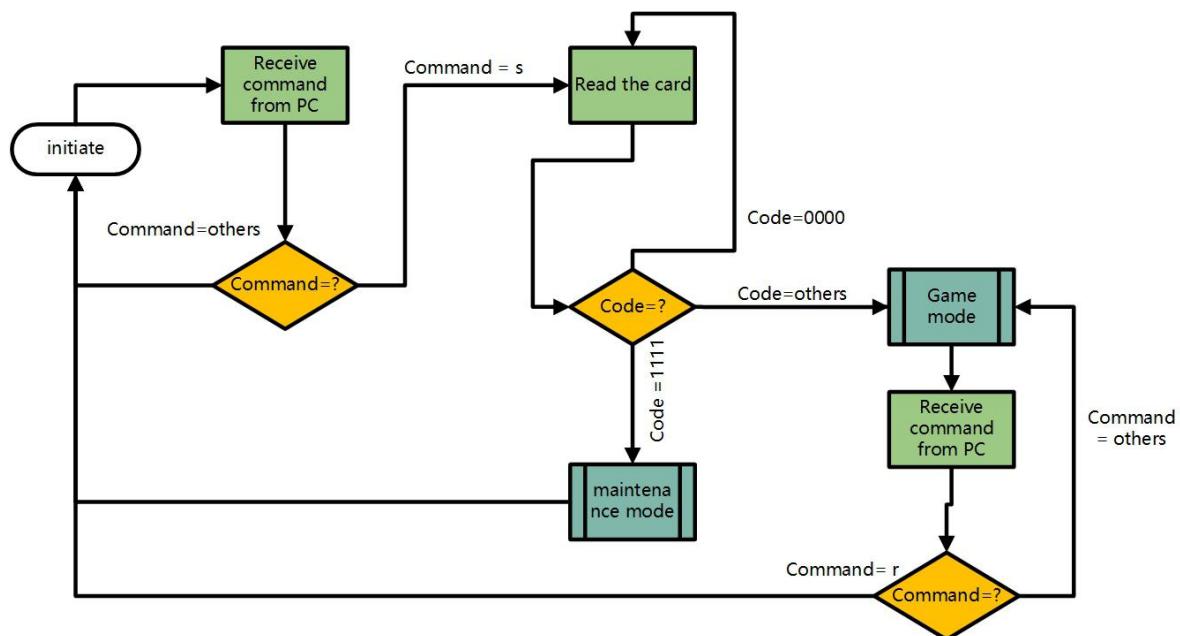


Figure 37 Structure Diagram for MBED – Overview

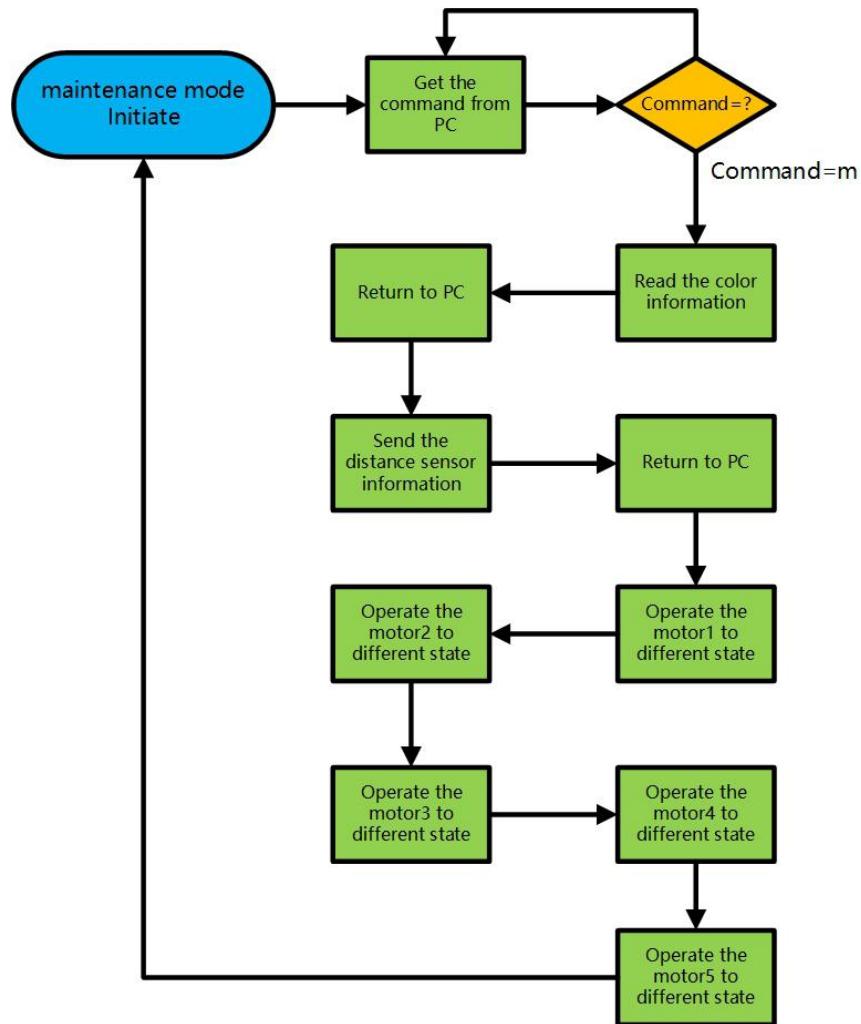


Figure 38 Structure Diagram for MBED – Maintenance

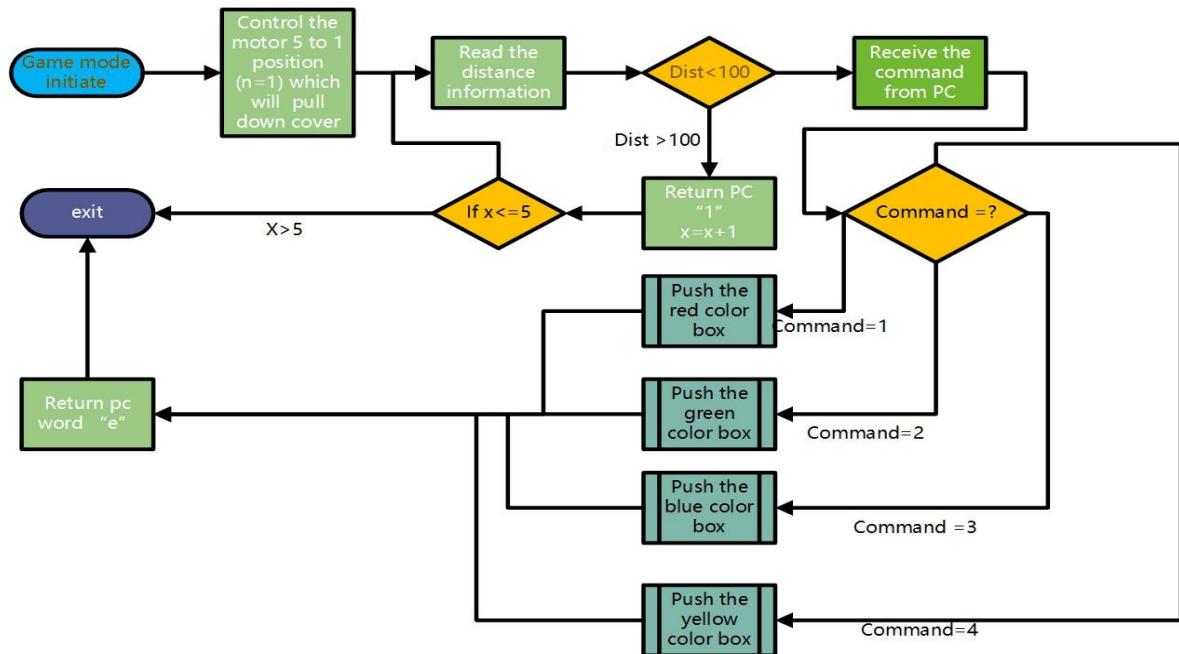


Figure 39 Structure Diagram for MBED – game mode

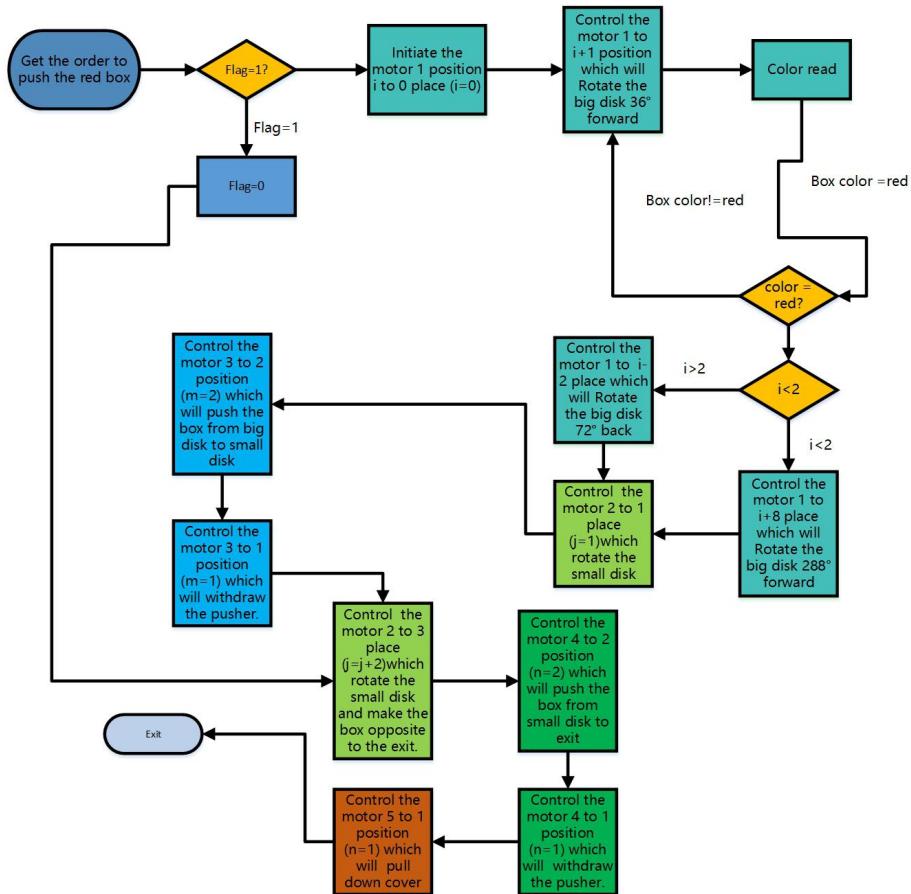


Figure 40 Structure Diagram for MBED – pushing process

4.1.3.2 Pseudocode

In the pseudocode part. We omit the detailed sentence in the code in order to present the global view of the whole code .

```

1.   while(1){
2.     sys_init();
3.     do {
4.       get the pc command;
5.     } while(command!='s');
6.     read the card and transmt back the card information to pc;
7.     if(key!=0000) { //players cards
8.       control the motor5 to position 2;
9.       read the distance information and make the judgement and send
10.      the corresponding information to PC
11.      do{
12.        get pc command;
13.        Initiate all the motors position;
14.        if(command=='1'){ //we push the read color box
15.          if(flag_red==1){
16.            ...
17.          }
18.        }
19.      }
20.    }
21.  }
  
```

```
15.          rotate the small disk
16.          push the red color box from small disk directly;
17.          let flag_red is equal to 0; }
18.      else {
19.          rotate the bigger disk and judge the red color;
20.          rotate the red color box to the position where big
    disk and small disk are contacted with each other;
21.          push the box from big disk to small disk;
22.          rotate the small disk to the exit;
23.          push the red color box from small disk y;
24.          transmit e back to pc;     }
25.      }else if(command=='2'){
26.          if(flag_green==1) {
27.              rotate the small disk
28.              push the green color box from small disk directly
29.              let flag_green is equal to 0; }
30.          else {
31.              rotate the bigger disk and judge the green color;
32.              rotate the green color box to the position where b
    ig disk and small disk are contacted with each other;
33.              push the box from big disk to small disk;
34.              rotate the small disk;
35.              push the green color box from small disk ;
36.              transmit 'e' back to pc;
37.          }
38.      }else if(command=='3'){
39.          if(flag_blue==1)
40.          {
41.              rotate the small disk
42.              push the blue color box from small disk directly;
43.              let flag_blue is equal to 0; }
44.          else {
45.              rotate the bigger disk and judge the blue color;
46.              rotate the blue color box to the position where bi
    g disk and small disk are contacted with each other;
47.              push the box from big disk to small disk;
48.              rotate the small disk;
49.              push the blue color box from small disk y;
50.              transmit 'e' back to pc;
51.          }
52.      }else if(command=='4'){
53.          if(flag_yellow==1)
54.          {
55.              rotate the small disk
```

```
56.                                push the yellow color box from small disk directly
57.                                let flag_yellow is equal to 0; }
58.                                else {
59.                                    rotate the bigger disk and judge the red color;
60.                                    rotate the yellow color box to the position where
   big disk and small disk are contacted with each other;
61.                                    push the box from big disk to small disk;
62.                                    rotate the small disk;
63.                                    push the yellow color box from small disk y;
64.                                    transmit e back to pc;
65.                                }
66.                            }
67.                            }while(command!='r');
68.                        } else {
69.                            while(1){
70.                                do {
71.                                    get the pc command
72.                                    } while(command!='m');//maintaince mode
73.                                    read the color information and transimit to PC;
74.                                    read the distance information and transimit to PC;
75.                                    check the motors in every state;
76.                                }
77.                            }
78.                        }
79.                    }
```

4.1.4 Functions in diagram

4.1.4.1 Card_reading function

This function is used to get the card information from the four infrared sensors. We detect the voltage level of the pins and mix these voltage level information together into a 4-bit decimal numbers.

Function declaration

```
1. int readcard(int sensor1,int sensor2,int sensor3,int sensor4);
```

The more detailed algorithm you can see the sensors part.

4.1.4.2 System initiate function

As its name illustrates, this Function is used to do the initialization at the start of the code.it contains many functions:

- initiate lcd1602 display screen and set the back light to turn on.
- Initiate the distance sensor
- Initiate the color sensor and set the integration time.
- Initiate the port

Function declaration

```
1. void sys_init();
```

Function definition

```
1. void sys_init()
2. {
3.     par_port = new MCP23017(p9, p10, 0x40);
4.     par_port->config(0x0F00, 0x0F00, 0x0F00);
5.     lcd = new WattBob_TextLCD(par_port);
6.     BACK_LIGHT_ON(par_port);
7.     TOF_sensor.VL6180_Init(); //initialize the distance sensor
8.     rgb_sensor.enablePowerAndRGB(); // initialize the color sensor
9.     rgb_sensor.setIntegrationTime(10);
10.
11. }
```

4.1.4.2 Color recognition function

We design a color recognition function. Anytime we want to detect the color type, we can invoke this function directly which makes us codes more efficiently.

Function declaration

```
1. char colour_1();
```

The detailed algorithm, measurement and modulation method are discussed in the Sensors system part. When the color senor is applied in the real project. It is much complicated than the idea algorithm since it will under the environment influence.

4.1.4.3 FPGA communication functions

FPGA communication functions are a series of functions which are used to transmit the motor serial numbers and motor states. The detailed functions will be discussed in the communication part.

4.1.5 Communications in MBED

4.1.5.1 Communications structure

4.1.5.1.1 MBED ↔ PC

MBED offer libraries for serial port communication. Firstly we define the pins for serial communication

```
1. Serial pc(USBTX, USBRX); // 9600 baud
```

We can use the function defined in the libraries such as

```
1. pc.printf();
2. pc.getc();
```

4.1.5.1.2 MBED \longleftrightarrow FPGA

We define 7 pins and by controlling the voltage level of the pin, we can achieve the communication:

- From P14 to P16 are used to select the motor serial number. since we have 5 motors ,we need at least three bits to do the select.
- From P17 to P20 are used to select the motor states number. Since at most we have 10 states, we need at least four bits to select the states.

```
1. DigitalOut FPGA_1(p14); // motor serial number select 1
2. DigitalOut FPGA_2(p15); // motor serial number select 2
3. DigitalOut FPGA_3(p16); // motor serial number select 3
4. DigitalOut FPGA_4(p17); // motor state number select 1
5. DigitalOut FPGA_5(p18); // motor state number select 2
6. DigitalOut FPGA_6(p19); // motor state number select 3
7. DigitalOut FPGA_7(p20); // motor state number select 4
```

By using the functions we defined

The first two functions are used to reduce the workload during the program.

The second two functions are used to control the voltage level to transmit information.

```
1. void fpga_transmit_machine(int i); // 1-small disk rotating
2. // 2-big disk rotating 3-big disk pusher 4-small disk pusher 5-cover motor
3. void fpga_transmit_state(int i); // 0-9 states
4. void fpga_send_machine(int a, int b, int c); // control the voltage level of
   // the 3 pins
5. void fpga_send_state(int a, int b, int c, int d); // control the voltage level of
   // the 4 pins
```

Function definition

```
1. void fpga_transmit_state(int i){ // transmit the motor state
2.     switch(i)
3.     {
4.         case 1: fpga_send_state(0,0,0,1); break;
5.         case 2: fpga_send_state(0,0,1,0); break;
6.         case 3: fpga_send_state(0,0,1,1); break;
7.         case 4: fpga_send_state(0,1,0,0); break;
8.         case 5: fpga_send_state(0,1,0,1); break;
9.         case 6: fpga_send_state(0,1,1,0); break;
10.        case 7: fpga_send_state(0,1,1,1); break;
11.        case 8: fpga_send_state(1,0,0,0); break;
12.        case 9: fpga_send_state(1,0,0,1); break;
13.        case 0: fpga_send_state(1,0,1,0); break;
14.    }
```

```
15.    }
16. void fpga_transmit_machine(int i){      //transmit The motor serial number
17.     switch(i)
18.     {
19.         case 1: fpga_send_machine(0,0,1); break;//
20.         case 2: fpga_send_machine(0,1,0); break;//
21.         case 3: fpga_send_machine(0,1,1); break;//
22.         case 4: fpga_send_machine(1,0,0); break;//
23.         case 5: fpga_send_machine(1,0,1); break;//
24.     }
25. }
26.
27. void fpga_send_state(int a,int b,int c,int d){ // control the voltage levels
28.     FPGA_4=a;
29.     FPGA_5=b;
30.     FPGA_6=c;
31.     FPGA_7=d;
32. }
33. // control the voltage levels of 3 pins
34. void fpga_send_machine(int a,int b,int c){
35.     FPGA_1=a;
36.     FPGA_2=b;
37.     FPGA_3=c;
38. }
```

4.1.5.2 communications protocols

4.1.5.2.1 MBED ↔ PC

Main Part

- Receive 's'
Meanings: it tells the MBED to operate the main program and after that it will start the card reading process.
- Send "00xxxx"

```
pc.printf("%06d",key);
```

Meanings: According to the protocols, the first 2 bits mean that this is the card information and the following 4 bits carry the user information.

- Send "1"

```
1. pc.printf("1");
```

Meanings: it tells the PC that the user is out of the range and the timer is keeping the time.

- Send "2"

```
1. pc.printf("2");
```

Meanings: it tells the PC that the user is out of the range for 20s.

- Send "3"

```
1. pc.printf("3");
```

Meanings: it tells the PC that the user is in the range.

Game Mode

- Receive '1'
Meanings: it tells theMBED to push the red box.
- Receive '2'
Meanings: it tells theMBED to push the green box.
- Receive '3'
Meanings: it tells theMBED to push the blue box.
- Receive '4'
Meanings: it tells theMBED to push the yellow box.
- Receive 'r'
Meanings: it tells theMBED to reset the whole program.
- Send "e"

```
1. pc.printf("e");
```

Meanings: it tells the PC that the goods-delivering process has finished

Maintenance Mode

- Receive 'm'
Meanings: it tells theMBED to operate the maintenance mode.
- Send the sensors conditions

```
1. pc.printf("%d",dist); //距离
2. pc.printf("red=%d,green=%d,blue=%d",rgb_sensor.getRedData(),rgb_sensor.getGreenData(),rgb_sensor.getBlueData());
3. pc.printf("%c",colour_1());
```

4.1.5.2.2 MBED ↔ FPGA

The MBED use a serial protocol to communicate with FPGA. The instructions from MBED to FPGA has 7 bits , no purity bit and stop bit.

The following is the instructing table for the communication between MBED and FPGA

fpga_transmit_machine()	fpga_transmit_state()	instruction	action
1	*	****001	motor 1 is specified
2	*	****010	motor 2 is specified
3	*	****011	motor 3 is specified
4	*	****100	motor 4 is specified
5	*	****101	motor 5 is specified
*	1	0001***	motor * to the state1

*	2	0010***	motor * to the state2
*	3	0011***	motor * to the state3
*	4	0100***	motor * to the state4
*	5	0101***	motor * to the state5
*	6	0110***	motor * to the state6
*	7	0111***	motor * to the state7
*	8	1000***	motor * to the state8
*	9	1001***	motor * to the state9
*	0	1010***	motor * to the state10

4.2 PC

4.2.1 PC introduction

4.2.1.1 Main Window's Design and Functions

Function section:

There are six buttons in main window, one of which is used for starting the game and three for querying information which include ranking table, emotion distribution and game background. These four buttons can be clicked to control the machine to enter operation mode. And there is also a button for engineers to maintain the machine. Once engineer click this button, the user interface process will jump to maintenance mode. The last one is designed for selecting language(English or Chinese).

Introduction section:

There is a block of texts about product introduction. Users can click button below to play audio introduction or pause, resume it.



Figure 41. UI design of Main Window

4.2.1.2 Introduction of Two Modes

Operation mode:

(1) New Game button: This button is designed for users to start game. After clicking this button, users will be prompted to insert the card which contain their own information, 4-bit binary number. And then the user need choose their emotion and preference which is related with subsequent music which is played by PC.

(2) Ranking List button: This button is used for querying the ranking table of all 15 users. After

or before new game,users can click this button to query their own scores and ranking information from a ranking table.

(3)Data Querying button:This button is used for querying emotion distribution of all trials.There will be a pie chart which display total number of two kinds of emotions on PC screen after clicking this button.Workers of playground can check information about visitors' emotion to improve recreational facilities.

(4)Game Background button:Users which are interested in game background can click this button to query it.Once users click this button,PC will display several images and relevant words.

Maintenance mode:

Engineer can click the maintenance mode button to enter maintenance mode.After this operation,PC will play audio to prompt engineer to insert the card which can only be held by engineer.After PC recognizing information of card,it will jump to register page where engineer need to input correct password to enter maintenance page.The last page of maintenance mode is to control motors debugging themselves and PC can display some parameters about sensors.

4.2.1.3 Language Selection

There is a language selection button on main window.Users can switch language randomly before PC enter next page.

4.2.2 Maintenance Mode

4.2.2.1 Introduction to Maintenance Mode Functions

In maintenance mode,engineer need to input correct password after inserting engineer card to debug sensors and motors of this system.When debugging,some parameters will be displayed on PC screen.If any sensor or motors break down,user interface will display information to hint that engineer need to maintain this machine.

4.2.2.2 Structure of Process

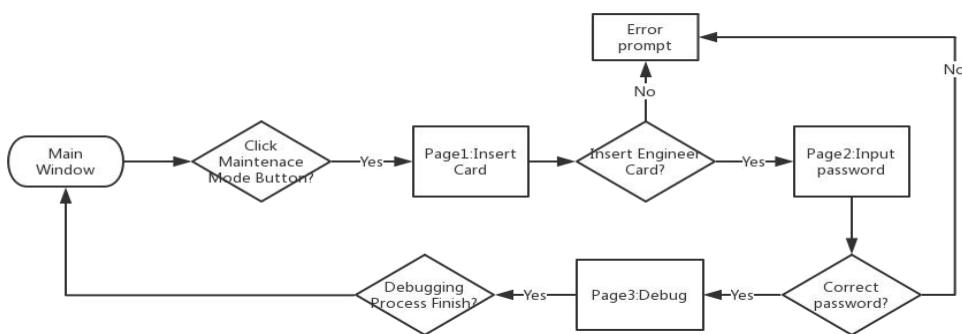


Figure 42. Flow chart of maintenance mode

4.2.2.3 UI Design of Maintenance Mode

Page 1:This page is to prompt engineer to insert engineer card,the button which is in the bottom of this page is designed to return main window.



Figure 43. UI design of Page1 of maintenance mode

Page 2: This page is to prompt engineer to key in password, the button which is in the bottom of this page is designed to return main window and another one which is in the light white rectangle is used for entering next page. Besides, there is a textbox which is container of password.

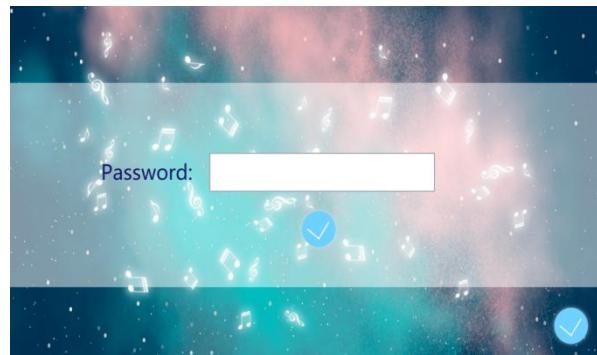


Figure 44. UI design of Page2 of maintenance mode

Page 3: This page is used to start debugging and there is a button whose content is start debugging. Below this button, there is a textbox which is designed to display parameters. As same as page1 and page2, a button in the bottom of this page is used for return main window.

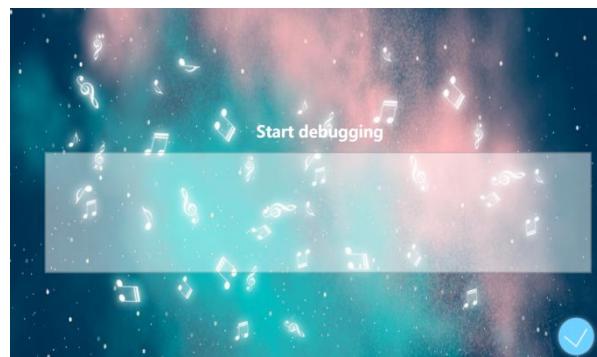


Figure 45. UI design of Page3 of maintenance mode

4.2.2.4 Main Code of Maintenance Mode

Password checking:

1. `//event which is triggered after clicking button in the center of Page2`
2. `private void Button_Click_1(object sender, RoutedEventArgs e)`
3. `{`

```

4.      //check password
5.      if (T2.Text == "111")
6.      {
7.          //enter Page3
8.          Window win = (Window)this.Parent;
9.          win.Content = new Maintenance3();
10.     }
11.    else
12.    {
13.        //error prompt
14.        if (MainWindow.lan == false)
15.            MessageBox.Show("密码错误!!!");
16.        else
17.            MessageBox.Show("Wrong password!!!");
18.    }
19. }
```

Other main codes are contained in following contents.

4.2.3.Operation Mode

4.2.3.1 Game Process

4.2.3.1.1 Introduction to Game Process Functions

In game process of operation mode,user can play game.In this process,to begin with,users need to insert card to identify their authentications and then they need to choose their emotions(happy or nervous) and the following step is to choose music style(popular or classic).After hearing the song snippet out,user can answer a question related with this song(song name or singer).

4.2.3.1.2 Structure of Process

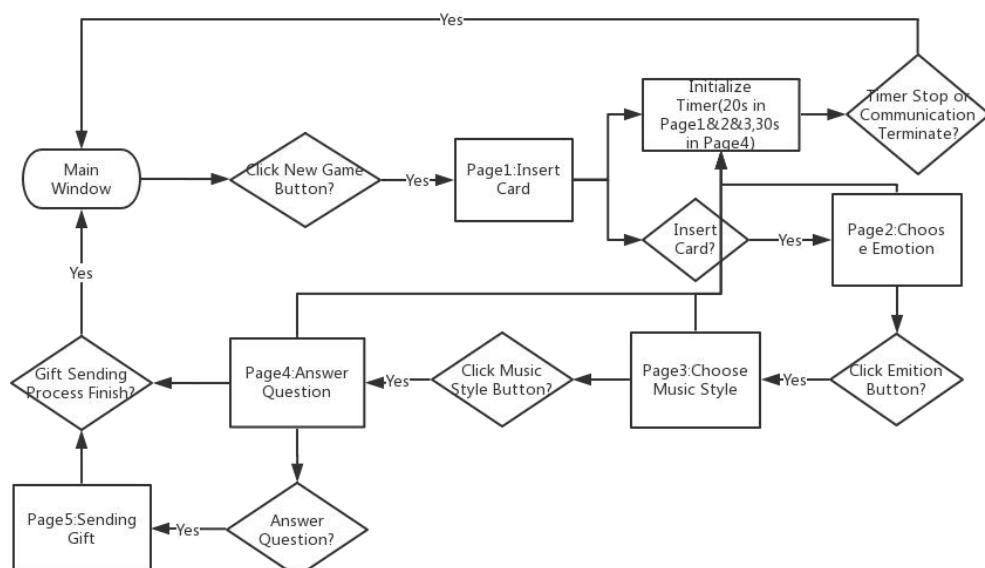


Figure 46. Flow chart of game process

4.2.3.1.3 UI Design of Game Process

Page 1:The page is same as page 1 in maintenance mode.

Page 2:There are three buttons in this page,one of which is used to return main window as same as in maintenance mode.And other buttons' contents are emotion,user can choose one of them which relates to following music which is played by PC and user interface enter next page.



Figure 47. UI design of Page2 of game process

Page 3:This page is similar with page2,the sole difference between page 2 and page 3 is that content of buttons which are in the center of user interface are music type.

Page 4:There are three labels whose contents are questions about music and in the left of each of them are a button,users can click relevant button to answer question.And there is also a button which can be controlled music replaying.



Figure 48. UI design of Page4 of game process

Page 4:There are three labels whose contents are questions about music and in the left of each of them are a button,users can click relevant button to answer question.And there is also a button which can be controlled music replaying.

Page 5:There is a label to show the game result of user and a area to display video which are used to give entertainment during sending gift process for user .

4.2.3.1.4 Main Code of Game Process

1. //event which is triggered after clicking one of three buttons which are used for answering question
2. **private void** Button_Click_2(**object** sender, **RoutedEventArgs** e)
3. {

```
4.      //pause music
5.      player.Close();
6.      /*close communication serialport and timer*/
7.      //open new page
8.      Window win = (Window)this.Parent;
9.      win.Content = new final();
10.     //judge if user's answer is correct
11.     if (T1.Text == answer)
12.     {
13.         /*update database information*/
14.         //record the status of result of answering question
15.         sf = true;
16.         if(Emotion.emotion == 1)
17.             /*send information to mbed*/
18.         else
19.             /*send information to mbed*/
20.     }
21.     else
22.     {
23.         //record the status of result of answering question
24.         sf = false;
25.         if(Emotion.emotion == 1)
26.             /*send information to mbed*/
27.         else
28.             /*send information to mbed*/
29.     }
30. }
```

Omitted and other main codes are contained in following contents.

4.2.3.2 Query Process

4.2.3.2.1 Two Modes of Querying Data

There are two modes when querying data, one of which is to check ranking list and another one is to query the chart about emotion distribution.

4.2.3.2.2 Information of Ranking

4.2.3.2.2.1 Ranking Querying Functions

If users want to know their total score which is recorded and updated every time they play this game and ranking positions among 15 users, they can click certain button to check it.

4.2.3.2.2.2 UI design of Ranking Table

There are a datagrid which contains items of ranking, user ID and score. In this page, user can change progress bar.

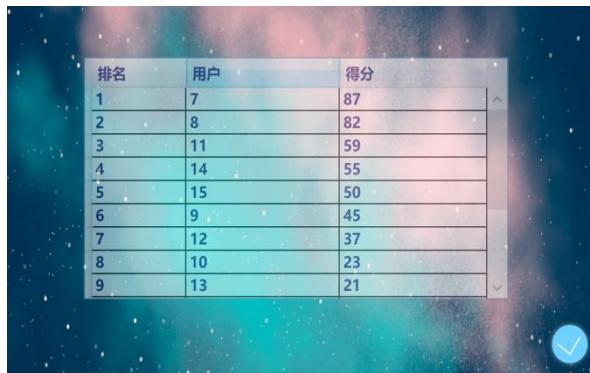


Figure49. UI design of ranking list page

4.3.3.2.2.3 Main Code of Ranking Table

Initialize and show ranking list:

```
1.      //event which is triggered when the page is initialized
2.      private void D1_Loaded(object sender, RoutedEventArgs e)
3.      {
4.          /*connect database with VS*/
5.          if (MainWindow.lan == true)
6.          {
7.              /*open and configure database*/
8.              DataTable dt = new DataTable();
9.              DataColumn col = new DataColumn("Ranking", typeof(int));
10.             dt.Columns.Add(col);
11.             mydata.Fill(dt);
12.             //add an item(ranking) for datatable(English)
13.             for (int i = 1; i < 16; i++)
14.             {
15.                 dt.Rows[i - 1]["Ranking"] = i;
16.             }
17.             //display content of datatable on datagrid
18.             D1.ItemsSource = dt.DefaultView;
19.             D1.Columns[0].Width = 80;
20.             D1.Columns[1].Width = 130;
21.             D1.Columns[2].Width = 125;
22.         }
23.         else
24.         {
25.             //Chinese version ...
26.         }
27.     }
```

4.2.3.2.3 Data of Emotion

4.2.3.2.3.1 Emotion Data Querying Functions

Users can choose to check the total number of happy users and nervous users and their proportions. There is a pie chart which is used to show these data.

4.2.3.2.3.2 UI design of Chart of Emotion Distribution

There is a emotion distribution pie chart which are implemented by using Visifire plug-in and a returning button in this page.



Figure50. UI design of emotion distribution page

4.2.3.2.3.3 Main Code of Emotion Distribution Chart

Initialize and show ranking table:

```
1.  private void Page_Loaded_1(object sender, RoutedEventArgs e)
2.  {
3.      /*configure and open database,execute select command on data table stored in SQL*/
4.      while (reader.Read())
5.      {
6.          /*store information gotten from data table in some variables*/
7.      }
8.      //declare items which are contained in pie chart,using List
9.      List<string> strListx = new List<string>() { "开心", "焦虑" };
10.     List<string> strListx1 = new List<string>() { "Happy", "Nervous" };
11.     List<string> strListy = new List<string>() { happy, nervous };
12.     GC.Children.Clear();
13.     //judge language and create pie chart by using a function which are declared,following this event
14.     if(MainWindow.lan==false)
15.         CreateChartPie("开心 v.s. 焦虑", strListx, strListy);
16.     else
17.         CreateChartPie("Happy v.s. Nervous", strListx1, strListy);
18.     }
19.     //function which are used to create a pie chart
20.     public void CreateChartPie(string name, List<string> valuex, List<string> valuey)
21.     {
22.         //create chart object and set some parameters
23.         Chart chart = new Chart();
24.         chart.Width = 380;
25.         chart.Height = 250;
26.         chart.Margin = new Thickness(10, 5, 10, 5);
27.         chart.ToolBarEnabled = false;
28.         chart.ScrollingEnabled = false;
```

```
29.     chart.View3D = true;
30.     chart.BorderThickness = new Thickness(0);
31.     chart.Background = new SolidColorBrush(Colors.White);
32.     chart.Background.Opacity = 0.5;
33.     //create title object which is shown above pie chart and set some parameters of it
34.     Title title = new Title();
35.     title.Text = name;
36.     title.Padding = new Thickness(0, 10, 5, 0);
37.     title.FontColor = new SolidColorBrush(Colors.MidnightBlue);
38.     title.FontSize = 17;
39.     //add title for chart
40.     chart.Titles.Add(title);
41.     //create dataseries object and declare a datapoint which is used to insert data to chart
42.     DataSeries dataSeries = new DataSeries();
43.     dataSeries.RenderAs = RenderAs.Pie;
44.     DataPoint dataPoint;
45.     //insert data into chart and set some parameters for displaying data
46.     for (int i = 0; i < valuex.Count; i++)
47.     {
48.         dataPoint = new DataPoint();
49.         if (i == 0)
50.         {
51.             dataPoint.Color = new SolidColorBrush(Colors.PaleTurquoise);
52.         }
53.         else
54.             dataPoint.Color = new SolidColorBrush(Colors.LightPink);
55.             dataPoint.LabelFontSize = 13;
56.             dataPoint.LabelFontColor = new SolidColorBrush(Colors.MidnightBlue);
57.             dataPoint.AxisXLabel = valuex[i];
58.             dataPoint.LegendText = "#" + valuex[i];
59.             dataPoint.YValue = int.Parse(valuey[i]);
60.             dataSeries.DataPoints.Add(dataPoint);
61.     }
62.     //update chart
63.     chart.Series.Add(dataSeries);
64.     Grid gr = new Grid();
65.     gr.Children.Add(chart);
66.     GC.Children.Add(gr);
67. }
```

4.2.3.2.4 Game Background

4.2.3.2.4.1 Game Background Functions

There is a animation which is similar with slide show and beside each image,text description is displayed and changed with the change of image.By watching it,user can know game background.

4.2.3.2.4.2 UI design of Game Background

There is a image control which is a container of images and a label which is used to display text description of relevant image.



Figure 51. UI design of game background page

4.2.3.2.4.3 Main Code of Game Background

```
1.     ObservableCollection<BitmapImage> bmList;
2.     // a index which is used to indicate the image status
3.     int index = 0;
4.     bool isRendering = false;
5.     public Background()
6.     {
7.         InitializeComponent();
8.         //a function which is used to update content for image control in xaml
9.         InitList();
10.        //a function which is used for adding image source and changing text description
11.        CompositionTarget.Rendering += new EventHandler(CompositionTarget_Rendering);
12.        BackgroundWorker bw = new BackgroundWorker();
13.        //function which is used to generate a stay duration for each image
14.        bw.DoWork += new DoWorkEventHandler(bw_DoWork);
15.        bw.RunWorkerAsync();
16.    }
17.    //record the image is displayed and pause changing picture for 3 seconds
18.    void bw_DoWork(object sender, DoWorkEventArgs e)
19.    {
20.        while (true)
21.        {
22.            isRendering = true;
23.            System.Threading.Thread.Sleep(3000);
24.        }
25.    }
26.    //update image source and show image in control
27.    public void InitList()
28.    {
29.        bmList = new ObservableCollection<BitmapImage>();
30.        for (int i = 1; i < 6; i++)
```

```
31.    {
32.        BitmapImage bmImg = new BitmapImage(new Uri(@"images/"+i+.png", UriKind.
33.            Relative));
34.    }
35. }
36. void CompositionTarget_Rendering(object sender, EventArgs e)
37. {
38.     if (isRendering)
39.     {
40.         if (index < bmList.Count)
41.         {
42.             this.imgViewer.Source = bmList[index];
43.             this.imgViewer.Width = this.imgViewer.Source.Width;
44.             this.imgViewer.Height = this.imgViewer.Source.Height;
45.             //change text description for change of image
46.             if (MainWindow.lan == false)
47.             {
48.                 if (index == 0)
49.                 {
50.                     /*show proper text*/
51.                 }
52.                 else if (index == 1)
53.                 {
54.                     /*show proper text*/
55.                 }
56.                 //else if ... else ... index from 0 to 4
57.             }
58.             else
59.             {
60.                 //English version ...
61.             }
62.             index++;
63.         }
64.         else
65.         {
66.             index = 0;
67.         }
68.         isRendering = false;
69.     }
70. }
```

4.2.4.Database

4.2.4.1 Functions of Database

There are three data tables in database constructed in SQL and database in this project has three functions.

- (1)Store ranking information.
- (2)Store emotion information.
- (3)Store music information and questions.

4.2.4.2 Connection between VS and SQL

Before executing some operations on data table,connections between VS and SQL has to be established so that operations can be executed successfully.

```

1. string con = "Data Source=.;Initial Catalog = Group04;User ID=login;
   Password=19980412";
2. SqlConnection connection = new SqlConnection(con);
3. SqlCommand cmd = connection.CreateCommand();
4. connection.Open();

```

After using database,close operation need to be executed.

```
1. connection.Close();
```

4.2.4.3 Introduction to Database in Two Processes of Operation Mode

In game process, database is used to store music information and questions.

In querying process, database is used to store emotion and ranking information.

4.2.4.4 Database Process Structure

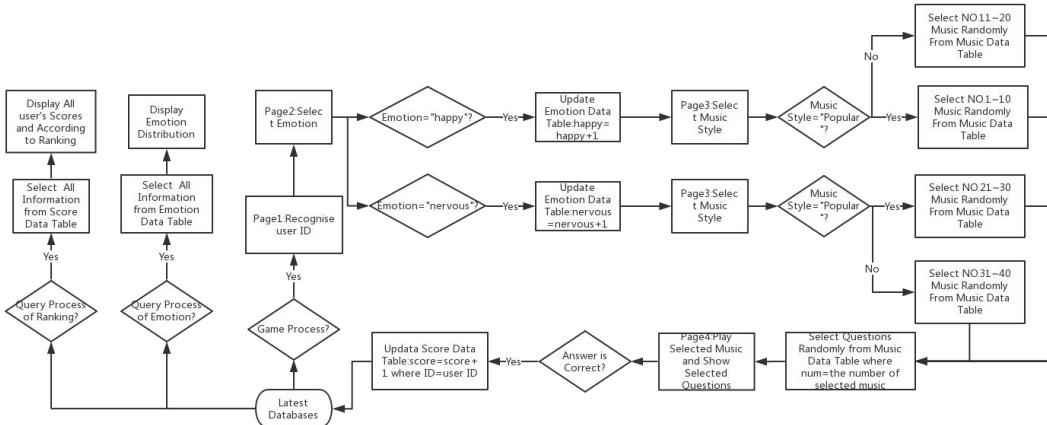


Figure52. flow chart of database

4.2.4.5 Database in Game Process

4.2.4.5.1 Data Structure in Game Process

There are 40 rows for every item. Some descriptions of each item are here.

number:a item which stores the music serial number which is equal to its store name so that we can indexes and play music by comparing item number with music store name.

name:a item which stores the music name about relevant item number.This is a choice in user interface's questions.

singer:a item which stores the singer of this song about relevant item number.This is a choice in user interface's questions.

n1&n2:wrong names of song.They are choices in user interface's questions.

s1&s2:wrong answers of singer of song.They are choices in user interface's questions.

	number	name	singer	n1	n2	s1	s2
▶	1	牵丝戏	银临	红昭愿	典狱司	双笙	任然
	2	讨厌	Fine乐团	喜欢	秘密	Fine	Fun
	3	忽然之间	莫文蔚	忽然	突然之间	王菲	张靓颖
	4	你是我的风景	何洁	风景	你就是我的风景	尚雯婕	张靓颖
	5	漂洋过海来看你	周深	漂洋过海	陌生的城市	何洁	张靓颖
	6	we do not tal...	Charlie Puth ...	anymore	we do not talk	Charlie Puth	Selena Gomez
	7	Nothing gonn...	Westlife	Nothing	Nothing gonn...	Boyzone	James Blunt
	8	Lemon tree	Fol Garden	Lemon Garden	Orange tree	Smart Garden	Garden
	9	You are not al...	Michael Jacks...	Alone	You are alone	Justin	Michael
	10	Here with you	Asher Book	With you	Here	Asher	Ash Book
	11	稻香	周杰伦	稻花	稻米	周华健	周立波
	12	敢问路在何方	蒋大为	With you	Here	佟大为	周大伟
	13	红豆	王菲	绿豆	黑豆	那英	蔡健雅
	14	旅行的意义	陈绮贞	旅行	原因	蔡健雅	王菲
	15	练习	刘德华	着急	开始练习	马德华	张学友
	16	Where Did Yo...	Reagan James	Where Do Yo...	Where	Reagann James	Reagan Jame

Figure53. data table of music

4.2.4.5.2 Data Invocation(Read) in C# Programme

```

1.    private void Page_Loaded_1(object sender, RoutedEventArgs e)
2.    {
3.        /*initialize timer*/
4.        //declare a object which is used to generate random number
5.        Random rd = new Random();
6.        //a variable which is used to store music item number(number in data table)
7.        int number;
8.        //a random variable which is used to adjust positions in user interface of three question choices
9.        int numch = rd.Next(1,7);
10.       //a random variable which is used to adjust question type(name or singer)
11.       int numsn = rd.Next(0,2);
12.       //select music in different positions of data table according to user's language,emotion and preference
13.       if (Emotion.emotion == 1 & Choose.kind == 1 & MainWindow.lan==false)
14.           number = rd.Next(1,6);
15.       else if (Emotion.emotion == 1 & Choose.kind == 1 & MainWindow.lan == true)
16.           number = rd.Next(6,11);
17.       else if (Emotion.emotion == 1 & Choose.kind == 2 & MainWindow.lan==false)
18.           number = rd.Next(11,16);
19.       else if (Emotion.emotion == 1 & Choose.kind == 2 & MainWindow.lan == true)
20.           number = rd.Next(16,21);
21.       else if (Emotion.emotion == 2 & Choose.kind == 1 & MainWindow.lan==false)
22.           number = rd.Next(21,25);
23.       else if (Emotion.emotion == 2 & Choose.kind == 1 & MainWindow.lan == true)
24.           number = rd.Next(25,31);
25.       else if (Emotion.emotion == 2 & Choose.kind == 2 & MainWindow.lan == false)
26.           number = rd.Next(31,36);

```

```
27.     else
28.         number = rd.Next(36,41);
29.     /*configure and open database*/
30.     //commands executed on data table,here is select command
31.     cmd.CommandText = "SELECT name,n1,n2,singer,s1,s2 FROM Group04.dbo.music WHERE number=@num";
32.     //set a parameter to indexes music whose number in data table is equal to variable which is declared as number
33.     SqlParameter paraNum = new SqlParameter("@num", System.Data.SqlDbType.Int);
34.     paraNum.DbType = System.Data.DbType.Int32;
35.     paraNum.Value = number;
36.     paraNum.Direction = System.Data.ParameterDirection.Input;
37.     cmd.Parameters.Add(paraNum);
38.     //execute select command
39.     SqlDataReader reader = cmd.ExecuteReader();
40.     if(true)
41.     {
42.         num = Convert.ToString(number) + ".mp3";
43.         //display questions on PC screen
44.         try
45.         {
46.             if (reader.Read())
47.             {
48.                 if (numsn == 0)
49.                 {
50.                     answer = reader.GetString(0);
51.                     if (numch == 1)
52.                     {
53.                         T1.AppendText(reader.GetString(0));
54.                         T2.AppendText(reader.GetString(1));
55.                         T3.AppendText(reader.GetString(2));
56.                     }
57.                     else if (numch == 2)
58.                     {
59.                         T1.AppendText(reader.GetString(0));
60.                         T2.AppendText(reader.GetString(2));
61.                         T3.AppendText(reader.GetString(1));
62.                     }
63.                     //else if ...    else ...
64.                 }
65.             else
66.             {
67.                 answer = reader.GetString(3);
68.                 if (numch == 1)
```

```
69.         {
70.             T1.AppendText(reader.GetString(3));
71.             T2.AppendText(reader.GetString(4));
72.             T3.AppendText(reader.GetString(5));
73.         }
74.         //else if ... else ...
75.     }
76. }
77. /*close connection between VS and SQL*/
78. }
79. catch
80. {
81.     MessageBox.Show("发生错误！");
82. }
83. //play music
84. player.Open(new Uri(num, UriKind.Relative));
85. player.Play();
86. }
87. }
```

4.2.4.6 Database in Query Process

4.2.4.6.1 Database of Emotion Statistics

4.2.4.6.1.1 Data Structure of Emotion Statistics Table

There are only two items to record the number of two emotion, one of which is declared as happy and the other is declared as nervous in SQL. And there is only one row in this data table, the contents of two blocks are the number of each emotion.

	happy	nervous
▶	236	77
*	NULL	NULL

Figure54. data table of emotion

4.2.4.6.1.2 Data Invocation(Read and Update) in C# Programme

```
1. //event including read operation executed on database
2. private void Page_Loaded_1(object sender, RoutedEventArgs e)
3. {
4.     /*configure and open database*/
5.     //select command
6.     SqlCommand cmd = connection.CreateCommand();
7.     cmd.CommandText = "SELECT * FROM Group04.dbo.ms";
8.     //execute sql command
9.     SqlDataReader reader = cmd.ExecuteReader();
10.    //store read result of two emotion from data table into two variables
11.    while (reader.Read())
12.    {
13.        int hap = reader.GetInt32(0);
```

```

14.     int ner = reader.GetInt32(1);
15.     happy = hap.ToString();
16.     nervous = ner.ToString();
17. }
18. /*generate chart*/
19. }
1. //event which is triggered after user click emotion button
2. private void Button_Click_1(object sender, RoutedEventArgs e)
3. {
4.     /*stop communication with mbed and navigate to next page*/
5.     /configure and open database*/
6.     //update command
7.     cmd.CommandText = "UPDATE Group04.dbo.ms SET happy=happy+1";
8.     //execute command
9.     cmd.ExecuteNonQuery();
10.    /*close connection between VS and SQL*/
11.    //recorder which records current user's emotion
12.    emotion = 1;
13. }

```

4.2.4.6.2 Database of Ranking Information

4.2.4.6.2.1 Data Structure of Score Statistics Table

There are two columns in this data table. One of them is to indicate user ID and the other is store score. 15 users relate to ID from 1 to 15.

	ID	Score
▶	1	9
	2	9
	3	7
	4	6
	5	5
	6	14
	7	87
	8	82
	9	45
	10	23
	11	59
	12	37
	13	21
	14	55
	15	50
*	NULL	NULL

Figure55. data table of users' scores

4.2.4.6.2.2 Data Invocation(Read and Update) in C# Programme

```

1. private void D1_Loaded(object sender, RoutedEventArgs e)

```

```
2.    {
3.        if (MainWindow.lan == true)
4.        {
5.            /*configure and open database*/
6.            //select all scores and descending sort
7.            string sql = "SELECT TOP(15) ID,Score FROM Group04.dbo.Ranking_Table ORDER BY Score DESC";
8.            /*add data to datagrid and display it on PC screen*/
9.        }
10.       //else ...
11.    }
```

Update operation is similar with that in emotion data updating.Omit codes here.

4.2.5. Communication

4.2.5.1 Communication with mbed

4.2.5.1.1 Communication Process Structure

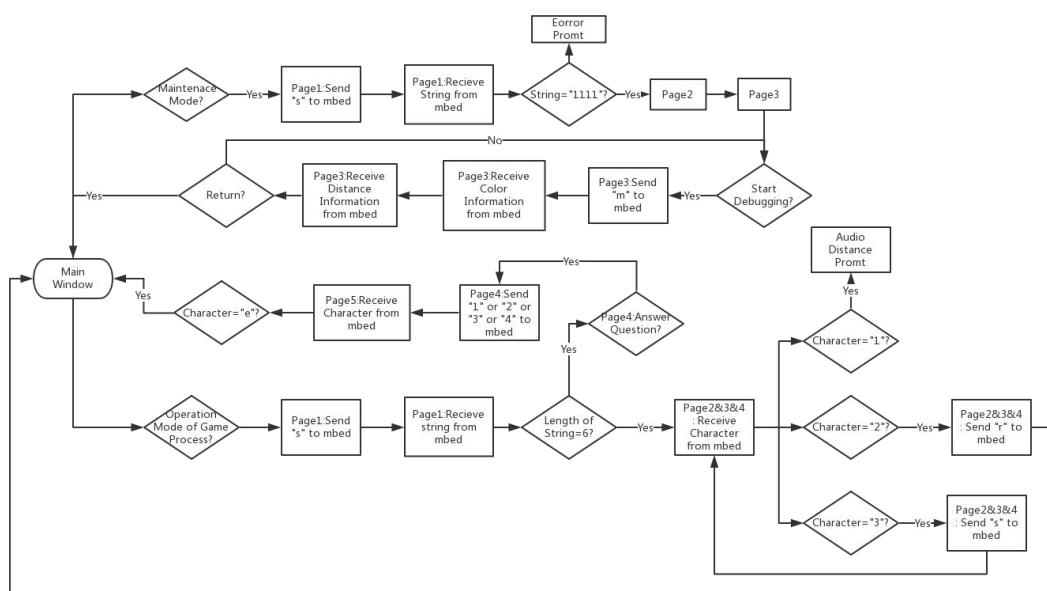


Figure56. flow chart of communication

4.2.5.1.2 Communication in Maintenance Mode

4.2.5.1.2.1 Data Sending of Maintenance Mode

When communicating with mbed, PC send specific character to control mbed. Here are details.

4.2.5.1.2.2 Data Receiving(Multi-thread Programming) of Maintenance Mode

When communicating with mbed, PC receive specific character or string from mbed and according these characters to control user interface. Here are details.

In page 1 of maintenance mode:

send s:tel mbed to send engineer ID to PC

In page 3 of maintenance mode:

send m:tell mb6d to enter maintenance mode

In page 1 of maintenance mode:

receive 1111:indicate this card inserted is engineer card.

```
1. //function operated every time PC get information from mbed
2. private void SPDataReceived1(object sender, SerialDataReceivedEventArgs e)
3. {
4.     string command;
5.     command = COM1.ReadExisting();
6.     string select = command.Remove(2);
7.     string num = command.Substring(2);
8.     this.Dispatcher.Invoke(new Action(() =>
9. {
10.     //check if card inserted is engineer card
11.     if (num == "1111")
12.     {
13.         /*jump to next page*/
14.         COM1.DataReceived -= new SerialDataReceivedEventHandler(SPDataReceived1);
15.     }
16.     else if (select == "00" && num != "1111")
17.     {
18.         if(MainWindow.lan==false)
19.             MessageBox.Show("请插入工程师卡");
20.         else
21.             MessageBox.Show("Please insert the card of engineer");
22.     }
23.     COM1.DiscardInBuffer();
24. });
25. }
```

In page 3 of maintenance mode:

receive r or g or b or y:indicate mbed send correct information of sensor color to PC and PC should display them.

receive n:indicate color sensor break down.

receive a string:indicate mbed send information of distance sensor and PC should display it.

receive 0:indicate distance sensor breakdown.

```
1. private void SPDataReceived2(object sender, SerialDataReceivedEventArgs e)
2. {
3.     string command;
4.     this.Dispatcher.Invoke(new Action(() =>
5. {
6.     command = Maintenance1.COM1.ReadExisting();
7.     Maintenance1.COM1.DiscardInBuffer();
8.     if (MainWindow.lan == false)
9.     {
10.        //receive information about sensor color
```

```
11.         if (command == "r" | command == "g" | command == "b" | command=="y")
12.     {
13.         T.AppendText("颜色: ");
14.         if (command == "r")
15.             T.AppendText("红" + Environment.NewLine);
16.         else if (command == "g")
17.             T.AppendText("绿" + Environment.NewLine);
18.         else if(command=="b")
19.             T.AppendText("蓝" + Environment.NewLine);
20.         else
21.             T.AppendText("黄" + Environment.NewLine);
22.     }
23.     else if (command == "n")
24.         T.AppendText("颜色: 错误！！！" + Environment.NewLine);
25. //receive information about distance color
26.     else if (command == "0")
27.         T.AppendText("距离: 错误！！！" + Environment.NewLine);
28.     else
29.     {
30.         T.AppendText("距离: " + command + Environment.NewLine);
31.     }
32. }
33. else
34. {
35.     //English version ...
36. }
37. });
38. }
```

4.2.5.1.3 Communication of Game Process in Operation Mode

4.2.5.1.3.1 Data Sending of Operation Mode

When communicating with mbed,PC send specific character or string to control mbed. Here are details.

In page 2&3&4 of game process:

send r:control mbed stopping the whole machine.

send s:tell mbed to send distance information or user ID to PC.

send m:tell mbed to controll the machine entering maintenance mode.

In page 5 of game process:

send 1 or 2 or 3 or 4 :tell mbed to control sorter pushing box of proper color.

4.2.5.1.3.2 Data Receiving(Multi-thread Programming) of Operation Mode

When communicating with mbed,PC receive specific character from mebed and according these characters to control user interface. Here are details.

In page 1 of game process:

receive string whose length is equal to 6:indicate that string received is user ID.

In page 2&3&4 of game process:

receive 1:indicate that user is out of game scope.

receive 2:indicate that user is out of game scope for some duration so that game need to be terminated.

receive 3:indicate that user which is out of game scope just now comes back,game continues.

In page 5 of game process:

receive e:indicate the process of pushing box is over and PC need to come back to main window.

```
1. //function operated every time PC get information from mbed
2. public void SPDataReceived1(object sender, SerialDataReceivedEventArgs e)
3. {
4.     string command;
5.     this.Dispatcher.Invoke(new Action(() =>
6.     {
7.         //information received from mbed
8.         command = COM1.ReadExisting();
9.         COM1.DiscardInBuffer();
10.        if (true)
11.        {
12.            if (command == "1")
13.            {
14.                if (MainWindow.lan == true)
15.                {
16.                    spVoice.Speak("Out of game area");
17.                }
18.                else
19.                    spVoice.Speak("您已不在游戏范围内");
20.            }
21.            else if (command == "2")
22.            {
23.                /*some operations on timer and audio*/
24.                COM1.WriteLine("r");
25.                COM1.DiscardInBuffer();
26.                COM1.DataReceived -= new SerialDataReceivedEventHandler(SPDataReceived
27.                1);
28.                /*termination*/
29.            }
30.            else if (command == "3")
31.            {
32.                COM1.WriteLine("s");
33.            }
34.            else if (command == "e")
35.            {
36.                COM1.WriteLine("r");
37.            }
38.        }
39.    }
40. }
```

```
36.             COM1.DataReceived -= new SerialDataReceivedEventHandler(SPDataReceived
1);
37.             /*come back to main window*/
38.         }
39.         else if (command.Length == 6)
40.         {
41.             //calculate user ID which need to be used in update ranking list
42.             string num = command.Substring(2);
43.             string select = command.Remove(2);
44.             string userID1 = command.Substring(2, 1);
45.             string userID2 = command.Substring(3, 1);
46.             string userID3 = command.Substring(4, 1);
47.             string userID4 = command.Substring(5, 1);
48.             int userID1I = Convert.ToInt32(userID1);
49.             int userID2I = Convert.ToInt32(userID2);
50.             int userID3I = Convert.ToInt32(userID3);
51.             int userID4I = Convert.ToInt32(userID4);
52.             userID = 8 * userID1I + 4 * userID2I + 2 * userID3I + 1 * userID4I;
53.             COM1.DataReceived -= new SerialDataReceivedEventHandler(SPDataReceived
1);
54.             /*stop timer and jump to next page*/
55.         }
56.     }
57. });
58. }
```

4.2.5.2 Communication in WPF

4.2.5.2.1 Communication between Window and Pages

(1)Window to Page:

```
1. //this is an example,Ranking can be substituted as any page's name which you want to jump
p to
2. this.Content = new Ranking();
```

(2)Page to Window

method1: normal skip between window and page whose parent is main window.

```
1. Window win = (Window)this.Parent;
2. MainWindow mw = new MainWindow();
3. //fix new window's position as same as current one
4. win.WindowStartupLocation = WindowStartupLocation.Manual;
5. mw.Left = win.Left;
6. mw.Top = win.Top;
7. //close current page and open a new window
8. mw.Show();
9. win.Close();
```

method2: shutdown whole programme to prevent interface from going to die.

```
1.     this.Dispatcher.Invoke((ThreadStart)delegate()
2.     {
3.         Application.Current.Shutdown();
4.     });
5.     System.Diagnostics.Process.Start(System.Reflection.Assembly.GetExecutingAssembly().L
ocation);
```

4.2.5.2.2 Communication between Different Pages

```
1.
2. //this is an example,Emotion can be substituted as any page's name which you want to jump to
3. Window win = (Window)this.Parent;
4. win.Content = new Emotion();
```

4.2.6.Mechanism of Error Processing

4.2.6.1 Timers

In this system, we regulate operable time for users, if user has no response on interface for some duration, the interface will initialize automatically. So we need a timer to finish this procedure. Implementation codes are here.

```
1. //declare a timer and start counting when page is loaded
2. dtimer = new System.Windows.Threading.DispatcherTimer();
3. dtimer.Interval = TimeSpan.FromSeconds(1);
4. dtimer.Start();
5. //function which is operated every 1 second
6. dtimer.Tick += dtimer_Tick;
7. void dtimer_Tick(object sender, EventArgs e)
8. {
9.     //count down
10.    countsecond--;
11.    if (countsecond == 0)
12.    {
13.        //stop timer
14.        dtimer.Stop();
15.        /*other operations needed*/
16.    }
17. }
```

4.2.6.2 Distance Prompt

Every time user is out of game scope, PC will prompt him. Details about communication are in 4.2.5.1.3 and that about audio prompt are in 4.2.7.3 .

4.2.7 Other Functions

4.2.7.1 Animation of Main Window

4.2.7.2.1 Algorithm of Animation

By using reflection theory, animation of bubble doing reflection motion within window can be designed.

```
1.      //function which is operated every 50ms in this programme by declaring a dispatcherTi
mer
2.      private void Timer_Tick(object sender, EventArgs e)
3.      {
4.          get position of image of bubble in whole screen
5.          double X = Canvas.GetLeft(Spirit);
6.          double Y = Canvas.GetTop(Spirit);
7.          //get relative position of bubble compared to left top point of screen
8.          Window window = Window.GetWindow(Carrier);
9.          Point point = Spirit.TransformToAncestor(window).Transform(new Point(0,0));
10.         if (AniF == false)
11.         {
12.             //parameters can adjust according to size of screen and window
13.             //when bubble knock into left or right edge of window,change speed of bubble in x
direction
14.             if ((Convert.ToInt32(point.X) >= 1169 & Convert.ToInt32(point.X) <= 1177) | Convert.
ToInt32(point.X) <= 5)
15.             {
16.                 speedx = 0 - speedx;
17.             }
18.             //when bubble knock into top or bottom edge of window,change speed of bubble i
n y direction
19.             if ((Convert.ToInt32(point.Y) <= 715 & Convert.ToInt32(point.Y) >= 709) | Convert.To
int32(point.Y) <= 5)
20.             {
21.                 speedy = 0 - speedy;
22.             }
23.         }
24.         //update position of bubble,initial position is (0,0)
25.         Canvas.SetLeft(Spirit, X + speedx);
26.         Canvas.SetTop(Spirit, Y + speedy);
27.         AniF = false;
28.     }
```

4.2.7.3 Video & Audio

Video:In xaml,insert a MediaElement control and in C#,added source for MediaElement to display video.

```
1.      private void Page_Loaded_1(object sender, RoutedEventArgs e)
2.      {
3.          /*codes about communication*/
4.          //judge the result of answering question
5.          if (Select.sf == true)
6.          {
7.              if(MainWindow.lan==false)
8.                  T.Text= "回答正确";
```

```
9.         else
10.            T.Text = "True";
11.        //open player
12.        /*about playing music*/
13.        //add source
14.        mediaElement.Source = new Uri("C:/true.mp4");
15.      }
16.    else
17.    {
18.      ///false answering version ...
19.    }
20.    //play video
21.    mediaElement.Play();
22.    /*codes about timer*/
23.  }
```

Audio:(1)audio prompt:In C#,declare a SpeechSynthesizer object and use its SpeakAsync() method to implement audio prompt and introduction of product.

This is an example using in product introduction.

```
1.  //declare a object
2.  SpeechSynthesizer spVoice = new SpeechSynthesizer();
3.  //event which is triggered when user click product introduction button
4.  private void Button_AudioPlay(object sender, RoutedEventArgs e)
5.  {
6.    //obtain content of introduction
7.    string TTS;
8.    TTS = T.Text;
9.    //if user click this button when the audio introduction is pausing
10.   if (first)
11.   {
12.     //if user click this button at the first time
13.     if (F)
14.     {
15.       spVoice.SpeakAsync(TTS);
16.       first = false;
17.       F = true;
18.     }
19.     //if user click this button when the audio introduction is pausing
20.     else
21.     {
22.       spVoice.Resume();
23.       first = false;
24.     }
25.   }
26.   //if user click this button when the audio introduction is playing
```

```
27.     else
28.     {
29.         spVoice.Pause();
30.         first = true;
31.     }
32. }
```

(2)play music:In C#,declare a MediaPlayer object and then add path for file,finally,use its Play() method to play audio.

```
1.     MediaPlayer player = new MediaPlayer();
2.     player.Open(new Uri(num, UriKind.Relative));
3.     player.Play();
```

4.2.7.4 Language Switching

By using a public bool variable "lan" to record the language user selected in main window,in other pages,programme can judge value of this variable to implement language switching.

```
1.     //event which is triggered when user click language switching button
2.     private void Button_Click_5(object sender, RoutedEventArgs e)
3.     {
4.         if (lan == false)
5.         {
6.             /*change contents in current window*/
7.             //indicate English
8.             lan = true;
9.         }
10.        else
11.        {
12.            /*change contents in current window*/
13.            //indicate Chinese
14.            lan = false;
15.        }
}
```

5.Sensors

5.1 TCS3472_I2C Color sensor

5.1.1 Measuring and Modulation

In our real project, the color sensor can be influenced by many different aspects listed below, if we don't consider the following aspects, the final result which we test by the sensors will be distinct with the practical condition. we use the scientific method to confirm each influence factor and do the correction to our algorithm.

- The distance between color sensor and objects.

We do the several tests, each test to detect four different colors(red/blue/yellow) in the same nature light intensity but different distance(0.5cm,1cm,1.5cm). The final results are in the table below.

0.5cm	R value	G value	B value
Red	123	22	13
Green	23	114	14
Blue	12	17	65
Yellow	114	120	15

1cm	R value	G value	B value
Red	79	13	19
Green	25	98	19
Blue	18	21	42
Yellow	89	90	19

1.5cm	R value	G value	B value
Red	50	24	15
Green	19	50	21
Blue	25	23	32
Yellow	61	59	17

By analysis the data in the form, we can find when the distance between color sensor and objects is around 0.5cm, the detections get the better result.

However, that doesn't mean the closer the sensor is, the better the detection result is. This is because The rotating block might bump into the color sensor if the distance is too close which will totally interrupt the process.

Besides too close distance will cause the sensor's built-in led light to interfere with the detection results.

Owing to these theorem, We design the actual mechanical position.



Figure 57 the relevant position of the TCS3472_I2C Color sensor with the objects

- The color purity of objects

In order to solve the color purity problem, the boxes are made by the paper which are printed in the pure color such as red(255,0,0),blue(0,0,255),green(0,255,0), yellow(255,255,0).

This method will greatly increase the probability of successful detection.

- The nature light intensity

We put the color sensor in the game box. When we close the door of the box, we greatly reduce the influence of ambient light.

5.1.2 algorithm

From the libraries on theMBED cloud, we can use the functions to get the RGB value directly. As I mentioned before, it is very common for the variance of the detected data owing to the tiny change of the environmental reasons (The distance between color sensor and objects, The color purity of objects, The nature light intensity). Therefore, The algorithm looks strange after modification.

```
1. char colour_1(){
2.     uint8_t a= rgb_sensor.getRedData();
3.     uint8_t b= rgb_sensor.getGreenData();
4.     uint8_t c= rgb_sensor.getBlueData();
5.     long w; //距离黄色的距离 unit8 0~255
6.     long x; //距离红色的位置
7.     long y; //g
8.     long z; //b
9.     w=((120-a)*(120-a)+(120-b)*(120-b)+(10-c)*(10-c)); //c 语言没有乘方符号
   ^ !!!!!
10.    x=((120-a)*(120-a)+(10-b)*(10-b)+(10-c)*(10-c));
11.    y=((10-a)*(10-a)+(115-b)*(115-b)+(10-c)*(10-c));
12.    z=((10-a)*(10-a)+(10-b)*(10-b)+(60-c)*(60-c));
13.    if((a>=15) || (b>=15) || (c>=15))
14.    {
15.        if((x>=w)&&(x>=y)&&(x>=z)){
16.            return 'r';
17.        }else if((w>=x)&&(w>=y)&&(w>=z)){
18.            return 'y';
19.        }else if((z>=x)&&(z>=y)&&(z>=w)){
20.            return 'b';
21.        }else if((y>=x)&&(y>=w)&&(y>=z)){
22.            return 'g';
23.        }else
24.            return 'n';
25.    }
26.    else
27.        return 'n';
28.
29. }
```

In the algorithm, we set a Rectangular coordinate system. The red blue green yellow colors are transformed in to the point in the coordinate. they are the four standard points. Of course, their Coordinate values are modulated during the several experiments. Once the new (R,G,B) point detected, it will be put into the coordinate and the distance between the point and the standard points(w x y z) will be calculated and compared. The detected value will be specified to the standard color whose distance is smallest.

We can get the color information from the return data.

5.2 VL1680 TOF Sensor.

5.2.1 Modulation

The VL1680 TOF Sensor is used to detect the distance by calculating the time between the infrared light send by the sender and receive by the receiver.

Owing to this reason, it is important to avoid bright light and the bright light will influence the sensor hugely.



Figure 58 distance sensor

5.2.2 algorithm

From the libraries on the MBED cloud, we can use the functions to get the distance which is the number between 0 to 255. If the distance is larger then 20cm, the value of the return value will always be 255.

Firstly we will set the pins

```
1. VL6180 TOF_sensor(p28, p27); // 距离传感器 27-yellow-clock 28-orange-data
```

Then we will initiate the sensor

```
1. TOF_sensor.VL6180_Init(); //initialize the distance sensor
```

Use the function, we can get the distance information

```
1. dist = TOF_sensor.getDistance();
```

5.3 infrared light sensors (card read)

5.3.1 simple introduction

In order to read the 4 bits card information, we need to utilize four infrared light sensor. There are two The signal ends. One is transmitter for sending the infrared light, and another is receiver for receiving the infrared light. Its working principle is that light emitted from the transmitter into the air, Once the infrared light is reflected by the objects and received by the receiver, it will output a high level voltage.

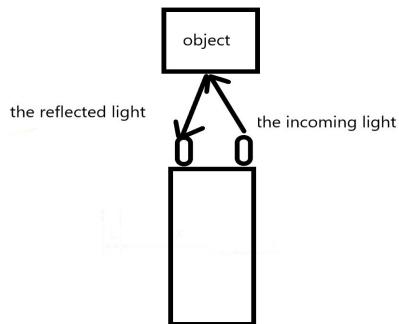


Figure 59 Schematic diagram of principle



Figure 60 Sensor physical drawing

5.3.2 modulation

The infrared light sensor can be also influenced by other aspects, there are two main aspects:

- Mutual interference

This interference is mainly because that the infrared light which is send by the sensor 1 is received by another sensors

In order to solve this problem, we use small boards to separate the sensors.

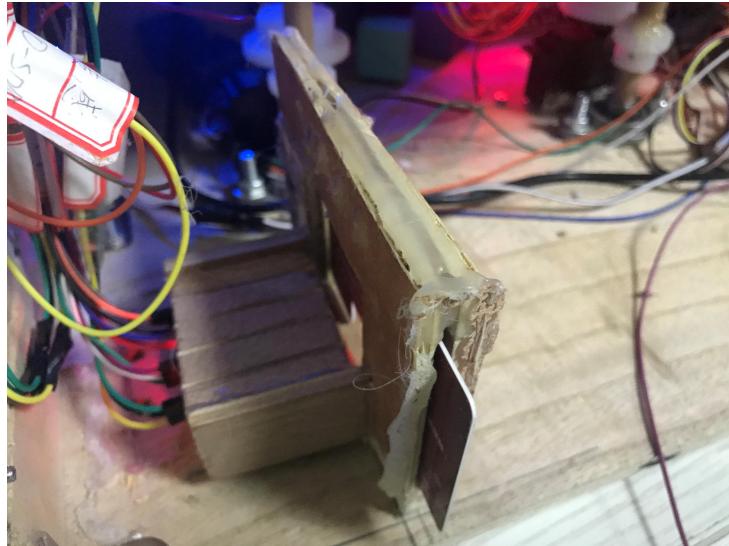


Figure 61 infrared sensors

- Diffuse reflection in the box

This is mainly because that when we close the box, The inside of the box is closed. If there is no black infrared absorbing wallpaper on our walls, then infrared rays will randomly reflect inside the box, greatly affecting the work of the sensor.

In order to solve this problem, We put a black sticker right under the sensor.

5.3.3 algorithm

Firstly we will set the pins

```
1. DigitalIn sensor1(p21);
2. DigitalIn sensor2(p22);
3. DigitalIn sensor3(p23);
4. DigitalIn sensor4(p24);
```

Then We detect the voltage level of the pins and mix these voltage level information together into a 4-bit decimal numbers.

```
1. int readcard(int sensor1,int sensor2,int sensor3,int sensor4){
2.     int key;
3.     int a=sensor1;
4.     int b=sensor2;
5.     int c=sensor3;
6.     int d=sensor4;
7.     a=a*1000;
8.     b=b*100;
9.     c=c*10;
10.    key=a+b+c+d;
11.    wait(1);
12. //    pc.printf("%06d",key);//00xxxx
13.    return key;
```

14. }

5.4 Circuit Diagram

All the pins have something to do with MBED have shown in the circuit.

- Blue lines: signal lines
- Red lines: power line
- Black lines: ground line

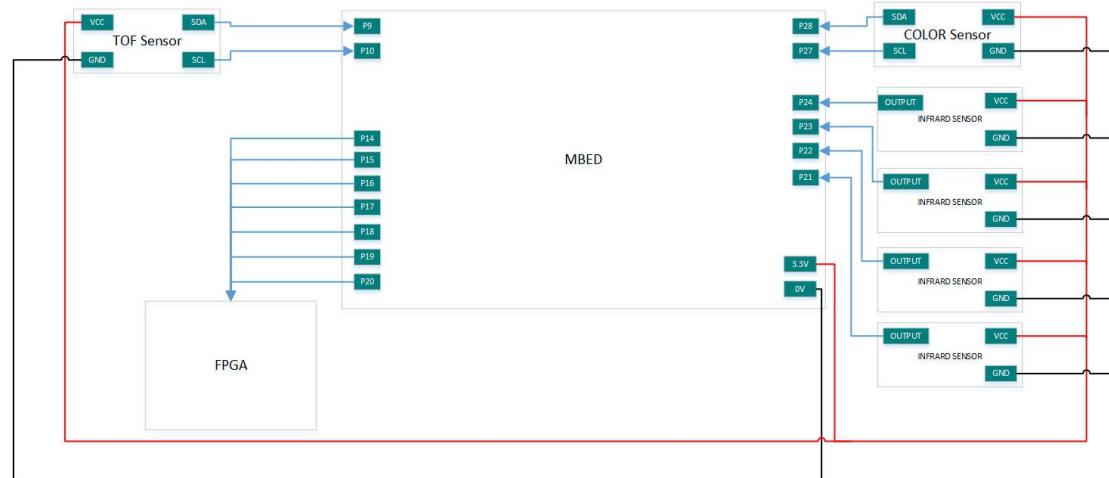


Figure 62 Circuit diagram

6. Marketing and web

6.1 Market analysis

6.1.1 Overview of Product Market Development

6.1.1.1 Capacity of Market Development and Forecast

With the improvement of people's living standards, studies show that the passenger flow of Chinese playgrounds has been increasing year by year. Due to the limited entertainment facilities can't meet the huge passenger flow, the queue time is getting too long. Based on this problem, the development of entertainment facilities which are suitable for using during queuing is crucial to the development of the playground industry and the demand for such products is becoming more and more obvious.

6.1.1.2 Customers' Requirements

On the one hand, during the queuing process, most tourists need entertainment products to kill time. And in this process, tourists are prone to fall into anxiety, as a consequence, they need such products that can be used to calm their emotions. On the other hand, the age distribution of the playground is relatively wide, mainly including young people, children and their parents, so visitors need a product suitable for most age groups.

6.1.2 Analysis of Current Product Status

6.1.2.1 Customer Structure Analysis

First of all, from our product positioning, it is a game machine. Obviously, the First level customers and buyers are mainly marketing managers for amusement parks and game halls, After that, the players who use our machines for entertainment are also our customers, this is the second level customers.

Secondly, our product positioning is the psychological relief machine which can relieve the pressure and ease the mood. From this perspective, some psychological consultation rooms may purchase our products. Therefore, the first level user is the psychological consultant in the consultation room. Secondary customers are people who use our machines for psychological counseling.

6.1.2.2 Improvements Needed of Product

In order to meet the needs of people of all ages and of different musical preferences, the product not only needs to provide more bigger database and more flexible mapping structure to store music information, but also needs more memory space for storing music. In addition, in order to ensure that the product is suitable for queue lines with different flow speeds, it is necessary to adjust the game time. If the queue line flows slow, the whole game time and the number of game times per person should be extended taking advantage of programme. And for the mechanical part, the current product needs to enhance the visual effects or add the entertainment structures.

6.1.2.3 Sales Data of Similar Product Analysis

Similar Product is a term used in economics. Refers to the fact that several commodities that can bring consumers similar satisfaction are able to replace each other. If the price of commodity A rises, customers will seek for commodity B that is cheaper than commodity A and can bring similar satisfaction.

For our product, our similar product is the console that soothes people's mind. There are few similar machines on the market But there are many similar applications, unfortunately they all lack our physical capabilities.

6.2 Product advantage

Our product advantages are mainly reflected in cost, technology and quality.

1,Cost advantage:We rely on low costs to achieve higher profitability than our peers. Our company achieves cost advantage through economies of scale, proprietary technology and preferential raw materials. Through price competition, we have made the same industry hesitant. Then improve the competitiveness of our products.

2,Technological superiority:Our employees have stronger technical strength and the ability to research and develop new products. Through technological innovation, we can develop new processes and reduce the original production cost.

3,Quality advantage:The quality of products is always an important factor affecting consumer's purchasing tendency. Quality is the guarantee of product credit. We strictly manage and continuously improve the quality of our products. So as to increase the market share of products and achieve competitive advantage.

6.3 Product publicity

In order to effectively let users better understand our products. We promote our products in various ways. For example, poster, promotional video and web page publicity. Combining in these ways to make people know about our products and try to use it. Next I'll introduce you to our propaganda means.

6.3.1 Poster



Figure 63 Final Poster

Our poster backgrounds are mainly use red and gray. Corresponding to our products is red color represents joy and gray color represents sadness. The record in the middle of the poster represents the main function of our product. The main function of our product is to relax the user's mood through listening to the music.

6.3.2 Promotional video

We also produced promotional videos for our company and product. The advantage of video is to show our product more directly to people. It has rich content and direct information about our product. What's more, video is more attractive to users.

6.3.3 Website

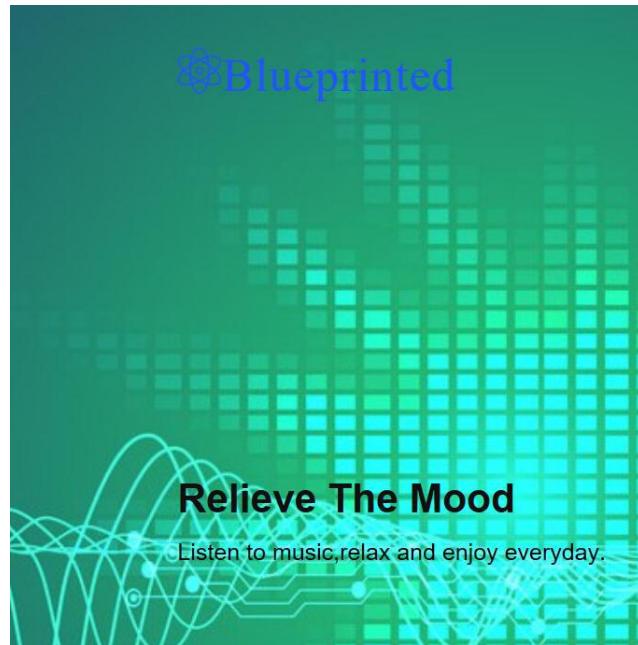


Figure 64 Website

In order to make it easier for users to understand our product information on the Internet. We also provide web page for them. It includes multiple services in our homepage. Such as product information, company cooperation, team introduction, contact information and so on. All of these things can be found on our website.

6.3.4 Exhibition stand

At the final exhibition, we designed the display stand for our product. On the stand board, we stuck our poster and our product name--Music Box. And we made some stickers to make it look more beautiful.



Figure 65 Exhibition Stand Board

6.4 Team Member Biography

Haoxuan Du is the general manager of the project. He is responsible for the overall planning and management of team members. He has a good overall outlook and management ability. He can find out the problems of the project and put forward original opinions. He will also work with employees to overcome difficulties.

Zeyu Xia is the digital system engineer. He has a strong knowledge of data structure and algorithm .He also has strong coding ability and problem analysis ability. He can flexibly use appropriate algorithms to solve problems. What ' s more,He can accomplish his task efficiently.

Ruoxi Gao is the software engineer. She is responsible for the design,development and testing of software engineering systems. She can use her professional knowledge of software engineering to solve key problems and technical problems in engineering.

Yuze Sun is the secretary and chief mechanical design engineer of our company. He is responsible for recording project meeting and writing project plans. He is also responsible for the mechanical design of the products and selection of related materials. He is familiar with the working principle and basic structure of various machines.He can skillfully use sports design, structural design to meet customer needs.

Senpu Yuan is the web designer.He is responsible for designing the homepage of our company,publicizing and promoting the product on the Internet. He can reasonably use words,pictures and animation to design layout for web pages.

Yutong Xu is the workshop engineer. He is responsible for the construction of our machine. With a strong confidence, he can deal with various issues of fixation and adhesion. At any critical moment, he gives some pragmatic approaches to our team, making great contributions.

7. Project Management

7.1 Task routine

Task processor
Decide what product to make
Each team member learns the corresponding software system
web designer design menu bar
software engineer learn object-oriented programming and wpf (windows presentation foundation) programming
workshop engineer receive wood working training
Chief digital systems engineer completes the card identification module
Chief digital systems engineer drives servomotor
Install and debug the communication serial port of each sensor
web designer add various modules related to our company information
software engineer develops Desktop programs, write the basic Framework for user interface

System Project Technology Report
Group 4

CAD designers hand down the three dimensional model and three view of shooting machine, then compute and label the corresponding size
software engineer writes the login and registration interface of users
CAD designers use CAD to make the three-view drawing and begin the work of building three-dimensional model.
web designer add relevant text information and picture information
software engineer operates, access databases and data binding
CAD designers finish the work of building model and plotting and begin designing the size of components
workshop engineer buying components and steel pipe
Chief digital systems engineer implements scoreboard display numbers
Implement the communication between mbed and UI and interface which is used to display data
Consider wire allocation and the specific location in the base of shooting machine
workshop engineer builds the solid model
web designer perfect and optimize the WEB
beautify UI
Design posters and promote video
debug programme, detect and modify problems
Finish all the work

7.2 Project plan

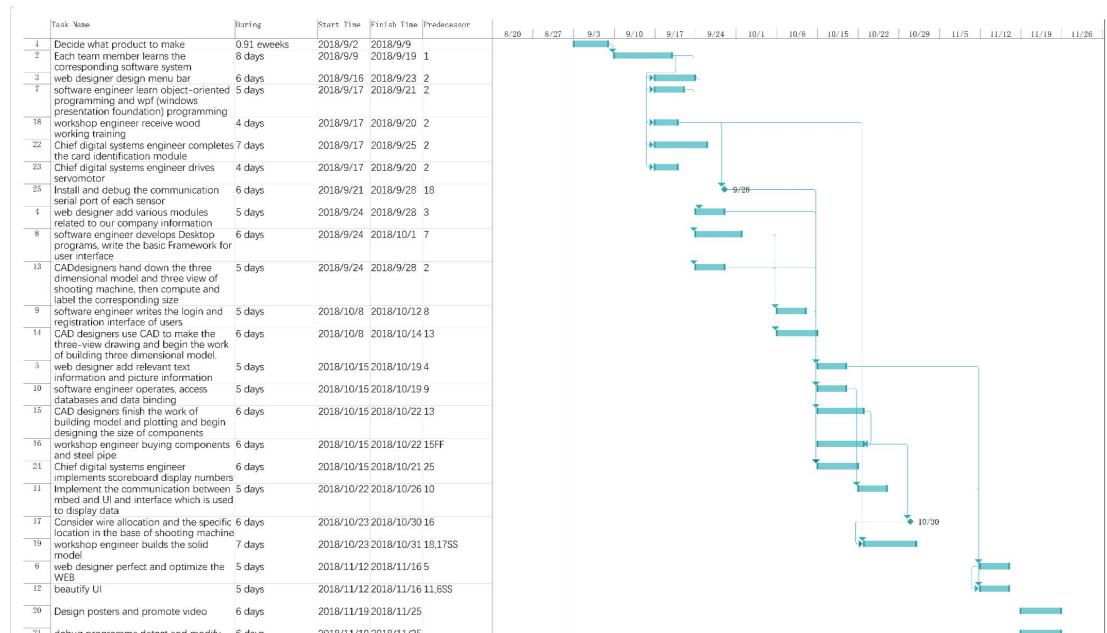


Figure 66 Initial Project Plan

7.3 project risk

Also, the risks in each step was evaluated and corresponding methods was designed. Then evaluate table will be placed in the appendix.

7.4 Team work

We initially assigned everyone's position at the first meeting.

Team members	Post
Haoxuan Du	Project Manager Electronic Engineer
Zeyu Xia	Chief Digital Systems Engineer
Ruoxi Gao	Chief Public Relations Software Engineer
Yuze Sun	Secretary Mechanical Design Engineer
Senpu Yuan	Web Designer Mechanical Assembly Engineer
Yutong Xu	Product Publicist Workshop Engineer

In the following team meeting, We have defined the specific tasks of each team member.

Team members	Work
Haoxuan Du	Making Project Plan Digital System Design Arranging Team Meeting FPGA Coding
Zeyu Xia	Digital System Design Sensor Coding Sensor Testing MBED Coding
Ruoxi Gao	User Interface Design Software Design PC Maintenance Mode
Yuze Sun	Team Meeting Submission Mechanical Structure Design Mechanical Design Drawing Workshop Engineer
Senpu Yuan	Web Page Design Material Purchase Workshop Engineer
Yutong Xu	Product Publicity Product Wiring

7.5 Meetings

To solve the problems generated during the project process timely and check whether the work had been done, project meetings were held every week. All the details, such as the status of every tasks, review of previous action, the problems and solutions, the plan for the next week would be covered during these meetings and recorded in meeting minutes. And the minutes are contained in appendix.

7.6 Experience and Lessons

1. There must be conflicts during a project. You should try to understand the thoughts of your teammates and to encourage other members when they are in trouble.
- 2.. The physical product may have a significant difference with your original design. Taking more factors into consideration and do not expect it can succeed without any faith.
3. The project plan is expected to be done in coordination with the entire group and everyone should totally express their suggestions
4. Clearly know the objective and requirements of a project. Before taking any action, you would better consult a lot of materials and references to get more information.

8.Appendix

8.1 meeting minutes

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	04
Week:	4		
Time:	20: 00-21: 00, 24 th September 2018		
Place:	'Jieke' coffee shop		
Present:	All		
Apologies:	None		
Agenda:	<ol style="list-style-type: none">1. Scheme of project2. Issues of communication3. Utility of motors and its location in mechanical architecture4. The detailed introduction of each part5. A brief introduction of the whole project6. The method of throwing the ball		

System Project Technology Report
Group 4

Review of previous actions:	1. There are some divergence about the way of shooting 2. Make sure the specific location of sensor 3. Have a clear idea about what material each part needs.
Main business:	1. Many project schemes were discussed and the most appropriated one, mini shooting machine, was chosen by voting 2. The appearance of the shooting machine was discussed and confirmed by every member 3. The materials required in the project were ensured 4. The exact functions of different blocks were discussed and members had a more explicit task in current stage before the next meeting 5. Some details were confirmed in the meeting such as the information about sensors and motors and their locations and utilities 6. Members discussed about items of devices which are needed in the near future
Action	1. The appearance as a mini shooting machine was agreed by all members, and mechanical architecture of appearance are to be drawn up by Chief Mechanical Engineer by using CAD 2. The basic frame of Web is to be completed by Web designer 3.. The operate mode of user interface is to be designed by Software engineer 4. The details about how to complete the communication between mbed and PC are to be discussed by Software engineer and Chief electronics and wiring engineer 5. The explicit method of controlling motors using FPGA is to be confirmed by Chief digital systems engineer 6. The details about how to achieve the functions of every blocks by using different technologies need to be discussed in the next meeting

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	04
Week:	4		
Time:	16: 00-17: 30, 27 th September 2018		
Place:	'Jieke' coffee shop		
Present:	All		
Apologies:	None		

System Project Technology Report
Group 4

Agenda:	<ol style="list-style-type: none"> 1. Scheme of project 2. Issues of communication 3. Utility of motors and its location in mechanical architecture 4. The detailed introduction of each part 5. The method of throwing the ball
Review of previous actions:	<ol style="list-style-type: none"> 1. There are some divergence about the way of shooting. 2. Make sure the specific location of sensor. 3. Have a clear idea about what material each part needs.
Main business:	<ol style="list-style-type: none"> 1. Discuss the way of throwing the ball, and determine to use a catapult to throw the ball into basketry. 2. Think about the pattern of moving basketry for the purpose of increasing difficulty. 3. Discuss whether to stick to this plan or think about the other projects. 4. Find some DIY video on the internet.
Action	<ol style="list-style-type: none"> 1. The mechanical architecture is agreed to be modify, changing the way of throwing. This work will be done by Chief Mechanical Engineer with CAD. 2. Workshop engineers go to construction material market to find the appropriate materials for each part of the machine. 3. Whether to insist on this machine or choose another infrastructure such as a maze game should be discussed in the next meeting. 4. Team members write business plan together and have a complete book in the early October.
	5.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	4
Week:	7		
Time:	19:00-21:10, 8 rd October 2018		
Place:	Library seminar 102 room		
Present:	All		
Apologies:	None		

System Project Technology Report
Group 4

Agenda:	<ul style="list-style-type: none"> 1、 Discuss the contents what is controversial and puzzled during the process. <ul style="list-style-type: none"> (1) The planning function of FPGA board . (2) The process of maintenance model (click bottom-insert card-start simulation-present the state). (3) We still don't decide the position of identical sensor which will determined by the relevant position with lever in the middle term. (4) Number of the rim is one. (5) The materials of lever will be chosen between the wood and hard paper. (6) We still don't decide the usage of the color sensor. (7) We still don't decide whether to use the second motor. 2、 Discuss the arrangement in the following week.(details in Action part)
Review of previous actions:	None
Main business:	<ul style="list-style-type: none"> 1、 Everyone will finish one part of the Business plan. We are supposed to finish the whole part of the planning before we have a meeting with Party A. 2、 Next meeting time approved
Action	<ul style="list-style-type: none"> 1、 The first version pictures through AutoCAD will finish before next meeting. 2、 The communication between Mbed and C# will finish before next meeting. 3、 On the Saturday of this weekend. Our group members will go the market to select some tools and materials. 4、 The Publicizing network is supposed to finish between week 8.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	4
Week:	7		
Time:	19:00-21:10, 17th October 2018		
Place:	Library seminar 102 room		
Present:	All		

System Project Technology Report
Group 4

Apologies:	None
Agenda:	<ul style="list-style-type: none"> • New project • Application of the color sensor • Function of machine • Automation of machines
Review of previous actions:	None
Main business:	<ul style="list-style-type: none"> • Discuss the feasibility of new projects. • Discuss the use of the color sensors. • Think about mechanical mechanism briefly. • Everyone goes back to thinking how to improve the project. • Working out the next meeting time.
Action	<ul style="list-style-type: none"> • Complete c# and mbed. • Modify CAD.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	04
Week:	Week 8		
Time:	18:00-21:00, 20 th October 2018		
Place:	A shop in school named JieKe.		
Present:	All team members.		
Apologies:	None		
Agenda:	<ol style="list-style-type: none"> 1. Scheme of project 2. Mechanical design 3. Application of sensors 4. Discussion on mechanical drawing 		

System Project Technology Report
Group 4

Review of previous actions:	<ol style="list-style-type: none"> 1. Make sure that mbed,PC and FPGA can keep normal communication. 2. Confirm the use of color sensor. 3. Discussion of the specific mechanical structure and mechanical principle. 4. The deadline of certain tasks was confirm exactly.
Main business:	<ol style="list-style-type: none"> 1. Discuss the new project plan after the meeting with First party. 2. According to the new requirements put forward by First party,we talk about the intelligence of our product. 3. Continue to discuss the specific use of different sensors and motors.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	4
Week:	9		
Time:	12:00-12:40, 1 st November 2018		
Place:	Library seminar 102 room		
Present:	All		
Apologies:	None		
Agenda:	<ul style="list-style-type: none"> • The size of the board, how to connect the boards together. • C# progress, detailed computer interface design steps. • Song categories under different emotions. • Determine the answer pattern and reward system. • Poster designer's progress. • Display the concrete results of web design. • The establishment of shipping mechanism. 		
Review of previous actions:	None		

System Project Technology Report
Group 4

Main business:	<ul style="list-style-type: none"> Determine the use of five mm thick wood panels, using small triangular scaffolds to fix adjacent boards. The specific process of interaction between players at the computer interface is determined. The classification of songs under different emotions is divided into two broad categories, namely happiness and depression. According to the player who answers the song he heard, decide whether to pass the checkpoint. If customs clearance is available, the machine will reward players with a mood note and a candy. If customs do not pass the test, the player will get a mood note. The poster has been completed and the result is very good. The database is basically completed, waiting for specific padding. By discussing and determining the mode of delivery using the turntable mode, the right goods are distributed to the player through sorting.
Action	<ul style="list-style-type: none"> Everyone reported the progress of his current work. The poster was checked. Established the way of shipment. There are three different styles of songs. So there are six different classifications. Three team members each selected twenty corresponding songs. The database is being built, waiting for the song to finish filling the database.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	04
Week:	10		
Time:	15:30-16:30, 8 th November 2018		
Place:	Library seminar 102 room		
Present:	All		
Apologies:	None		
Agenda:	1. Combination of mechanical system and electronic system 2. Implementation of information fusion from different sensors on PC 3. Solution for conflict between voice prompts, speech input and music playing 4. Division of function between mbed and FPGA 5. Confirmation of animation form on PC 6. Supplement of machining tools 7. Confirmation of appearance of video		

System Project Technology Report
Group 4

Review of previous actions:	<ol style="list-style-type: none"> 1. The categories of songs were confirmed exactly 2. The database which store information of songs was checked 3. The function implementation of button was tested 4. The design of Web page and poster was modified
Main business:	<ol style="list-style-type: none"> 1. Many solutions for combination of mechanical system and electronic system were discussed in order to implement accuracy of control on mechanical architecture from electronic device, the final solution was agreed by all members 2. The detailed process of voice prompts, speech input and music playing were discussed and the only proper solution was ensured 3. The explicit algorithm for achieving integrated function of reading information which come from many types of sensors on PC was confirmed 4. The method for how to associate songs with database and how to play music required by selecting from database was discussed and ensured 5. Some details were confirmed in the meeting such as function division of two hardware modules for the sake of simplification of processing and debugging 6. Members discussed about types of animation and chose one which are easy to realize and attractive for uses 7. Members discussed about gadgets which are needed when processing 8. The overall style and keynote of video was confirmed
Action	<ol style="list-style-type: none"> 1. The explicit solutions about technology implementation for connect mechanical system with electronic system are to be confirmed by Chief Mechanical Engineer and Chief digital systems engineer 2. The explicit function implementation about communication, animation and voice module of GUI is to be completed by Software engineer 3. The details about how to simplify the code quantity and complexity are to be discussed by Chief digital systems engineer and Chief electronics and wiring engineer 4. The Web pages are to be beautified by Web designer 5. The details about how to adjust parameters to implement accurate control on mechanical module need to be discussed in the next meeting

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	04
Week:	09		
Time:	10:30-11:30, 14 th November 2018		
Place:	E-III-104		
Present:	All		

System Project Technology Report
Group 4

Apologies:	None
Agenda:	<ol style="list-style-type: none"> 1. Further combination of mechanical system and electronic system 2. Details of assembling process 3. Optimization of animation form on PC 4. Discussion about rudiment of sorter 5. Discussion about video recording
Review of previous actions:	None
Main business:	<ol style="list-style-type: none"> 1. Detailed discussion about how to control mechanical system accurately, using electronic devices, by convenient methods 2. Agreement on details about how to connect mechanical modules 3. Formulation of animation on PC based on old version 4. Discuss the algorithm of motor rotation 5. Discuss details about how to shoot the video, for example, accurate duration of different contents
Action	<ol style="list-style-type: none"> 1. The explicit solutions about technology implementation for controlling sorter are to be confirmed by Chief Mechanical Engineer and Chief digital systems engineer 2. The function implementation about communication with mbed when starting game and in maintenance mode and explicit animation of GUI is to be completed by Software engineer 3. The Poster are to be beautified by Chief pr 4. The details about how to modify codes to implement proper communication between PC and mbed in the whole game process need to be discussed in the next meeting

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	4
Week:	11		
Time:	16:00-18:00 , 27th November 2018		
Place:	Library seminar 102 room		
Present:	All		
Apologies:	None		

Agenda:	<ul style="list-style-type: none"> • Decide decoration style of our mechanism. • Debug the system code • Relocate the dislocation of the electric line • Design the poster • Redecorate the GUI and debug the C# code.
Review of previous actions:	None
Main business:	<ul style="list-style-type: none"> • Monitor the running state of motor • Monitor whether the sensor is working properly • Requirements for poster and video design • Debugging Interface Program Code • Solving the legacy problems of mbed • Appearance of decorative products
Action	<ul style="list-style-type: none"> • Continuously adjust the position of the motor to find the best reasonable position. • Beautify GUI interface and debug bugs. Avoid stuck by mistake. • Wiring inside the machine, reasonable planning of space. • Complete the task of product description. • Reasonable layout of parts inside the machine. • Complete the post-processing of video.

Meeting Minutes			
Group Name:	Blueprinted	Group Number:	4
Week:	11		
Time:	16:00-18:00 , 27th November 2018		
Place:	Library seminar 102 room		
Present:	All		
Apologies:	None		

Agenda:	<ul style="list-style-type: none"> • Decide decoration style of our mechanism. • Debug the system code • Relocate the dislocation of the electric line • Design the poster • Redecorate the GUI and debug the C# code.
Review of previous actions:	None
Main business:	<ul style="list-style-type: none"> • Monitor the running state of motor • Monitor whether the sensor is working properly • Requirements for poster and video design • Debugging Interface Program Code • Solving the legacy problems of mbed • Appearance of decorative products
Action	<ul style="list-style-type: none"> • Continuously adjust the position of the motor to find the best reasonable position. • Beautify GUI interface and debug bugs. Avoid stuck by mistake. • Wiring inside the machine, reasonable planning of space. • Complete the task of product description. • Reasonable layout of parts inside the machine. • Complete the post-processing of video.

8.2 Risk management

Project Name:shooting machine	Prepared by Team:group 04	Date:2018/9/20
Course Code:B39VS	Contact Name:Yuze Sun	Customer/End User group:
Year of study/Discipline:year 3	Project Manager:Haoxuan Du	Project Sponsor:

WBS Number	Risk Event	Risk Type	Probability	Impact	Risk Response	Risk Owner

System Project Technology Report
Group 4

1	Disagreement between team members	Organizational Risk	30%	Lead to inefficiency of decision-making process and problems on relationship between members as well	The project manager need to coordinate the relationship between members and make correct decisions Every team member need to respect others' opinions	All members
1	Difficult time coordination to finish certain task between team members	Organizational Risk	20%	Lead to inefficiency of functional implementation or even have to postpone the project's scheduling	Every team member need to consider other members' schedule and make plan based on it Manager should make regulations to avoid these situations	All members
2	Requirement of position changes or dimission from team members	Organizational Risk	10%	Disturb the division of the project and affect original learning and completing plan of some members	Team members should choose proper job position Manager should keep contact with every member to keep troubles from spreading	Management director

System Project Technology Report
Group 4

2	Injury during the process of completing mechanical structure	Health and Safety Risk	10%	Delay project progress and cause job position vacancies	Make sure that WE have been trained well before working in the workshop Each member should work with at least one supervisory person	Workshop engineer
3	Inaccurate measurement of mechanical components sizes	Technical Risk	15%	Affect the function of mechanical design	The size need be measured more times The data need to be checked by at least one member	Mechanical designer & Workshop engineer
6	Complicated mechanical which is excessively time-consuming	Technical Risk	10%	Affect the schedule of project and may cause delay of other tasks	The mechanical design should be designed carefully and members need to check it after completing the design Manager need to supervise the mechanical process in time	Mechanical designer

System Project Technology Report
Group 4

6	The components made by hand cannot meet anticipation	Technical Risk	15%	Disturb the progress of project and may cause the faultiness of some functions	ME should be required to simplify the mechanical structure before designing Make a second mechanical plan which is simple to accomplish	Workshop engineer
7	Failure of grasping required technologies	Technical Risk	40%	Disturb the progress of project	Manager should make learning schedule and check the performance every week Assign more than one member on a job position	All members
10	The electronic device can't match the mechanical components	Technical Risk	25%	Cause the faultiness of some functions	ME should coordinate with EW and DE in time when designing mechanical structure The mechanical design need to be checked by every member before manufacturing	Mechanical designer & Workshop engineer & Electronics and wiring engineer & Digital systems engineer

System Project Technology Report
Group 4

2	Potential bugs on communication between PC and mbed,mbed and FPGA	Technical Risk	10%	Lead to the imperfection of project and affect the schedule of project	DE,EWand SE should keep contact with each other and test every block when completing certain task to keep problems from accumulating DE and SE should test programme more times before coming into use Manager need to supervise and check communication function on time	Electronics and wiring engineer & Digital systems engineer & Software engineer
10	Failed database retrieval and data management	Technical Risk	5%	Lead to the imperfection of project and affect the schedule of project	SE should test the utility of database every time new items are added to the database Manager should check this block on time	Software engineer
10	Latent bugs in mbed,FPGA or GUI programme	Technical Risk	15%	Lead to the imperfection of project and affect the schedule of project	Programme need to be check more than one time by SE,EW and DE Team member should negotiate fixed time to check programme every week	Electronics and wiring engineer & Digital systems engineer & Software engineer

System Project Technology Report
Group 4

9	Change of project alternatives in the middle of project because of problems on technologies	Project Management Risk	20%	Result in waste of time and may cause potential disharmony between team members	Team member should discuss about every detail during meeting Manager need to make decision to modify something impractical on time	All members
9	Wrong operation or damage for mbed/FPGA/ sensors/motors	Operational Risk	Less than 5%	Affect the schedule of project and result in waste of fund	1.EW and DE need to read instructions of these devices in detail and be careful when using devices	Electronics and wiring engineer & Digital systems engineer
3	Misleading propagand on web pages	Credit Risk	Less than 5%	Cause problems during exhibition	Manager should supervise the content on the web pages to bring it correspondence with fact during the process of designing it	

System Project Technology Report
Group 4

11	The project cannot be completed on time	Project Management Risk	5%	Cannot submit the finished product in time	Manager should make working schedule and check the performance every week Every member should make proper plan give an account of job schedule to manager in time	
11	The product can't appeal users and meet their requirements	Project Management Risk	10%	Fail the bidding of project	1.Manager need to communicate with customers in time and give members feed back on time	
10	Bad physical and mental condition of members during the process of project	Health and Safety Risk	Less than 5%	Affect the schedule of project and increase the time cost	1.Members make preparation such as first-aid medicines if needed 2.Manager care about mental situation of members and give some encouragement and guidance if necessary	

System Project Technology Report
Group 4

12	Natural disaster, harsh climate and other force majeure	External Risk	Less than 5%	Hinder the process of the project	1. Members can select a standby working and meeting station if needed	
----	---	---------------	--------------	-----------------------------------	---	--

8.3 Authors of this document

Session	Authors
Introduction	Yutong XU
Mechanical Design	Zeyu Sun
Digital System Design	Haoxuan Du
Software Design	Zeyu Xia/ Ruoxi Gao
Sensor system	Zeyu Xia
Market and Web	Senpu Yuan
Project Plan	Haoxuan Du