/\* //// DRAGON AWAKEN //// \*/

/\* // MAIN COMPONENTS // \*/

// Lightsaber toy or led set (x1)

// ARDUINO UNO (x1)

// VS1053b MP3 Shield for ARDUINO (x1)

// Micro Servo motor SG90 or compatible with wiring (x1)

// HC - SR04 Ultrasonic Distance Measuring Sensor Module for Arduino (x1)

// SD card (x1)

// externally powered speakers (x2) [avoid powering from the Arduino, this might cause malfunction of MP3 Shield].

// regular LED (x6)

// ARDUINO jumper cables and alligator cables (>20+)

// USB cable (x1)

// PC/MAC (x1)

/\* /// For configuration please see wire diagram and inside-lighsaber pictures. Thank you /// \*/

/\* servo+led+lightsaber+sensor+random+MP3 \*/

/\* START OF MP3 Shield configuration MP3TF \*/

#include <SPI.h>

//Libraries

#include <SdFat.h>

#include <SdFatUtil.h>

//and the MP3 Shield Library

#include <SFEMP3Shield.h>

//create and name the library object

SFEMP3Shield MP3player;

SdFat sd;

SdFile file;

// Define variables that we might use but in this specific example are not needed.

// See https://github.com/mpflaga/Sparkfun-MP3-Player-Shield-Arduino-Library

byte temp;

byte result;

/\* END OF MP3 Shield configuration MP3TF \*/

#include <Servo.h>

#include <NewPing.h>

#define TRIGGER\_PIN 10

#define ECHO\_PIN 9

#define MAX\_DISTANCE 300

// Create servo Motor object

Servo myservo;

NewPing sonar(TRIGGER\_PIN, ECHO\_PIN, MAX\_DISTANCE);

// Define random variable to determine if an individual has the force

long randNumber;

int timer =50;

int sound = 250;

void setup() {

Serial.begin(115200);

pinMode(TRIGGER\_PIN, OUTPUT);

pinMode(ECHO\_PIN, INPUT);

myservo.attach(11);

// if analog input pin 0 is unconnected, random analog

// noise will cause the call to randomSeed() to generate

// different seed numbers each time the sketch runs.

// randomSeed() will then shuffle the random function.

randomSeed(analogRead(0));

//mp3 shield powering in varibles initializing

sd.begin(SD\_SEL, SPI\_HALF\_SPEED);

//boot up the MP3 Player Shield

result = MP3player.begin();

MP3player.setVolume(10, 10);

}

void loop() {

delay(1000);

int uS = sonar.ping();

long distance,pos=0,i;

long duration;

Serial.print("Ping: ");

distance =uS / US\_ROUNDTRIP\_CM;

Serial.print(distance);

Serial.println("cm");

Serial.println("Hello");

Serial.println("Preparing to play a song");

int randNumber = 1;

Serial.println(randNumber);

//

Serial.print(distance);

Serial.println(" cm");

//the person has to be within the sensors range and closer than 10cm and farther than 30cm to the installation

if(distance > 10 && distance<=30){

//It is assumed that only 20% has the force

Serial.println("Dragon sound");

//add track for person with the force

char trackName[] = "track001.mp3";

//tell the MP3 Shield to play that file

result = MP3player.playMP3(trackName);

//check result, see readme for error codes.

if(result != 0) {

Serial.print("Error code: ");

Serial.print(result);

Serial.println(" when trying to play track");

}

//darth vader turns to look at you if you have the force

myservo.write(180);

for (int thisPin = 2; thisPin < 8; thisPin++) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// loop from the highest pin to the lowest:

for (int thisPin = 7; thisPin >= 2; thisPin--) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// wait 1.5 seconds

delay(1500);

myservo.write(0);

}

//the person has to be within the sensors range and closer than 30cm and farther than 60cm to the installation

if(distance > 30 && distance<=60){

if (randNumber==1) {

Serial.println("Dragon sound");

//add track for person with the force

char trackName[] = "track002.mp3";

//tell the MP3 Shield to play that file

result = MP3player.playMP3(trackName);

//check result, see readme for error codes.

if(result != 0) {

Serial.print("Error code: ");

Serial.print(result);

Serial.println(" when trying to play track");

}

//darth vader turns to look at you if you have the force

myservo.write(180);

for (int thisPin = 2; thisPin < 8; thisPin++) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// loop from the highest pin to the lowest:

for (int thisPin = 7; thisPin >= 2; thisPin--) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// wait 1.5 seconds

delay(1500);

myservo.write(0);

}

}

//the person has to be within the sensors range and closer than 60cm and farther than 100cm to the installation

if(distance > 60 && distance<=100){

if (randNumber==1) {

Serial.println("Dragon sound");

//add track for person with the force

char trackName[] = "track003.mp3";

//tell the MP3 Shield to play that file

result = MP3player.playMP3(trackName);

//check result, see readme for error codes.

if(result != 0) {

Serial.print("Error code: ");

Serial.print(result);

Serial.println(" when trying to play track");

}

//darth vader turns to look at you if you have the force

myservo.write(180);

for (int thisPin = 2; thisPin < 8; thisPin++) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// loop from the highest pin to the lowest:

for (int thisPin = 7; thisPin >= 2; thisPin--) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// wait 1.5 seconds

delay(1500);

myservo.write(0);

}

}

//the person has to be within the sensors range and closer than 100cm and farther than 150cm to the installation

if(distance > 100 && distance<=150){

if (randNumber==1) {

Serial.println("Dragon sound");

//add track for person with the force

char trackName[] = "track004.mp3";

//tell the MP3 Shield to play that file

result = MP3player.playMP3(trackName);

//check result, see readme for error codes.

if(result != 0) {

Serial.print("Error code: ");

Serial.print(result);

Serial.println(" when trying to play track");

}

//darth vader turns to look at you if you have the force

myservo.write(180);

for (int thisPin = 2; thisPin < 8; thisPin++) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// loop from the highest pin to the lowest:

for (int thisPin = 7; thisPin >= 2; thisPin--) {

// turn the pin on:

digitalWrite(thisPin, HIGH);

delay(timer);

// turn the pin off:

digitalWrite(thisPin, LOW);

}

// wait 1.5 seconds

delay(1500);

myservo.write(0);

}

}

else{

Serial.print(distance);

char trackName[] = "track003.mp3";

if(result != 0) {

Serial.print("Error code: ");

Serial.print(result);

Serial.println(" when trying to play track");

}

//tell the MP3 Shield to play that file

result = MP3player.playMP3(trackName);

Serial.println("Out of range");

delay(3000);

}

}