#include <Keypad.h>

#include <DFPlayer\_Mini\_Mp3.h>

SoftwareSerial mySerial(10, 11); // RX, TX

const byte ROWS = 3; //four rows

const byte COLS = 2; //four columns

//define the cymbols on the buttons of the keypads

char hexaKeys[ROWS][COLS] = {

{'1','4'},

{'2','5'},

{'3','6'}

};

byte rowPins[ROWS] = {2,3,4}; //connect to the row pinouts of the keypad

byte colPins[COLS] = {6,5}; //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad

Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

String brailleChar="000000";

long sequence=0;

const int selectionPin=8;

const int enter=14;

const int busyPin=12;

const int prev=17;

const int playStop=16;

const int next=15;

void setup() {

Serial.begin(9600);

mySerial.begin(9600);

mp3\_set\_serial (mySerial); //set softwareSerial for DFPlayer

delay(1000);

mp3\_reset(); //soft-Reset module DFPlayer

delay(1000); //wait 1ms for respon command

mp3\_set\_volume (30); //set Volume module DFPlayer

delay(1000);

playTrack(0);

pinMode(selectionPin,INPUT\_PULLUP);

pinMode(enter,INPUT\_PULLUP);

pinMode(prev,INPUT\_PULLUP);

pinMode(playStop,INPUT\_PULLUP);

pinMode(next,INPUT\_PULLUP);

pinMode(busyPin,INPUT\_PULLUP);

}

void loop() {

while(digitalRead(selectionPin)==HIGH){

findSquence();

Serial.print("hi");

}

while(digitalRead(selectionPin)==LOW){

Serial.print("hiiiiiiiiiiiii");

tracksPlay();

}

// delay(1000);

}

void playTrack(unsigned int i){

mp3\_play(i);//now the time is

while(digitalRead(busyPin)==HIGH){;}

while(digitalRead(busyPin)==LOW){;}

delay(50);

}

void findSquence(){

while(digitalRead(enter)==HIGH && digitalRead(selectionPin)==HIGH){

int customKey = customKeypad.getKey();

if(customKey>48){

// Serial.println((customKey-48));

brailleChar[(customKey-49)]='1';

// Serial.println(brailleChar);

}

}

// Serial.println(brailleChar);

checkSequence();

}

void checkSequence(){

long sequence=brailleChar.toInt();

switch (sequence) {

case 100000:{

Serial.println("A");

playTrack(1);

brailleChar="000000";

//do something when var equals 1

break;

}

case 110000:{

Serial.println("B");

playTrack(2);

brailleChar="000000";

//do something when var equals 2

break;}

case 100100:{

Serial.println("C");

playTrack(3);

brailleChar="000000";

//do something when var equals 2

break;}

case 100110:{

Serial.println("D");

playTrack(4);

brailleChar="000000";

//do something when var equals 1

break;

}

case 100010:{

Serial.println("E");

playTrack(5);

brailleChar="000000";

//do something when var equals 2

break;}

case 110100:{

Serial.println("F");

playTrack(6);

brailleChar="000000";

//do something when var equals 1

break;

}

case 110110:{

Serial.println("G");

playTrack(7);

brailleChar="000000";

//do something when var equals 2

break;}

case 110010:{

Serial.println("H");

playTrack(8);

brailleChar="000000";

//do something when var equals 1

break;

}

case 10100:{

Serial.println("I");

playTrack(9);

brailleChar="000000";

//do something when var equals 2

break;}

case 10110:{

Serial.println("J");

playTrack(10);

brailleChar="000000";

//do something when var equals 1

break;

}

case 101000:{

Serial.println("K");

playTrack(11);

brailleChar="000000";

//do something when var equals 2

break;}

case 111000:{

Serial.println("L");

playTrack(12);

brailleChar="000000";

//do something when var equals 1

break;

}

case 101100:{

Serial.println("M");

playTrack(13);

brailleChar="000000";

//do something when var equals 2

break;}

case 101110:{

Serial.println("N");

playTrack(14);

brailleChar="000000";

//do something when var equals 1

break;

}

case 101010:{

Serial.println("O");

playTrack(15);

brailleChar="000000";

//do something when var equals 2

break;}

case 111100:{

Serial.println("P");

playTrack(16);

brailleChar="000000";

//do something when var equals 1

break;

}

case 111110:{

Serial.println("Q");

playTrack(17);

brailleChar="000000";

//do something when var equals 2

break;}

case 111010:{

Serial.println("R");

playTrack(18);

brailleChar="000000";

//do something when var equals 1

break;

}

case 11100:{

Serial.println("S");

playTrack(19);

brailleChar="000000";

//do something when var equals 2

break;}

case 11110:{

Serial.println("T");

playTrack(20);

brailleChar="000000";

//do something when var equals 1

break;

}

case 101001:{

Serial.println("U");

playTrack(21);

brailleChar="000000";

//do something when var equals 2

break;}

case 111001:{

Serial.println("V");

playTrack(22);

brailleChar="000000";

//do something when var equals 1

break;

}

case 10111:{

Serial.println("W");

playTrack(23);

brailleChar="000000";

//do something when var equals 2

break;}

case 101101:{

Serial.println("X");

playTrack(24);

brailleChar="000000";

//do something when var equals 1

break;

}

case 101111:{

Serial.println("Y");

playTrack(25);

brailleChar="000000";

//do something when var equals 2

break;}

case 101011:{

Serial.println("Z");

playTrack(26);

brailleChar="000000";

//do something when var equals 1

break;

}

//////////////////////////////////////////////////////////////////

case 10000:{

Serial.println("1");

playTrack(28);

brailleChar="000000";

//do something when var equals 2

break;}

case 11000:{

Serial.println("2");

playTrack(29);

brailleChar="000000";

//do something when var equals 1

break;

}

case 10010:{

Serial.println("3");

playTrack(30);

brailleChar="000000";

//do something when var equals 2

break;}

case 10011:{

Serial.println("4");

playTrack(31);

brailleChar="000000";

//do something when var equals 1

break;

}

case 10001:{

Serial.println("5");

playTrack(32);

brailleChar="000000";

//do something when var equals 2

break;}

case 11010:{

Serial.println("6");

playTrack(33);

brailleChar="000000";

//do something when var equals 1

break;

}

case 11011:{

Serial.println("7");

playTrack(34);

brailleChar="000000";

//do something when var equals 2

break;}

case 11001:{

Serial.println("8");

playTrack(35);

brailleChar="000000";

//do something when var equals 1

break;

}

case 1010:{

Serial.println("9");

playTrack(36);

brailleChar="000000";

//do something when var equals 2

break;}

case 1011:{

Serial.println("0");

playTrack(27);

brailleChar="000000";

//do something when var equals 1

break;

}

default:

playTrack(74);

brailleChar="000000";

// if nothing else matches, do the default

// default is optional

break;

}

}

void tracksPlay(){

int i=36;

if(digitalRead(busyPin)==HIGH){

mp3\_play(++i);

delay(50);

}

else if(digitalRead(busyPin)==LOW){;}

while(digitalRead(selectionPin)==LOW){

if(digitalRead(playStop)==LOW){

mp3\_stop();

delay(50);

delay(250);

}

else if(digitalRead(prev)==LOW){

if((--i)<37){

i=73;

mp3\_stop();

delay(50);

mp3\_play (i);

}

else{

mp3\_stop();

delay(50);

mp3\_play (i);

}

delay(250);

}

else if(digitalRead(next)==LOW){

if((++i)>73){

i=37;

mp3\_stop();

delay(50);

mp3\_play (i);

}

else{

mp3\_stop();

delay(50);

mp3\_play(i);

}

delay(250);

}

}

delay(250);

}