#include "LedControl.h"

/\* pin 12 is connected to the DataIn pin 11 is connected to the CLK pin 10 is connected to LOAD \*/

**LedControl** lc=**LedControl**(12,11,10,5);

int lightsonC=0;

int lightsonS=0;

int arrows=0;

int arrowt=50;

int CylonON=0;

int CylonOFF=-3;

int CylonSON=0;

int CylonSOFF=0;

unsigned long beatDelaytime=40;

int beatC=0;

int beatS=0;

int helloT=300;

int helloS=0;

int helloR=0;

int LazerON=0;

int LazerOFF=-2;

int LazerSON=0;

int LazerSOFF=0;

unsigned long OKdelaytime=200;

int OKc=0;

int OKs=1;

int OKt=0;

int num=0;

int rown=2;

int firsttime=1;

int intense = 0;

int times;

int Matrixn;

int colN;

int rowN;

int randomN;

int tbON=0;

int tbOFF=-3;

int tbS=0;

int waveC[40]={0,1,2,3,4,5,6,7,0,1,2,3,4,5,6,7,0,1,2,3,4,5,6,7,0,1,2,3,4,5,6,7,0,1,2,3,4,5,6,7};

int waveS[40]={0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3,4,4,4,4,4,4,4,4};

int waveR[40]={0,1,2,3,4,5,6,7,7,6,5,4,3,2,1,0,0,1,2,3,4,5,6,7,7,6,5,4,3,2,1,0,0,1,2,3,4,5,6,7};

int waveK[40]={0};

int wavenum=0;

int wavereturn[40]={0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0};

int musicR[32]={7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7,7};

int musicC[32]={1,2,3,4,6,7,0,1,3,4,5,6,0,1,2,3,5,6,7,0,2,3,4,5,7,0,1,2,4,5,6,7};

int musicS[32]={0,0,0,0,0,0,1,1,1,1,1,1,2,2,2,2,2,2,2,3,3,3,3,3,3,4,4,4,4,4,4,4};

int musicbarnum=0;

int randmusic;

int musicBarcounter=0;

int dotsR=2;

int rainS=0;

int rainR[40]={0,0,0,0,0,-1,-1,-1,-1,-1,-2,-2,-2,-2,-2,-3,-3,-3,-3,-3,-4,-4,-4,-4,-4,-5,-5,-5,-5,-5,-6,-6,-6,-6,-6,-7,-7,-7,-7,-7};

int rainON[40];

int rainnum=0;

int fileR=0;

int fileS=0;

int filenum=0;

volatile int mode=1;

int playlist=1;

int mNum=0;

int timenum=0;

int change=0;

static unsigned long last\_interrupt\_time = 0;

unsigned long interrupt\_time = millis();

unsigned long time = millis();

unsigned long lasttime=0;

int Clearfirsttime=1;

void setup()

{

 int devices=lc.getDeviceCount();

 for(int address=0;address<devices;address++)

 {

   lc.shutdown(address,false);

   lc.setIntensity(address,3);

   lc.clearDisplay(address);

   attachInterrupt(0, add, HIGH);

 }

 randomSeed(analogRead(5));

 delay(100);

}

void Clear()

{

 if (Clearfirsttime==1)

 {

   while (mNum<=4)

       {

        lc.clearDisplay(mNum);

        lc.setIntensity(mNum,3);

        mNum++;

      }

      mNum=0;

 }

}

void changecheck()

{//for future improvements

}

void delaycheck()

{

 delay(100);

 timenum++;

 changecheck();

}

void add()

{

 interrupt\_time=millis();

 change=1;

  if (interrupt\_time - last\_interrupt\_time > 500)

  {

    Clearfirsttime=1;

    firsttime=1;

    mNum=0;

    playlist=1;

    if (mode<18)

    {

      mode++;

    }

    else

    {

      mode=1;

    }

  }

  last\_interrupt\_time = interrupt\_time;

  time=millis();

  lasttime=time;

}

void AllLightsOn()

{

 lightsonC=0;

 lightsonS=0;

 while (lightsonS<=4)

 {

   while (lightsonC<=7)

   {

     lc.setColumn(lightsonS,lightsonC,B11111111);

     lightsonC++;

     delay(20);

     changecheck();

   }

   lightsonC=0;

   lightsonS++;

 }

 while (timenum<=12)

 {

   delaycheck();

 }

 timenum=0;

 while (lightsonS>=0)

 {

   while (lightsonC>=0)

   {

     lc.setColumn(lightsonS,lightsonC,B00000000);

     lightsonC--;

     delay(20);

     changecheck();

   }

   lightsonC=7;

   lightsonS--;

 }

 while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

 lightsonC=7;

 lightsonS=4;

 while (lightsonS>=0)

 {

   while (lightsonC>=0)

   {

     lc.setColumn(lightsonS,lightsonC,B11111111);

     lightsonC--;

     delay(20);

     changecheck();

   }

   lightsonC=7;

   lightsonS--;

 }

 while (timenum<=12)

 {

   delaycheck();

 }

 timenum=0;

 while (lightsonS<=4)

 {

   while (lightsonC<=7)

   {

     lc.setColumn(lightsonS,lightsonC,B00000000);

     lightsonC++;

     delay(20);

     changecheck();

   }

   lightsonC=0;

   lightsonS++;

 }

 while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

}

void Arrows ()

{

 while(arrows<=4)

 {

   lc.setRow(arrows,0,B00110000);

   lc.setRow(arrows,1,B01100000);

   lc.setRow(arrows,2,B11000000);

   lc.setRow(arrows,3,B10000001);

   lc.setRow(arrows,4,B10000001);

   lc.setRow(arrows,5,B11000000);

   lc.setRow(arrows,6,B01100000);

   lc.setRow(arrows,7,B00110000);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B01100000);

   lc.setRow(arrows,1,B11000000);

   lc.setRow(arrows,2,B10000001);

   lc.setRow(arrows,3,B00000011);

   lc.setRow(arrows,4,B00000011);

   lc.setRow(arrows,5,B10000001);

   lc.setRow(arrows,6,B11000000);

   lc.setRow(arrows,7,B01100000);

   arrows++;

 }

 arrows=0;

 delay(arrowt);

 changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B11000000);

   lc.setRow(arrows,1,B10000001);

   lc.setRow(arrows,2,B00000011);

   lc.setRow(arrows,3,B00000110);

   lc.setRow(arrows,4,B00000110);

   lc.setRow(arrows,5,B00000011);

   lc.setRow(arrows,6,B10000001);

   lc.setRow(arrows,7,B11000000);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B10000001);

   lc.setRow(arrows,1,B00000011);

   lc.setRow(arrows,2,B00000110);

   lc.setRow(arrows,3,B00001100);

   lc.setRow(arrows,4,B00001100);

   lc.setRow(arrows,5,B00000110);

   lc.setRow(arrows,6,B00000011);

   lc.setRow(arrows,7,B10000001);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B00000011);

   lc.setRow(arrows,1,B00000110);

   lc.setRow(arrows,2,B00001100);

   lc.setRow(arrows,3,B00011000);

   lc.setRow(arrows,4,B00011000);

   lc.setRow(arrows,5,B00001100);

   lc.setRow(arrows,6,B00000110);

   lc.setRow(arrows,7,B00000011);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B00000110);

   lc.setRow(arrows,1,B00001100);

   lc.setRow(arrows,2,B00011000);

   lc.setRow(arrows,3,B00110000);

   lc.setRow(arrows,4,B00110000);

   lc.setRow(arrows,5,B00011000);

   lc.setRow(arrows,6,B00001100);

   lc.setRow(arrows,7,B00000110);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B00001100);

   lc.setRow(arrows,1,B00011000);

   lc.setRow(arrows,2,B00110000);

   lc.setRow(arrows,3,B01100000);

   lc.setRow(arrows,4,B01100000);

   lc.setRow(arrows,5,B00110000);

   lc.setRow(arrows,6,B00011000);

   lc.setRow(arrows,7,B00001100);

   arrows++;

 }

 arrows=0;

   delay(arrowt);

   changecheck();

   while(arrows<=4)

 {

   lc.setRow(arrows,0,B00011000);

   lc.setRow(arrows,1,B00110000);

   lc.setRow(arrows,2,B01100000);

   lc.setRow(arrows,3,B11000000);

   lc.setRow(arrows,4,B11000000);

   lc.setRow(arrows,5,B01100000);

   lc.setRow(arrows,6,B00110000);

   lc.setRow(arrows,7,B00011000);

   arrows++;

 }

arrows=0;

delay(arrowt);

}

void BlinkingEyes()

{

 lc.setRow(1,1,B00011100);

 lc.setRow(1,2,B00111110);

 lc.setRow(1,3,B01100011);

 lc.setRow(1,4,B01100011);

 lc.setRow(1,5,B00111110);

 lc.setRow(1,6,B00011100);

 //

 lc.setRow(3,1,B00111000);

 lc.setRow(3,2,B01111100);

 lc.setRow(3,3,B11000110);

 lc.setRow(3,4,B11000110);

 lc.setRow(3,5,B01111100);

 lc.setRow(3,6,B00111000);

 while (timenum<=18)

 {

   delaycheck();

 }

 timenum=0;

 lc.setRow(1,1,B00000000);

 lc.setRow(1,2,B0000000);

 lc.setRow(1,3,B01111111);

 lc.setRow(1,4,B01111111);

 lc.setRow(1,5,B0000000);

 lc.setRow(1,6,B00000000);

 //

 lc.setRow(3,1,B00000000);

 lc.setRow(3,2,B0000000);

 lc.setRow(3,3,B11111110);

 lc.setRow(3,4,B11111110);

 lc.setRow(3,5,B0000000);

 lc.setRow(3,6,B00000000);

 while (timenum<=5)

 {

   delaycheck();

 }

 timenum=0;

}

void CylonScanner()

{

 while (CylonON<=7 && CylonSON<=4)

     {

       lc.setColumn(CylonSON,CylonON,B11111111);

       lc.setColumn(CylonSOFF,CylonOFF,B00000000);

       CylonOFF++;

       CylonON++;

       if (CylonON>7)

       {

         CylonSON++;

         CylonON=0;

       }

       if (CylonOFF>7)

       {

         CylonSOFF++;

         CylonOFF=0;

       }

       delay(18);

       changecheck();

     }

     CylonON=4;

     CylonOFF=7;

     CylonSON=4;

     CylonSOFF=4;

     while (CylonON>=0 && CylonSON>=0)

     {

       lc.setColumn(CylonSON,CylonON,B11111111);

       lc.setColumn(CylonSOFF,CylonOFF,B00000000);

       CylonOFF--;

       CylonON--;

       if (CylonON<0)

       {

         CylonSON--;

         CylonON=7;

       }

       if (CylonOFF<0)

       {

         CylonSOFF--;

         CylonOFF=7;

       }

       delay(18);

       changecheck();

     }

     CylonON=3;

     CylonOFF=0;

     CylonSON=0;

     CylonSOFF=0;

}

void Dots()

{

 Clear();

 while (dotsR<=4)

 {

   lc.setRow(0,dotsR,B11100000);

   lc.setRow(4,dotsR,B00000111);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(0,dotsR,B11101110);

   lc.setRow(4,dotsR,B01110111);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(1,dotsR,B11100000);

   lc.setRow(3,dotsR,B00000111);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(1,dotsR,B11101110);

   lc.setRow(3,dotsR,B01110111);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(2,dotsR,B11100111);

   dotsR++;

 }

 dotsR=2;

 while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

 //

 while (dotsR<=4)

 {

   lc.setRow(0,dotsR,B00001110);

   lc.setRow(4,dotsR,B01110000);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(0,dotsR,B00000000);

   lc.setRow(4,dotsR,B00000000);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(1,dotsR,B00001110);

   lc.setRow(3,dotsR,B01110000);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(1,dotsR,B00000000);

   lc.setRow(3,dotsR,B00000000);

   dotsR++;

 }

 dotsR=2;

 delay(150);

 changecheck();

 while (dotsR<=4)

 {

   lc.setRow(2,dotsR,B00000000);

   dotsR++;

 }

 dotsR=2;

 delay(150);

}

void Drops()

{

 if (firsttime==1)

 {

   while (rainnum<=39)

   {

     while (rainS<=4)

     {

       rainON[rainnum]=random(1,256);

       lc.setRow(rainS,rainR[rainnum],rainON[rainnum]);

       rainnum++;

       rainS++;

     }

     rainS=0;

   }

  rainnum=0;

 }

 firsttime=0;

 while (rainnum<=39)

   {

     while (rainS<=4)

     {

       rainR[rainnum]++;

       if (rainR[rainnum]==8)

       {

         rainR[rainnum]=0;

         rainON[rainnum]=random(1,256);

       }

       lc.setRow(rainS,rainR[rainnum],rainON[rainnum]);

       rainnum++;

       rainS++;

     }

     rainS=0;

   }

  rainnum=0;

  while (timenum<=3)

 {

   delaycheck();

 }

 timenum=0;

}

void HeartBeat()

{

 lc.setColumn(0,0,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,1,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,2,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,3,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,4,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,5,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,6,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(0,7,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,0,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,1,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,2,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,3,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,4,B00001000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,5,B00001100);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,6,B00000110);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(1,7,B00000011);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,0,B00000011);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,1,B00000110);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,2,B00001100);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,3,B00011000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,4,B00110000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,5,B01100000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,6,B11000000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(2,7,B11000000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,0,B01100000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,1,B00110000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,2,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,3,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,4,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,5,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,6,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(3,7,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,0,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,1,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,2,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,3,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,4,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,5,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,6,B00010000);

 delay(beatDelaytime);

 changecheck();

 lc.setColumn(4,7,B00010000);

 delay(beatDelaytime);

 changecheck();

 beatC=0;

 beatS=0;

 while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

 while (beatS<=4)

 {

   while (beatC<=7)

   {

     lc.setColumn(beatS,beatC,B00000000);

     beatC++;

     delay(beatDelaytime);

     changecheck();

   }

   beatC=0;

   beatS++;

 }

while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

}

void HeartsFlashing()

{

 //first heart

 lc.setRow(0,0,B00000110);

 lc.setRow(0,1,B00001001);

 lc.setRow(0,2,B00001000);

 lc.setRow(0,3,B00001000);

 lc.setRow(0,4,B00000100);

 lc.setRow(0,5,B00000010);

 lc.setRow(0,6,B00000001);

 lc.setRow(1,0,B01100000);

 lc.setRow(1,1,B10010000);

 lc.setRow(1,2,B00010000);

 lc.setRow(1,3,B00010000);

 lc.setRow(1,4,B00100000);

 lc.setRow(1,5,B01000000);

 lc.setRow(1,6,B10000000);

 //second heart

 lc.setRow(2,0,B01100110);

 lc.setRow(2,1,B10011001);

 lc.setRow(2,2,B10000001);

 lc.setRow(2,3,B10000001);

 lc.setRow(2,4,B01000010);

 lc.setRow(2,5,B00100100);

 lc.setRow(2,6,B00011000);

 //third heart

 lc.setRow(3,0,B00000110);

 lc.setRow(3,1,B00001001);

 lc.setRow(3,2,B00001000);

 lc.setRow(3,3,B00001000);

 lc.setRow(3,4,B00000100);

 lc.setRow(3,5,B00000010);

 lc.setRow(3,6,B00000001);

 lc.setRow(4,0,B01100000);

 lc.setRow(4,1,B10010000);

 lc.setRow(4,2,B00010000);

 lc.setRow(4,3,B00010000);

 lc.setRow(4,4,B00100000);

 lc.setRow(4,5,B01000000);

 lc.setRow(4,6,B10000000);

 while (timenum<=5)

 {

   delaycheck();

 }

 timenum=0;

 lc.setRow(0,0,B00000110);

 lc.setRow(0,1,B00001111);

 lc.setRow(0,2,B00001111);

 lc.setRow(0,3,B00001111);

 lc.setRow(0,4,B00000111);

 lc.setRow(0,5,B00000011);

 lc.setRow(0,6,B00000001);

 lc.setRow(1,0,B01100000);

 lc.setRow(1,1,B11110000);

 lc.setRow(1,2,B11110000);

 lc.setRow(1,3,B11110000);

 lc.setRow(1,4,B11100000);

 lc.setRow(1,5,B11000000);

 lc.setRow(1,6,B10000000);

 //second heart

 lc.setRow(2,0,B01100110);

 lc.setRow(2,1,B11111111);

 lc.setRow(2,2,B11111111);

 lc.setRow(2,3,B11111111);

 lc.setRow(2,4,B01111110);

 lc.setRow(2,5,B00111100);

 lc.setRow(2,6,B00011000);

 //third heart

 lc.setRow(3,0,B00000110);

 lc.setRow(3,1,B00001111);

 lc.setRow(3,2,B00001111);

 lc.setRow(3,3,B00001111);

 lc.setRow(3,4,B00000111);

 lc.setRow(3,5,B00000011);

 lc.setRow(3,6,B00000001);

 lc.setRow(4,0,B01100000);

 lc.setRow(4,1,B11110000);

 lc.setRow(4,2,B11110000);

 lc.setRow(4,3,B11110000);

 lc.setRow(4,4,B11100000);

 lc.setRow(4,5,B11000000);

 lc.setRow(4,6,B10000000);

 while (timenum<=5)

 {

   delaycheck();

 }

 timenum=0;

}

void Hello()

{

 lc.setRow(1,7,B10001001);

 lc.setRow(2,7,B11101110);

 lc.setRow(3,7,B11100111);

 delay(helloT);

 changecheck();

 lc.setRow(1,6,B10001010);

 lc.setRow(2,6,B00000100);

 lc.setRow(3,6,B01001000);

 lc.setRow(4,6,B10000000);

 delay(helloT);

 changecheck();

 lc.setRow(1,5,B10001011);

 lc.setRow(2,5,B11000100);

 lc.setRow(3,5,B01001000);

 lc.setRow(4,5,B10000000);

 delay(helloT);

 changecheck();

 lc.setRow(1,4,B11111010);

 lc.setRow(2,4,B00100100);

 lc.setRow(3,4,B01001000);

 lc.setRow(4,4,B10000000);

 delay(helloT);

 changecheck();

 lc.setRow(1,3,B10001001);

 lc.setRow(2,3,B11000100);

 lc.setRow(3,3,B01000111);

 delay(helloT);

 changecheck();

 lc.setRow(1,2,B10001000);

 lc.setRow(2,2,B00000100);

 lc.setRow(3,2,B01000000);

 delay(helloT);

 changecheck();

 lc.setRow(1,1,B10001000);

 lc.setRow(2,1,B00001100);

 lc.setRow(3,1,B11000000);

 while (timenum<=50)

 {

   delaycheck();

 }

 timenum=0;

 while (helloS<=4)

 {

   while (helloR<=7)

   {

     lc.setRow(helloS,helloR,B00000000);

     helloR++;

   }

   helloR=0;

   helloS++;

 }

 helloS=0;

 helloR=0;

 while (timenum<=5)

 {

   delaycheck();

 }

 timenum=0;

}

void LazerEYE()

{

 while (LazerON<=7 && LazerSON<=4)

     {

       lc.setColumn(LazerSON,LazerON,B00011000);

       lc.setColumn(LazerSOFF,LazerOFF,B00000000);

       LazerOFF++;

       LazerON++;

       if (LazerON>7)

       {

         LazerSON++;

         LazerON=0;

       }

       if (LazerOFF>7)

       {

         LazerSOFF++;

         LazerOFF=0;

       }

       delay(5);

       changecheck();

     }

     LazerON=5;

     LazerOFF=7;

     LazerSON=4;

     LazerSOFF=4;

     while (LazerON>=0 && LazerSON>=0)

     {

       lc.setColumn(LazerSON,LazerON,B00011000);

       lc.setColumn(LazerSOFF,LazerOFF,B00000000);

       LazerOFF--;

       LazerON--;

       if (LazerON<0)

       {

         LazerSON--;

         LazerON=7;

       }

       if (LazerOFF<0)

       {

         LazerSOFF--;

         LazerOFF=7;

       }

       delay(5);

       changecheck();

     }

     LazerON=2;

     LazerOFF=0;

     LazerSON=0;

     LazerSOFF=0;

}

void MusicBars()

{

 if (firsttime==1)

 {

   musicbarnum=0;

   while (musicbarnum<32)

   {

     musicR[musicbarnum]=7;

     musicbarnum++;

   }

   firsttime=0;

   musicbarnum=0;

 }

 randmusic=random(0,4);

 while (musicBarcounter<=3)

 {

   lc.setLed(musicS[musicbarnum],musicR[musicbarnum],musicC[musicbarnum],true);

   lc.setLed(musicS[musicbarnum],musicR[musicbarnum]-1,musicC[musicbarnum],false);

   if (randmusic==1)

   {

     musicR[musicbarnum]++;

   }

   else if (randmusic==2)

   {

     musicR[musicbarnum]--;

   }

   if (musicR[musicbarnum]==8)

   {

     musicR[musicbarnum]=7;

   }

   if (musicR[musicbarnum]==-1)

   {

     musicR[musicbarnum]=0;

   }

   musicbarnum++;

   musicBarcounter++;

 }

 musicBarcounter=0;

 if (musicbarnum>31)

 {

   musicbarnum=0;

   musicBarcounter=0;

   delay(10);

 }

}

void OK ()

{

 lc.setColumn(2,1,B11111111);

 lc.setColumn(2,6,B11111111);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(2,0,B11111111);

 lc.setColumn(2,7,B11111111);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,7,B11111111);

 lc.setColumn(3,0,B11111111);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,6,B10000001);

 lc.setColumn(3,1,B00011000);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,5,B10000001);

 lc.setColumn(3,2,B00111100);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,4,B10000001);

 lc.setColumn(3,3,B01111110);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,3,B11111111);

 lc.setColumn(3,4,B01111110);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,2,B11111111);

 lc.setColumn(3,5,B11100111);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(1,1,B11111111);

 lc.setColumn(3,6,B11000011);

 delay(OKdelaytime);

 changecheck();

 lc.setColumn(3,7,B10000001);

 while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

 while (OKt<=2)

 {

   lc.setRow(1,0,B01111111);

   lc.setRow(1,1,B01110001);

   lc.setRow(1,2,B01110001);

   lc.setRow(1,3,B01110001);

   lc.setRow(1,4,B01110001);

   lc.setRow(1,5,B01110001);

   lc.setRow(1,6,B01110001);

   lc.setRow(1,7,B01111111);

   lc.setRow(2,0,B11000011);

   lc.setRow(2,1,B11000011);

   lc.setRow(2,2,B11000011);

   lc.setRow(2,3,B11000011);

   lc.setRow(2,4,B11000011);

   lc.setRow(2,5,B11000011);

   lc.setRow(2,6,B11000011);

   lc.setRow(2,7,B11000011);

   lc.setRow(3,0,B10000111);

   lc.setRow(3,1,B10011110);

   lc.setRow(3,2,B10111100);

   lc.setRow(3,3,B11111000);

   lc.setRow(3,4,B11111000);

   lc.setRow(3,5,B10111100);

   lc.setRow(3,6,B10011110);

   lc.setRow(3,7,B10000111);

   while (timenum<=20)

 {

   delaycheck();

 }

 timenum=0;

   while (OKs<=3)

   {

     while (OKc<=7)

     {

       lc.setRow(OKs,OKc,0);

       OKc++;

     }

     OKs++;

     OKc=0;

   }

   OKs=0;

   while (timenum<=10)

 {

   delaycheck();

 }

 timenum=0;

   OKt++;

 }

 OKt=0;

}

void PulsatingBar()

{

 if (firsttime==1)

 {

   while (num<=4)

   {

     while (rown<=4)

     {

       lc.setRow(num,rown,B11111111);

       rown++;

     }

     rown=2;

     num++;

   }

 }

 firsttime=0;

 while (num<=4)

 {

   lc.setIntensity(num,intense);

   num++;

 }

 intense++;

 num=0;

 while (timenum<=4)

 {

   delaycheck();

 }

 timenum=0;

 if (intense==6)

 {

   while (intense>=1)

   {

     while (num<=4)

     {

       lc.setIntensity(num,intense);

       num++;

     }

     intense--;

     while (timenum<=4)

 {

   delaycheck();

 }

 timenum=0;

     num=0;

   }

 }

}

void RandomFlashing()

{

 if (firsttime=1)

 {

   while (times<=400)

   {

     Matrixn=random(0,5);

     colN=random(0,8);

     rowN=random(0,8);

     lc.setLed(Matrixn,rowN,colN,true);

     times++;

   }

 }

 while (randomN<=2)

 {

   firsttime=0;

   Matrixn=random(0,5);

   colN=random(0,8);

   rowN=random(0,8);

   lc.setLed(Matrixn,rowN,colN,true);

   randomN++;

 }

 randomN=0;

 Matrixn=random(0,5);

 colN=random(0,8);

 rowN=random(0,8);

 lc.setLed(Matrixn,rowN,colN,false);

 delay(1);

}

void SingleFile()

{

 while (fileS<=4)

 {

   while (fileR<=7)

   {

     lc.setRow(fileS,fileR,B10001000);

     fileR++;

   }

   fileR=0;

   fileS++;

 }

 fileS=0;

 delay(150);

 changecheck();

 while (fileS<=4)

 {

   while (fileR<=7)

   {

     lc.setRow(fileS,fileR,B01000100);

     fileR++;

   }

   fileR=0;

   fileS++;

 }

 fileS=0;

 delay(150);

 changecheck();

 while (fileS<=4)

 {

   while (fileR<=7)

   {

     lc.setRow(fileS,fileR,B00100010);

     fileR++;

   }

   fileR=0;

   fileS++;

 }

 fileS=0;

 delay(150);

 changecheck();

 while (fileS<=4)

 {

   while (fileR<=7)

   {

     lc.setRow(fileS,fileR,B00010001);

     fileR++;

   }

   fileR=0;

   fileS++;

 }

 fileS=0;

 delay(150);

}

void TopBottomBars()

{

 while (tbOFF<=7)

 {

   while (tbS<=4)

   {

     lc.setRow(tbS,tbON,B11111111);

     lc.setRow(tbS,tbOFF,B00000000);

     tbS++;

   }

   tbS=0;

   delay(50);

   changecheck();

   tbOFF++;

   tbON++;

 }

 while (timenum<=6)

 {

   delaycheck();

 }

 timenum=0;

 tbON=0;

 tbOFF=-3;

}

void Wave()

{

 while (wavenum<40)

 {

     if (wavereturn[wavenum]==0)

     {

       if (waveR[wavenum]<=7)

       {

         lc.setLed(waveS[wavenum],waveR[wavenum]-1,waveC[wavenum],false);

         lc.setLed(waveS[wavenum],waveR[wavenum],waveC[wavenum],true);

         if (waveK[wavenum]==0)

         {

           waveR[wavenum]++;

           if (waveR[wavenum]>7)

           {

             waveR[wavenum]=7;

             waveK[wavenum]=1;

           }

         }

         else

         {

             waveK[wavenum]=0;

             wavereturn[wavenum]=1;

         }

       }

     }

     else

     {

       if (waveR[wavenum]>=0)

       {

         lc.setLed(waveS[wavenum],waveR[wavenum]+1,waveC[wavenum],false);

         lc.setLed(waveS[wavenum],waveR[wavenum],waveC[wavenum],true);

         if (waveK[wavenum]==0)

         {

           waveR[wavenum]--;

           if (waveR[wavenum]<0)

           {

             waveR[wavenum]=0;

             waveK[wavenum]=1;

           }

         }

         else

         {

             waveK[wavenum]=0;

             wavereturn[wavenum]=0;

         }

       }

     }

     wavenum++;

 }

       wavenum=0;

 delay(5);

}

void loop()

{

 Clear();

 Clearfirsttime=0;

 switch (mode)

 {

   case 1:

   time=millis();

     if (time-lasttime>30000)

     {

       mNum=0;

       if (playlist<17)

       {

         playlist++;

       }

       else

       {

         playlist=1;

       }

       lasttime=time;

       while (mNum<=4)

       {

         lc.clearDisplay(mNum);

         lc.setIntensity(mNum,3);

         mNum++;

       }

       delay(100);

       mNum=0;

       firsttime=1;

     }

   switch (playlist)

   {

     case 17:

     AllLightsOn();

     break;

     case 2:

     Arrows();

     break;

     case 3:

     BlinkingEyes();

     break;

     case 4:

     CylonScanner();

     break;

     case 5:

     Dots();

     break;

     case 6:

     Drops();

     break;

     case 7:

     HeartBeat();

     break;

     case 8:

     HeartsFlashing();

     break;

     case 9:

     Hello();

     break;

     case 10:

     LazerEYE();

     break;

     case 11:

     MusicBars();

     break;

     case 12:

     OK();

     break;

     case 13:

     PulsatingBar();

     break;

     case 14:

     RandomFlashing();

     break;

     case 15:

     SingleFile();

     break;

     case 16:

     TopBottomBars();

     break;

     case 1:

     Wave();

     break;

   }

   break;

   case 18:

   AllLightsOn();

   break;

   case 3:

   Arrows();

   break;

   case 4:

   BlinkingEyes();

   break;

   case 5:

   CylonScanner();

   break;

   case 6:

   Dots();

   break;

   case 7:

   Drops();

   break;

   case 8:

   HeartBeat();

   break;

   case 9:

   HeartsFlashing();

   break;

   case 10:

   Hello();

   break;

   case 11:

   LazerEYE();

   break;

   case 12:

   MusicBars();

   break;

   case 13:

   OK();

   break;

   case 14:

   PulsatingBar();

   break;

   case 15:

   RandomFlashing();

   break;

   case 16:

   SingleFile();

   break;

   case 17:

   TopBottomBars();

   break;

   case 2:

   Wave();

   break;

   if (change==1)

   {

   }

 }

}