```
#include "mbed.h"
#include "Grove_LCD_RGB_Backlight.h"
#include "lib_matrix.h"
#include "ascii_char_et.h"
Grove_LCD_RGB_Backlight lcd(PTC2,PTC1);

SPI spi(D11,D12,D13);          // Arduino compatible MOSI, MISO, SCLK
DigitalOut cs(D10);
DigitalIn cib1(D8);
DigitalIn cib2(D9);
DigitalIn cib3(PTE20);
DigitalIn cib4(PTE21);
DigitalIn cib5(PTE29);
DigitalIn cib6(PTE30);
DigitalIn cib7(PTB0);
DigitalIn bum1(D6);
DigitalIn bum2(D7);
DigitalIn trou(D5);
DigitalIn recup(D4);

DigitalOut LedBumper1(PTC12);
DigitalOut LedBumper2(PTC13);

//son
Serial serie(PTE22,PTE23);



char playPause[5] = {0x7E,0x03,0xAA,0xAD,0xEF}; //son
char stop[5] = {0x7E,0x03,0xAB,0xAE,0xEF};
char next[]= {0x7E,3,0xAC, 0xAF, 0xEF};
char previous[]= {0x7E,3,0xAD, 0xB0, 0xEF};
char tab_volume[]= {0x7E,4,0xAE, 0,0, 0xEF};
char num[]= {0x7E,5,0xA2, 0,0, 0xEF};
void send(char []);
void up();
void down();
void number(int nb);
int volume=0;

char tab[20];

int etatcib1, etatcib2, etatcib3, etatcib4, etatcib5, etatcib6, etatcib7, etatbum1, etatbum2, etattrou, etatrecup;
int i = 0;

void MAX7219_init(char noChips)      //matrice 8x8
{
    cs = 1;                        // CS initially High
    spi.format(8,0);               // 8-bit format, mode 0,0
    spi.frequency(1000000);        // SCLK = 1 MHz
    while(noChips)
        MAX7219_config(--noChips);
}
//---------------------------------------------------------------------------
void MAX7219_config(char chip)
{
    MAX7219_write(DECODE_MODE_REG,DISABLE_DECODE,chip);
    MAX7219_write(INTESITY_REG,BRIGHTNESS,chip);
    MAX7219_write(SCAN_LIMIT_REG,SCAN_ALL_DIGITS,chip);
    MAX7219_write(SHUTDOWN_REG,NORMAL_OPERATION,chip);
    MAX7219_write(DISPLAY_TEST_REG,DISABLE_TEST,chip);
}
```

```
//---------------------------------------------------------------------------
void MAX7219_write(char regName,char data,char chip)
{
   cs = 0;

   spi.write(regName);
   spi.write(data);
   while(chip--)
      MAX7219_NoOperation();       //Used for daisy chained (Cascaded) arrangements

   cs = 1;
}
//---------------------------------------------------------------------------
void MAX7219_displayText(char* text)
{
   char chip = 0;

   while(*text) {
      char row = (*text++) - 32;//(Text-32)...because the first 32 ASCII character codes are none
Printable (control chars)

      for(int col = 0; col < 8; col++) {
         MAX7219_write( col+1, symbol[row][col], chip );
      }

      chip++;
   }
}
void MAX7219_displayText(char* text, char indice, char nb_chip)
{
   char chip = 0;
```

```
    char * current_pointer;

    current_pointer=text+indice;

    while(chip<nb_chip) {

        char row = (*current_pointer++) - 32;//(Text-32)...because the first 32 ASCII character codes are
none Printable (control chars)

        for(int col = 0; col < 8; col++) {

            MAX7219_write( col+1, symbol[row][col], chip );

        }

        chip++;

    }

}

void MAX7219_display(unsigned const char led[], char no_chip)

{

    for(int col = 0; col < 8; col++) {

        MAX7219_write( col+1, led[col], no_chip );

    }

}

//-----------Passes the data to the adjacent MAX7219 in the Daisy Chain---------

void MAX7219_NoOperation()

{

    spi.write(NO_OP_REG);

    spi.write(0x00);              //Don't care (Can be any arbitrary value)

}



int main()

{


    cib1.mode(PullUp);

    cib2.mode(PullUp);

    cib3.mode(PullUp);
    cib4.mode(PullUp);
    cib5.mode(PullUp);
    cib6.mode(PullUp);
    cib7.mode(PullUp);
    bum1.mode(PullUp);
    bum2.mode(PullUp);
    trou.mode(PullUp);
    recup.mode(PullUp);

    int affiche;
    int i;
    lcd.setRGB(0,0,0);
    i=0;
    MAX7219_init(4);

//son
    int choix;
    char son0[7] = {0x7E,0x05,0xA2,0x00,0x00,0xA7,0xEF};
    char son1[7] = {0x7E,0x05,0xA2,0x00,0x01,0xA8,0xEF};
    char son2[7] = {0x7E,0x05,0xA2,0x00,0x02,0xA9,0xEF};
    char son3[7] = {0x7E,0x05,0xA2,0x00,0x03,0xAA,0xEF};
    char son4[7] = {0x7E,0x05,0xA2,0x00,0x04,0xAB,0xEF};
    char son5[7] = {0x7E,0x05,0xA2,0x00,0x05,0xAC,0xEF};
    char son6[7] = {0x7E,0x05,0xA2,0x00,0x06,0xAD,0xEF};
    char son7[7] = {0x7E,0x05,0xA2,0x00,0x07,0xAE,0xEF};
    char son8[7] = {0x7E,0x05,0xA2,0x00,0x08,0xAF,0xEF};
    char son9[7] = {0x7E,0x05,0xA2,0x00,0x09,0xB0,0xEF};



    while(1) {
      LedBumper1.write(0);
    LedBumper2.write(0);
    ///MAX7219_displayText("PLAY");

    etatcib1 = cib1.read();
    etatcib2 = cib2.read();
    etatcib3 = cib3.read();
    etatcib4 = cib4.read();
    etatcib5 = cib5.read();
    etatcib6 = cib6.read();
    etatcib7 = cib7.read();
    etatbum1 = bum1.read();
    etatbum2 = bum2.read();
    etattrou = trou.read();
    etatrecup = recup.read();
    printf("C1: %d | C2: %d | C3: %d | C4: %d | C5: %d | C6: %d | C7: %d | B1: %d | B2: %d | T: %d | Recup: %d \r\n", etatcib1, etatcib2, etatcib3, etatcib4, etatcib5, etatcib6, etatcib7, etatbum1, etatbum2, etattrou, etatrecup);


    LedBumper1.write(0);
    LedBumper2.write(0);

    MAX7219_displayText("PLAY");

    if(etatrecup==0) {
       lcd.setRGB(0,0,255);

    }

    if ((cib1==0) || (cib2==0) || (cib3==0) || (cib4==0) || (cib5==0) || (cib6==0) || (cib7==0)) { //toutes les cibles
       number(2);
```

```
   MAX7219_displayText("wp!");
   MAX7219_display(led_blank,3);
   affiche = 1;
   lcd.clear();
   i = i+5;
   lcd.setRGB(255,255,0);
   lcd.locate(0,0);
   lcd.print("+5");
   wait(1);
   MAX7219_displayText("PLAY");
   lcd.clear();

} else if((bum1==0) || (bum2==0)) { //bumper
   number(3);
   MAX7219_displayText("wow!");
   LedBumper1.write(1);
   LedBumper2.write(1);
   affiche = 1;
   lcd.clear();
   i = i+10;
   lcd.setRGB(0,255,255);
   lcd.locate(0,0);
   lcd.print("+10");
   wait(1);
   LedBumper1.write(0);
   LedBumper2.write(0);
   lcd.clear();
   MAX7219_displayText("PLAY");

} else if(trou ==0) {  //trou
   number(4);
```

```
    MAX7219_displayText("woaw");
    affiche = 1;
    LedBumper1.write(1);
    LedBumper2.write(1);
    wait(0.2);
    LedBumper1.write(0);
    LedBumper2.write(0);
    wait(0.2);
    LedBumper1.write(1);
    LedBumper2.write(1);
    wait(0.2);
    LedBumper1.write(1);
    LedBumper2.write(1);
    lcd.clear();
    i = i+50;
    lcd.setRGB(255,255,255);
    lcd.locate(0,0);
    lcd.print("+50");
    wait(1);
    lcd.clear();
    MAX7219_displayText("PLAY");

} else if(recup==0) { // capteur récupérateur
    number(5);
    affiche = 1;
    MAX7219_display(led_blank,0);
    MAX7219_display(led_blank,1);
    MAX7219_display(led_blank,2);
    MAX7219_display(led_blank,3);
    MAX7219_displayText("END!");
    lcd.clear();
```

```
            lcd.setRGB(255,255,0);
            lcd.locate(0,0);
            lcd.print("game over");
            wait(2);
            i=0;






        }
        if (affiche==1) { //boucle play
            lcd.locate(0,0);
            affiche = 0;
            lcd.setRGB(255,0,0);
            sprintf(tab,"scord --> %d",i);
            lcd.print(tab);
            number(1);
        }
    }
}


void number(int nb) //son
{
    int sum=nb+5+0+0xA2;
    printf("Ok, jouons la piste %d\n\r",nb);
    for(int i = 0; i < 4; i++) {
        serie.putc(num[i]);
    }
    serie.putc((char)nb);
    serie.putc((char)sum);

    serie.putc(0xEF);

}
```