Search the Arduino Playground

Democode for the `LedControl-library` (http://playground.arduino.cc/Main/LedControl)

Here are a few demo sketches using the `LedControl-library` (http://playground.arduino.cc/Main/LedControl). You can either copy+paste the code directly into the editor of the Arduino IDE, or download the sketches in a zip-file : `LedControlDemos.zip` (http://playground.arduino.cc/uploads/Main/LedControlDemos.zip).

Note :

> The `LedControl` library must obviously be installed on your computer, otherwise the demos won't even compile...

e#Output)
e#InputTOC)
e#ui)
e#Storage)
e#Communication)
e#General)

Table of contents

Demo for a Led matrix

Here is a small demo to be run on a LED matrix. A matrix of at least 5x7 LEDs is recommended, but there is no problem running it on a smaller matrix, it simply won't look that good.

Please have a close look at the initialization-code and the `setup()` function. If you don't to pull the `MAX72XX` from powersaving mode with the `shutdown()` function or if you forget to call `setIntensity()` for setting the display brightness, your LEDs may remain dark even though the sketch runs without errors.

Otherwise the code should be straight forward. `rows(), columns() and single()` show the different ways to update the matrix. `writeArduinoOnMatrix()` is a bit of eye-candy, that shows how you could drive a small text display if you had a few spare `MAX72XX` .

Please update the pin-numbers used in the code according to you hardware!

```
//We always have to include the library
#include "LedControl.h"

/*
 Now we need a LedControl to work with.
 ***** These pin numbers will probably not work with your hardware *****
 pin 12 is connected to the DataIn
 pin 11 is connected to the CLK
 pin 10 is connected to LOAD
 We have only a single MAX72XX.
 */
LedControl lc=LedControl(12,11,10,1);

/* we always wait a bit between updates of the display */
unsigned long delaytime=100;

void setup() {
  /*
   The MAX72XX is in power-saving mode on startup,
   we have to do a wakeup call
   */
  lc.shutdown(0,false);
  /* Set the brightness to a medium values */
  lc.setIntensity(0,8);
  /* and clear the display */
  lc.clearDisplay(0);
}

/*
 This method will display the characters for the
 word "Arduino" one after the other on the matrix.
 (you need at least 5x7 LEDs to see the whole chars)
 */
void writeArduinoOnMatrix() {
  /* here is the data for the characters */
  byte a[5]={B01111110,B10001000,B10001000,B10001000,B01111110};
  byte r[5]={B00111110,B00010000,B00100000,B00100000,B00010000};
  byte d[5]={B00011100,B00100010,B00100010,B00010010,B11111110};
  byte u[5]={B00111100,B00000010,B00000010,B00000100,B00111110};
  byte i[5]={B00000000,B00100010,B10111110,B00000010,B00000000};
  byte n[5]={B00111110,B00010000,B00100000,B00100000,B00011110};
  byte o[5]={B00011100,B00100010,B00100010,B00100010,B00011100};

  /* now display them one by one with a small delay */
  lc.setRow(0,0,a[0]);
  lc.setRow(0,1,a[1]);
  lc.setRow(0,2,a[2]);
  lc.setRow(0,3,a[3]);
  lc.setRow(0,4,a[4]);
  delay(delaytime);
  lc.setRow(0,0,r[0]);
  lc.setRow(0,1,r[1]);
  lc.setRow(0,2,r[2]);
  lc.setRow(0,3,r[3]);
  lc.setRow(0,4,r[4]);
  delay(delaytime);
  lc.setRow(0,0,d[0]);
  lc.setRow(0,1,d[1]);
  lc.setRow(0,2,d[2]);
  lc.setRow(0,3,d[3]);
  lc.setRow(0,4,d[4]);
  delay(delaytime);
  lc.setRow(0,0,u[0]);
  lc.setRow(0,1,u[1]);
  lc.setRow(0,2,u[2]);
  lc.setRow(0,3,u[3]);
  lc.setRow(0,4,u[4]);
  delay(delaytime);
  lc.setRow(0,0,i[0]);
  lc.setRow(0,1,i[1]);
  lc.setRow(0,2,i[2]);
  lc.setRow(0,3,i[3]);
  lc.setRow(0,4,i[4]);
  delay(delaytime);
  lc.setRow(0,0,n[0]);
  lc.setRow(0,1,n[1]);
  lc.setRow(0,2,n[2]);
  lc.setRow(0,3,n[3]);
  lc.setRow(0,4,n[4]);
  delay(delaytime);
  lc.setRow(0,0,o[0]);
  lc.setRow(0,1,o[1]);
  lc.setRow(0,2,o[2]);
  lc.setRow(0,3,o[3]);
  lc.setRow(0,4,o[4]);
  delay(delaytime);
  lc.setRow(0,0,0);
  lc.setRow(0,1,0);
  lc.setRow(0,2,0);
  lc.setRow(0,3,0);
  lc.setRow(0,4,0);
  delay(delaytime);
}

/*
  This function lights up a some LEDs in a row.
 The pattern will be repeated on every row.
 The pattern will blink along with the row-number.
 row number 4 (index==3) will blink 4 times etc.
 */
void rows() {
  for(int row=0;row<8;row++) {
    delay(delaytime);
    lc.setRow(0,row,B10100000);
    delay(delaytime);
    lc.setRow(0,row,(byte)0);
    for(int i=0;i<row;i++) {
      delay(delaytime);
      lc.setRow(0,row,B10100000);
      delay(delaytime);
      lc.setRow(0,row,(byte)0);
    }
  }
}

/*
  This function lights up a some LEDs in a column.
 The pattern will be repeated on every column.
 The pattern will blink along with the column-number.
 column number 4 (index==3) will blink 4 times etc.
 */
void columns() {
  for(int col=0;col<8;col++) {
    delay(delaytime);
    lc.setColumn(0,col,B10100000);
    delay(delaytime);
    lc.setColumn(0,col,(byte)0);
    for(int i=0;i<col;i++) {
      delay(delaytime);
      lc.setColumn(0,col,B10100000);
      delay(delaytime);
      lc.setColumn(0,col,(byte)0);
    }
  }
}

/*
 This function will light up every LED on the matrix.
 The led will blink along with the row-number.
 row number 4 (index==3) will blink 4 times etc.
 */
void single() {
  for(int row=0;row<8;row++) {
    for(int col=0;col<8;col++) {
      delay(delaytime);
      lc.setLed(0,row,col,true);
      delay(delaytime);
      for(int i=0;i<col;i++) {
        lc.setLed(0,row,col,false);
        delay(delaytime);
        lc.setLed(0,row,col,true);
        delay(delaytime);
      }
    }
  }
}

void loop() {
  writeArduinoOnMatrix();
  rows();
  columns();
  single();
}
```

Demo for a 7-segment display

Here is a demo to be run on a 7-segment display. The initialization of the devices is exactly the same as in the matrix demo. `scrollDigits()` uses the `setDigits()` method for (hex-) numbers between 0 and 15.

`writeArduinoOn7Segment()` is a little bit more interesting, as it uses the `setChar()` method for the predefined characters *'A','d'* and the `setRow()` function for creating a mock up of the missing characters.

Please update the pin-numbers used in the code according to you hardware!

```
//We always have to include the library
#include "LedControl.h"

/*
 Now we need a LedControl to work with.
 ***** These pin numbers will probably not work with your hardware *****
 pin 12 is connected to the DataIn
 pin 11 is connected to the CLK
 pin 10 is connected to LOAD
 We have only a single MAX72XX.
 */
LedControl lc=LedControl(12,11,10,1);

/* we always wait a bit between updates of the display */
unsigned long delaytime=250;

void setup() {
  /*
   The MAX72XX is in power-saving mode on startup,
   we have to do a wakeup call
   */
  lc.shutdown(0,false);
  /* Set the brightness to a medium values */
  lc.setIntensity(0,8);
  /* and clear the display */
  lc.clearDisplay(0);
}


/*
 This method will display the characters for the
 word "Arduino" one after the other on digit 0.
 */
void writeArduinoOn7Segment() {
  lc.setChar(0,0,'a',false);
  delay(delaytime);
  lc.setRow(0,0,0x05);
  delay(delaytime);
  lc.setChar(0,0,'d',false);
  delay(delaytime);
  lc.setRow(0,0,0x1c);
  delay(delaytime);
  lc.setRow(0,0,B00010000);
  delay(delaytime);
  lc.setRow(0,0,0x15);
  delay(delaytime);
  lc.setRow(0,0,0x1D);
  delay(delaytime);
  lc.clearDisplay(0);
  delay(delaytime);
}

/*
  This method will scroll all the hexadecimal
 numbers and letters on the display. You will need at least
 four 7-Segment digits. otherwise it won't really look that good.
 */
void scrollDigits() {
  for(int i=0;i<13;i++) {
    lc.setDigit(0,3,i,false);
    lc.setDigit(0,2,i+1,false);
    lc.setDigit(0,1,i+2,false);
    lc.setDigit(0,0,i+3,false);
    delay(delaytime);
  }
  lc.clearDisplay(0);
  delay(delaytime);
}

void loop() {
  writeArduinoOn7Segment();
  scrollDigits();
}
```

Demo for driving more than one `MAX72XX`

Here is a demo that shows how to address more than one `MAX72XX` from a single `LedControl` variable. You would obviously need at least two cascaded devices to test this.

The demo doesn't do anything exciting. It simply uses a big loop in which it switches all LEDs on all devices on and off, one after the other.

The other point of interest is the `setup()` function that initializes all devices in a loop.

Please update the pin-numbers used in the code according to you hardware!

```
//We always have to include the library
#include "LedControl.h"

/*
 Now we need a LedControl to work with.
 ***** These pin numbers will probably not work with your hardware *****
 pin 12 is connected to the DataIn
 pin 11 is connected to the CLK
 pin 10 is connected to LOAD
 ***** Please set the number of devices you have *****
 But the maximum default of 8 MAX72XX will also work.
 */
LedControl lc=LedControl(12,11,10,8);

/* we always wait a bit between updates of the display */
unsigned long delaytime=500;

/*
 This time we have more than one device.
 But all of them have to be initialized
 individually.
 */
void setup() {
  //we have already set the number of devices when we created the LedControl
  int devices=lc.getDeviceCount();
  //we have to init all devices in a loop
  for(int address=0;address<devices;address++) {
    /*The MAX72XX is in power-saving mode on startup*/
    lc.shutdown(address,false);
    /* Set the brightness to a medium values */
    lc.setIntensity(address,8);
    /* and clear the display */
    lc.clearDisplay(address);
  }
}

void loop() {
  //read the number cascaded devices
  int devices=lc.getDeviceCount();

  //we have to init all devices in a loop
  for(int row=0;row<8;row++) {
    for(int col=0;col<8;col++) {
      for(int address=0;address<devices;address++) {
        delay(delaytime);
        lc.setLed(address,row,col,true);
        delay(delaytime);
        lc.setLed(address,row,col,false);
      }
    }
  }
}
```