Control Scroll Text Speed on 8X8 LED Matrix

http://playground.arduino.cc/Main/DirectDriveLEDMatrix

This example scroll to the left the "Hello" message.

HARDWARE REQUIRED

Arduino Board
(1) 8 x 8 LED Matrix
(16) 220 Ohm resistors
hook-up wire
breadboard

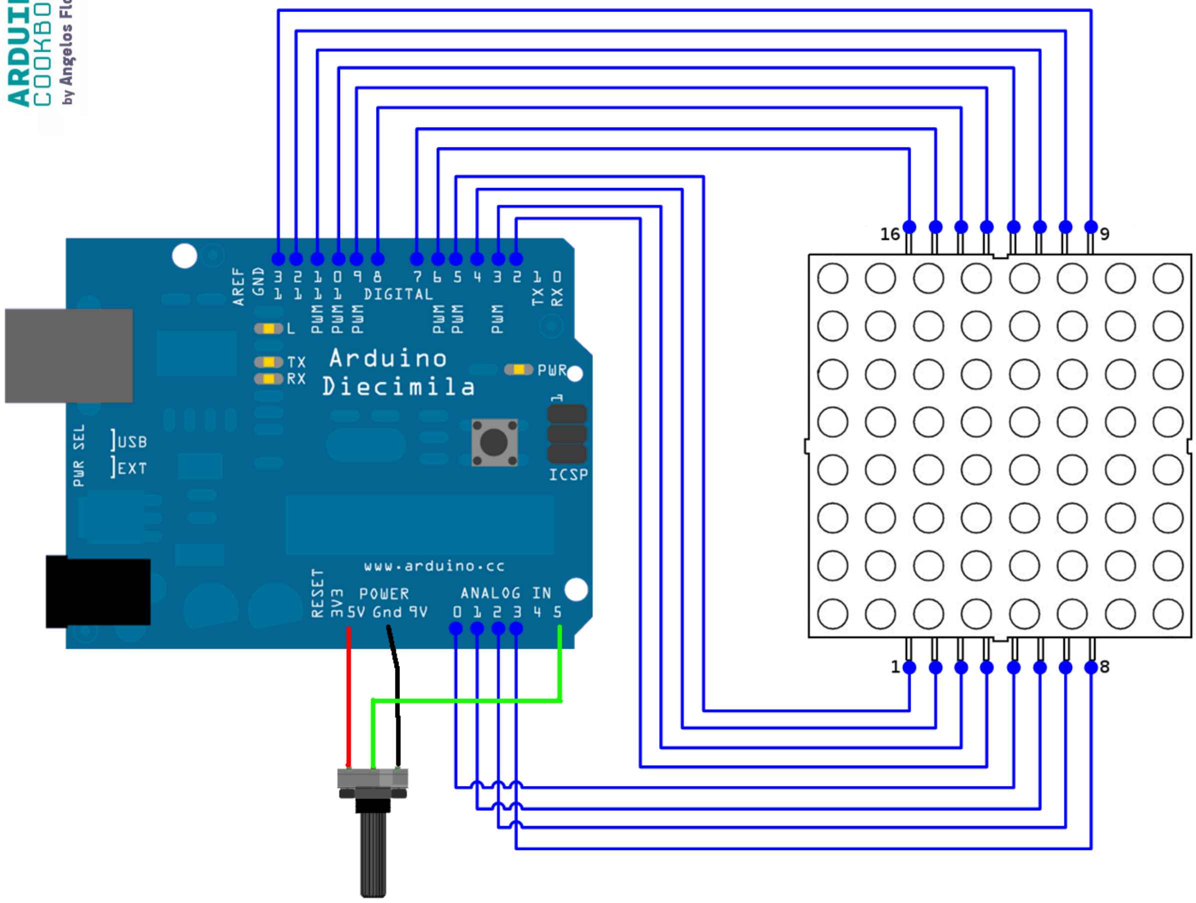
CIRCUIT

Direct wiring an Arduino to an LED matrix

```
// pin[xx] on led matrix connected to nn on Arduino (-1 is dummy to make array start at pos 1)
int pins[17]= {-1, 5, 4, 3, 2, 14, 15, 16, 17, 13, 12, 11, 10, 9, 8, 7, 6};
```

IMAGE

ARDUINO COOKBOOK by Angelos Floros



CODE

```
/*
 * Show messages on an 8x8 led matrix,
 * scrolling from right to left.
 *
 * Uses FrequencyTimer2 library to
 * constantly run an interrupt routine
 * at a specified frequency. This
 * refreshes the display without the
 * main loop having to do anything.
 *
 */
```

```
#include <FrequencyTimer2.h>

#define SPACE { \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0}, \
    {0, 0, 0, 0, 0, 0, 0, 0} \
}

#define H { \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 1, 1, 1, 1, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}  \
}

#define E  { \
    {0, 1, 1, 1, 1, 1, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 1, 1, 1, 1, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 1, 1, 1, 1, 1, 0}  \
}

#define L { \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 0, 0}, \
    {0, 1, 1, 1, 1, 1, 1, 0}  \
}

#define O { \
    {0, 0, 0, 1, 1, 0, 0, 0}, \
    {0, 0, 1, 0, 0, 1, 0, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 1, 0, 0, 0, 0, 1, 0}, \
    {0, 0, 1, 0, 0, 1, 0, 0}, \
    {0, 0, 0, 1, 1, 0, 0, 0}  \
}
```

CODE

```
byte col = 0;
byte leds[8][8];

// pin[xx] on led matrix connected to nn on Arduino (-1 is dummy to make array start at pos 1)
int pins[17]= {-1, 5, 4, 3, 2, 14, 15, 16, 17, 13, 12, 11, 10, 9, 8, 7, 6};
// col[xx] of leds = pin yy on led matrix
int cols[8] = {pins[13], pins[3], pins[4], pins[10], pins[06], pins[11], pins[15], pins[16]};

// row[xx] of leds = pin yy on led matrix
int rows[8] = {pins[9], pins[14], pins[8], pins[12], pins[1], pins[7], pins[2], pins[5]};

const int analogInPin = A5;  // Analog input pin that the potentiometer is attached to
const int numPatterns = 6;

int sensorValue = 0;        // value read from the pot
int outputValue = 0;        // value output to the PWM (analog out)

byte patterns[numPatterns][8][8] = {H,E,L,L,O,SPACE};

int pattern = 0;

void setup() {
  Serial.begin(9600);
  // sets the pins as output
  for (int i = 1; i <= 16; i++) {
    pinMode(pins[i], OUTPUT);
  }

  // set up cols and rows
  for (int i = 1; i <= 8; i++) {
    digitalWrite(cols[i - 1], HIGH);
  }

  for (int i = 1; i <= 8; i++) {
    digitalWrite(rows[i - 1], HIGH);
  }

  clearLeds();

  // Turn off toggling of pin 11
  FrequencyTimer2::disable();
  // Set refresh rate (interrupt timeout period)
  FrequencyTimer2::setPeriod(2000);
  // Set interrupt routine to be called
  FrequencyTimer2::setOnOverflow(display);

  setPattern(pattern);
}

void loop() {
    // read the analog in value:
  sensorValue = analogRead(analogInPin);
  // map it to the range of the analog out:
  outputValue = map(sensorValue, 0, 1023, 20, 100);


    pattern = ++pattern % numPatterns;
    slidePattern(pattern, outputValue);

  Serial.print(“pattern = “ );
  Serial.print(pattern);
  Serial.print(“\t sensorValue = “);
  Serial.print(sensorValue);
  Serial.print(“\t outputValue = “);
  Serial.println(outputValue);
}
```

CODE

```
void clearLeds() {
  // Clear display array
  for (int i = 0; i < 8; i++) {
    for (int j = 0; j < 8; j++) {
      leds[i][j] = 0;
    }
  }
}

void setPattern(int pattern) {
  for (int i = 0; i < 8; i++) {
    for (int j = 0; j < 8; j++) {
      leds[i][j] = patterns[pattern][i][j];
    }
  }
}

void slidePattern(int pattern, int del) {
  for (int l = 0; l < 8; l++) {
    for (int i = 0; i < 7; i++) {
      for (int j = 0; j < 8; j++) {
        leds[j][i] = leds[j][i+1];
      }
    }
    for (int j = 0; j < 8; j++) {
      leds[j][7] = patterns[pattern][j][0 + l];
    }
    delay(del);
  }
}

// Interrupt routine
void display() {


  digitalWrite(cols[col], HIGH);  // Turn whole previous column off
  col++;
  if (col == 8) {
    col = 0;
  }
  for (int row = 0; row < 8; row++) {
    if (leds[col][7 - row] == 1) {
      digitalWrite(rows[row], HIGH);  // Turn on this led
    }
    else {
      digitalWrite(rows[row], LOW); // Turn off this led
    }
  }
  digitalWrite(cols[col], LOW);
  // Turn whole column on at once (for equal lighting times)
}
```