Project Title : Programming 8x8 LED Matrix

Project Link : <https://create.arduino.cc/projecthub/SAnwandter1/programming-8x8-led-matrix-23475a?ref=platform&ref_id=424_popular_part_beginner_&offset=7>

Hardwares

- Jumper wires (generic) x 16

- Breadboard (generic) x 2

- Arduino UNO x 1

- Adafruit 8x8 LED Matrix x 1

- Adafruit USB Cable – Standard A-B for ARDUINO x 1

Softwares

- Code

//update from SAnwandter

#define ROW\_1 2

#define ROW\_2 3

#define ROW\_3 4

#define ROW\_4 5

#define ROW\_5 6

#define ROW\_6 7

#define ROW\_7 8

#define ROW\_8 9

#define COL\_1 10

#define COL\_2 11

#define COL\_3 12

#define COL\_4 13

#define COL\_5 A0

#define COL\_6 A1

#define COL\_7 A2

#define COL\_8 A3

const byte rows[] = {

ROW\_1, ROW\_2, ROW\_3, ROW\_4, ROW\_5, ROW\_6, ROW\_7, ROW\_8

};

const byte col[] = {

COL\_1,COL\_2, COL\_3, COL\_4, COL\_5, COL\_6, COL\_7, COL\_8

};

// The display buffer

// It's prefilled with a smiling face (1 = ON, 0 = OFF)

byte ALL[] = {B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111,B11111111};

byte EX[] = {B00000000,B00010000,B00010000,B00010000,B00010000,B00000000,B00010000,B00000000};

byte A[] = { B00000000,B00111100,B01100110,B01100110,B01111110,B01100110,B01100110,B01100110};

byte B[] = {B01111000,B01001000,B01001000,B01110000,B01001000,B01000100,B01000100,B01111100};

byte C[] = {B00000000,B00011110,B00100000,B01000000,B01000000,B01000000,B00100000,B00011110};

byte D[] = {B00000000,B00111000,B00100100,B00100010,B00100010,B00100100,B00111000,B00000000};

// E, F, G … Z

}

float timeCount = 0;

void setup()

{

// Open serial port

Serial.begin(9600);

// Set all used pins to OUTPUT

// This is very important! If the pins are set to input

// the display will be very dim.

for (byte i = 2; i <= 13; i++)

pinMode(i, OUTPUT);

pinMode(A0, OUTPUT);

pinMode(A1, OUTPUT);

pinMode(A2, OUTPUT);

pinMode(A3, OUTPUT);

}

void loop() {

// This could be rewritten to not use a delay, which would make it appear brighter

delay(5);

timeCount += 1;

if(timeCount < 20)

{

drawScreen(A);

}

else if (timeCount < 40)

{

drawScreen(R);

}

else if (timeCount < 60)

{

drawScreen(D);

}

else if (timeCount < 80)

{

drawScreen(U);

}

else if (timeCount < 100)

{

drawScreen(I);

}

else if (timeCount < 120)

{

drawScreen(N);

}

else if (timeCount < 140) {

drawScreen(O);

}

else if (timeCount < 160)

{

drawScreen(ALL);

}

else if (timeCount < 180)

{

drawScreen(ALL);

}

else {

// back to the start

timeCount = 0;

}

}

void drawScreen(byte buffer2[])

{

// Turn on each row in series

for (byte i = 0; i < 8; i++) // count next row

{

digitalWrite(rows[i], HIGH); //initiate whole row

for (byte a = 0; a < 8; a++) // count next row

{

// if You set (~buffer2[i] >> a) then You will have positive

digitalWrite(col[a], (buffer2[i] >> a) & 0x01); // initiate whole column

delayMicroseconds(100);

digitalWrite(col[a], 1); // reset whole column

}

digitalWrite(rows[i], LOW); // reset whole row

// otherwise last row will intersect with next row

}

}

- Why you like it, and how it may help your project

It’s quite easy for starters like me, since this project only uses very basic functions like digitalWrite(), delay() etc… So I can review basics of Arduino codes so it can be helpful for my next project.