

BicoKit

Understanding the basics of dot matrix display

Introduction

Mastering energy was the result of understanding physic laws, and we not only teach these laws at school. but also spend time doing practical work with different set-ups. Information technology spreads so fast and the problem was to use it and not understand it. Teachers did not follow; they gave a screen to kids and consider that playing with an application is computer programming. They ignore all about computer history and basic concepts.

Arduino made a great job promoting the understanding of a little bit of electronics and programming in C, simplifying hardware difficulties and proposing libraries that hide concepts.

Screens are definitively part of our life. We cannot explain there complexity, same as for cars. But, is understanding a cardan mechanism something to be learned? We say yes..

So for the matrix displays. Let us start from scratch and understand the underlying concepts

We tried here to list all the useful knowlege, up to you to select the useful ones for the initial and subsequent steps.

Basics on Leds

Many concepts to explain.

Anode-Cathode. Current–voltage characteristic, Limiting resistor . PWM for dimming. Light intensity units. Eye response to pulsing light and intensity.

Basics on wiring

Microprocessor port and pins. Source/sink current. Input state same as no state or floating. Pull-up can drive a Led Common anode, common cathode. Possible connection schemes. Principle of Charlieplexing (see bottom note 1).

BicoKit is wired the simplest way on an Arduino. This limits to a 6x8 matrix but is cheap and does not limit pedagogical features.

Basics on programming

BicoKit uses the Arduino IDE and C. C should have been taught before, not the Arduino way, but insisting on the #define, the hierarchy of modules, the documentation.

Basics on software development tools

Arduino IDE is not the best example of a software development envirom[^]nment, but it has the advantage to be simple and well spread, that is easy to find help for installing or degugging.

Note 1 Understanding and programming Charlieplexing is surely interesting but it exploit a single concept and is not very rich in programming challenges.. Didel has a kit

<https://www.tindie.com/products/jdn/charlieplexing-kit-for-arduino/>

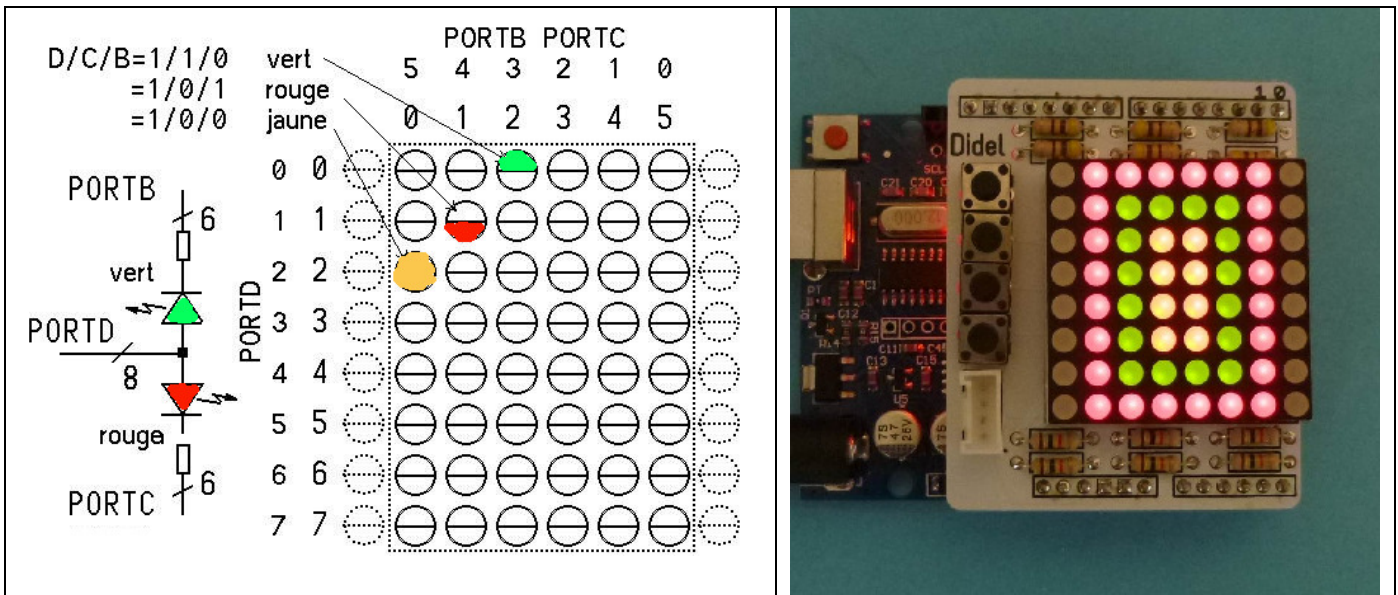
and documentation <https://www.didel.com/kits/CharlieArduino.pdf>

One finds a lot of explanations, examples and youtubes on the Web.

BicoKit schematic and definitions

Note: As for the previous page, this is a document for an educator who knows Arduino and C.

The Led array circuit is from Guoyexing Optoelectronix, model GYXM-1388ASRG, doc on Web. The CSRG model (common cathode) can also be used, just need to reverse voltage pins.



There are 8 rows of 6 bicolor diodes. It may be good to hide the unused row on both side.

Simple access

Rows are connected on PORTD. Bit 0 (also Arduino Pin 0=Rx) selects upper row (see note 1). Pins 1 to 7 select the following rows. Understand 0 to 7 is a bit number. The weights are 1,2,4,8, 10,20,40,80 in hexadecimal. Selecting the fourth row for instance (and only that one) can be written

```
PORTD = 1<<4; or PORTD = 0x10 or PORTD = 16;
```

PORTB, bits 0 to 5, selects column 0 to 5 in reverse order. It has been considered that the corner up left pixel has coordinates (0,0). It is also good for conceptualizing the difference between dot coordinates and bit numbers, and forcing definitions.

Selected diodes are green. In order to make the last row appear green, one need two instructions:

```
PORTD = 0x80; PORTB = 0;
```

PORTC, bits 0 to 5, selects column 0 to 5 in reverse order.

Selected diodes are red. In order to make the first row appear red, one need two instructions:

```
PORTD = 0x01; PORTC = 0;
```

Of course, clearing both ports B and C will show orange/yellow Leds (note 2).

Note 1 Rx is used while downloading the program, but there is no compatibility problem since ports B and C are floating during the transfer. Same for Tx. No blinking during transfer.

Note2 Charging pins with leds in parallels i compatible with the microcontroller. Reducing resistance values is not recommended.

Selecting several rows will of course light the leds at the crossing of row/columns.

bitmaps

In order to fill the bitmap with any data, one have to scan:

select first line, write color data, small delay, next line, etc

Understanding delay is important. For instance, no delay at all makes the last line brighter – looping back takes more time than selecting a row.

Too long delay gives flickering.

How to reduce intensity? Shorten all delays? Add a longer delay after the last row?— ok if you clear the row during the additional delay.
Best solution may be to have 2 delays for every row (PWM), even more delays if differences of intensity must occur on a line.

! Do not write delay(10);, use #define to specify the functional delay the application need: DOn Doff Dreturn etc

At this lever, play building tables of sprites, do vertical and horizontal scroll.

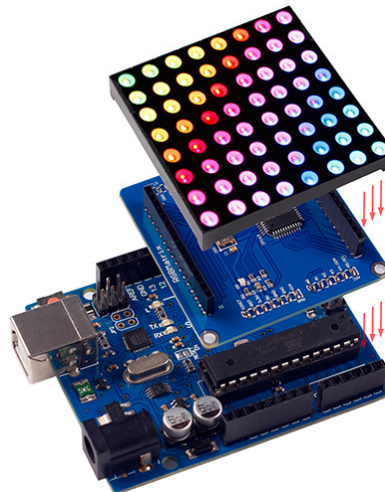
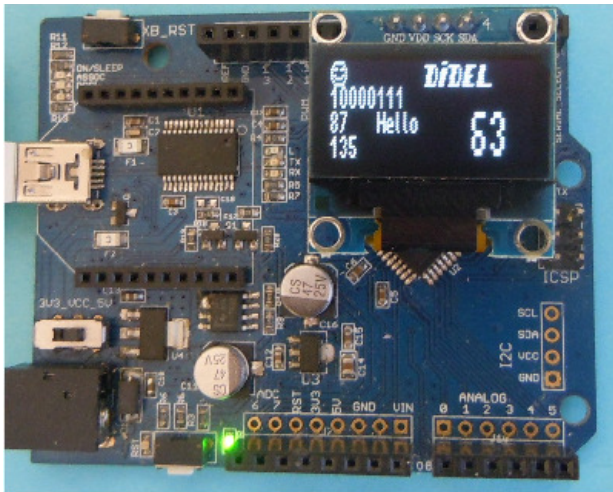
Bitmap

4 solutions for a bitmap in memory.

- 1) 6x8 bytes, 2 bits per byte since 2 Leds per pixel
- 2) 6x8 bytes, 3 bits per byte since colors per pixel
- 3) 2 tables of 8 bytes, easy to transfer to registers,
- 4) 3 tables of 8 bytes

Programming any of these is not easy. Algorithms use logical operations.
It is easier to work with the bmp if functions PixOn(x,y); PixOff(x,y); and ClrScreen(); are given.
From these, on can draw lines, understand anti-aliasing, Bresenham algorithm. One can see how difficult it is to move objects, compare the sizes of code for creating a sprite using the library or acting directly on the registers.

As a next step, Oleds with a resolution of 64x128 pixels can be inserted directly on the Arduino pins and OLedMap or OledPix library used. Colorduino follows a different track. There are many other proposals for graphic displays, some with touch panel. Of course, using Android or game builders will attract kids. The value of the BicoKit is its simplicity (and low cost).
Up to you to decide your pedagogic objective according to you public and available time.



Note Simple test programs are available under www.didel.com/BicoKit.zip.