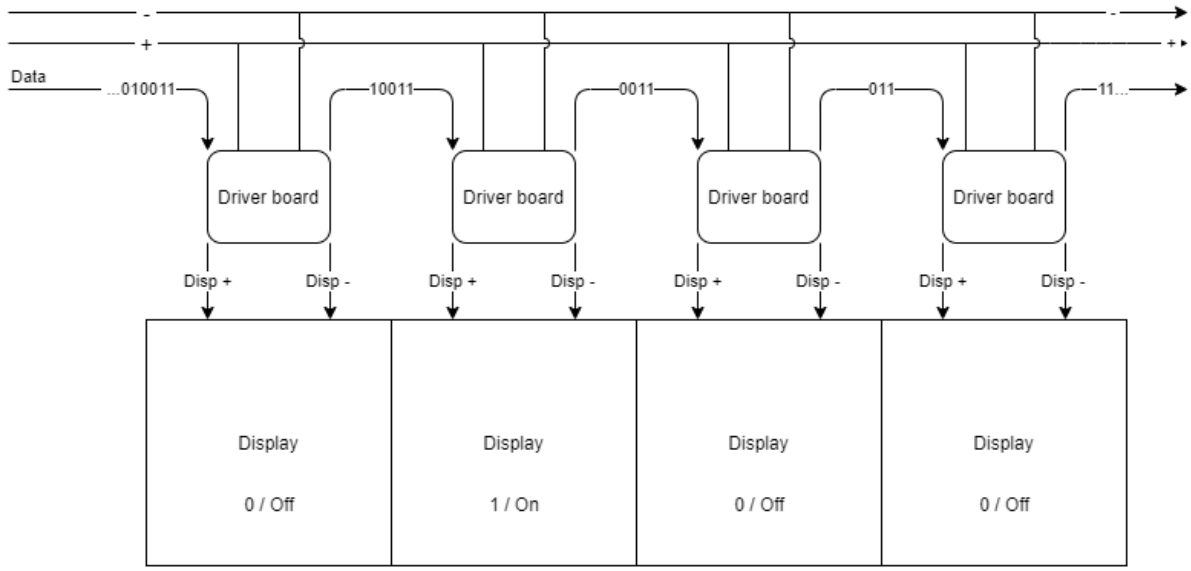
Programming the DecoChrom ATTiny display driver board with Arduino Uno

Timo Luukkonen 18.3.2021

Introduction

The DecoChrom ATTiny display driver board is based on the ATtiny85 chip. To make it function the chip needs to be programmed, e.g. by using an Arduino Uno as a programmer.

Wiring example of the driver boards with the DecoChrom displays:



Step 1: Add Support for the ATtiny85

First you need to add support for the Attiny85 to the Arduino Board Manager:

- From the **Arduino IDE**, go to *Arduino -> Preferences,* then scroll down to *Additional Board Managers URLs*
- Copy & paste the following URL:

https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json

- Press "OK" at the bottom then restart the Arduino IDE

Step 2: Install the ATtiny board package

- From the **Arduino IDE** go to *Tools -> Board -> Boards Manager*
- A new tab will open and at the top of the tab type: attiny
- Select Install on the *Attiny* by David. A Mellis
- Restart the Arduino IDE

Step 3: Set the Arduino Into ISP Mode

What we want to do, is to be able to program the ATtiny85 from the Arduino IDE.
This requires to burn the bootloader to the ATtiny85.
To do this, we'll need to "prep" the Arduino by uploading the ISP sketch to it.

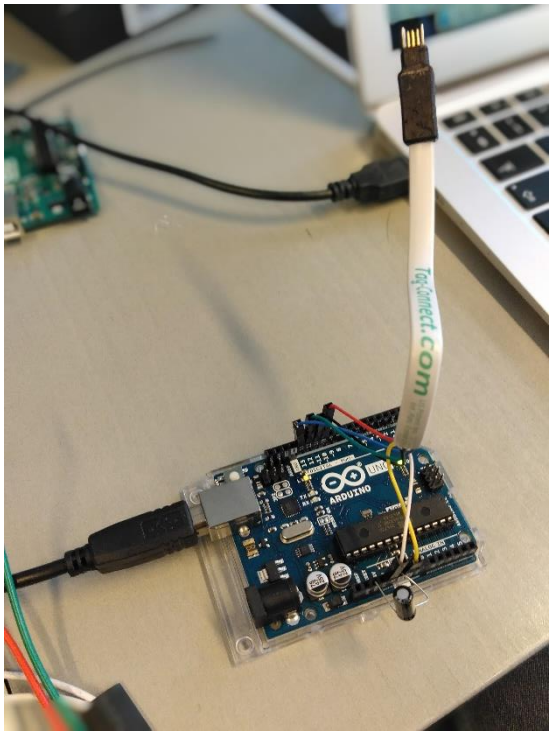
- In the **Arduino IDE** select *File -> Examples -> 11. Arduino ISP -> **ArduinoISP***
- Upload the opened sketch to your Arduino.

A message will appear saying **"Done Uploading".**

Step 4: Connect the Plug of nails to the Arduino

To get a connection from the Arduino to the ATtiny85, use the Tag-Connect Plug-of-nails connector.

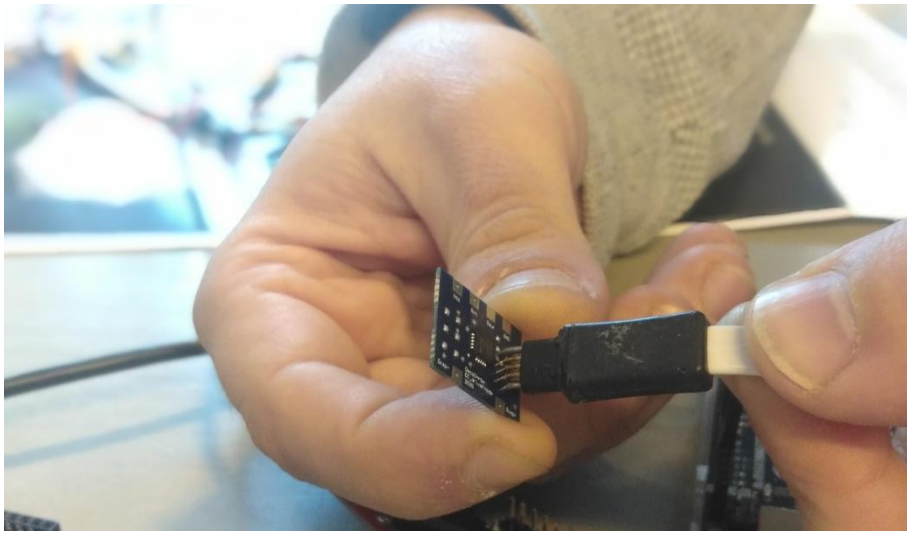
- The wires connect to the Arduino in the following way:



Tag-Connect Plug of nails Wire Color	Arduino Pin
White	5V
Yellow	GND
Red	10
Black	11
Green	12
Blue	13

Also, add an electrolytic capacitor between the Arduinos RESET & GND Pins, to prevent the Arduino from restarting itself.
10uF is recommended.

- Connect the cable pins to the board, by using the guide pins:



If you're using a different cable, check the pin connections from the table in the end of this document.

Step 5: Burn the bootloader in to the chip

Next, we need to burn the Arduino bootloader onto the ATtiny85 chip, to make sure the chip will accept any programs uploaded via the Arduino IDE.

To do this, the following settings needs to be selected from the **Arduino IDE**:

- *Tools -> **ATtiny25/45/85***
- *Tools -> Processor -> **8 MHz (internal)***
- *Tools -> Programmer -> **Arduino as ISP***
- Finally select *Tools -> **Burn Bootloader***

A message will appear saying **"Done Burning Bootloader"**

Step 6: Upload the display driver program to the chip

Now we can upload the actual display driver program to the chip.

- Download the program file from the following URL**:**

https://github.com/DecoChrom/ApplicationCases/blob/master/EC_display_serial_control/EC_display_driver.ino

- Open the program in **Arduino IDE**, and select *Upload.*

A message will appear saying **"Done Uploading".**

Now the chip is ready to be used as a driver module to the DecoChrom displays.

Attachements / Links / Sources

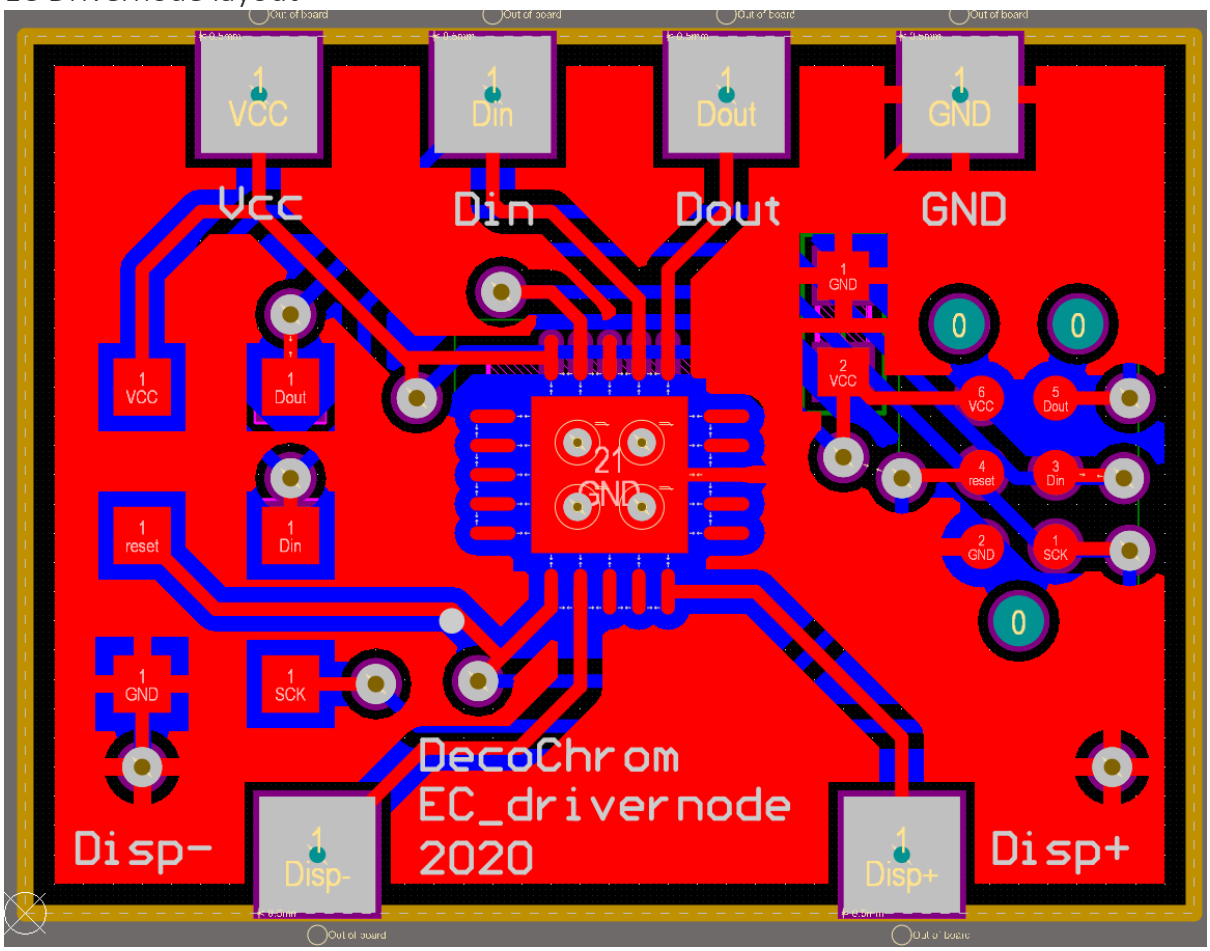
Instructables – How to program ATtiny85 from an Arduino Uno
https://www.instructables.com/How-to-Program-an-Attiny85-From-an-Arduino-Uno/

Github – DecoChrome source codes
https://github.com/DecoChrom/ApplicationCases/tree/master/EC_display_serial_control

Pin wiring from Arduino -> DecoChrom ATTiny display driver board

Arduino pin	DC ATTIny display driver board pin number	DC ATTIny display driver board pin name
5V	6	VCC
GND	2	GND
10	4	Reset
11	5	Dout/PCINT0/MOSI
12	3	Din/PCINT1/MISO
13	1	SCK

EC Drivernode layout



EC Drivernode Schematic