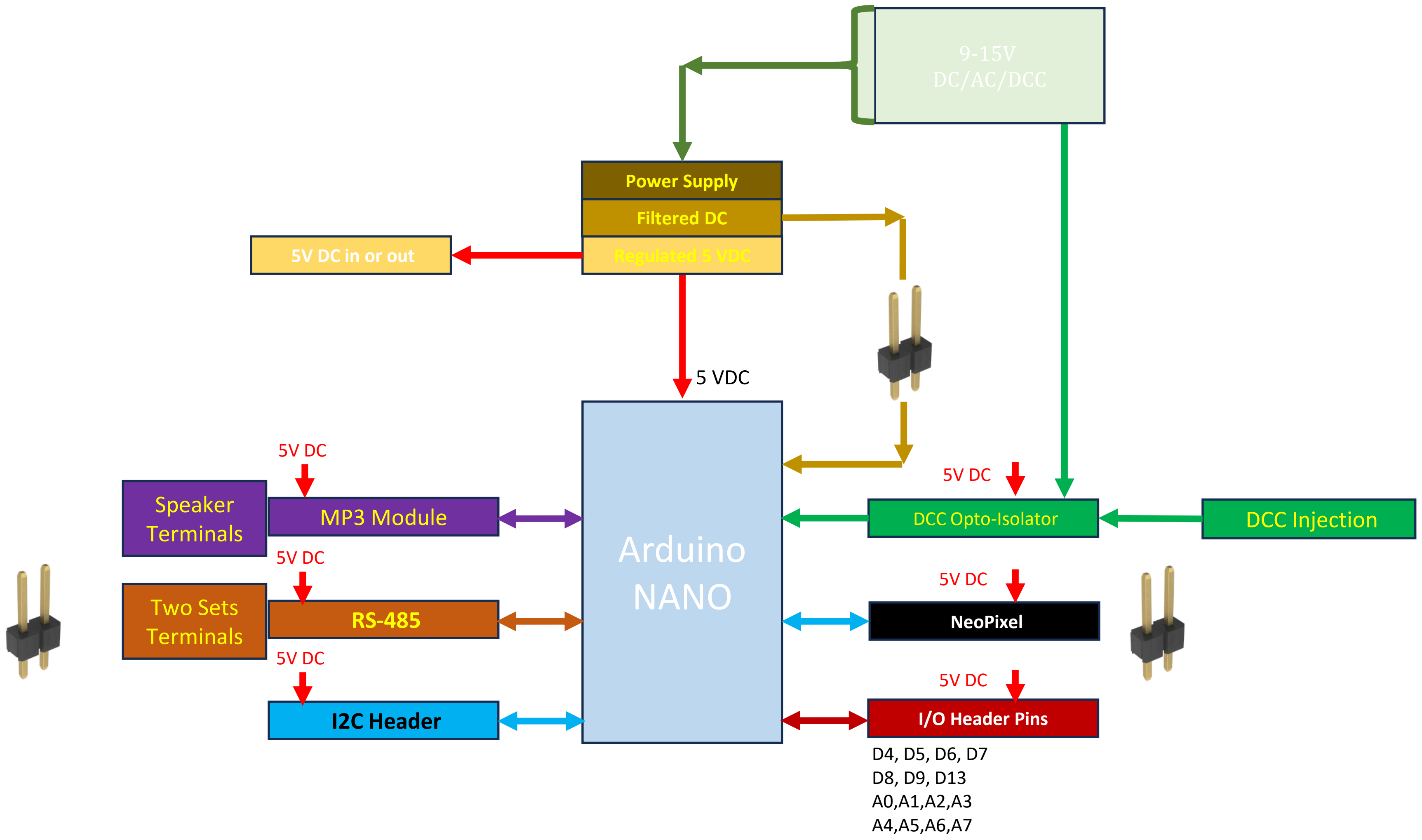
MaxDuino R1.4
Design Walkthrough

Note: In some cases photos of a previous revision are used where there is no material difference to the main message.

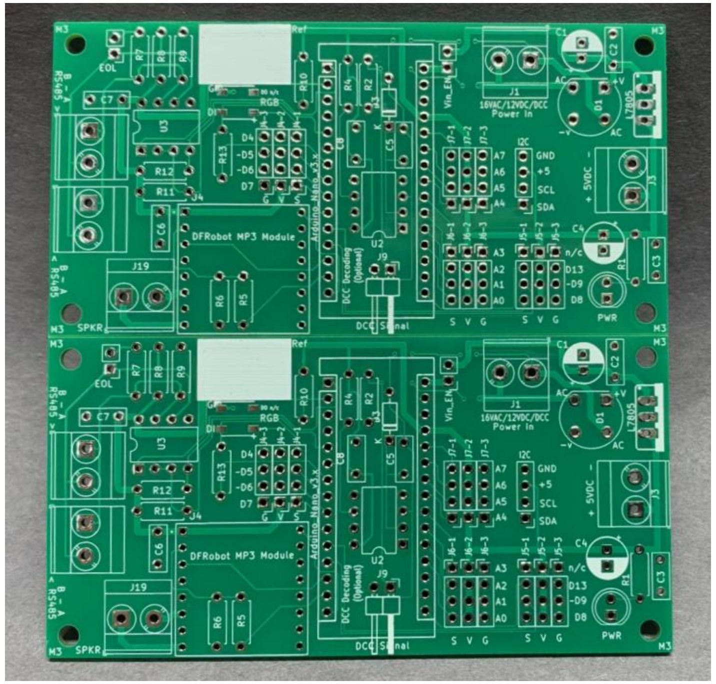
This Block Diagram Prepared with white background
for Journal Article



MaxDuino Block Diagram

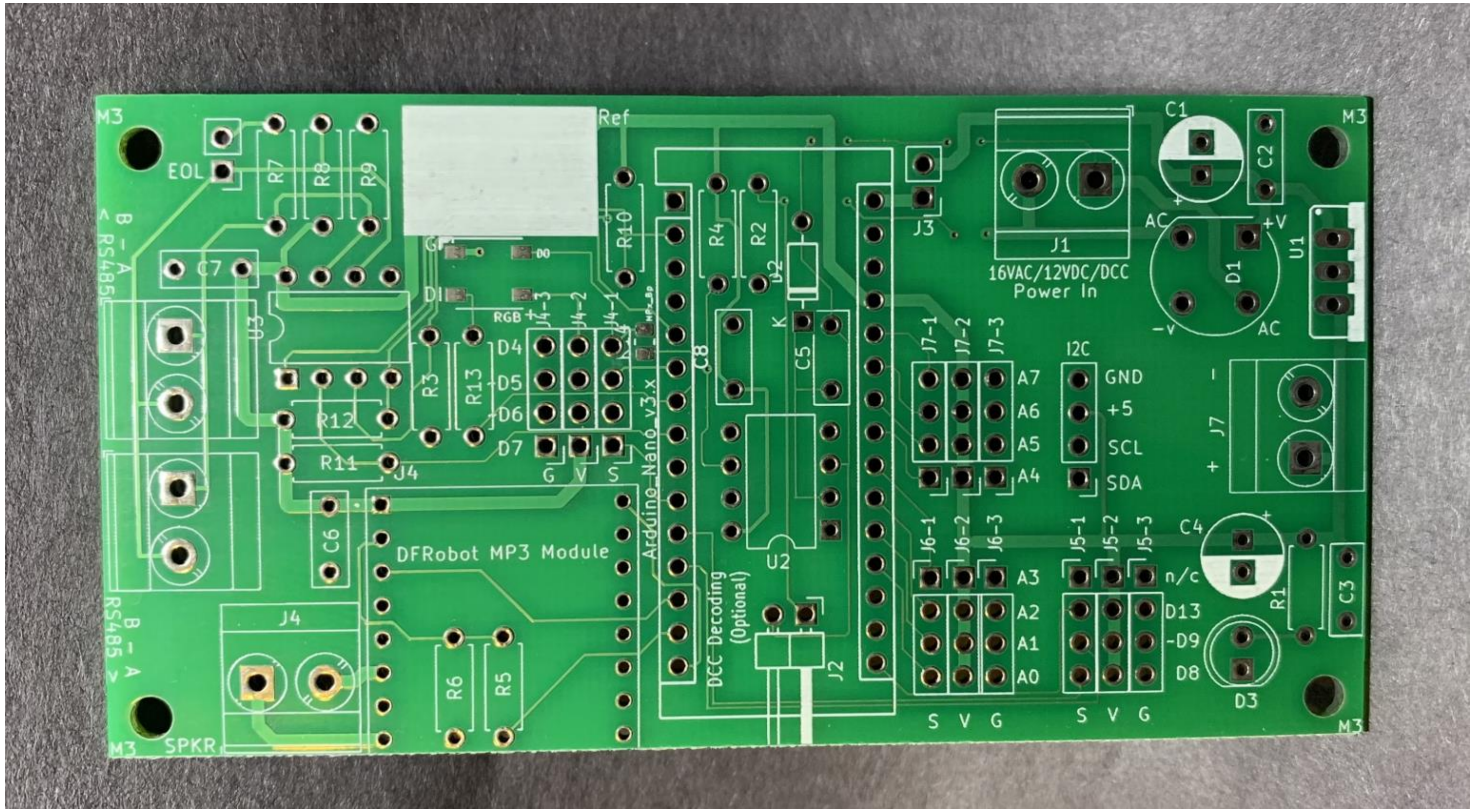
MaxDuino PCB

(General arrangement)

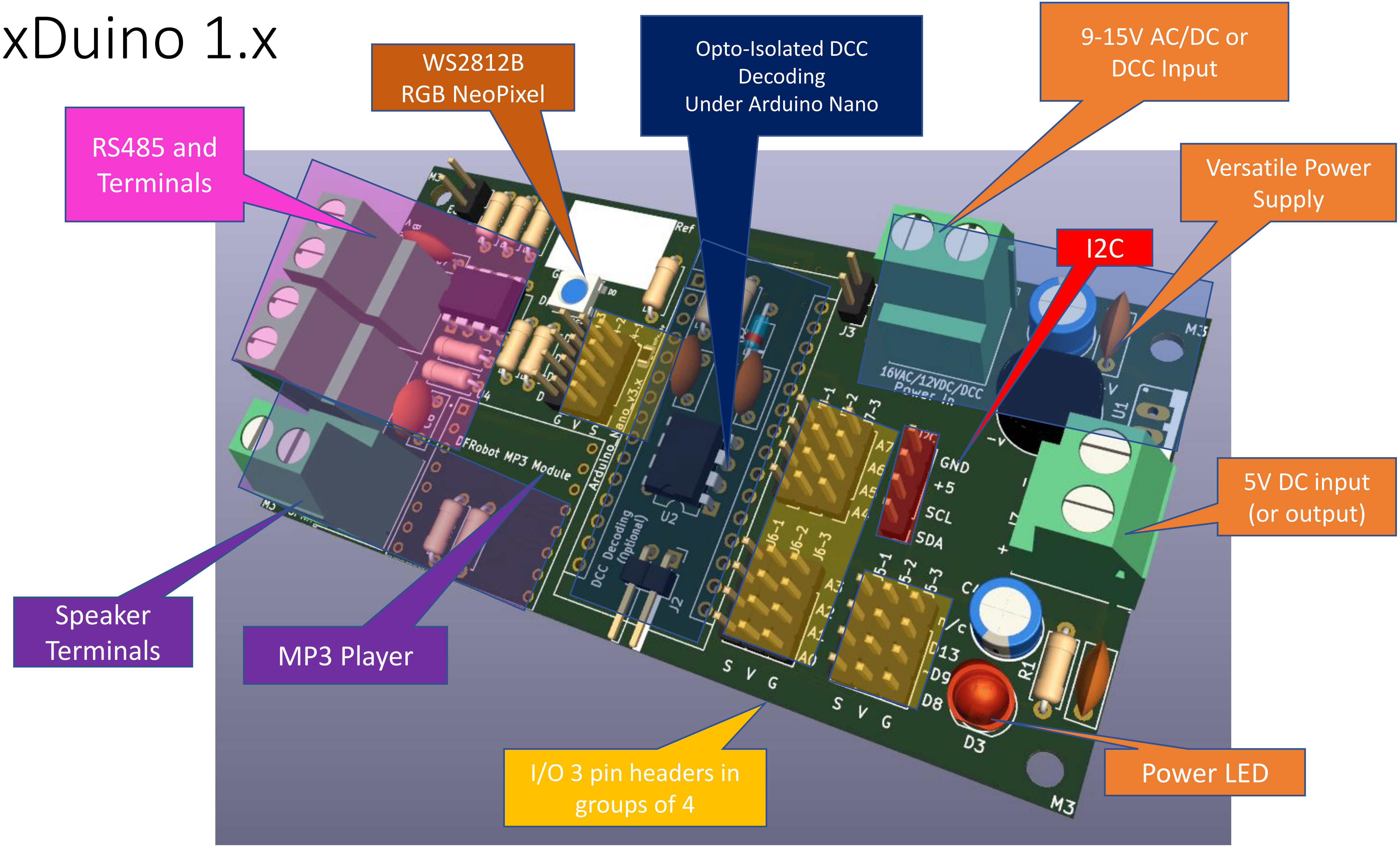


Two panels are on one PCB
under 100mm x 100mm

One MaxDuino.
Vee cuts in the PCB allow clean separation with minimal force.



MaxDuino 1.x



RS485 and Terminals

WS2812B RGB NeoPixel

Opto-Isolated DCC Decoding Under Arduino Nano

9-15V AC/DC or DCC Input

Versatile Power Supply

I2C

5V DC input (or output)

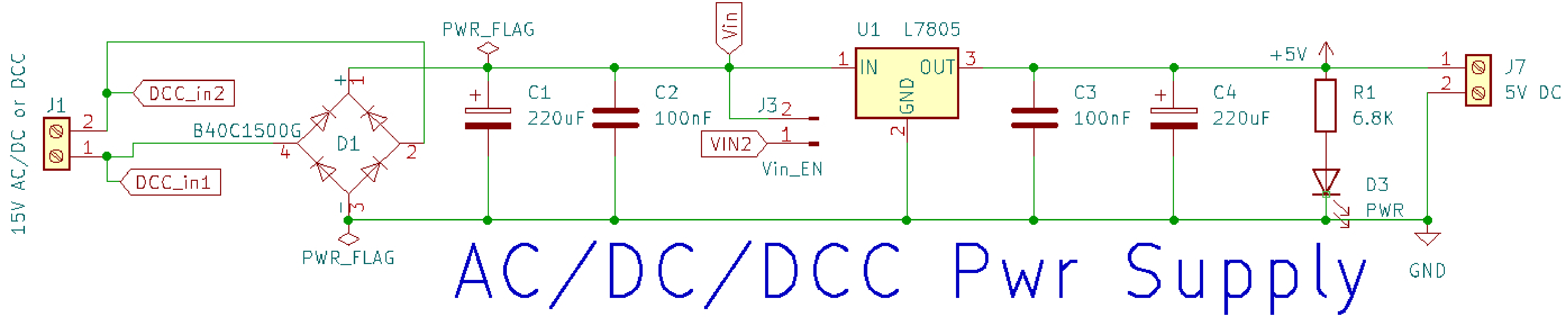
Speaker Terminals

MP3 Player

I/O 3 pin headers in groups of 4

Power LED

Design Review - Power Supply



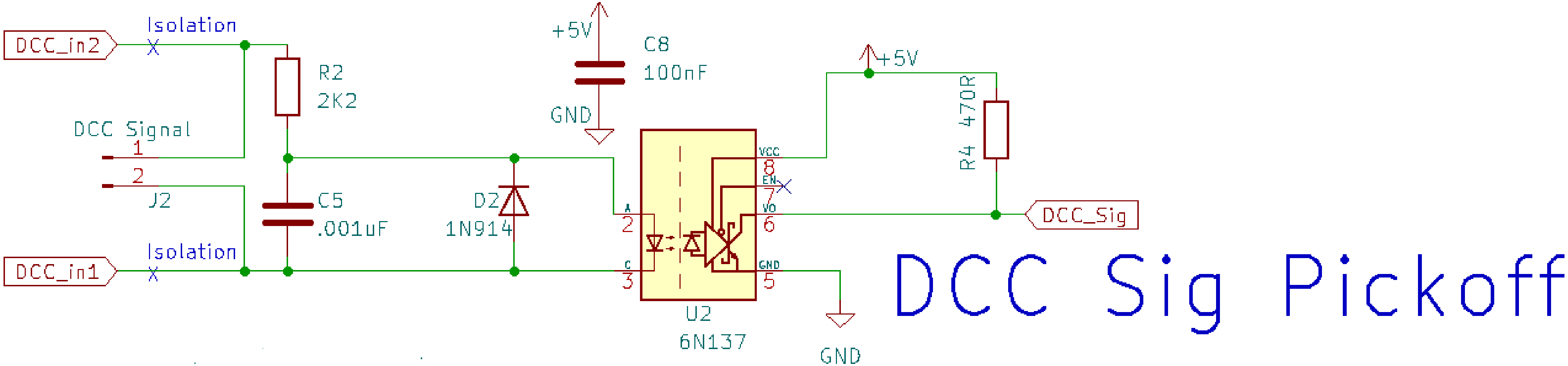
Power comes in from the left. Up to 9-15V AC/DC, or a DCC feed. It then goes through full wave bridge rectifier D1 and filter capacitors C1 and C2. This filtered DC voltage is now the input to the 5 volt regulator chip L7805.

The 5 volt regulator is capable of providing 1.5 amps and is internally protected against thermal overloads and short circuits. The 5 volt output is filtered with C3 and C4 to produce a steady 5 volt supply for significant projects. These capacitors are also needed to ensure stability of the voltage regulator. A LED is used to indicate the presence of 5 volts (even without any Arduino present.) On the right hand side is a second set of terminals which can be used as a 5 volt output to supply other circuits, or this can be used as a 5 volt input (in which case the L7805 regulator and everything to the left of it is not required.

Notes: If DCC is used as a power source the use of a heat sink on the L7805 is strongly recommended. Also if DCC is used as the power source that signal is also routed to the DCC optoisolator circuit.

The input to the voltage regulator chip Vin can be routed to the Vin pin of the Arduino by inserting a jumper at Vin_En. This puts the Arduinos own 5 volt regulator in parallel with the L7805. (You must verify Vin is 12VDC or less else the Arduinos on board regulator could be damaged).

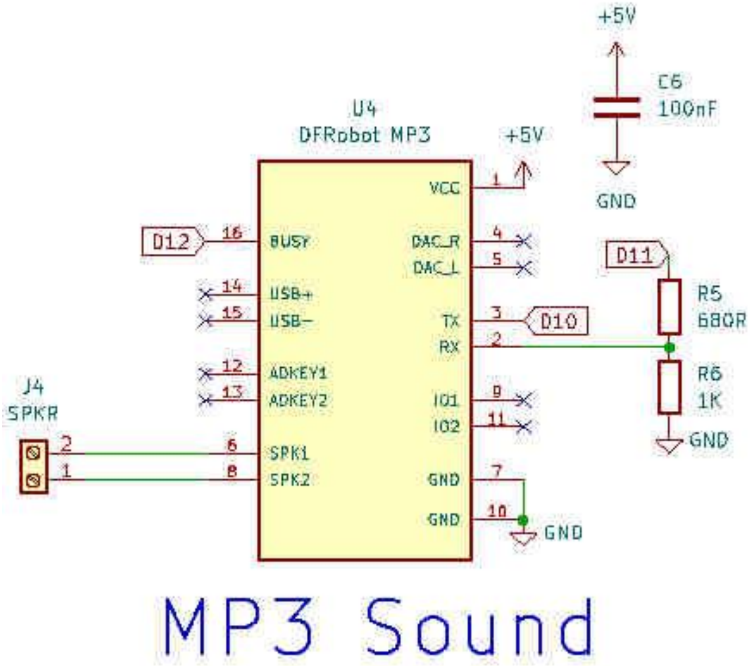
Design Review – DCC Decoder



The DCC signal from the power circuit comes in on the left as DCC_in1 and DCC_in2. These are routed through two isolation points which are thin traces and are clearly marked on the bottom of the MaxDuino PCB.
The DCC signal goes onto the 2K2 current limiting resistor R2 and then through the optoisolator U2. D2 provides reverse polarity protection for the optoisolator diode. C5 (in conjunction with R2) provide transient noise rejection. Pin 7 is not used as the optoisolator has an internal pulllup. The output of the optoisolator goes to Arduino an pin and is pulled high by R4. **Note the DCC input goes to Arduino Pin D3**. This is very important because this is only one of two pins on the Nano that can have interrupts attached which is important for time critical decoding.

NOTES: If DCC is not used as the MaxDuino power supply but you still want to decode dcc signals then the two DCC isolation traces must be made open circuit. Once these traces are opened then the MaxDuino power source and the DCC decoding are isolated from each other. An independent DCC signal for decoding can be input via J2 which is a pair of horizontal pins located on the PCB under the Arduino.

Design Review – MP3 Player



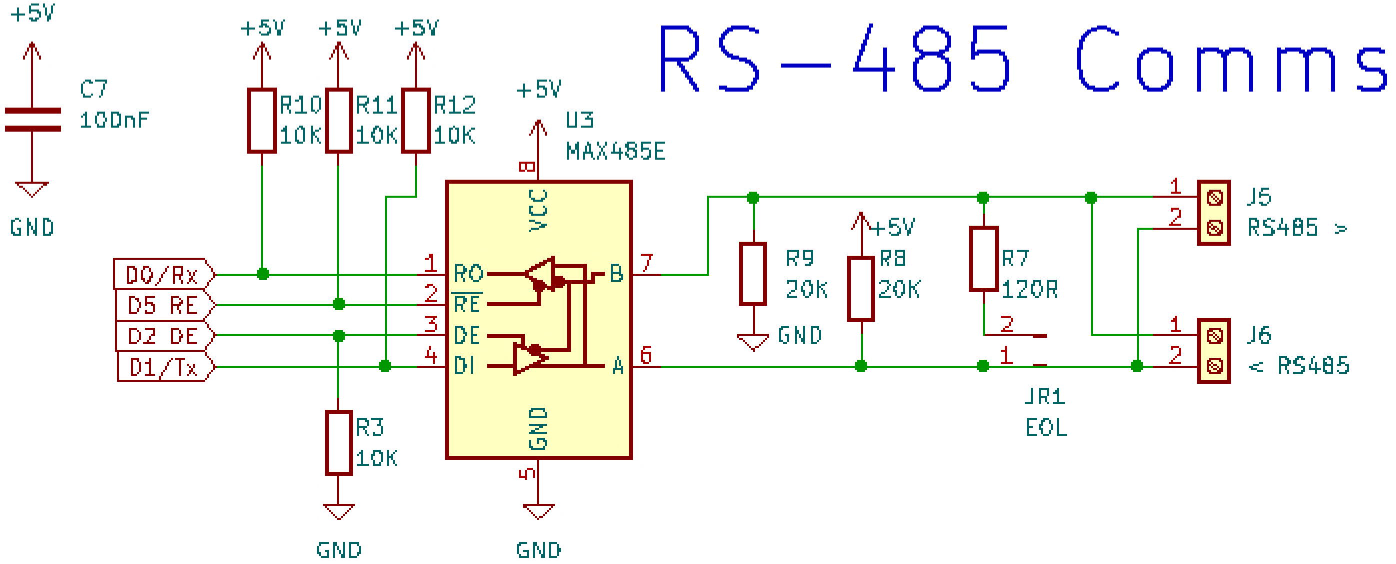
This circuit utilizes the common DFRobot MP3 player module. This module has an SD card reader and so it can play MP3's located on a memory card. These can be sound effects, station announcements, or any other sounds as required.
Any small 8ohm speaker can be connected to the speaker terminals. To avoid speaker damage the speaker should be rated at 3 watts or more.

The Tx pin from this module connects to D10 on the Arduino while the Rx pin is connected to D11. The use of software serial is required to implement this but as the transmission speed is a relatively slow 9600 bps this is not an issue. A pair of resistors are used to lower the voltage received by this module as it can glitch if the Rx voltage is even slightly over its 5 volt limit.
(Alternate design – omit R6, use 1K resistor for R5)

The busy pin is taken back to the Arduino on pin D12 to provide real time status information.
It is possible to 'ask' the module for its status via software also. Both approaches have been tested and work as expected.
The following library has been tested `#include <DFPlayerMini_Fast.h>`

Note: The **DFRobot MP3** player module is recommended. There is a similar module with part number MP3-TF-16P but it did not work well for me. Specifically It did not respond to some serial commands. (e.g.: setting volume)

Design Review



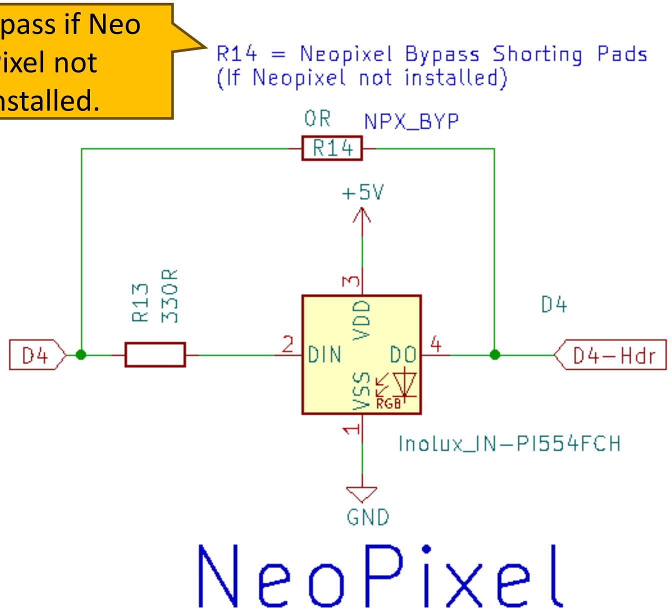
This circuit utilizes the common MAX485 chip to receive RS485 communications coming in from the right and convert them into standard logic level signals that the Arduino can read. There are two sets of terminals in parallel for the RS485 signals since this module is typically just one 'drop' and the RS485 signals will daisy chain onward. R7 is an EOL (End of Line) termination resistor. The EOL jumper at R7 should only be installed on one of the MaxDuino boards and preferably the one farthest from the sending device (Typically a USB/RS485 interface). This chip interfaces with the Arduino Tx/Rx pins to take advantage of the Arduino UART. As of **R1.3** both D2 **and** D5 set the direction of transmission. The use of pull up and pull down resistors on these means that in circuit serial programming can work without disrupting other RS485 communications on the wires. (In the instance of a large network for example).

The following libraries were used in testing

```
#include <Auto485.h>
#include <CMRI.h>
```

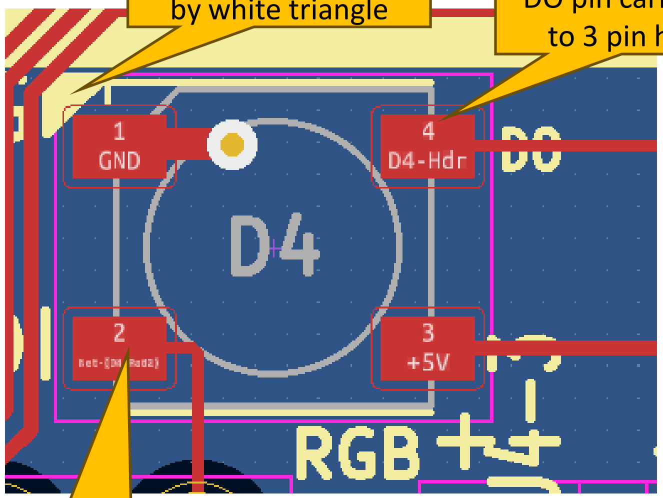
Design Review – NeoPixel

A bypass if Neo Pixel not installed.

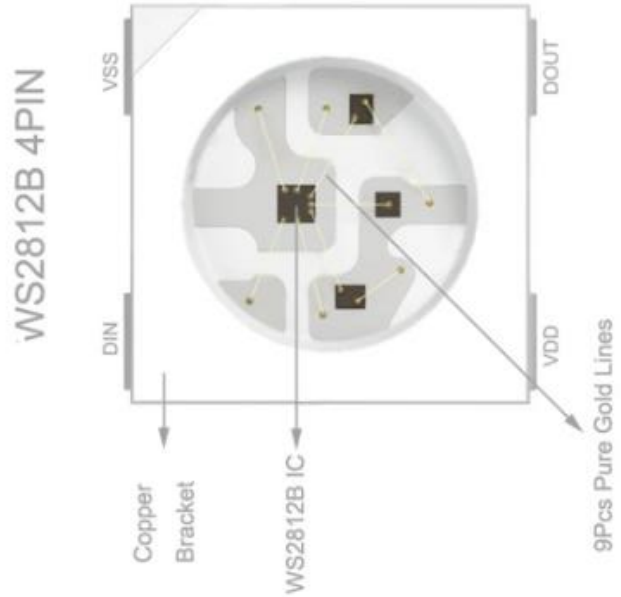


Ground pin denoted by white triangle

DO pin carries over to 3 pin header



DI pin from NANO pin D4 via 330R



Pinout for WS2812B NeoPixel chip.
Note **Vss = Ground**. **VDD** = **+ 5V**
The orientation shown matches the PCB

This small circuit utilizes the common addressable LED WS2812B NeoPixel chip to provide a wide range of colour feedback options. The circuit utilizes one Arduino pin (D4) and has an on board 330 ohm series resistor for the input pin. The three pin header with D4 is also available and suitable for longer strings of NeoPixel addressable LED's. Be sure to consider power requirements when using long (more than 10 elements) strings of NeoPixels. Note a bypass set of pads exist (R14 is specified as a SMD component nut is not normally installed – only the solder pads are used and can shorted if the NeoPixel is not installed. Doing so connects D4 directly to the header pin bypassing the onboard Neo Pixel. If the header pin D4 is used for additional NeoPixels the onboard NeoPixel is index 0, and the external NeoPixels start at index 1. The following library was used in testing `#include <FastLED.h>`.

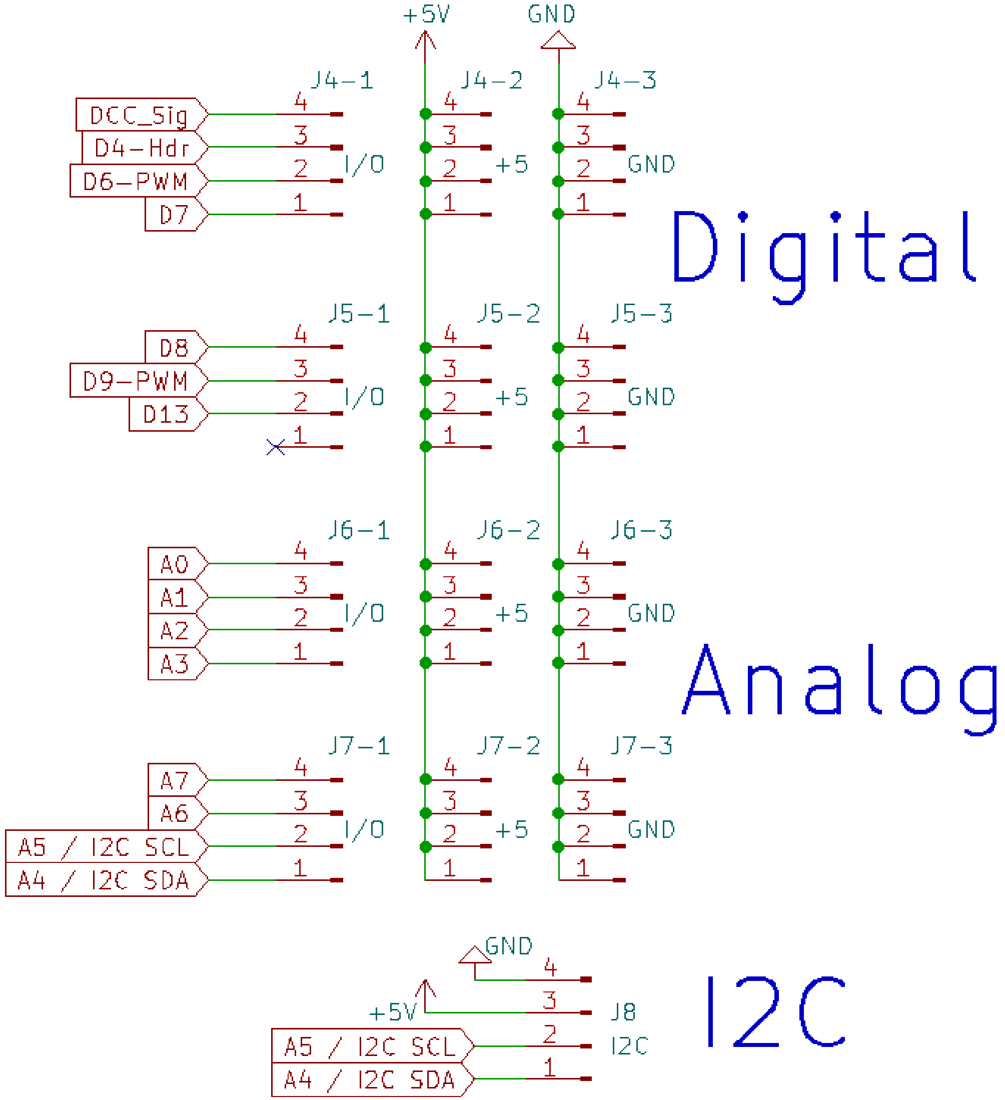
Design Review – Header Pins

The I/O pins are arranged as four groups of four.
Spaced to enables colour coded pins in strips of 4.
Each of the I/O pins is presented as a 3 pin header.
The pin order is Signal, +5V, Ground or S V G for short.
This pin order is a common but not universal
(so always check your modules)

On the PCB some pins are marked with a tilde (~).
This is indicating these pins are PWM enabled.

Notes: There are subtle differences in the arrangement of the header pins between PCB version 1.3 and 1.4
Version 1.4 is shown.

The DCC decoder circuit ties into **pin D3**.
It has a strong (470 ohm) pullup located on the PCB.
Pin D4 is normally the output of the on-board NeoPixel.
A4/A5 pins are also part of the I2C header.
The I2C header provides +5, Ground, SCL and SDA
(In the same order as commonly found on OLED displays.)
Be sure to check your pin order when using I2C.

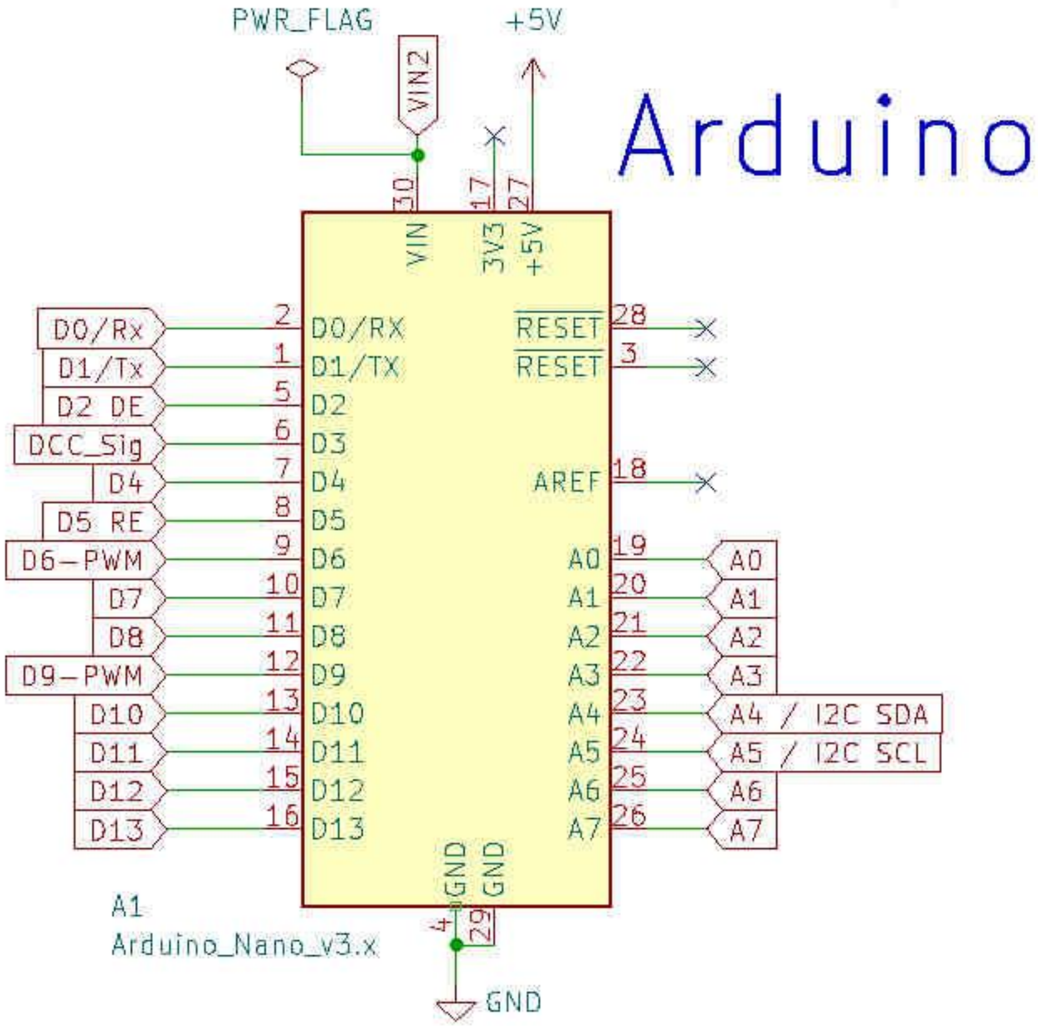


Design Review – Arduino

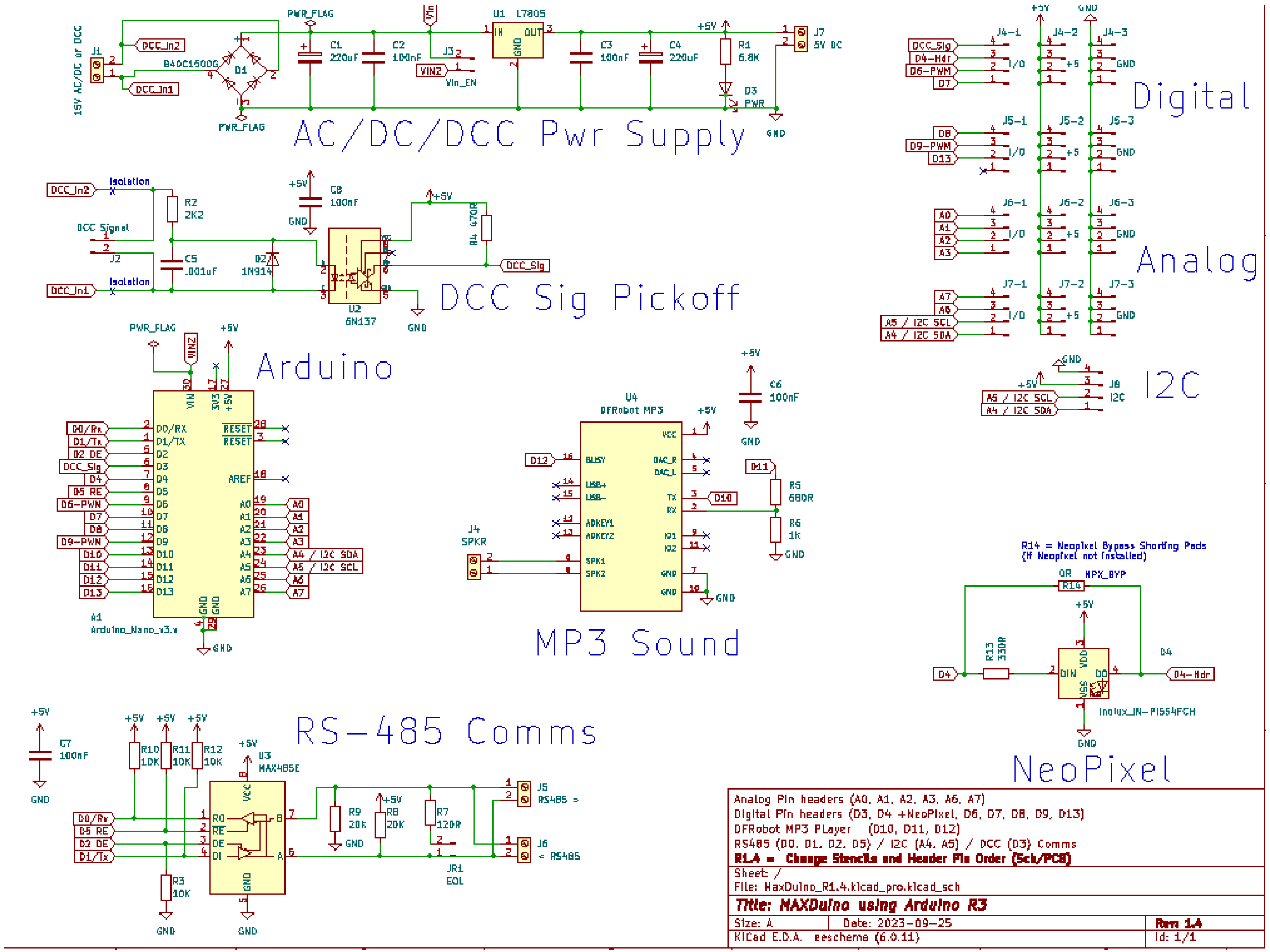
The Arduino NANO module is almost fully utilized as shown. Only 4 pins have no connections to them.

As of R1.3 both D2 and D5 are used for RS485 direction control.

It is recommended to install female header pins on the PCB so that the NANO module can be simply plugged into those header pins. This also provides ample clearance for the DCC decoder circuit located under the Arduino.



Design Review Full Schematic



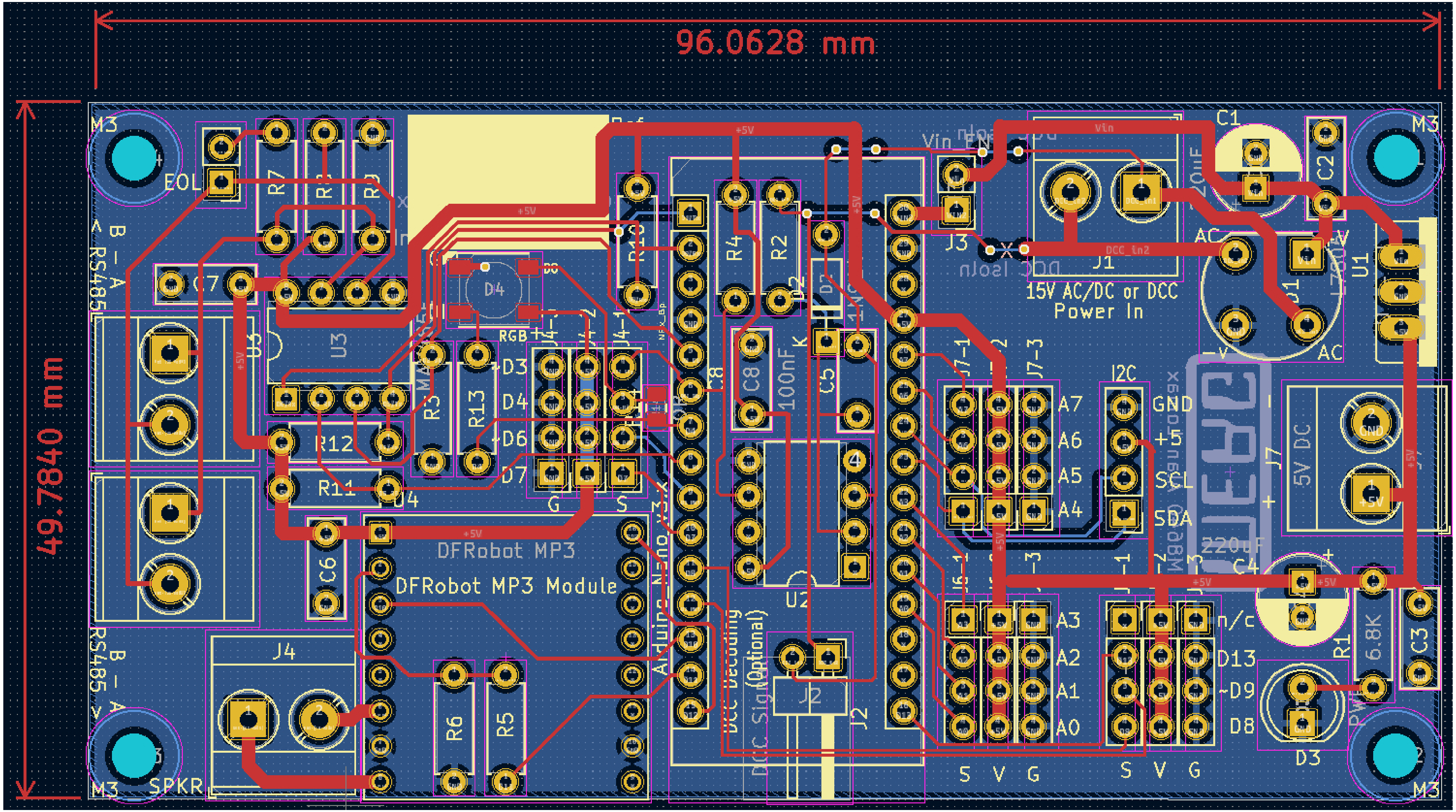
Analog Pin headers (A0, A1, A2, A3, A6, A7)
Digital Pin headers (D3, D4 +NeoPixel, D6, D7, D8, D9, D13)
DFRobot MP3 Player (D10, D11, D12)
RS485 (D0, D1, D2, D5) / I2C (A4, A5) / DCC (D3) Comms
R1.4 = Change Stencils and Header Pin Order (Sch/PCB)

Sheet: /
File: MaxDuino_R1.4.kicad_pro.kicad_sch

Title: MAXDuino using Arduino R3

Size: A	Date: 2023-09-25	**Rev: 1.4**
KiCad E.D.A. eeschema (6.0.11)		Id: 1/1

Design Review Full PCB



Management of Change

Rev 0 – Dec 2022 first 5 PCB’s (enough for 10 prototypes) ordered. Platform for developing test code.

Rev 1 – May 29 2023 second batch of 5 PCB’s ordered. Tried fixing DCC decoding, added 5 volt power input, added NeoPixel, changed 3 pin header and I2C header arrangement, Larger power traces, clarified stenciling. Found flaw in DCC Decoder section. One PCB (Two MaxDuino) were given to Ian for evaluation.

Rev 1.1 Revised DCC Decoder circuit. Additional stencil near RGB NeoPixel to aid with orientation, additional stencil for double isolation of DCC signal pickoff from power supply section.
Ordered two batches of 5 PCB’s June 29 2023 from JLCPCB. All testing passed.

Rev 1.2 Revised RS485 circuit to use 2 pins for enabling direction control. This puts both into high impedance state when doing ICSP. Also added flow through for the on board NeoPixel and ability to bypass the NeoPixel if not needed.

Rev 1.3 Previous revision changed the pin for DCC decoding – to one that does not support interrupts. This version moves DCC decoding back to pin D3 while retaining 2 pin direction control on RS485. All testing passed on October 14 2023.

Rev 1.4 During final prep of MERG article editor suggested minor changes that in turn suggested minor fixes to front and back stenciling. Circuit was electrically unchanged from version 1.3