A Fully-Configurable DCC Dual Power Breaker – Part 1

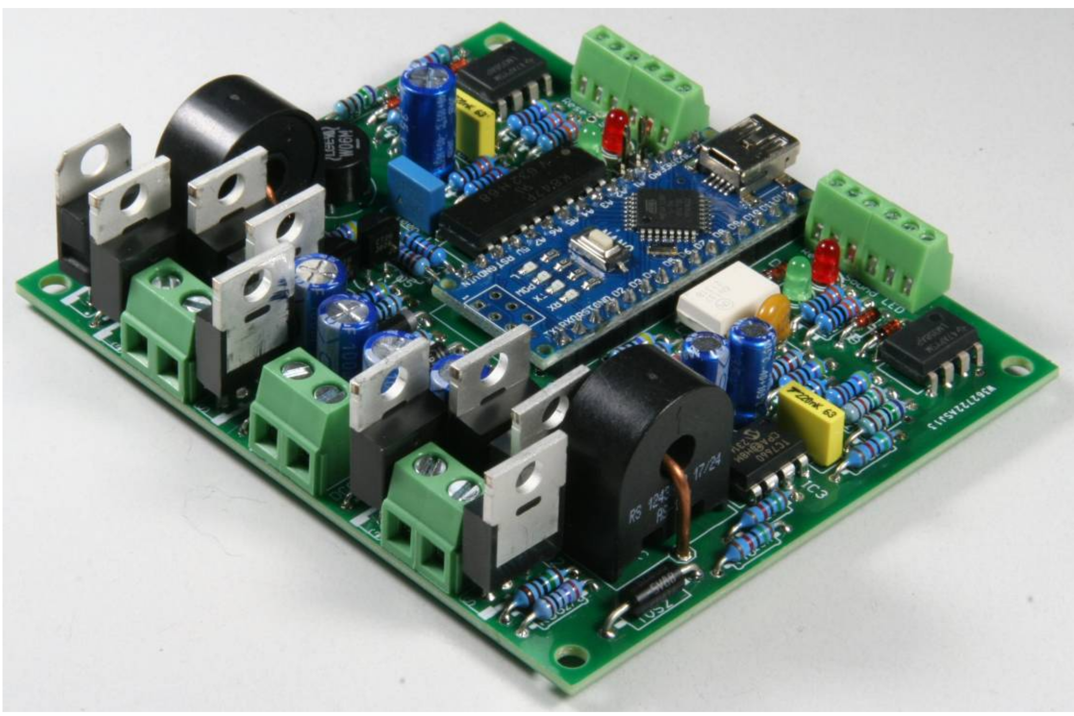
Introduction

Some time ago there was a fair amount of discussion in the MRH Forum on the subject of DCC power breakers, to protect the DCC command station and boosters from track short-circuits or other overloads, with everyone agreeing that they were a "good thing" to have.

However, I was a little taken aback at the relatively high cost of the devices, particularly if your layout is divided into a fair number of power districts - and I could also find almost no information on DIY versions or how these devices were put together.

A couple of basic designs were published by Mike Bolton of MERG around 20 years ago but only built as prototypes, and never proposed as finished designs. The designs were not easily configurable, other than by adjusting component values, and had a few other drawbacks such as not having the control circuitry isolated from the high-voltage DCC sections – an essential feature if the power breaker is to be installed as a reliable means of protecting your DCC system from potential damage.

Hence (as a professional electronics engineer) I decided to develop my own design, starting from Mike Bolton's ideas, and the result - a dual-channel power breaker which can be built for between $40 and $45, including the printed-circuit board - is shown below [1] –



All photographs and diagrams by the author

1. Assembled Dual Power Breaker

I believe the dual power breaker (DPB) design presented here does meet all of the necessary criteria, including the switching required to cut power to both rails – not just to a single rail. The relatively high-voltage components which control the DCC outputs to the track are optically isolated from the low-voltage control circuitry, and all aspects of the operation of each of the two channels are independently configurable using the included Arduino microcontroller.

I designed the DPB to use readily-available through-hole components only so that, provided you are capable of using a fine-tip soldering iron (and are comfortable assembling electronic components on to a PCB) you can easily buy the components and build it. The PCB details are available as a set of Gerber files so that it can be procured from a number of suppliers. The only particular PCB requirement is that it needs to be built using 2oz copper (70 micron), rather than the standard 1oz (35 micron) copper, to handle DCC currents up to 5 amps.

The fitted Arduino Nano microcontroller controls the DPB operational parameters. You can set the parameters independently for each of the two channels. These include setting the current limit for each channel (power district), the delay time after the district current exceeds the set limit before power is cut off, and the subsequent delay time before the DPB tries to reconnect power to that district.

After power is cut to either (or both) of the two districts the DPB will attempt to reapply power automatically after the set delay. If an overload is still present, the DPB will cut off power again, and will continue to cycle through this sequence until you remove the cause of the track overload (or you cut off power to the DCC command station or booster at source).

You can disable the auto-reconnect feature if you wish, and connect a manual pushbutton to either or both channels of the DPB to reapply power to the layout. The manual pushbutton can also be used to override a long reconnect delay and switch track power on again.

Additionally, you can connect an external LED and alarm sounder to each channel to notify you when and where a power break occurs – since you will not neccessarily mount the DPB where it will be visible during layout operation.

As well as acting as a straightforward power breaker, you can also configure the DPB as an automatic reverse loop controller. In autoreverser mode both channels are used to control the DCC phase applied to the reversing loop or wye, while still providing fully-programmable over-current protection for the district represented by the loop or wye.

Specification

Maximum DCC Current through DPB (total of both districts) – 5 amps
(Note – the PCB design will handle up to 8 amps, but will get warm)

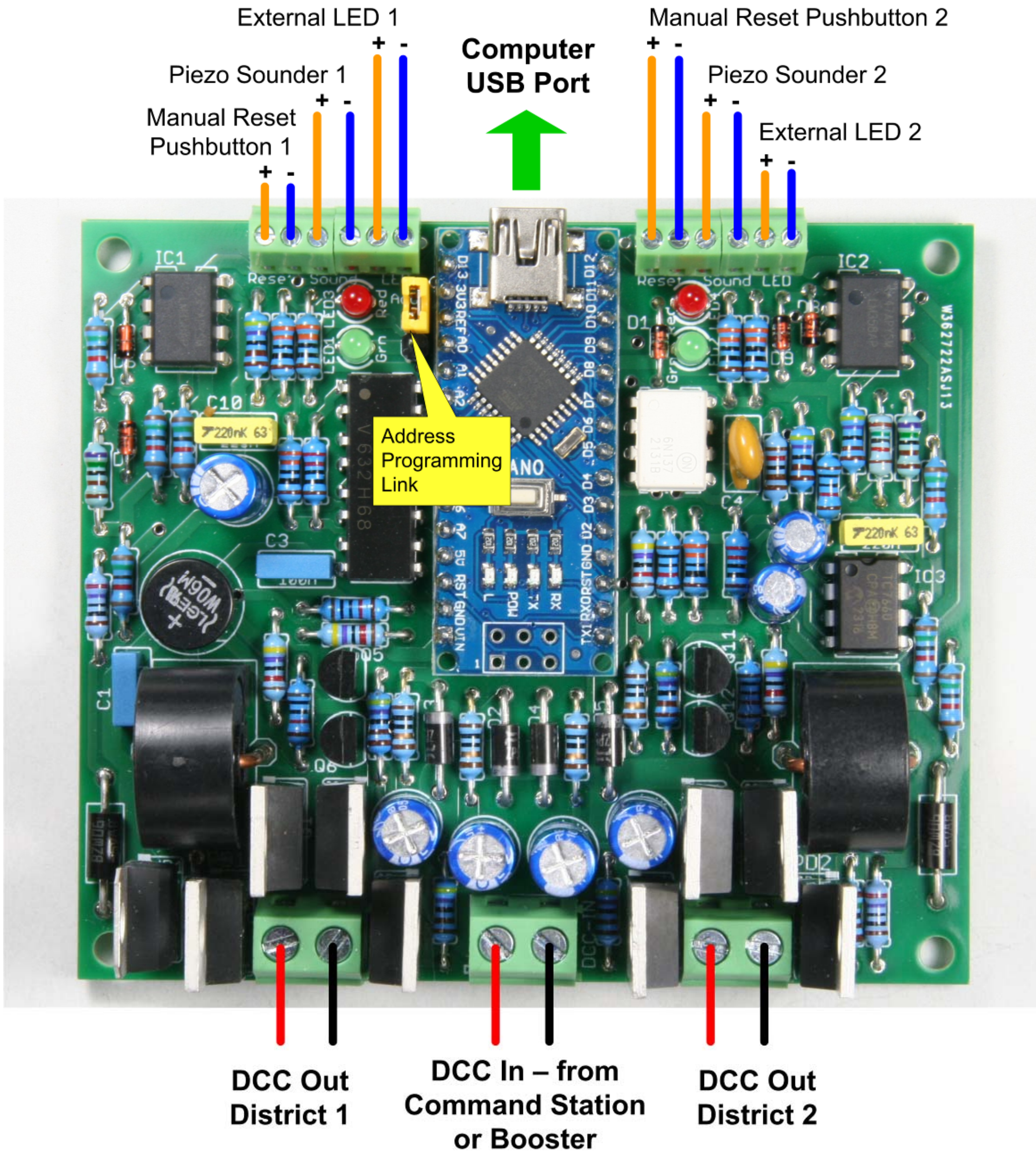
District Trip Current – settable in 0.25 amp steps from 0.25A to 5A

District Overload Delay Time – settable in 1 msec steps from 5ms to 255ms

District Reconnect Delay Time – settable in 0.25 sec steps from 0.25s to 60s

Connections

Connections to the DPB are quite simple, as shown in [2].

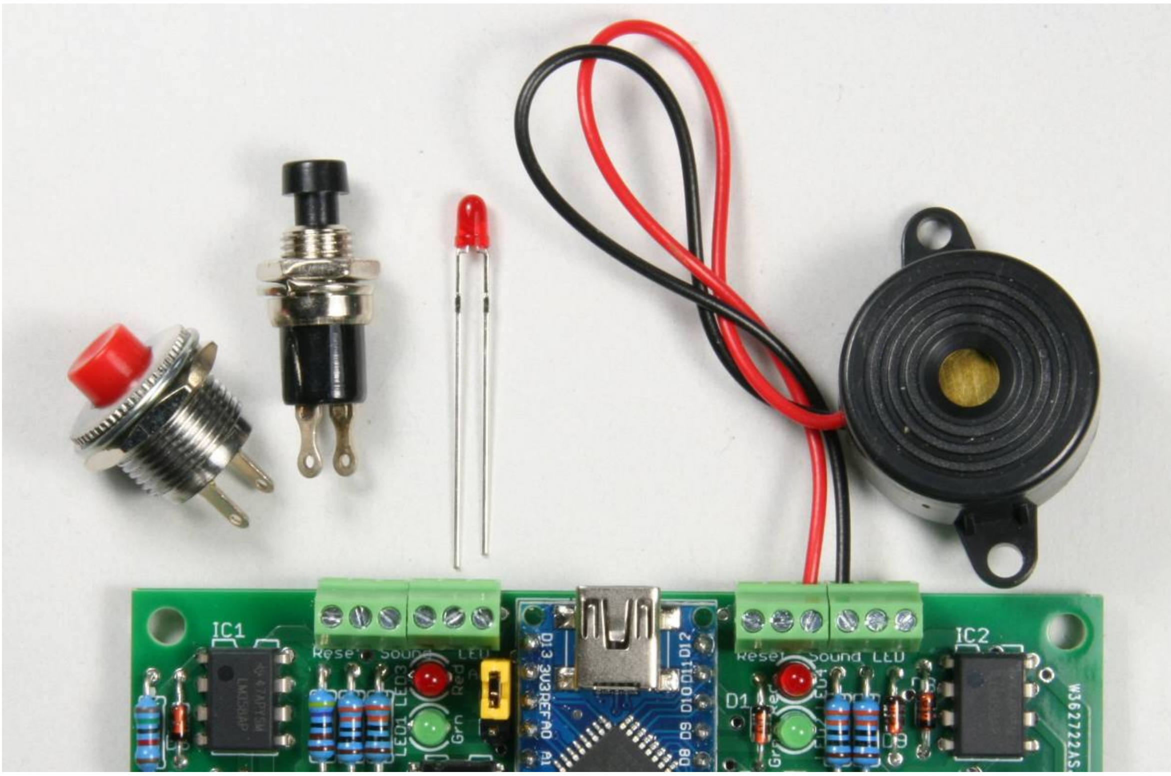


2. Dual Power Breaker connections

The DCC track output from your command station or booster is taken to the central 2-way terminal block, and you connect one of your layout's power districts to either of the other two 2-way terminal blocks. The Arduino Nano has a standard USB (Micro-B) connector to link it to your computer.

The DPB provides auxiliary connections for each district to an external LED, alarm sounder, and manual reset pushbutton via a 6-way header or terminal block, should you have a need for these optional features. Note that the blue (-) connections in [2] are all linked to the DPB control circuitry ground (0 volt). You do not need to make any connection from these ground connections to the ground connection (if any) of your command station or booster. The DCC connections are completely isolated from the DPB ground.

Some examples of these auxiliary components are pictured in [3] –



3. Reset pushbuttons, external LED and piezo sounder

The reset pushbuttons should be momentary, push-to-make types such as you can find on Amazon (https://a.co/d/5XRBpI3 or https://a.co/d/72tZtje), where you can also find suitable piezo sounders (https://a.co/d/1hLcV8g or https://a.co/d/0VL5x3q). Note that the current consumption of any sounder you attach must not exceed 20mA – which rules out most of the electromagnetic types.

The external LEDs should be either red, orange, yellow or green (not white or blue) and fitted as shown in [3] with the shorter lead to the terminal block ground (shown blue in [2]). Series resistors for both external LEDs are already fitted on the DPB and provide a current of around 10mA to each LED.

Setup and Control

Although you will need to use your computer to load the operational software code (a 'sketch') to the Arduino Nano, once this done a connection to the computer is not essential. The DPB is DCC-compatible and you can set all parameters using any standard handheld controller.

Your first step is to set a DCC address for the DPB. Connect the DPB to the DCC command station or booster track output, then fit a jumper across the 2-pin header indicated on the PCB in [2] (where the jumper is shown in the inactive or 'parked' position on one of the link pins). With the jumper fitted, select an accessory address on your handheld controller, and send a turnout command (Normal or Route, straight or diverging). The DPB will then set itself to the selected accessory address (in the standard range 1 – 2043).

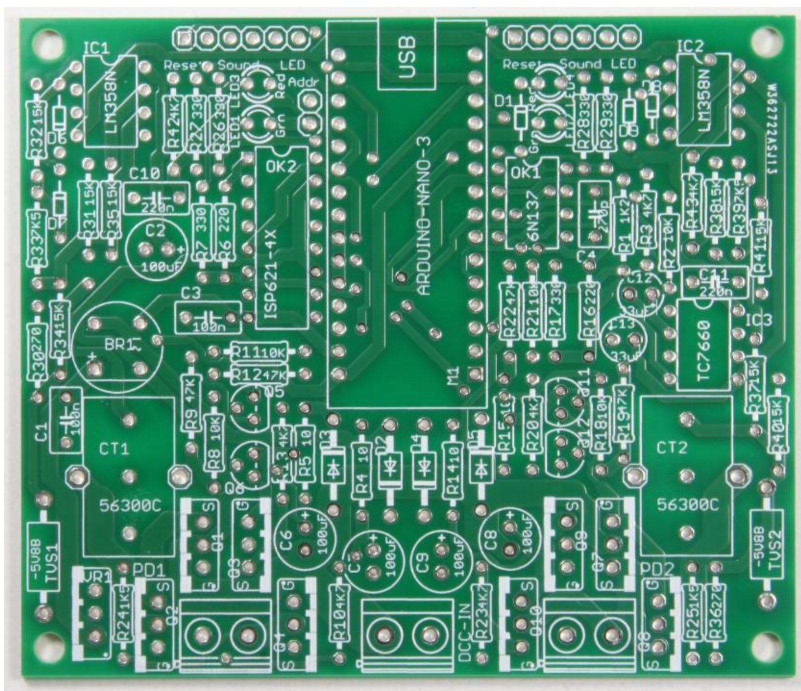
DPB parameters are held in a set of configuration variables (CV41 through CV56). Once you have programmed the DPB address, you can then set the value of each CV by issuing a program-on-the-main (POM) command from your handheld controller using the set accessory address this time as a locomotive address.

However, without adding extra complexity to the DPB design so that you could connect the DPB to your programming track, there is no convenient way via DCC of verifying that the CV values have been set as required.

The Arduino sketch, therefore, includes a facility to let you read and write all CV values using the Arduino Nano USB connection to your computer in conjunction with the Arduino IDE Serial Monitor and a set of simple commands. These commands and their use in setting up the DPB will be described in detail in Part 2 of this article.

Interested in building your own DPB ?

For anyone who would like to build their own Dual Power Breaker, full constructional details will be covered in Part 2. The first step, however, is to purchase a printed circuit board. This is a double-sided, through-hole-plated PCB, size 87mm x 74mm (3.4” x 2.9”) as shown below [4] –



4. Bare printed-circuit board

For most of my previous projects I have used OSH Park, a small company located in Lake Oswego, Oregon. Unfortunately, they only have the capacity to offer heavyweight 2oz (70 micron) copper on thin (0.8mm) boards, rather than on the standard, more rigid, 1.6mm thick boards which are needed to support the heavier components used on the DPB.

Hence, this time, I used my alternative supplier, PCBWay which, like OSH Park, allows me to share the uploaded PCB design. PCBWay supply boards in multiples of five so, if you don’t intend to build that many DPB units, you might consider sharing the cost with perhaps a neighbouring model railroader, or a member of your club. To order a set of boards, follow this link –

www.pcbway.com/project/shareproject/W362722ASJ13_DualPowerBreaker_3C_109e00d0.html .

On the PCBWay website, if you then want to order a set of PCBs, click the button labelled “Add to cart”. Alternatively, if you would like to use your own preferred PCB supplier, you can download the full set of Gerber files by clicking on the “Download Gerber file” link. This gets you a .zip file which you can pass on directly to your supplier, but remember to specify also that the PCB must be manufactured with 2oz (70 micron) copper rather than the normal 1oz (35 micron) copper in order to handle the high currents involved.

Returning to the PCBWay order webpage, after clicking the “Add to cart” button you will then need to login, if you already have an account with PCBWay, or register for an account with your e-mail address and a password of your choice. Once logged in you will have to click “Add to cart” again to open the submission form where you can, if you wish, increase the number of boards to order by clicking in the Quantity textbox and selecting the required number (in multiples of five). Note that, although the initial set of five boards will cost you $36.10, each additional set of five will only add about $5 to the total amount, which substantially reduces the cost per board. On the form you can also change the colour of the solder mask or silkscreen printing to suit your own preferences but be aware that some choices will substantially increase the production cost shown on the form. I suggest, however, that you leave the other options just as they are.

Once you are happy with the order, click the “Submit” button. There will be a short delay while PCBWay checks the board parameters for manufacture. Once this is done, you can click the “Proceed to checkout” button. Check that your name and address are correct and select a payment method. You can pay either with a credit card or via PayPal, but note that PCBWay will probably add a small fee (to cover bank charges) depending on the chosen payment option.

PCBWay will also automatically select the fastest and most expensive shipping option (usually DHL or FedEx). I generally opt for the much cheaper Global Standard Shipping which has always proved reliable, with delivery taking about a week to the UK.

Using PayPal and this shipping option, the total cost for five boards (at the time of writing this article) came to $45.15 ($9.03 per board) or, for ten boards, to $48.00 ($4.80 per board).

When you are happy with the details, click the “Place order” button and wait for your PCBs to arrive. PCBWay lets you track the process of your order on their website through all the manufacturing stages and, once complete, will send you a tracking link to follow the progress of your delivery.

Please note that neither A-Train Systems nor myself have any connection with PCBWay other than as a very satisfied customer of their services.

Ordering components

The parts required to build a complete Dual Power Breaker, together with sources of supply, are listed in the two tables, [5] and [6], below.

Part	Reference	Quantity	Value	Notes
Resistor – Metal Film, 0.25 Watt	R4, R5, R14, R15	4	10R	All resistors 10mm pitch on PCB
" "	R6, R16	2	220R	
" "	R30, R36	2	270R	Must be 1% tolerance
" "	R7, R17, R26–	6	330R	

Part	Reference	Quantity	Value	Notes
	R29			
" "	R1	1	1K2	
" "	R24, R25	2	1K5	
" "	R3, R10, R13, R20, R23, R42, R43	7	4K7	
" "	R33, R39	2	7K5	Must be 1% tolerance
" "	R2, R8, R11, R18, R21	5	10K	
" "	R31, R32, R34, R35, R37, R38, R40, R41	8	15K	Must be 1% tolerance
" "	R9, R12, R19, R22	4	47K	
Capacitor – Polyester Film	C1, C3	2	100nF 63V	5mm pitch
" "	C10, C11	2	220nF 63V	5mm pitch
Capacitor – Electrolytic	C12, C13	2	33uF 35V	2mm pitch, 5mm diameter
" "	C2, C6–C9	5	100uF 35V	2.5mm pitch, 6.3mm diameter
Capacitor – Disc Ceramic	C4	1	270pF	5mm pitch
Diode Bridge Rectifier	BR1	1	W01G – W06G	Any voltage rating > 50V acceptable
Diode	D1, D6–D9	5	1N4148	
Diode	D2-D5	4	UF4002	
Transistor PNP	Q5, Q6, Q11, Q12	4	2N3906	
Power MOSFET N-Channel	Q1-Q4, Q7-Q10	8	IRFZ44N or IRFZ44Z	
Transient Voltage Suppressor - Bidirectional	TVS1, TVS2	2	BZW06-5V8B, BZW04-5V8B, or P6KE6.8CA	Must be bidirectional (B or CA suffix)
Voltage Regulator - 9 Volt	VR1	1	L7809CV or LM7809	
Voltage Converter	IC3	1	TC7660	
Dual Operational Amplifier	IC1, IC2	2	LM358N, LM358AN, LM358P, or LM358AP	
Optocoupler	OK1	1	6N137	
Quad Optocoupler	OK2	1	ISP621-4X, LTV-847, ACPL-847, or K847PH	
Arduino Module	M1	1	Nano-3	
Current Transformer 300:1	CT1, CT2	2	Murata 56300C or Talema AS103	
Light-Emitting Diode	LED1, LED2	2	3mm Green	
Light-Emitting Diode	LED3, LED4	2	3mm Red	

Part	Reference	Quantity	Value	Notes
Terminal Block – 2-Way	DCC In, PD1 Out, PD2 Out	3		5.08mm (0.2”) pitch
Pin Header	JP1, JP2	1	1 x 6	2.54mm (0.1") pitch
"	JP3	0	1 x 2	2.54mm (0.1") pitch
Jumper Link Open - 0.1" pitch	-	1		
Terminal Block – 3- or 6-Way 2.54mm (0.1") pitch	-	4 or 2		Optional – in place of Pin Headers JP1, JP2

5. Table 1 – Dual Power Breaker – List of components

Suggested suppliers for the parts listed above are RS Components (uk.rs-online.com/web/) or Farnell (uk.farnell.com/) for users in the UK, or Mouser (www.mouser.com/) or Digikey (www.digikey.com/) for users in the USA. Digikey’s prices tend to be a little higher than those from Mouser. Both Mouser and Digikey also have European-based operations, but tend to have higher prices than RS Components or Farnell.

The Arduino Nano-3 module can best be obtained from one of the many suppliers operating on eBay, although you will have to make your own judgement as regards who will give you the best service, based on their feedback from previous customers. An alternative is to source the module directly from China, using a website such as AliExpress, where you will generally benefit from substantially lower prices, but will again have to make a judgement on the reliability of a particular supplier based on their published customer feedback.

Table 2 [6] below gives suggested part numbers for each DPB component from each of the listed suppliers.

Reference	RS Cmps	Farnell	Mouser	Digikey
R4, R5, R14, R15	1997587	3496945	603-MFR25SFTE52-10R	13-MFR-25FTE52-10RCT-ND
R6, R16	1997610	3496953	603-MFR25SFTF52-220R	13-MFR25SFTF52-220RCT-ND
R30, R36	1997917	3496955	603-MFR25SFTF52-270R	13-MFR-25FTE52-270RCT-ND
R7, R17, R26–R29	1997662	3496925	603-MFR25SFTE52-330R	13-MFR-25FTE52-330RCT-ND
R1	1997604	4312074	603-MFR-25FBF52-1K2	13-MFR-25FRF52-1K2TR-ND
R24, R25	1997606	3496914	603-MFR-25FBF52-1K5	13-MFR-25FRF52-1K5TR-ND
R3, R10, R13, R20, R23, R42, R43	199762	3951808	603-MFR-25FBF52-4K7	13-MFR-25FRF52-4K7TR-ND
R33, R39	1997939	3952712	603-MFR-25FBF52-7K5	13-MFR-25FRF52-7K5TR-ND
R2, R8, R11, R18, R21	1997658	3951796	603-MFR-25FBF52-10K	13-MFR-25FRF52-10KTR-ND
R31, R32, R34, R35, R37, R38, R40, R41	1997597	3496907	603-MFR-25FBF52-15K	13-MFR-25FRF52-15KTR-ND

R9, R12, R19, R22	1997669	3496933	603-MFR-25FBF52-47K	13-MFR-25FRF52-47KTR-ND
C1, C3	2416523	2763238	80-R82DC3100DQ50J	495-75656-1-ND
C10, C11	2416543	4143525	80-R82DC3220SH60J	399-9705-1-ND
C12, C13	2286739	9693653	710-860010572004	732-8734-1-ND
C2, C6–C9	7270688	2917910	667-ECA-1VM101	399-6104-ND
C4	2621878	3651154	594-S271K43SL0N6TK5R	445-CC45SL3AD271JYVNACT-ND
BR1	7082668	3775028	625-2W04G-E4	4878-B250R-ND
D1, D6–D9	7390290	2675146	512-1N4148TA	1N4148FSTR-ND
D2-D5	6289732	2677309	637-UF4002	UF4002-E3/54GITR-ND
Q5, Q6, Q11, Q12	7390375	1574372	637-2N3906	2N3906FS-ND
Q1-Q4, Q7-Q10	5409777	8650225	942-IRFZ44NPBF	IRFZ44NPBF-ND
TVS1, TVS2	6997981	2889054	511-BZW04-5V8B	497-12758-1-ND
VR1	7140666	4472964	863-MC7809CTG	497-14786-5-ND
IC3	2070297	9762680	579-TC7660CPA	TC7660CPA-ND
IC1, IC2	7092100	3117069	595-LM358AP	296-9554-5-ND
OK1	8051267	2453244	859-6N137M	160-1791-ND
OK2	6935952	1339043	859-LTV-847	160-1370-5-ND
M1 - Arduino Nano	eBay	eBay	eBay	eBay
CT1, CT2	1243880	1454868	580-56300C	419-AS-103-ND
LED1, LED2	1808502	4538243	604-WP710A10GD	754-1217-ND
LED3, LED4	1780909	3648335	604-WP710A10ID	754-WP710A10SURDK-ND
DCC In, PD1 Out, PD2 Out	1468345	4179098	571-2828372	5239-ELK508V-02P-ND
JP1, JP2, JP3	2518632	1593422	517-929834-01-24-RK	3M9457-24-ND
Jumper Link (0.1")	2518682	3226076	855-M7583-46	S9337-ND
Terminal Blocks (3- or 6-Way 0.1")	1731629	3817463	571-2828343 or 571-282834-6	277-1277-ND

6. Table 2 – Dual Power Breaker – Component supplier references

Notes :

1. You may be able to source equivalent parts locally at a lower cost, using the details available for each suggested part in the table above – search the supplier's website using the part number to view the key parameters you are looking to match (assuming that you have sufficient electronics knowledge to understand the specifications). Although eBay is a very good source for components, especially for small quantities of passive components, headers, connectors or hardware, you need to be wary of very low cost parts, since these are often of low quality or may be manufacturers' substandard rejects.
2. The total cost of parts for a single DPB (when this article was written) should be between $40 and $45 (£40 to £45 in the UK), including the PCB, but could be much more if all components are ordered in small quantities from the major suppliers.
3. Buying electronics components singly or in small quantities is much more expensive than buying in bulk (in quantities of 10 or more), so it is well worth considering carefully at the outset how many DPB units you might build, possibly jointly with friends, and then procuring all of the required components in a single purchase. This will also minimise any shipping charges.

Dual Power Breaker technical details

For those who are interested, you can download a technical description of the DPB circuitry, including a full schematic, from a special download section of my A-Train Systems website at www.a-train-systems.co.uk/dpb-download.

The folder there also includes a copy of the Gerber files for the printed-circuit board if you prefer not to visit the PCBWay website. If you are ordering a PCB from your preferred supplier then you will have to specify that the board must be manufactured using 2oz (70 micron) copper to handle the high currents involved. This parameter is not included as part of the Gerber files data.

Software for the Dual Power Breaker

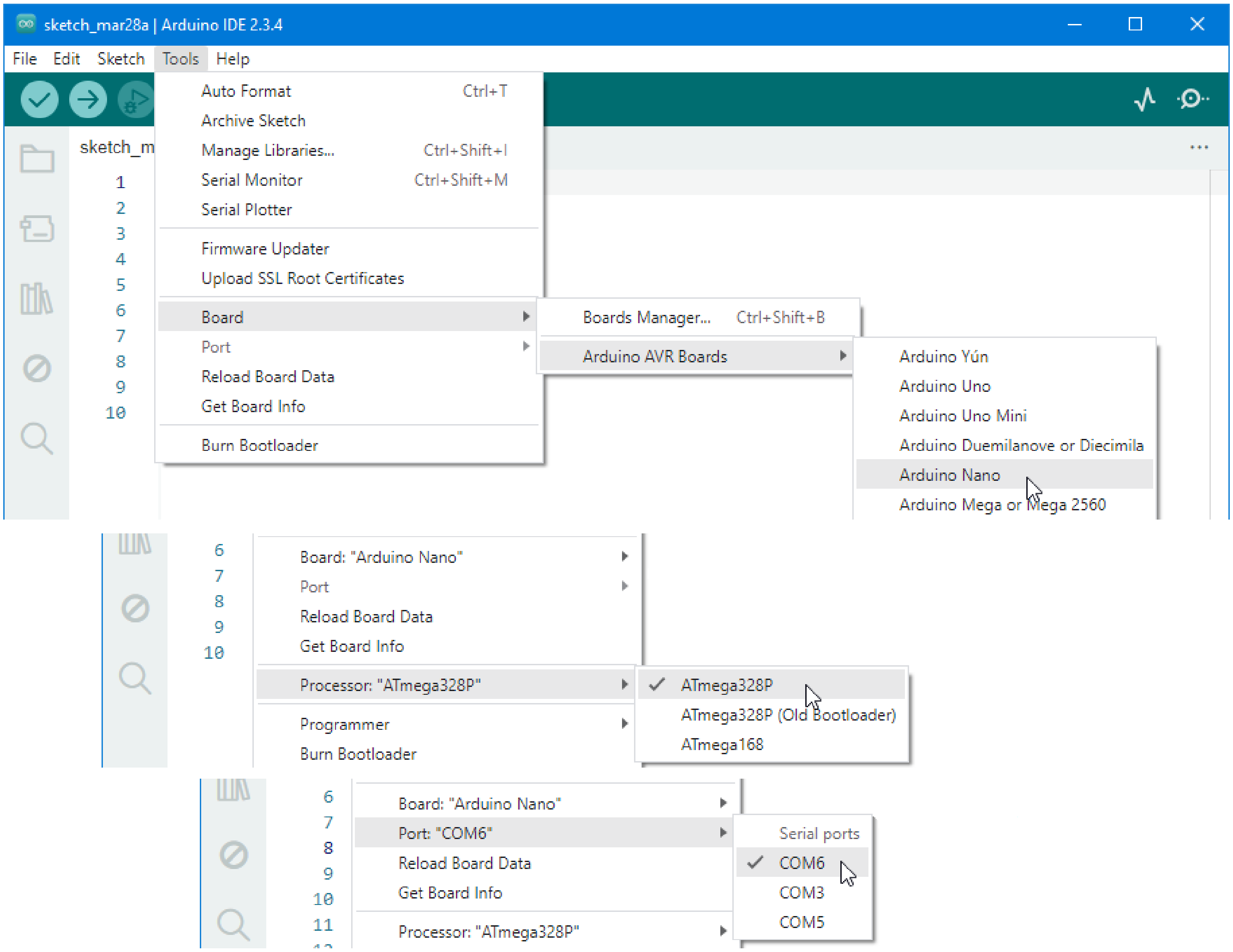
For anyone who is not familiar with the Arduino hardware and software which form the basis for the control section of this project, the article by Joe Fugate in the April 2024 issue of *MRH* “Using Servos for Turnout Control – Part 1” (https://online.fliphtml5.com/buups/bvxr/index.html#p=57) has a section which provides a very good introduction.

There is also a lot of useful information on the Arduino website but it can prove difficult to find exactly what you need. Have a look at https://docs.arduino.cc/hardware/nano/#features for details of the Nano module used here in the DPB, including links to tutorials such as https://docs.arduino.cc/tutorials/nano/nano-getting-started/.

Once you have the Arduino IDE installed and set up on your computer, and you have mastered the basics by working through some of the examples supplied, you can have a look at the latest version of the sketch for the DPB (DualPowerBreaker_3-8.ino) by downloading it from my A-Train Systems website (www.a-train-systems.co.uk/dpb-download) to any convenient folder on your computer, and then opening it in the Arduino IDE.

Assuming that you have an Arduino Nano-3 module to hand, connect a Mini-B USB cable from your computer to the Nano USB socket. Next, if you have not already done so, you need to set the correct board type for the sketch by opening the Tools menu and selecting ‘Board’, followed by ‘Arduino AVR Boards’ and finally ‘Arduino Nano’, as shown in [8].

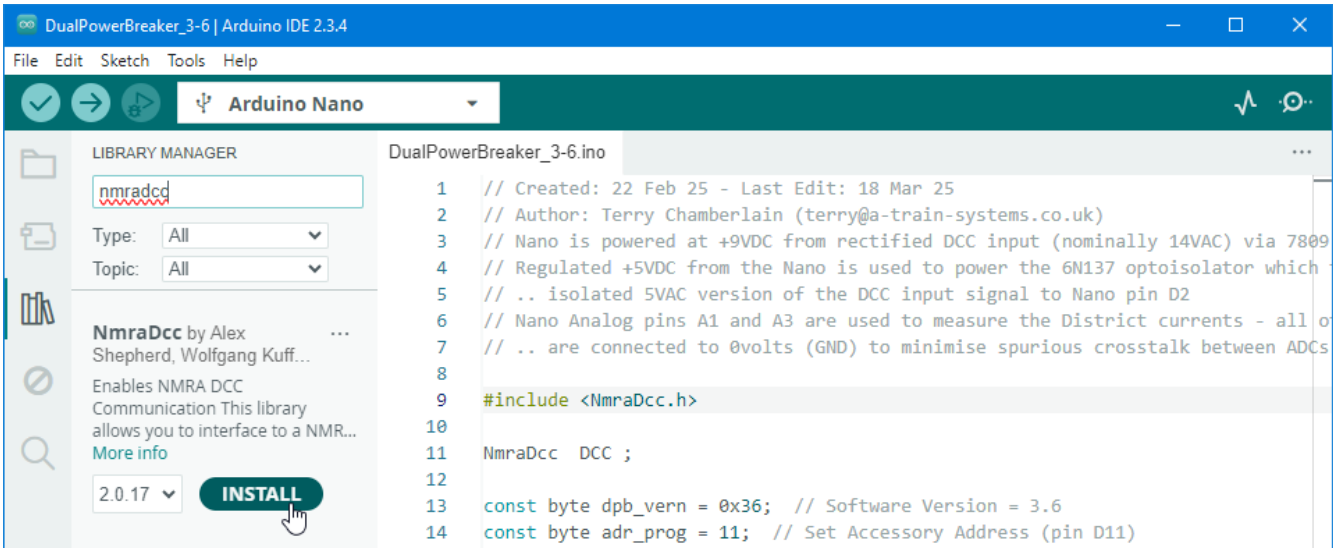
With the board type selected as Nano, the next step is to select the type of processor fitted to the module, followed by the COM port used by your USB connection to the Nano – again as shown in [7] –



7. Select Arduino Nano, followed by Processor and Port

If you have any difficulty finding the right Port, a useful tip is to disconnect the USB cable and look at the list of available ports (if any) displayed when you click Port on the Tools menu. Now close the Tools menu, plug the USB cable back in, then look at the Port list again – there should now be one extra port shown which is, therefore, the one to use.

The next step is to load an additional library, “NmraDcc” to the IDE. Click on the Libraries icon on the left side of the IDE window then type ‘nmradcc’ in the search box and press the Return key, as shown in [8], where the correct library should appear at the top of the list –



8. Using Library Manager to load the NmraDcc library

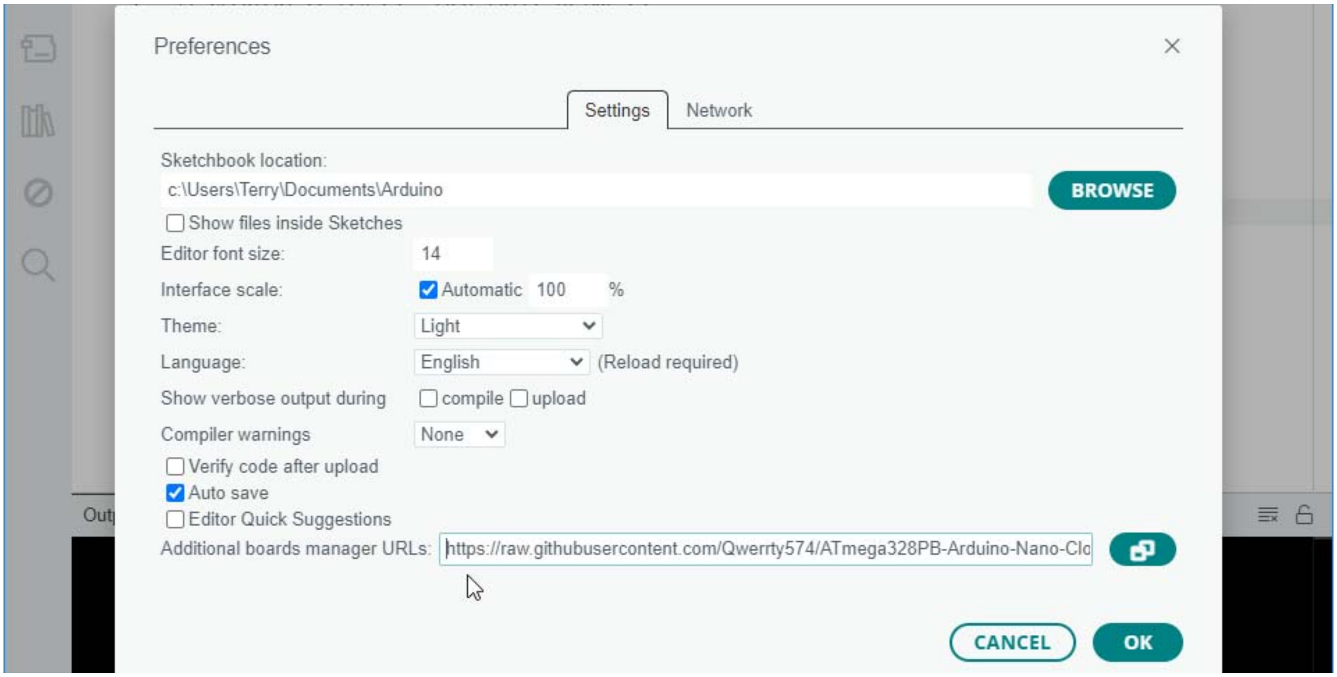
Click the Install button and wait a few moments until installation is complete, when you can click the Libraries icon again to close the Library Manager.

Now click the Upload () button to transfer the compiled sketch to the module. You should see the Tx and Rx LEDs on the Nano light for varying lengths of time as the transfer proceeds. Once the sketch has been uploaded, the Nano will retain all of the code even when it is switched off.

If the transfer fails, it may be that you have an older version of the Nano module. In this case, go back to the Tools menu and select the Processor as 'ATmega328P(Old Bootloader)' and try the Upload operation again.

Should the transfer still fail, have a careful look at the markings on the Nano processor (the large chip on the module). Several batches of the Nano I received recently were fitted with an ATmega328PB processor which may not be recognised by the Arduino IDE as correct for a Nano module. While uploads with one batch worked by using the 'ATmega328P(Old Bootloader)', for the others I needed to load an additional board type into the IDE. If you have an ATmega328PB module then try the following enhancement to the Arduino IDE.

Open the File menu and click on Preferences to display the Preferences window as shown in [9] –

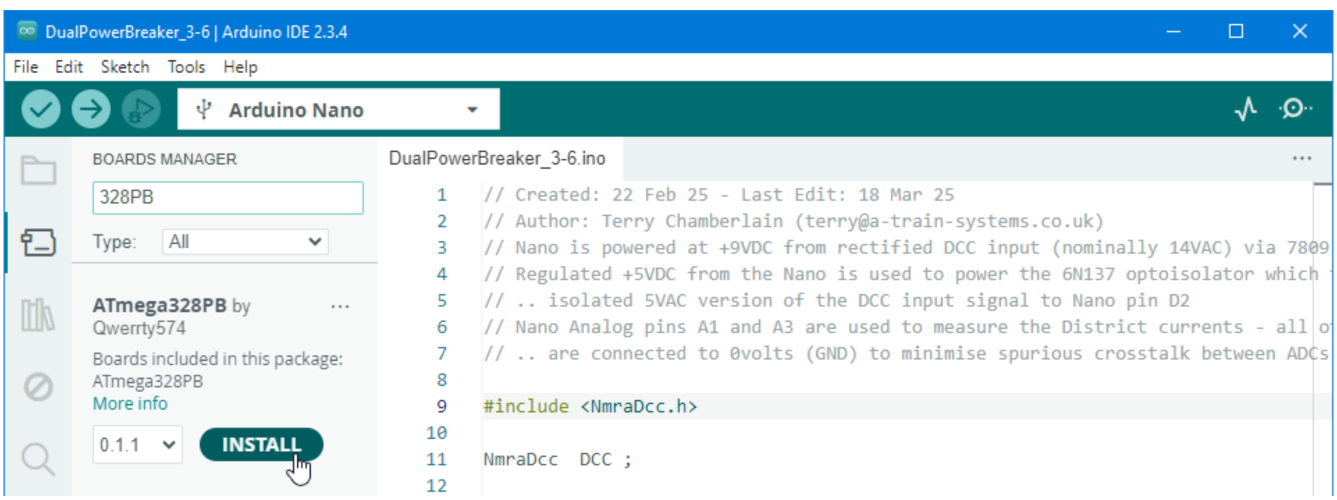


9. Adding a board type for the ATmega328PB processor

At the bottom of the window, copy and paste the following text into the box labelled ‘Additional boards manager URLs’ (without any spaces) –

https://raw.githubusercontent.com/Qwerrty574/ATmega328PB-Arduino-Nano-Clone-with-Bootloader/master/package_m328pb_index.json

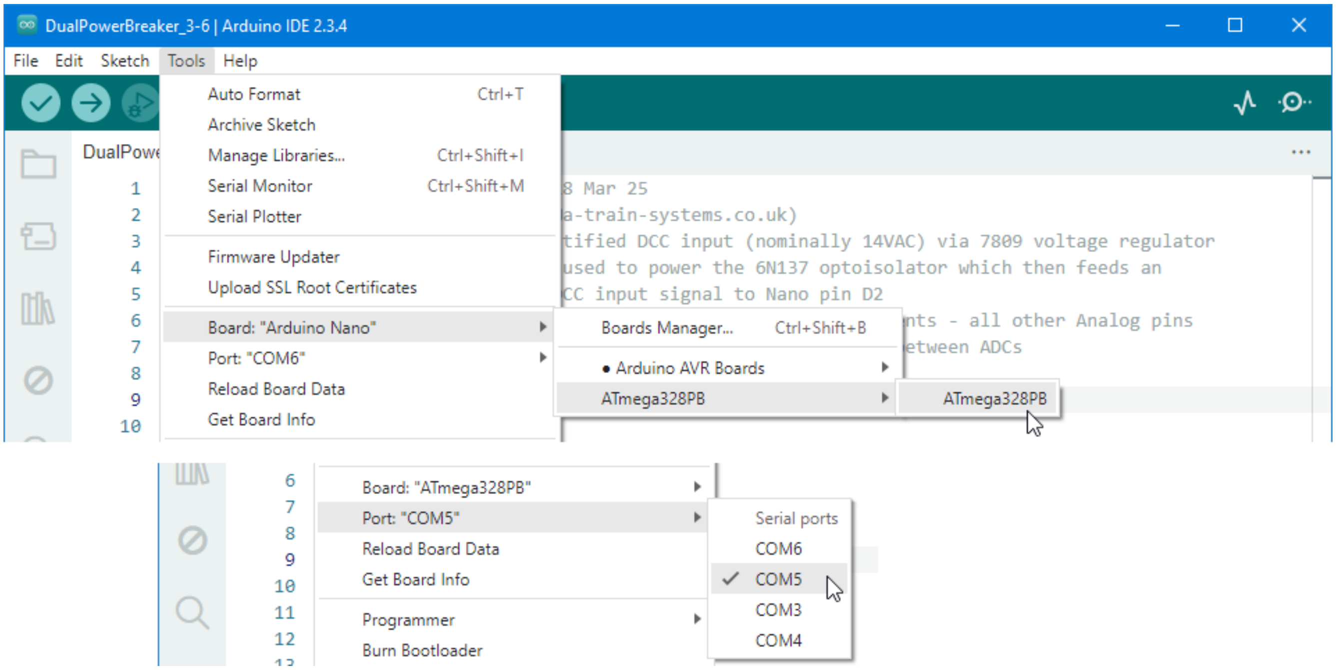
– then click OK. Complete the addition of the new board type by clicking on the Boards icon on the left side of the IDE window, then type ‘328PB’ in the search box and press the Return key, as shown in [10], where the correct board package should appear at the top of the list –



10. Installing the ATmega328PB processor board type

Click ‘Install’ to complete the addition of the board package before clicking the Boards icon again to close the Boards Manager.

Ensure that the Nano is still connected to a USB port of your computer then return to the Tools menu and this time select the Board type as ATmega328PB as shown in [11]. Check that the correct Port is still selected (just in case the allocation has changed – as in [11]), then try uploading the sketch once again – hopefully with success this time.



11. Select ATmega328PB, and check for correct Port

If you find that uploading the sketch still fails, it may be that the bootloader software in your Nano has been corrupted and needs to be reloaded. Fixing such a problem is a bit outside the scope of this article, but have a look at the Arduino Support article at –

https://support.arduino.cc/hc/en-us/articles/4841602539164-Burn-the-bootloader-on-UNO-Mega-and-classic-Nano-using-another-Arduino

- and documentation on one of the Arduino built-in examples at –

https://docs.arduino.cc/built-in-examples/arduino-isp/ArduinoISP/

- which give you very straightforward instructions on how to re-program the bootloader.

Part 2 of this article will cover the details of building your own Dual Power Breaker, together with all aspects of the associated software used to set-up and operate one or more Dual Power Breakers fitted to your own layout, including its use as an automatic reverser unit.

Dr Terry Chamberlain

Terry Chamberlain got into model railroading almost by accident in the 1990s when he responded to a request from some modellers in California to build a DCC system based around an Atari personal computer – and he had to build a simple layout to prove that it all worked. Eventually the project evolved into A-Track, a Windows application to provide full computer support for the complete range of NCE DCC systems, with facilities similar to JMRI's Decoder Pro and Panel Pro.

Terry is a professional electronics engineer and spent most of his career in the UK defence industry designing, and managing the development of, large real-time computer systems for the Royal Navy. Now that he is fully retired he is beginning to make progress building the small logging and mining layout he has been planning for many years (after several visits to Colorado) – but keeps getting distracted by new computer and electronics projects for model railroading.