Contents

Chapter 1
DCC evolution 4

Chapter 2
DCC components 8

Chapter 3
Basic DCC technology 11

Chapter 4
Basics of decoders 15

Chapter 5
Decoder programming 22

Chapter 6
How to choose a DCC system 33

Chapter 7
Wiring a layout for DCC 41

Chapter 8
Decoder installation. 52

Chapter 9
Using LEDs and bulbs with DCC 62

Chapter 10
Installing speakers and sound decoders 68

Chapter 11
Operating with DCC 75

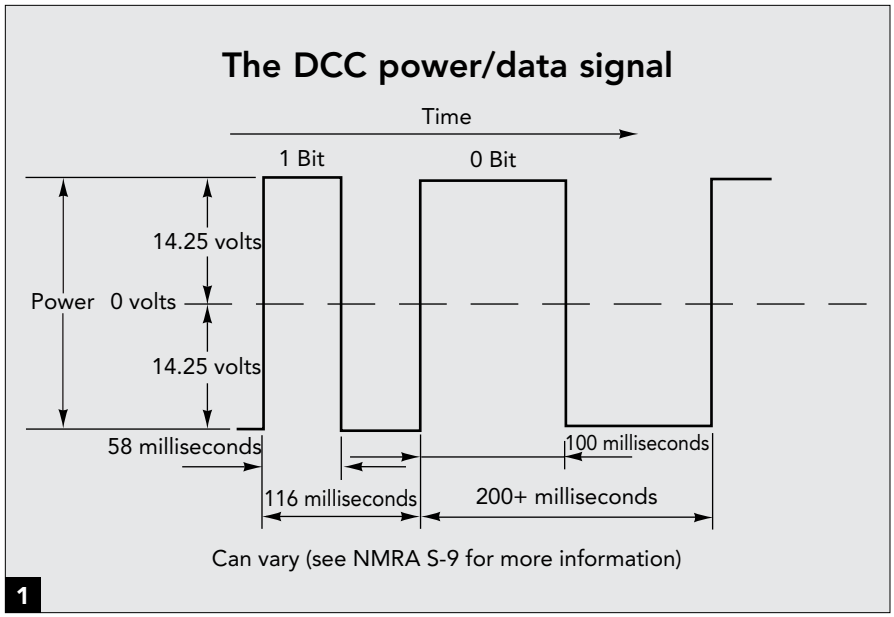
Chapter 12
DCC and today's technology 81

Glossary of DCC terms 85

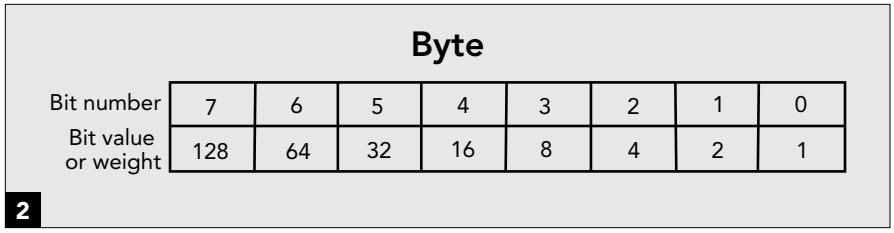
Troubleshooting. 86

DCC manufacturers and suppliers 87

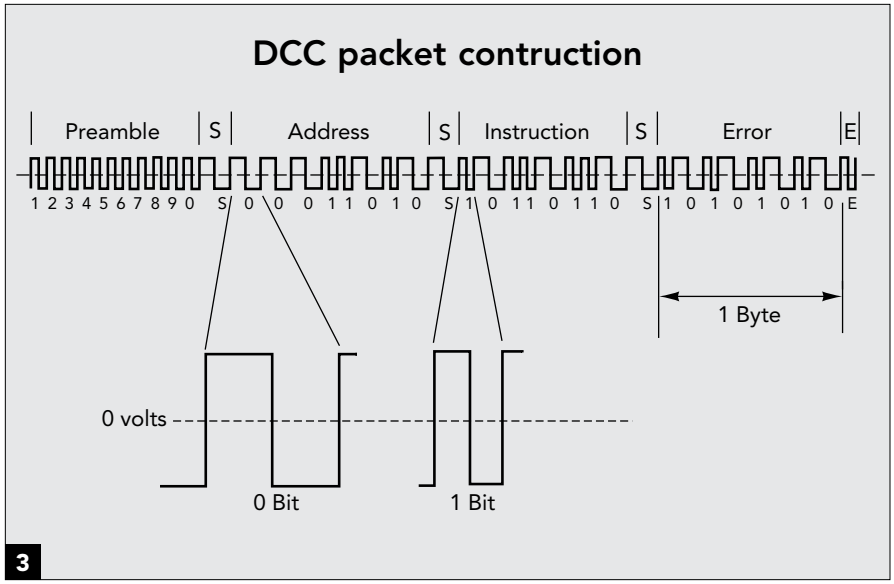
Acknowledgements. 87



The DCC signal is bipolar and switches between plus and minus. This keeps power continually on the rails, with data encoded by the length of time between the plus and minus transitions. The short signal is a "1" bit and the longer time is a "0" bit. The DCC standard is 14.25 volts.



The "0" and "1" bits are assembled into groups of eight bits called a byte. Bits are numbered from right to left, and each has a value, or weight, which doubles for each increasing bit number. The sum of all bit values is 255.



Bytes are put together in a group called a packet. Each packet starts with a preamble, followed by the address of the intended decoder. Next is the instruction. The packet ends with an error byte, which the decoder uses to verify the accuracy of the packet.

Power and data on the rails

The DCC system provides the power and necessary command signals to operate one or more locomotives, along with accessories such as turnouts and signals. The command instructions are imbedded in the electrical power that's fed to the rails to run locomotives or accessories. This is accomplished by rapidly changing the polarity from positive to negative. These positive-to-negative (plus-to-minus) transitions transmit the digital data, **1**.

The data is put into groups called packets. A decoder uses diodes to rectify the bipolar DCC signal. This recovers DC power independent of the data packet contents. A microprocessor in the decoder converts the power transitions to data.

The bit

Digital Command Control uses computer technology, so it shouldn't be surprising that the two share a similar nomenclature. The smallest element of digital information is a bit. Digital data is made up of 0s and 1s, each of which is one bit. This is because an electronic switch can be either on (1) or off (0). By combining bits into groups, we can make them indicate any number or value or have them represent any piece of information.

In DCC, time is used to define a 1 or 0—similar to the long and short whistle signals used by railroads, or the dot and dash of telegraphy. A bit is a cycle of one positive and one negative pulse. The 1 bit is shorter than the 0 bit—specifically, a "1" is between 55 and 61 microseconds long for a half cycle. The "0" is from 90 to 99,000 microseconds for a half cycle. The long 0, covered later in this chapter, is known as a stretched zero.

The byte

To make more sense out of 1s and 0s, they're put into groups of eight bits called bytes. Each of the eight bits in a byte has a position number and a weight or value. The first is bit no. 0 with a "weight" or value of 1. The count continues by doubling the weight of each bit to the last bit (no. 7) in the byte, which has a value of 128. The bits

4 Track packets (commands)

These are examples of actual packets, which were captured using a program called DCCMON. This program monitors packets on the rails in real time.

Packet	Binary					Hex	Instruction
1	0 11111111	0 00000000	0 11111111	1		(FF 00 FF)	Idle
2	0 00100101	0 01111001	0 01011100	1		(25 79 5C)	Addr: 37 Forward speed 12/13
3	0 11010110	0 00001001	0 01111001	0	10100110 1	(D6 09 79 A6)	Addr: 5641 Forward speed 6/6
4	0 11001000	0 00111110	0 01010101	0	10100011 1	(C8 3E 55 A3)	Addr: 2110 Reverse speed 3/3
5	0 11011001	0 00110011	0 01110101	0	10011111 1	(D9 33 75 9F)	Addr: 6451 Forward speed 15/16
6	0 11000011	0 11110011	0 10000010	0	10110010 1	(C3 F3 82 B2)	Addr: 1011 Function Group 1
Byte	1	2	3	4			

The binary is the only part that is transmitted. The DCCMON program generated the hex and also added the description of the commands. The preamble isn't shown.

The binary commands also show the "0" bit separators between bytes and the "1" at the end of the packet. Packet 1 is an idle command that's used by the command station when there is no data to send. Packet 2 is a three-byte packet to locomotive no. 37 and is a speed command. Packets 3, 4, and 5 are four-byte packets to locomotives with long addresses. Direction and speed are included in these packets. Packet 6 tells locomotive no. 1011 to blow its horn.

are counted from right to left, **2**. The values of all 8 bits in a byte add up to 255. This will begin to make sense as you begin programming decoders using configuration variables (CVs), covered in chapter 5.

The packet

The Internet uses packets to transmit data. Packets include the recipient's address, followed by data. This is how each individual computer receives information on the Internet over a common line. Digital Command Control systems also use a form of packets to transmit data.

The command station assembles the DCC packets based on commands from a cab or throttle. Each packet has four parts: Preamble, address, instruction, and error code, **3**. All parts of the packet are separated by 0 bits, **4**.

The preamble is a minimum of ten 1 bits, and can be longer. More than eight bits are needed so the decoder can determine the start of a packet to get synchronized with the command station. No other combination of ten or more 1 bits together is possible with the coding of the packets.

The address byte specifies the intended decoder. The address can be more than one byte long.

The instruction byte or bytes tell the decoder what task to execute, such as speed or function. This can also be more than one byte long.

The command station generates the error byte using the data in the packet. All of the bits in the packet are used to generate the error byte. The error byte is a unique number developed from the data in the packet.

Decoder operation

A decoder first gets into sync by finding a preamble. Once the decoder finds the preamble, it knows that the next parts are address, instruction, and error byte. All decoders receive the complete packet and then go through the same process the command station did to generate the error-code byte. If the error byte received from the command station matches the one generated by the decoder, the packet is considered valid. If the error code doesn't match, the whole packet is rejected.

Once the decoder determines that the packet is valid, it checks the address. If the address matches the decoder's address, the decoder then checks the instruction byte. If the instruction—such as a change in speed—is valid for the decoder, it executes the instruction. If the instruction is invalid, nothing is done.

Types of packets

Packets are transmitted using a method known in the computer industry as "ship and pray"—the DCC packets are shipped out, and the system "prays" that they get to the destination. Rail/wheel contact makes for a poor electrical connection, so to ensure that the packet gets to the decoder, the command station sends multiple copies of the packet. The basic packet is only four bytes long, but with long addresses (four-digit instead of two-digit), the packet size is expanded.

Programming step-by-step

There are two basic methods of programming a decoder: Using the programming track or using program-on-the-main (POM) or on-the-fly programming.

Here's a step-by-step description of programming a decoder on the programming track using an NCE Power Cab (other systems will vary—check your manual for specifics).

Place the engine on the programming track. With power on, press the PROG/ESC key four times until you see:
SEL MODE XX:XXAM
USE PROGRAM TRK
Press ENTER. You should then see:
PROG TRK
1=STD 2=CV 3=REG
Press 2 for CV programming and the display will be:
PROG CV
PROG CV NUM:
Enter the CV number to be programmed—for example, 1
for the short address:
PROG CV
-----WAIT------
The response after the wait will be either the value in CV1 or:
PROG CV
CANNOT READ CV
Sometimes sound decoders have difficulty reading back the value on the program track. If it read OK, the display will show:
PROG CV
CV VALUE = XX
Press ENTER and then
PROG CV
CV NUM 001 = XX
Press ENTER, and the number will be programmed into CV1. The display will come back asking for the next CV to be programed:
PROG CV
PROG CV NUM:
When you're finished, press ESC/PROG to get out of the programming mode.

Program-on-the-main, or on-the-fly programming, lets you program CV settings while the engine is in use on the layout (not on the programming track). With this method, it's not possible to read back CV values. Also, unlike program-track programming, you must use the address of the locomotive to be programmed.

Here's an example, again using the NCE Power Cab. Press ESC/PROG once and the screen shows:
SEL MODE XX:XXXM
PROGRAM ON THE MAIN
Press ENTER and you'll see:
OPS PROG XX:XXXM
PROG LOCO: 1234
Be sure that the proper locomotive number is selected, or you may program the wrong one. Press ENTER and the display is:
LOC:1234 XX:XXXM
1=ADR 2=CV 3=CFG
There are other options if you press ENTER, or if you know the option number, just press the number key. There are nine options. Option 9 allows you to set individual bits in a CV. The screen looks like this for option 9:
CV: 00X 76543210
BITS 00000000
Type the number of the bit to change, and it will change to a 1; type the same number (1, 0) again to toggle it back to a 0. Press ENTER to program the CV and more CVs, or ESC/PROG to return to standard operation. For the 2=CV mode simply press 2, then the CV number, and then the value. ENTER to program and return to normal operations.

values in these CVs on-screen using a mouse, then send the settings to the decoder.

Another way to control speed is with CV25 (an optional CV used in some decoders). This CV sets a speed-curve profile. Programming a value in this CV turns on one of the predefined speed profiles in the decoder.

You can also adjust the speed of a locomotive when it's in a consist using CV23 for acceleration and CV24 for deceleration.

Some decoders now have a switching speed option activated with a function key. When activated, the speed is reduced by half, and the acceleration and deceleration are also reduced. This is handy when using a road locomotive as a switcher.

Simplified decoder programming

It's sometimes difficult to determine which CV controls what feature. If you get lost in the translation from CV to function, help is available. The DecoderPro program helps by putting the function name on the screen and then letting you enter a value or move a slider with a mouse to change

Accessory Decoders				
CV No.	CV function	Requirements	Default Value	Description
513	Low order address bits	Mandatory	–	High bits in CV521
519	Manufacturer version number	Mandatory	–	Version of the decoder
520	Manufacturer ID value	Mandatory	–	Assigned by NMRA
521	High order address bits	Mandatory	–	Low bits in CV 513
541	Configuration data	Optional	–	Like CV29 for accessory decoders

a CV value. DecoderPro is updated frequently and is free over the Internet (jmri.sourceforge.net). As new decodes become available, a new profile is set up online.

Accessory decoder programming

Accessory decoders use a different address range than mobile decoders: from 1 to 2044 (but not all DCC systems use the full range), **9**.

Programming an accessory decoder is different than programming a mobile decoder. Instead of using a programming track, the decoder is set to program mode with a programming jumper, wire, or push button. When in program mode, the decoder waits until the command station sends an accessory packet, then it stores the address in the decoder. The temporary jumper is removed, and the decoder retains the address information.

There are many variations on programming accessory decoders, so always check the decoder's manual for the correct procedure.

Accessory decoder commands

Some of the accessory decoders used for switch machines also have the ability to operate more than one switch machine with a single command. Some systems also have the ability to set up routes through a number of switch machines. NCE calls them "macros" and Digitrax refers to them as "routes." In either case, this allows you to assign a number to a route that will operate multiple switch machines (see your system manual for more information on these functions.)

In my main yard there is one track entering and one track exiting the area, so a number of routes need to be set up. I placed a map of the yard on the layout's fascia that shows the different switch and route numbers. The numbers are color coded so operates can differentiate between switches and routes.

I have a crossover between East and West main lines that's controlled by two solenoid switch machines. These machines need to operate together as either reverse for the crossover or normal for the mainline. I installed separate decoders for each machine and programmed them to the same address. A single command operates both switch machines.

NCE makes a Mini Panel that allows you to program operations on switch machines. This type of panel can be programed for many flexible routes. You can even set up an N-X (eNtrance-eXit) control panel. This is a type of panel used on prototype railroads for complex switching arrangements. The operator presses one button for the entrance to the yard and a second button for the exit route. When a valid route is detected, the panel issues the proper commands to align the appropriate turnouts for the route. There are some provisions for setting this up in JMRI with PanelPro.

9

Accessory decoders can control switch machines, as shown above, or many other accessories on a layout.

Nonfunctional decoders

Many decoder manufacturers report that a good share of decoders sent in for repair only need to be reset to factory default to be restored. If you have a decoder that's not functioning, there's a way to rescue it without sending it back to the manufacturer.

The read-only CV8 is mandatory and contains the manufacturer's identification number (see page 29). Writing to this CV will not erase the identification number. Instead, it can cause the decoder to be reset to factory settings. The reset response varies between manufacturers. Some flash the lights as a response.

If you have a QSI decoder, refer to the decoder manual for information on resetting. By using different CVs, it's possible to reset only part of the decoder while retaining other parts.

This is the Digitrax DC100 set, with a combined command station and booster (right). The UT400 cab (left) is Digitrax's top-of-the-line unit, and is also available in a wireless (radio) version.

The ECoS system by ESU has several unique features, including a 7" touch screen and the ability to function as a bridge between the rails and another DCC system.

need to be from the same manufacturer as your DCC system. Some systems, including the Digitrax Zephyr, ESU ECoS, NCE Power Cab, and MRC Prodigy, include a power supply.

The command station and first booster are normally combined into one unit and powered by a single transformer. Adding more boosters means adding more input power. The easiest way to do this is to use a separate transformer for each added booster. The transformer should have a current rating slightly higher than the rated output of the booster. Most use 15- to 18-volt transformers.

Using a separate transformer for each booster eliminates the possibility of short circuits between boosters and common transformers. It also makes troubleshooting easier. A transformer with a power switch is handy; if it doesn't have one, use a power strip with a switch and circuit breaker. Plug all your power supplies into a single power strip. This allows power for the entire layout to be turned off with a single switch.

Computer interface

Most DCC systems can be connected to a PC or Macintosh using the USB serial port as an interface; the ECoS system uses an Ethernet connection. Some older systems use the RS232 serial port—adapters (with and without cords) are available to connect an RS232 to a USB port.

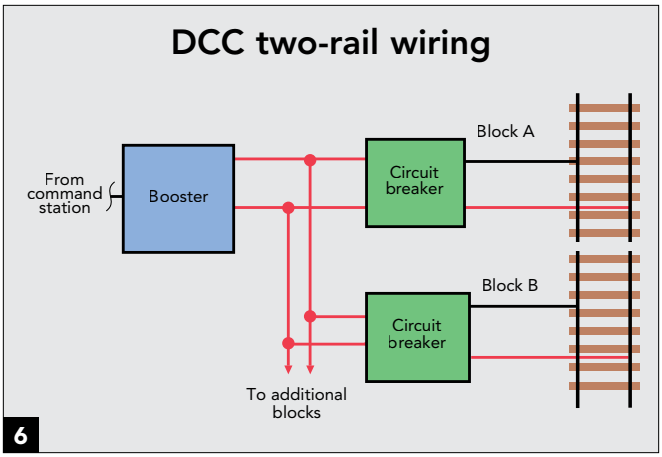
Programs are available to do a number of different tasks. One example is the Java Model Railroad Interface (JMRI) DecoderPro, which programs decoders and then remembers the settings of the decoders' CVs. This makes it easy to refresh a decoder if it gets "amnesia." (More on JMRI in Chapter 12.)

You can also use on-screen control panels to operate turnouts with a mouse. This makes setting up a Centralized Traffic Control (CTC) operation a lot easier than wiring all of the switch machines to a control panel. With the coming of DCC bidirectional data, the control panel could also show the location of each locomotive.

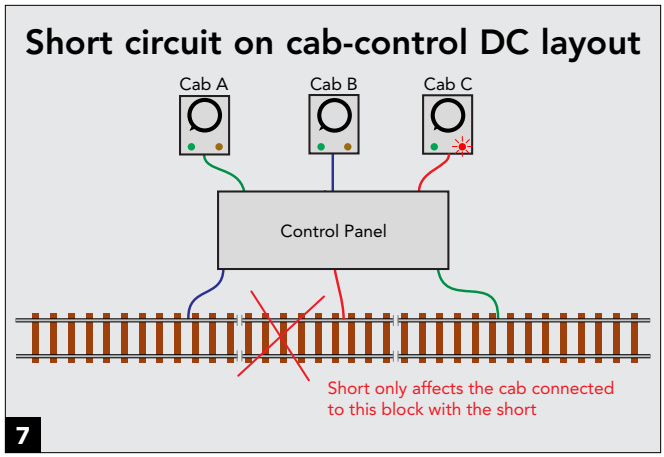
Decoders

Any brand of decoder should work with any DCC system, so selecting a locomotive decoder depends on your scale (the decoder must fit in your model), the current draw of your locomotive (the decoder power rating must be high enough), and your preference of sound effects (the latest sound decoders provide a wide range of engine sounds to match specific prototype steam and diesel locomotives).

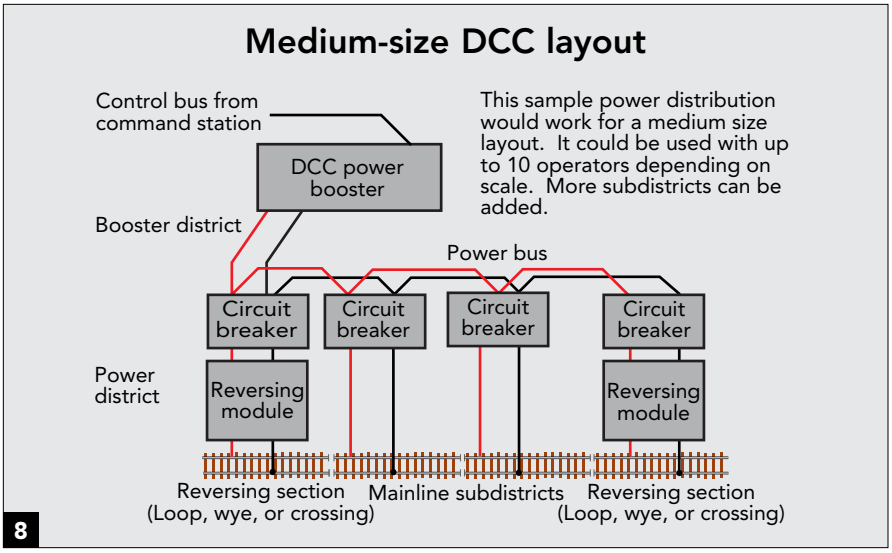
To find more information on specific decoders, see their instruction manuals—most of which can be downloaded from manufacturers' Websites.



There are two choices for wiring a layout. Common-rail wiring can cut down the number of wires, but can be a little confusing. It's recommended for layouts with two or more boosters. Two-rail wiring (shown here) is easier to understand and the more popular method.



Short circuits on DC layouts only affect the train in the block with the short. When a short occurs on a layout with DCC, the booster shuts down, killing power to the entire layout. Adding more blocks and circuit breakers limits the areas affected by short circuits.



Booster districts can be divided into power districts by adding circuit breakers between the booster and sections of the main line.

layout. When a short circuit occurs, only the operator running the involved train is affected, **7**. Other trains continue to run because they are controlled by separate power packs.

With DCC, the power to the rails comes from a booster that's fed instructions from a command station. These instructions come from each operator's cab. On a layout with a single power booster and no other circuit protection, a short circuit will shut down the entire layout, stopping all trains.

The DC cab-control method of separating track into blocks, with circuit protection for each block, can be used on a DCC layout to minimize the effects of short circuits.

This is like a house wired with a main circuit breaker that shuts off all the power, followed by several secondary circuit breakers or fuses used for smaller areas of the home, **8**. If the total amount of power used in a home exceeds the rating of the main breaker, it will trip; otherwise, overloads and shorts only affect their respective circuits.

DCC power boosters have their own built-in circuit breakers that are like the main breaker in a home. Additional breakers can be added between the booster and track.

Blocks for DCC

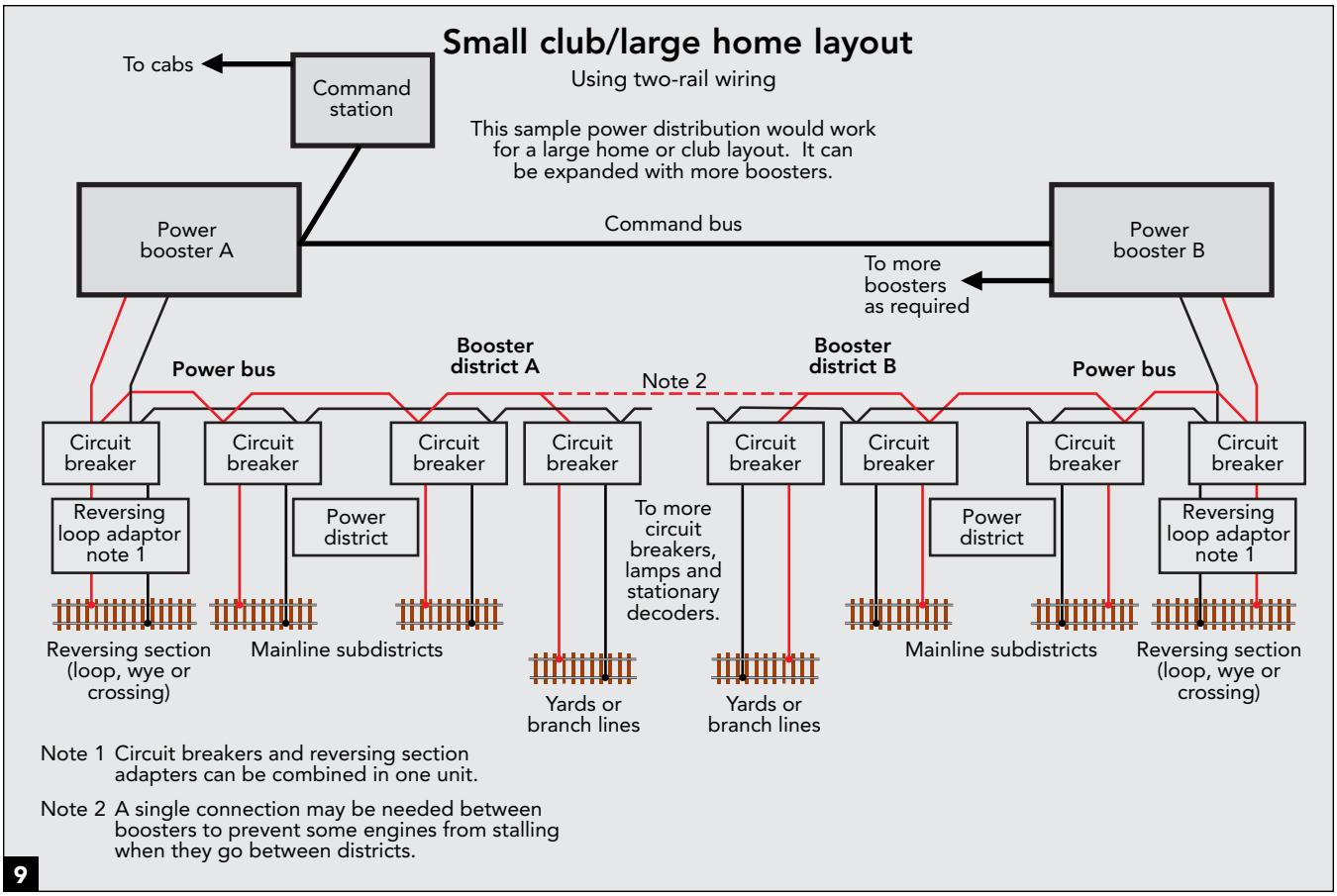
Dividing a layout into blocks for DCC can be done in much the same way as DC. The difference is that more than one train can be running independently in a block. Also, blocks can be longer, so you'll usually wind up with fewer blocks in DCC. This results in more locomotives in a single, longer electrical block. You may need to wire these blocks with heavier wire.

Blocks fed by a booster are booster districts; blocks feed from circuit breakers connected to boosters are power districts. Decide where to place block boundaries based on the probability of where short circuits will occur. Yards, industrial areas, and branch lines are good candidates for separate power. Sections of main line with passing sidings or crossovers are another. You can always divide power districts into smaller blocks later if needed.

When making a booster district or power district, the rails must be gapped to electrically isolate the district's rails.

My DC layout had 16 blocks on the main line (including two reverse loops) plus another four for yards and branch lines for a total of 20 blocks.

With DCC, my layout now has ten power districts. The main line is now protected by six circuit breakers, **9**. Two of the six mainline blocks are reverse loops and use reverse loop adapters that include circuit breakers.



9

Track powered by each booster is a booster district. Each district can be subdivided into power districts using circuit breakers. Additional boosters can be added for larger layouts.

Power districts

More power districts and boosters are needed when you have exceeded the power of the existing booster(s). When more than one power district is used, the DCC signal in each district must be in phase with the adjacent district, meaning the polarity of the rails at the gapped joint between districts must match. If the polarity doesn't match, a locomotive passing over the border will cause a short circuit as its metal wheels bridge the gap.

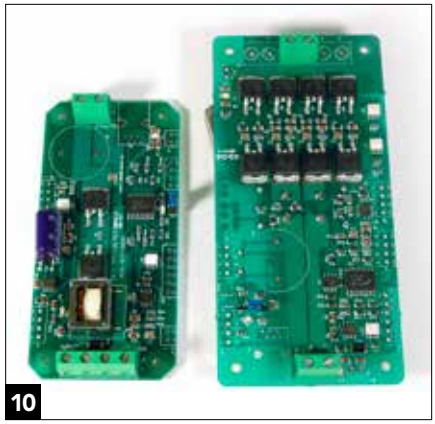
There can also be a problem with some engines stalling halfway across the gaps. Modern models with all-wheel electrical pickup won't have this problem, but some older locomotives have electrical pickup staggered between trucks (or the drivers and tender) so that the left-rail pickup is on one side of the gap and the right-rail pickup is on the other. Check your system manual for any special instructions on connecting boosters together.

Power district protection

Booster districts can be further divided into power districts so a single short will not shut down the whole booster district. Power districts are less expensive to add than boosters. Each block should have individual circuit protection. Power districts are protected with circuit breakers, **9**. Don't use fuses, as they'll have to be replaced each time a short occurs.

The circuit breaker should be rated or set to trip at a current less than the output of the booster. Circuit breakers also must cut the current off in less time than the booster's built-in breaker (some circuit breakers have adjustable timing). Some circuit breakers have special settings to work with entry-level systems. Circuit breakers designed to work with DCC automatically reset themselves about once per second until the short is cleared.

Some circuit breakers have an output to feed an LED to show when the breaker is tripped and power is off. Some manufacturers include more than one breaker in a single unit. These circuit breakers can be driven by a single booster.



10

The circuit board on the left is a Power Shield (PSX-1) single circuit breaker. The circuit board on the right is a PSX-AR reversing module with circuit breaker. These solid-state circuit breakers are designed to sense the difference between a short circuit and the high inrush current used to charge capacitors.

This is a novel way using an iPhone or tablet in video mode to take a ride on a layout. The resulting video will show you views you've never seen before. *Adrian Pardo*

A variety of DCC topics and videos of sound-equipped models can be viewed online at *Model Railroader's* website and its subscription service, *Model Railroader* Video Plus.

Your smartphone or tablet can operate a layout with Wi-Fi using Java Model Railroad Interface (JMRI), a free online program for programming decoders and operating your model railroad.

Ride the rails

An iPhone or other type of smartphone can be used to take a ride on the layout using the video mode, **1**. I found it very interesting when I installed a miniature video camera in one of my cabooses. It's amazing what you find in tunnels!

Searching for solutions

Search engines like Google, Yahoo!, and Bing make it possible to find information on almost any aspect of DCC. All it takes is a few key words to find a wealth of information on items like installing decoders, wiring, and lighting, among other topics.

If you can't find answers to your questions with search engines, you may want to check out the various Yahoo! Groups (www.groups.yahoo.com) that focus on DCC topics. These groups are made up of fellow modelers from around the world. With DCC, it's a pretty safe bet that someone else has had the same problem you do and has an answer.

There are Yahoo! Groups devoted to almost every DCC manufacturer. Other groups cover broader topics like layout sound and DCC wiring. Answers to questions often come from somebody knowledgeable about the topic, and there will likely be multiple responses. Most groups have archives with a string of questions and answers that you can search by keyword. These groups are not help lines sponsored by a manufacturer with someone available 24/7, so it may take some time for someone to read your question and return with an answer. There is even an occasional bit of humor sprinkled in with the replies.

It's easy to search for the group or groups that you're interested in. Some groups need to approve new members. Responses from the group come as an e-mail. There are a few options on how you can receive the replies.

If you can't find a group you're interested in, you can create a new group. The groups can be open or closed. The latter lets you restrict membership to members of your club or other interested persons. I belong to a restricted group interested in DCC that meets once a week for lunch.

Video channels

Video is a great source of information. YouTube has a large number of videos on DCC and other model railroading topics, but the quality varies.

Model Railroader magazine has videos on its website, as well as its subscription service *Model Railroader* Video Plus, **2**. On these two sites, you can see and hear the latest sound-equipped locomotives and learn how to program decoders.

Java Model Railroad Interface (JMRI)

Java Model Railroad Interface was originated by a group of modelers in the Silicon Valley. They realized it would be a lot more convenient to be able to program and store decoder settings using the power of a computer. Java was selected because it will run on many different OS platforms. Java is available for Windows, Macintosh or Linux, and it works with a full range of DCC systems, including C/MRI, Digitrax, EasyDCC, Lenz, and NCE, among others.

JMRI is divided into three areas with different applications: DecoderPro for programming decoders; PanelPro for layout operation, signaling and control panels; and SoundPro for controlling layout sound and lighting.

Two program levels of JMRI are maintained: a test version and a release version. The test version is used for debugging newly added and modified software. When the test version works satisfactorily, it's elevated to the release version.

When new decoders are released, new definitions are needed. One of the JMRI contributors can volunteer to create the new software definition. The DecoderPro software is user-friendly because it uses names for the configuration variables and CV numbers, which are both displayed.

How we use DecoderPro is similar to how we use newer telephones. In the past, we used to remember

WiThrottle can be used to run a layout. The dual throttle app is shown. *B. Hoffman*

The new Tool Box app from Digitrax provides manuals, videos, and more. It's available for iOS and Android devices.

telephone numbers. Now we store names and the phone remembers the numbers. DecoderPro remembers the CV numbers and shows you the name of the function. This takes a lot of the mystery out of decoder programming.

Once you are satisfied with a decoder's settings, they can be saved in a roster file. Complete rosters of all your locomotives can be stored for future use, which is handy if you need to restore a decoder. A stored file can program a new decoder and then change the locomotive address.

An interesting ability of PanelPro is setting up remote wireless control panels on smartphones or tablets. Multiple panels can be accessed and toggled between using the tab function of the browser. Larger tablets work best in this application. Smartphones work better as cabs.

Another feature of JMRI is WiThrottle, **3**, which sets up a Wi-Fi link to smartphones and tablets. The app is designed for the iPad, iPhone, or iPod touch. WiThrottle Lite is free and will run one locomotive (address) at a time. It controls speed and emergency stop plus all 28 functions. An inexpensive version of this app will run two trains, do consisting, and set up routes through JMRI. It can be set up as a road or yard throttle.

WiThrottle is available from the Apple App Store. These devices work only with Apple's iOS system. The Wi-Fi connection doesn't need an Internet connection; it will operate with a router.

Visit WiThrottle's Website, www.withrottle.com, for more information.

There are a lot of help files on the JMRI website. There are also several YouTube videos that show setting up and operating JMRI.

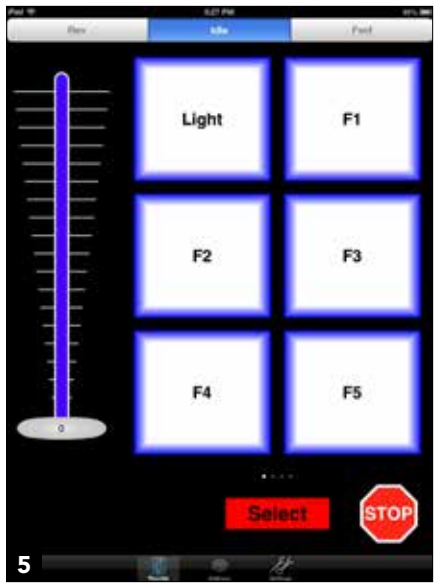
Since JMRI is an open-source project, it's growing and evolving in many directions at once. If you're interested in getting involved in one of the many levels, check out the Website, jmri.sourceforge.net

Digitrax Tool Box

Digitrax has an app for iPad, iPhone (iOS), and Android devices called the Tool Box, **4**. This app is well done with lots of text, manuals, and how-to videos. There are even manuals for discontinued products available at this site. To get the app for either iOS or Android, go to www.digitrax.com and do a search for Tool Box.

Touchscreen Cabs

A smartphone or tablet is a powerful battery-powered handheld portable computer. They have the capability of being wireless cabs with a touchscreen. Recently, there's been a lot of activity in turning handheld smart devices into portable wireless cabs/throttles. There are many different ways that this can be accomplished. An operator can bring their smartphone or tablet from home to the club and use it as their cab for the operating session, then go home and use it there. The club could have a Digitrax system and the home an NCE system and



This screenshot taken on an iPad illustrates the controls for a locomotive. This is the standard version for locomotive controls.

the smart device would work for both through JMRI, **5**.

Wi-Fi on Android OS tablets and phones

Android is an operating system used in most of the new non-Apple tablets and smartphones, **6**. There is an app called Engine Driver that uses the same Wi-Fi connection to JMRI. This app will work with WiThrottle, allowing a mix of IOS and Android devices on the same JMRI system.

TouchCab

TouchCab, **7**, is an app that will work with an ECoS or Lenz command