User's Manual for the ProMiniAir Transmitter and Receiver

OScaleDeadRail
https://oscaledeadrail.com

Darrell Lamm
darrelllamm0@gmail.com

June 4, 2023

Contents

List of Figures

List of Tables

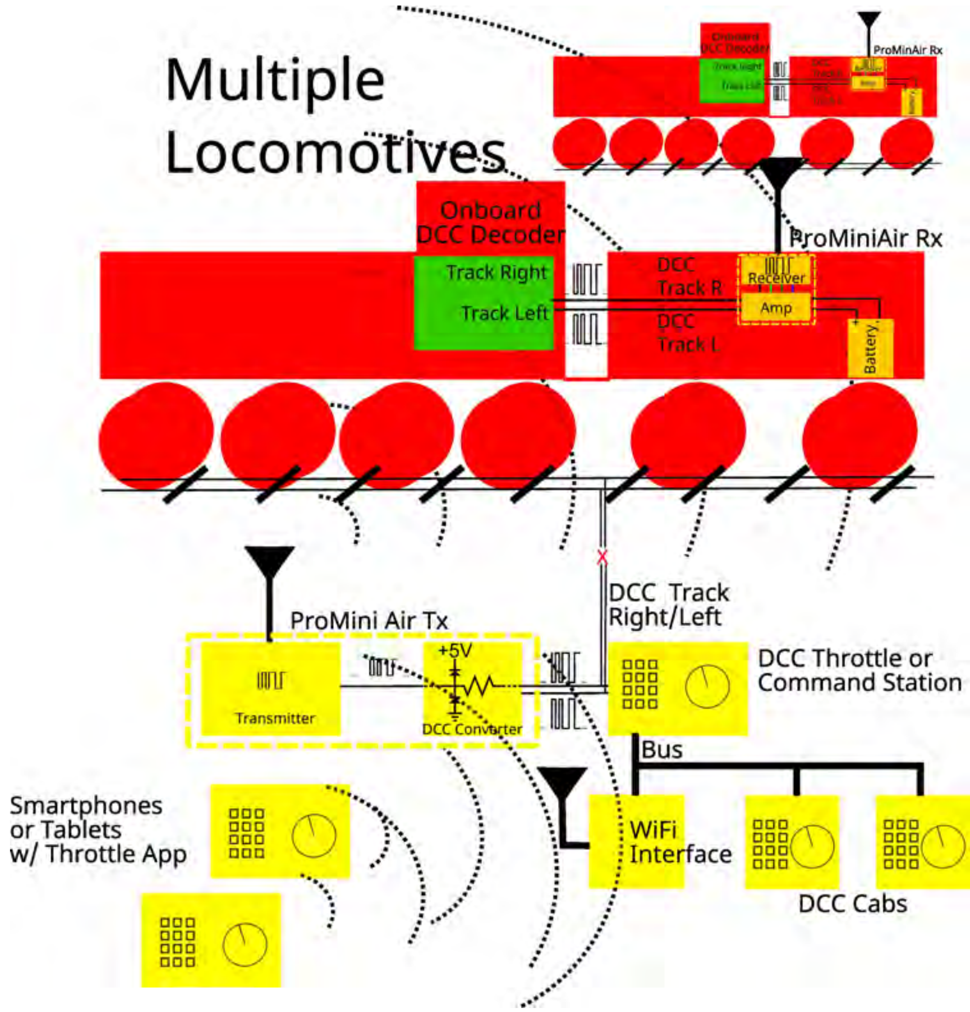


Figure 1: Typical model railroad battery-powered radio-controlled application.

1 Introduction

Before understanding how to set up and use the ProMiniAir transmitter and receiver, let's set the stage with Figure 1. What's different from typical model railroads is that power comes from a battery rather than electrified rails, and DCC (more about this in a moment) control information is received by a radio receiver (from a radio transmitter connected to a DCC throttle) rather than from DCC signals imposed on the track voltage by a throttle. This battery-powered, radio-controlled method of operation is sometimes called "dead-rail" because neither power nor signal is applied to the rails.

2 About DCC

We'll introduce some terminology that may be new. The National Model Railroad Association (NMRA) set forth a standard [1] for communicating with decoders onboard locomotives and other model railroad devices called Digital Command Control (DCC). DCC-compliant throttles control and configure these devices by sending coded, digital voltage waveforms over wires or tracks (usually) to these devices, which sift out those addressed to them. These coded waveforms contain digital messages or "packets" specify an action and an "Address" that determines the recipient device. Most DCC packets are addressed to a specific recipient, although a few kinds of messages are for *all* listening recipients.

Any DCC output (say from Digitrax, NCE, or other DCC throttle such as the low cost WiFi-equipped EX-CommandStation we offer) can be wirelessly transmitted using the ProMiniAir transmitter over the "Airwire Channels" 0-16 in the North American 902–928 MHz or European 868-870 MHz "ISM Band" band (the Airwire or European channel (channel 18) and output power level is adjustable as described below) to a variety of RF receivers, including the ProMiniAir receiver, that in turn convert the RF transmissions back to DCC signals any DCC decoder can "understand."

We also provide a North American channel 17 for Stanton-Cab transmitters (to the ProMiniAir receiver) or Stanton-Cab receivers (from the ProMiniAir transmitter).

After the ProMiniAir has been interfaced and powered as described in the next section, the ProMiniAir transmitter or receiver can be operated without further ado. However, you can reconfigure some aspects of the ProMiniAir's operation as described later in this Manual.

3 Quick-Start

We provide the ProMiniAir transmitter or receiver as a fully-assembled, modular device, already connected to any additional modular components necessary to act as either a transmitter or receiver.

For the ProMiniAir transmitter, you simply connect the DCC Converter to a DCC throttle's Track R/L output that is used as both a power and DCC data source (Figures 3 and 2), or simply plug in power to use the WiFi-equipped EX-CommandStation we offer as a fully-standalone transmitter that allows you to use a smartphone's WiThrottle-compatible app to connect to the ProMiniAir transmitter (Figures 4 and 5).

If you are using the integrated PMA Tx/WiFi+EX-CommandStation, see these instructions.

For the ProMiniAir receiver, connect battery power to the power harness plug and plug the DCC amplifier's output to the Track R/L inputs to DCC decoder (Figure 6 and 8).

Both the ProMiniAir transmitter and receiver can be optionally connected to an Organic Light-Emitting Diode (OLED) display that uses the "I2C" interface that uses ProMiniAir outputs GND, VCC (+5V), SDA (3.3V data), and SCL (3.3V clock) as input. The display

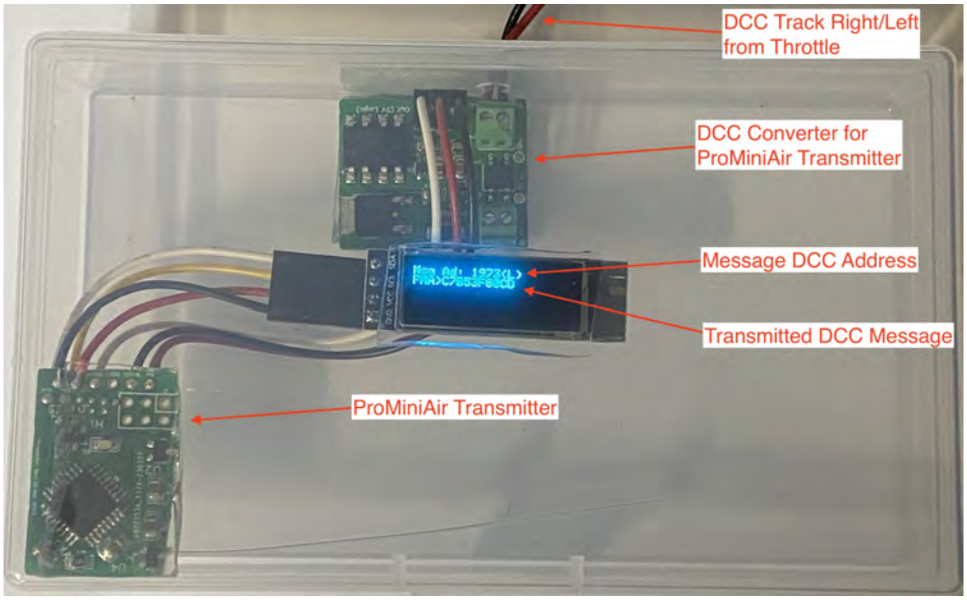


Figure 2: The ProMiniAir transmitter for a DCC throttle

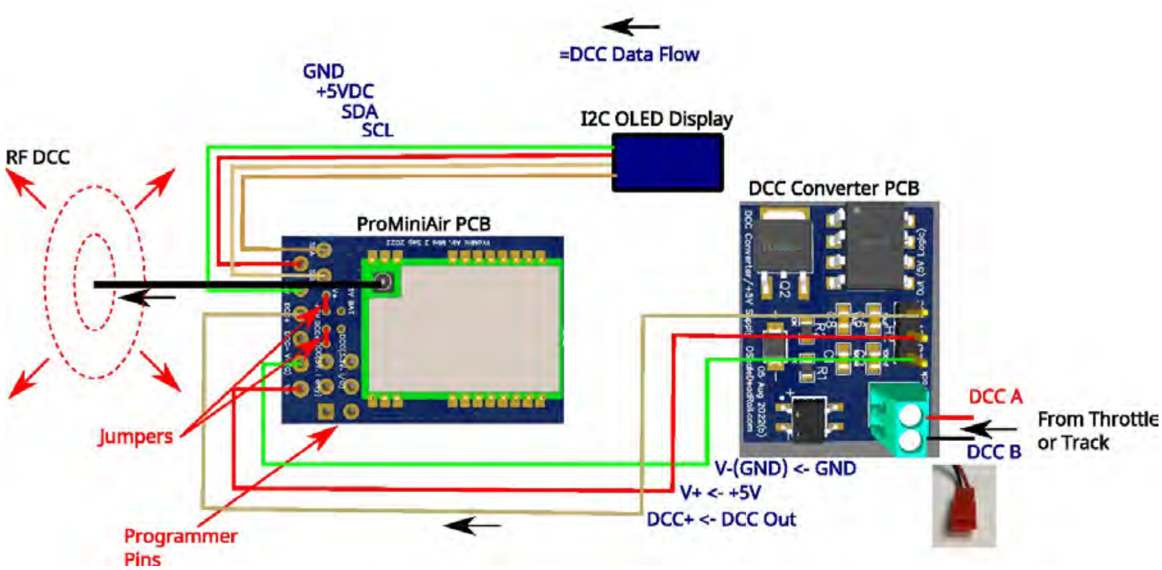


Figure 3: The components and connections needed for the ProMiniAir transmitter for a DCC throttle

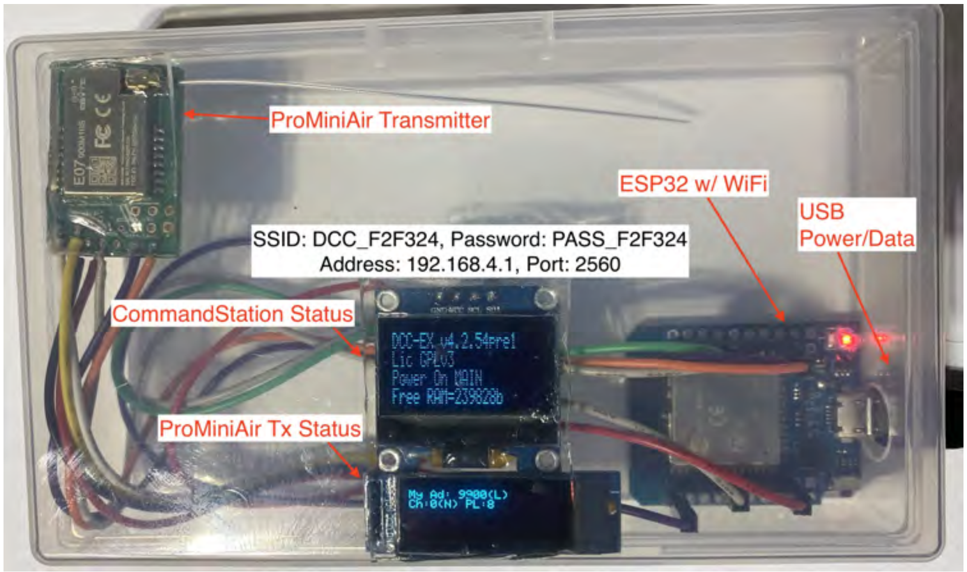


Figure 4: Integrated ProMiniAir/WiFi-equipped EX-CommandStation transmitter. In this case, the DCC converter is eliminated, and the ProMiniAir transmitter is directly integrated with the WiFi-equipped EX-CommandStation for a complete, stand-alone solution allowing connections from both USB-connected and smartphone throttles.

provides information on DCC messages received (PMA < `DCC_Packet`) and sent (PMA > `DCC_Packet`). See Figures 10 and 11. Note that some figures show older LCDs that are no longer used with ProMiniAir.

Of course the OLED display cannot be used once the ProMiniAir *receiver* is mounted, but we strongly suggest that you keep the OLED display connected to the ProMiniAir *transmitter* so you can monitor the ProMiniAir transmitter's state such as broadcast channel and power, and DCC message content.

3.1 Using Multiple Transmitters with Multiple PMA Receivers on Different Channels

Only on startup, the ProMiniAir Receiver will "listen" on its Default Channel, and if it doesn't find a valid signal, it will start scanning on Channels in the following order: 0(A), 18(E), 17(S), 1(A), 2(A), 3(A), 4(A), 5(A), 6(A), 7(A), 8(A), 9(A), 10(A), 11(A), 12(A), 13(A), 14(A), 15(A), and 16(A); where A=Airwire, E=EU, and S=Stanton Cab. This feature allows an easy technique for using multiple compatible transmitters (ProMiniAirtransmitters, such as Airwire transmitters, etc.) that transmit on *different* Channels using the following technique:

1. Ensure all transmitters and receivers are OFF.
2. Turn ON Transmitter #1 on Channel C#1 (C#1 = 0, 1, ..., 18)
3. Turn ON only the ProMiniAir Receivers you want to auto-scan and "lock" on Channel C#1. It is useful to test the locos to ensure the onboard ProMiniAir Receiver has

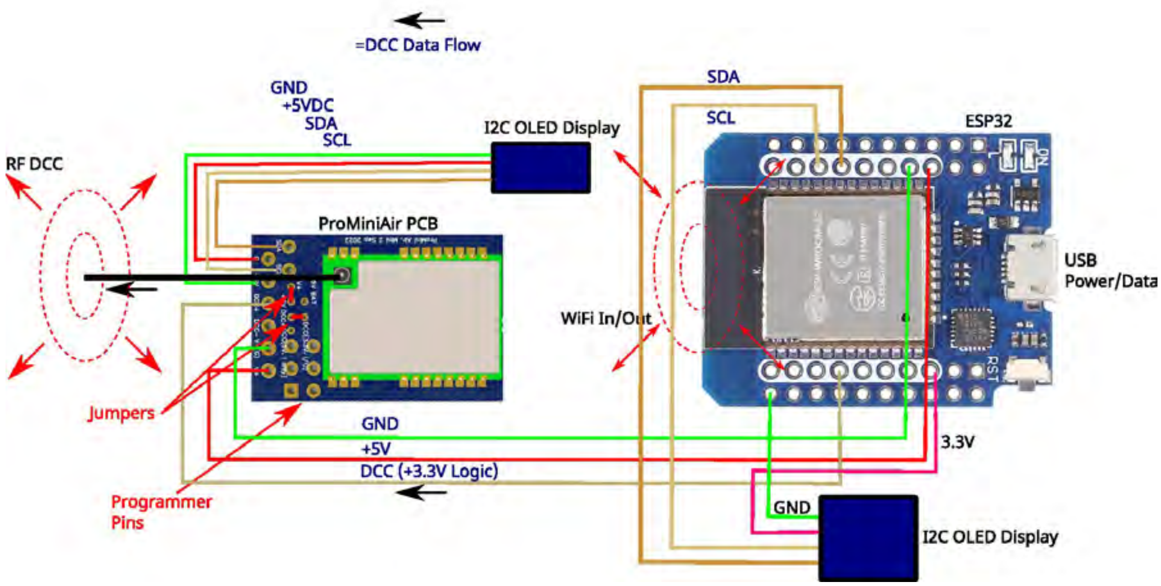


Figure 5: The components and connections needed for the "stand-alone" ProMiniAir transmitter, which is integrated with a WiFi-equipped EX-CommandStation.

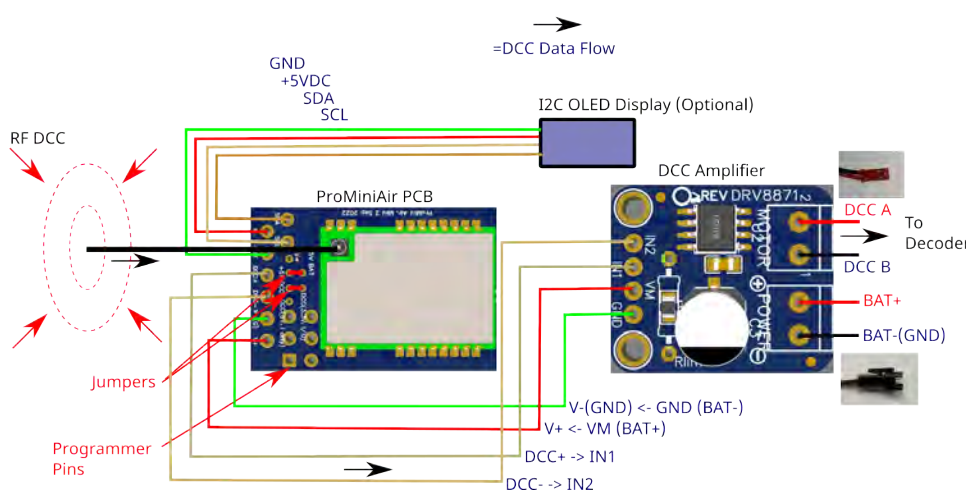


Figure 6: The components and connections needed for the ProMiniAir receiver (DRV8871 amplifier)

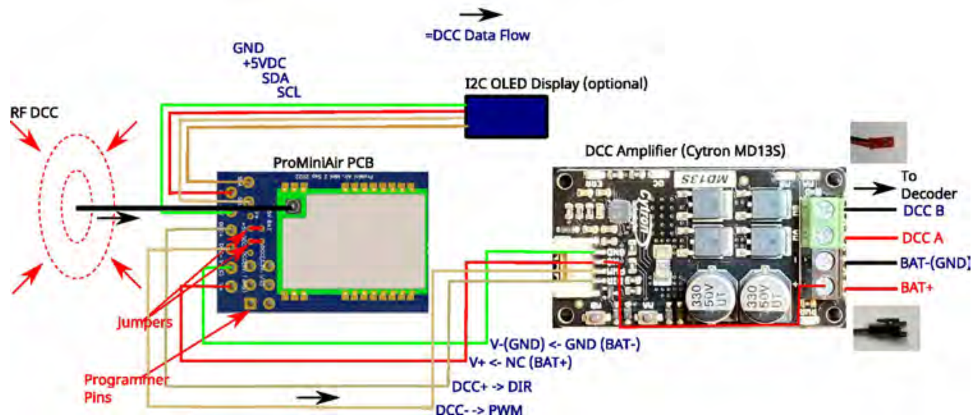


Figure 7: The components and connections needed for the ProMiniAir receiver (Cytron amplifier)

properly locked on Channel C#1.

4. Turn OFF Transmitter #1. *Leave the receivers ON!* The ProMiniAir Receivers will stay locked on Channel #1, patiently waiting for a valid signal on this Channel!

5. Turn ON Transmitter #2 on Channel C#2 (please make sure C#2 is NOT the same as C#1).

6. Turn ON the next set of ProMiniAir Receivers that you want to auto-scan and "lock" on Channel C#2. Again, it's useful to test the locos operating on this Channel.

7. Turn OFF Transmitter #2. At this point, no transmitters should be ON, but two groups of ProMiniAir Receivers locked locked on to Channel C#1 or C#2 remain ON.

8. Continue to Steps 5–7 for all additional Transmitters and groups ProMiniAir Receivers.

9. At this point in time *all* Transmitters should be OFF with all ProMiniAir Receives patiently waiting in groups locked onto Channels C#1, C#2, etc.

10. Now turn ON *all* of the Transmitters! Each group of ProMiniAir Receivers will now receive valid signals on the Channel it locked onto at startup!

3.2 Additional Technical Information

The ProMiniAir requires interfacing to additional electronics that are provided. For the transmitter, the DCC signals that would normally be applied to the tracks by a DCC throttle

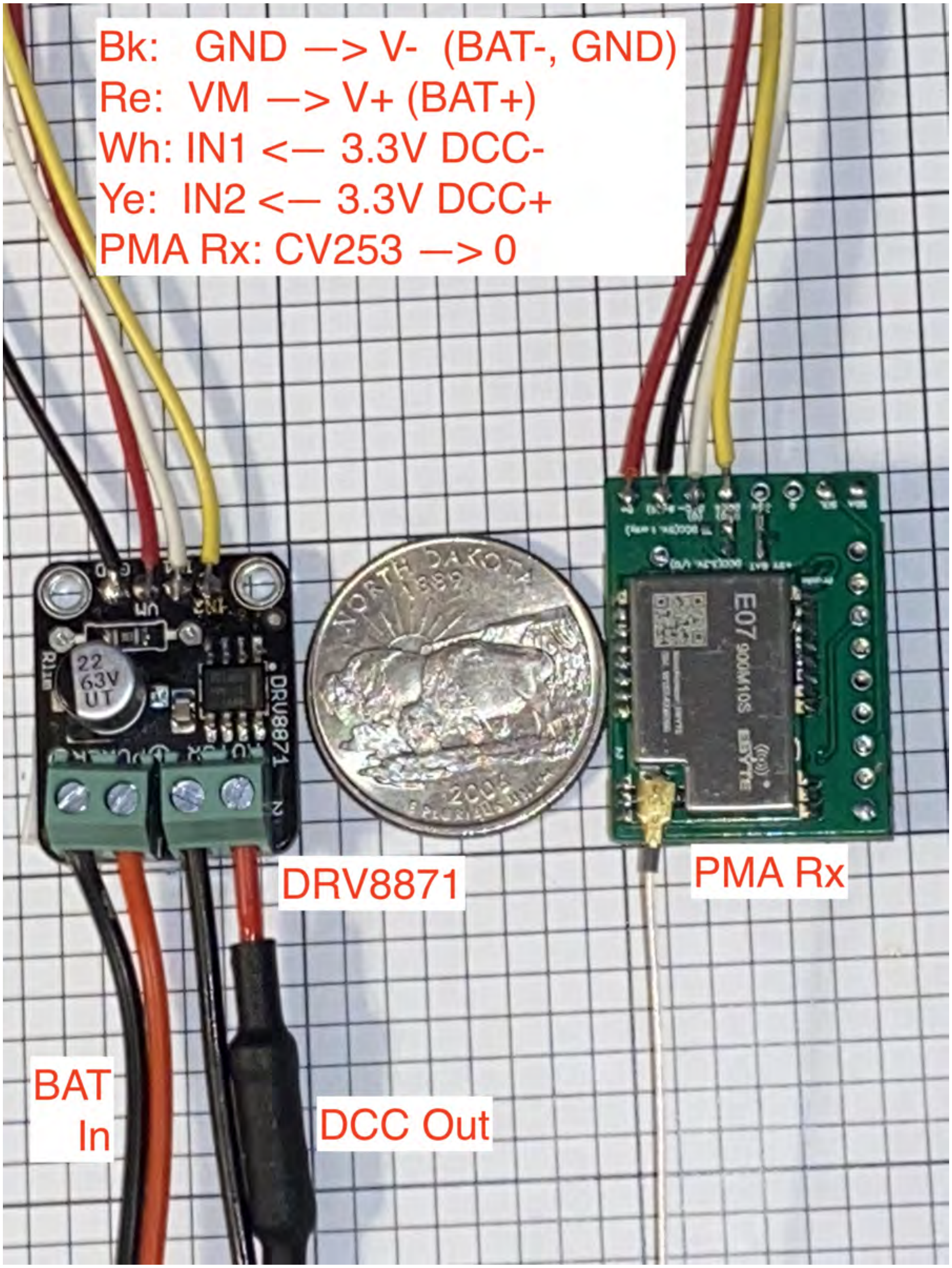


Figure 8: Example ProMiniAir receiver connections with a DRV8871 amplifier. Note the 1 ohm current-limiting resistor on the DCC out.

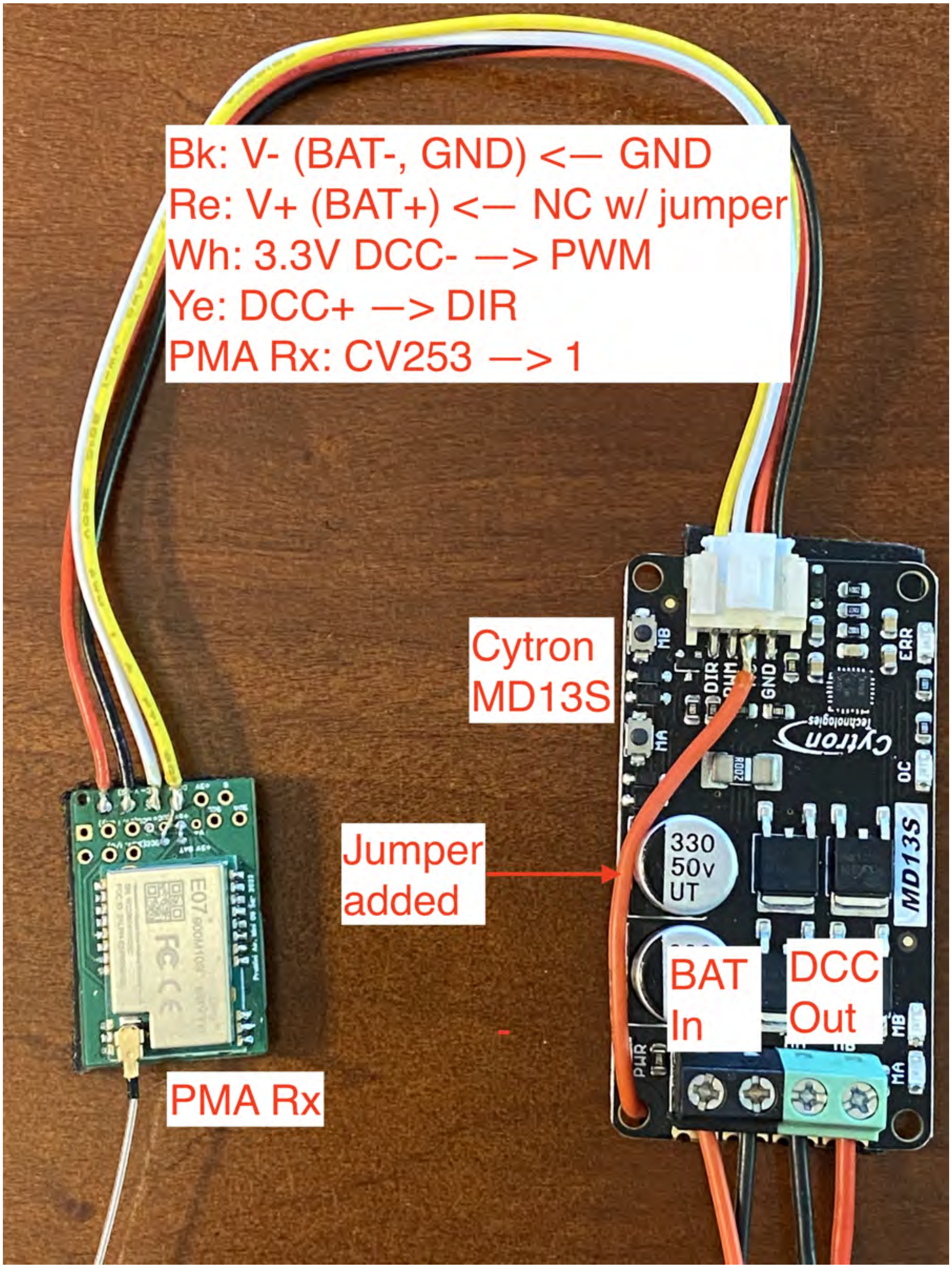


Figure 9: Example ProMiniAir receiver connections with a Cytron MD13S amplifier

Figure 10: LCD showing message DCC sent (Tx: sent wirelessly; Rx: sent to DCC amplifier)

are input to a simple "DCC Converter" that provides both 5V DC power and 5V logic DCC signal for the ProMiniAir transmitter. See Figure 3 and 2.

Should the wires become disconnected, see Figure 5 for the connections of the ProMiniAir transmitter integrated with a WiFi-equipped EX-CommandStation.

For an small additional cost, we integrate the ProMiniAir transmitter with a WiFi-equipped EX-CommandStation for a fully-integrated stand-alone transmitter solution. Once connected to the EX-CommandStation's WiFi network, the user can employ a WiThrottle-compliant smart phone app, such as WiThrottle, as a throttle for dead-rail control. Also, the user can connect a computer to the EX-CommandStation's USB port, and use various computer-driven throttles, including web-based throttles. See this web page for full details on using smartphone apps and the ProMiniAir integrated with a WiFi-equipped EX-CommandStation.

For the receiver, the 3.3V logic DCC signal output by the ProMiniAir receiver is converted by a "DCC amplifier" back to "track DCC" signals that an onboard DCC decoder can "understand." See Figures 6, 7 and 8. We generally integrate either the small (1" x 0.8") 3.6A DRV8871 amplifier or the more powerful 13A Cytron amplifiers.

We decided to take this modular approach to allow: 1) the user to replace failed components (and not the entire assembly) and 2) us to reduce our exposure to supply chain disruptions and parts obsolescence, keeping your costs low.

We believe this "modular" approach gives you flexibility for both installation and parts replacement. As an additional benefit of the ProMiniAir's modularity, the user is that a

Figure 11: LCD showing message DCC received (Tx: received from DCC throttle; Rx: received wirelessly)

transmitter can be repurposed as a receiver or vice versa with a simple firmware change.

3.3 Antennas for the ProMiniAir

The ProMiniAir *transmitter* comes with a "whip" antenna that preserves FCC/IC/ETSI/CE approval for the ProMiniAir *transmitter* containing the Anaren A110LR09x or A1101R09x, or the Ebyte E07-900M10S transceiver daughterboards. The approval numbers are the following:

- FCC ID: X7J-A1107 (Anaren), 2ALPH-E07900M10S (Ebyte)
- IC: 8975A-A11072401

The OScaleDeadRail site discusses antennas that may be used with the ProMiniAir *receiver* in cases where the provided whip antenna is not feasible to use. We do not recommend you replace the approved antenna for the ProMiniAir *transmitter* because it may violate the FCC/IC/ETSI/CE approvals for an "active transmitter."

4 ProMiniAir Settings/Configuration

Note: The instructions here refer to both the ProMiniAir transmitter (Tx) and receiver (Rx). Where necessary, we will try to differentiate between these options with "Tx" or "Rx" for the ProMiniAir transmitter or receiver, respectively.

The ProMiniAir has many default configuration settings that should make it useful "out of the box," and "OPS Mode" (sometimes called "Programming on the Main" or "PoM") re-configuration by the DCC throttle described below can change these settings, perhaps the most important of which is the Airwire Radio Frequency (RF) channel to transmit/receive on. But first, we need to explain how to make the ProMiniAir "listen" to the DCC throttle's re-configuration commands explicitly meant for the ProMiniAir.

While DCC throttles are mostly concerned with commanding the speed, direction, and other behavior of locomotives, they can be used to reconfigure the "decoders" that are busy interpreting the DCC commands sent by the DCC throttle. Usually, reconfiguration involves changing lighting effects and other behavior of devices on a locomotive *at a specific address*, and "OPS mode" is very convenient for doing so. Each DCC throttle manufacturer has a slightly different method for putting the throttle into "OPS mode" so that it can communicate with a decoder at a specific address, but once in this mode, the DCC throttles all send the same NMRA-compliant, DCC packets to reconfigure the recipient decoder by means of changing the *value* of a *Configuration Variable* (CV).

This last point is the source of a lot of confusion. A *Configuration Variable* (CV) is the *fixed address number with a set purpose* where we will deliver a *change in its value held at this address*. We usually refer to the *fixed address with a set purpose* as "CV#", where # is some number. For instance, CV1 holds the value of a device's "short address" (whose *value* can be between 1 and 127). The *value* stored at this address can be changed, so we often refer to the *value* held at the CV address # as "CV#=*value*."

The ProMiniAir "listens" to electrical signal DCC commands from the throttle (Tx) or from a wireless DCC signal (Rx), and if the address of a command matches the ProMiniAir's address, which is 9900(Tx)/9901(Rx) by default (CV17=230, CV18=172(Tx)/173(Rx)), while the throttle is in "OPS mode", the ProMiniAir can be re-configured. *Note: Previous versions of ProMiniAir had default long addresses of 9000(Tx)/9001(Rx) (CV17=227, CV18=40(Tx)/41(Rx))!*

Table 1 provides CVs that can be changed to re-configure the ProMiniAir. To change these values, the throttle that the ProMiniAir connects (Tx) or RF DCC signal listens (Rx) to must first select the current address of the ProMiniAir, which is 9900(Tx)/9901(Rx) by default. Then according to the instructions for the particular throttle, set the throttle in "OPS mode" or "programming-on-the-main" of Configuration Variables (CVs).

Table 1 also provides valid values for the CVs, so if the user attempts to set an invalid value for a CV, the entry will be ignored and will NOT take effect! All changes to CV values made in "OPS mode" are persistent after power-down except for CV254 (RF power level), which will reset to the original default value of 8 upon power-up. This setting has no important effect on the ProMiniAir receiver (Rx). This feature prevents the ProMiniAir transmitter from producing unexpected high-power RF output upon turn-on.

In general, it should not be necessary to change the ProMiniAir's address. But, if the user needs to change the address, the following information will help. It's a little complicated. If the user is not changing from a short to a long address or vice versa, no change in CV29 is needed. The fifth bit of CV29 specifies whether to use a "long" (bit 5=1) or "short" (bit 5=0) address. For the ProMiniAir, no other bits of CV29 are relevant, so either set CV29=0 to use the "short" address specified in CV1 or set CV29=32 to use the "long" address specified by CV17 and CV18.

When resetting the "long address" for the ProMiniAir, the user *must* set CV17 *before* setting CV18! Once the user programs the value of CV18, the ProMiniAir's address will change to the new address, whose value is set by the funny formula:

$$\text{Address} = (\text{CV17value} - 192) * 256 + \text{CV18value},$$

so for the default values for CV17 and CV18 in Table 1: (231-192)*256 + 15/14(Tx/Rx) = 9900(Tx)/9901(Rx). Changing the address by first setting CV17 and then CV18 means that the ProMiniAir will no longer accept "OPS mode" changes at the *old* address, and the user must change the throttle's address to the *new* address before going back into "OPS mode" for any further configuration changes to the ProMiniAir. Similarly, if the user changes the value of CV29, then the address for the ProMiniAir may have changed from long-to-short or short-to-long address, so the user must change the address on the throttle to communicate with the ProMiniAir in "OPS mode" at the *new* address.

It may become necessary to reset the ProMiniAir completely. An example is when power "glitches" corrupt the EEPROM memory during start-up when the ProMiniAir is writing to EEPROM. The user can force a "factory" reset where you return the ProMiniAir to its original settings by setting the throttle or wireless transmitter into "OPS" mode, selecting the ProMiniAir's DCC address, and setting CV8 to 8. The ProMiniAir will ignore other values for CV8. If successfully executed, the ProMiniAir will display a reset message if an OLED is connected. Regardless, *do not turn off the power during reset*, which may last several seconds. Turning off power during reset almost guarantees EEPROM corruption, requiring yet another "factory" reset, so please wait at least ten seconds before turning off the ProMiniAir following a factory reset.

The LCD shown in Figure 12 displays the ProMiniAir's address, Airwire channel number, and power level.

Some final points about the reconfiguration of the ProMiniAir:

- While many DCC throttles have alternative methods for re-configuring decoders, such as "Service Mode" and "Quick Decoder Setup", the only method that the ProMiniAir will respond to is the very simple "OPS Mode" or "programming-on-the-main mode" (POM) that allows specific CVs to be *directly* set.

- Technically the ProMiniAir is not a "multi-function" decoder, but we have chosen the addressing and reconfiguration of the ProMiniAir as if it were one because the user usually becomes quite adept at quickly:

    1. changing the decoder address to the ProMiniAir's,
    2. putting the throttle in "OPS mode",
    3. making CV changes to re-configure the ProMiniAir,
    4. exiting "OPS mode", and finally
    5. changing the DCC throttle address back to whatever locomotive the user is trying to control.

Table 1: ProMiniAir Settings and Configuration

Feature	CV #	Valid CV Vals	Default	Comments
RF Channel	CV255	0–16, 17, 18	0	Airwire Ch. 0-16, S-Cab Ch. 17, EU Ch. 18
RF Power	CV254	0–10	8	Experimentation is required
Locked Anti-phase On/Off	CV253	0 or ≠ 0	1	DCC+/DCC- output type (Rx)
DCC sampling time	CV252	0–255	4	Time in 1/4 s intervals for DCC sampling before turn-off
Transceiver sleep time	CV251	0–255	0	Time in 1/4 s intervals for transceiver sleep
DC Level	CV248	0 or ≠ 0	1	DCC A level when no valid DCC and DCC filtering on
DCC filtering	CV246	0 or ≠ 0	0	DCC filtering OFF (= 0) or ON (≠ 0) (Rx)
Channel wait period	CV245	0–60	1	Channel waiting period in sec (Rx)
Transceiver DEVIATN code*	CV243	0-255	64	The transceiver's GFSK "DEVIATN" value
Number of preamble bits*	CV242	≥ 25	30	Tx only! 0 value uses internal defaults
Timer long counts*	CV241	≥ 67	27	(256-CV241)*0.5uS = DCC 1/2 "zero" duration
Timer short counts*	CV240	0–255	141	(256-CV240)*0.5uS = DCC 1/2 "one" duration
Configuration	CV29	0–255	32	CV29=32 to use long address
Long addr low byte	CV18	0–255	14(Tx), 15(Rx)	Default CV17 & CV18 make Address = 9900(Tx)/9901(Rx)
Long addr high byte	CV17	192–231	231	Program CV17 *before* CV18!
Factory reset	CV8	8	NA	CV8=8 causes "factory" reset
Short addr	CV1	1–127	3	CV29=0 to use short address
*Resets to default on power-up				

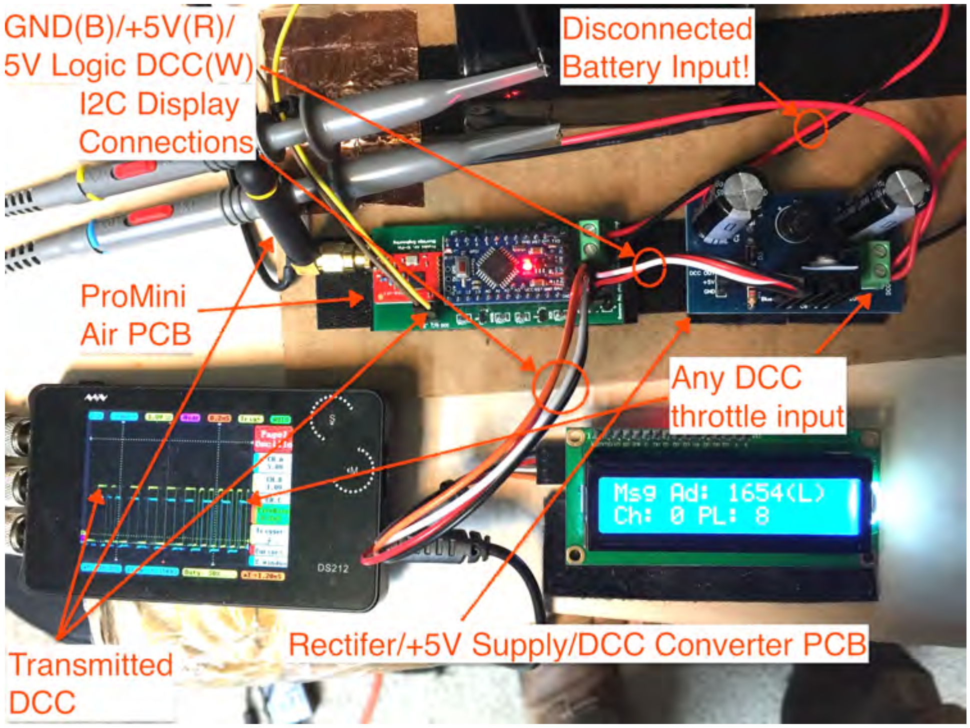


Figure 12: ProMiniAir with an LCD and DCC Converter/+5V Power Supply

5 Integration

To complete the integration of the ProMiniAir receiver (Rx) or transmitter (Tx), you must establish several connections.

5.1 Overview of Connections

The Anaren and Ebyte transceiver daughterboards have a versatile U.FL plug for antenna connections. You can plug in either the Anaren whip antenna we provide or a U.FL-to-SMA or U.FL-to-RP-SMA cable that screws into a remotely-mounted antenna. See Figure 13 for details on these connections.

See Figure 14 for the "ISP" connections that for reprogramming the ProMiniAir. Section A provides details on how to re-program the ProMiniAir.

Which power and data connections you use depends on whether the ProMiniAir will act as a receiver (Rx) or a transmitter (Tx). THERE IS NO PROTECTION AGAINST INCORRECT BATTERY OR EXTERNAL POWER CONNECTIONS!!! You will destroy the ProMiniAir **immediately** if you reverse the GROUND and POSITIVE POWER SUPPLY connection!

We will break down these connections for the ProMiniAir receiver and transmitter in the following two sections.

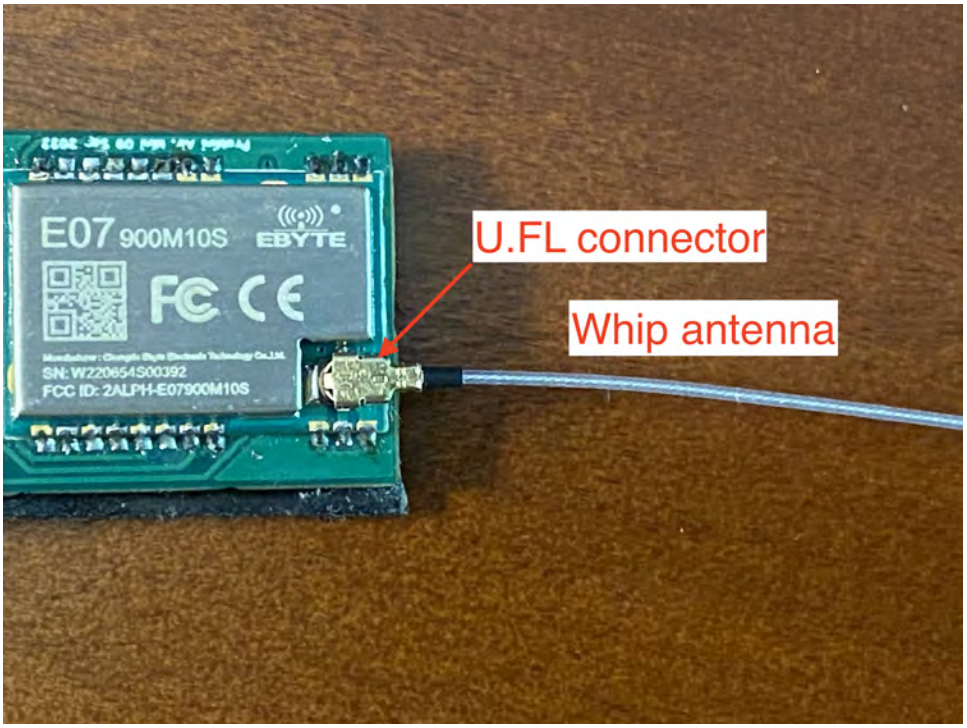


Figure 13: ProMiniAir antenna connector (U.FL)

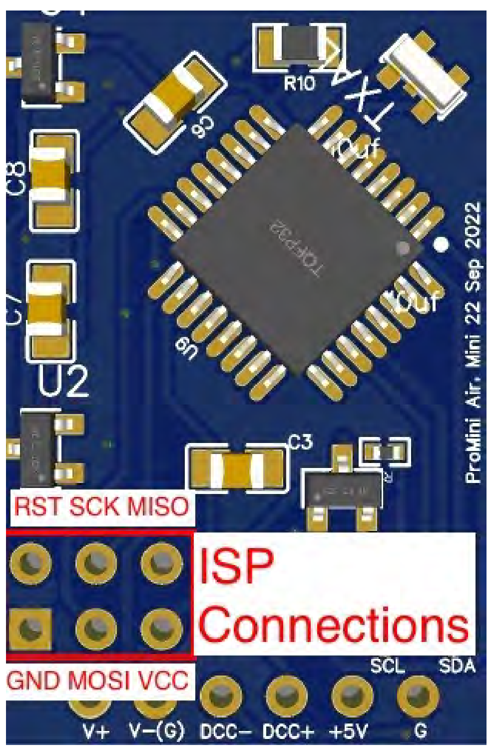


Figure 14: ISP programmer connections for the ProMiniAir Tx and Rx

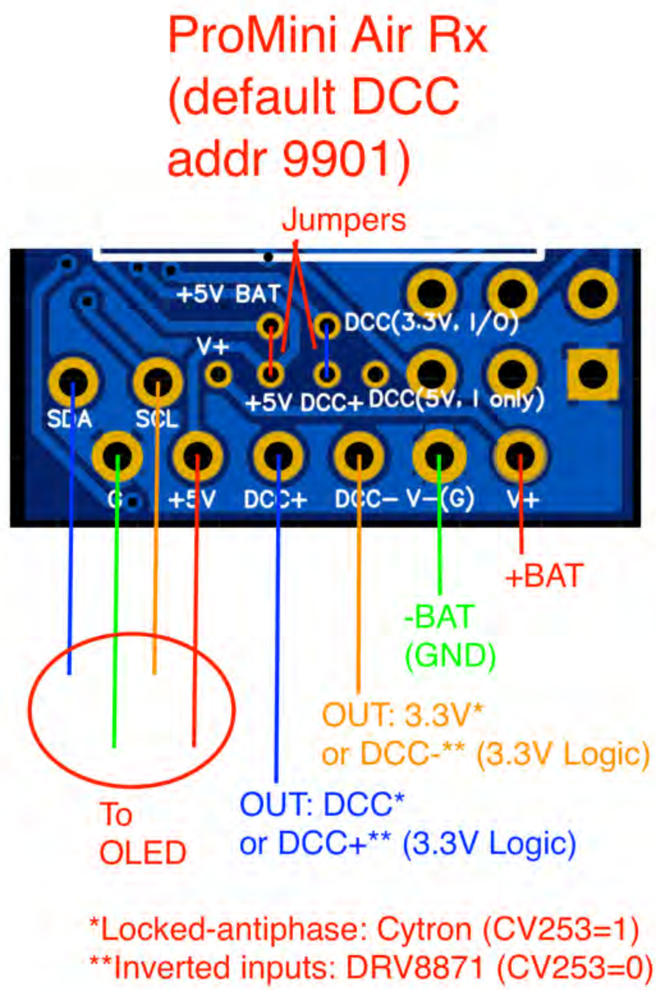


Figure 15: The ProMiniAir receiver's power and data connections

5.2 ProMiniAir Receiver Connections

See Figure 15 for the ProMiniAir receiver's typical power and data connections we use when integrating the ProMiniAir with an amplifier. V+ and V-(GND) are received from the amplifier, which also uses battery power, and the DCC+ and DCC- outputs return to the amplifier. Note that the "+5V" jumper is connected to "+5V BAT" to provide +5VDC to a OLED display from an onboard voltage regulator that converts V+/V- to +5V/GND.

It is possible to use a separate +5VDC/GND voltage source connected to V+ and V-, and in this case the "+5V" jumper is connected to "+V" instead of "+5V BAT" to provide +5VDC/GND output to any connected OLED display as well as powering the onboard electronics.

The ProMiniAir receiver must connect to an external DCC amplifier that converts the DCC+/DCC- 3.3V logic from the ProMiniAir receiver to DCC Track R/L that a DCC decoder requires. This DCC amplifier uses battery power and the DCC+/DCC- inputs from the ProMiniAir receiver to provide the power and DCC messages, coded as a bipolar DCC waveform, to the decoder for both power and DCC messages.

We generally provide one of two "DCC amplifiers" with the ProMiniAir recever:

- The excellent Cytron MD13S high-power (13A continuous) DCC amplifier may be purchased here It uses "locked-antiphase" modulation requiring the ProMiniAir to output 3.3V logic-level DCC on DCC+ and a "high" 3.3V output on DCC-. This output scheme is activated by setting CV253 to 1, the default.

- The AdaFruit DRV8871 or DRV8871 clones are a smaller (1.0"x0.8"x0.4") 3.6A amplifier requiring two 3.3V logic-level DCC inputs that are inverted with respect to each other. This output scheme is activated by setting CV253 to 0. On power-up, some decoders produce large "in-rush" currents because their large "keep-alive" capacitors are in near short-circuit condition before they charge. These in-rush currents will cause the DRV8871 to shut down unless the output current is limited. We insert a 1 ohm resistor in series with the DCC output to limit the current during short-circuit conditions.

- You can successfully use more expensive high-amperage amplifiers (about $30 US as of 2020) found at Pololu here or here. These amplifiers are smaller (0.8" x 1.3") than the Cytron.

Should you ever fully reset the ProMiniAir receiver (by setting CV8 to a value of "8" in PoM mode at the ProMiniAir's DCC address), you will need to set the value of CV253 at the ProMiniAir's DCC address (9901 by default) in PoM mode back to the proper value (which is "1" by default).

5.3 ProMiniAirTransmitter Connections

The ProMiniAir's transmitter connections are shown in Figure 16.

5.4 Integration of the ProMiniAir Receiver into a Locomotive

Of course, the real purpose of the ProMiniAir receiver and DCC amplifier is to integrate them into a locomotive for wireless DCC control using an onboard battery as power. For most installations, all that is necessary is to connect battery power to the DCC amplifier (which in turn supplies power to the ProMiniAir receiver) and the amplifier's DCC output to the on-board DCC decoder.

6 RF Receiver Compatibility

We have successfully tested the ProMiniAir transmitter with several wireless, RF receivers, operating in the 902–928 MHz Industrial, Scientific, and Medical (ISM) band that are designed to interface with onboard DCC decoders:

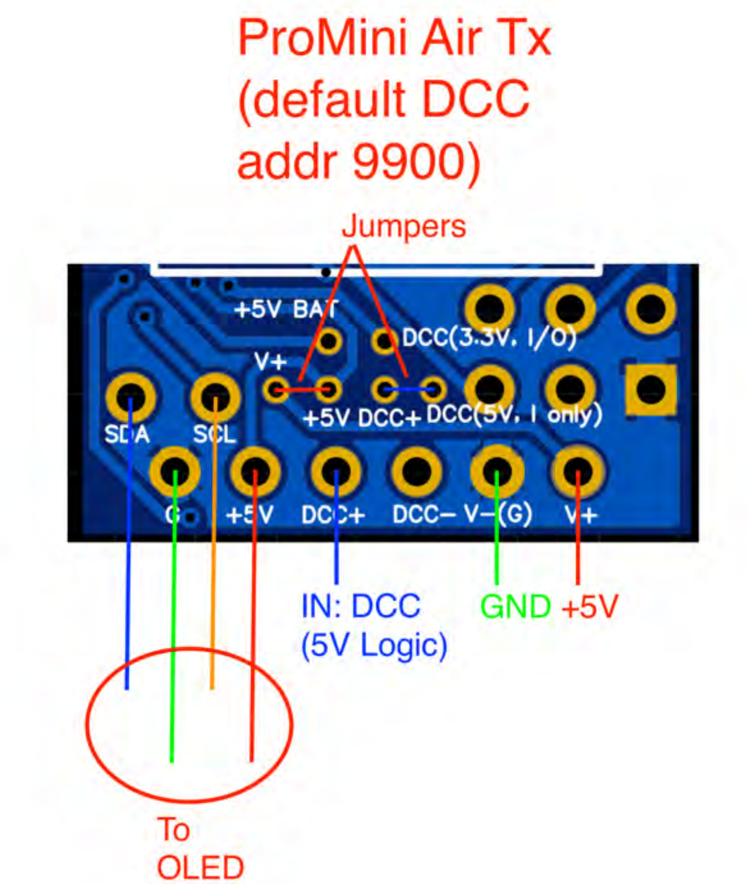


Figure 16: The ProMiniAir transmitter's power and data connections

- Airwire CONVRTR series
- QSI Solutions Gwire
- Tam Valley Depot DRS1, MkIII and MkIV
- Stanton Cab (S-Cab) receivers such as the LXR-DCC
- NCE D13DRJ wireless DCC decoder
- OScaleDeadRail ProMiniAir receiver.

We have also verified that the ProMiniAir transmitter works in the European ISM band at 869.85MHz with the Tam Valley Depot DRS1, Mk IV, receiver. This is Channel 18 on the ProMiniAir transmitter (and receiver).

Table 2: RF Receiver Summary

Name	Channels	Antenna	RF Loss Output	CV27 Relevant?
CONVRTR	0-16	Int or U.FL	DC	Yes
Gwire	0-7	Wire or U.FL	Pulses	No
DRS1, MkIII	16, 17	Wire	Pulses	No
DRS1, MkIV	0-17,18	Int or U.FL	Pulses	No
S-Cab LXR-DCC	17	Int	DC	Yes
NCE D13DRJ	16, 17	Int	Pulses	No
ProMiniAir	0-17,18	SMA or U.FL	DCC Idle/DC*	No/Yes*

Int=Internal, *software configurable

6.1 Airwire CONVRTR Series

The company CVP manufactures and supports its Airwire series of products that include hand-held wireless DCC-compliant throttles (such as the T5000 and T1300) and receivers, such as the CONVRTR series that seamlessly connects to DCC decoders onboard the locomotive. As a general comment, CVP provides excellent, detailed installation and operation documentation and that's in part why they are dominant in some segments of wireless model railroad control. The CONVRTR receiver has some sophisticated features, such as setting its Airwire RF channel purely in software, as described in its User Guide.

However, the CONVRTR interacts with the Airwire wireless throttles in ways that make it challenging to transmit "garden variety" DCC wirelessly to the CONVRTR for proper operation. The Airwire wireless throttles send a non-dcc "cutout" (1/2 "one" and 1/2 "zero") after the end packet bit from the previous DCC packet followed by numerous (30) DCC preamble bits by as a "keep-alive" signal for the CONVRTR. A red LED on the CONVRTR board indicates received signal quality and flickers least when receiving numerous DCC preamble bits with the leading "cutout." The *brightness* of the LED is an indication of received RF power. Typical DCC throttles are not designed with these concerns in mind

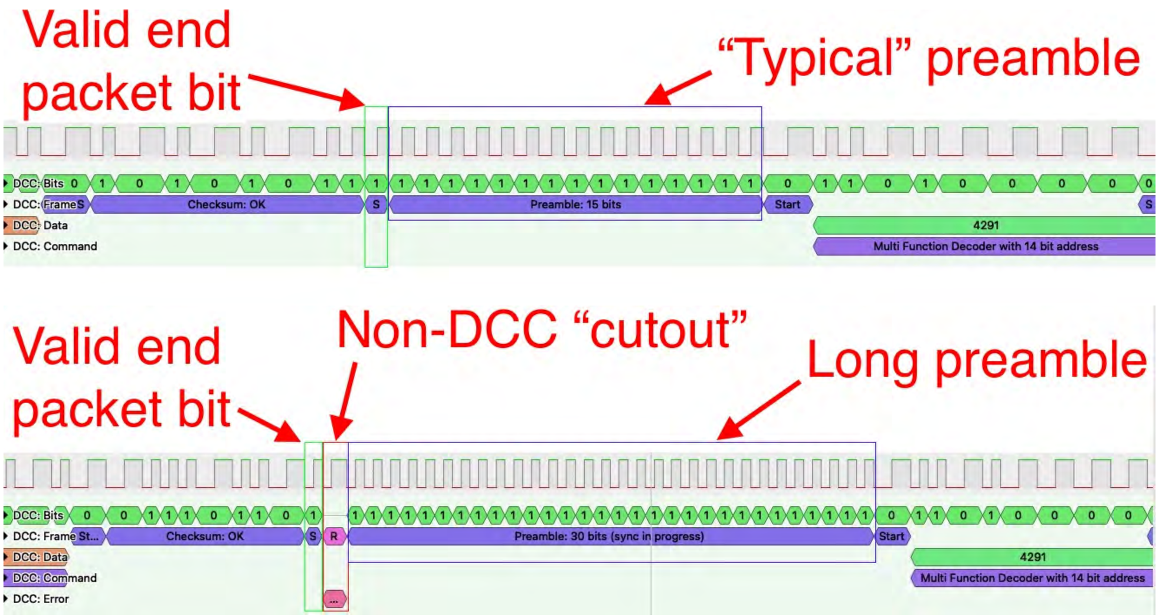


Figure 17: Data collected from a "typical" and Airwire throttle demonstrating the large number of preamble bits and leading "cutout" required to keep Airwire receivers "happy"

and do not output enough DCC preamble bits or the leading "cutout" before the preamble to keep the CONVRTR "happy." See Figure 17 for the unusual features of a Airwire throttle's transmitted DCC.

To combat this problem, the ProMiniAir transmitter software intercepts "garden variety" DCC from the throttle and attempts to interleave a large number (30 or more) of DCC preamble bits with a leading "cutout" to maintain communication with the CONVRTR. Also, the ProMiniAir transmitter inserts additional DCC "idle" messages, which has been observed to to improve Airwire CONVRTR compatibility. Auto-insertion of DCC Idle messages can be turned of by setting the ProMiniAir's CV244 to 0, but doing so will probably interfere with Airwire CONVRTR compatibility.

We caution the user that "Service Mode" commands will be sent wirelessly to *all* "listening" wireless decoders because "Service Mode" does *not* use a specific DCC address! So, all wireless DCC receivers should be turned *off* except for the single "target" for "Service Mode" reprogramming. Note that most decoder reprogramming can be done in "OPS" mode, which will send reprogramming commands to a specific DCC address, so resorting to "Service Mode" should be a last resort.

Like the Gwire receiver below, the Airwire CONVRTR has a U.FL connector for connecting a shielded antenna cable from the receiver to an externally-mounted antenna. An internal antenna option is available as well for CONVRTR mountings that are not surrounded by metal.

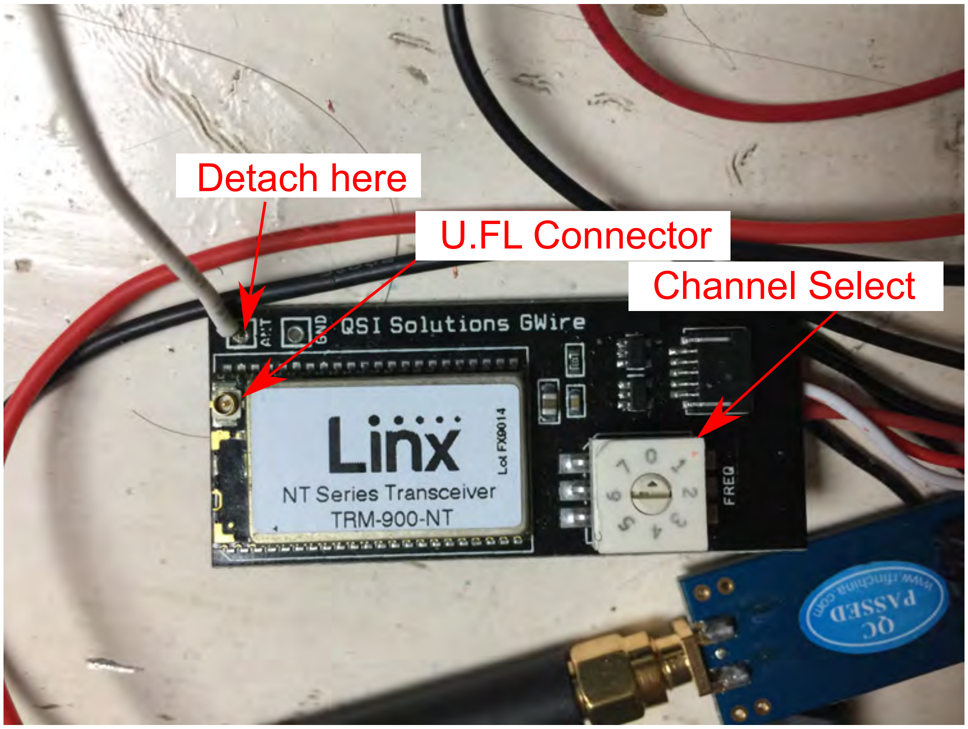


Figure 18: Gwire U.FL connector. If using the U.FL connector, detach the wire antenna.

6.2 QSI Solutions Gwire

The Gwire receiver operates on Airwire RF channels 0-7, which must be selected from a dial on the device itself and is a suitable wireless DCC receiver for the ProMiniAir. A nice feature of this receiver is an onboard U.FL connector (see Figure 18) that allows the user to connect a shielded antenna cable between the receiver and an externally-mounted antenna. The U.FL connector is useful when the antenna must be mounted on the exterior of a metal locomotive or tender shell. See Blueridge Engineering's website here and here for details on how to interface the Gwire to any onboard DCC decoder. The Gwire presents no difficulties for the ProMiniAir transmitter, and you can find it on eBay at reasonably low prices.

6.3 Tam Valley Depot DRS1, MkIII

The DRS1, MkIII receiver operates only on Airwire RF channel 16 (actually, 916.49 MHz, which is close enough to Airwire channel 16 at 916.37 MHz) and makes a suitable wireless DCC receiver. This receiver has a long, single-wire antenna that provides efficient RF reception (see Figure 19). However, the user should place this wire outside any metal shell, which may be inconvenient in some mounting applications. The DRS1, MkIII, presents no difficulties for the ProMiniAir transmitter. The DRS1, MKIV, described in the next section, supersedes this receiver.

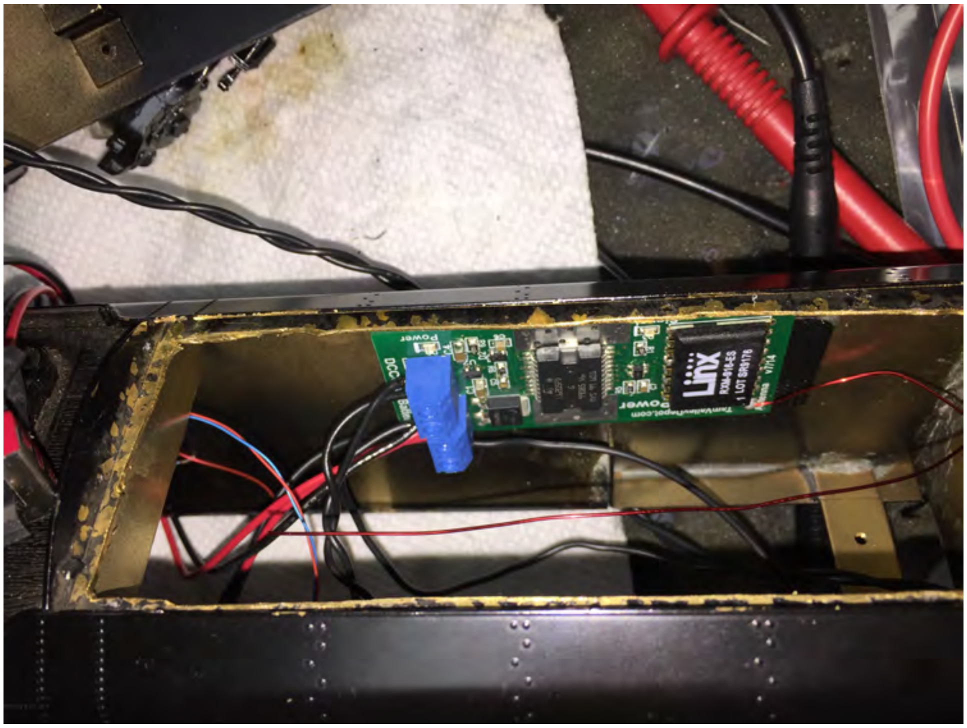


Figure 19: Tam Valley Depot DRS1, MKIII in an open-cavity install. Note the built-in long wire antenna.

6.4 Tam Valley Depot DRS1, MkIV

The DRS1, MkIV, receiver completely departs from the DRS1, MkIII. It operates at the original Tam Valley 916.48 MHz frequency (S-Cab channel 17), Airwire Channels 0-16, and the European ISM band at 869.85MHz (channel "18"). The DRS1, MkIV, comes with either an internal antenna or a U.FL plug connecting to an external antenna. The DRS1, MkIV presents no difficulties for the ProMiniAir transmitter. This receiver is an interesting choice because it changes channels automatically until it finds sufficient RF signal carrying DCC packets. See Figures 20 and 21.

6.5 NCE D13DRJ Wireless Decoder

The NCE D13DRJ, now, sadly, discontinued, is a dead-rail DCC decoder that originally touted compatibility with the Stanton Cab. The decoder's documentation is found here. As can be seen in Figure 22, the decoder is tightly-integrated with the receiver.

The NCE D13DRJ uses the same receiver chip (the Linx RXM-916-ES operating at 916.48MHz) as the older Tam Valley Depot Mk III receiver. The NCE D13DJR has been verified to work with the ProMiniAir transmitter on Airwire Channel 16 (916.37MHz) or Channel 17 (916.48MHz).

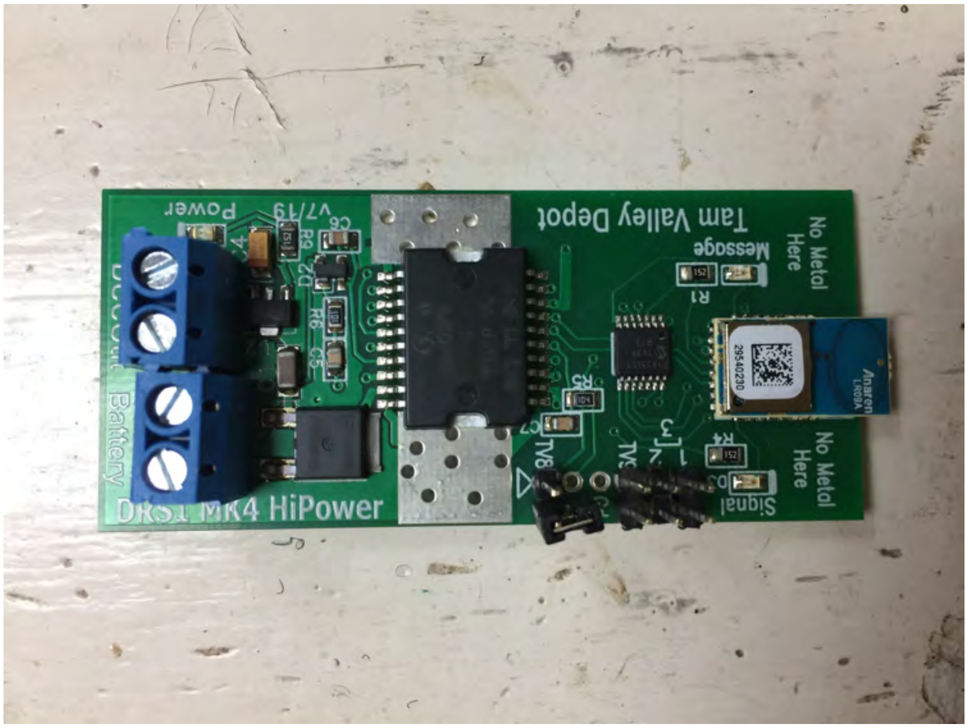


Figure 20: The recently-released Tam Valley Depot DRS1, MkIV receiver. Note the internal antenna on the right side of the board.

6.6 Stanton S-Cab Receivers

The Stanton Cab (or S-Cab) is a series of dead-rail transmitters and receivers developed and sold by dead-rail pioneer Neil Stanton, Ph.D. S-Cab products are available at this site

Stanton offers a hand-held transmitter, the S-Cab Throttle, specifically designed to transmit to S-Cab RF receivers. These receivers include the S-CAB Radio Receiver (LXR-DCC) and Loco Receivers for HO, On3, On30, and some S-scale installations. Also, Stanton will provide an S-Cab receiver coupled with decoders for larger scales. The available options are discussed on the S-Cab website here.

The S-Cab Throttle and receivers operate at 916.48MHz or 918.12MHz (single frequency only!). The former frequency is close to Airwire Channel 16 (916.36MHz), and the latter is the same frequency as Airwire Channel 11. However, Airwire hand-held transmitters *WILL NOT WORK* with S-Cab receivers at either Channel 16 or 11. And Airwire receivers *WILL NOT WORK* with the S-Cab Throttle.

We have devised the RF settings that allow the ProMiniAir transmitter to operate with the S-Cab receivers (such as the LXR-DCC, Figure 23), and these settings are now added an S-Cab compatible Channel 17. This addition required moving the European Channel 17 to Channel 18. See Figure 24 for a demonstration of the ProMiniAir Tx transmitting to the S-Cab LXR-DCC receiver.

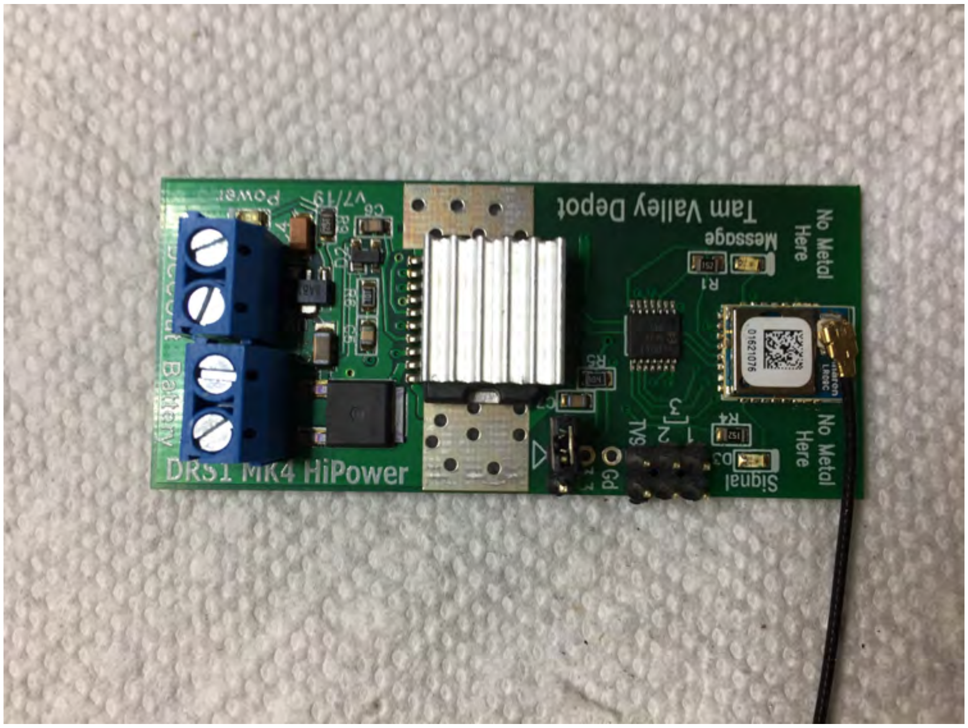


Figure 21: The recently-released Tam Valley Depot DRS1, MkIV receiver with a U.FL plug for connecting to an external antenna.

6.6.1 Operating the S-Cab Throttle with the ProMiniAir Rx

The specialized RF settings for Channel 17 also allow the S-Cab Throttle to transmit to the ProMiniAir receiver with just a tiny wrinkle to establish communication (more about this below).

Since the S-Cab Throttle transmits on Channel, the ProMiniAir Rx must use its automatic “channel search” capability to “find” the S-Cab Throttle’s intermittent transmissions on this Channel with it’s unique RF settings.

The S-Cab Throttle’s intermittent transmissions are where the “wrinkle” occurs. The ProMiniAir Rx’s channel search after power on quickly searches for transmissions in the following channel sequence: 0(A), 18(E), 17 (S-Cab), 1(A), 2(A), 3(A), ..., 16(A), where (A) mean Airwire channel, (E) means European ISM frequency 869.85MHz, and (S-Cab) means for S-Cab at 916.48MHz with other specialized settings.

The S-Cab Throttle’s transmissions are intermittent, so if the operator does nothing, the S-Cab Throttle might not be transmitting in the short time window when the ProMiniAir Rx is looking for transmissions on Channel 17. To force the S-Cab Throttle into nearly continuous transmissions, slide the speed control up and down continuously for several seconds while the ProMiniAir Tx is powering up to guarantee the ProMiniAir Tx has transmissions on Channel 17. If the ProMiniAir Tx does not “sync up” with the S-Cab Throttle, try again by turning the ProMiniAir Tx off and then back on while sliding the S-Cab’s speed control up and down.

Figure 22: The NCE D13DRJ wireless decoder

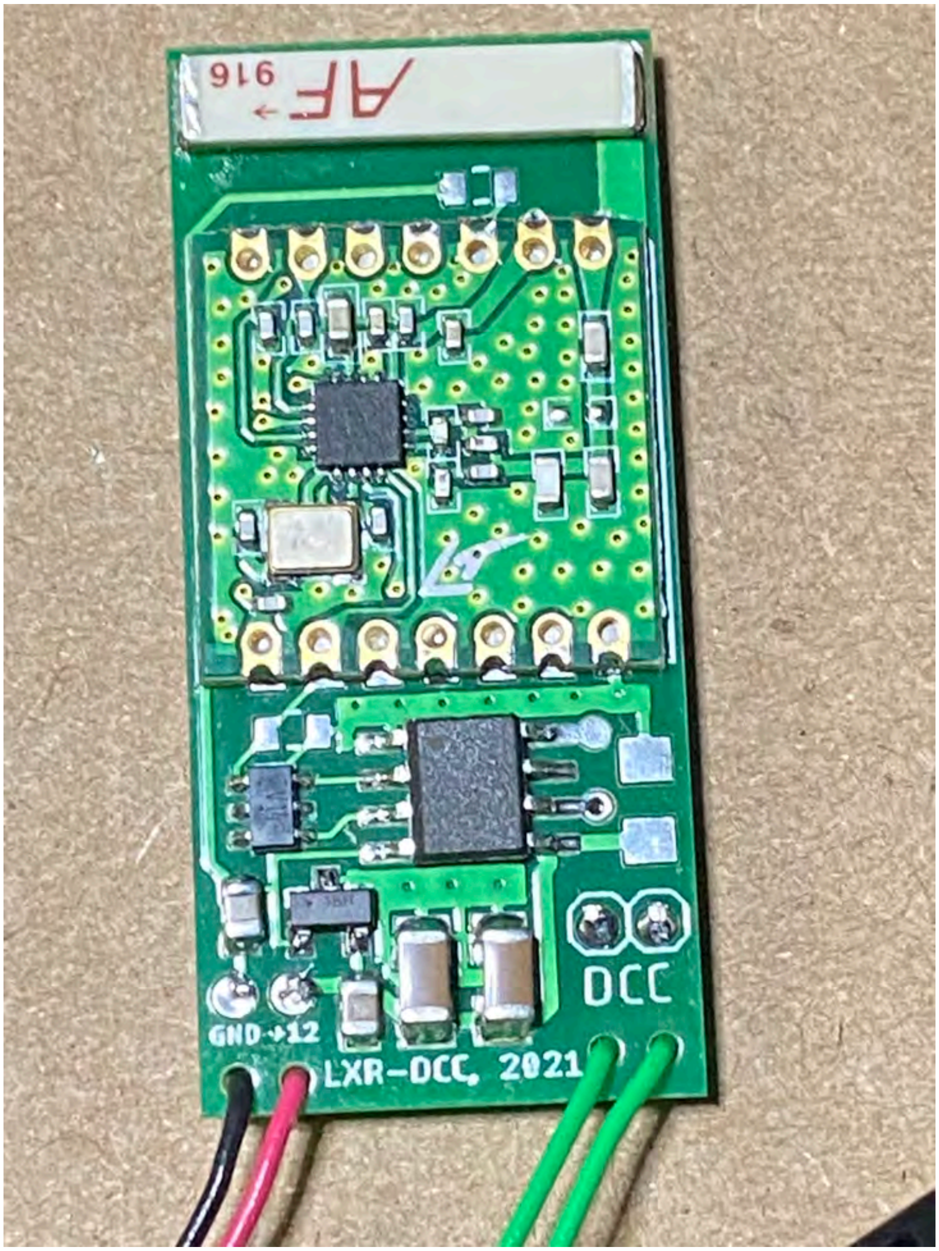


Figure 23: The S-Cab LXR-DCC receiver with a built-in antenna.

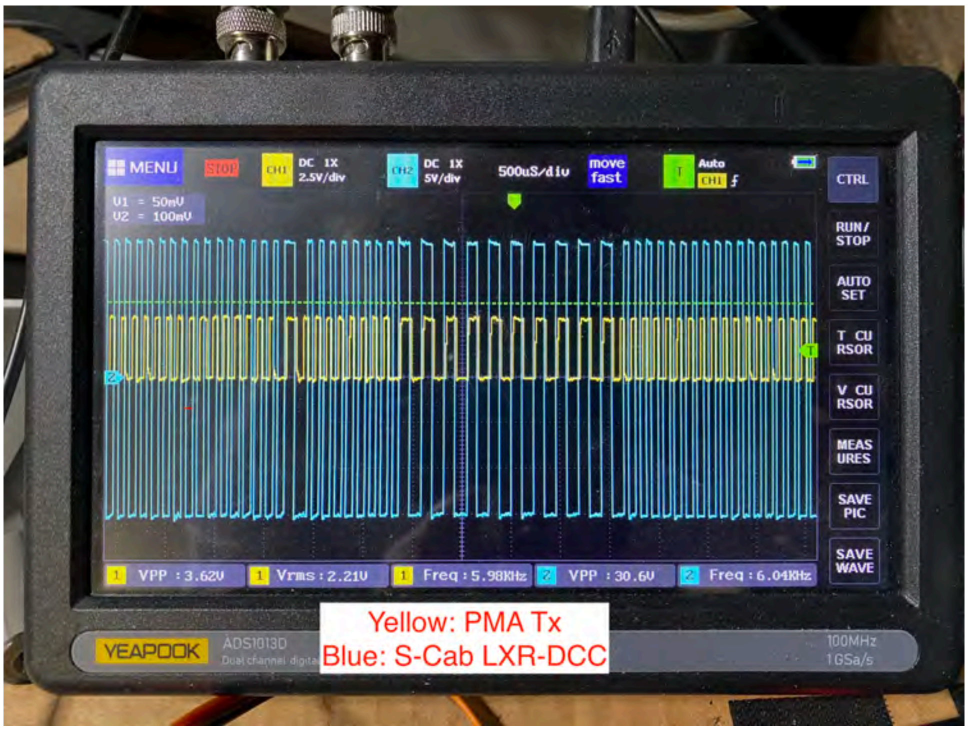


Figure 24: Demonstration that the ProMiniAir transmitter (yellow waveform) successfully transmits to the LXR-DCC receiver (blue waveform) on Channel 17. Note the very slight time delay of the LXR-DCC's waveform.

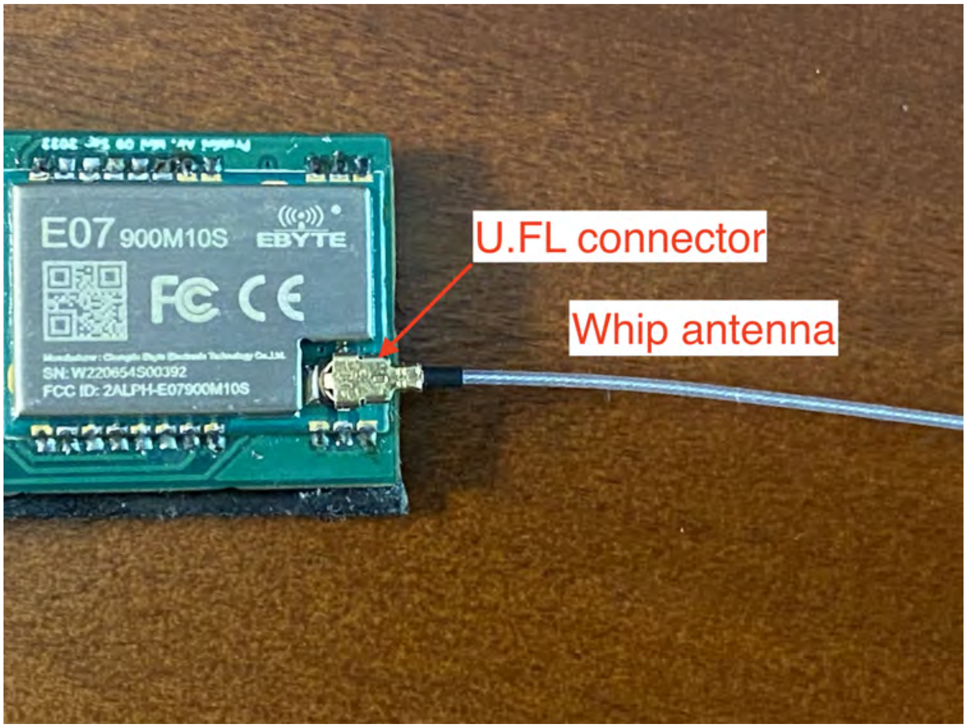


Figure 25: The ProMiniAir receiver with a U.FL antenna.

The Channel 17 settings are devised for S-Cab products operating at 916.48MHz. Contact the author should you need this interoperability at 918.12MHz.

6.7 OScaleDeadRail ProMiniAir Receiver

The inexpensive ProMiniAir receiver presents no issues when used with the ProMiniAir transmitter. This receiver operates on Airwire RF channels 0–16, S-Cab channel 17, or the European ISM band at 869.85MHz (channel "18") and requires a separate amplifier to convert the ProMiniAir's DCC+/DCC- 3.3V logic outputs to bipolar DCC that provides sufficient power to the decoder.

6.8 Receiver and Decoder Behavior with Loss of RF

The designers of various DCC-compatible RF receivers have a several of strategies for what output to provide to the onboard DCC decoders when a valid RF signal is lost:

1. Output the random pulses that the RF receiver naturally outputs when a valid RF signal is lost. This option will cause most DCC decoders to maintain direction and speed while the DCC decoder "sifts" the random pulses searching for valid DCC packets.

2. Output a fixed, positive Direct Current (DC) voltage to one of the DCC decoder's "Track" inputs and a zero voltage DC the other "Track" input when either a) valid

RF signal is lost or b) for some RF receivers such as the Airwire CONVRTR do not receive "keep-alive" DCC packets often enough from the RF transmitter. How the DCC decoder responds to these DC "Track" inputs depends upon DCC decoder configuration and, unfortunately, DCC decoder manufacturer discretion.

3. Output a DCC "Idle" message that keeps the decoder "happy" and continues its current operating state.

Several NMRA-specified *Configuration Variables* (CVs) affect how decoders handle the loss of valid DCC packets, which is important to understand when using the ProMiniAir transmitter because the RF receivers may lose or receive corrupted RF signal from the ProMiniAir transmitter.

The NMRA standard S-9.2.4, Section C "Occurrence of Error Conditions" [3] describes "Multi-Function Digital Decoder shall have a Packet Update time-out value." Further down on line 60, the standard states, "A value of 0 disables the time-out (i.e., the user has chosen not to have a time-out)". This part of the NMRA standard is *not* universally implemented by manufacturers, and it affects how decoders will respond to the loss of RF transmission of DCC packets. To implement this requirement, the NMRA standard NMRA standard S-9.2.2 [2] has defined the "recommended", but *not* "mandatory", CV11, Packet Time-Out Value. A value of CV11=0 is defined to turn off the time-out, but CV11 is frequently *not* implemented.

However, another CV that *is* often implemented addresses some aspects of the loss of DCC. The "optional" CV27, Decoder Automatic Stopping Configuration is under re-evaluation by NMRA, but the NMRA has taken no definite action for some time. Here is what the NMRA standard [2] currently (as of 2019) states about CV27:

> Configuration Variable 27 Decoder Automatic Stopping Configuration
> Used to configure which actions will cause the decoder to stop automatically.
>
> Bit 0 = Enable/Disable Auto Stop in the presence of an asymmetrical DCC signal, which is more positive on the right rail.
> "0" = Disabled "1" = Enabled
>
> Bit 1 = Enable/Disable Auto Stop in the presence of an asymmetrical DCC signal, which is more positive on the left rail.
> "0" = Disabled "1" = Enabled
>
> Bit 2 = Enable/Disable Auto Stop in the presence of an Signal Controlled Influence cutout signal.
> "0" = Disabled "1" = Enabled
>
> Bit 3 = Reserved for Future Use.
>
> Bit 4 = Enable/Disable Auto Stop in the presence of reverse polarity DC.
> "0" = Disabled "1" = Enabled
>
> Bit 5 = Enable/Disable Auto Stop in the presence forward polarity DC.

"0" = Disabled "1" = Enabled

Bits 6-7 = Reserved for future use.

Since DCC decoder manufacturers frequently *do* implement CV27, what electrical output the DCC-compatible RF receiver provides to the DCC decoder upon loss of a valid RF signal will influence how the DCC decoder responds. We will break this down for various brands of DCC-compatible RF receivers *in the 902-928 MHz ISM band* in the following subsections.

Note that some DCC decoders will *not* honor CV27=0; i.e., all auto-stopping features *disabled*. For example, with CV27 set to 0, the Zimo MX-696, and probably other Zimo DCC decoders as well, will continue speed and *forward* direction if *positive* DC level is input to the *"Right Track" DCC input*, and a zero DC level is input to the *"Left Track" DCC input*. Under these "track voltage" conditions, the locomotive will *stop* if originally moving *backward*. Some (but not all) DCC-compatible RF receivers, such as the Airwire CONVRTR, provide these DC inputs if a valid RF signal is lost, but only if connected correctly.

The "correct" connection relates to how the user connects the DCC output from the RF receiver to the "Track Right" and "Track Left" inputs of the DCC decoder. Under normal circumstances, when there is a valid RF signal, which way the DCC decoder connects to the RF receiver does *not* matter. Under the exceptional case of DC-only output by the RF receiver, if it loses a valid RF signal, which way the DCC decoder connects to the RF transmitter *does* matter. The user will likely want the locomotive to continue *forward* with the loss of a valid RF signal, so some experimentation is required to determine which of the RF transmitter DCC outputs should connect to which of the DCC decoder's "Track" inputs to achieve the desired behavior.

As a further complication, we recommend that the user *turn off* the decoder's "analog" mode of operation by setting Bit 2 of CV29 to 0 to force the decoder to use "NMRA Digital Only" control of "Power Source Conversion" (see [2]). If Bit 2 of CV29 *is* set to 1, and again we emphasize the user should probably *not* activate this feature, then "Power Source Conversion Enabled" and then CV12 determines the power source; the most common of which is CV12=1, "Analog Power Conversion."

6.8.1 Airwire CONVRTR Series

When the CONVRTR loses a valid RF signal or receives insufficiently large number of DCC preamble bits with a leading "cutout," it detects these conditions and sends a fixed DC voltage to the decoder. Consequently, the user *should* set CV27 according to the description above.

While it seems that you would want the locomotive to stop if its RF receiver loses a valid RF signal, consider what might happen in tunnels or remote locations. Getting stuck under these circumstances if a valid RF signal is lost is probably not what the user wants, so we strongly suggest that the user set CV27=0.

The user is cautioned, however, that some DCC decoders, such as the new ESU LokSound 5 L DCC, do *not* honor the CV27=0 setting unless the decoder's "Track Right/Left" is connected "correctly" to the CONVRTR's "A/B" output. Experimentation may be required to determine the correct connection, but my experience is the following: CONVRTR A <–> Decoder Track Right & CONVRTR B <–> Decoder Track Left.

6.8.2 Stanton Cab LXR-DCC Receiver

If the S-Cab LXR-DCC loses valid RF reception, it will output a DC voltage. Depending on which of the LXR-DCC's output are connected to which DCC track input to the decoder, this DC voltage will be considered as "positive" or "negative" DC voltage, and the decoder will respond depending on the decoder's configuration. Consequently, like the CVP Airwire CONVRTR, the user *should* set CV27 according to the description above.

6.8.3 NCE D13DRJ

The NCE D13DJR wireless decoder is tightly integrated with its Linx RXM-916-ES receiver. By default, the decoder will maintain direction and speed, and lighting outputs when it loses valid RF signals.

6.8.4 QSI Solutions Gwire and Tam Valley Depot DRS1 Series

The QSI Solutions Gwire and Tam Valley Depot DRS1, MkIII and MkIV DCC-compatible RF receivers will output random pulses to the onboard DCC decoder when a valid RF signal is lost, so setting CV27 is probably of no use. On the "plus" side, most DCC decoders will maintain locomotive direction and speed in the presence of these random pulses since the DCC decoder is actively sorting through these pulses for valid DCC packets, which is usually the behavior the user wants.

This Blueridge Engineering webpage describes how to modify the GWire for use as an RF receiver easily for any onboard DCC decoder.

6.8.5 OScaleDeadRail ProMiniAir Receiver

The OScaleDeadRail ProMiniAir receiver has a default long address of 9901. Like the ProMiniAir transmitter, the ProMiniAir's receiver's channel can be reset in "OPS Mode" by setting CV255 to a value in the range of 0—18. The ProMiniAir receiver has the following options when a valid RF signal is lost:

- Output a DCC "Idle" message that will keep the decoder "happy." The decoder will continue to operate in its current state, including maintaining its current speed and direction.

- Output either fixed positive or negative voltage DC to the onboard DCC decoder: In this case, setting CV27 for the onboard DCC decoder at *its* address *is* relevant. The user can set the ProMiniAir receiver to output fixed DC voltage when it loses a valid RF signal by setting CV246 to 1 in "OPS mode" at the ProMiniAir's address. A positive DC voltage is output by setting the ProMiniAir receiver's CV248 to 1 in "OPS mode" at the ProMiniAir's address or a negative DC voltage is output by setting CV248 to 0. If the user does not want the locomotive to stop with the loss of

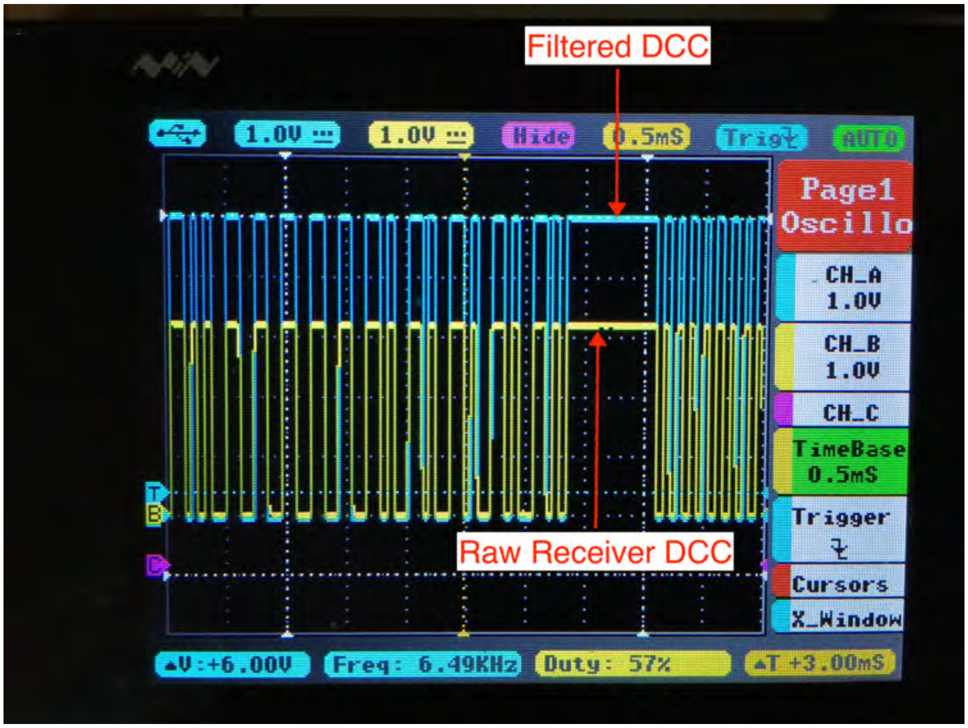


Figure 26: ProMiniAir's raw transceiver and filtered DCC outputs with valid RF DCC signal

a valid RF signal, then set CV27=0 for the onboard DCC decoder at *its* address. Of course, setting CV27 to other values (see above) in the DCC decoder will determine how the DCC decoder responds to the fixed DC voltage that the ProMiniAir outputs to the onboard DCC decoder upon loss of a valid RF signal.

See Figures 26, 27, and 29 for how the ProMiniAir can be configured.

A Firmware Installation

The ProMiniAir Tx and Rx are provided with the firmware already loaded. These instructions are for advanced users who want to update the firmware.

The source code is available from this GitHub site. Locate the source code in a directory where the Arduino IDE can find it. You should retain the subdirectory structure to access the "project" with the Arduino IDE.

Depending on whether you want a transmitter or receiver, edit config.h to select the "define" for the transmitter or receiver.

For a receiver (Rx), config.h should look like this:

Figure 27: ProMiniAir's raw transceiver and filtered DCC outputs (DCC "Idle") with no valid RF DCC signal and filtering off (CV246=0).

Figure 28: This is the output produced by the Gwire and Tam Valley Depot receivers with no valid RF DCC signal.

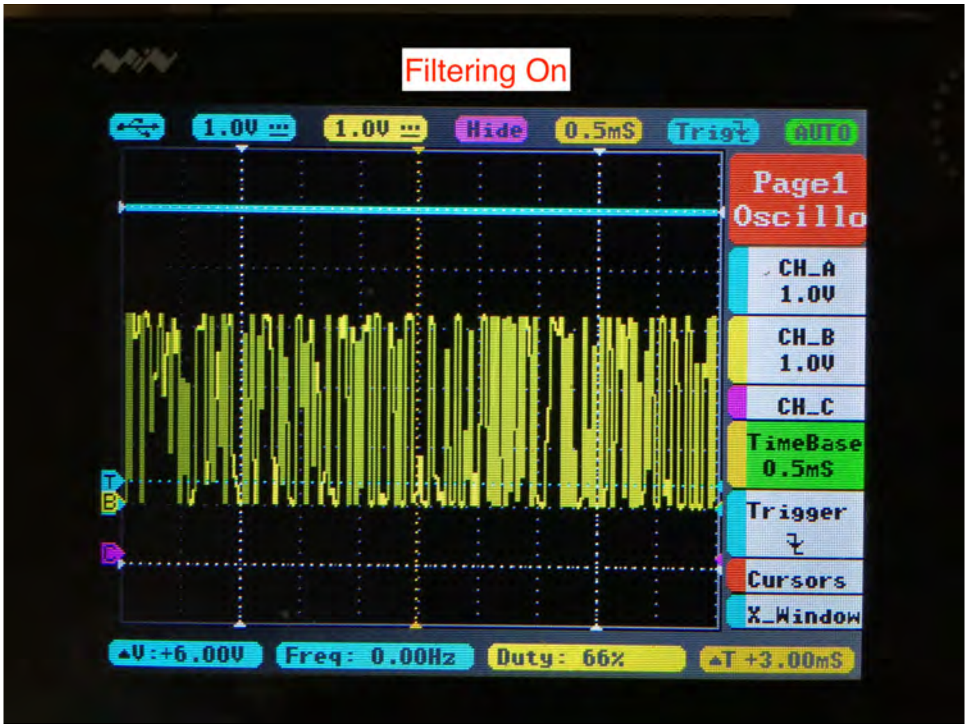


Figure 29: ProMiniAir's raw transceiver and filtered DCC outputs with no valid RF DCC signal and filtering on (CV246=1, CV248=1). This is the output produced by the Airwire CONVRTR as well.

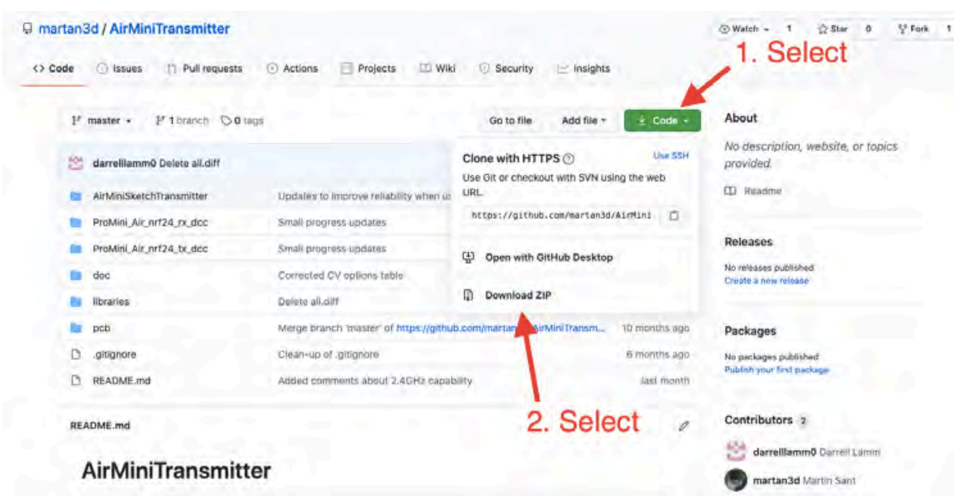


Figure 30: How to download the GitHub zip file that will maintain the directory structure

```
// #define EU_434MHz
/* For World-Wide 2.4GHz ISM band*/
// #define NAEU_2p4GHz

/////////////////////////////
// Set Transmitter or Receiver
/////////////////////////////
/* Uncomment ONLY ONE #define*/
/* For receiver*/
#define RECEIVER
/* For transmitter*/
// #define TRANSMITTER

///////////////////////////////////////////////
// Set the default channel for NA/EU 900MHz only!
///////////////////////////////////////////////
#if defined(NAEU_900MHz)
/* Uncomment ONLY ONE #define*/
/* To set the default to NA channel  0 for 869/915MHz ISM bands only!*/
#define NA_DEFAULT
/* To set the default to EU channel 18 for 869/915MHz ISM bands only!*/
// #define EU_DEFAULT
#endif

//////////////////////////////////////////
// Set the transceiver's crystal frequency
//////////////////////////////////////////
/* Uncomment ONLY ONE #define*/
/* For 27MHz transceivers (e.g., Anaren 869/915MHz (CC110L) and Anaren 869MHz (CC1101) radios)*/
// #define TWENTY_SEVEN_MHZ
/* For 26MHz transceiver (almost all other radios, including Anaren 433MHz (CC1101), 915MHz (CC1101), and 2.4GHz (CC2500) radios)*/
#define TWENTY_SIX_MHZ

...
```

If you want a transmitter (Tx), then config.h should be the following:

```
// #define EU_434MHz
/* For World-Wide 2.4GHz ISM band*/
// #define NAEU_2p4GHz

/////////////////////////////
// Set Transmitter or Receiver
/////////////////////////////
/* Uncomment ONLY ONE #define*/
/* For receiver*/
// #define RECEIVER
/* For transmitter*/
#define TRANSMITTER

///////////////////////////////////////////////
// Set the default channel for NA/EU 900MHz only!
///////////////////////////////////////////////
#if defined(NAEU_900MHz)
/* Uncomment ONLY ONE #define*/
/* To set the default to NA channel  0 for 869/915MHz ISM bands only!*/
#define NA_DEFAULT
/* To set the default to EU channel 18 for 869/915MHz ISM bands only!*/
// #define EU_DEFAULT
#endif

//////////////////////////////////////////
// Set the transceiver's crystal frequency
//////////////////////////////////////////
/* Uncomment ONLY ONE #define*/
/* For 27MHz transceivers (e.g., Anaren 869/915MHz (CC110L) and Anaren 869MHz (CC1101) radios)*/
// #define TWENTY_SEVEN_MHZ
/* For 26MHz transceiver (almost all other radios, including Anaren 433MHz (CC1101), 915MHz (CC1101), and 2.4GHz (CC2500) radios)*/
#define TWENTY_SIX_MHZ

...
```

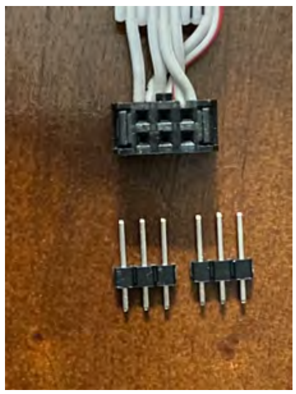
Two further options are available. The first option selects the crystal frequency of the FCC/EC-approved transceiver: 27MHz (Anaren) or 26MHz (Ebyte). The second option specifies North American or European default use.

You load the firmware into the Pro Mini MCU using an "AVR ISP," such as the Sparkfun Pocket AVR Programmer or a less-expensive clone. This "ISP" downloading mode will bypass and erase the bootloader to directly load the firmware into the Pro Mini MCU. On boot-up with the bootloader now erased, the ProMiniAir's MCU will almost instantly supply "3.3V logic DCC" to the DCC amplifier, which provides the DCC decoder with standard DCC waveforms. There is no "boot-up DC" and no need to set CV29, bit2=0. (I set it anyway.) With this solution, all DCC decoders I've tried (ESU, Zimo, MTH) startup without the "boot-up jerk."

Physical connection between the ISP programmer and the ProMiniAir is established using the ISP holes shown below.

This "ISP" form of loading firmware is not as extensively used by folks using the Arduino IDE, but ISP loading is easily accessible within the Arduino IDE. The overly-brief method of ISP programming steps are the following:

1. Insert pins into the ISP's 6-pin output plug. These pins will be inserted into conducting ISP holes in the ProMiniAir's PCB in a later step.



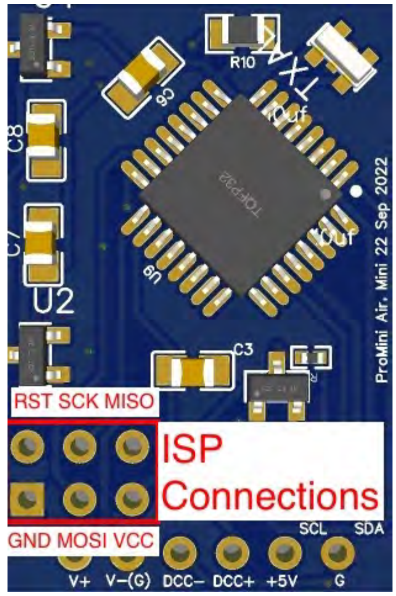

2. From the Arduino IDE, select the 16MHz Atmega328 MCU

3. From the Arduino IDE, Select Tools → Programmer → "USBtinyISP" (or whatever ISP programmer you are using).

4. Select the "AirMiniSketchTransmitter_NMRA" sketch.

5. Connect the USBtinyISP (or other) Programmer (with power switch ON) to the computer using the USB connector.

6. Insert the ISP pins into the ProMiniAir's ISP holes in alignment with the pin-out as shown.



7. Tilt the inserted pins to make firm electrical contact with the sides of the conducting holes and hold them in this position. The plug should be inserted into the “MCU” side as shown.

8. Select Sketch → Upload using a Programmer. The Arduino IDE will compile the sketch and download the resulting firmware to the Pro Mini via the USBtinyISP, bypassing (and erasing) the bootloader.

9. Once download is complete, wait five seconds or so to allow the ProMiniAir to boot up the first time to load the EEPROM, and then release the pins’ contact with the ProMiniAir ISP holes.

We do not provide a specialized ISP connector since establishing good electrical contact by merely tilting the pins works well and keeps the size of the ProMiniAir small and compact.

Once the ProMiniAir receiver or transmitter firmware is installed in the Pro Mini and inserted into the ProMiniAir PCB, the ProMini Air is ready for integration as described in Section 5!

B Regulatory Notices

The following is a required FCC notice:
This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

The following is a required IC notice:
Notice: This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Avis: Cet appareil est conforme avec Industrie Canada RSS standard exempts de licence (s). Son fonctionnement est soumis aux deux conditions suivantes: (1) cet appareil ne peut pas provoquer d'interférences et (2) cet appareil doit accepter toute interférence, y compris les interférences qui peuvent causer un mauvais fonctionnement du dispositif.

References

[1] Anonymous. Communications Standards For Digital Command Control, All Scales. NMRA Standard S9.2, National Model Railroad Association, Inc., National Model Railroad Association, Inc., P.O. Box 1328, Soddy Daisy, TN 37384-1328, July 2004. https://www.nmra.org/sites/default/files/s-92-2004-07.pdf.

[2] Anonymous. Configuration Variables For Digital Command Control, All Scales. NMRA Standard S9.2.2, National Model Railroad Association, Inc., National Model Railroad Association, Inc., P.O. Box 1328, Soddy Daisy, TN 37384-1328, July 2012. https://www.nmra.org/sites/default/files/standards/sandrp/pdf/s-9.2.2_decoder_cvs_2012.07.pdf.

[3] Anonymous. Fail-Safe Operating Characteristics For Digital Command Control, All Scales. NMRA Standard S9.2.4, National Model Railroad Association, Inc., National Model Railroad Association, Inc., P.O. Box 1328, Soddy Daisy, TN 37384-1328, July 2012. https://www.nmra.org/sites/default/files/s-9.2.4_2012_07.pdf.

Abbreviations/Acronyms

DCC Digital Command Control

ISM Industrial, Scientific, and Medical

LCD Liquid Crystal Display

NMRA National Model Railroad Association

OLED Organic Light-Emitting Diode

RF Radio Frequency

Glossary

GFSK Frequency-Shift Keying (FSK) is an encoding technique in which digital information is encoded by a discrete frquency shift in the carrier frequency. In Gaussian Frequency-Shift Keying (GFSK), a temporal Gaussian filter smoothes the driving pulse that controls the frequency shift to reduce "sideband" output. The "DEVIATN" value specifies the frequency shift for the Texas Instruments CC1101 (and CC110L) transceiver chip used in the ProMiniAir.

I2C Inter-Integrated Circuit, sometimes abbreviated "IIC." A synchronous, multi-master, multi-slave, packet switched, single-ended, serial computer bus invented in 1982 by Philips Semiconductor (now NXP Semiconductors).

ISM Band For the purposes of this document and for the US only, the 902–928 MHz RF band reserved for: "Operation of equipment or appliances designed to generate and use locally radio frequency energy for industrial, scientific, medical, domestic or similar purposes, excluding applications in the field of telecommunications." Other ISM bands exist as well, both in the US and internationally.

OPS Mode "Operations Mode" or "Programming-On-the-Main" (PoM). The mode of a DCC throttle that can change the configuration variable (CV) for a DCC decoder at a specific address while the locomotive or device is operating.