

### 

|  |
| --- |
| STARTER SETUP |
|  |

# 

### 

Table of Contents

[**Introduction**](#_gvjwp33234u) **3**

[Welcome](#_lm0ygfgcmwfw) 3

[Our Mission](#_stltgdmjpels) 3

[**Hardware**](#_50ki6l9f46bz) **5**

[The component parts you need to get started](#_4omvgaajnjl1) 5

[**Connecting it all together**](#_3iphn3qkkued) **6**

[A diagram showing how to connect the components](#_akbydkxs3ehq) 6

[**Download Required Resources**](#_tjza92drzko2) **7**

[Arduino IDE](#_ph71vfnih8nr) 7

[Arduino Timers library](#_qgp79u6h5pbj) 7

[DCC++ EX library](#_nxraw0ephi9o) 7

[CommandStation Sketch](#_gkawp9df3qxf) 7

[**Unzip & Move Resources**](#_md4te8gwxeej) **8**

[Unzip & Move the library files to libraries folder](#_ji0rv0tci7vj) 8

[Unzip & Move the Command Station EX files](#_9fxms7tzqg3y) 8

[**Upload Resources to Arduino**](#_fz5s3vf67fuu) **9**

[Uploading the Sketch to Arduino Board](#_n64bz6n3xdx1) 9

[**Testing**](#_5idq3aovdr6u) **11**

[The Serial Monitor](#_egibi37jlrp8) 11

[**For Engineers**](#_oxomoqyqcdit) **12**

[**Troubleshooting:**](#_4hd13hvwspea) **12**

[**For Dev Team:**](#_qeja1hrt0h29) **12**

[**Common Documentation Elements**](#_d2h9hb4m8egw) **14**

### 

# 

# Introduction

## Welcome

Welcome to DCC++ EX, an open-source hardware and software platform for the operation of DCC equipped model railroads based on Greg E. Berman's DCC++ code.

DCC-EX picks up where the original DCC++ left off. By addressing some previous issues, we have released a re-named version of this as DCC++ Classic giving the model railroader everything they enjoyed in the original version - in a stable release.

In addition, we have continued development with an all new version called DCC++ EX [for EXtended] that builds on the DCC++ Classic version and now includes many enhancements and upgrades. We intend to organize this all in one place so that development can continue and the platform can reflect new hardware and technology as it comes within focus.

The Classic version can be found on our Github portal DCC++ EX and is no longer supported. The EX version will soon be available on our [website](https://dcc-ex.com/) where you will find a support link.

## Our Mission

Our mission is to open up the world of model railroading and make it universally accessible.

In order to achieve that, DCC++ EX will provide the model railroad community with a complete, open source DCC (Digital Command Control) system. One that is simple, affordable and expandable, to control model trains on layouts of almosts any size and the associated accessories. Further, it is our goal that this project be organized, documented, and maintained so that it can continue far into the future.

For the moment we are focussing this document on modellers who are new to DCC++ and just want to operate their layouts safely. More experienced modellers will find additional guidance at the end of this document.

This guide will help you through the initial stages of installing and starting to use the new DCC++ EX model railway control system.

It is aimed at someone who is not technically savvy. and will cover the following areas:-

* What equipment you will need to have or buy
* What cables you will need to have or buy
* How to connect it all together
* What power unit you will need and how to connect that
* Where to find the software for the Command Station
* How to instal the software
* Test that you have power to the rails

# 

# 

# 

# 

# 

# 

# Hardware

## The component parts you need to get started

You either need to already have access to or buy

* Command Station : This consists of two parts
  + An Arduino UNO or MEGA
  + A motor shield [also known as a motorboard]
* Controller : There are several ways to control the Command Station
  + The JMRI Train Controlling and Decoder Programming Software
  + The Arduino Serial Monitor
  + A third party throttle
* Track Power Supply :A DC power supply to provide power to the rails
* Computer : You will need a laptop or other computer
  + To download the software that runs the Command Station
  + To operate the layout either
    - By using the JMRI control application
    - Utilising the Arduino Serial Monitor
* The DCC++ EX Software The DCC++ EX software (called a "sketch") is loaded onto the your Arduino command station
* A USB A - B cable They look like this.

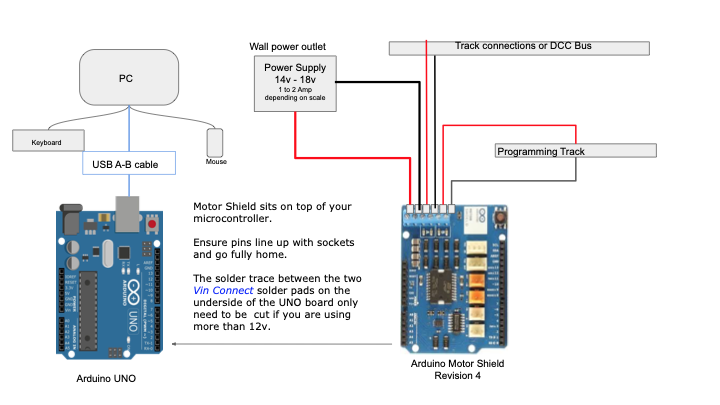


# Connecting it all together

## A diagram showing how to connect the components

Although an Arduino UNO is shown, the set up is the same if you have a MEGA microcontroller.

**Do not connect any of the boards to your computer just yet.**



# Download Required Resources

## Arduino IDE

Download and install the Arduino IDE and Install it on your system. Follow the below link for instructions or Skip this step if you already have it installed.

Link: <https://www.arduino.cc/en/Guide>

|  |  |
| --- | --- |
|  | This is a 192Mb download. |

# 

## Arduino Timers library

Download the Arduino Timers library using this link:

<https://github.com/davidcutting42/ArduinoTimers/archive/master.zip>

## DCC++ EX library

Download the DCC-EX library file from here:

<https://github.com/DCC-EX/CommandStation/archive/master.zip>

## CommandStation Sketch

Download the CommandStation Sketch here:

<https://github.com/DCC-EX/CommandStation-DCC/archive/master.zip>

## 

## 

## 

## 

# Unzip & Move Resources

## Unzip & Move the library files to libraries folder

1. Unzip the two library files (ArduinoTimers-master.zip and CommandStation-Master.zip) one at a time to your desktop or downloads folder. You should then have these two folders extracted:

**ArduinoTimers-Master**

**CommandStation-Master**

1. Rename these two folders to remove the “-master” to look like this:

**ArduinoTimers**

**CommandStation**

1. Select both the above mentioned folders and move them to your Arduino libraries folder, usually:

**\Documents\Arduino\libraries\**

## Unzip & Move the Command Station EX files

1. Unzip the **CommandStation-DCC.zip** file to your download or desktop folder
2. Move it to your Arduino projects folder, which should be:

* **/documents/Arduino**

You may notice other project folders here, and should see the libraries folder where we just copied your libraries to.

1. Rename this folder like you did with the library folders to remove the “-master” so that the folder name is simply, “CommandStation-DCC”.

# Upload Resources to Arduino

## Uploading the Sketch to Arduino Board

1. Open the **Arduino IDE** program from the operating system menu on your computer. Then click on “**File -> Open**” and navigate to your projects folder (/documents/Arduino) and find the CommandStation-DCC folder.
2. Click on that folder to open it and find the main project file. All Arduino main files end in **.ino** (as in Ardu-INO) and should have the Arduino logo on them to advertise their association with the Arduino IDE. Click on “**CommandStation-DCC**” to open the project in the **IDE.** The project will open either in a new window or the same window depending on your settings. If there are two windows, you can close the first window if you like because we won’t need it anymore. You will see tabs for the 3 files used by CommandStation-EX running across the top of the Arduino IDE.
3. Connect a serial cable from your computer to your Arduino Command Station. You should see lights blink on the Command Station board and get an acknowledgement from your computer that you just plugged something in.
4. Select “Tools -> Board” from the Arduino IDE menu and find your board. You should be using one of the following:
   * **Arduino Mega or Mega 2560**
   * **Arduino Uno**
   * **Arduino Nano**
5. Choose the serial port your operating system assigned to your Arduino board under **Tools -> Port**. The IDE should automatically detect the correct port, but you may have to select it. If you have more than one Arduino board plugged in or have more than one you plug in at different times. Once you select a board, the system will remember that port for that Arduino.

|  |  |
| --- | --- |
|  | If you have more than one Arduino board plugged in or have more than one you plug in at different times. Once you select a board, the system will remember that port for that Arduino. |

1. The last step is to upload the sketch to your Arduino. Click the **Upload** button at the top left that looks like a right arrow. It is next to the checkmark button that lets you compile and verify the program separately. This test is conducted before uploading and will flag any errors anyway. Besides, those of us on the DCC++ EX team never make mistakes, so it should compile and upload perfectly! The compile and upload process should take between 30 seconds to2 minutes. Once complete, you should be able to run your DCC++ EX Command Station!

# Testing

## The Serial Monitor

The Arduino IDE has a **Serial monitor** built in that allows us to monitor things connected to the serial port. DCC++ EX has messages and debugging included.

With the Command Station still connected to your computer via the USB cable and the correct port selected, choose **Tools -> Serial Monitor** from the menu. A new window will open and you should see a line of text giving you status information that looks like this:

**DCC++ EX Command Station V2.1.1<NO:SERIAL>**

This lets you know everything is working. As another quick test, after making certain that your connections to the track are correct and that there are no shorts, you can type:

**<1>**

Into the serial monitor command box at the top, to the left of the “send” button and press send. If your motor controller has LED indicators, they should light and you will have power to the track. If you have a locomotive with a sound decoder on the track, it should start to make idle sounds.

To turn off the power, simply enter:

**<0>**

And hit the send button.

To run trains, you will need to connect a controller like a throttle or JMRI.

# 

# 

# For Engineers

Browse the code or use these links to clone the DCC-EX Command Station files. Create repositories:

The command station library file:

<https://github.com/DCC-EX/CommandStation.git#master>

The Arduino Timers libary file:

<https://github.com/davidcutting42/ArduinoTimers.git#master>

The main Command Station Sketch:

<https://github.com/DCC-EX/CommandStation-DCC.git#master>

To clone, use GitHub Desktop or Git Bash. In git bash, navigate to where you want to clone the repository and from the "$" prompt, type  
"**git clone https://github.com/DCC-EX/CommandStation.git**"   
and  
<https://github.com/DCC-EX/CommandStation-DCC.git>

# Troubleshooting:

The Com port under “Tools -> Port” is grayed out.  
You don’t see anything from the serial monitor when you open it

# For Dev Team:

**Question: Can we zip up all the files into one zip file?**

**Question: Are we going to need detailed instructions on how to use an unzip program? Pictures? It will be pages long!**

**Question: What are we going to call these files? How do we refer to the library? DCC-EX Library? DCC-EX CommandStation Library?**

**Question: Is number 7 correct for Mac and Linux? What are their folder paths?**

NOTE: We need to rename the repos. the library has 'master' appended to the end of it. In the Arduino IDE, we unzip and have folders with –master tacked on and that looks terrible. Step number 6 is confusing and we should not have to do it.

**Question: How many definitions do we need? Do we need to say what a library is? What the Arduino IDE is? Etc.**

**My Mega build shows:**

Sketch uses 22656 bytes (8%) of program storage space. Maximum is 253952 bytes.

Global variables use 450 bytes (5%) of dynamic memory, leaving 7742 bytes for local variables. Maximum is 8192 bytes.

**My Uno Build Shows:**

ketch uses 21564 bytes (66%) of program storage space. Maximum is 32256 bytes.

Global variables use 416 bytes (20%) of dynamic memory, leaving 1632 bytes for local variables. Maximum is 2048 bytes.

# Common Documentation Elements

1. Information

|  |  |
| --- | --- |
|  | Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. |

1. Warning

|  |  |
| --- | --- |
|  | Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. |

1. Image

|  |
| --- |
|  |
| Image Caption |