OPEN SOUND DATA  
MANUAL  
~How to use Open Sound Data~

rev.5 (Mar 1, 2020)

INDEX

* [What is Open Sound Data?](#1-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3%) 
  + [1.1 Introduction](#11-%E3%81%AF%E3%81%98%E3%82%81%E3%81%AB)
  + [1.2 Terms of Use](#12-%E5%88%A9%E7%94%A8%E8%A6%8F%E7%B4%84)
  + [1.3. refusal](#13-%E3%81%8A%E3%81%93%E3%81%A8%E3%82%8F)
  + [1.4.list of sound data](#14-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%89) 
    - [1.4.1 Diesel and diesel locomotive](#141-%E3%83%87%E3%82%A3%E3%83%BC%E3%82%B)
    - [1.4.2. locomotive](#142-%E6%A9%9F%E9%96%A2%E8%BB%8A)
    - [1.4.3. JNR/JR Limited Express Type](#143-%E5%9B%BD%E9%89%84%E3%83%BBjr-%E7%8)
    - [1.4.4 Japanese National Railways/JR Commuter Model, etc.](#144-%E5%9B%BD%E9%89%84%E3%83%BBjr-%E9%8)
    - [1.4.5. private railway](#145-%E7%A7%81%E9%89%84)
  + [1.5. mental attitude](#15-%E5%BF%83%E6%A7%8B%E3%81%88)
  + [1.6 What to prepare](#16-%E7%94%A8%E6%84%8F%E3%81%99%E3%82%8B)
* [LokSounddecoder](#2-loksound%E3%83%87%E3%82%B3%E3%83%BC%E) 
  + [What is ESU, the developer?](#21-%E9%96%8B%E7%99%BA%E5%85%83%E3%81%AE)
  + [2.2. History of LokSound](#22-loksound%E3%81%AE%E6%AD%B4%E5%8F%B2)
  + [2.3.LokSound 5 series](#23-loksound-5%E3%82%B7%E3%83%AA%E3%83%B)
  + [2.4 Why use LokSound?](#24-loksound%E3%82%92%E3%81%AA%E3%81%9C%)
* [3.how to useLokProgrammer](#3-lokprogrammer%E3%81%AE%E4%BD%BF%E3%81) 
  + [Download and install the LokProgrammer software.](#31-lokprogrammer%E3%82%BD%E3%83%95%E3%8)
  + [3.2 Start LokProgrammer](#32-lokprogrammer%E3%81%AE%E8%B5%B7%E5%8)
  + [3.3 Connecting the writing device](#33-%E6%9B%B8%E8%BE%BC%E3%81%BF%E8%A3%85)
  + [3.4 Writing Open Sound Data](#34-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3)
  + [Description of the sound editing screen](#35-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%89)
  + [What is a sound slot?](#36-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%89)
  + [3.7 Change the function number](#37-%E3%83%95%E3%82%A1%E3%83%B3%E3%82%AF)
* [4.how to make a sound](#4-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%89%) 
  + [4.1 Introduction](#41-%E3%81%AF%E3%81%98%E3%82%81%E3%81%AB)
  + [4.2 Preparing for the recording](#42-%E5%8F%8E%E9%8C%B2%E3%81%AB%E5%90%91)
  + [4.3 Tips for recording](#43-%E5%8F%8E%E9%8C%B2%E3%81%AE%E3%82%B3)
  + [4.4 Sounds that must be recorded](#44-%E5%8F%8E%E9%8C%B2%E3%81%97%E3%81%AA)
  + [4.5. sound processing and noise reduction](#45-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%89)
  + [4.6 Processing and editing of running sounds](#46-%E8%B5%B0%E8%A1%8C%E9%9F%B3%E3%81%AE)
  + [4.7 Departure Bell Cutout](#47-%E7%99%BA%E8%BB%8A%E3%83%99%E3%83%AB)
  + [4.8. How to make VVVF sound](#48-vvvf%E3%82%B5%E3%82%A6%E3%83%B3%E3%8)
  + [4.9. How to make a diesel sound](#49-%E3%83%87%E3%82%A3%E3%83%BC%E3%82%BC)
  + [4.10. How to use SpectraLayers](#410-spectralayers%E3%81%AE%E4%BD%BF%E3%)
* [5.mounting method](#5-%E6%90%AD%E8%BC%89%E6%96%B9%E6%B3%95) 
  + [5.1 Precautions for installation](#51-%E6%90%AD%E8%BC%89%E3%81%99%E3%82%8B)
  + [5.2 Use the auxiliary board for installation](#52-%E6%90%AD%E8%BC%89%E8%A3%9C%E5%8A%A9)
  + [5.3 DCC Connector](#53-dcc%E3%82%B3%E3%83%8D%E3%82%AF%E3%82)
  + [5.4.AUX,HEAD,TAILwiring](#54-auxheadtail%E9%85%8D%E7%B7%9A) 
    - [5.4.1. HEAD, TAIL in bipolar mode](#541-%E4%B8%A1%E6%A5%B5%E6%80%A7%E3%83%A)
    - [5.4.2. AUX, HEAD, TAIL open collector in DCC standard wiring method](#542-dcc%E6%A8%99%E6%BA%96%E9%85%8D%E7%B)
  + [5.5. KATO HO single train cars KUMOHA40, KIHA110, etc.](#55-kato-ho%E5%8D%98%E8%A1%8C%E8%BB%8A%E)
  + [5.6. KATO HO Power Car, Kiha80, etc.](#56-kato-ho%E5%8B%95%E5%8A%9B%E8%BB%8A%E)
  + [5.7. Tomix HO](#57-tomix-ho)
  + [5.8. Tramway HO](#58-%E3%83%88%E3%83%A9%E3%83%A0%E3%82%A6)
  + [5.9. ENDO HO](#59-%E3%82%A8%E3%83%B3%E3%83%89%E3%82%A6)
  + [5.10. Dentetsu Kobo Plastic HO](#510-%E3%81%A7%E3%82%93%E3%81%A6%E3%81%A)
  + [5.11. Ngauge](#511-n%E3%82%B2%E3%83%BC%E3%82%B8) 
    - [5.11.1. kato 113 series](#5111-kato-113%E7%B3%BB)
    - [5.11.2. KATO C59 DCC sound processing](#5112-kato-c59-dcc%E3%82%B5%E3%82%A6%E3%)
* [6.frequently used settings](#6-%E3%82%88%E3%81%8F%E4%BD%BF%E3%81%86%) 
  + [6.1 Setting frequently used CVs](#61-%E3%82%88%E3%81%8F%E4%BD%BF%E3%81%86)
  + [6.2.1 Setting up RailCom](#62-railcom%E3%81%AE%E8%A8%AD%E5%AE%9A)
* [7.column](#7-%E3%82%B3%E3%83%A9%E3%83%A0) 
  + [How to get started with Open Sound Data](#71-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3)
  + [The Future of Open Sound Data 7.2.](#72-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3)
  + [7.3 Encountering LokSound V3.5](#73-loksound-v35%E3%81%A8%E3%81%AE%E5%87)
* [8. support](#8-%E3%82%B5%E3%83%9D%E3%83%BC%E3%83%88)
* [9. FAQ](#9-faq) 
  + [What manufacturer's command station can I run?](#91-%E3%81%A9%E3%81%AE%E3%83%A1%E3%83%BC)
  + [Will you release sound data for Europe and the United States?](#92-%E6%AC%A7%E7%B1%B3%E3%81%AE%E3%82%B5)
  + [Can open sound data be written to a decoder other than ESU?](#93-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3)
  + [9.4 Can I use LokProgrammer to rewrite the cantum, etc.?](#94-lokprogrammer%E3%81%A7%E3%82%AB%E3%8)
  + [How do I prevent the decoder from breaking?](#95-%E3%83%87%E3%82%B3%E3%83%BC%E3%83%80)
  + [9.6. Jittery movement after writing open sound data](#96-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B3)
  + [The decoder does not work, does not respond, or has broken down.](#97-%E3%83%87%E3%82%B3%E3%83%BC%E3%83%80)
  + [9.8. Can I run it with a PWM power pack?](#98-pwm%E3%83%91%E3%83%AF%E3%83%BC%E3%83)
  + [9.9. What should I do about line voltage?](#99-%E7%B7%9A%E8%B7%AF%E9%9B%BB%E5%9C%A7)
  + [9.10. LokProgrammer does not write sounds properly](#910-lokprogrammer%E3%81%A7%E3%82%B5%E3%)
  + [9.11. Sounds written by LokProgrammer are strange.](#911-lokprogrammer%E3%81%A7%E6%9B%B8%E3%)
  + [9.12. ECoS2 rewrites the address without permission](#912-ecos2%E3%81%A7%E5%8B%9D%E6%89%8B%E3)
  + [9.13. I lost the address.](#913-%E3%82%A2%E3%83%89%E3%83%AC%E3%82%B)
  + [9.14. Some addresses do not work with some command stations](#914-%E3%82%B3%E3%83%9E%E3%83%B3%E3%83%8)
  + [9.15. Is it possible to write sound data for LokSound 5 to LokSound V4?](#915-loksoundv4%E3%81%ABloksound5%E5%90%)
  + [9.16. Is it possible to write sound data for LokSound V4 to LokSound 5?](#916-loksound5%E3%81%ABloksoundv4%E5%90%)
  + [The volume is too low and I want to adjust it.](#917-%E9%9F%B3%E9%87%8F%E3%81%8C%E5%B0%8)
  + [9.18. I want you to provide sound data for \_\_\_\_.](#918-%E2%97%8B%E2%97%8B%E3%81%AE%E3%82%B)
  + [9.19. Can I run it in analog?](#919-%E3%82%A2%E3%83%8A%E3%83%AD%E3%82%B)
  + [9.20. Unable to extract sound data from LokSound decoder](#920-loksound%E3%83%87%E3%82%B3%E3%83%BC)
  + [9.21. There are many LokSound products, which one is the best?](#921-loksound%E3%81%AF%E8%89%B2%E3%80%85)
  + [9.22. Do I need a license with ESU for commercial use with open sound data?](#922-%E3%82%AA%E3%83%BC%E3%83%97%E3%83%B)
  + [9.23. LokSoundV4, which I wrote over a year ago, does not work or produce any sound.](#923-%E5%B9%B4%E4%BB%A5%E4%B8%8A%E5%89%8)
  + [9.24. Do I need permission from the railroad company?](#924-%E9%89%84%E9%81%93%E4%BC%9A%E7%A4%B)
  + [9.25. What is the copyright of the sound data?](#925-%E3%82%B5%E3%82%A6%E3%83%B3%E3%83%8)
  + [9.26. Model stores and model manufacturers. How can we use open sound data to do business?](#926-%E6%A8%A1%E5%9E%8B%E5%BA%97%E3%83%B)
  + [9.27. I heard that LokProgrammer is not available for commercial use?](#927-lokprogrammer%E3%81%AF%E5%95%86%E7%)
* [10.at the end](#10-%E6%9C%80%E5%BE%8C%E3%81%AB) 
  + [10.1. Creator and sound source provider](#101-%E3%82%AF%E3%83%AA%E3%82%A8%E3%82%A)
  + [10.2 Groups and organizations](#102-%E5%9B%A3%E4%BD%93%E3%83%BB%E7%B5%8)
  + [10.3. copyright](#103-%E8%91%97%E4%BD%9Ccopyright)
  + [10.4 MIT License](#104-mit-license)

# What is Open Sound Data?

## 1.1 Introduction

Open Sound Data is a project to release Japanese-style DCC sound data free of charge for ESU's LokSound decoders, which were relatively widespread in Japan, so that all users can work together to boost DCC.

It is not completely free, but the terms of use are relatively relaxed, so that individuals who share the concept of open sound data can enjoy it without problems.

We also have terms and conditions that allow small model railroad stores to do business as well, so that they can use open sound data to develop their business in the long tail.

For more information, please refer to the Open Sound Data web page.

Open Sound Data is independently operated and sponsored by DCC Electronic Arts Coalition members and DesktopStation, with funding provided by DCC Electronic Arts Coalition members and DesktopStation, in cooperation and collaboration with DCC users and clubs in Japan. It is not affiliated in any way, capital or business, with ESU Germany or any of the model manufacturers mentioned in this book.

Open Sound Data:   
https://desktopstation.net/sounds/

## 1.2 Terms of Use

The person who downloads the sound data or receives the data file, or the person who writes the data to the decoder and uses it, is described as the "user.

By downloading or obtaining the sound data published on this page, you agree to these terms of use.

If you do not agree, or if you fail to comply with the terms of use while using Open Sound Data data, you will not be able to use Open Sound Data. You must immediately delete the decoder and all stored data from your computer or storage medium.

* You agree or sympathize with the concept and purpose of Open Sound Data.
* To not commit any act that may cause disadvantage, damage, or discomfort to the administrator of this service, creators, other users, or other third parties.
* Users who comply with the terms of use may write and use the sound data on this page into their own DCC decoders free of charge.
* If you have received this sound data itself for a fee, or a written decoder or vehicle equipped with this data for more than the original price, you must request a refund from the seller. You must also inform DesktopStation of the details of the person who used the sound data for sales purposes.
* Users are allowed to modify the sound data on this page. However, you must contact DesktopStation and distribute the modified sound data on this page to unspecified people or redistribute it via web or other media, except for the services that model railroad stores and model manufacturers can provide as described below. You must also apply the Terms of Service as is.
* Modification of the sound data on this page is permitted, but the modified data alone must not be sold. It is permissible to offer it as a writing service to decoders and to sell the written models. In addition, when selling or providing services, it must be clearly indicated to the customer that the data was created based on open sound data.
* Users (companies, sole proprietors, and individuals) who write sound data to decoders using distributed sound data, or sell decoders or vehicles equipped with decoders as part of their business must contact DesktopStation to obtain permission to start writing services in advance. DesktopStation must be notified in advance of your intention to start the writing service (free of charge), and the price of the writing service must be clearly stated in the store or on the website. In the case of a paid service, the price of the work shall be clearly stated in the store or on the website. DCC reserves the right to deny or revoke permission for the use of sound data if, based on past history, reputation, credit information, or other information, it is determined that the use of the sound data would interfere with DCC activities or be inappropriate in terms of public order and morals.
* Direct links to the sound data files on this page are prohibited. Please make sure to link to this page.
* Reproducing the sound data files on this page on other pages or distributing them on DVDs or other media to an unspecified number of people is prohibited. No permission will be given.
* It is prohibited to point out the illegality of Open Sound Data itself, related data, developers, collaborators, etc. in public without evidence, or to obstruct their activities.
* In the event that a vendor or sole proprietor who misuses sound data does not agree to settle for compensation for the misuse, you agree at the time of downloading or obtaining the data that the vendor's name, address, and representative will be posted in an unspecified location.
* The copyright of the sound data is not waived.
* Acceptance of the fact that no warranty is given for any malfunction, accident, or damage caused by sound data.
* It should be noted that individual support will not be provided.
* These Terms of Use are subject to change without notice or notification.

## 1.3. refusal

The sound data published here is not related to the railroad companies, rolling stock, or other abbreviations or names mentioned. If a trademarked name is listed, that name belongs to the company that owns the trademark rights, and Open Sound Data has nothing to do with it. We have created sounds based on the motifs of the vehicles, eras, and atmospheres described, and we have used many sounds from non-real vehicles and synthesized sounds.

To ensure that the Open Sound Data is free from copyright and neighboring rights, the sound sources are recorded by collaborators, and the processing is done independently. In addition, we have taken great care in creating the data so that it does not conflict with the use of the train melody or the trademark of the railroad company, which are legally valid rights.

The running sounds are not strictly acceleration and deceleration in accordance with real time, and are designed to look good on the model. In addition, there are various formats with different details, such as before and after updates, and the sound has not been created with all of them in mind. If you are concerned about them, please customize and detail them by yourself.

The Copyright Act clearly defines the scope and definition of copyrightable material. Not all sounds are subject to copyright, and there are many that cannot be claimed. For example, buzzers, machine sounds, warning sounds, and non-creative sounds cannot be legally claimed. However, rights other than copyright, such as neighboring rights that accrue to the recording artist, may be relevant. Even in cases where rights are recognized under the law, there is a time limit for claiming the rights. After this period, the rights are lost and third parties can use them freely. These interpretations are explained on the web by patent attorney offices and copyright management organizations. Please refer to them on your own.

* LokSound and LokProgrammer are trademarks of ESU electronic solutions ulm GmbH & Co KG, Germany.
* RailCom is a trademark of Lenz AG, Germany.
* Windows is a trademark of Microsoft Corporation.
* Other product names and company names are trademarks or other rights reserved by their respective companies.

## 1.4. list of sound data

The following sound data is available as of December 2020.

### 1.4.1 Diesel and diesel locomotive

* Series Kiha40 Diesel Car
* DML30HSE Kiha183 Series
* DMH17 Series Kiha82
* DML30HSE Kiha181 Series
* DMH17C Vertical engine (general-purpose) Kiha10/20/55 etc., Kominato Kiha200
* DMH17H Horizontal engine (general-purpose) Kiha81/82/58/52/45/35/23/28 etc., Meitetsu Kiha8000
* DMF15HSA (general-purpose) Kiha40, 47, 48, etc., Kiha183/184
* DMF15HSA late-year one-man specification (general-purpose)
* DMF13HZA Kiha261
* DMF13HZA Kiha110
* NTA-855-R-1 Kiha110
* SA6D125H HOT7000 series
* SA6D125H Kiha120 Series
* SA6D125H Shikoku 2000 Series, N2000
* SA6D140HE Kiha E130 Series, Kiha E120 Series
* NTA-855-R-1 Kiha75 Series
* NTA-855-R-1 Kiha85 Series \* Also suitable for Meitetsu Kiha8500 Series and Aizu Kiha8500 Series.

### 1.4.2. locomotive

* JNR 8620 Steam Locomotive
* JNR 9600 Steam Locomotive
* JNR C56,C12 steam locomotive
* JNR C58 Steam Locomotive
* JNR C57,C59 steam locomotive
* JNR D51 Steam Locomotive
* JNR C61 steam locomotive
* JNR C62 steam locomotive
* JNR DE10 Diesel Locomotive
* JNR Class EF81 Electric Locomotive for both AC and DC
* JRF EH200 class DC electric locomotive
* The railroad in Oigawa, Japan DD20 diesel locomotive

### 1.4.3. JNR/JR Limited Express Type

* MT54, Japanese National Railways Express, Limited Express Type 165, 183 Series, etc.
* JNR Series 781 AC Limited Express Train
* JNR Series 381 DC Limited Express Train
* JRE Series E257-0 VVVF DC Limited Express Train
* JRE Series E259/E657 VVVF DC Express Train/AC Express Train
* JRE Series E353 VVVF DC Limited Express Train
* JRW Series 683 and 289 Limited Express Train
* JRW Series 285 Limited Express Train "Sunrise Express" (Toshiba-IGBT/Mitsubishi-IGBT)
* JRC Series 373, 383 (Toshiba GTO VVVF)
* JRE Series E5 Shinkansen
* JRW Series 500 Shinkansen
* JRW/JRC Series N700A Shinkansen
* JRW/JRC Series 700 Shinkansen

### 1.4.4 Japanese National Railways/JR Commuter Model, etc.

* Old Kokuden Type (Hanging) KUMOHA12,40 etc.
* 4 types of old country suspensions (TDK-528, TDK-544, HS-266-A, DK-91B)
* MT54, JNR suburban type 113, 115 series, 185 series, etc.
* Japan National Railways(JNR) Series 211/213 DC Train
* Japan National Railways(JNR) Series 205-5000 (Toyo IGBT VVVF)
* Japan National Railways(JNR) Series 209
* E231 series
* Series E233 Commuter Type
* Series E233 - Suburban Type
* Series E235 Commuter Model
* JRW Series 207-1000 Toshiba GTO VVVF
* JRW Series 221 field chopper
* JRW Series 223 Hitachi IGBT VVVF
* JRW Series 223 Mitsubishi IGBT VVVF
* JRW Series 223 Toshiba IGBT VVVF
* JRW Series 225 Toyo IGBT VVVF
* JRW Series 321 Toyo IGBT VVVF

### 1.4.5. private railway

* Tokyu Series 1000
* Tokyu Series 8500
* Tokyu Series 7200 & Toyotetsu Series 1800 Toyo Cars
* Tokyu Series 5050, Series 5000 \*Under development
* Tokyu Series 2020, Series 6020, Series 3020 \*Under development
* Tokyu and Izu Express Series 8000
* IZUKYU Series 100 KUMOHA100
* Keio 1000 Series Hitachi 2 Level IGBT VVVF
* Keio 1000 series TOYO IGBT VVVF
* Keio 1000 series(2nd generation) Toyo GTO VVVF
* Keio Type 5100 TDK-544
* Keio Series 3000 Field Chopper Car
* Keikyu Type 600 Toyo GTO VVVF
* Keikyu New 1000 Series 1033F Siemens GTO VVVF (Doremifa Inverter) \*Also available for 2100 Series.
* Keikyu New 1000 Series style Toyo IGBT VVVF \*Under development
* Keisei Type 3700 Toyo GTO VVVF
* Tobu 8000 series
* Tobu Series 6050
* Tobu Series 500 VVVF Limited Express Train
* Tobu Series 7800 Type TDK-544
* Tobu Series 3000 DK-91B
* Tobu Series 3050 HS-266-A
* Tobu Series 3070 TDK-528
* Tobu Series 5050 TDK-544
* Tobu Series 5700 TDK-528
* Hankyu Series 8300 Toyo GTO VVVF
* Kintetsu Limited Express Train MB3127 Early Type General-purpose Sound   
  (for Series   
  12000, 12200, 12400, 12410, 12600 and 30000)
* Kintetsu Limited Express Train MB3127 Late Type General-purpose Sound   
  (for Series   
  12200, 12400, 12410, 12600 and 30000)
* Kintetsu Series 22000 Mitsubishi GTO VVVF Limited Express Train
* Kintetsu Series 22600 and 21020 Mitsubishi IGBT VVVF limited express trains, \*under development
* Kintetsu Resistance-controlled Commuter Cars   
  (1800, 1810, 2400, 2410, 2430, 2444, 2600, 2610, 2800 Series)
* Kintetsu 1000 Series Type TDK-544
* Shizuoka A3000 \*Under development
* Shizuoka 1000 \*Under development
* Nagano Electric Railway Series 2000

## 1.5. mental attitude

Open Sound Data is not a simple and straightforward system to use with digital model railroading using DCC, so even just using it requires a certain amount of knowledge and study. By starting with DCC and open sound data, you will experience a lot of failures.

For this reason, Open Sound Data is not for everyone, nor is it simple enough for everyone to enjoy. It is intended for people who want to enjoy model railroading with gimmicks that have never existed before, who want to make sounds, and who want to pursue the insatiable quest for automatic operation and computer control.

In order to get equipment on a shoestring budget, we have to import it from model railroad stores in Europe and America, using our English skills. If something goes wrong, you will be required to communicate with foreign stores via e-mail. You need to have enough time to enjoy even that.

If you feel that you can never do such a thing, that it is too difficult, or that you don't want to fail, then you will never be able to use Open Sound Data. If you are concerned about the regret of making mistakes, I recommend using an analog-oriented system or another sound system instead of DCC or Open Sound Data.

On the other hand, if you have a positive mindset that failure is for improving your skills, that you don't feel any hurdle in learning, and that you want to enjoy gimmicks and sounds more deeply, Open Sound Data can be a great help.

Are you ready to get your hands on open sound data? If you are ready, let's move on to the next chapter.

## 1.6 What to prepare

Open Sound Data assumes that people who already own DCC command stations, etc. will use it.

* Windows 10 PC (even an inexpensive one is fine)
* LokProgrammer software (Windows only)
* LokProgrammer hardware unit (ESU 53451 or 53452)
* Decoder tester (e.g. ESU or LaisDcc) or vehicle equipped with LokSound5
* LokSound5 decoder (see below)
* AC adapter (e.g. 12V/2A)

# LokSound decoder

## What is ESU, the developer?

The ESU company was founded in 1998 and is headquartered in Ulm, Germany. The official name of the company is Electronic Solutions Ulm, and it is called ESU by its initials.

It used to be an OEM manufacturer of Merklin's controller and decoder parts, and sometimes it was common with ESU ECoS1 in the days of Central Station 1 (CS1).

As of 2020, they do not seem to be active as an OEM manufacturer for Merkurin, but they provide OEM services to small and medium-sized model manufacturers and sell original German-type model railroad cars under the ESU brand with their technical and financial resources.

ESU's products have an established reputation for stability, just like German companies. They are one of the DCC manufacturers that can be used with confidence, as they do not suffer from compatibility problems that were once a problem in Japan.

In Japan, there are several distributors that offer products such as command stations, decoders, and model cars (HO). The vehicles in particular are well known for their flashy gimmicks that are built in from the start, such as sound, uncoupling, raising and lowering of the pantographs, steam, and lights.

## 2.2. History of LokSound

The first product to be released in 1999 was the LokSound 1 (classic). It is described as featuring a small size of 43mm x 16mm. This was followed by LokSound 2 with 1Mbit to 3Mbit ROMs, initially 44mm x 19.5mm in size, and then 36mm x 15.5mm in size with larger capacity and smaller size.

http://www.esu.eu/produkte/fruehere-produkte/fruehere-loksound-decoder/

I think it was the LokSound2 (2001) that started to be introduced in Japan. At that time, it seemed to be highly regarded for its smaller size than SOUNDTRAXX. Then, LokSound 3 was introduced in 2005.

LokSound V3.5 has been introduced in magazines and books in Japan, and some people have obtained it from model stores that import and sell it, or through personal import, and it has become popular among DCC sound users because of its ability to create sound data in block diagrams.

LokSound V4 was introduced in 2011. V4 significantly relaxed memory and sound programming constraints, allowing users to create sound data at a level closer to the current form. Since V3.5 was such a big update, many users were required to recreate their data or relearn. In return, however, it is now possible to reproduce various scenes in high quality sound.

In 2021, an updated version of LokSound 5 will be released, including the Nano series with enhanced features and a smaller size, and a KATO DCC-friendly compatible series.

| **Name** | **time** | **Features** |
| --- | --- | --- |
| LokSound 1 | 1999 | It is called LokSound classic. |
| LokSound 2 | 2002 | High-capacity ROM (1Mbit to 3Mbit); OEM products for Roco also exist. |
| LokSound 3 | 2005 | 16Mbit large capacity ROM, mfx(M4) support. 4 channels of simultaneous sound. |
| LokSound 3.5 | 2006 | 16Mbit large capacity ROM, 4 channels of simultaneous sound. |
| LokSound V4 | 2011 | Major advances in sound programming, 32MBit large-capacity ROM, and 8 simultaneous sound channels. |
| LokSound 5 | 2019 | Faster writing speed, unification of Micro series to Next18, large capacity ROM 128MBit, 10 channels of simultaneous sound. |
| LokSound 5 (Update) | 2021 | The Nano series is now available, DCC friendly, with 12 channels of simultaneous sound. |

## 2.3.LokSound 5 series

The LokSound 5 series was announced in January 2019 and consists of three major, distinct platforms. This division remains the same as before, but improvements have been made, including a smaller size.

The internal performance is basically the same, with the main differences being the output current of the motor and the number of auxiliary signals. The open sound data is designed to be compatible with all LokSound 5 series, so even if the data is set for micro, for example, it can be written to the unmarked version without problems.

The internal architecture has also changed significantly, from the old ATMEL AVR microcontroller to Microchip's SAMD series ARM microcontroller, which offers enhanced performance. The details are ATSAMC21G17A (ARM Cortex-M0+ 48MHz, FLASH 128KB, RAM 16KB) with external winbond W25Q128JV 16MB (128MBit) external SPI FLASH) and DC/DC converter.

As for performance, the number of simultaneous sound channels has been increased to 10, and the sound quality has been greatly improved from 16000 Hz to 32500 Hz. Internal user variables and shift variables have been added, and the range of sound data expression has been enhanced. Furthermore, the writing speed of sound data has been improved to about twice as fast, and the writing time for sound data has been reduced from more than 30 minutes to about 10 minutes, depending on the data, to improve development efficiency.

A DCC-only version that removes the functions of the Merklin Motorola mfx (the name of Merklin's two-way model communication technology, also known as M4) is available at a lower price for the Americas and Oceania. The DCC-only version, which removes the functions of the M4, is available at a low price. This is presumably a pricing strategy to compete with the leading sound DCC manufacturers in the US, such as Soundtraxx and Digitrax. In addition, the unit of acceleration/deceleration time is different from that of the European and global versions, and can be set very long.

| **Art.No** | **Name** | **Scale** | **Capacity** | **Explanation** |
| --- | --- | --- | --- | --- |
| 58410,58412,58416,58419 | LokSound 5 | HO, O | 1.5A | The so-called Muji. |
| 58420,  58429 | LokSound 5 DCC | HO, O | 1.5A | So-called unbranded. This is a product for America and Oceania, and does not have support for mfx, MM2, etc. Instead, it is about 10% cheaper. No speaker is included. |
| 58810,58813,58816,58818 | LokSound 5 micro | N | 0.75A | Ultra-compact, Next18 connector standard. |
| 58820,  58823,  58828 | LokSound 5 micro DCC | N | 0.75A | Ultra-compact and comes standard with Next18 connector. About 10% more expensive than unmarked. This is a product for America and Oceania, and does not support mfx, MM2, etc. Instead, it is about 10% cheaper. No speaker is included. |
| 58315 | LokSound 5 L | O, G | 3.0A | For large capacity motor output, O and G gauges. |
| 58325 | LokSound 5 L DCC | O, G | 3.0A | High capacity motor output. Product for America and Oceania. No speaker is included. |
| 58513,58515 | LokSound 5 XL | O, G | 4.0A | Ultra high capacity motor output. |
| 58731 | LokSound 5 micro DCC Direct Kato Japan | N | 0.9A | DCC friendly, for KATO. |
| 58741 | LokSound 5 micro DCC Direct Kato USA | N | 0.9A | For locomotive mounting |
| 58721 | LokSound 5 micro DCC Direct unmarked | N | 0.9A | For locomotive mounting |
| 58923 | LokSound 5 Nano DCC | z,N,TT | 0.9A | Ultra-compact LokSound5 |
| 58210 | LokSound 5 Fx DCC/MM/SX/M4 | HO | -Mr. | 8-pin NEM652, low-cost version with motor functions removed |
| 58219 | LokSound 5 Fx DCC/MM/SX/M4 | HO | -Mr. | 21MTC, low-cost version with motor functions removed |

The size difference between micro(N) and unmarked(HO) in the LokSound5 decoder is shown below.

| **micro** | **unlabeled** |
| --- | --- |
|  |  |

## 2.4 Why use LokSound?

The reason why Open Sound Data uses ESU's LokSound decoder is because it is the only ideal, well-prepared platform for developing sound data.

The following three points are the main reasons.

* An environment where sound data can be easily developed on a PC (sound data development tools are provided)
* Sound decoder performance, sound quality, and stability of operation are ensured.
* There are more than a certain number of users using it in Japan.

Digitrax and ZIMO also provide a sound development environment, but they do not completely meet these three requirements. Digitrax has good availability because KATO is the distributor, but not all of their products are available. The availability of ZIMO sound decoders is very poor and importing is the first choice.

The fact that it can be developed easily means that even those new to DCC sound can get started and customize it on their own with a little modification work. The concept of open sound data cannot be realized if the data can only be edited and customized by professionals, and ZIMO and Digitrax are implemented by entering scripts, which is a big hurdle for creating sound data.

ESU's products are generally more expensive, but considering the unnecessary troubles and instability that they cause, we believe that they are definitely superior in terms of total satisfaction. Also, if you import them personally, the price difference will not be so great, so Open Sound Data will continue to use ESU's LokSound 5 series as our premise.

LokSound has had users in Japan since around 2001, and by the time LokSound V3.5 was released, it had become so popular that it was introduced in magazines and books. The LokProgrammer software has evolved since that time, but the LokProgrammer hardware, which is the writing device, has not changed and can be used as is.

# 3. how to use LokProgrammer

In this section, I would like to summarize the most frequently used features of LokProgrammer and where they can be found. Note that there are two types of LokProgrammer with the exact same name: the writing device (hardware, ESU) and the editing and writing software (software). Both are used as a set, so don't confuse them.

## Download and install the LokProgrammer software.

Download the LokProgrammer software (free software, no charge) from the ESU page and install it.

LokProgrammer   
http://www.esu.eu/en/downloads/software/lokprogrammer/

Note that you do not need a hardware LokProgrammer (ESU 53451 or 53452) if you are not going to write or test run the program. You only need a computer. There is no need to connect it.

## 3.2 Start LokProgrammer

Immediately after startup, the screen looks like the following. Open a file of open sound data to access various functions. You can create a new file, but it is more difficult, so it is better to use an existing file.

For example, if you open the sound of Kiha110, it will look like this

Open Sound Data Kiha110   
https://desktopstation.net/sounds/osd25.html

The tabs on the left side of the screen have been increased in various ways, which I will explain below. The screen will switch to allow you to edit and configure.

## 3.3 Connecting the writing device

The LokProgrammer writer is a device used to write sounds and settings to the LokSound decoder. LokProgrammer is a device used to write sounds and settings to a LokSound decoder, and is connected between the PC and the decoder.

LokProgrammer writers that are connected via USB cable need to install FTDI's USB serial driver. Normally, the driver is installed automatically when the USB cable is plugged in, and no special operation is required, but it may not be installed automatically in some environments.

In this case, you need to download and install FTDI's USB serial driver by yourself in advance. You can find out how to install and check the driver by searching for it.

FTDI's USB driver page:   
https://www.ftdichip.com/Drivers/VCP.htm

Please note that it will not work just by connecting the USB cable; it will not recognize or write properly unless the AC adapter is connected.

If you forgot to attach the AC adapter, plug in the USB cable, or both, the following message will appear when you try to write sound data in LokProgrammer.

AC adapters are no problem, as long as they are compatible with the Φ2.5mm DC jack and are 12V-16V. The following is a list of adapters available at Akizuki Denshi that have been tested with the LokProgrammer. Please note that these adapters are not recommended by ESU, so please use them at your own risk.

| **manufacturer** | **Type** | **Specifications** | **Sales Location** | **Remarks** |
| --- | --- | --- | --- | --- |
| GO FORWARD ENTERPRISE | GF48-US1240 | DC12V/4A | Akizuki Denshi M-00244 | For Z, N, HO |
| GO FORWARD ENTERPRISE | GF65I-US1640 | DC16V/4A | Akizuki Denshi M-00407 | For HO |
| Adapter Technology | STD-12020U | DC12V/2A | Akizuki Denshi M-06239 | For Z, N, HO |
| XIAMEN UME ELECTRONICS | AD-D120P200 | DC12V/2A | Akizuki Electronics M-10659 Z | For N and HO |

If you have an old LokProgrammer and it doesn't work on Windows 10! If you have an older LokProgrammer that doesn't work on Windows 10, you can use the USB serial adapter from Akizuki Denshi. This is the same one that comes with the latest LokProgrammer. We recommend that you buy it along with the AC adapter at Akizuki Denshi.

USB serial adapter from Akizuki Denshi: http:  
//akizukidenshi.com/catalog/g/gM-08343/

## 3.4 Writing Open Sound Data

On the LokProgrammer screen, download the open sound data in advance, unzip the zip file, and then open the esux file. Connect the LokProgrammer writing device with the USB cable and AC adapter, and press the icon of the following notes in the situation that the power supply is ready, and the sound data and CV setting data will be written.

If the firmware of the LokSound decoder is outdated, the firmware will be written before the sound data is written. If the LokProgrammer is updated, the firmware will be updated again. In some cases, the firmware will be updated.

The Write button on the document icon allows you to write the data set in the Decoder tab.

If the connection to the decoder does not work, an error message will be displayed. Check if the feeder wire is connected, if the wiring is not broken, if it is mounted on a vehicle, if there is a wiring error, or if there is poor contact between the wheel and the rail.

Most of the open sound data is for the LokSound 5 series, and although there is a mixture of data created with LokSound 5 and LokSound 5 micro, it will be automatically converted for the LokSound 5 series. The LokSound5 series will automatically convert the data and write it without problems.

However, it is not possible to write sound data for LokSound 5 to a LokSound V4 series decoder. Also, sound data cannot be read out from the decoder. Only the setting values can be retrieved.

## Description of the sound editing screen

Click on the Sound tab, and you will see a screen like the one below. This screen is the one you will use most often for sound editing.

## What is a sound slot?

Think of a sound slot as a channel for sound data, and LokSound5 can play sounds simultaneously using 10 sound slots. Various sounds can be layered on top of the sound of the MG, blower, and running sounds to better reproduce sounds and situations.

Various sounds can be registered individually in the sound slot and played according to conditions to express the movement of the vehicle. The following is an example of a VVVF sound slot.

Note that only one sound can be played at a time in a single sound slot. When a sound slot finishes playing, you can move on to the next block to play a different sound. Therefore, if you want to play multiple sounds at the same time, each block can be configured to associate other sound slots, which can then be used to call up other sound slots.

## 3.7 Change the function number

You may want to change the pre-defined function numbers (F0, F1, F2, etc.) in the open sound data, and LokProgrammer will show you how to do it.

First, open the sound data of the open sound data you want to change in LokProgrammer.

From the Decoder tab, click on Function Mapping to bring up a list of functions. This screen is an important setting area where you can freely assign sound functions, auxiliary outputs, and internal special functions to function numbers.

If you press the pull-down on the function number you want to change, you'll see Driving, Direction, and many others. This is shown in the table of explanation below.

| **Item Name** | **Explanation** |
| --- | --- |
| Driving | In operation or stopped? |
| Direction | Whether the direction of travel is Fwd or Rev. |
| F0 | Function 0 (F0) setting or |
| F1 | Function 1 (F1) setting or |
| F2-F28 | Function 2 (F2) to F28 or |

The actual screen is shown below.

Each function number can be set as valid or invalid. Multiple functions can be linked together to create separate conditions. The conditions can be further divided into whether the vehicle is running or not, and which direction it is traveling.

As for the settings, the following items can be selected, and their meanings are as follows

| **Choice item name** | **Explanation** |
| --- | --- |
| Ignore | Ignore (disabled) |
| On | Valid when this function number is turned ON. |
| Off | Valid when this function number is OFF. |
| Yes | Valid only for Driving items. Valid while driving. |
| No | Valid only for Driving items. Valid when the car is stopped. |
| Forward | Valid only for the Direction item. Valid when the direction of travel is forward. |
| Reverse | Valid only for the Direction item. Valid when the direction of travel is backward. |

Normally, you only need to set On, but if you want to reproduce, for example, the changeover indicator lights of an electric locomotive, you will need to set conditions that take into account the direction of travel and other function states while operating multiple auxiliary signals. You can set various conditions to match the movement of the function you want to create.

After completing the function mapping settings, write to the decoder using the Write only setting data button below. The sound data will not be written (or changed), and the writing process will end immediately. If you want to write sound data together, select the note icon on the right to write both the changed function mapping data and the sound data.

# 4. how to make a sound

## 4.1 Introduction

In this section, I will list the things you will need to create a sound, how to get the equipment (how to import it), and the means and tips for recording. The things you will need are as follows

**What to use for editing: 1.**

* Windows PC (for Mac users, please use Bootcamp, Parallels, or other virtual PC software)
* [LokProgrammer](http://www.esu.eu/en/downloads/software/lokprogrammer/) (software, free)
* [LokProgrammer](http://www.esu.eu/en/products/lokprogrammer/) (hardware, imported, about 15,000 yen)
* [LokSound5decoder](http://www.esu.eu/en/products/loksound/) (micro is OK)
* Decoder tester (ESU, LaisDcc, whatever)
* [Audacity](https://www.audacityteam.org/) (sound editing software, free, or others if you prefer)
* SpectraLayers Sound editing software. You can erase specific sounds with pinpoint accuracy.

You need to get a LokProgrammer, LokSound, and a decoder tester to get started. Most people who use [Open Sound Data](https://desktopstation.net/sounds/) probably have all of these, but if you don't, I recommend importing them.

The following two stores are often used by DCC enthusiasts in Japan.

* [Model Barn Shop](https://www.modellbahnshop-lippe.com/Digital/Digital+boxes/ESU-53451/gb/modell_4042.html)lippehttps://www.modellbahnshop-lippe.com/Digital/Digital+boxes/ESU-53451/gb/modell\_4042.html
* [EURO LOK](https://www.tee-usa.com/store/product3714.html)   
  SHOPhttps://www.tee-usa.com/store/product3714.html

## 4.2 Preparing for the recording

Don't enter in a form. The most important thing is technique and know-how. That is more important than equipment. Do you have a smartphone? To be frank, you can use **a single smartphone for** recording. However, it is better to have a good external microphone for your phone, because it is difficult to record and it is prone to noise and wind noise. I heard that the sound of [Kiha261 was](https://desktopstation.net/sounds/osd23.html) also recorded with an iPhone and an external microphone.

Be sure to set the recording to the highest quality. If you record in low quality, it will be almost useless.

However, if you have a good recorder, that's the best way to go. When I look around, I get the impression that most people use TASCAM. I think the most important thing is a microphone with a windjammer, rather than a recorder.

Don't get bogged down in formality, but start by trying to record with your own equipment or a windproof microphone that you can buy for a little money. If you can afford it, you can buy a recorder.

## 4.3 Tips for recording

* Watch out for wind noise.
* Definitely use a windproof microphone.

There is a huge difference between a microphone with and without a wind shield (like a fluffy cat hair). If you listen to a recording made without a wind shield, you will hear so much wind noise that it will be useless. If you don't have a wind shield, just covering the microphone with a towel or handkerchief will also make a difference.

Although not the same as wind noise, in some cases, a microphone with high sensitivity can also pick up the sound of touching the microphone. In this case, it is necessary to avoid touching the microphone as much as possible, or to attach a sponge or soundproof sheet to reduce the touching sound. The microphone that comes with the recorder is protected against this, but be careful with external microphones that are very expensive.

**Getting as close as possible to where the sound comes from (from YOMIX's recording technique)**   
http://blog.livedoor.jp/yomi\_tetu/archives/5467087.html  
  
 Sound attenuates with the square of the distance, so I tried to be aware of that. It's hard to get a clear engine sound unless the windows are open. Moreover, the square of the distance means that the engine sound is almost inaudible at the edge of the car, away from the engine. However, directly above the engine, the sound is blocked by the floor, making it almost impossible to hear anything above the kHz order. In this case, the car does not have a window, so after much deliberation, I decided to record directly above the engine. As a result, the closer distance won out and we were able to record the turbine sound. By the way, I said "directly above the engine," but I really shot on the floor of the seat directly above the engine. The microphone was placed directly on the floor in the space under the seat, and the space was sealed with a bag. This allowed me to reduce the sound of the air conditioner, the rustling inside the car, and the broadcast. I was able to record the sound of the engine and turbine with the maximum signal-to-noise ratio that an individual can achieve.

The farther away you get, the louder the sound gets. Try to find a place to record as close as possible while keeping safety as the top priority.

For example, in-train announcements are recorded by placing the microphone right in front of the speaker. For under-floor noises, after taking a seat near the motor in the case of VVVF, or near the engine in the case of diesel, cover the microphone with a bag and hold it down so that it does not pick up ambient sounds.

For SIV and compressor sounds, it is better to record them from the street if there is a road nearby, rather than from the station platform, to get closer and get better quality.

* I will endure again and again

It's not uncommon for people around you to cough. It's also normal for the sound to be covered by a train coming from the opposite platform. There's no such thing as a one-shot deal.

## 4.4 Sounds that must be recorded

The following is a list of what to record and what you will need. Note that you should record the same sound as many times as possible. In many cases, there are only a few that are usable properly.

**Items to be recorded in the car**

* In-train announcements
* Running sounds in the train (stopping - accelerating - coasting - decelerating - stopping)
* Door opening and closing sounds
* Sounds of signal systems such as ATS and ATC (driver's seat)

**Record at a station** (if possible, record at an open station, not in a tunnel or at a station built by excavation)

* Subfloor boogie loosening, loosening sound
* brake noise
* Compressor noise
* The Sound of SIV
* Station Announcements

Make sure you know what you're doing, such as recording from the street or looking for the shortest speaker in the station. Some people use selfie sticks to record, but this is absolutely not allowed. You may get an electric shock from hitting an overhead wire. Do not cause any trouble to the railroad company. People who cause trouble are not qualified to use open sound data.

**Recorded at garages and stations that stay overnight.**

* Panters up and down
* Startup sound, power-off sound

The next step is to edit the sound. This is the process of removing noise or making a small sound louder. The main software used is Audacity. It is a fact that most of the DCC sound users use Audacity.

Audacity  
https://www.audacityteam.org/

## 4.5. sound processing and noise reduction

Let's talk about the processing of the recorded sounds. The processing involves the following tasks

* Extract the sound you need.
* Eliminate unwanted sounds (noise reduction process, high-pass filter, low-pass filter, etc.)
* Correct the loudness of the sound (amplification, normalization)
* Create sound loops (whistle, SIV, bell, diesel idle sound, blowing up sound, etc.)

These can be done with Audacity, the free sound editing software I mentioned earlier.

I hope you will look up the basic operations on your own, such as the introductory site. Here are the filters I often use.

In particular, I use the "noise reduction" function. For example, suppose there is a door opening and closing noise as shown below.

When you want to eliminate white noise or background sound, select the area where the background sound enters as shown below.

Select Noise Reduction to register the selected area as noise.

Next, select the areas where you want to reduce the noise. Here it's all of them. The background sound is intruding into the entire sound of the door opening. By removing this sound, we can extract only the sound of the door opening.

Call up Noise Reduction again and make adjustments. Adjust the level of noise reduction while checking the preview. If you apply too much, the sound will become tinny and strange, so aim for the very edge and keep readjusting.

Pressing OK on the noise reduction process will reduce the noise as shown below. You can see that the background sound has been reduced and cleaned up.

Cut off the front and back, and adjust the length so that the sound is only for door opening.

If you save the file in WAV format, you can register it in LokProgrammer and use the DCC decoder to play the sound.

## 4.6 Processing and editing of running sounds

First, the running sound. The following is an example of YOMIX's diesel sound (Kiha261).

Next is an example of a Tokyu 50x0 train, given to me by Kawaii-san.

The approach to creating a diesel and a train is completely different.

In a diesel, the sound of the engine blowing, the sound of gear changes, and the sound of the turbo change changes with each gear shift, so the sound is processed so that the combination of these sounds is switched in sequence. The key is to play with the sound in a natural way after the gearshift is switched. For coasting, the only sound is the rattle of the idle sound, so I added the sound of running wind and rail joints to make it sound natural. For deceleration, the main sounds are the squeak of the brakes and the sound of the engine brakes, so you can overlay those sounds.

In the case of trains, the sound of both acceleration and deceleration is created by chopping up the sound into six or seven stages; VVVF has a unique sound even in deceleration, so the deceleration sound must be created properly so that it sounds according to the speed.

The approach to making and editing is different, so be careful about that. I will now explain how to chop up the diesel running sound.

## 4.7. departure bell cutout

All of a sudden, it got complicated by talking about the sound of running diesel, so I'll take a short break.

Here I would like to explain the know-how of cutting out the departure bell. This is also the sound of the departure bell of Mr. Kawaii's Tokyu train.

5050\_Departure\_Bell\_20200423.wav http://buin2gou.sakura.ne.jp/sblo\_files/powerele/image/5050\_E799BAE8BB8AE38399E383AB\_20200423.wav

First, we will separate the contents of the departure bell with three labels, start, loop, and end, as appropriate.

From here, we will search for looped sounds (a range of sounds that are played repeatedly) and cut them out.

First of all, if you look at the waveform carefully, you will see similar shapes repeated over and over again. In other words, if you can determine the range of similarities, you can continue to make the same sound.

First, we will examine the area between start and loop. Look at the point where the amplitude of the sound is the smallest, and zoom in on that. Then, we found the following part. Adjust the zero-crossing part here by shifting its position so that it is at the border between the start and loop labels. You can do this by D&D on the circle.

In the same way, look for the border between the loop and end labels. Make sure to determine whether it is above or below the wave, so that it connects nicely with the border between start and loop that we just determined. In this case, we will cut at the top, where the amplitude is zero.

Select the loop range and press Shift+Space to play the loop.

Once you have it the way you want it, export the multiple labels and save the file. I was able to export a file that looks like this

bell.  
ziphttp://buin2gou.sakura.ne.jp/sblo\_files/powerele/image/bell.zip

## 4.8. How to make VVVF sound

First, I would like to explain the VVVF sound editing next.

The driving sound in the open sound data is made up of three major components: acceleration, deceleration, and driving sound (wind noise).

The sound of the running wind will be lower or higher depending on the speed. This is set to change with speed in the sound slot settings, but I will explain the details in the future.

The data for the VVVF system in Open Sound Data is based on the data created by MB and Kawaii, so the template is almost identical. The slots (channels that produce sound) marked with red arrows in the following figure correspond to these.

What I am going to introduce here is not the sound of driving wind, but the sound editing process to produce the unique sound of acceleration and deceleration from LokSound. When recording, you will probably use a recorder (or even a smartphone) to record in the car, and the flow should be stop - acceleration - coasting - deceleration - stop. From this, cut out only the acceleration and deceleration parts.

We're going to cut the two types of acceleration and deceleration, and divide each one into 6 steps. 7 or 8 is fine, but for open sound data, we're going to divide it into 6 steps. 6 steps means that there are 6 speed steps. 127 in DCC. In DCC, there are 127 speed steps, but in order to link them to the sound, we deliberately divided them into six. 127 would have been fine, but it would have been very complicated, so we settled on a policy of dividing them into six based on what we know so far.

The actual sound program for the VVVF in LokProgrammer is shown below. As you can see, it's just six yellow blocks with six separate sounds for acceleration and six separate sounds for deceleration, fitted together to make a sound depending on the speed.

Figure 1 Acceleration

Now, let's cut out the running sound from the acceleration, coasting, and deceleration sounds of the Tokyu 50X0 series. Before cutting out the sound, I need to modify the sound in various ways, but I will assume that the modifications have already been done, and that the sound is ideal for acceleration and deceleration.

Figure 2 Deceleration

Figure out the range from departure to coasting by listening beforehand. Divide this range into six parts. There are various ways to divide the range, but the trick is to make the slow speed as short as possible. First, we will label the part immediately after the departure of the train.

Label the selection. You can adjust the position after labeling.

Audacity makes it very easy to export WAV files along with labels at any time, which I think is essential for DCC sound creation. This can be used not only for VVVF, but also for other applications. This is a must for DCC sound creation.

Repeat this process to create six blocks until the acceleration is complete.

We finished building the blocks for the acceleration section.

The trick is to cut between the blocks at the zero-crossing point (where the value is zero). This is because cutting at non-zero crossings may result in a mumbling sound. This little bit of editing will surely eliminate the mumbling sound. Also, if you can adjust it further, it is best to cut at the zero crossing of the VVVF sound divider.

This technique is also used to create a looped sound (the same sound played over and over). This is an essential technique, especially for diesel, so be sure to learn it. In the case of loops, it also adds the additional patience of finding similar waveforms and zero crossings before and after the loop.

Next, we will create a deceleration block. In the same way, select the beginning of the deceleration and label it.

I labeled the place where the deceleration begins. We will repeat this process.

The section to the stop was moderately adjusted and labeling was completed.

From the File menu, select Export Multiple Files to actually export a WAV file.

Specify the output destination folder. It is a good idea to create a folder just for running sounds.

The labeled ranges will be individually exported to a WAV file, as shown below.

The exported WAV file can be reflected in the data by overwriting the sound file list in LokProgrammer (drag and drop from Explorer, etc.). This file is the sound file that is associated with the acceleration and deceleration programs described above. If you replace the file, the sound will automatically switch to the replaced one.

In this article, I will explain the block diagram of VVVF coasting, acceleration and deceleration, which I have not explained enough. However, it is a simpler movement than diesel.

For the VVVF running sound, three sound slots are used, as shown below. In addition, there are various other sounds such as door opening/closing sound, brake loosening/slackening sound, etc., but they are set by mapping according to the timing, and only need to be played at the same time, so they must be made separately from the running sound.

First, the coasting sound. It is roughly divided into three blocks: function off, stop, and running. The running part is the biggest, but all it does is make the running wind and bogie growl higher or lower, louder or smaller, depending on the speed.

It is shown below in blocks. The most important part is that the acceleration and deceleration sections are separated. In other words, when you are accelerating (speeding up), the upper block is used, and when you are decelerating (speeding down), the lower block is used.

Assign the acceleration sound slot to the mapping for the objects in the red frame of acceleration as shown below. By doing so, the acceleration sound will always be played during acceleration.

The sound slots for acceleration are the following blocks. We will apply the divided sound of acceleration to each slot. The sound slots for deceleration are separate. Only when you are in the acceleration range will the acceleration sound be played at the same time as the coasting sound, depending on your speed.

For deceleration, assign a sound slot for deceleration in the Mapping of the block in the red frame of deceleration, as shown below.

The contents of the sound slot for the deceleration to be assigned are as follows

The squeaking and puffing sounds when the car stops are implemented in the D-S block as shown below. The timing of the sound is as shown in the arrow req=0 & spd <= 5, so that the sound is made when the speed becomes less than 5/255. This number will be decided on a case-by-case basis, as it may depend on the length of the sound and the habits of the vehicle.

This is how you can make the VVVF sound. I think that trains in general can be made in the above way.

## 4.9. How to make a diesel sound

Let's discuss the diesel sound. First, let's look at the movement of the diesel running sound used in Kiha 110 and Kiha 261.

Kiha110https://desktopstation.net/sounds/osd25.html   
  
 Kiha261https://desktopstation.net/sounds/osd23.html

In the case of VVVF, it is so simple that it doesn't need to be explained, but in the case of diesel, when accelerating, the speed of the engine differs from the actual speed. This is because there is a gearbox, and the sound changes according to the number of gear steps. It is a little complicated, so I will explain it in order.

First of all, the whole block diagram looks like this. The flow is to move from one block to another depending on the speed. It is divided into four major chunks: function off, when stopped, when idle, and acceleration.

The driving sound is achieved by moving within this block diagram as follows: function off, stop, accelerate, idle, accelerate, ・・・・, idle, stop, function off.

The orange arrows show the flow of moving more and more blocks as the speed increases. The speed is in the range of 255 for maximum speed and 0 for stopping. The maximum speed is 255, and the maximum speed is 0. If you are a programmer, you will understand this easily, but if you are not familiar with it, please note that the maximum value is not 100.

For example, if the speed changes from 10 to 30, this block will sound until this point. The speed is set to increase gradually, so the block will not go to the right immediately. It will take about a minute, which is determined by the acceleration/deceleration time of the Driving Characteristics (which can be set in the Decoder tab). This is determined by the acceleration/deceleration time in Driving Characteristics (which can be set in the Decoder tab), which is determined by the playback time of the sound. Initially, you should be fine as long as you get it about right.

When the acceleration is over, the sound moves to the "idle" block at the bottom. At this time, the sound passes through a block called DCx. It would be unnatural to switch the sound to idle immediately after the acceleration is over, so I crossfade the sound or make it cut off at the right moment so that the sound idles at the right time to blow up. CDx is the opposite of idle, it is used to create and place the sound when the sound shifts from idle to acceleration.

Since it may be difficult to understand, I tried to map the running sound data of Kiha261 to the block diagram. I chopped up the sound wave form of the enclosed area, and assigned the sound to the blocks. It is not possible to make all the sound data to be set in these blocks completely from this raw waveform, but this is the image.

Let's take a look at how each block is set up.

This section describes the properties inside the block.

Restore  
 When checked, this function allows you to interrupt playback of a high priority sound, and then resume the interrupted sound when you finish playback. A similar feature is available in the Unlimited checkbox in the sound slot. This one works on a per-slot basis, not on this block.

Sample  
 Select the sound to be played when entering this block from the sound file list.

When "Repeat Playback   
loop" is checked, the sound set in "Sample" will continue to be played. When the condition to move to the next block is satisfied (describe the condition in the arrow), the loop will automatically stop and move to the next block. If you don't check "loop", you can specify the number of loops you want to play by putting numbers in Min and Max. if you change the numbers in min and max, it will randomly decide the number of loops between them and play them.

Checking the   
FlagsDrivestop checkbox will prevent the vehicle from moving while it is in this block.

Mapping  
 You can set up sound slots, etc. to be associated with this block. The sound slots that are associated with this block will move together with this slot at the same time. If you assign too many slots, it will exceed the number of slots that can be played at the same time and cause strange behavior.

I will explain exactly how to change the sound.

First, I'm at a place called "S" (i.e., F1 is turned on and the car is stopped. The sound is a rattle and idle sound, but it is not running. (= F1 is turned on and the car is stopped. Look at the red arrows. The arrow going out from the first S says "2 :[share1 ! = 200 & S1 = false & req > 0]" should be written.

The first number is the priority number, and the smaller the number, the higher the priority. When you have multiple arrows coming out of a single block, you can use this number to decide which arrow has priority. After that, share1 ! = 200 & S1 = false & req > 0, but it's complicated, so just look at the last req>0.

req" means "request for speed". In other words, req> 0 should be understood to mean "request for velocity is greater than 0". When this condition is satisfied, the block is moved along the arrow. What I mean by "speed request" is when you use your throttle to change the speed of the vehicle higher than 0 to specify a speed.

Note that the word spd (speed) will appear later, but this is the actual speed of the vehicle (motor). The speed request is the speed specified by the person playing. Since there is acceleration and deceleration, they may not always match. Think of it as always being different during acceleration and deceleration. It is easy to confuse the two, so be careful.

Back to the story, after that, we enter a block called SD. This block is actually a "container" block, with blocks inside. You can think of it as a box that holds the blocks together. Inside, there are just blocks that are arranged to produce the brake loosening/slackening sound. They just make a pushy or whooshing sound.

Contents of the SD block (container)

After that, the arrow 1:[true] leads to a block called Throttle Up A\_02. This means "under any condition". In other words, there is no specific condition, but you can move it.

In the SD block, when the brake loosening/slackening sound finishes playing, it will unconditionally move to Throttle Up A\_02.

Pay attention to the throttle up A\_02. There should be two arrows. The red arrow points to the block of D1, and the blue arrow points downward. The arrow toward D1 is 2:[true], while the arrow toward the bottom is 1:[acc<0].

What I mean is that **acc<0 means "if the acceleration is negative"**. In other words, if the train is decelerating. I guess there was a passenger who tried to jump on the train right after it started, so the train stopped suddenly. It would be unnatural if it moved to the D1 block, which accelerates the sound, even though it started decelerating, so I made a blue arrow that goes to the idle.

The arrow conditions are described on the left side below.

You can edit the contents by double-clicking on the condition.

For now, I was able to explain up to D1 block.

For your reference, I have placed the sound data assigned to S to D1 below.

Data cooperation: Mr. YOMIX, Mr. yusa  
 [kiha261\_A.zip](http://buin2gou.sakura.ne.jp/sblo_files/powerele/image/kiha261_A.zip)

The block assignments are as follows

| **Block Name** | **layout file** |
| --- | --- |
| S | Idle.wav |
| Throttle Up A01 | A01.wav |
| Throttle Up A02 | A02.wav |
| D1 | Aloop.wav |
| DC1 | AX.wav |
| idol | Idle.wav |

## 4.10. How to use SpectraLayers

Using SpectraLayersPro, a software program that can analyze frequencies and edit sounds, I was able to crisply process the recorded sounds for DCC sound, and I would like to show you how to use the function.

It is a paid software, but it can be purchased for around 5,000 yen every few months at online sales.

SpectraLayers   
https://www.sourcenext.com/product/vegas/spectralayers/

First, the following screen will be displayed immediately after startup.

Here, you can open a recorded MP3 file or WAV file (real sound data recorded with a recorder) from D&D or the menu, and prepare the one that contains the sound of the coupling part from the E5 Series running sound.

E5 series recording sound:   
http://buin2gou.sakura.ne.jp/sblo\_files/powerele/image/E5\_test1.wav

This is what it looks like when opened in Audacity.

When you open it, you will see something fuzzy. This is the sound data displayed by frequency. If you remember your elementary school, junior high school, or high school science class, you'll know why sound can be classified by frequency.

Now, play back the sound. You will see that there are some noises.

If you look at it, you will see some suspicious vertical streaks. This software allows you to play back only a specific range. Use the range selection tool to surround the vertical streaks and press the play button. You should be able to hear the sound of the vertical streaks.

I'm starting to see that these vertical streaks are noise and that I should turn them off. I'm wondering how to erase them. There are several ways to do it, but I think the best way is to stamp it, so let's use the stamp tool. First, click on the PickSource button, and then select the original location of the stamp. You can choose the size and aspect ratio of the stamp in the Tools menu above (Size, Aspect Ratio).

First, I was able to erase it cleanly with a stamp. Other than the stamp, you can use the vertical selection tool and the magic selection tool. Once selected, you can press the Delete key to remove the area.

The magic selection tool is just right for getting rid of the noise that gets mixed in.

At any rate, if you make full use of this, you can create the data. The final cleaned object is shown below. All vertical streaks have been removed.

Finished sound data:   
http://buin2gou.sakura.ne. jp/sblo\_files/powerele/image/E5\_testAfter.wav

Lastly, a note on saving WAV files: if you save them in WAV format, please save them as Int 16bit; you can also choose Float 32bit, but LokProgrammer will not register it. If you make the mistake of exporting a WAV file in Float format, Audacity will open the file in Float 32bit without any problem, and you can save it in Int format in Audacity.

It is recommended to use Spectra Layers Pro in combination with Audacity.

# 5. Installation method

## 5.1 Precautions for installation

There are some important things to consider when installing a DCC decoder in a vehicle.

* Thoroughly insulate the wiring and decoder.
* Keep internal wiring to the minimum necessary. Keep wiring short.
* Always use a tester to check for shorts, and make sure there is no contact between the wiring coming from the wire and the function or speaker.
* Before installing a LokSound decoder, be sure to install an inexpensive decoder for a test run. Do not use an inexpensive decoder for a test run.
* Use a decoder tester to verify that the running and sound functions of the LokSound decoder are working properly.

We have a checklist for you. Be sure to check it at the time of loading.

| **Check** | **To be confirmed.** |
| --- | --- |
|  | Either the command station limits the current, or the AC adapter has a smaller current capacity. |
|  | Have you insulated the conductive part of the speaker? |
|  | Is there any contact between the speaker wires and the wires from the line, or are they close together? |
|  | Have you checked all of them in the tester? |
|  | Did you use an inexpensive decoder to check the operation when the car was stopped? |
|  | Did you use an inexpensive decoder to break in at a stoplight? |

Electricity has the tendency to flow towards the lower resistance. This is similar to the way water flows in the direction of lower resistance, such as dikes and dams. If there is nothing to stop it, it will flow a lot at once. Since water is viscous, the flow of water is not that large, but since there is no viscosity in electricity, if there is no resistance, it will flow as far as the limit of the power supply. Therefore, the flow of current generates heat (just like an oven) and leads to destruction.

Therefore, insulation measures are the most important to prevent shorts (a blunder of assuming zero resistance). This is the part that must not be skipped.

## 5.2 Use the auxiliary board for installation

If you have a decoder that can be replaced, such as Next18 or MTC21, be sure to check the operation first. If you use an inexpensive DCC decoder in the 1000 JPY range for the operation check, the damage will be minimal even in the worst case.

Many people have broken expensive LokSound decoders in the past, but now there are ways to test them without breaking them, such as Next18 and MTC21. There is no way not to take advantage of this.

We are distributing an auxiliary board called ExpBoard to make it easier to DCC Japanese vehicles. Please take advantage of it.

| **manufacturer** | **Product name** | **Connector** | **Scale** | **ArtNo** | **URL** |
| --- | --- | --- | --- | --- | --- |
| TRAINO | ExpBoard EC-Slim Easy | Next18 | N | ECS-E1 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ecslim-ver2amanual.pdf) |
| TRAINO | ExpBoard EC-Slim Standard | Next18 | N | ECS-S2 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ecslim-ver2amanual.pdf) |
| TRAINO | ExpBoard EC-Slim Economy | Next18 | N | ECS-C6 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ecslim-ver2amanual.pdf) |
| TRAINO | ExpBoard EC-Slim Type T | Next18 | N | ECT-S2 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ectmanual.pdf) |
| TRAINO | ExpBoard EC-Slim Type F | Next18 | N | ECF-S2,ECF-E4 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ecfmanual.pdf) |
| TRAINO | ExpBoard EC-Slim Type H | Next18 | N | ELH-S1,ELH-E1 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/ELH-Manual.pdf) |
| TRAINO | ExpBoard EC-Slim SL Type A | Next18 | N | SL-C57/D51 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/expforsl-manual.pdf) |
| TRAINO | ExpBoard EC-Slim SL Type B | Next18 | N | SL-C59/C62 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/expforsl-manual.pdf) |
| TRAINO | ExpBoard EC-Slim SL Type C | Next18 | N | SL-C58 | [URL](https://desktopstation.net/wiki/lib/exe/fetch.php/expforsl-manual.pdf) |
| DesktopStation | ExpBoard for KATO HO | Next18 | HO | 10024 | [URL](https://desktopstation.net/wiki/doku.php/expboardnext18) |
| DesktopStation | ExpBoard for General HO | Next18 | HO | 10023 | [URL](https://desktopstation.net/wiki/doku.php/expboardgeneral) |
| DesktopStation | ExpBoard for YP | MTC21 | HO | 10026 | [URL](https://desktopstation.net/wiki/doku.php/expboardendo) |
| DesktopStation | ExpBoard for YP SHORT | MTC21 | HO | 10035 | [URL](https://desktopstation.net/wiki/doku.php/expboardendo) |
| DesktopStation | EF81 Light Board | -Mr. | HO | 10031 | [URL](https://desktopstation.net/wiki/doku.php/ef81lightpcb) |
| DesktopStation | ExpBoard EH200 | MTC21 | HO | 10030 | [URL](https://desktopstation.net/shop/products/detail/55) |
| Fujigaya2 | ExpBoard for KATO HO DE10 | Next18 | HO | 10033 | [URL](https://desktopstation.net/shop/products/detail/65) |

## 5.3 DCC Connector

When a DCC decoder is installed in a vehicle, a connector can be used to facilitate decoder replacement. In addition, highly functional connectors with a pin count of around 20 pins are now becoming the norm, especially in Europe and the United States.

In the past, direct wiring to the decoder was the main method, which required rewiring in case of trouble and was very difficult to get used to.

In Japan, NEM652 is often installed as standard in KATO's HO motive power, but in some cases, DCC is not considered for bogie motor type (E5 series, EF81, EF510). Tomix, Pea, Katsumi, Dentetsu Kobo, U-TRAINS, etc. are analog-only and do not support DCC as standard.

OpenSound Data recommends MTC21 and Next18 as the standard for DCC connectors. The following is a list of typical DCC connectors.

| **Connector** | **Number of pins** | **scale** | **Notes** |
| --- | --- | --- | --- |
| NEM651 | 6 | N | His successor is Next18. |
| NEM652 | 8 | HO | Successors are PluX or MTC21 |
| NEM662 Next18 | 18 | N | Increasingly popular in N gauge |
| NEM660 MTC21 | 21 | HO | Developed by Mercklin, in a standards battle with PluX |
| NEM658 PluX22 | 21 | HO | PluX8 and 16 appear to be fading out. |

Note that there are two different types of MTC21 that use the exact same connector but have two different function outputs. One that has AUX3 and AUX4 as power outputs for Merklin support, and one that has AUX3 and AUX4 as logic outputs for DCC. When purchasing a decoder, please carefully check the output type of AUX3 and AUX4 before purchasing. Some decoders can be switched by changing the settings, while others are fixed due to hardware limitations.

KATO HO standard motive power and many HO vehicles designed more than 10 years ago use NEM652 (NMRA 8-pin). The pin layout is as follows.

The pin layout of the Next18 connector is shown below. The Next18 connector is manufactured by an overseas manufacturer, and is difficult to obtain in Japan.

|  | **pin** | **pin** |  |
| --- | --- | --- | --- |
| RAIL\_A | 9 | 10 | RAIL\_A |
| Head | 8 | 11 | Motor- |
| Speaker+. | 7 | 12 | AUX2 |
| COM+ | 6 | 13 | AUX4 |
| GND | 5 | 14 | GND |
| AUX3 | 4 | 15 | COM+ |
| AUX1 | 3 | 16 | Speaker- |
| Motor | 2 | 17 | Tail |
| RAIL\_B | 1 | 18 | RAIL\_B |

The specifications of the MTC21 connector are as follows: Since it uses a 1.27mm pitch pin header and pin frame, the availability of connector parts is very good in Japan.

| **Assignment** | **pin** | **pin** | **Assignment** |
| --- | --- | --- | --- |
| Sensor 1 | 1 | 22 | Track Left |
| Sensor 2 | 2 | 21 | Track, right. |
| AUX6(L) | 3 | 20 | GND |
| AUX4(L) | 4 | 19 | Motor |
| SUSI CLK | 5 | 18 | Motor |
| SUSI DAT | 6 | 17 | AUX5(L) |
| taillight | 7 | 16 | COM+ |
| headlight | 8 | 15 | AUX1 |
| Speaker 2 | 9 | 14 | AUX2 |
| Speaker 1 | 10 | 13 | AUX3(L) |
| No pin | x-mark (used to indicate an incorrect answer in a test, etc.) | 12 | VCC +5V |

## 5.4.AUX,HEAD,TAIL wiring

When wiring lights with DCC, there are two types of modes: bi-polar mode and open collector mode. Bi-polar mode is commonly used in analog, so you may be familiar with it. On the other hand, the open collector mode is familiar to electricians, but not so familiar to modelers. In other words, you can think of it as a mode in which the internal switch also turns on when it is turned on, allowing current to flow.

### 5.4.1. HEAD, TAIL in bipolar mode

You will connect the wires as shown below. This is the exact same wiring configuration as a normal analog light unit. In the light unit, the taillight LED and the headlight LED are connected to each other just in reverse (called reverse parallel).

If the light unit does not support the DCC method (COM+ and the three wires of HEAD/TAIL) and it is difficult to modify it, you can use it without modifying the light unit by wiring it using an ExpBoard or bi-polar conversion board that supports the bi-polar mode.

### 5.4.2. AUX, HEAD, TAIL (open collector) of DCC standard wiring method

The wire named COM+ has a positive voltage on it.

On the other hand, when AUX1, AUX2, HEAD, TAIL are turned on by a function, the internal switch connects them to the negative side. (In the electrician's world, this movement is called "sink"; it is the same as the drainage flowing into a kitchen sink. (In the electrician's world, this movement is called "sink." It is the same as the drainage flowing into the kitchen sink.) When the light is off, the internal switch is off, so it is not connected to the minus side, so no current flows and the light does not turn on.

The resistor can be positioned either before or after the LED, but the polarity of the LED should be chosen so that the longer LED wire (A, anode) is on the COM+ side.

## 5.5. KATO HO (single train cars KUMOHA40, KIHA110, etc.)

KATO's HO Kiha110 200 series (1-615) has NMRA connectors, but is not fully DCC ready.

However, I know that I can make the board fully DCC-ready by cutting some of the patterns and modifying it, so I modified the board to DCC sound, even though not a single word in the manual says that it is DCC-ready.

The bottom of the floorboard looks like the following. There is an iron plate which is used as a weight, but this will be replaced by ExpBoard Next18 for KATO HO.

The board before modification is shown below.

There are patterns that tell you to cut and jumper, so I did as the board told me to cut and jumper the patterns.

In a nutshell, the following is a list of the parts.

Important! C, T, and H are written on the LED lighting board, but if you wire it as it is, things will go wrong. If you wire them exactly as they are written on the LED lighting board, things will go wrong.

FWD side (at the end of →) C: same wiring is OK T: connect H(Rear,NMRA Pin2) H: connect T(Head,NMRA Pin6)

C: Same wiring is OK T: Connect T(Head,NMRA Pin6) H: Connect H(Rear,NMRA Pin2)

The opening in the floorboard is only big enough for the Next18 connector. Therefore, the decoder cannot be inserted in this way.

Therefore, we will hollow out the opening to allow for the decoder to be mounted.

## 5.6. KATO HO (power cars, Kiha80, etc.)

We will be installing the LokSound5 micro on KATO's HO Kiha80 M cars.

* Soldering tools, including soldering irons
* Pin vise,Tweezers
* Speaker.
* LokSound5 micro
* ExpBoard Next18 for KATO   
  HOhttps://desktopstation.net/wiki/doku.php/expboardnext18
* KATO HO Kiha80(M) 1-611https://www.katomodels.com/product/ho/kiha82

We will work on the loading process.

Solder the phosphor bronze copper plate. If it is bent too much, it will not hit the metal weight underneath firmly and cause poor contact.

Make sure that it is in contact with the metal weight underneath, like a plate spring.

Solder and install.

Cut out the red line. The seat part is also cut with a cutter. If you don't cut this part, you can't install the Next18 decoder.

Make sure the decoder is in place.

No problem.

Use a pin vise to make a hole for the speaker to pull the sound down.

The speaker can be hidden in the toilet or other part of the room, so you can use a pin vice to drill a hole here, wire the speaker, and solder it. How to hide the speaker depends on the size of the speaker, so I'm sure everyone will have their own ideas.

All that's left to do is cover the vehicle, check it works, and you're done!

## 5.7. Tomix HO

As usual, Tomix's HO cars are not designed to be DCC'd at all, so you will have to disassemble them and pull the wires around. I hope you will be prepared for the fact that the hurdles are very different from those of KATO's HO cars, and that you will be able to DCC them.

In this section, we will use Kiha261 (HO-9047, HO-9097) as the subject of our DCC sounding work. The first thing I did was to rewire the motor. First, remove the upper part of the train and take out the floor board and motor.

Remove the seat and you will see the motor.

Remove the metal weight.

Remove the copper bar as well.

Use a soldering iron to remove the copper plate attached to the motor.

Solder a thin wire (AWG30 or AWG32 is recommended) that you have prepared separately and attach it.

Now that the wires have been pulled out, we will reassemble it.

Push the motor back into its original position.

Put the copper bar back.

Put the metal weight back.

When pulling out the electric wire, check the position to pull it out as it will interfere with the seat part. After confirming the position, make a hole with a pin vise, etc., and pull out the wire.

I pulled out the wire to the seat. There is also a way to pull the wires around the floor without pulling them out. You can wire the wires according to your own preference.

Here we have ExpBoard Next18 for General HO (https://desktopstation.net/wiki/doku.php/expboardgeneral) and PUI Audio's ASE02506MS-LW90-DSM-R speakers. This is a diesel car. Since this is a diesel car, the trick is to choose speakers that produce a clear diesel engine sound, and the ASE02506MS are thin and sound pretty good.

LokSound5 micro installed.

The decoder and ExpBoard should be well hidden.

The DCC conversion process is complete!

## 5.8. Tramway HO

I'm going to try my hand at DCC sounding a Tramway Kiha 40-500. Here's a list of the steps from the box to the removal.

Research to convert it to DCC, and take it apart. Unfortunately, the 8-pin and other connectors of NEM652 were not attached.

This is the first time I've bought a Tramway car, but it has a steel plate for weight on the outer chassis. Thanks to this, the under-floor parts are only made of plastic parts, so it is easy to build inside the car. Instead, the rigidity is sacrificed...

The connection to the upper lighting board is by pogo pins. This is one way to use it...

Light board (bottom). The wires from the wire, motor, and the other side board are wired. Eventually, only the common, F0 output, and interior light control signals will be wired.

I found a [schematic of the lighting board on Tramway,](https://www.mmjp.or.jp/tramway/goods/pg350.html) so I rewrote it in BSCh and assigned it to DCC. I haven't tested it yet. I haven't tested it yet, so it may be wrong! Please be careful.

Note: Operation has been verified by modifying the circuit as per this schematic, but operation is not guaranteed.

This is illustrated on the board. The red line is the pattern cut. Use a cutter or similar tool to cut and remove the copper pattern on the board. This is to make the NMRA 8-pin open collector output compatible from the bipolar type.

Note that the color and pin number of the NMRA pins are shown in the diagram, but it does not mean that you should wire them as they are, because the meaning changes with FWD/REV in the light system!

Here, instead of adding two light decoders such as the common FL12 or the one-coin decoder FL, we have dealt with the problem by cutting patterns and adding wiring to the board so that the ESU LokSound pins can be used as they are. This requires complicated wiring work in the car and is for advanced users, so if you think it will be difficult, it is better to use the FL decoder.

Now let's get to work. First, use a soldering iron to remove the wiring that is directly connected to the wire. With the tail light wiring, we are adding wires to the LEDs.

I added wiring to the board. This is for the headlight, tail light and interior light. The circuit board on the chassis side does not need to be modified at all.

After soldering the wires, fix the wires in place. I used the original adhesive tape.

The cover is now installed. Some holes were drilled to pull out the wires. The wires were connected to the pads of the ExpBoard Next18 for General HO.

Since we don't want it to move due to vibration, we fixed it with acetate adhesive tape (insulating and easy to install!). The speakers were also painted black. I also painted the speakers black.

By using Next18, the pile of wires will not stand out greatly when the car body is covered.

Next, I considered installing an interior light. On the ceiling side of the Kiha 40, there is a terminal for the interior light. When I measured it with a tester, the circuit diagram said 4.7kΩ, but the actual value was 2.2kΩ. I guess I shouldn't trust the constant in that schematic.... The wiring seems to be correct. The other thing that bothered me is that it doesn't say anything about which side is the positive side. I wonder if everyone uses a tester to check? It would have been better if it was written in silk on the back of the board. I marked it with masking tape as you can see in the picture. Please refer to it.

The current limiting resistor is 150 ohm with 3 series LEDs. The current limiting resistor is 150 Ω. I attached electrical wires and tab terminals.

In the meantime, I'll do some calculations and see if it shines a little.

Since there are 3 LEDs in series, Vf=2.53=7.5V (assuming Vf of white LEDs is about 2.5V) Resistance is 2.2kΩ+150Ω=2.35kΩ When voltage is 12V, V=IR→V-Vf=IR, so 12-7.5=I2.35k, I=(12-7.5)/2.35 = 1.9[mA]

I could tell from my calculations that it was vague and barely glowing.

After installing the LED tape, it looks like the following.

I tried to light it up. Since it is a DCC model, the interior lights can be turned on and off separately from the headlights and taillights, so it is very convenient to use. It seems to avoid the dumb operation where the lights keep glowing even though the train is being sent to a garage.

## 5.9. ENDO HO

We will use ExpBoard YP to create DCC sound for the Kintetsu 22000 series pre-painted kit.

ExpBoard YP https://desktopstation.net/wiki/doku.php/expboardendo

First, I soldered the ExpBoard YP. It is very difficult because of all the small parts, so it is assumed that you have advanced electronics skills. The MTC21 connector is a 1.27mm pin header, but you need to remove the index pins (black ■) in advance. This is to prevent accidental insertion.

IC1 and IC2 have the same shape, but different components, so be careful. IC1 is marked **A1, and** IC2 is marked **G3.** The Schottky diode and the constant current diode are attached to D1 and D2, but even if you make a mistake and put them in the opposite positions, they will work as usual. However, make sure that the anode and cathode are in the correct positions.

After soldering, I also attached the speakers.

Preparation work for the actual integration.

Mounted the soldered board on the ceiling and checked its position.

Thanks to the ExpBoard YP, the headlights, taillights, interior lights, sound, etc. are very easy to install, but the underfloor area is not considered at all, so I spent a surprising amount of time installing the connectors.

The ExpBoard YP has a bi-polar board, which means it can use bi-polar outputs instead of the open collector outputs normally used in DCC. The ExpBoard YP is a bi-polar board, so it can use bi-polar output instead of open collector output, which is usually used in DCC.

This is how it worked for the time being.

## 5.10. Dentetsu Kobo Plastic HO

Dentetsu Kobo is a 1/80 HO brass model manufacturer located in Kanda Jimbocho, Tokyo. It has a reputation for precision and high reproduction, and is popular among model enthusiasts.

Dentetsu Kobo  
  
 Kanda Yoh Building 1F1, 1-52-12 Kanda Jimbocho, Chiyoda-ku, Tokyo, Japan   
https://dentetsukobo.jimdofree.com/

Here, I will be doing a DCC soundification of the 115 series that I released a while back.

There is no specific stock information on the Internet (Dente Kobo doesn't post stock information on the Web, so the only way to check stock is to go there in person or call them...), but they say they have it, so if you're interested, why don't you contact them or visit them on a Friday or Saturday business day?

For your information, the prices are listed below for the 4-car basic set.

* Shonan color: 46,800 yen (tax not included)
* Yokosuka (Suka) color $47,250 (excluding tax)

I couldn't make a long formation (no place to put them, too many of them would make me angry, etc.), so I chose the Ska color because it looks better in a 3-car formation than the Shonan color.

In the case of the Shonan color, the M car becomes the middle car. In the Ska color, the M car is the lead car, so the Ska color requires more work. This method can be used for both the lead car and the M car, so I think it will be more helpful for you.

So, I'm going to DCC Dente Kobo's 115 series! Of course, I will use open sound data.

Preparation of components

As you may have noticed if you've played around with the Open Sound Data, the choice of speakers actually depends on the sound. Some sounds are better suited to a sugar cube, while others are so bass-heavy that you need a larger enclosure to get the most out of them. The perfect speaker for the MT54 is the ASE02808MR-LW150-R, although it is big.

For the DCC conversion, I selected ExpBoard General HO as the Next18 as it is a plastic HO. Also, I didn't want to mess with the light board, so I used the bipolar board (AYA014-2) from DCC-Kan. I don't want to convert the Japanese model to DCC without this anymore (lol).

| **Parts** | **Quantity** |
| --- | --- |
| Dentetsu Studio Series 115-800 Suka Color Basic 4 Cars (1/80) | 1 |
| ESU LokSound5 micro DCC (Next18 version) | 1 |
| ExpBoard Next18 for General HO | 3 |
| PUI Audio ASE02808MR-LW150-R | 1 |
| DCC Hall AYA014-2 Bi-polar PCB | 2 |
| LaisDcc Next18 Decoder | 3 |

Conversion of vehicles to DCC (M cars)

It's simpler than the Tomix 113 series I bought before, and the wiring drawer is very easy.

Pull the wires from the copper plate. Use the pre-installed wiring for the motor.

I'll put it back together.

Drill holes in the floorboard cover beforehand. Also, insulate the terminals of the interior light output with tape. This is for the AUX1 wiring later.

Be creative in choosing where to drill holes so that the cables can be pulled out.

Choose the position of the speaker. I chose the middle car side. It is better to cut the backrest. It is also a good idea to drill a hole in the floorboard cover so that the sound can escape.

Wire to ExpBoard General HO. It won't take long. Be sure to check for shorts with a tester when you're done!!!

Preparation for the interior light. Since the terminals for the pogo pins of the interior light have already been insulated, solder them as shown below, and wire COM+ and AUX1.

In wiring the headlight and taillight, we will use a bipolar board for easy mounting. We will also install them in a less intrusive way this time. First, remove the copper plate that is wired to the switch. Accident prevention.

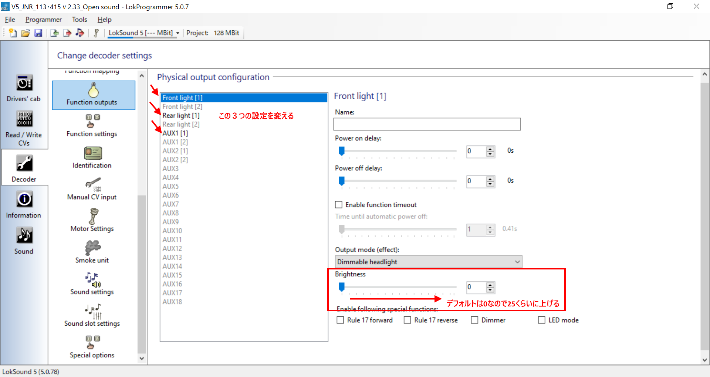
The other side is wired with insulating tape and a copper plate on top of it. This way you can hijack the contacts of the headlight/tail light springs and use them with DCC.

Fix the bipolar board and you are done.

The top cover, or casing, or case, was installed without any problems.

Writing sounds

I had no choice but to use the MT54 JNR suburban model, which is talked about as having the best sound quality in the world! I heard that 113 series sound is closer than 115, so I chose 113 series sound data (V5\_JNR\_113・415v2.33\_Open sound). Note that the output of F0 headlight/tail light and AUX1 interior light are set to 0 by default, so it is better to change the output value to 25 before writing.



I tried to move it.

I moved it around and... quite an understatement, it's bad! It looks (and sounds) like the real thing. I can't even imagine not having the sound anymore, it's so bad. People who are experiencing DCC sound for the first time will probably not be able to stop laughing. It's so bad, you'll start laughing.

Video on Youtube:   
https://www.youtube.com/watch? v=hJqH0iEmcyQ&feature=emb\_title

Next, we will move on to the processing of the T-car of the Dentetsu Studio 115 Series Suka Color. This is the DCC processing of the lead car without a motor.

Open it first.

Remove the floor plate. A precision flathead screwdriver is helpful.

The first LED is split by this switch. Since it is controlled by the decoder, I removed the copper plate with a soldering iron to avoid wiring to the switch.

This is the insulating process for the copper plate for the interior light drawer of the floorboard plastic, and the hole for the wiring drawer. It is easy to open with a pin vise.

Forcibly soldered to the metal weight, since there was no part of the copper plate that was clean...

The floorboard plastic is reinstalled, and the ExpBoard General HO is installed to mount the Next18 decoder.

All that's left to do is to wire the interior lights (COM+ and AUX1 are soldered together. ) and wired the headlight and tail light using the DCC-Kan bipolar board. The headlight and taillight are wired using the DCC-Kan bipolar board. Very easy. I insulated it with acetate tape and placed a copper plate so that I could apply a spring and use it for wiring from the DCC decoder.

Check operation and complete.

Decoder settings

For the decoder, I used laisdcc's 860015 Next18 DCC Decoder (PanGu series). By default, AUX1 (interior light) is assigned to F1, so I just changed this to F3. In the "h) Function Remapping" section of the LaisDcc decoder manual, CV35 is AUX1, which defaults to 4 (F1), so just change this to 16.

## 5.11. N gauge

### 5.11.1. kato 113 series

At last, the KATO 113 series Tokaido Line colors have been renewed! Of course, I wanted to recreate that nostalgic MT54 sound, so I decided to install the usual LokSound decoder.

The first step is to convert the M car, Moha 113, to DCC.

The ExpBoard EC-Slim is equivalent to a standard set, with additional interior lighting.

ExpBoard EC-Slimhttps://desktopstation.net/wiki/doku.php/expboardecn

Remove the body and under-floor cover of the Moha113, and consider how to install it. The sound decoder will be placed on the left side of the screen where the interior light unit will be installed.

I also decided to install the speaker on the right side. I will use a Takachi case (SW-15B) and a speaker purchased from ali.

The case of the takachi is shaved thinly in half. This is what it looks like before assembly.

The unit and case are fixed with clear rubber adhesive. Also, the metal terminals on both sides of the unit are turned upward and placed against the body, so that they can be pressed against the floor.

As usual, plug the EC-Slim into the EM13 mounting. Since the room is blue, we will use a blue cable. If the cable is the same color, you may want to put a sticker with the name of the signal on it, since there is a possibility of making a mistake.

Pull the cable out from the gap between the under-floor parts and make sure that the under-floor parts are firmly attached. (Actually, this time, I started with the usual gray cable, but changed it to a blue cable in the middle of production.

It is a good idea to do a driving test once in this form.

Place the EC-Slim and speakers on top of the power unit and position them to determine the length of the wiring.

Cut the wires, peel off the tip coating, and solder them.

Put the board on and wire TLK1,2 MT1,2. Make sure that the wiring for the rail side and the motor side is correct. Also wire the speakers.

After a test run to make sure everything is okay, we put the body over to test the sound. This time, it's a little bigger than the usual plastic enclosure, so it might be a little louder.

### 5.11.2. KATO C59 DCC sound processing

Introduction of DCC sound processing for C59 (KATO product).

The ExpBoard is similar to the one used for the C57, but the terminal locations are different.

The one for the C57 was installed on the bottom side of the decoder, but this one is designed to be installed on the top side of the decoder and speaker, and has wiring pads on both sides.

The C59's tender is long, but it has a radius on this part, so I used the luthier to carve it in a little.

The speaker units are the same as always, distributed by Nagoden.

Nagoden http://www007.upp.so-net.ne.jp/nagoden/

The terminals should be cut short so that they do not touch the weight where the current from the rail flows, and solder-plated for easy wiring.

The enclosure is a combination of the plastic board and plastic rod that I introduced earlier. The corners are trimmed and the grooves for the speaker wires are sanded.

Temporarily place the speaker and enclosure inside the die-cast parts. If the die cast expands, re-machine the part that is hit.

Make sure that each part will fit inside the weight. In particular, if the PCB juts out above the weight, slightly shave the area where the speaker hits.

Take out the wires for the motor and speakers from the ExpBoard. This time, I took out all the wires from the bottom side. The power is supplied from the rails by pressing the 0.2mm white wire against the left and right weights.

In addition, the headlight wiring is not used. I replaced it with this decoder for headlights (PetitDecoder-SL), which is under development (I will explain in detail separately). the C57 seemed to be difficult to remove, but in the case of the C59, I was able to replace the board in this condition.

Assembling the parts I made. The printed circuit board and speaker are held together with a small amount of rubber adhesive. The LokSound board is also die-cast and insulated with tape to prevent shorts.

It fits perfectly like this. Pull it out, being careful not to get the wires to the motor mixed up.

Cover it with the tender top parts. At this time, make sure that the subfloor and other parts are not pushed inside and raised.

Assemble the tender by running the speaker wires through it.

Disassembly and modification of the engine side is the same as for the C57. Remove the metal plate from the die-cast and connect the motor wiring to the wiring from the tender. Be sure to insulate the wiring with Kapton tape to prevent it from coming into contact with the body (die cast).

As you can see, no wiring is visible from the outside.

After this, the second car is completed by rewriting it to the open sound data being developed.

# 6. frequently used settings

## 6.1 Setting frequently used CVs

The following is a list of commonly used CVs in LokSound 5 with open sound data. LokProgrammer allows you to change the settings without being aware of the CVs, which may be different for LokSound V4.

| **CV Number** | **CV Name** | **Setup Method** | **Remarks** |
| --- | --- | --- | --- |
| CV1 | short address | European Style 1-100  American style 1-127 | Valid when bit5 of CV29 is 0. |
| CV17 | Long Address #1 | 192-230. Used in combination with CV18. | Valid when bit5 of CV29 is set to 1. |
| CV18 | Long Address #2 | 128-255. used in combination with CV17. | Valid when bit5 of CV29 is set to 1. |
| CV2 | Start voltage | Voltage at the start of running. Default is 1. | Note that 255=100% is a relative specification. |
| CV5 | Maximum voltage | 1-255. 255 gives 100% voltage at maximum speed. At 127, the voltage is half 50%. | Note that 255=100% is a relative specification. |
| CV63 | volume (sound) | 0-129, max at 192, normal volume at 128. |  |
| CV29 | Basic Settings | The default is 30. | 28/128Step, Analog enabled, Short address, Speed table enabled |
| CV155-CV170 | SoundCV | Each data may have its own settings. |  |
| CV258-CV448 | Function Outputs | You can set the output settings for headlight/tail light and AUX. Please refer to the ESU manual for details. |  |

CV29, in particular, is the basis of the decoder settings.

For example, the decimal number 3 (which you use in your daily life to represent 0 to 10) is represented as 11 in binary. Bit 1 represents 2, and bit 0 represents 1. So, 1+2=3. Bit 2 is 4, bit 3 is 8... bit 7 is 128... Note that CV29 cannot calculate numbers without understanding this bit representation.

The value of CV29 specifies a setting in bit representation to allow various settings to be made with a single CV value, as shown below. This is a technique called bit flags, which is very common in the embedded programming world. Since CVs are not infinitely configurable, this method is used in some cases.

Note that bit string means binary representation and dec means decimal representation.

| **bit** | **dec** | **Function** | **Meaning of the setting** | **Remarks** |
| --- | --- | --- | --- | --- |
| 0 | 1 | direction of travel (esp. train) | 0: Forward direction (FWD),  1: Reverse direction (REV) | This is used when the motor wiring is wrong. |
| 1 | 2 | Speed step | 0: 14 steps,  1: 28/128 steps | Should be set to 1 unless there is a reason not to. |
| 2 | 4 | analog function | 0: OFF,  1: Analog operation permitted | If you want to operate with an analog DC power pack. |
| 3 | 8 | RailCom | 0: OFF, 1: ON | If you are using RailCom, please set it to 1. |
| 4 | 16 | Speed table | 0:Speed table OFF,  1:Speed table ON | Set to 1 for open sound data. |
| 5 | 32 | long address | 0: Use short address (2 digits),  1: Use long address (4 digits) | Please switch according to the address you are using. |
| 6 | 64 | unused | Set to zero. | -Mr. |
| 7 | 128 | unused | Set to zero. | -Mr. |

Let's take a look at some specific examples to show you how to set it up.

For example, if you selected the CV29 function as shown below, it would be B00011110 (binary representation).

| **bit** | **dec** | **Function** | **Selected settings** |
| --- | --- | --- | --- |
| 0 | 1 | direction of travel (esp. train) | 0: Direction of travel is positive (FWD) |
| 1 | 2 | Speed step | 1: 28/128 steps |
| 2 | 4 | analog function | 1: Analog operation permission |
| 3 | 8 | RailCom | 1: ON |
| 4 | 16 | Speed table | 1:ON |
| 5 | 32 | long address | 0: Use short address (2 digits) |
| 6 | 64 | unused | 0 |
| 7 | 128 | unused | 0 |

The value to be set for CV29 can be expressed by the following formula

127x0 + 64x0 + 32x0 + 16x1 + 8x1 + 4x1 + 2x1 +   
1x0=   
16+8+4+2= 30 (decimal)

For CV29, write 30 in the CV write.

But it's a hassle, isn't it? With DSair2, all you have to do is select the function you want in CV29, and CV Programmer will automatically calculate and write it for you.

DSair2 CV   
Programmerhttps://desktopstation.net/wiki/doku.php/dsair2\_cvprogrammer

6.2.1 Setting up RailCom

Be sure to enable (turn on) CV29 bit3 as described above. To use RailCom with vehicles equipped with open sound data, a Detector or Display (Lenz LRC120, DesktopStation RailComDisplay) is required.

If multiple Detectors or Displays are used, wire them in parallel.

# 7. column

## How to get started with Open Sound Data

Yaasan (DesktopStation)

After the standards wars of the early 1990s, digital model railroading has achieved convergence with the DCC standard proposed by Lenz. There are many advantages of digitizing model trains. Automatic operation, affinity with computer control, wiring saving, simultaneous control of multiple units, control by multiple people, and so on.... Of these, I think the most obvious is sound. Since the days of analog model railroading, the quest for sound has been tested, and I understand that with DCC sound, a universal platform has been created.

In Japan, Digitracs, Soundtracs, ESU, etc. have been mainly used since around 2000. ZIMO, Hornby, etc. were in the minority. Among them, Digitrax seems to have occupied a large share, probably due to KATO's domestic agency and the spread of command stations. Since the data can be rewritten, some model stores have released the Japanese model data, and it seems to have been used by many users.

On the other hand, Digitracs sound decoders, which had been reasonably popular in the Japanese DCC community, were becoming obsolete around 2020, when the low number of simultaneous sounds and poor sound quality became noticeable. In addition, we felt that there were significant issues in terms of support for DCC connectors such as Next18 and MTC21, which started to be widely used around 2017. In addition, there were some concerns about stability and quality.

On the other hand, I strongly felt that the Japanese DCC sound was a key solution. While we were wondering how to proceed, we happened to have an opportunity to meet people who were thinking in the same direction, and as a result of various discussions, we were able to find that we could clear our concerns and issues by focusing on ESU's LokSound, although the price would be higher than Digitrax's product. We feel that we have created the prototype for today's open sound data.

At the same time, there was a discussion on DCC connector support within the DCC Electronic Craft Coalition, and we decided to proceed with these movements as a set.

In 2019, we will launch Open Sound Data, and with the volunteer efforts of many collaborators, we will make the sound data available to the public and create an easy DCC environment for Japanese vehicles with the ExpBoard series.

As a result of these activities, information sharing and support goods have been enhanced, and the issue of installing DCC sound in Japanese vehicles is gradually being resolved, and the hurdle is getting lower every year. At the same time, the risk of modification failure is also being reduced while achieving higher function with auxiliary functions.

After the many failures of the first DCC boom around 2000-2010, 2020 may be a test to see if it will be the first year of Japan-style DCC.

## The Future of Open Sound Data

Yaasan (DesktopStation)

We will release more sound data to the public, and at the same time, we will create an environment where users can create more and more data as creators.

For example, the publication of documents such as this book, development seminars, and offline and online events such as Open Sound Data Meetings.

The next step, around 2030, will be the in-house production and open specification of sound decoders and writing devices. The next step is to open up the specifications of sound decoders and writing devices. Competing overseas DCC companies have completely enclosed sound decoder systems, and users have to recreate data from scratch if they change manufacturers.

We have been gradually accumulating ideas and technologies, and we know that if we can build the following items: ultra high-density components, ARM or RISC V-based high-performance microcontroller, DCDC converter, flash memory, Raicom-based decoder-to-writer sound data and firmware writing technology, bootloader, and sound data creation tool. We know that we can achieve this if we can build the following items. Some of these technologies have already been obtained by the DCC Electronic Engineering Coalition, and we are trying to expand the development gradually.

If an open sound decoder can be realized, the same sound data can be used even if the manufacturer changes. This will make it easier for people to switch from one manufacturer to another and have far more choices.

We would like to create an environment where all Japan can enjoy digital model railroading and DCC, and in the future, Japan will lead the future of model railroading overseas.

If Japan is lagging behind in DCC, we should break the bonds of the existing DCC manufacturers with the technology and ideas to reset them and lead the world.

For this purpose, the users of open sound data also need to cooperate greatly. We would be very grateful for your cooperation.

## Encountering LokSound V3.5

Contributed by Mr. MB3110A

Back in 2007, while I was researching to model the diesel sound of Hornby's HST, I found out that South West Digital in the UK had LokSound V3.5, which was programmed to sound like the HST, and I ordered it by mail.

Even now, South West Digital only accepts orders by e-mail, but I remember sending them an order e-mail using my poor English, and when I received a reply saying "In stock", I sent them my credit card number (for security purposes, I split the card number in two halves on their advice) and waited for the order to arrive. I have memories of waiting for the order to arrive.

It was only later that I realized that LokSound decoders allow you to rewrite sounds by yourself, making it easy to create your own original sound decoders. 2008, I also ordered a LokProgrammer to create my own sounds, and started to create my first Kiha40 sound. I was not good at computers. I wasn't confident that I could do it as I was not good with computers, but I decided to give it a try...

M is for engine stop, S is for idle, D is for coasting, A is for acceleration, CX is for deceleration, and other transition sounds can be configured in a template, and the transition conditions are simple and easy to understand by entering the throttle opening values. I still use it as the basis for all my core sound schedules.

The v3.5 specs only store 8Mbit (about 69 seconds of sound), so I had a hard time trying to squeeze in less than the capacity by using a lot of "LOOP playback" that repeats the same sound. Even so, it was satisfying enough compared to the 1M bit (about 11 seconds) when it was v2.

Later, with a minor change around 2008, it became capable of storing up to 16M bits (about 138 seconds of sound), giving you more room to edit sound data.

This is an example of a sound schedule for Kiha40 Sound (first generation).

The A stage is not used, the D stage is used for acceleration sound, and CX is used for idle sound (coasting). At the time, it was a perfect spec for a diesel car, but now that I think about it, the response was poor because there was no immediate function (to cut off the sound in the middle of playback and force the transition to the next sound), and the only way to overcome this was to make a short, beautiful LOOP sound. There was also no function to synchronize the sound of the current breaker with the running sound, so it may be a little lacking as a sound program.

In 2011, LokSound V4 underwent a full model change and the sound capacity was increased to 32M bits (about 267 seconds). The program is now completely different, and the sound is able to respond to complex movements and evolve dramatically. At first I had no idea how to program, so I used the conversion function of LokProgrammer to convert the LokSound V3.5 data to V4 for analysis.

This evolution from LokSound V3.5 to LokSound V4 was shocking, and it can be said that this sound decoder established the basis for what is now LokSound 5.

In particular, the Immediate function (which cuts off the sound in the middle of playback on the spot and forcibly shifts to the next sound) is attached to the transition of the sound schedule, and the transition response is greatly improved. Since then, we have produced many sound data.

# 8. support

If you have any questions or concerns about Open Sound Data or LokSound decoders, please use the Digital Model Railroad Forum. As Open Sound Data is not a profit-making business, support will not be provided in principle.

Open Sound Data, DesktopStation, DCC Electronic Arts Coalition, and its creators will not be held responsible for any damage or destruction of vehicles or equipment caused by the use of Open Sound Data. Please use Open Sound Data at your own risk.

Basically, we do not accept requests, modification requests, or change requests for open sound data. However, if you provide us with a sound source that you have recorded yourself and declared Creative Commons CC0, and if it matches the direction of the creator, we may create the sound data. There is no guarantee that sound data will be created even if you record it, but on the other hand, if you do not record the sound source yourself and provide it, you can be sure that no sound data will be provided.

The Digital Model Railroad Forum is managed and operated by DesktopStation, and there are no registration or annual fees.

Please note that even if you ask questions or seek advice in the Digital Model Railroad Forum, we cannot guarantee that the problem will be solved. The answers are provided by volunteers in good faith, and are not official answers from ESU. This is just an exchange of information between users.

Digital Model Railroad Forum https://desktopstation.net/bb/

If you want to get official support from the developer, there is a support forum run by ESU. Please note that Japanese is not supported and questions should be asked in German or English.

Open Sound Data is not affiliated with ESU in any way. You may not request support for ESU's products from the operator of Open Sound Data.

http://www.esu.eu/forum/forenuebersicht/

# 9. FAQ

This section contains answers to common problems and questions about the LokSound sound decoder, as a volunteer. Please note that we do not guarantee that these answers will solve your problems, and you will basically have to solve them yourself. DesktopStation and Open Sound Data are not affiliated with ESU, nor do we have any distributorship or support agreement in Japan. We do not have a distributorship or support agreement with ESU in Japan. We will not be able to provide any support for LokSound decoders.

To receive official support from ESU, please visit the ESU website.

## 9.1 What manufacturer's command station can I run?

As long as the command station complies with the NMRA DCC standard, it can be used with any DCC command station or DCC controller from any manufacturer in the world.

Please note that the old Merklin Command Station, CS2, CS3, etc. will not work in Merklin Digital mode or mfx mode. Both CS2 and CS3 support DCC mode, so please use them in DCC mode.

## 9.2 Will you release sound data for Europe and the United States?

Sound data for Europe and the United States is officially released free of charge by ESU.

projects.esu.eu http://projects.esu.eu/

This is outside the scope of open sound data. We do not accept any inquiries or support requests for Western sound data, for which we do not have any rights.

## 9.3 Can open sound data be written to a decoder other than ESU?

The open sound data has been created specifically for ESU's LokSound V4 or 5 series. It cannot be written to sound decoders from other manufacturers. There is also no policy to support them at all.

It is not possible to write to other companies' sound decoders such as ZIMO, Digitrax, SOUNDTRAXX, Uhlenbrock, etc. Each company that develops sound decoders may release their own sound libraries. Users of each sound decoder should refer to the website of each manufacturer.

|Manufacturer | URL | Sound region | |:-|:-| |ESU | http://projects.esu.eu/ | Western type | |ZIMO | http://www.zimo.at/web2010/sound/tableindex.htm | Western type | |Uhlenbrock | http://www.d-i-e-t-z.de/7\_6.htm | Western type | |Digitrax | https://www.digitrax.com/sound-depot/list/ | Western type |

## 9.4 Can I use LokProgrammer to rewrite the cantum, etc.?

LokProgrammer is a product of ESU. Qantum is a sound decoder system used by Tenshodo on an OEM basis from Broadway Limited. Due to the difference in manufacturers, it is not possible to change the sound or functions.

Similarly, open sound data can only be written to ESU's LokSound decoders. Similarly, open sound data can only be written to ESU's LokSound decoders, not to any other manufacturer's sound decoders.

## 9.5 How do I prevent the decoder from breaking?

Decoders that you have bought must be checked for operation with a decoder tester (ESU 53900 Decoder Tester or LaisDCC 860033 Decoder Tester Pro) before being installed in a vehicle. If the product is broken (or has been broken) when you buy it and check its operation with the decoder tester, there is a very high possibility that it is an initial defect. For the LokSound5 micro, which has a particularly complex manufacturing process, the data shows that the market defect rate after shipment from the manufacturer is about 1 or 2 out of 100 units. At this point, you should contact the manufacturer or distributor for an initial defect replacement.

It is also imperative that users do not install expensive LokSound decoders right away, but work on installing them in the vehicle using MTC21 or Next18 DCC connectors, and then check the operation with inexpensive LaisDcc decoders. The cause of many decoder failures in the past has been user error in the installation process. Complex installation work must be done carefully and with a lot of time and checking.

## 9.6. Jittery movement after writing open sound data

Even if the open sound data is correctly written to the LokSound decoder, there may be cases where the movement is jerky, especially at low speeds. The reason for this may be that the Basic settings, Slow Speed settings, and BEMF (motor speed detection function) settings in the Load Control section of the Motor Settings tab do not match the motor.

Basic settings should usually be fine as they are, but if they do not work, use the auto-tune function to adjust them. Instructions on how to use the auto-tuning function can be found in the ESU LokSound decoder user manual.

The Slow Speed settings are provided to adjust for low speed driving, but we recommend that you do not use them, as they do not work well in most cases. Therefore, the regulation parameter "K slow" and the Largest internal speed step that uses "K slow" should be set to 0.

For BEMF, most of the open sound data will have the adjusted value, but if the value has been changed, changing it to the adjusted value will often work. If the value has changed, change it to the adjusted value. It is quite difficult to set the frequency and accuracy of the BEMF speed detection, because the speed cannot be obtained correctly if the setting is too long or too short.

| **BEMF setting items** | **Before change (example)** | **After adjustment (example)** |
| --- | --- | --- |
| slow speed Back EMF sampling period | 2.5ms | 5ms |
| Full speed Back EMF sampling period | 2.5ms | 15ms |
| slow speed length of measurement gap | 0.3ms | 1.5ms |
| full speed length of measurement gap | 0.3ms | 2ms |

## 9.7 The decoder does not work, does not respond, or has broken down.

It is an essential requirement that you follow the precautions to avoid breaking the decoder in the previous question. If you fail to do so, the cause of the failure will be the user.

If the decoder stops working after being installed in a vehicle equipped with a decoder, there is a very high possibility that the failure is due to wiring error, insulation failure, poor fixation or contact/short circuit in the decoder installation process.

Check the following points.

* Cannot read the address (CV1), does not work with the set address
* Whether the IC on the surface of the decoder is burnt or not, whether the heat-shrinkable tubing is torn by heat or not

If the address can be read out, the decoder is functioning properly, but some functions may not be working properly due to malfunction. In some cases, even if there is no apparent problem, there may be a malfunction. In addition, although it is very rare, there are cases where the initial failure is found later due to poor soldering or poor contact of parts.

If the vehicle is equipped with a DCC connector, replace the decoder and check its operation. If the decoder works normally after replacement, it is highly likely that the decoder is faulty.

The most important measure is to use a valuable decoder in a way that does not break it. If it is damaged, it will result in a great loss. Be sure to follow the steps to avoid damaging the decoder, and proceed with the DCC of your model vehicle safely.

LokSound decoders are extremely precise. Not only LokSound, but also other decoders can fail if they are not properly installed and used. If you cause a failure due to your own fault, it is not possible to replace it. In particular, novice users who are not familiar with the installation process, and who neglect to check the operation of the decoder, may install an expensive LokSound decoder from the beginning and cause it to fail without realizing it.

* Make sure that the corresponding scale (gauge) of the decoder matches the scale and gauge of the vehicle to be built. This is especially important for die-cast and brass models such as locomotives.
* Take advantage of DCC connectors such as Next18 and MTC21, etc. Test the operation of the LokSound decoder with an inexpensive decoder before installing it.
* For vehicles that are not DCC-ready, use a mounting auxiliary board such as ExpBoard.
* Speakers have a strong magnetic force, which can cause them to stick to drivers and other devices, resulting in short circuits and broken wires in the decoder wiring.

As for failure factors, general know-how includes the following.

* If the speaker wiring comes in contact with a wire or related signal line such as a function or motor, a short circuit will occur and the internal circuit will be destroyed.
* If a small scale decoder such as the LokSound5 micro is used in a very heavy die-cast HO or O gauge train, and the train is repeatedly driven at high speed or stopped suddenly, the induced voltage from the motor (the motor becomes a generator) can cause the decoder to over-voltage and destroy it.
* The spring-loaded bogies often used in Tomix products are prone to short-circuiting due to derailments or shocks at points, etc., and overheating phenomena can damage the vehicle; be aware that some vehicles are not designed for continuous AC current use at all, such as DCC.
* In the LokSound5 micro, there is a scattered occurrence of initial failures with a probability of 1-2% in the 2019-2020 period. Be sure to check the operation with a decoder tester at the time of purchase.

## 9.8. Can I run it with a PWM power pack?

LokSound decoders will not work with PWM power packs, use the DCC command station or use a Qantum power pack or pure analog power pack.

If you use a PWM power pack, the decoder will not fail, but it will not work properly. The decoder does not work properly, so it may be mistaken for a failure.

## 9.9. What should I do about line voltage?

LokSound decoders are built according to standard DCC specifications, but you can judge them by their scale and gauge. Note that some DCC command stations may have a fixed voltage. Even if it is slightly different, it should work fine. However, in some cases, decoders that do not meet the NMRA standards or DCC decoders from some manufacturers that prioritize cost reduction may induce failures.

| **scale gauge** | **Voltage Range** |
| --- | --- |
| z, N | 12 V |
| HO | 15-16V |

## 9.10. LokProgrammer does not write sounds properly

If you cannot get the sound to work after writing it with LokProgrammer, you may end up with a sound that does not work properly. Please check the following and try writing again. In particular, it is often the case that the communication data is corrupted and the sound cannot be written properly.

* Destruction of communication data due to dirty rails, dirty wheels, or poor contact between rails and wheels
* When installed in a vehicle, poor wiring connections or disconnection
* Firmware is old, incompatible with LokProgrammer to write (update firmware with LokProgrammer)
* LokProgrammer is too new to write well (loss of backward compatibility, writing LokSoundV4 with LokProgrammer 5 or later, etc.)
* LokProgrammer is too old to write well (loss of upward compatibility).
* A bug in LokProgrammer that causes strange writing and unintentional changes in settings (try an older version).
* CV is not resetting properly.
* The decoder is malfunctioning (normal operation is also not possible if the decoder is malfunctioning)

Even if it is written correctly, it may be mistaken as not working if you use the DCC command station incorrectly.

* You forgot to press F1.
* Line power is not turned on.
* The AC adapter is not included.

Even if the sound data can be written successfully, there may be problems with the vehicle itself.

* There is no decoder in the vehicle.
* It has a different decoder.
* There are no speakers in the room.
* Internal disconnection.
* The tracks aren't connected.

If you are not familiar with the LokProgrammer, use a vehicle that has been tested to work properly (vehicle with DCC installed from the beginning) and a decoder tester.

## 9.11. Sounds written by LokProgrammer are strange.

In some cases, open sound data is not released in a perfect state. In some cases, the creator may not intend or be aware of special conditions that cause the problem. Basically, it is up to the users to solve the problem themselves, but please check the following points.

* Failed to write sound data.
* A bug in LokProgrammer caused the settings to go wrong (this happened with the LokProgrammer 5.1.0 version at the end of November 2020).
* Sound data is corrupted.

Troubleshooting methods include the following

* If it happens with a specific sound data, write a different sound and see if it works properly with it.
* Once the decoder is removed from the vehicle, check its operation with a decoder tester (to see if it was installed incorrectly in the vehicle).
* Try changing to another decoder (to see if it is a decoder-specific phenomenon)

## 9.12. ECoS2 rewrites the address without permission

ESU's ECoS2 is equipped with a function called RailComPlus, which can communicate with LokSound decoders. The problem is that the address is rewritten when multiple decoders with same address are recognized. This is the problem that the address is rewritten by this function.

One way to deal with this is to stop RailComPlus from working.

* Turn off the RailCom function except for one car (CV29 bit3 is turned off).
* Do not check "Enable RailComPlus automatic announcement" in the LokProgrammer settings.

This is an ECoS2 specific phenomenon, so it does not affect those who are using the other command station.

## 9.13. I lost the address.

If you use a command station product such as DSair2, you can read the addresses automatically. If you are using a command station that reads the addresses manually, you can check the current address settings by reading all CV1, CV17, CV18, and CV29, writing down the values, and then calculating them using a calculator on the web.

In addition, you can easily check the current address using LokProgrammer.

## 9.14. Some addresses do not work with some command stations

In the American system, 1-127 is a short address (set by CV1), and in the European system, 1-99 is a short address. In the American system, 1-127 is a short address (set by CV1), and in the European system, 1-99 is a short address. 100 onwards is a long address in the European system.

In Japan, due to the popularization of the DCS50K, the U.S.-style controller is the mainstream, but if you have a European-style controller and try to move a vehicle in this address range (there are many vehicles in Japan such as 103, 113, 115, etc. that you would like to use as an address), it may not work and you will have a problem. This can be a real headache.

The only way to do this is to not use 100-127.

## 9.15. Is it possible to write sound data for LokSound 5 to LokSound V4?

Sound data for LokSound5 cannot be written to LokSoundV4. It is not backward compatible. Sound data must be recreated from scratch in order to be created for LokSound V4.

## 9.16. Is it possible to write sound data for LokSound V4 to LokSound 5?

It is possible to write sound data for LokSound V4 to LokSound 5. The data will be automatically converted to LokSound5. Note that the sound quality will be equivalent to LokSound V4.

## 9.17 The volume is too low and I want to adjust it.

Rewrite CV63 with the CV write function (programming function). The default value depends on the sound data; it can be set from 1 to 192, with 100% volume being 128, and 192 being 150% volume (maximum). Set it to the value you like while experimenting.

If you want to adjust the volume while driving, you can temporarily reduce the volume by repeatedly turning F7 on and off (double click).

## 9.18. I want you to provide sound data for \_\_\_\_.

We do not accept any requests or opinions from users. We only list representative sounds that meet our quality standards, in consultation with their creators.

## 9.19. Can I run it in analog?

In some cases, it does not work stably with PWM power packs. If you are using a pure DC analog power pack (including pure analog adapters), you will need to adjust the data in LokProgrammer. Therefore, it is possible to run a vehicle with open sound data in an analog environment, but we do not guarantee that it will work with all analog power packs. Please use at your own risk. We do not guarantee that all analog power packs will work.

Please note that some registered model stores may have their own support, but this does not constitute an official Open Sound Data guarantee of operation. It has been reported that some power packs (the type that emit special PWM pulses) do not work properly with the adapter.

Although not exactly analog, LokSound 5 can also be run as a Qantum compatible mode by changing the settings.

## 9.20. Unable to extract sound data from LokSound decoder

Due to the specifications of the LokSound decoder, it is not possible to extract sound from the decoder to the PC. The only data that can be retrieved is the configuration data (CV).

This is a common specification for all LokSound decoders, not just for open sound data.

## 9.21. There are many LokSound products, which one is best?

LokSound is available in unmarked, micro, XL, and M4 models, but micro is the only option for Japanese N-gauge trains. For N, the Japanese model cars are not designed to be equipped with a decoder, so a small one is needed to keep the detail. For Japanese HO (No.16) plastic cars, micro is not a problem because of its low current consumption. The difference in function also does not affect the Japanese model. If you have more than 3 cars made of brass, or if you have die-cast bodies of HO or #16 locomotives, we recommend using MTC21 type (LokSound5 unmarked).

## 9.22. Do I need a license with ESU for commercial use with open sound data?

Open Sound Data is developed and distributed using sound samples that we have independently recorded.

In addition, the sound data has been recorded and developed in a way that avoids the restrictions in the license document so as not to affect the ESU license. Therefore, there is no need to sign a license agreement with ESU as long as you use open sound data. The LokProgrammer license document also states that the creator of sound data created with LokProgrammer retains the intellectual property rights for all sound data created by the user.

For commercial use, you do not need to sign a license agreement with ESU, but you do need to sign a license agreement with the administrator of the open sound data.

On the other hand, if you want to do business using the ESU's official overseas sound data or the ESU's sound library (a database of sound samples, also known as a template pack), you will need to sign a contract with the ESU. Open Sound Data is a sound library distributed by ESU. Open Sound Data does not use any sounds or sources that are the intellectual property of ESU. This restriction is detailed in the license document (there are two versions, one for sound data and one for LokProgrammer) on the ESU website, so please refer to it yourself.

## 9.23. LokSoundV4, which I wrote over a year ago, does not work or produce any sound.

It has been reported that LokSound V4 that has been placed for a long period of time can cause data loss and other problems. It is said that the function can be restored by rewriting the file several times, but the cause and countermeasure are not yet known. However, there is no known cause or countermeasure for this problem. The best way to prevent this problem is to run the program occasionally without leaving it for a long time.

This has not been confirmed for LokSound5.

## 9.24. Do I need permission from the railroad company?

If you want to incorporate any part of the railroad company's intellectual property rights into any product, product, or data, whether for a fee or free of charge, you must obtain permission from the railroad company. Intellectual property rights include copyrights, neighboring rights, trademarks, and patents. Model railroad vehicles and logos are usually designed and trademarked by the railroad companies, so this is why model manufacturers need to obtain permission.

On the other hand, there are some model car manufacturers that sell model cars without obtaining permission, because those for which the company has rights and for which the number of years protected by the Copyright Law has passed since the design was made public (\*) or for which the trademark rights are not maintained are no longer protected. For example, Japanese National Railways railroad cars up to the mid-Showa period have already been removed from protection as of 2020.

Open Sound Data does not include any of the proprietary intellectual property rights of railroad companies (logos, car designs, departure melodies, etc.) or music that is held in trust by copyright management organizations. For example, if a train name is registered as a trademark, it will be replaced with the name in common use. In addition, the trademarks of the respective companies are clearly indicated.

However, we will not use any material that is already in the public domain, or any material that, by definition under the Copyright Act, does not give rise to copyright rights for the railroad company, car manufacturer, or other company (such as the sound of machinery, noise by general definition, non-musical sounds, broadcasts, chimes, buzzers, warning sounds, etc.). We have checked the opinions of experts, laws, and past precedents, and use them to the extent that there are no problems.

\*Note: When copyright is held by an individual, the term of copyright protection (50 or 70 years, depending on the law revision) is stipulated after the individual's death. If the copyright is held by a company or organization, the term of protection is stipulated from the time of publication.

## 9.25. What is the copyright of the sound data?

The following is a summary of copyright and neighboring rights for sound.

http://powerele.sblo.jp/article/185428579.html

Copyright does not automatically accrue to everything, and as stated in the law, certain conditions must be met for it to be covered by copyright law. For example, copyright is not granted for the sounds of machines, things that are treated as noise by general definition, voices that are not music, buzzers, warning sounds, etc. For more details, please refer to the summary page of the above blog, which introduces websites with the views of intellectual property experts.

On the other hand, if a sound is recorded that does not have a copyright, the recording artist is subject to neighboring rights. For this reason, in Open Sound Data, we ask sound providers and creators to provide us with their own recordings (in principle, those that have been declared Creative Commons CC0), which are then processed and edited by the creators before use. Since copyright is generated by the processing and editing work, the creator holds the copyright to the Open Sound Data data. Please understand that there is a copyright on the processed and edited sound data as well as the sound itself.

## 9.26. Model stores and model manufacturers. How can we use open sound data to do business?

There is no charge for permissions. There are a few companies that do not cooperate with the promotion of DCC, do not pay, do poor work, or resell their products at auctions for a high price. There are some people who do not want to cooperate in the promotion of DCC, or who do not pay their bills, do poor work, or resell their products at high prices at auctions.

When granting permission, we also ask for your cooperation in the Open Sound Data activities.

Please note that Open Sound Data only provides the data. The operator of Open Sound Data does not provide any support for various problems, malfunctions, or failures that may occur at model stores or model manufacturers, as these problems should be handled by ESU.

If you report any bugs in the open sound data we provide, we will try to fix them as soon as possible, but we cannot guarantee that we will.

## 9.27. I heard that LokProgrammer is not available for commercial use?

The following is a translation of the part of the ESU license document that describes the items related to the created sound data.

You are granted a license to use, modify, and combine sound samples provided by third parties to create your own sound compositions, but you are free to distribute them as you wish, provided that you comply with the terms of this Agreement, and provided that you You may distribute it as you wish, provided that  
 You may freely distribute the Sound Samples as you wish, provided that:   
(i) the Sound Samples are mixed to create an original work;   
(ii) the individual Sound Samples are not used in isolation; and (  
iii) the Sound Samples or their derivatives are not used to create a sound library for distribution to third parties.  
  
 You are free to distribute them as you wish, provided that.

This could be interpreted as prohibiting the creation of a completely different collection of sound samples (sound library) from the sound data that ESU owns the rights to, and the processing of that data. One might think that an expansive interpretation of a sound library would include open sound data, but ESU's definition of a sound library seems to be a collection of sound samples (http://www.esu.eu/en/downloads/sounds/ generation-3/loksound-sound-library/).

# 10. at the end

We would like to thank the following people for their cooperation in running Open Sound Data. We would like to thank them.

## 10.1. Creator and sound source provider

Mr. MB3110A, Mr. Kawakei, JR Hamamatsu, Tochigi General Rolling Stock Terminal, Mr. Henohenomoheji, Michael of Adatara, Mr. Uedanejiro, Pacific231, Mr. Salam, Tomon Ikkoku, Mr. YOMIX, Mr. yusa, Mr. Katsuta Kobo, Mr. HK1000, Mr. HISAO KOBAYASHI, KC Mr. Tanoue, Mr. Nagoden, Mr. Fujigaya2, Mr. MECY, Mr. Nucky

## 10.2 Groups and organizations

Mr. Osaka Kameya, Mr. Ishida Shoten

## 10.3 MIT License

MIT License

Copyright (c) 2021 DesktopStation / Yaasan

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. SOFTWARE.

Yaasan (DesktopStation), Henohenomoheji(TRAINO)  
Tokyo JAPAN  
This document is copyrighted by DesktopStation and may be used under the MIT License.

Copyright(C)2020- DesktopStation Co.,Ltd, DCC Electronic Arts Coalition