

Instruction Manual

ZIMO App

QUICK START GUIDE for the first official
version of the ZIMO APP

Table of contents

1.0. CONNECT Connecting command station, libraries, etc	Page 3
1.1. SYS DB (V) – system-wide database for vehicles	Page 4
1.2. APP DB (V) – the local database	Page 5
1.3. APP DB (V) – ADDING, CLONING ADDRESSES, etc.	Page 6
1.4. LOCO – the screen for controlling vehicles	Page 7
1.5. LOCO – Assignment of GUI elements	Page 8
1.6. The "blue needle curve" – meaning and creation	Page 9
1.7. LoR and LoR active – the favourites list	Page 10
1.8. SYS DB (A) and APP DB (A)	Page 11
1.9. APP DB (A) – GUI elements Accessory Part Library	Page 12
1.10 Operational Mode CV Programming OP PROG	Page 13



INSTALLING the ZIMO APP from the Google Play Store:

NOTES on the version currently available in the shop:

This is the first edition available via the Google App Store, making it the first official version of the ZIMO App. This does not mean that it is a finished product. In the coming weeks, months and years, there will be numerous software updates, which will be easy to manage via OTA (over-the-air), some of which will interact with other ZIMO system products and decoders.

To get straight to the point: the ZIMO app is part of the ZIMO control system (MX10 or MX10EC, MX33, StEin modules, etc.). There are no plans to make it accessible to other digital systems – this would result in restrictions based on the "lowest common denominator" principle – and there would be a constant additional effort to maintain compatibility with third-party products. How this blocks innovative projects can already be observed in the standardisation committees for digital technology as a whole.

One more – perhaps somewhat disappointing – note: the ZIMO app does not aim to give disused mobile phones a new purpose. The performance requirements are quite high, which is no problem for current devices; difficult to specify, more detailed information to follow.

The ZIMO libraries are of great importance in this and other new ZIMO products (or software versions); a brief explanation can be found on the CONNECT screen page in this manual.

A note on the next planned enhancements to the ZIMO app software (updates): System-controlled traction operation, addition of signals in the "Accessories" section, Service Mode Programming, extension of Operational Mode Programming (topic-oriented), CV #300 – Procedure, Registration (including according to RCN-218), ...

A video tutorial on the ZIMO app is expected to be released at the end of March or beginning of April 2026. It will provide a comprehensive overview of all functions and their practical application.

We wish you every success in using the app. As the application is continuously being optimised and expanded, feedback for further improvement is always welcome at office@zimo.at.

(Note: The symbol ↗ means, the display has changed in the meantime, new screenshots coming soon!)

1. Open the Google Play Store on an Android smartphone.

2. Tap the magnifying glass "Search" at the bottom to perform a search.

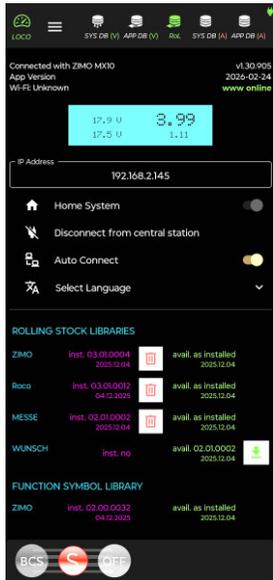
3. Type **Z** into the search field that opens at the top and enter ZIMO App. Press Enter or tap the magnifying glass on the keyboard to

4. Tap on the line with the ZIMO app you have found...

5. ...the detailed app window with screenshots opens. Pressing the blue Install button starts the download.

6. The ZIMO app will be downloaded and will be installed automatically. Within a short time, an Open button will appear to start the app, which will also appear on the smartphone's home screen.

1.0. CONNECT – Setting up and controlling the Wi-Fi connection to the command station, online update of libraries, etc.



Swipe up the screen to view the full library lists.



This is where the app starts, and you can return here at any time if necessary (icon "CONNECT" in the header or via the menu in the header).

To connect to the MX10/MX10EC command station via WLAN, connect it to a router or home network via LAN. After starting the ZIMO APP and opening the "Connect" start screen, enter the IP address of the command station. This can be found in the MX10 menu under "PC Config+Monitor" and then "LAN IP" (where it can also be changed).

On the CONNECT screen; the following information and actions:

- **Header:** Menu  to all screens of the app and shortcuts to selected screens (here: **LOCO**, **SYS DB (V)**, **APP DB (V)**, **LoR**, etc. In the top right corner: **green/yellow/red** icon for **the status of the Wi-Fi connection** to the command station; on all screens in this location.
 - Display of the **active Wi-Fi network** and the **software version** of the connected **command station**
- Replication of **the MX10 displays** (voltage and current values for both rail outputs) of the connected command station; this is mainly used to quickly check the connection after entering the IP address and Wi-Fi.
- NOTE: The connection is only established if this display is illuminated (usually blue) and the numerical values are not all displayed as "zero"; if this is not the case, the app must be closed and restarted, and possibly also the MX10.
- Set the **IP address** of the MX10 command station to which you are connecting. NOTE: WLAN networks – app and MX10 – must match!
 - **HOME system** switch; this allows the connected command station to be declared the HOME system for the app device.
 - **Connect/disconnect** the ZIMO app to/from the command station via Wi-Fi.
 - **Auto Connect** (recommended).
 - **Language** setting.
 - Display of the **app version**

The ZIMO Libraries:

In the ZIMO context, "libraries" refer to collections (of locomotive images, function symbols, etc., including attributes and search criteria) that are **provided by ZIMO** (or business partners) and **continuously updated** with additional entries. The latest versions of the libraries can (and should) be loaded into ZIMO control devices, i.e. controllers, the ZIMO app, and computer programmes such as ZSP or ZCS. However, **users** of ZIMO devices can also create **their own libraries** and load them additionally.

In ZIMO terminology, libraries are to be distinguished from "databases", which are user-generated lists in the system or control devices. For example, the SYS DB (F) – the system-wide database of vehicles prepared and activatable in the system (with images, symbols from libraries, **but also current driving data**) or the SYS DB (W) – the counterpart for accessories such as points and signals.

The most important (and existing from the outset) libraries are:

- ZIMO Rolling stock library:** locomotives, carriages, consists (ev), ... photos and parameters (including for SEARCH),
- ZIMO Accessory parts library:** Points, signals, layout equipment, etc. Graphics, parameters
- ZIMO Function symbols library:** Function symbols for function keys, etc. Graphics, parameters

The CONNECT screen shows which libraries are installed in the app, and new editions (updates) or completely new available libraries can be downloaded.

Not yet (Feb. 2026), but a little later: Creation of a USB stick for ZIMO controllers to load the libraries into devices without online connection.

On the CONNECT screen, as on all other screens, always at the bottom left:

- Buttons for **BCS** (= collective stop) and **OFF** (= track power out). To prevent incorrect operation, activation always takes place in two stages: by touching **S** (to "arm" the other buttons) and then touching **BCS** or **OFF**. Switch back on by **pressing BCS** or **OFF** (without **S**).

1.1. **SYS DB (V)** – the system-wide database for vehicles and vehicle addresses (same content as SYS DB in MX10 and controllers)



Scroll up the screen to see more rows.



DELETE button



The system-wide **SYS DB (V)** resides as an original in the MX10 or MX10EC command station; it is loaded, displayed and edited in **ZIMO controllers** (called up using E+6), in **ZIMO apps** as **SYS DB (V)** and in external programmes.

Loading into the ZIMO app happens automatically after start-up or by touching the REFROUND button; this should not actually be necessary, because the synchronisation of the databases between devices, controllers and apps happens automatically. But there are cases where this does not work ...

The vehicle addresses contained therein also originate from controllers (A button → **LOCO IN**) and ZIMO apps – see ADD button on the APP DB (V) screen – as well as from external sources (computer programmes such as ZCS, ...) or (planned) directly from decoders (INVENTORY SEARCH, i.e. by logging in according to RCN-218, ...).

Above the database lines themselves, a **sorting sequence** can be selected (name, address, priority = line colour).

The entries in the **SYS DB (V)** consist of **green, turquoise, blue and grey lines**, with the **colour allocation** depending on previous use in the app: after initial start-up and loading, only **grey lines** are initially displayed.

Green lines: also included in APP DB (V) and **LoR** (loco recall *), usually already operated by this device.

Turquoise lines: **LoR** as **green**, but address in system-controlled double or multiple traction (marked T1, T2, ...).

Blue lines: also included in APP DB (V), but not in LoR, usually imported from SYS DB (V), but not driven.

Grey lines: not in this APP DB (V), but only in **SYS DB (V)** and possibly in APP DBs of other controllers or apps.

The lines of the **SYS DB (V)** contain **information about the vehicles** or vehicle addresses **in question**; the same line structure is also used in the **SYS DB (V)** database, excerpts from it are also used in other cases!

Name column: if a name has been assigned to this address; otherwise, automatic replacement name

Address column: Vehicle address, if applicable (**turquoise row**) reference to affiliation with a traction unit

Column Vehicle image: if an image has been assigned to this address, a miniature version is displayed here

Column Additional information: including external control (FS), if in own APP DB (L) but driven by another device

Column DCC direction/current direction of travel: Forward/reverse and east/west; **currently** only if **RailCom message**

Column DCC drive level/current speed: Drive level 0 ... 126 (or 0 ... 28) / **current (km/h)** only if **RailCom**

Function table column: States of functions F0 ... F9; enlarge miniature display by touching!

Direct activation from the **SYS DB (V)** for **driving mode (LOCO IN)** screen) of a vehicle is done by touching the relevant line – in the **Name** or **Address** column area – regardless of the type (colour) of the line.

The **blue sliders** – only effective in the **grey rows** – are used to select which addresses from the **SYS DB (V)** are to be transferred ("imported") to your own **APP DB (V)** (i.e. in this device) – this is done for all selected rows using the **IMPORT** button; it is only effective for the **grey rows**, because otherwise they have already been imported in the past. In the **SYS DB (V)** – i.e. in the current view – the rows turn **blue** as a result.

NOTE: As described above, **"implicit importing"** is also possible by activating directly from the **SYS DB (V)**.

The **red sliders** are used to select which addresses are to be deleted – this is done for all selected rows using the button and the **DELETE** button that appears afterwards. The relevant lines are then marked with >Recycle bin<. This refers to the **system-wide SYS DB (V)** – also in the command station! – where the line will no longer be available at the next system start, but can be reactivated from the recycle bin in the meantime. At the same time, it is also deleted from **APP DB (V)** and **LoR**, if necessary.

The effect of **DELETE** is therefore the same as deleting grey lines from the **SYS DB** in the MX32 or MX33 controller.

ATTENTION: Deletion may become ineffective if the relevant line "returns" via another controller or app.

*)"LoR" = ZIMO traditional term for "loco recall" = favorites list; i.e. the addresses that have already been active (in **LOCO**).

1.2. APP DB (F) – the database of "locally" available addresses (the counterpart –to the CAB DB in the controller)



After Touch Mozart 200 – line comes the LOCO (IN) screen



The lines of the APP DB (V) contain all vehicle addresses that were previously **selected and imported** (using the IMPORT button) into the SYS DB (V) at the start of operation or during operation. It also contains the **implicitly imported** addresses, which were activated directly from the SYS DB (V), and of course those addresses that were **created** in the APP DB (V) itself, i.e. using the ADD button.

The type of rows – represented by the colours **green, turquoise and blue**; grey does not occur – and the columns – (*name, address, etc.*) of the APP DB (V) are the same as in the SYS DB (V); **see lists on the previous page!**

New rows (IMPORTED from the SYS DB (V) or ADDED from the APP DB (V) itself) are initially **blue**; after initial activation, they turn **green** or, after being integrated into a traction, **turquoise**.

The APP DB (V) is a kind of **"central processing station"** of the ZIMO app for vehicle addresses and vehicles: creating and adding **new addresses**, defining and changing **names, cloning vehicles, moving them** between databases (sidebar "DATABASE ACTIONS"). *See next page!* However, the assignment of vehicle images and function symbols is not done here, but in **LOCO**. Most important in the APP DB (V):

Touch a line in the Name or Address column → Activate = Drive in the LOCO (IN) screen and simultaneous recording in the **LoR** (loco recall). This applies regardless of the type (color) of the line.

◀ *In the example, the line "Mozart ..." of the APP DB (V) is touched, whereupon the LOCO screen for this locomotive is opened. In the example, the appropriate image and function symbols are already linked to the vehicle address, as they were previously assigned on this device or on another, possibly on an MX32 or MX33 controller. An "empty" address could also be activated from the APP DB (V), and the assignment of GUI elements in LOCO could be carried out with the help of the libraries.*

The **GUI and driving data** are automatically kept **fully synchronised** between SYS DB (V), APP DB (V), LOCO and LoR screens, and wherever modifications may still be taking place. This applies not only within the **ZIMO app**, but also with the **system devices** (MX10 command station, MX32 and MX33 controllers), where synchronisation is also sought (sometimes limited by temporary connection failures).

The full APP DB (V) is retained even after exiting and reactivating the ZIMO app, and even after loading the SYS DB (F) anew.

On the APP DB (V) screen, to the right of the actual database, there is a **"narrow" version of the LOCO screen or speed controller** that was last active. On this screen, the train in question can continue to be driven as before, with certain restrictions: using the slider for speed control, the direction button and the mini function panel, which can be enlarged if necessary.

Touch button "DATABASE HANDLING" → instead of the "narrow speed controller" – i.e. to the right next to the APP DB (V), several icons are displayed for **manipulations in and between databases**.

In this state the lines of the APP DB (V) are used to **highlight lines** (including multiple lines) by highlighting them in grey; then the icon for the desired action is selected. This is why the display message **"Activating a vehicle is not possible at this time"** appears. This refers, among other things, to "artificial" **insertion into/removal** from the LoR (which causes the colour of the line to change between **green** and **blue** to indicate **removal** from the APP DB (V)), or to **complete deletion**.

On the APP DB (V) screen, as on almost all other screens, always at the bottom left:

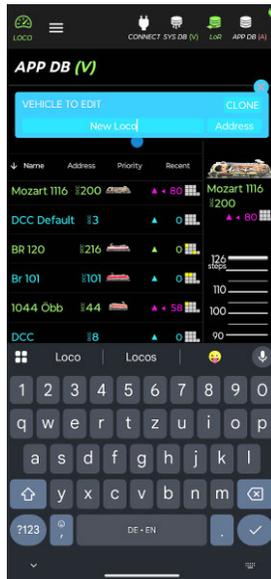
Buttons for **BCS** (= collective stop) and **OFF** (= track no power). To prevent incorrect operation, **activation** always takes place **in two stages**: by touching **S** (to "arm") and then touching **BCS** or **OFF**. Switch back on by **pressing BCS** or **OFF** (without S).



After touching the icon "... LoR ...", the previously selected BR 120 line turns green.



1.3. APP DB (V) – Create and ADD addresses, CLONE vehicles, EDIT vehicle properties ...



To use this part of the APP DB (V) functionality:

Touch the "ADD vehicles, ..." button "→" input block (instead of the previous two buttons) with the input fields Name, Address, CLONE button; touch a field → matching on-screen keyboard.

There are three different processes that can now be started using the input block:

Add a newly created address

both to the APP DB (V) itself and automatically to the SYS DB (V) and the entire system (MX10, controllers):

- Tap on the Name field (to bring up the keyboard),
- type in the desired name in the Name field,
- finish with the "OK" button (at the bottom right of the keyboard) to initiate automatic keyboard change,
- Tap on the Address field,
- enter the desired address in the Address field,
- finish by pressing the "OK" button (at the bottom right of the keyboard).

* If the vehicle does not need a name, do not type anything in the field, but press the "OK" button on the keyboard immediately. The "New Locomotive" (example) with DCC address 69 (example) is now in the APP DB (V), also in SYS DB (V), etc. The GUI elements (vehicle image, function symbols, etc.) are not assigned here, but later in LOCO.

Modifying the name of a vehicle in the APP DB (V):

- Touch the relevant line to automatically transfer the name and address to the input fields.
- Tap on the Name field (to call up the keyboard),
- Modify the name in the Name field,
- and confirm with the "OK" button (at the bottom right of the keyboard).

The vehicle with the modified name is now in the APP DB (V), etc.

Creating a "clone" of an original vehicle from the APP DB (V) – IMPORTANT: the clone has identical function symbols,

- Tap on the line of the original vehicle to highlight it with a grey background,
- Tap the CLONE button to bring up automatic suggestions for the name and address of the clone.
- Tap on the Name field (to bring up the keyboard).
- Modify the suggested name – as for the original with (2) – in the Name field,
- Finish with the "OK" button (at the bottom right of the keyboard).
- Tap on the Address field,
- Modify the suggested address in the Address field.
- Finish by tapping the "OK" button (at the bottom right of the keyboard).

The vehicle with the modified name and address is now in the APP DB (V), etc.

The GUI elements (vehicle image, function icons, etc.) are identical to the original vehicle; they can of course be modified later in LOCO.



1.4. LOCO – the screen for controlling vehicles



This is the screen for controlling a locomotive (in future also for traction units and trains); it can be called up from the APP DB (V) or LoR screen by touching a line (name, address field), or by using the marked locomotive names or addresses to the right and left of the image of the currently active locomotive in LOCO IN.

The basic layout of the LOCO screen is similar to the MX32/MX33 controllers.

The examples on the left show a vehicle with RailCom feedback (hence the SPEEDO with a magenta needle to display the km/h value and a magenta ring around the hub of the SPEEDO). The ZIMO APP uses a replica of an ETCS speedometer (ETCS = European Train Control System) instead of the SPEEDO discs on the controllers. The target speed, which is calculated in advance using the "blue needle curve" (see following chapter), is displayed by circular arcs (light grey during acceleration or yellow during braking).

Later (as part of the expansion of the ZIMO system), the speed profile of the track will be decisive in accordance with ETCS, with braking curves and braking bars on the display. The current light grey circular arc is not ETCS-compliant, but is useful in the experimental phase.

The direction field (similar to the direction button with LEDs on controllers) shows magenta direction arrows for V/R and east/west with ZIMO RailCom direction feedback, while during the direction change, yellow-green arrows appear after pressing the button as a preview of the future direction.

For a vehicle without RailCom feedback, the speed and direction values can only be displayed as specified by the DCC commands: therefore, no magenta, but a SPEEDO with a blue needle, blue display of the km/h value and blue ring around the hub of the SPEEDO, as well as blue direction arrows.

Currently (early 2026), true direction feedback (and the magenta display described above) is only available with ZIMO decoders; otherwise, there are only blue direction arrows indicating the DCC direction.

The H "Stop" button acts as an emergency stop button; it must be pressed while the slider is not set to "0" or (even if set to "0") but still in the braking phase. Note: the effect of "Emergency" can be set in the decoder configuration; for ZIMO decoders, CV #111 specifies a replacement braking time (instead of CV #4).

The function keys are arranged as a scrollable list (3 fields next to each other), divided into fields of 10, similar to the controller, with sliders in between for displaying and setting parameters (theme organisation, time, analogue values, etc.).

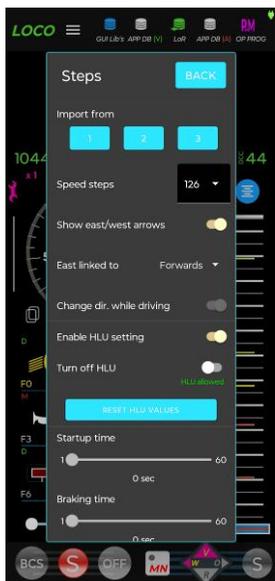
The "Levels" settings menu (button with cogwheel symbol) can only be accessed when the slider is set to "0" can be used to define a range of display and functional measures.

The adjustability of the HLU limits can also be enabled in the slider here: the CV values for UH, U, L, KU, FL (CVs # 51 ... 55) can be changed using the bars in the slider area. The new values are immediately programmed into the decoder and are therefore effective immediately.

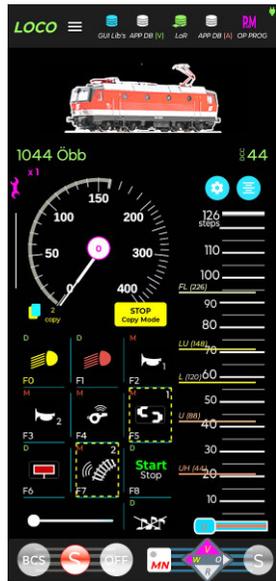
ATTENTION: The speed limits are specified in the app in drive levels (scale 1 ... 126), while on the MX33 they are specified and programmed in the form of internal drive levels, which means approximately double the value. Please note: some of these options are not yet implemented (as of early 2026)!

The "Scale" settings menu (button with scale symbol) contains further variants. As with "Steps": some of these options are not yet implemented (as of early 2026)!

The "Curve" settings menu (button with curve symbol) can only be accessed when the slider is set to "0". Please note: these options are not yet implemented (as of early 2026)!



1.5. LOCO – Assignment of GUI elements (vehicle image, function symbols) from libraries or copying between addresses



Vehicle image (from the "Rolling Stock" library) and function symbols (from the "Function Symbols" library), i.e. part of the GUI elements for a vehicle (for one address), are assigned to addresses in different ways from different sources (depending on the situation, capabilities of control devices and decoders):

- Assignment of GUI elements on the ZIMO controller (MX32, MX33); automatically transferred to the ZIMO app
- Creation of the GUI using external software (such as ZCS), saving it in the ZIMO system from there; retrieval from controllers, app,
- Creation in the ZIMO app, LOCO screen (i.e. here), by selecting from libraries of vehicle images and function symbols,
- Also in the ZIMO app, LOCO (IN), by copying and pasting from existing GUIs of other addresses, (advantage: additions already defined in the copy address, such as mini sliders, etc., are also transferred)
- Retrieve GUI from the decoder using ZIMO file transfer (only with newer ZIMO decoders ..., or according to RCN-218).

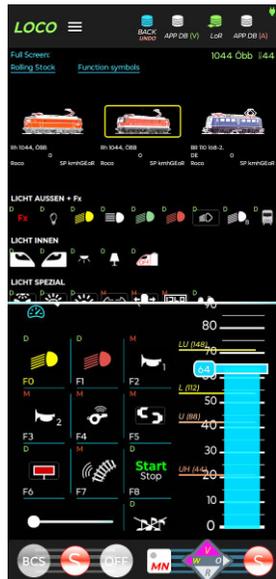
Touching the library icon in the header line on the left brings up excerpts from the Rolling Stock Library and the Function Symbol Library in the upper half of the screen. By touching "Rolling stock" or "Function Symbols" under the heading "Full Screen", a larger display of one of the two libraries can (but does not have to) be shown. The desired vehicle image or function symbols (first the first function symbol, then the others) can be selected

by scrolling horizontally/vertically and touching within the list of vehicle images or within (first) the function symbol groups (the rows) and then the individual symbols.

The search for a vehicle image is supported by a filter tool (any 3 character strings) if required.

In the case of vehicle images, only one is needed; the last touch on such an image is therefore valid. In the case of function symbols, numbered frames are assigned when touched – if desired, several can be marked in succession – and then copied in succession to the function key fields in the lower half of the screen based on the numbers.

The selection procedure is completed by touching CONFIRM (top left) or BACK (UNDO) (next to it).



Touching the Copy&Paste symbol (initially white, below the SPEEDO on the left) initiates the copying of function symbols between addresses: by touching again, the active address can be defined as the copy or paste address.

In the address that is currently the "copy address", function symbols are marked with a touch, which creates a numbered frame and copies them to the clipboard. After another address is made the "paste address" (or has already been set to it), the symbols from the clipboard can be transferred there. The locations (Fu numbers) of the copy and paste addresses do not need to be identical.

The clipboard can also be emptied several times in succession (for different "paste addresses"). Conversely, a "paste address" can also be defined, and the required symbols are "collected" in changing "copy addresses".

Copying within an address (moving function symbols) is also possible by alternately defining it as a "copy address" and a "paste address".

When copying and pasting between "copy" and "paste addresses", not only the symbols themselves are transferred, but also the additions (such as mini sliders, DMZ = duration/moment/time settings, etc.).

1.6. The "blue needle curve" – meaning and creation – method and chapter currently (early 2026) under construction!



What is meant by "blue needle curve"?

The "blue needle curve" is a graphical representation of the relationship **between drive level** (usually 0–126) **and speed** in kilometres per hour. Depending on the decoder type and its settings (CVs #2, #5, #6, or #67 ... #94), this relationship may be linear or non-linear. However, there are also external influences, such as brake sections, HLU, ABC, etc.) or mechanical conditions.

The "blue needle curve" determines what the SPEEDO in the app or controller displays when there is no feedback about the actual speed from the locomotive.

ZIMO decoders provide feedback on **the actual speed via RailCom**, whereby the underlying internal vehicle measurement exhibits a high degree of linearity, and by adjusting a single CV (# 136: "RailCom km/h factor") it is already possible to receive the "finished" scale km/h value from the vehicle.

However, there are cases where the actual speed is still not available, including ...

- when (third-party) decoders without RailCom speed reporting are in use,
- if RailCom reception is temporarily disrupted (e.g. illuminated carriages on the track),
- if a future speed is to be predicted (and not the current speed).

... and therefore the **speed** to be displayed in km/h can **only** be **reconstructed** from the **drive level**, for which the **"blue needle curve"** is needed.

How is the "blue needle curve" defined?

The "blue needle definition" works as follows:

- Call up the tool icon in the upper left corner of the SPEEDO to open the diagram for the "blue needle curve" to be created, either with a default curve or (by touching "Import") the previously valid "blue needle curve" from the command station (set from the controller).
- Drive initially at a low drive level (e.g. approx. 30), the SPEEDO displays the RailCom speed from the decoder (magenta hub ring and lettering in hub) or (if no feedback, blue hub ring) a speed determined from the drive level (via default or system "Blue needle curve"); at the same time, a segment of the curve is formed in the diagram, initially in magenta (if RailCom speed, otherwise blue); this can be corrected using the "Correction Slider" to the left of the SPEEDO. After stabilisation, the new value is confirmed and **fixed** by **touching the hub**, visible in the diagram as a blue line.
- Repeat the process with higher drive levels (around 60 and 100).
- The "BACK" symbol makes the new curve effective for use (see above).

1.7. LoR and LoR active – the favourites list with the traditional designation "LoR" = loco recall



A vehicle address or a vehicle enters the LoR either

- implicitly by **activation**, e.g. from the *DB APP (V)*, which puts it into the **LOCO** operating state – and thus automatically into the **LoR**, where it remains for "retrieval" even if another vehicle has become active in the meantime – or
- by clicking the **DATABASE HANDLING** button on the *DB APP (F)* screen and then "Add to LoR".

Individual addresses can be activated from the LoR as well as from the *DB APP (V)* (i.e. the operating state or **LoR** screen can be opened) by touching the relevant line.

The **LoR** can be displayed in two ways: "**LoR**" and "**LoR active**":

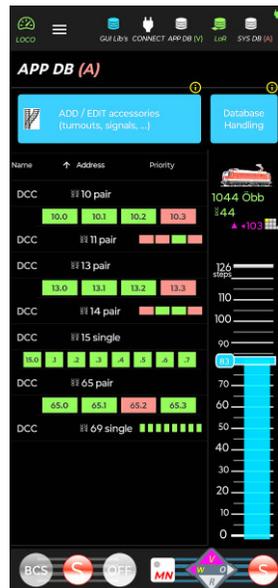
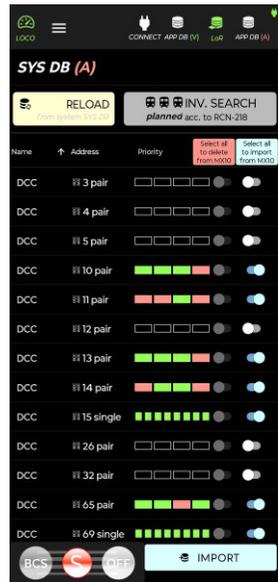
- as "**LoR**", i.e. a list formatted in the same way as the *DB APP (F)*, but only the green (colour of *the LoR*) lines; therefore, there are fewer lines and the desired vehicle is easier to find.
- as "**LoR active**", with its own speed controller (horizontal), direction and MN button for each address in *the LoR*. This means that several vehicles can be displayed and controlled simultaneously. When changing driving data (speed, functional states, etc.), the address in question has the highest priority in the transmission cycle; otherwise (i.e. "only" in **LoR** priority, but not in **LOCO**), the transmission frequency is not as high as when active in LOCO state (= "in the foreground").

The LoR is also the platform for forming tractions!

To be continued!

At the beginning, all entries are grey!

1.8 SYS DB (A) and APP DB (A) – the databases for accessory addresses and items



Unlike *SYS DB (V)*, *SYS DB (A)* does not correspond to a corresponding database in MX10 or in the controller; however, it is treated very similarly to *SYS DB (V)* in the app,

i.e.: IMPORT from *SYS DB (A)* to *APP DB (A)* using the **blue slider**, deletion of entries from the *system-wide SYS DB (W)* – i.e. in the command station (!) using the **red slider**, i.e. first move to the >recycle bin<; as a marker that the line will no longer be present the next time the system is started.

The *SYS DB (A)* in the ZIMO app looks similar to an "ACC LIST" in the ZIMO MX32 and MX33 controllers (see MX33 operating instructions, chapter "ACC LIST").

Here is an example of an ACC LIST in the MX33:



INSERT FOR EXPLANATION

In the DCC world, the addressing of vehicles and accessory items is organised differently:

1 vehicle address (address space 1 – 10239) = **1 decoder** = (normally) **1 vehicle**

1 accessory address (address space 1 – 511) = **1 decoder** for **4 two-term items** (left/right, etc.) = **4 subaddresses** or **8 single-term items** (on/off, etc.) = **8 sub-addresses**

(Instead of "1 decoder", there can be several, all with the same accessory address but different sub-addresses) or (without sub) for **1 "extended" accessory item** (signal, ...)

In some digital systems, addresses and sub-addresses are numbered consecutively (i.e. 1, 2, 3, 4, 5, 6, etc.) and the items are addressed by their numbers. ZIMO uses the NMRA and VHDM original representation (i.e. 1.0, 1.1, 1.2, 1.3, 2.0, 2.1, etc.).

The logic described above applies uniformly across manufacturers for "two-term" accessory items, i.e. left/right turn-outs, red/green signals, shunting and blocking signals, etc., as well as "single-term" items, i.e. single lights, uncoupling magnets, etc. The **accessory addresses** in the DBs are marked with [pair] or [single].

The above-mentioned "extended" accessory items are generally "multi-term" signals that have a single accessory address but more than two terms, referred to as "aspects" or "signal aspects". However, over time, a variety of coding has emerged that must be taken into account by its own configuration logic.

"Extended" is a term from the standardisation of NMRA and Railcommunity and describes the variant intended for standardisation there, but it could also be applied to other methods that have already been introduced individually.

Accessory items are generally activated by touching the buttons on the address lines in APP DB or SYS DB (A).

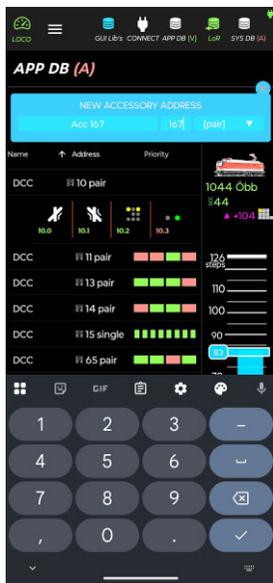
The *SYS DB (A)* contains lines for the accessory addresses available in the system, including buttons for the accessory items:

- (for 4 two-term items [pair], i.e. points, etc.) displayed with name, address, 4 left/right fields or symbols,
- (for 8 single-term items [single], i.e. single lights, etc.) displayed with name, address, 8 on/off buttons or symbols,
- (for 1 "extended" item [extended], i.e. signals, etc.) displayed with name, address, current position code or symbols.

Symbols can be displayed if they have been previously assigned to the accessory addresses (which is done in *APP DB (A)* or comes from the decoder via automatic registration).

The *APP DB (A)* is basically structured in the same way, but it contains "only" a subset of the accessory addresses, namely those that are to be operated by the app in your own device and which have therefore been imported (see above). The buttons can be enlarged by activating the address (touch on address or name) and then contain the addresses and sub-addresses or – if assigned – symbols.

The *APP DB (A)* provides access to extended display formats, including non-address-oriented ones, i.e. freely positionable positions within "tableaux" and (as soon as available) "display tables" within the app.



ADD new accessory addresses, EDIT existing ones in the APP DB (A)

First, **touch the button** "Accessories (turnouts, signals, etc. ADD, EDIT" (only visible in the APP DB (A) display) to start the ADD, ... procedure; An input block with input fields appears, and after touching one of the fields (name, address, mode, etc.), a keyboard appears.

Two different processes can be carried out here (in the input block mentioned above):

Create a new accessory address and add it to both the APP DB (A) itself and the SYS DB (A) and in the system centre (MX10 command station): Enter name (optional) & address & "OK" (all via keyboard).

In the example, accessory address 400 (for 4 pair functions) is added and inserted in the appropriate place. At the time of adding, the accessory items have not yet been used and the fields are therefore empty.

Editing the name of an existing accessory address:

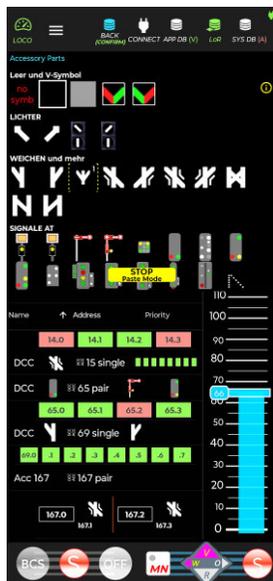
Select the existing line with the desired address in the APP DB (A); modify the existing name for this address or enter a new one.

Several (2, 3 or up to 4) accessory addresses can also be added as **an** automatically generated **sequence address group** with a common name, which sometimes makes operation clearer. This is done by entering "pair 2 x" or "pair 4 x", or "single 2 x", etc. in the field provided.

The **REFRESH** button should not (hardly) need to be used, because the synchronisation of the object databases between the command station and the ZIMO app is automatic. However, there are cases where this does not work.

However, if the ZIMO app is not connected to the controller when a new address is added, **"RELOAD"** must be pressed. The **"RELOAD"** button is also necessary if the smartphone or tablet is connected to a different command station than before. **"RELOAD"** removes all previous entries from the SYS DB (A) of the app (and not from the command station) and transfers those of the new command station.

1.9 APP DB (W) – Assignment of GUI elements (symbols for accessory items) from the Accessory Part Library



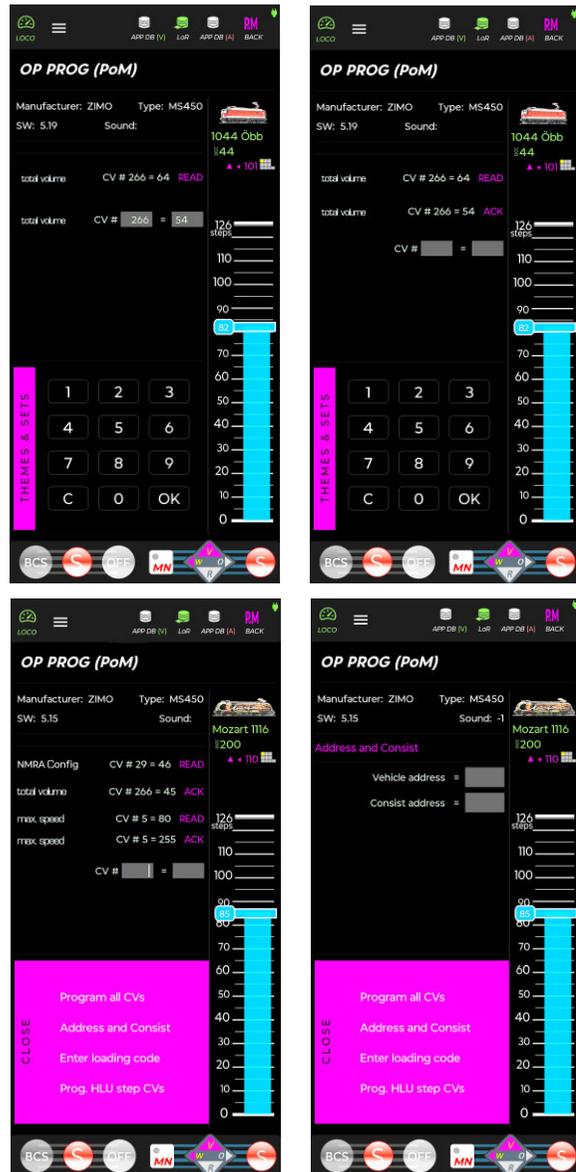
In this situation (i.e. when APP DB (A) is on the screen), tapping on  to display databases opens the **Accessory Parts Library**, which is displayed in a similar way to the Function Symbols for vehicles.

From there, symbols can be transferred to the fields of the APP DB (A) using copy & paste.

◀ In the example (left), the screen is shown after opening the library; then the desired symbols are assigned to decoder: 11 with address 11, and the assignment procedure is exited (BACK button), after which the normal screen of the APP DB (A) reappears with the newly entered symbols.

APP and INSTRUCTIONS ARE CONTINUED (entry procedure for SIGNALS) !

1.10 Operational Mode CV Programming in Operational Mode (= OP PROG, also called "PoM")



To use this part of the APP DB (V) functionality:

- Touch the "PoM" button (in the header, on the right) → "PoM" screen (Programming on the Main = OP PROG)
- ◀ The first input line appears, namely input fields for the first CV number and the corresponding CV value:
 - ... **Programming** a CV using the on-screen keyboard (with or without feedback possible, see below for result messages) (if not automatic) **Touch on 1st input field - enter CV number - OK - enter CV value - OK - message**
 - ... **Reading** a CV via the on-screen keyboard (only successful if RailCom feedback or other feedback) (if not automatic) **Touch on 1st input field - type in CV number - OK - (again) OK - message follows C key** (on the keyboard) to delete typed characters.

Messages about success (or failure) after entering and completing (OK) a line, after each programming or readout process - a new line with input fields appears automatically.

- ACK** - after programming a CV and confirmation by "RailCom" (or other feedback)
- READ** - after reading a CV value using "RailCom" (or other feedback)
- SENT** - after programming a CV if there is no feedback option or no feedback is received
- NACK** - after programming attempt, if no confirmation (even though feedback system is available)
- N-RD** (= NO-READ) after unsuccessful read attempt (usually because no feedback)

If a correction is to be made in a line: Either create a new line with the same CV number or return to the old line (1st input field) by touching it.

The completed CV programming and readout processes are **listed below each other**, which serves as an overview and can also be reused later.

NOTE: CV programming via the ZIMO app will **be greatly expanded in the future**, providing even more convenient features. The same applies to other areas of the app, however...

Some of the planned enhancements (to be incorporated free of charge via software updates): Inclusion of CV pages (e.g. for assigning speaker outputs and filter handling in ZIMO decoders), special graphical solutions for specific decoder topics (Swiss mapping, effects, etc.), load code management for ZIMO sound projects, CV set management, etc.

Using the selection table, the "Address and Consist" topic can be opened with the THEMES & SETS button; the two input fields corresponding to the topic appear for selection:

Touching one of the fields automatically reads the current value first; either the vehicle address or the consist address; this is done for testing purposes, as it is already known.

Then, by filling in the input field, a new address can be entered into the decoder, whereby the address of the "new old" vehicle is automatically corrected in APP DB (V) and SYS DB (V); the newly addressed locomotive retains all attributes (names, function symbols, etc.).

ATTENTION: Addressing procedure currently (February 2026) only functional when re-addressing from a "large" address (128 or higher) to another large address (also > 128). Attempts to re-address between large and small addresses or vice versa may render the decoder inoperable (can only be re-addressed in service mode).



ZIMO Elektronik GmbH
Schönbrunner Str. 188 A - 1120 Vienna