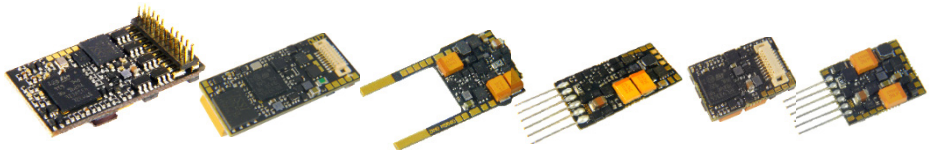
Newsletter August 2022

ZIMO ELEKTRONIK GmbH | Schönbrunner Straße 188, 1120 Wien | Austria | office@zimo.at | www.facebook.com/zimo.elektronik | www.zimo.at

PREFACE: This newsletter is "a little different" from other issues (past and probably future ones). It is NOT about the latest news on products such as the MX33, StEin, etc. - that will come back shortly. This newsletter has only one topic, the ZIMO decoders. Here, too, it is not so much about the current characteristics, but more about the fundamentals and future prospects.

Of course, we are aware of the frequently voiced criticism of our "early announcement policy", but on the other hand: the products are built for a long service life and should remain "state-of-the-art" for as long as possible. We therefore offer software updates, which are more than just bug fixes: Anyone who buys a ZIMO decoder today is also acquiring its conversion into a decoder of the year 2025 (for example). This applies to all decoders, whether it is actually purchased as such or is supplied with a vehicle as an OEM product.

In ZIMO decoders there ***is more than meets the eye***



Images: small selection of ZIMO MS decoders

ZIMO decoders have an excellent motor regulation (maybe the best in the world...),

ZIMO decoders have excellent sound properties and sound projects just like it (maybe the best in the world...),

ZIMO decoders also have many other characteristics, which are appreciated by model railroad enthusiasts.

So far so good..., BUT:

ZIMO is not only concerned with maintaining its current product technology, but also takes future developments into account during the design stage, for example when equipping them with processor and memory capacities that have sufficient reserves or the software structure.

The real railroad, the "role model", is changing, and the model railroad shouldn't remain stuck in the past indefinitely. Of course, old engines (especially steam engines...) are nicer than modern electrics, and we should be able to (but not forced to) control them in the "old way".
But modern control methods must have their place in model railroad technology and should be up to date. This may currently not apply to every single layout, but certainly to an ever-increasing number of layouts. Some characteristic terms play a role in modern train protection technology: "cab signaling", "high-performance blocks", "ETCS (European Train Control System)", "Movement Authorities"…

ZIMO decoders (as well as DCC command stations, cabs, StEin modules...) are prepared to bring the modern railroad world to the layouts via future software updates. How this is supposed to work is described to some extent in a recently granted ZIMO patent (DE 10 2014 002 297 B4).

Some of the new features implemented in ZIMO decoders today, such as the gyro functions for large scale trains (measurement of inclines, declines, curves), are on the one hand useful additions to current operation (e.g. sound optimization), but on the other hand are also part of a larger overall concept.

The foregoing relates in part to the future, however important new features such as "Track-on Search" and "GUI takeover" will soon be incorporated into production, see chapter "DCC has a dark spot..."

The ZIMO newsletters of the past two years were adorned here with a corona virus icon. Many events have been canceled during this time, but ZIMO and the entire industry came through it well.

The current multiple crisis - lack of material, lack of energy, lack of personnel (and above all a potentially expanding war nearby) - is probably even more dangerous than Corona.

The problem of the hour for the model railroad industry is the lack of parts, and in many cases it is already very bad. The market (on the sales side) is reasonably intact (but for how much longer..?).

ZIMO Elektronik GmbH is quite well positioned for the current situation: the complete in-house electronics production, instead of the usual outsourcing to "contract assemblers", is now paying off more than ever (although unplanned...).

We can therefore bypass the often very cumbersome management of external service providers during hardware design changes, which is important because we have to redesign circuits and circuit boards at short notice if certain components are unavailable in the long term.
ZIMO is generally (more so than others...) a very open company: Personnel status visible on the website, unrestricted telephone service, customer visits (open house days and otherwise)...
With that in mind, the problems ZIMO faces should not be denied, which are often noticeable in delayed deliveries and repairs, e.g.:

The ZIMO "leading-edge strategy" has a downside: the required components are in great demand all over the world.

The assembly often takes place "in stages" (due to late parts deliveries): subsequent repairs are unavoidable.

The mentioned hardware redesigns reduce personnel resources needed for the development of new products.

Component stock had to be tripled, which means three times the financing and storage costs.

Due to a lack of qualified employees, 24/7 production is currently not possible.

But in "times like these", things could be a lot worse...

A ZIMO decoder always ***remains a ZIMO decoder***

ZIMO, between 2010 and 2022, has become one of the leading decoder suppliers to the model railroad industry.

An important comment on this:

ZIMO decoders that are supplied to loco manufacturers as part of OEM business and are installed at the factory, are NOT special customer order versions, but **standard ZIMO decoders** that can also be bought from ZIMO dealers.

This applies to the hardware, as long as it is not an individual locomotive board with integrated decoder. This applies to the software (firmware), but any sets of CVs may be activated upon delivery for certain models.

It is often the case though that new or special features are integrated into the software for manufacturers, which then also flow into regular production, but NONE of the ZIMO properties, especially the "specialties", are left out:

The ZIMO "specialties" are ALWAYS on board,
a bit more on large-scale decoders than small-scale.

HLU *= Halt-Low speed-Ultra low speed*
is a fixed component of all ZIMO digital systems and decoders. In addition to the normal DCC commands, HLU information is placed on isolated track sections and is executed by all decoders: five-stage speed limits or stop before the red signal.

OW *= Ost-West (East-West)*
enables driving in the desired layout-related direction (known as east or west, technically the polarity position of the DCC rail signal) or provides information about the re-railing direction of a vehicle.

SEARCH *for addresses*
individual unknown addresses ("On Track Search") or in the course of a general registration ("Stock Search" or Automatic Registration).

i.a. *including the features currently under development.*

Of course, every ZIMO decoder can be updated. It's not just about error corrections (such - unfortunately - too), but ZIMO decoders are getting better and better during their entire life cycle.

ZIMO *builds* ***"sound decoders"*** *and* ***"non-sound decoders".***
Terms such as "driving decoder" or "locomotive decoder" should NO LONGER be used and "sound modules" - separate from the decoder - have long been obsolete.

The old designations not only are the cause of confusion but also point to an outdated product strategy, whereby decoders without sound would form the basis and the sound decoders would be the "specialty".

ZIMO does NOT build "cheap & easy" driving decoders (with low processor performance, etc.), as is often the case today. Especially with regards to advanced decoder control methods (see previous page), "non-sound decoders" too require full resources.

The new generation ZIMO ***sound decoders*** *were created as the basis of the product range; the* ***non-sound decoders*** *are "sound decoders without sound", the main difference is that sound generating components are omitted.*

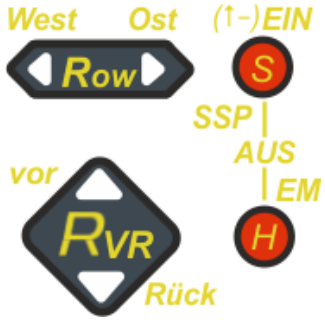
The new generations of ZIMO decoders are all **from a single source**. They are divided into 4 groups (each with its own initial letters in the product code):

Memory help: M = Motor, S = Sound, N = No-Sound, F = Function

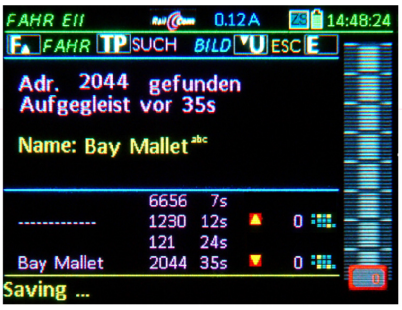
MS-Decoder: Sound-Decoder from MS440 … MS990 (in production)
MN-Decoder: Non-Sound-Decoder (successive introduction from 2022/4)
FS-Decoder: Function-Sound-Decoder derived from the MS-Decoder
FN-Decoder: Function-Decoder derived from the MN-Decoder

In addition, all MS and MN decoders can be reconfigured to behave like "real" FS and FN function decoders in the future.

The ***HLU*** *speed limits are set by the* ***StEin*** *(stationary) modules, each with 8 track exits, each track section under the control of a CTC program, in order to operate routes, block sections, hidden yards, etc.*



The ZIMO direction control: ***VR*** *and* ***OW*** *that is* ***V****(Forward)-****R****(Reverse) and* ***O****(East)-****W****est, is implemented in the MX33 controller using its own buttons. The multi-colored display LEDs (RGB) integrated in these buttons also provides information at any time about the direction status or the re-railing direction of the vehicle.*



The On-track ***SEARCH*** *is used to find the unknown address of one or more vehicles. The vehicle to be searched for is briefly de-energized (by "tilting"; hence "tilting search"); its address and (if already available) the name appear in the display after a few seconds.*

The Stock ***SEARCH****: Matching the addresses in the database with the decoders reporting from the layout;* ***see next page.***

The MS large scale decoders are equipped with a gyro sensor; other decoder types will follow. This can be used in the future to influence the sound, e.g.

DCC has a "dark spot"
but not much longer with ZIMO

The "GUI" *) of a new engine automatically appearing in the cab display was a system-function in the "mfx" world right from the start.

In the current DCC technology - probably the only disadvantage compared to "mfx" - it is relatively cumbersome to get the GUI of a vehicle on the cab: it is loaded from different sources or created "manually", but it does NOT come from vehicle (decoder) itself, where it should actually come from.

*) The "GUI" is a Graphical User Interface and consists of the engine's name, often with an added image, a compilation of symbols or control icons (i.e. function icons) etc., which makes it easy to control the locomotive (the train) with the cab.

The **transmission of the GUI** from the vehicle to a DCC system (or its operating devices) is currently the **subject of development and standardization efforts** in the DCC world. The European association of digital manufacturers, the "RailCommunity", is working on the so-called "automatic registration" and proposed the standard RCN-218, which deals with both finding new addresses and reading out the GUI from the decoder.

The new **ZIMO** MS sound and MN non-sound decoders (and some of the phased-out MX decoders) support **three** different **registration and GUI transmission methods** (some already today, some after completion of the ongoing development work):

- as an **mfx-capable** decoder (together with a Märklin mfx digital system): Märklin "mfx",
- in the capacity as a **DCC compliant** decoder (together with a DCC compliant digital system) the "automatic registration" according to the "RailCommunity" RCN-218 or its NMRA counterpart,
- in the **ZIMO environment** (i.e. together with a ZIMO or a DCC compatible system): a procedure that uses the means of the RCN-218 for the address registration (finding new addresses), but goes its own way with the GUI transmission: the **"ZIMO File Transmission".**

ZIMO File Transmission: the characteristic PROPERTIES

See next page for ***technical explanations***

- COMPATIBILITY with the traditional address orientated DCC technology, which is in contrast to the "session" reference of mfx or RCN-218. A particular focus is the mixed use of "old" decoders and "new" ones in a system (whereby only the latter can handle address registration and GUI transmission).
- FREEDOM with regard to different operating concepts through unlimited type and number of vehicle images, function symbols or control elements, without limitation to standardized images. Different operating devices receive the appropriate GUI data from ZIMO decoders or their loaded sound projects.

  The prerequisite for this is that the data for all GUIs has been saved beforehand - usually as part of a (sound) project in the (sound) memory of the decoder, where space is available for several GUI files.
- INTEGRATION of the address registration as well as the GUI transmission in the general DCC and RailCom communication, driving and switching operations continue (almost) unhindered. With ZIMO GUI transmission it takes...
  - NO "register first, then drive"; also works (maybe a little slower) with "Start driving immediately, get GUI sometimes later",
  - NO engine stops to retrieve the GUI from the vehicle, the GUI can be retrieved while driving (slightly slower if there are contact difficulties...),
  - NO lost address messages (RailCom Channel-1), which are often important in the operation of active trains, due to GUI retrieval of the newly reported vehicles,
  - NO DELAYING of simultaneously requested GUI transmissions or other "data files", but efficient and timely distribution.

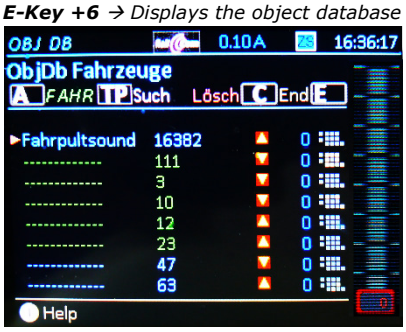
In the example below
*it is assumed that a **new vehicle** (the only one for the sake of clarity) has been placed on the layout. The **ZIMO sound decoder** in the vehicle contains a **sound project** with an **MX32 GUI file**.*

*First, the new vehicle must be "found", i.e. its address must be determined. This could also be done using a ZIMO On-track Search; but here the method of **stock search** is presented, which in principle works according to RailCommunity RCN-218 "Automatic registration" as far as the part for finding new addresses is concerned.*

Note: if the address of the new vehicle were known in advance, searching/finding could be omitted; then go to menu, then "GUI from decoder...)

Start the stock search on the MX32/MX33 controller from any operating status:

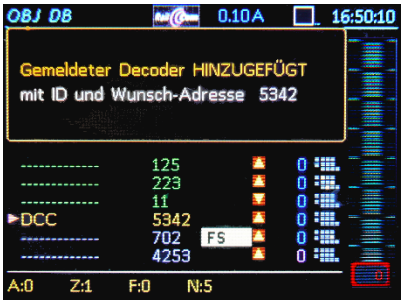
***E-Key +6** → Displays the object database*



***TP-Key** → Starts the stock search*

Since there is only one new address present in this example, it is found and marked quickly
(Address flashing until confirmed).

***A-Key** → Confirms the new address*



***A- Key** → Activates the new address*

The vehicle can now be driven WITHOUT delay; initially there are no GUI elements (but F0, F1, ...).

ZIMO File Transmission: the TECHNICAL EXPLENATIONS

As already mentioned: in contrast to the address registration, data spaces are NOT read out according to RCN-218 for the "GUI" transmission in the case of **"Z2Z"** ("ZIMO to ZIMO"), but rather the ZIMO proprietary "File Transmission" is used, because this better accommodates the future design plans of the "DCC-ZIMO" protocol.

"ZIMO File Transmission" means (to put it very simply): a data block of up to 1000 bytes is divided into numerous small pieces (2 to 3 bytes each), which are transmitted as a sequence of RailCom datagrams, each following the "normal" DCC packets (with any content, only the address must match), from the decoder to the DCC command station. The systems own DCC retrieval commands can therefore be omitted.

The ZIMO File Transmission is **not just about the GUI**, this is just one application among several: the general purpose is the transmission of data blocks from the decoder to the DCC command station, such as:

- **Texts**, i.e. information and alarm messages that the decoder wants to send to the system (e.g. a steamer: "pre-heat for 10 hours, then ready to drive")
- **Train data** (readable via train bus): type, weight, sound of each car, etc.,
- **Route profile**: Sensor data on gradients/slopes, curves, etc., **and others...**

Summary of the most important steps in file transmission:

- **Request** for the appropriate GUI by the operating cab. For example, an MX32 requires the vehicle's GUI for display on the MX32, the ZIMO app for display on a smartphone and the Roco app for display on a tablet. This request is first sent to the command station (via the system bus).
- **DCC-Request-Packet** from the command station to the relevant decoder.
- **RailCom-File-Start-Datagram** from the decoder to the command station. With this, the decoder reports that the following DCC packets it received via RailCom datagrams, each containing a few bytes of the file, will be answered.
- **DCC-Confirmation-Packet** to the decoder. The command station reports back that it is ready to receive datagrams from the decoder.
- **RailCom-File-Content-Datagrams** from decoder to command station (each with sequence numbers, content bytes and CRC redundancy check) as a response to ANY DCC packets received on its own address, until the entire file has been sent.
  This technical feature (waiver of special DCC request commands) is one of the main differences compared to the retrieval of data rooms according to RCN-218: the other control and reporting functions of DCC and RailCom are hardly restricted. With the ZIMO control concept in mind, this (along with other properties) weighs more heavily than the mathematically lower efficiency of the process.
- These datagrams are NOT answered individually by the command center.
- **File-End-Datagram** After the entire content of the file has been sent, a CRC redundancy check (over the entire file) is carried out as a check.
- **DCC-Claim-Packets** request for garbled or missing data content to be sent again- these datagrams are then retransmitted.

Note: *The process shown in the image sequence to the right, is not always carried out in this detail, i.e. not all keys have to be pressed every time. There are automated requests available that can be activated as needed when adding a new address or for starting the GUI transmission.*

Outlook:

The "ZIMO File Transmission" described here is part of a comprehensive concept for optimizing data traffic between the DCC system and decoders.

The DCC protocol now in use (without ZIMO files...) clearly dates back to the last century. It was created based on the technological conditions of the time, i.e. adapted to decoders with slow microcontrollers and small program memory. As a result, there are still measures in place such as sending the bits twice (simple but not very efficient data backup), the refresh cycle (all driving and function data are constantly repeated), or limiting the packet lengths.

In the future, data blocks (= larger data structures than the usual DCC commands) will flow in both directions, i.e. from the stationary devices such as DCC command station and track section modules - e.g. StEin - to the decoders, and from the decoders to the stationary devices.

This will require efficient transmission methods, and moreover "classic" and modern decoders will have to share the rail as a transmission medium.

But this is already part of ZIMO's tasks today and the years to come...

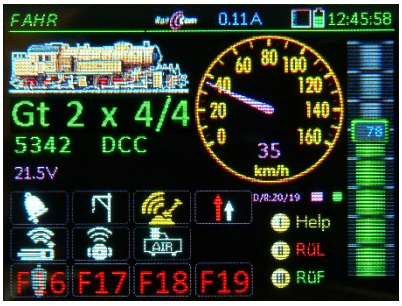
Example continued (previous page)

M-Key *→ Menu, scroll to "Load ZIMO GUI from decoder"*

A-Key *→ The GUI transmission is in progress and VISIBLE to the user: the arrival of the various GUI elements (locomotive image, name, speedometer, function icons...) can be seen (can also be canceled if desired). Any transmission errors can also be identified: symbols that remain empty and subsequent automatic delivery.*

After approx. 0.2 seconds (with no interference) the loco image, the selected speedometer and the name have already arrived; Function icons are still missing.

The first function icons...



...more, for the second functions page.

The GUI-transmission is complete, display returns to the first page of function icons.

Driving operations continue unchanged; GUI transmission would also have been completed if address was deactivated.