

Zimo-Scripte

Lokdecoder programmieren

Die aktuellen Lokdecoder aus dem Hause Zimo bieten dem anspruchsvollen Modelleisenbahner eigentlich schon fast alles, was man sich wünschen kann. Ambitionierte Digitalanwender haben aber eigentlich immer zumindest noch einen Wunsch mehr. Und denen bietet Zimo über sogenannte Scripte noch die Möglichkeit, zusätzlich individuelle Abläufe im Decoder zu programmieren.

Diese Funktion ist vermutlich nicht jedem Zimo-Anwender bekannt und setzt natürlich auch voraus, dass man das Programmiergerät MXULFA besitzt. Denn nur damit kann man diese Scripte in den Decoder laden.

Welche Decoder?

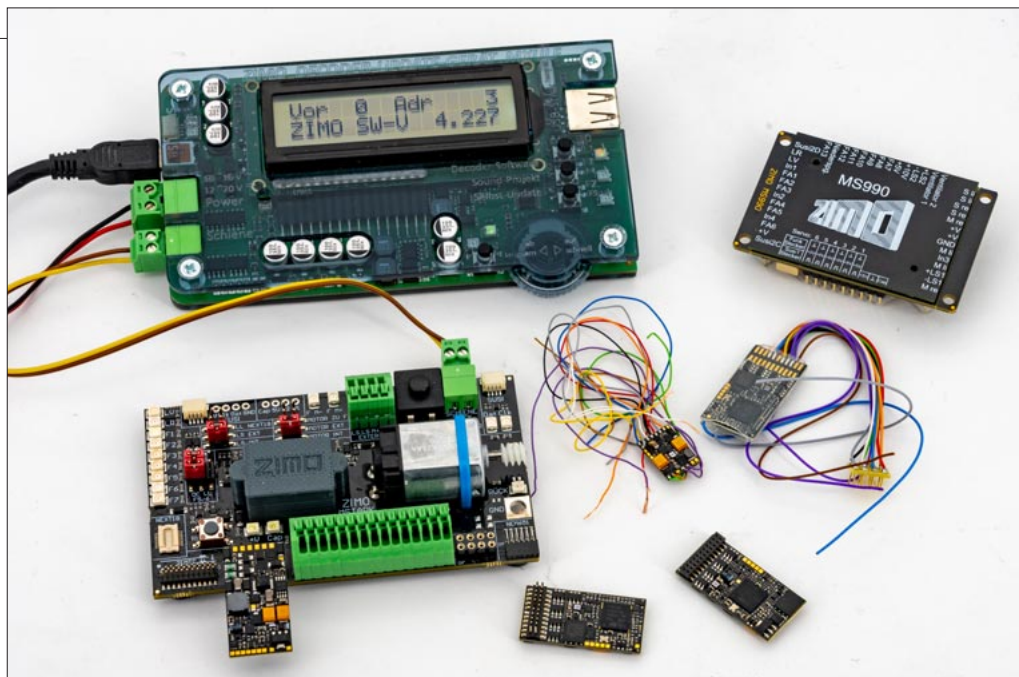
Unterstützt werden die aktuellen Sounddecoder der MX-Serie (8-Bit-Sound) und der MS-Serie (16-Bit-Sound). Allerdings sollte man darauf achten, dass man immer die aktuelle Firmware auf die Decoder lädt, sonst funktionieren eventuell nicht alle Möglichkeiten.

Um die Scripte zu erstellen und zu testen, benötigt man noch das Programm ZSP (Zimo-Sound-Programmer). Die Windows-Software kann man kostenlos von der Zimo-Webseite herunterladen und im Zusammenspiel mit dem MXULFA nutzen. Auch hier sollte man darauf achten, immer die neueste Version zu verwenden.

Wozu sind Scripte brauchbar?

Für die meisten Anwender bieten Zimos Sound-Decoder mehr, als sie vermutlich jemals nutzen können. Dennoch hat jeder Anwender andere Wünsche und die will man möglichst gut erfüllen. Dazu zählen auch komplexere Funktionsabläufe wie z.B. ein Kupplungswalzer. Über die Script-Programmierung kann jeder Anwender auch seine eigenen Wünsche programmieren. Im Prinzip ist alles denkbar, was der Decoder kann – und das in beliebigen Kombinationen und Abhängigkeiten.

Damit sind komplexere Sound-Abläufe machbar, die durch eine Funktionstaste ausgelöst werden können, aber auch durch externe Ereignisse, wie z.B. das Auslösen eines Reed-Kontaktes unter der Lok



Zimo-Scripte

Lokdecoder individuell programmieren

Eine Besonderheit der Sounddecoder von Zimo sind die erweiterten Programmiermöglichkeiten, mit denen sich die unterschiedlichsten Abläufe automatisieren lassen.

Programmierscripte unterstützen alle aktuellen Sounddecoder von Zimo, unabhängig von ihrer Größe.

durch einen Permanent-Magneten im Gleis. Dafür gibt es bei den neueren Zimo-Decodern sogenannte Schalteingänge. Je nach Typ und Ausführung können das bis zu vier Eingänge sein. Mit dieser Methode könnte man etwa eine kleine Pendelautomatik inklusive diverser Soundabläufe und anderer Funktionen programmieren. Alle diese Scripte laufen eigenständig im Decoder ab. Eine externe Steuerung jeglicher Art wird nicht benötigt.

Welchen Leistungsumfang bieten die Scripte?

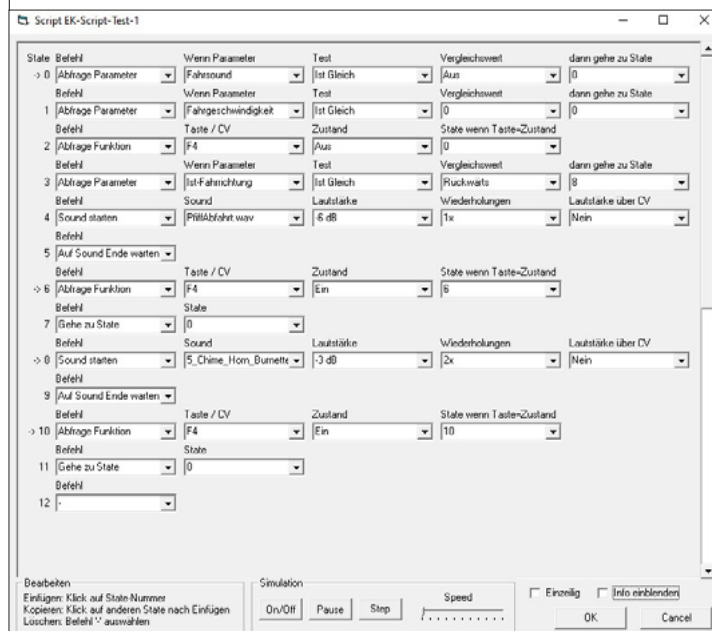
Da die Scripte als zusätzliche, eigenständige Programme im Decoder ablaufen, nutzen sie auch dessen Prozessor-Leistung und Speicherkapazitäten. Die älteren MX-8-Bit-Decoder können acht Scripte speichern und parallel ablaufen lassen. Da die neuen MS-16-Bit-Decoder über mehr Speicher und schnellere Prozessoren verfügen, gehen dort sogar bis zu 16 Scripte. Auch die Anzahl der unabhängig abspielbaren Soundkanäle und Sound-Samples liegen beim MS-Decoder höher. Dafür erlauben beide Decoder-Familien, bis zu 255 Befehle

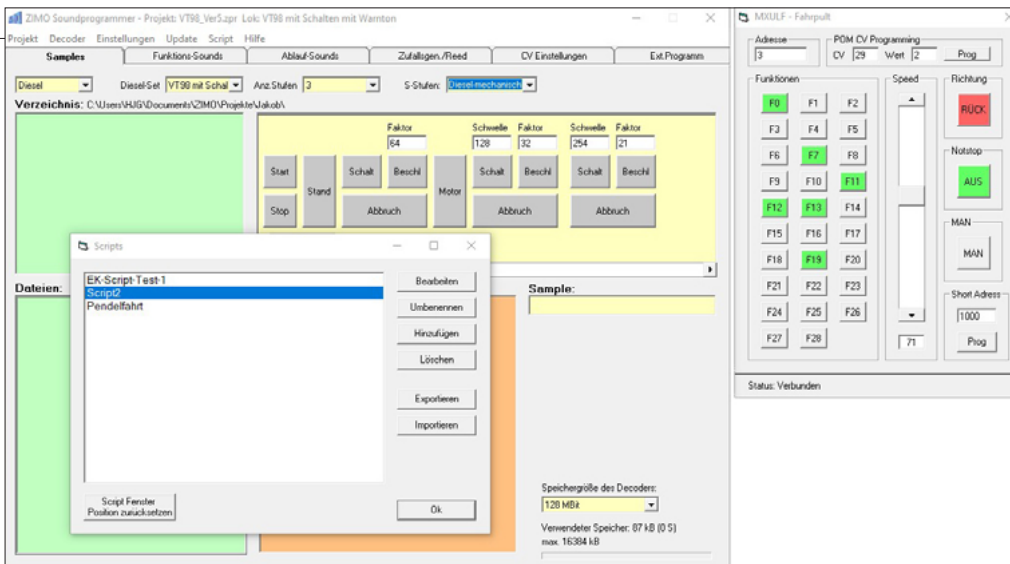
pro Script programmieren zu können. Damit dürfte sich fast jegliche erdenkbare Aufgabe programmieren lassen.

Wie werden Scripte programmiert?

Man muss zwar keine Programmiersprache beherrschen, es hilft

So sieht es aus, wenn man ein komplexeres Script im ZSP-Programm von Zimo erstellt.





aber dem allgemeinen Verständnis sicherlich ungewohnt, wenn man schon mal ein kleines Programm in irgendeiner Programmiersprache erfolgreich geschrieben hat. Das eigentliche „Programmieren“ geschieht dann über die grafische Oberfläche des Programms ZSP. Hier muss man keine komplizierten Befehle kennen und eingeben, vielmehr wählt man alle gewünschten Optionen per Mausklick aus und reiht die Befehlssequenz logisch hintereinander auf.

Aufpassen muss man nur, wenn man Abhängigkeiten programmiert, also „mache etwas Spezielles, wenn eine bestimmte Voraussetzung gegeben ist“, z.B. „läute die Glocke, wenn F18 gedrückt wird und die Lok gerade rückwärts fährt“. Dabei sind Erweiterungen, Abhängigkeiten und Verschachtelungen jeglicher Art denkbar. Solche Abläufe können sehr schnell komplex werden. Da die Decoder hier einfach nur stur die einzelnen Schritte abarbeiten, gibt es auch keine Fehlermeldungen im gewohnten Sinne. Entweder etwas funktioniert wie erwartet oder es passiert nichts oder das Falsche.

Zimo hat deshalb im ZSP einen Script-Simulator eingebaut. Hier kann man in einem gewissen Umfang ein Script ganz langsam Schritt für Schritt ablaufen lassen, um zu sehen, was danach jeweils passiert. Das ist gewissermaßen „Learning by Doing“. Hier kann man nichts kaputt machen und lernt schnell, was der Decoder kann und wie man diese Möglichkeiten für sich nutzen kann. Um das Studium der Decoder-Anleitungen und der

Script-Dokumentation kommt man allerdings nicht herum, will man genau verstehen, wie das alles funktioniert. Sobald man das Script fertig programmiert hat, speichert man es als integrierten Bestandteil des Sound-Projekts ab.

Wie kommt ein Script in den Decoder?

Spätestens hier kommt Zimos Programmiergerät MXULFA ins Spiel. Über eine USB-Verbindung wird es von ZSP angesteuert und das komplette Projekt inklusive der Scripts in den Decoder geladen. Das ganze geht ab hier auch mit dem (kostenlosen) ZCS-Programm, welches zur Einstellung aller CVs genutzt wird. Obendrein kann man solch ein Projekt auch auf einem USB-Memory-Stick abspeichern. Wenn

Über das ZSP-Programm von Zimo lassen sich beliebig viele Scripte programmieren und auch kopieren. Hier kann man auch die hinterlegten Sounds Probehören und über den virtuellen Fahrregler alles ausprobieren.

Im Simulator-Modus von ZSP kann man jedes Script Schritt für Schritt ausführen lassen, um sicherzustellen, dass alles genau so funktioniert, wie erwartet, oder um Programmier-Fehler schnell erkennen zu können.

man den in ein MXULFA einsteckt, kann der Programmierer das alles autonom, ganz ohne PC, in den Decoder schreiben.

Sobald die Lok auf dem Gleis steht und der Decoder läuft, sind auch sofort alle seine Scripte aktiv. Natürlich kann man via MXULFA jederzeit ein Script im Projekt auch wieder auslesen, löschen, modifizieren, kopieren oder ersetzen. Ausnahme sind die sogenannten ZPP-Dateien, das sind fertige Soundprojekte, deren Daten nicht rückwärts ausgelesen werden können.

Fazit

Scripte sind in der Welt der Zimo-Sounddecoder ein mächtiges Werkzeug, um eine Lokomotive nach allen Regeln der Kunst zum Leben zu erwecken. Das dürfte sicherlich nicht jedermanns Sache sein, weil es doch ein gewisses Verständnis für Programmabläufe und Decoder-Funktionen voraussetzt. Wer das aber kann und mag, hat hier ein sehr elegantes und robustes Werkzeug an der Hand, um auch noch das allerletzte an Möglichkeiten aus seinem Zimo-Sounddecoder heraus zu holen. Und das alles ohne Mehrkosten, nur das Programmiergerät ist Voraussetzung. Aber das braucht man eh, wenn man die Decoder-Firmware updaten und Sounds aller Art in den Decoder laden will.

HANS-JÜRGEN GÖTZ

