

WeichZwei-Dekoder: Weichen-Ansteuerung (Firmware "weichzwei")

Diese Seite beschreibt eine Firmware-Variante für den [WeichZwei-Dekoder](#). Um in den Genuss des hier beschriebenen Dekoders zu kommen, müssen Sie

- 1) Die [hier](#) beschriebene Hardware bauen
 - 2) Die auf dieser Seite erhältliche Firmware in den PIC-Prozessor laden
- Diese Firmware kann auch mit der Hardware [ZweiPol](#) verwendet werden (siehe auch [HIER](#))

[Anschluss](#) / [Download](#) / [Adressen lernen](#) / [Time Out Funktion](#) / [Abschaltzeit ändern](#)

Hier die Eigenschaften des WeichZwei-Dekoders beim Einsatz als Weichendekoder:

- Für Motorola (Märklin) oder DCC
- geeignet für 2-Spulen Antriebe (diese haben 3 Anschluss-Kabel)

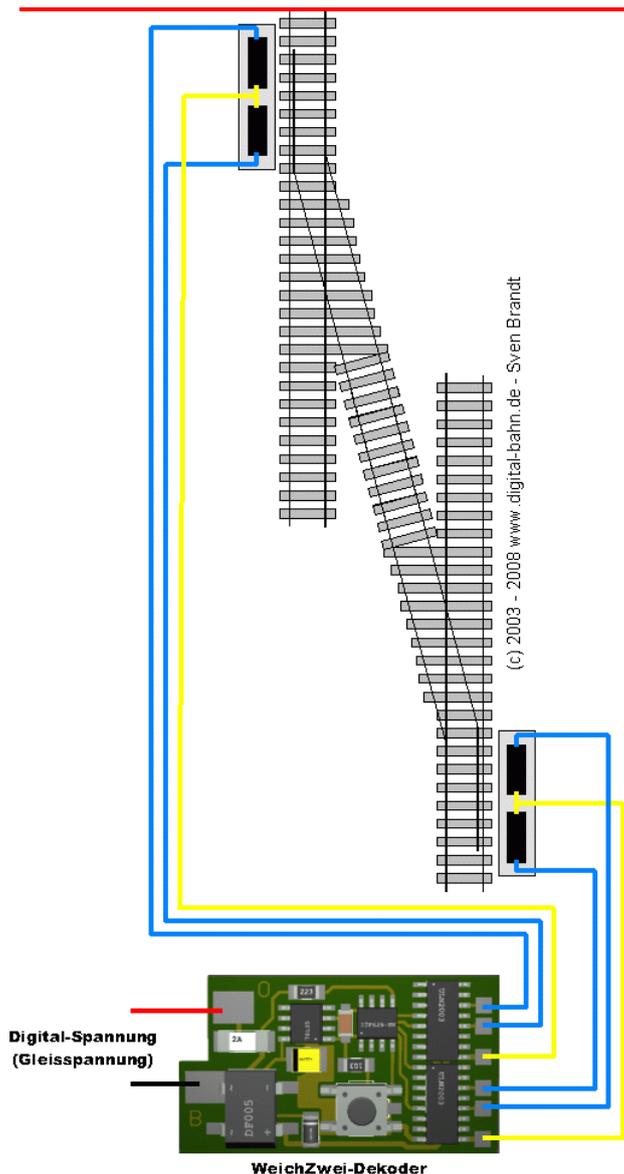
siehe auch [geeignete Weichenantriebe - eine Übersicht](#)

- Digitale Ansteuerung von 2 Weichen-Antrieben
- Die Adressen können beliebig vergeben werden (muss also nicht paarweise sein!)
- Address Learning Funktion: Nach dem Druck auf die Taste gelangt der Dekoder in den "Lern-Modus".
- Keine DIP-Schalter notwendig, also auch keine Adress-Tabellen
- Polung der Eingangssignale beliebig, keine Verpolungsmöglichkeit
- Durchbrennen des Weichenantriebes wird durch integriertes Time-Out auch bei defekter Endabschaltung und unlesbaren Abschalt-Befehlen verhindert

Einige Weichenantriebe benötigen polarisierte Ansteuerspannung. Die geeignete Hardware hierfür siehe [ZweiPol](#)

[nach Oben](#)

Anschluss



[nach Oben](#)

Download

Motorola Firmware Download

Protokoll	Motorola
Name	weichzwei_351_mm.hex
Version	V3.51 vom 28.06.2011
Prozessor	PIC 12F629
ID-Value	Cxxx
Shop Bestell-Nummer	pic629-00c-m
vordefinierte Adressen	1-grün / rot (Weiche#1 links / rechts) 2-grün / rot (Weiche#2 links / rechts)
Änderungen	V2.07 Verwendung von OSCCAL V2.11 keine funktionelle Änderung V2.12 Problem mit Sonderoption 901 der Intellibox behoben V2.20 Abschaltzeit im EEPROM änderbar V3.00 interne Neuorganisation V3.02 Code ist jetzt kompatibel zum 12F675 V3.09 Speicheroptimierung V3.11 weniger Verlustleistung im Zweipol Dekoder V3.12 Anpassung an DDW-Timing V3.18 Optimierung für ZweiPol V3.23 Optimierungspaket 2 für ZweiPol V3.26 ID für Manipulator hinzu V3.37 Korrektur 12F675 Kompatibilität V3.38 verbesserte Time-Out Funktion V3.51 Abschaltbefehl kann ignoriert werden, d.h. immer Abschalten nach TimeOut

DCC Firmware Download

Protokoll	DCC
Name	weichzwei_351_dcc.hex
Version	V3.51 vom 28.06.2011
Prozessor	PIC 12F629
ID-Value	Cxxx
Shop Bestell-Nummer	pic629-00c-d
vordefinierte Adressen	1 - grün / rot (Weiche#1 links / rechts) 2 - grün / rot (Weiche#2 links / rechts)
Änderungen	V2.08 erste Veröffentlichung V2.11 keine funktionelle Änderung V2.20 Abschaltzeit im EEPROM änderbar DCC Abschaltbefehl wird korrekt verarbeitet V3.00 interne Neuorganisation V3.02 Code ist jetzt kompatibel zum 12F675 V3.09 Speicheroptimierung V3.11 weniger Verlustleistung im Zweipol Dekoder V3.18 Kompatibilität zu RailCom Zentralen (NOP Befehl) V3.18 Optimierung für ZweiPol V3.23 Optimierungspaket 2 für ZweiPol V3.26 ID für Manipulator hinzu V3.37 Korrektur 12F675 Kompatibilität V3.38 verbesserte Time-Out Funktion V3.51 Abschaltbefehl kann ignoriert werden, d.h. immer Abschalten nach TimeOut

[nach Oben](#)

Adressen lernen (Address learning function)

Durch den Druck auf den Taster gelangt der Dekoder in den "Lern-Modus". Jetzt fängt die erste Weiche an, hin- und her zu schalten (neudeutsch: Die Weiche toggelt). Nun wird ein Weichen-Befehl erwartet. Durch Druck auf eine Keyboard-Taste (oder einen entsprechenden Befehl über PC und Interface) wird dieser Befehl an das Gleis gelegt. Dadurch wird dies zur Adresse der Weiche. Nun fängt die 2. Weiche an zu toggeln. Auch hier wird wieder durch ein Weichen-Befehl dem Dekoder mitgeteilt, unter welcher Adresse künftig diese Weiche geschaltet werden soll.

Im Programmier-Mode kann mit dem Taster auch die aktuelle Lern-Phase übersprungen werden. Wer also nur die Adresse der 2. Weiche wechseln möchte, drückt den Taster (und kommt in die Lern-Phase für die erste Weiche). Durch einen erneuten Druck den Taster wird diese Phase übersprungen und die Weiche behält die alte Adresse. Nun toggelt die 2. Weiche und hier kann nun die neue Adresse eingegeben werden (oder aber durch Tastendruck ebenfalls beibehalten werden).

Tipp: Reagiert die Weiche "falsch herum", dann einfach die Adresse noch mal mit der anderen Taste des Keyboard (z.B. Rot statt Grün) programmieren!

Die Adressen können auch mit dem HEX-Manipulator konfiguriert werden:

[Hex Manipu - Manipulieren des HEX-Files](#)

[nach Oben](#)

Zur Time-Out Funktion

Diese Funktion ist sozusagen die 4. Versicherung, bevor es zu einem Schmelzen der Antriebs-Spule kommt: Zunächst gibt es die Endabschaltung der Weichen. Durch sie trennt sich die Spule i.d.R. nach dem Umschalten selbst von der Spannung. Vorher sollte die Spannung von außen natürlich noch nicht abgeschaltet werden, da die Weiche noch nicht in der Endlage ist.

Dann greifen als nächstes die Ausschalt-Befehle der Zentrale (es werden 2 Stück gesendet). Durch sie nimmt der Dekoder die Spannung an seinem Ausgang weg.

Und sollte dies nicht funktioniert haben, da z.B. gerade in diesem Moment ein halb entgleister Zug die Abschalt-Befehle verstümmelt, greift die Time-Out Funktion: Jede Spule wird nach ca. 500 ms (konfigurierbar) abgeschaltet, auch wenn der Abschalt-Befehl nicht empfangen wurde. M-Weichen besitzen die Endabschaltung nicht. Sie würden daher bei fehlenden Abschalt-Befehlen verglühen...

siehe auch Schaltzeiten

[nach Oben](#)

Abschaltzeit für Time-Out ändern

Die Abschaltzeit für die Time-Out Funktion kann konfiguriert werden. Dadurch ist es z.B. möglich, "langsame" Weichenantriebe wie Motorantriebe einzusetzen.

Konfiguration der Time-Out Abschaltzeit

EEPROM-Zelle	\$20
zulässiger Wertebereich	1 bis 127 (dez.)
001 (dez) = 01 (hex)	ca. 0.5 sec.
002 (dez) = 03 (hex)	ca. 1.0 sec.
003 (dez) = 03 (hex)	ca. 1.5 sec.
usw.	
127 (dez) = 7F (hex)	ca. 63.5 sec.

Vorgehen siehe [➡ Daten im EEPROM ändern](#)

Die Parameter können auch mit dem HEX-Manipulator konfiguriert werden:

[➡ Hex Manipu - Manipulieren des HEX-Files](#)

[nach Oben](#)