

ACHTUNG! Diese Seite wird nicht mehr länger gepflegt.
Wechselt auf die englischsprachige Seite für eine aktuelle
Anleitung (Link unten links).

Auslesen des S88 Busses vom Ethernet

Allgemein

Für die Funktionalität muss das [IO Projekt](#) (base oder pro) auf den Netzer [geladen](#) werden.

Der S88 ist ein Rückmeldebus für Modellbahnsysteme.

Bis zu 32 verteilte Teilnehmer hintereinander geschaltet bilden ein zusammenhängendes Register, in das zyklisch Werte geladen werden. Diese (binären) Werte kommen von Schaltern, Gleis-Rückmeldern u.ä. Dingen, die rund um die Anlage verteilt sind.

Nachdem alle diese Werte einmal in das Register geladen wurden, wird nun der Inhalt sukzessive mit einem Takt in Richtung Zentrale geschoben (Bit für Bit) und dort zwischengespeichert und weiterverarbeitet.

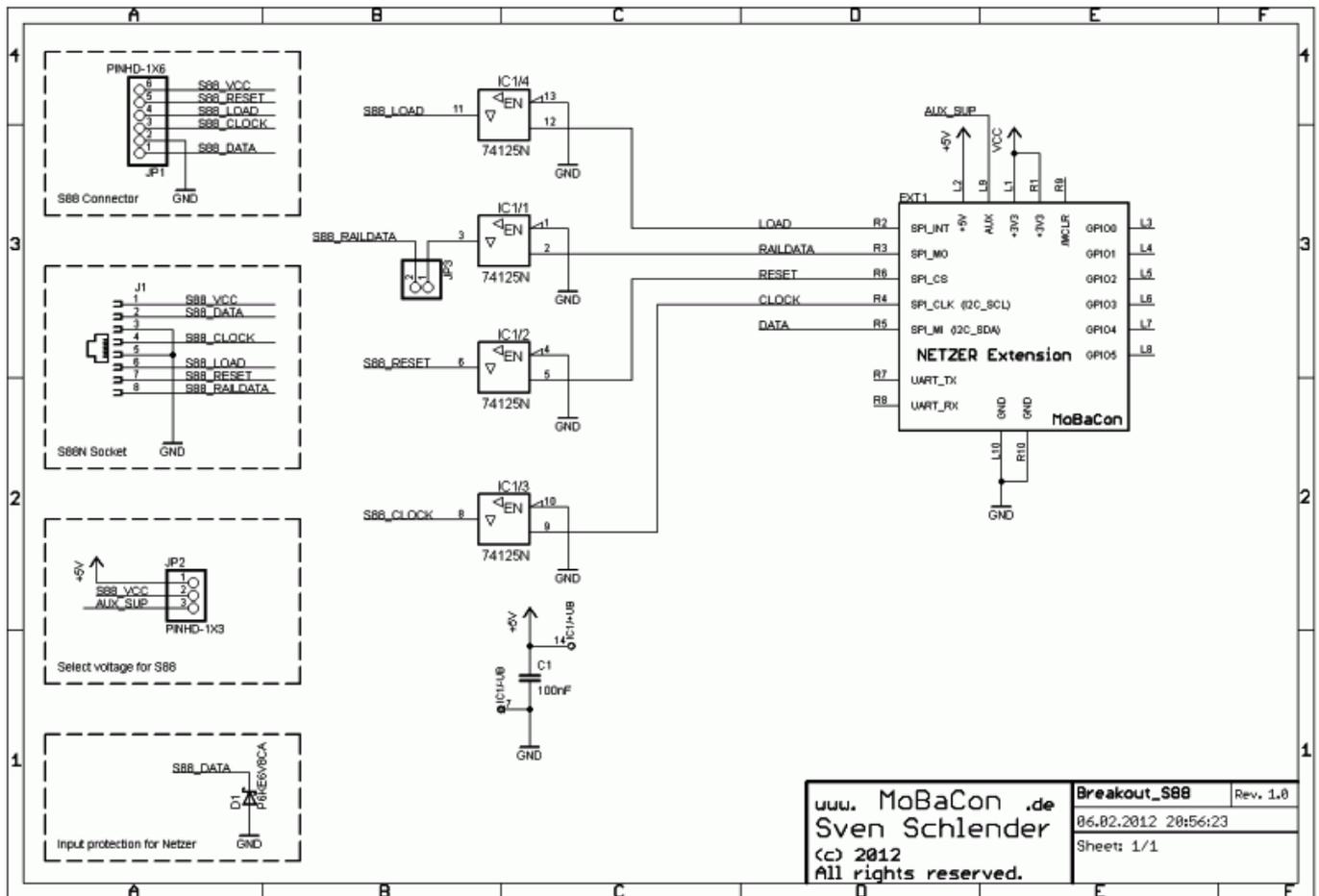
Der S88-Bus besteht insgesamt aus 3 Steuerleitungen, **CLOCK**, **LOAD** und **RESET** und einer Datenleitung, **DATA**.

Mehr Details zum S88-Bus findet sich bei [Sven Brandt](#) und bei [Der MoBa](#).

Netzer als Zentrale

Aufgrund der Flexibilität des Netzers ist es sehr einfach möglich einen S88-Strang mit dem Netzer auszulesen. Zur Anwendung kommt der [serielle Server](#) im SPI Master Modus. Neben dem Bus wird die Möglichkeit genutzt, dass IO-Pins direkt mit Escape-Sequenzen angesprochen werden können.

Hardware



Das Bild zeigt eine Schaltung für die Netzer [Spielfeld](#) S88-Erweiterungsplatine. Die Komponenten sind Standardkomponenten, die auch schnell auf einer Lochrasterplatine aufgebaut werden können.

Die Schaltung ist schnell erklärt. Da die Betriebsspannung des Netzers +3,3V beträgt, der S88-Bus jedoch im Normalfall mit +5V arbeitet, werden Bustreiber (**IC1**) zwischen die drei Steuerleitungen S88_RESET, S88_LOAD und S88_CLOCK geschaltet. Da die Tristate-Funktion der Bustreiber nicht benötigt wird, werden sie einfach immer eingeschaltet (EN=GND). C1 dient der Abblockung. +5V wird vom [Netzer Spielfeld](#) zur Verfügung gestellt.

J1 bietet eine Anschlussstelle nach dem herkömmlichen Verdrahtungsmuster für S88. JP1 ist eine RJ45-Buchse nach dem neuen Verdrahtungsstandard [S88N](#)

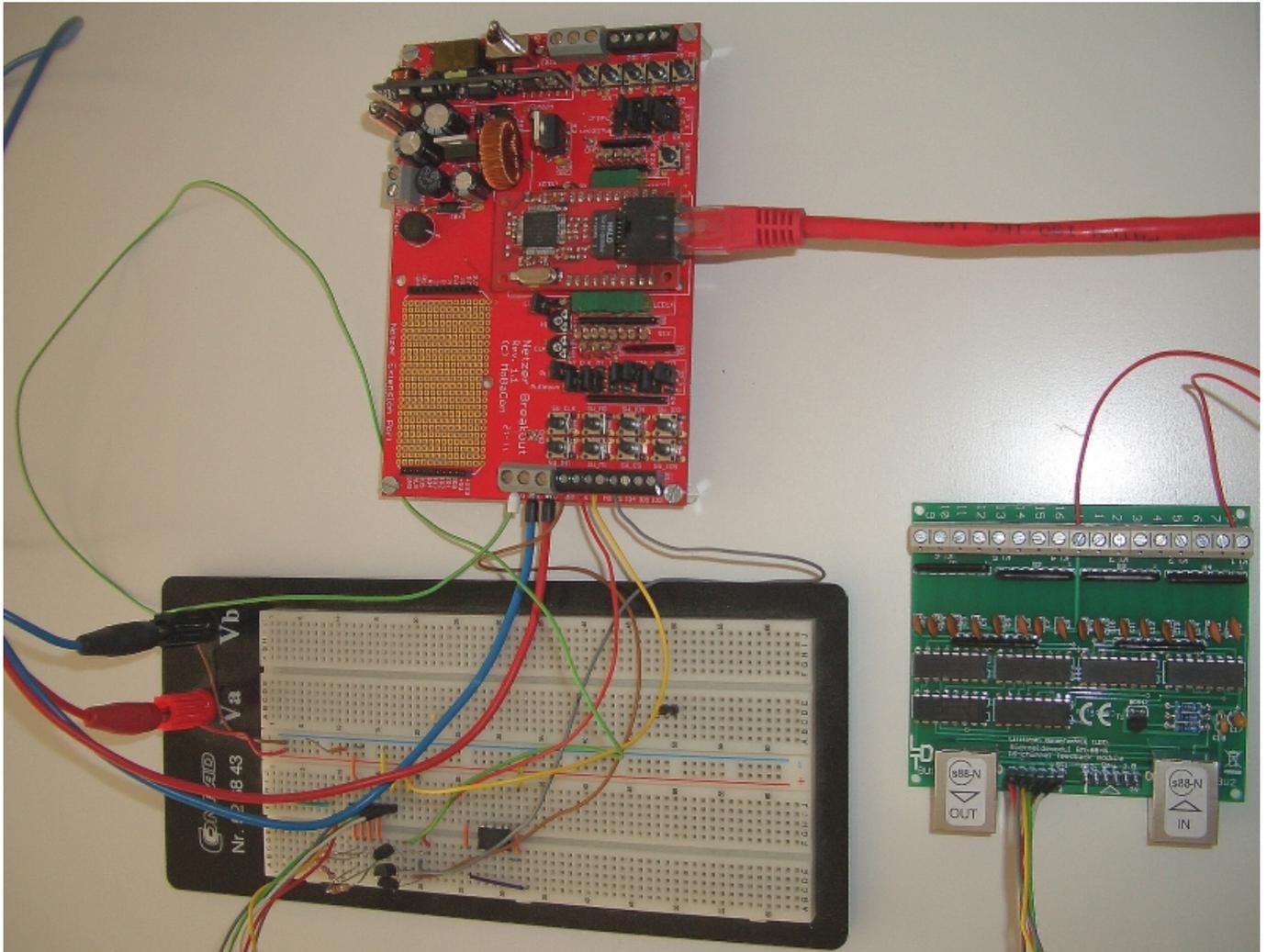
Nach dem S88-Verdrahtungsstandard S88N ist für die Versorgungsspannung alternativ zu +5V auch eine +12V zulässig, falls das die Module am Bus unterstützen. Für das Umschalten der Versorgungsspannung per Jumper dient **JP2**. VCC_AUX ist die primäre Versorgungsspannung vom Netzer Spielfeld und sollte entsprechend auf +12V geregelt werden.

Pin 8 des S88N-Standards ist für das Gleissignal gedacht. Testweise wird hier der SPI-Datenausgang noch über den Bustreiber an diesen Pin gelegt, um ggf. auch Steueraufgaben übernehmen zu können (falls JP3 gejumpert ist).

Die Datenleitung des Busses S88_DATA wird direkt an den Netzer angeschlossen, da der SPI-Dateneingang +5V-tolerant ist. Die Suppressor-Diode D1 dient dem zusätzlichen Schutz des Eingangs.

Für die weitere Dokumentation wird angenommen, dass

LOAD an SPI_INT und RESET an SPI_CS angeschlossen ist.



Einstellungen Netzer Webinterface

Der Netzer mit installierter **Pro**-Version (min. Release 1.5) muss zuvor über das Webinterface konfiguriert werden:

1. SPI-Master-Mode auf der Allgemein-Seite aktivieren (Danach Neustart nicht vergessen!)
2. SPI-Master-Einstellungen: Frequenz auf 5 kHz (bei langen Leitungen ggf. auf 1,3 kHz), SPI-Modus 2 (Ruhepegel des SPI-Taktes ist 1, Übernahme der Daten erfolgt bei fallender Flanke) Abtasten in der Mitte der Taktperiode (siehe Bild).

Allgemeine SPI Einstellungen (SPI Slave: Änderungen erst nach Neustart)

Taktmodus:

SPI Mode 0

SPI Mode 1

SPI Mode 2

SPI Mode 3

SPI Master Einstellungen

Taktfrequenz:

Abtastung des Eingangs: In der Mitte Am Ende

3. GPIO-Einstellungen: In diesem Beispiel wird SPI_INT sowie SPI_CS benötigt, demnach als digitale Ausgänge konfigurieren!

RX	h	Digitaler Eingang	<input type="radio"/>
SPI_CS	i	Digitaler Ausgang	<input checked="" type="radio"/>
SPI_INT	j	Digitaler Ausgang	<input checked="" type="radio"/>
SPI_CLK (I2C_SCL)	k	Wird vom seriellen Server verwendet.	<input checked="" type="radio"/>
SPI_MI (I2C_SDA)	l	Wird vom seriellen Server verwendet.	<input type="radio"/>
SPI_MO	m	Wird vom seriellen Server verwendet.	<input type="radio"/>

Speichern Zurücksetzen ID: Konfigurieren...

Herstellen der Verbindung

Für einen Verbindungstest unter Windows sollte ein Terminalprogramm verwendet werden, mit dem Bytes in Hexadezimal- oder Binärdarstellung versendet oder empfangen werden können wie z.B. das [Hammer-Terminal](#)). Die eigentliche [Netzwerkverbindung](#) wird mit **com2tcp** aufgebaut. Unter Linux kann auch ganz einfach **netcat** verwendet werden.

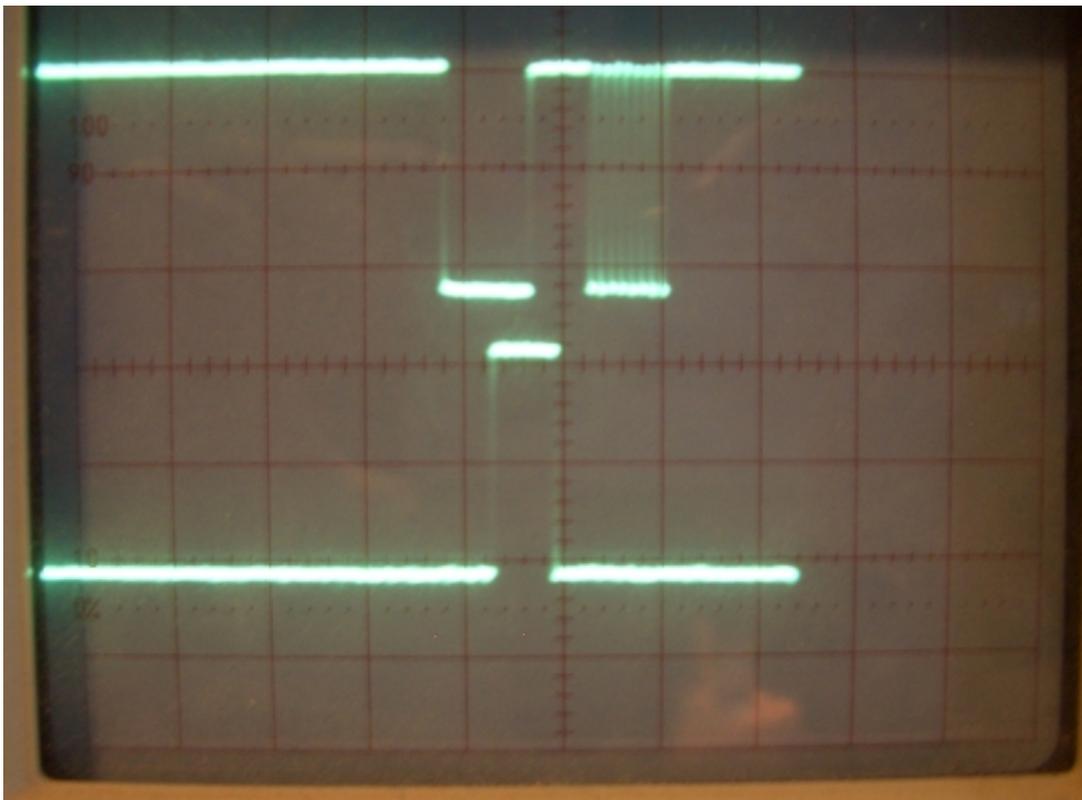
Beispielprotokoll

Für das Generieren der Steuersignale wird die folgende Sequenz benutzt:

#	ASCII-Notation	Hexadezimal-Notation	Erläuterung
1	\1	0x5C 0x31	Umstellen des SPI-Modes auf 1. CLOCK wechselt von 1 auf 0.
2	\j	0x5C 0x4A	LOAD (am INT-Pin) wird gesetzt.
3	\t	0x5C 0x74	Warte mindestens 100 µs (Ruhe).

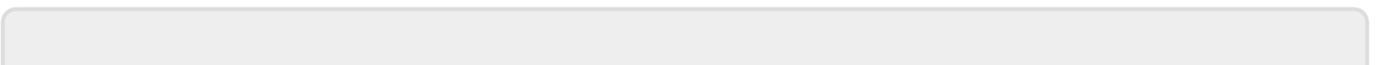
4	\2	0x5C 0x32	Schalte SPI-Mode auf 2 zurück. CLOCK wechselt von 0 auf 1. Damit werden durch das anliegende LOAD-Signal die Rückmeldesignale entlang des Busses in die Register übernommen.
5	\I	0x5C 0x49	Aktiviere RESET (am CS-Pin) von 0 auf 1. Damit werden eventuell vorhandene Latches zurückgesetzt.
6	\i	0x5C 0x69	Schalte RESET von 1 auf 0 zurück.
7	\j	0x5C 0x74	Schalte LOAD von 1 auf 0 zurück. Nun ist der S88 vorbereitet, um die Daten bitweise zum Netzer zu schieben.
8...	a	0x61	Nun können beliebige viele Daten übertragen werden (sinnvollerweise nur soviel wie S88-Module am Strang hängen - Module mit 16 Eingängen zählen natürlich doppelt). Jedes Byte (ohne Fluchtsequenz) bewirkt ein Schieben des Busses um 8 Bit (normaler SPI-Modus). Gleichzeitig werden die Daten von der Leitung MI übernommen und über das Socket versendet.

Und so sieht es dann aus (Oben ist der Takt, unten LOAD - das Timing des ersten Bytes):



PC-Software

Kommt bald...



From:

<http://mobacon.de/wiki/> - MoBaCon Wiki

Permanent link:

<http://mobacon.de/wiki/doku.php/de/netzer/s88>

Last update: **2014/02/09 13:32**

