

Schalten via RS485

Relais-Modul für den ElektorBus

Von Jens Nickel

Das ElektorBus-Projekt zeigt, wie groß das Interesse an der Hausautomatisierung ist. Hier stellen wir daher eine kompakte Platine vor, die zwei 230-V-Verbraucher schalten kann. Außerdem sind zwei Controller-Eingänge auf Schraubklemmen geführt, so dass sich auch die Stellung von Schaltern erfassen lässt. Die Software ist mit dem ElektorBus kompatibel. Ein weiterer Baustein also für die Haussteuerung über PC, Tablet und Smartphone!

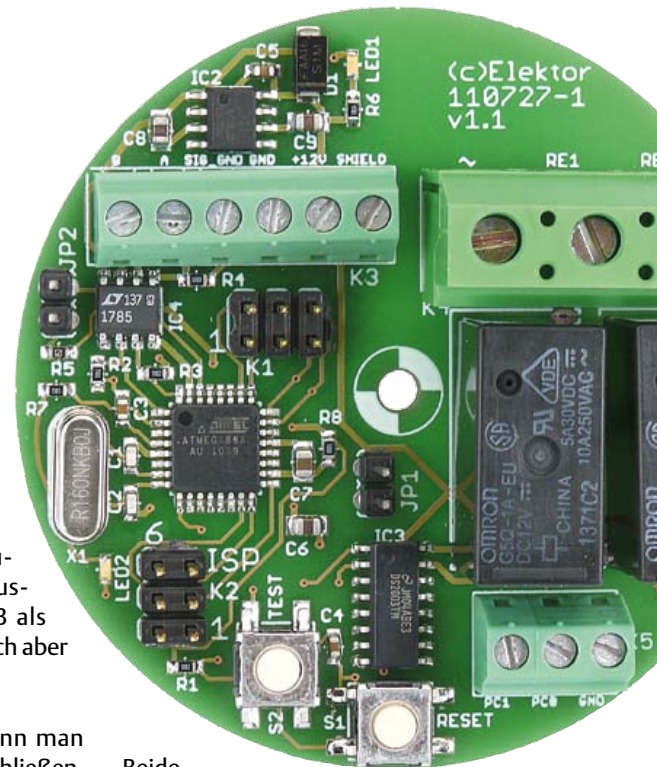
Bisher gab es für unser ElektorBus-Projekt nur Hard- und Software zu Experimentier- und Entwicklungszwecken; das soll nun schrittweise anders werden. Die interessanteste mögliche Anwendung eines Bussystems ist für viele Leser sicher die Hausautomatisierung. Daher beginnen wir mit einer Platine, die sich speziell für das Schalten von zwei (230-V-)Verbrauchern und das Erfassen von zwei Schalterstellungen eignet. Das kompakte Schalt-Modul ist mit einem ATmega88 und einem RS485-Treiber LT1785 ausgestattet; die gleichen ICs also, die sich auch auf dem Experimental-Knoten des ElektorBus finden [1]. Darüber hinaus ist Demo-Software für Controller, PC und Smartphone downloadbar, die zu den ElektorBus-Protokollen kompatibel ist. Das Relais-Modul, das von Elektor schon vollständig bestückt angeboten wird, ist aber natürlich nicht nur für dieses Projekt tauglich.

Bus-Anschluss

Das Schaltbild ist in **Bild 1** zu sehen. Die Versorgung mit 12 V Gleichspannung erfolgt über die Schraubklemme K3. Die RS485-Teilschaltung entspricht fast vollständig derjenigen des Experimental-Knotens, mit den Datenleitungen RO und DI, den Steuerleitungen DE und /RE (Flusskontrolle für den Halbduplexbetrieb), sowie den RS485-Leitungen A und B. Auch hier ist ein Abschlusswiderstand von 120 Ohm zwischen A und B über einen Jumper zuschaltbar. An der Schraubklemme K3 sind unsere

vier ElektorBus-Leitungen anzuschließen: 12 V und GND als Bus-Stromversorgung sowie A und B als Datenleitungen. Daneben finden sich aber noch zwei Extra-Anschlüsse.

An den Klemmenpin „Shield“ kann man die Abschirmung eines Kabels anschließen. Diese lässt sich dann über JP1 auf Masse legen. Typischerweise macht man das nur bei einem der Bus-Knoten, zum Beispiel dort, wo die Stromversorgung sitzt. „SIG_GND“ (Signal Ground) bietet einen Extra-Zugang zum Masse-Anschluss des RS485-Treibers. Im Auslieferungszustand des Schalt-Moduls ist dieser mit der Bus-Masse verbunden. Wir wollten aber die Möglichkeit vorsehen, für die RS485-Treiber eine Extra-Masse bereitstellen zu können. Falls Lasten geschaltet werden, treten lokale Störungen auf der Bus-Masse auf. Und falls die Massepotentiale der Treiber zu sehr differieren, können im ungünstigsten Fall Bits aus dem Nichts entstehen (wie wir schon einmal festgestellt haben [2]). Eine Lösung könnte sein, die Signal-Masse nur an einer Stelle des Busses mit der Bus-Masse zu verbinden. Bei den anderen Busteilnehmern werden beide Massen dann beispielsweise durch einen 100-Ohm-Widerstand getrennt. Das verwendete Bus-Kabel muss dann natürlich eine fünfte Ader mitbringen. Bisher haben wir das allerdings nicht getestet, wir wollen aber in Kürze beginnen, einmal ein ausgewachsenes Bus-System aufzubauen.



Beide neuen Pins sind in Kombination mit verschiedenen Kabeln in jedem Fall eine Spielwiese für Experimente – bei uns und bei Ihnen!

Controller-Pins

Die Test-LED, den Test-Button, den ISP-Programmierschlüssel sowie die Beschaltung von Versorgungs- und Taktpins des Controllers haben wir ebenfalls vom Experimental-Knoten übernommen. Statt einer einreihigen Stiftleiste für Erweiterungen haben wir diesmal aber eine 2x3-Stiftleiste genutzt, die kompakter ist und für die sich leichter passende Kabel finden lassen. Hierüber hat man wieder Zugriff auf vier Pins des Controllers, die als digitale Aus- oder Eingänge beziehungsweise vier analoge Eingänge (ADC0...ADC3) fungieren können. Zwei der Pins sind genauso wie die Masse auf eine zusätzliche Schraubklemme geführt. Hier kann man beispielsweise Lichtschalter anschließen. **Selbstverständlich dürfen die Schalter und alle beteiligten Kabel dann keine Netzspannung führen!**

Das Haupt-Feature des Moduls sind die zwei relais-geschalteten Ausgänge. Dazu wird

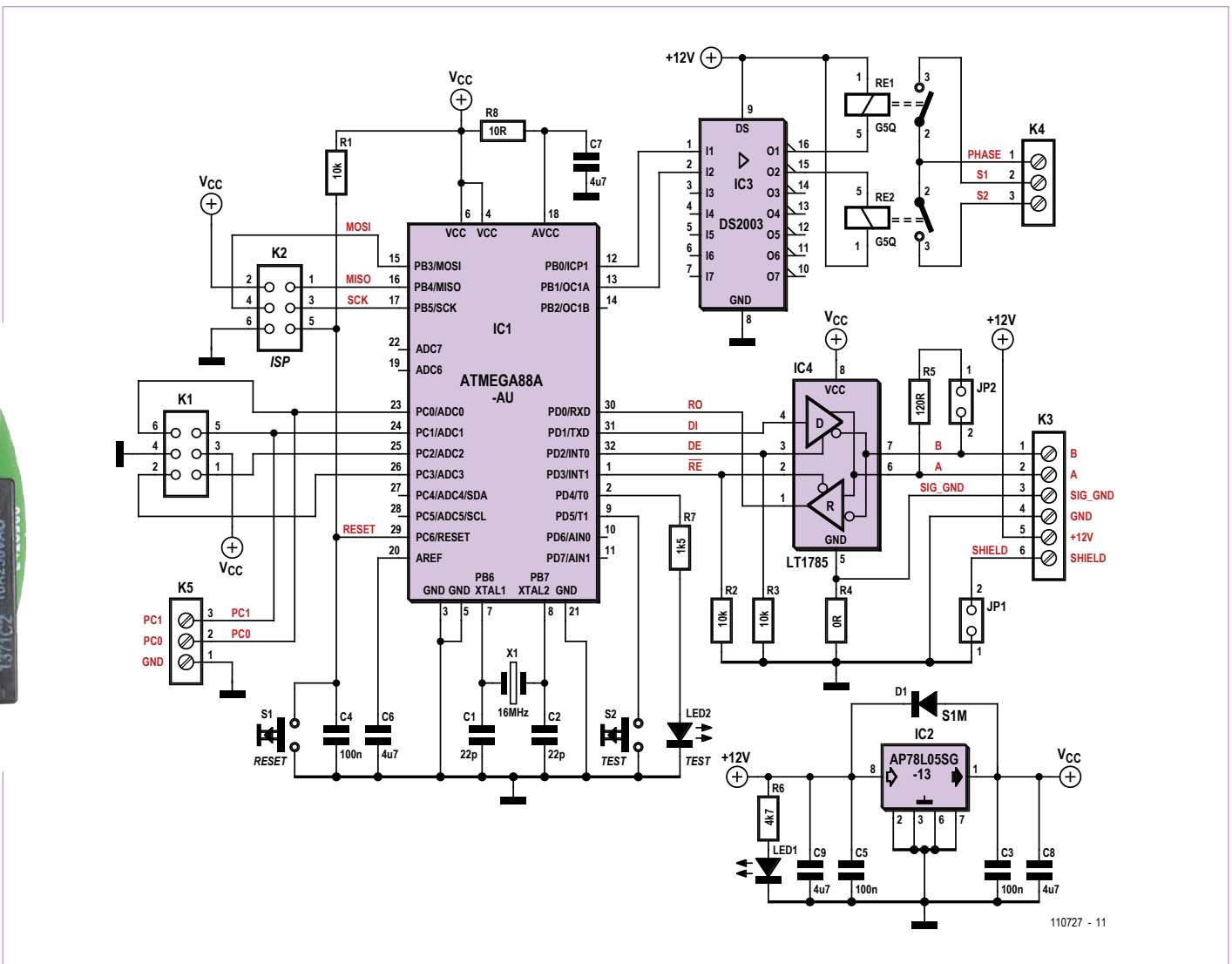


Bild 1. An das Modul lassen sich zwei Verbraucher, aber auch zwei Schalter über Schraubklemmen anschließen.

der Relaisreiber IC3 von zwei Controllerpins PB0 und PB1 angesteuert. Als Relais kommen „SPST-NO“-Typen zum Einsatz (Single Pole, Single Throw, Normally Open); sie verbinden in angezogenem Zustand eine Leitung „Phase“ mit den zwei Ausgängen „S1“ und „S2“. Alle drei Leitungen sind an extra-großen Schraubklemmen zugänglich. Es lassen sich Kleinspannungs- und auch Netzspannungs-Verbraucher wie Leuchten schalten. Wenn letztere angeschlossen werden, muss das Modul berührungsgeschützt und entsprechend den VDE-Bestimmungen (VDE 0100, VDE 0701) in ein Gehäuse ein-

gebaut werden! Näheres findet man auf der Sicherheitsseite in diesem Heft. Die Platine ist rund und kompakt, damit sie in einer 60-mm-Installationsdose Platz findet (Bild 2). **Nicht** benutzen darf man eine Verteilerdose, in die bereits 230-V-Leitungen gelegt sind, und die Busleitungen dürfen auch **nicht** parallel zu Netzspannungs-Leitungen in einem Kabelrohr geführt werden (siehe hierzu auch den Kasten „Kabel“). Für das Schaltmodul muss also eine Extra-Dose und für das Bus-Kabel ein Extra-Leerrohr vorgesehen werden. In Deutschland dürfen Elektroinstallationen nur von ausgebildeten

Fachkräften durchgeführt werden, die bei eingetragenen Betrieben (Installateursverzeichnis des Netzbetreibers) arbeiten.

Software

Erst die passende Firmware haucht dem Schalt-Modul Leben ein. Wir haben darauf verzichtet, den Controller im Auslieferungszustand zu programmieren, da die bestückte Platine ja für alle möglichen Einsatzszenarien tauglich sein soll. Es gibt aber wie immer Demo-Software, die natürlich ElektorBus-kompatibel ist. Die Relais lassen sich dann zum Beispiel vom PC aus schal-

Elektor Produkte & Service

- RS485-Schaltmodul (fertig aufgebaut und getestet 110727-91)
- RS485/USB-Konverter (fertig aufgebaut und getestet 110258-91)

- Gratis Software-Download

Alle Produkte und Downloads sind über die Website zu diesem Artikel erhältlich: www.elektor.de/110727

Stückliste

Widerstände (0603):

R1,R2,R3 = 10 k
 R4 = 0 Ω
 R5 = 120 Ω
 R6 = 4k7
 R7 = 1k5
 R8 = 10 Ω

Kondensatoren:

C1,C2 = 22 p, NP0, 0603
 C3,C4,C5 = 100 n, X5R, 0603
 C6,C7,C8 = 4µ7, X5R, 10 V, 0805
 C9 = 4µ7, X5R, 25V, 0805

Halbleiter:

D1 = Diode S1M, 1000 V, 1 A
 LED1,LED2 = LED grün (0603)
 IC1 = ATmega88A-AU
 IC2 = 5-V-Regler AP78L05SG-13
 IC3 = Relaisreiber DS2003TM/NOPB
 IC4 = RS485-Transceiver LT1785CS8#PBF

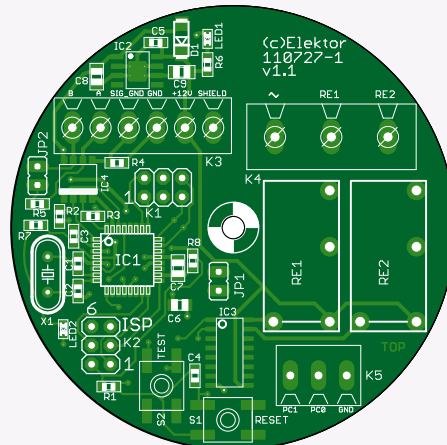


Bild 2. Um die Platine kompakt zu halten, wurden SMD-Bauteile eingesetzt. Die runde Form macht den Einbau in eine 60-mm-Installationsdose möglich.

Außerdem:

RE1,RE2 = 12-V-Relais SPST-NO (z.B. G5Q-1A-EU DC12)
 JP1,JP2 = 1x2 Stiftleiste 2,54 mm
 K1,K2 = 2x3 Stiftleiste 2,54 mm
 K3 = 6x-Schraubklemme, 3,81 mm
 K4 = 3x-Schraubklemme, 7,62 mm
 K5 = 3x-Schraubklemme, 3,81 mm
 X1 = Quarz, 16 MHz, HC-49S
 S1,S2 = Taster SPNO (zum Beispiel B3S-1000)
 Platine PCB 110727-1

oder
 Platine fertig aufgebaut und bestückt
 110727-91

ten, wenn man unseren RS485/USB-Konverter und die PC-Software „ElektorBus-Browser.exe“ benutzt. Dasselbe geht auch von einem Android-Smartphone oder Tablet aus, mit der in den letzten beiden Ausgaben vorgestellten Kombination aus AndroPod-Platine und der kostenlosen App „Elektor-BusBrowserForAndroPod“ [3][4]. Eine passende Benutzeroberfläche zum Schalten der Relais liefern wir auch gleich mit. Sie ist in HTML realisiert und daher gleichzeitig für den PC und das Handy verwendbar. Die Hardware wird wie in Bild 3 gezeigt verkabelt. Die gesamte Software kann man wie immer kostenlos von der Projekt-Website zu diesem Artikel downloaden [5], auch der Quellcode ist selbstverständlich dabei. Nach Entpacken des Zip-Archivs zieht man zuerst den Ordner „UIBus“ auf den Desktop, danach überträgt man bei Bedarf die darin enthaltenen Files aufs Smartphone (zum Beispiel mit der PC-Software AdifController, so wie in [4] beschrieben). Den Controller programmiert man mit der Firmware, die

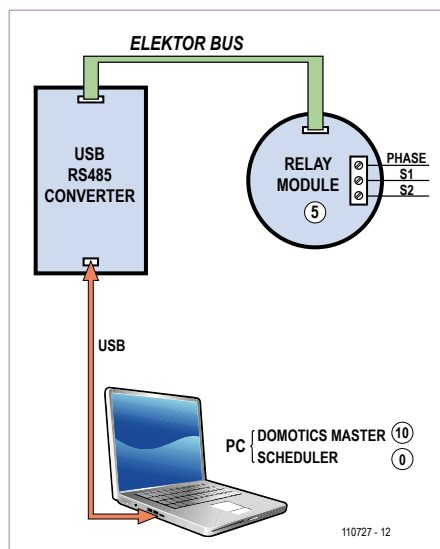


Bild 3. Für einen ersten Test verbindet man einen RS485/USB-Konverter und ein Relais-Modul über den Bus. Das Schaltmodul hat die Knoten-Adresse 5 (fest in der Demo-Firmware eincodiert).

als C-Code (für das kostenlose „AVR Studio“) und als Hex-File im Download enthalten ist. Das EEPROM des ATmega88 spielt diesmal keine Rolle, die Knotenadresse „5“ für die Schaltplatine ist fest im Programm eincodiert.

Test

Nach dem Start des ElektorBusBrowsers wird die HTML-Benutzeroberfläche angezeigt (siehe Bild 4). Bei der PC-Version muss man in der Combo-Box oben zuerst den COM-Port einstellen, an den der RS485/USB-Konverter angeschlossen ist, und danach auf den Knopf daneben drücken. Dann wirft man den ElektorBus-Scheduler mit dem Button „on“ an, der sich innerhalb des HTML-User-Interfaces befindet. Der Scheduler ruft in regelmäßigen Abständen den Master mit der Knotenadresse „10“ auf, der daraufhin einen eventuell vorliegenden Schaltbefehl an den Relais-Knoten versenden darf. Natürlich möchte man dann auch eine Rückmeldung haben. Wenn ein Schalt-

Kabel

Inzwischen haben wir auch einen kleinen Bus-Kabel-Test durchgeführt. Die Details sind im Labcenter der letzten Ausgabe nachzulesen, außerdem kann man auf YouTube ein Video ansehen [7]. Los ging es mit einem 10-adrigen Kabel aus alten Laborbeständen (ein Leitungspaar verdreht), dann kam ein CAT5E-Kabel an die Reihe (4 x Twisted Pair, geschirmt). Leitungslängen von rund 30 m waren in beiden Fällen kein Problem, so dass wir uns eine Kabel-Empfehlung sparen können. Man sollte nur darauf achten, dass für die Signale A und B ein verdrehtes Leitungspaar verwendet wird.

Das Bus-Kabel darf allerdings nicht zusammen mit 230-V-Leitungen in einem Kabelrohr geführt werden, dies ist nur bei ganz bestimmten, hierfür vorgesehenen Kabeltypen erlaubt (EN V 50090-5-2). Eines der erlaubten Kabel ist der Typ YCYM 2x2x0,8 für das EIB/KNX-Hausautomatisierungssystem. Es müsste möglich sein, ein solches Kabel auch für den ElektorBus einzusetzen, was wir aber nicht getestet haben. Das Kabel ist darüber hinaus mit einigen Euro pro Meter recht teuer.

befehl einging, versendet das Relais-Modul daher eine Nachricht, die den Status beider Relais enthält. Dazu muss das Modul nicht direkt vom Scheduler aufgerufen werden. Denn prinzipiell ließen sich über 100 Schalt-Module an den Bus anschließen, da würde es keinen Sinn machen, ständig den Status aller Relais abzufragen. Die Status-Nachricht wird vielmehr in der sogenannten *FreeBus-Phase* gesendet, die der Scheduler immer nach jeder Abfrage des Masters aufruft. In den FreeBusPhases dürfen alle Teilnehmer senden, die sporadisch etwas mitzuteilen haben. Hier können Kollisionen auftreten, wenn man mehrere solcher Knoten angeschlossen hat. Daher muss die Rückmelde-Message des Relais-Moduls – wenn man es wie wir ganz sicher gestalten will – wieder-

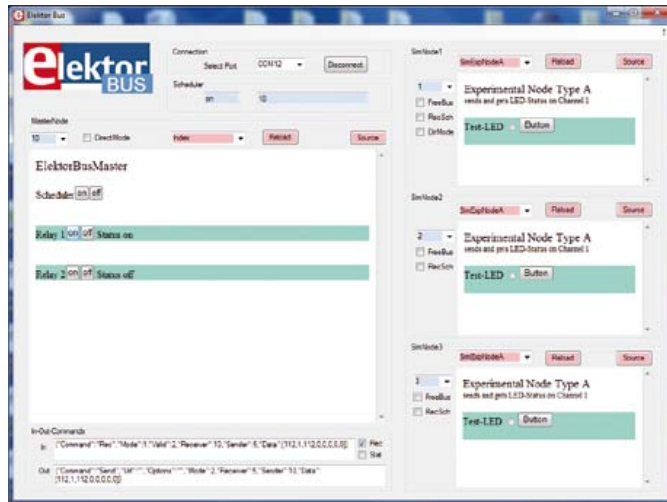


Bild 4. Die PC-Software ElektorBusBrowser mit der HTML-Benutzeroberfläche (links). In den Fenstern rechts sieht man simulierte Bus-Knoten, die sich zu Entwicklungszwecken aktivieren lassen [8].

Anzeige

EXPRESSLIEFERUNG AB 12 STUNDEN.

GARANTIERT PÜNKTLICH ODER GARANTIERT KOSTENLOS.

LEITON
RECHNEN SIE MIT BESTEM SERVICE

Superpünktlich wie im Flugel Denn der LeitOn Leiterplatten-Expressdienst braucht nur 12 Stunden, um 1- und 2-lagige Leiterplatten an den Versender zu übergeben. Schaffen wir das ausnahmsweise einmal nicht, sind die Platinen absolut gratis! Doch vor dem Bestellen kommt das Kalkulieren. Bei LeitOn geht das besonders einfach: online, und das auch für **Schablonen und flexible Leiterplatten!** Das ist weltweit einmalig! Neugierig geworden? Unsere persönliche Telefonberatung und unser kompetenter Außendienst helfen Ihnen gerne weiter.

www.leiton.de Info-Hotline +49 (0)30 701 73 49 0

FUNKtioniert!

Kabellose Klangerlebnisse – die Multikanal-Audiodfunkmodule **WA-TX-03-R** und **WA-RX-03-R** machen sie möglich. Bei der Übertragung von Audiosignalen im über 70dB großen Dynamikbereich wird dank Kompandertechnik eine sehr gute Rauschunterdrückung erreicht. Und somit eine hervorragende Tonqualität zu einem günstigen Preis. Die niedrige Betriebsspannung sowie die kompakte Größe machen die Anwendung äußerst komfortabel. Die Wireless Sender- und Empfängermodule eignen sich z.B. für kabellose Konferenzsysteme, Mikrofonsysteme für Amateure oder Audioguides in Museen.

Nutzen Sie unsere Technologie und Kompetenz für Ihre Ideen.

Vertrieb durch:
Reimesch
 Kommunikationssysteme GmbH
 Friedrich-Ebert-Str. · D-51429 Bergisch Gladbach
 Tel.: 0 22 04 / 58 47 51 · Fax: 0 22 04 / 58 47 67
www.reimesch.de · kontakt@reimesch.de

Vertrieb durch:
CIRCUIT DESIGN GmbH
 Schleißheimer Str. 263 · D-80609 München
 Tel.: +49 / 89 / 35 82 83-60 · Fax: +49 / 89 / 35 82 83-66
www.circuitdesign.de · info@circuitdesign.de

Listing: User-Interface mit HTML/Javascript (Ausschnitt)

```

<SCRIPT src='JSBus.txt' Language='javascript' ></SCRIPT>
<SCRIPT Language='javascript' >

function ProcessPart(part)
{
  if (part != null)
  {
    if(part.Sender == 5)
    {
      if(part.Channel == 0)
      {
        if (part.Numvalue == 0) {TextSetvalue ("StatusRelay1","off");}
        if (part.Numvalue == 1) {TextSetvalue ("StatusRelay1","on");}
      }

      if(part.Channel == 1)
      {
        if (part.Numvalue == 0) {TextSetvalue ("StatusRelay2","off");}
        if (part.Numvalue == 1) {TextSetvalue ("StatusRelay2","on");}
      }
    }
  }
}

function SwitchRelay1(Status)
{
  var parts = InitParts();
  parts = SetValue(parts, 10, 5, 0, 0, Status);
  SendParts(parts, true);
}

function SwitchRelay2(Status)
{
  var parts = InitParts();
  parts = SetValue(parts, 10, 5, 1, 0, Status);
  SendParts(parts, true);
}

</SCRIPT>
...

```

rum vom PC bestätigt werden. Diese *AcknowledgeMessage* für FreeBusPhase-Knoten wird jedoch automatisch versandt, und zwar von der Javascript-Bibliothek JSBus, die in das Demo-HTML-File eingebunden ist (siehe **Listing**). Die Bibliothek decodiert die Nachricht des Relais-Knotens auch, der

eigentliche Anwender-Code muss lediglich die beiden Parts (Informations-Einheiten) entgegennehmen, aus denen die Message besteht. Der erste Part repräsentiert den Status des Relais 1 (auf Channel 0 gesendet), der zweite Part den Status des Relais 2 (auf Channel 1 innerhalb der Message

gesendet). Für beide Parts wird wie immer die Funktion `ProcessPart` im Javascript-Teil der HTML-Seite aufgerufen. Innerhalb der Funktion verwenden wir den übertragenen Status-Wert (0 für „aus“ und 1 für „an“), um den Status-Text der Relais in der HTML-Oberfläche umzuschalten.

All dies haben wir schon des Öfteren in den ElektorBus-Artikeln beschrieben. Ab nun muss man aber nicht mehr in die Artikel selbst schauen, wenn man etwas nachschlagen will. Die Details des MessageProtocols, des ApplicationProtocols und des Rapid-Development-Systems mit HTML und Javascript kann man in der neuen ElektorBus-Referenz nachlesen, die kostenlos unter [6] herunterzuladen ist.

Ausblick

Das Schalten per PC und Smartphone ist schon eine recht nette Anwendung, aber so richtig schön wäre es, wenn wir das Ganze auch aus der Ferne bewerkstelligen könnten – am besten wäre ein weltweiter Zugang. Nun, hierfür wird es in naher Zukunft gleich mehrere Möglichkeiten geben. In der Pipeline ist ein größeres Mikrocontrollerboard, das natürlich mit RS485, aber auch mit einem Sockel für ein Netzwerkmodul ausgestattet ist. Darüber hinaus wollen wir die kommunikativen Fähigkeiten eines Smartphones oder Tablets, das an den Bus angeschlossen ist, noch ein wenig mehr ausreizen. Man darf auf jeden Fall gespannt sein, wo unser Bus noch überall hinfährt!

(110727)

Weblinks

- [1] www.elektor.de/110258
- [2] www.elektor.de/110225
- [3] www.elektor.de/110405
- [4] www.elektor.de/120097
- [5] www.elektor.de/110727
- [6] www.elektor.com/elektorbus
- [7] www.youtube.com/watch?v=rbdSTXNARmw
- [8] www.elektor.de/110708