

Simulationsbeispiele

Die Simulationsbeispiele können auf PCs mit **Microsoft Windows** unter **MathWorks Matlab** oder unter **GNU Octave** (Version $\geq 4.2.x$) verwendet werden. Auf PCs mit **Linux** (z.B. Ubuntu) steht der volle Umfang der Beispiele nur unter **GNU Octave** zur Verfügung. Hier bestehen Einschränkungen bezüglich der mex-Funktionen, die wir nur für **GNU Octave** unter **Ubuntu 16.04** bereitstellen. Nutzern von **Linux** empfehlen wir deshalb, die weiter unten beschriebene virtuelle Maschine zu benutzen.

Beispiele für RTL-SDR-kompatible USB-Miniaturempfänger

Der Betrieb von [RTL-SDR-kompatiblen USB-Miniaturempfängern](#) mit der mex-Funktion `mexrtlsdr` ist ohne größeren Aufwand nur unter **Microsoft Windows** möglich. Dazu müssen Sie mit dem in den Beispielen enthaltenen Programm `zadig-2.3.exe` den WinUSB-Treiber für den Empfänger installieren. Eine Anleitung dazu finden Sie [hier](#). Nutzer von **Microsoft Windows 10** machen jedoch sporadisch immer wieder die Erfahrung, dass Sie den Treiber nicht installieren können oder dass der Empfänger trotz erfolgreicher Installation nicht angesprochen werden kann. Sollte dies bei Ihnen der Fall sein, empfehlen wir, die weiter unten beschriebene virtuelle Maschine zu benutzen. Nutzer von **Linux** müssen auf jeden Fall die virtuelle Maschine benutzen. Falls Sie noch keinen Empfänger haben, empfehlen wir die Produkte der Firma NooElec, z.B. den [NooElec NESDR Mini 2+](#).

Wenn Sie unter **Microsoft Windows** den WinUSB-Treiber installiert haben, rufen Sie unter Matlab oder Octave die Funktion `rcv_broadcast` auf, die einen FM-Rundfunkempfang ermöglicht. Die voreingestellte Empfangsfrequenz von 99.8 MHz ist bei Ihnen möglicherweise nicht belegt, so dass Sie nur Rauschen hören und sehen, es geht aber zunächst nur darum, dass die Funktion läuft. Sie stoppen die Funktion durch Schließen des Anzeigefensters. Danach ermitteln Sie die Sendefrequenz eines starken FM-Rundfunksenders in Ihrer Umgebung, setzen diese Frequenz in der Funktion `rcv_broadcast` in der Variablen `center_frequency` (Zeile 24) und starten die Funktion erneut. Zur Suche nach einem starken FM-Rundfunksender können Sie auch die Funktion `rcv_scan_broadcast` aufrufen, die ein Spektrum des FM-Rundfunkbands (88...108 MHz) erstellt. Diese Funktion läuft aber nicht auf allen PCs störungsfrei. Beim ersten Durchlauf durch das Band, der etwa 10 bis 20 Sekunden dauert, wird der passende Wert für die Verstärkungseinstellung des Empfängers ermittelt; dabei kann die Anzeige noch leer bleiben, wenn die Empfangsbedingungen ungünstig sind. Danach sollte das Spektrum zu sehen sein.

Virtuelle Maschine

Zur Inbetriebnahme nach dem Download sind folgende Schritte erforderlich:

- Entpacken Sie die heruntergeladene Datei. Die entpackte Datei mit dem Dateinamen **SDR.vmdk** hat eine Größe von etwa 4.1 GB.
- Im **VMware Workstation Player** erzeugen Sie mit **Create a new virtual machine** eine neue virtuelle Maschine, bei der Sie:
 - die Option **I will install the operating system later** wählen;
 - **Linux** als **Guest Operating System** auswählen;
 - unter **Version** den Eintrag **Ubuntu 64-bit** selektieren;
 - unter **Virtual machine name** den Namen **SDR** eingeben;
 - die Option **Store virtual disk as a single file** wählen.
- Nach dem Erstellen der virtuellen Maschine ersetzen Sie die vom **VMware Workstation Player**

erstellte Datei **SDR.vmdk** durch die heruntergeladene und entpackte Datei gleichen Namens. Danach können Sie die virtuelle Maschine starten. Beim Start können Fehler auftreten, die mit **Retry** oder anderen auswählbaren Optionen zum Weitermachen umgangen werden können.

- Das **Kennwort** zum Einloggen in die virtuelle Maschine (Lubuntu 16.04) lautet **sdruser**.
- Starten Sie Octave mit dem Desktop-Symbol **Octave SDR**. Nun können Sie die Beispiele in Octave aufrufen.
- Schliessen Sie nun Ihren RTL-SDR-kompatiblen Empfänger an den Host-PC an. Damit der Empfänger an die virtuelle Maschine durchgereicht wird, müssen Sie ihn im **VMware Workstation Player** über den Menü-Punkt **Player/Removeable Devices** aktivieren.
- Starten Sie in Octave die Funktion `rcv_broadcast`, wie oben beschrieben.

Senden mit der Smart Watch TI EZ430-Chronos

Besonders interessant wird die Sache, wenn man nicht nur empfangen, sondern auch senden kann. Auch dazu gibt es eine preiswerte Möglichkeit: die Smart Watch [EZ430-Chronos](#) von Texas Instruments. Das Uhrenmodul kann dem Gehäuse entnommen und über einen Adapter an eine USB-Schnittstelle angeschlossen werden. Mit Hilfe der Software [SmartRF Studio](#) kann das Modul parametrieren und als Sender im Bereich von 868 MHz eingesetzt werden. Eine Portion Experimentierfreude müssen Sie dazu allerdings mitbringen. Das nachfolgend verlinkte Video erläutert die ersten Schritte. Wenn Sie zur Parametrierung im SmartRF Studio den Menüpunkt **File/Open Cfg** wählen, die Datei [ti_ez430_chronos_config.xml](#) laden und den Sender starten, müssten Sie das Signal mit der Funktion `rcv_chronos` empfangen können. Dazu muss allerdings die Empfangsfrequenz im Kopf der Funktion relativ genau eingestellt werden. Das erfordert in der Regel ein paar Versuche mit jeweils um etwa +/- 5 kHz angepasster Empfangsfrequenz. Alternativ können Sie die oben genannte Funktion `rcv_broadcast` zur Ermittlung der exakten Empfangsfrequenz verwenden, indem Sie die Empfangsfrequenz in der Variablen `center_frequency` (Zeile 24) auf den Sollwert 868 MHz einstellen und die Frequenzabweichung des Signals aus dem angezeigten Spektrum ablesen.

Bezugsquelle: [EZ430-Chronos-868](#)

Einführungsvideo: [EZ430-Chronos - Disassembly and Assembly](#)