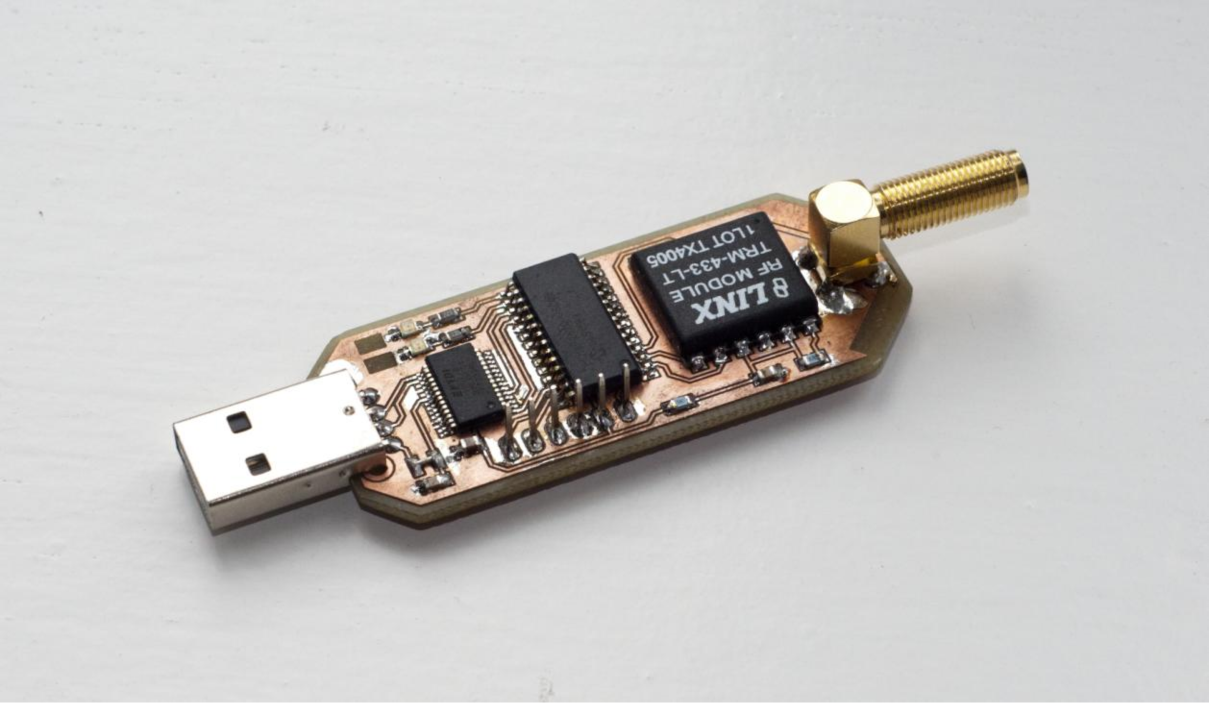

MultiTRX V3, firmware 1.0
Operating manual

First revision

Jon Petter Skagmo 2011 - skagmo.com

Introduction and overview

Tons of wireless sensors, remote power switches, alarms etc. use the 433.92 MHz ISM band for wireless transmission of data. Unfortunately, not many of these products can be easily controlled with a computer. The MultiTRX aims to solve the problem of easily decoding and encoding of signals from cheap consumer electronics from stores like Clas Ohlson. It does a lot of things not present in any other device. Among other exiting features it decodes and encodes the unique ID of remote power switches, enabling you to keep the pairing with your existing remote control in remote power switches which can only be paired with one device.

Getting started

Once plugged in, the MultiTRX will appear as a serial port on your computer. On Windows systems, check device manager to find out which COM-port number has been assigned to the device. On Linux based systems, the device will appear in */dev/* as *ttyUSBx*.

To test the device in Windows, download a serial terminal software like PuTTY, and start a serial session at 9600 baud (8N1). Send *$P* to the device and you should receive *$ACK*. Data received from RF should now appear in the terminal as they are received.

Receiving data

The MultiTRX can decode data from Esic temperature sensors, Doro Secur sensors and Nexa and Gao remote controls. See the complete command list on the next page.

To make sure this device will not be used to sabotage other users of remote power switches, a software limit on reception of Nexa and Gao devices has been added. This limit disables printing of these devices until at certain number of equal packets has been received. This means you have to hold the button on your remote a few seconds before the full packet is shown.

Controlling power-switches

The easiest way of controlling your remote power switch is to keep your (already paired) remote control close to the MultiTRX and copy the string received, eg. *$GABC2F0.* This way you can switch on and off this device by sending *$GABC2F1* and *$GABC2F0* , respectively.

Gao-specific

If you want to pair a Gao device with the MultiTRX directly, send the string above, but replace the power on/off parameter with the number two: *$GABC2F2.*

Nexa-specific

Nexa dimmers has the great feature of being able to dim to an absolute level in 16 steps. To do so, replace the power on/off parameter with the number two, and add an extra hexadecimal character (upper case) with the dim level: *$N1234ABC030 -> $N1234ABC032F* (will set the device to maximum brightness).

Complete command list

All communication is done at 8N1 9600 baud, ASCII-formatted data. Data from the board is terminated with CR and LF (ASCII character 13 and 10). Data sent to the board does not need any kind of termination, as the commands are fixed length. All hexadecimal characters must be in upper case!

Commands transmitted from MultiTRX (RF-reception)

Output	Description	Example(s)
$Dddd	Doro Secur sensor. d = dip switch settings (hex).	$DA07
$Eh,c,t,H	Esic temperature sensor. h = house code, c = channel, t = temperature (degrees Celsius), H = relative humidity (percent).	$E5,1,12.3,71
$Giiiiip	Gao remote power switch. i = unique ID (hex), p = power setting (off = '0', on = '1') and pairing mode ('2').	$GABC2F0 $GABC2F1
$Niiiiiiigcp	Nexa remote power switch. i = unique ID (hex), g = group enable ('0'/'1'), c = channel (hex), p = power setting (off = '0', on = '1'). Packets with dim level will not be received!	$N1234ABC010 $N1234ABC011

Commands transmitted to the MultiTRX (RF-transmission)

Output	Description	Example(s)
$Giiiiip	Gao remote power switch. i = unique ID (hex), p = power setting (off = '0', on = '1') and pairing mode ('2')	*$GABC2F0* *$GABC2F1*
$Niiiiiiigcp(l)	Nexa remote power switch. i = unique ID (hex), g = group enable ('0'/'1'), c = channel (hex), p = power setting (off = '0', on = '1', absolute dim level = '2'), l = dim level (hex). Dim level only applies if p = '2', so the length of this string varies.	$N1234ABC010 $N1234ABC011 $N1234ABC012F
$Whuup	Waveman remote power switch. h = house code ('A'-'P'), uu = unit code ('01'-'16'), p = power setting ('0'/'1')	$WA011

Compatible hardware

Sensors

- All wireless Doro Secur sensors (available from Alarm-Agenturet)
- Esic temperature sensor (available from Clas Ohlson)

Power switches

- Self learning Nexa (available from e.g. Jula)
- Waveman (available from Clas Ohlson)
- Gao/CO-tech (available from Clas Ohlson)

Hardware specification

Supported devices (RX)	Esic temperature sensor, Doro Secur sensors and Gao and Nexa remote power switches
Supported devices (TX)	Gao, Nexa and Waveman remote power switches
Modulation	OOK / AM
Output power	10 dBm
Receiver sensitivity	-112 dBm
Frequency stability	+/- 50kHz over -40 to 85 degrees Celsius