

# SEAHU SH017 (PiToDin) a RFX 433MHz obecně



## ÚVOD

Frekvence 433 MHz je v Evropě volně k dispozici, v Americe se pro tento účel používá frekvence 315 MHz. A právě tyto frekvence používá spousta elektrických zařízení od různých výrobců. Jde hlavně o různé dálkově ovládané zásuvky, spínače, bezdrátové zvonky, alarmy, meteorologické stanice, termostaty, čidla a bezpečnostní systémy.

Výhoda těchto zařízení je jejich nízká cena. Bohužel většinou dostanou koupit pouze jako sety např. dálkový ovládač + bezdrátová zásuvka, tlačítko + zvonek, atd. . Nebo jako uzavřený systém např. bezpečnostní set s několika čidly a řídicí stanicí.

Další výhoda je dostupnost a cena vysílacích a přijímacích modulu, takže se přímo vybízí možnost přidat tyto moduly k centrálnímu počítači, který by mohl všechny tyto levné zařízení ovládat. Lze tak levně vytvořit inteligentní domácí automatizaci.

A právě touto možností se tento článek zabývá. Jako centrální počítač je v tomto článku nasazen mini PC raspberryPI (nejlépe přímo řídicí jednotka Seahu SH017, která obsahuje tento počítač i možnost jednoduše připojit vysílací i přijímací moduly), ale po mírných úpravách lze použít jakýkoliv jiný PC.

Pro vytvoření inteligentní domácnosti je potřeba kromě hardwaru také nějaký systém (program). My jsme zvolili systém domoticz, z důvodu jednoduchého ovládaní, zabudované podpory mnoha zařízení, možnosti ovládaní přes web a mobilní telefony a jednoduchého programování obsluhy události.

PS: pro ten stejný účel se používají taky další frekvence (868 MHz, 2,4 GHz), ale zařízení na těchto frekvencích bývají většinou dražší.

## UKÁZKA KOMUNIKAČNÍHO PROTOKOLU

Zde je potřeba podotknout, že zde není žádná norma, která by usměrňovala způsob komunikace na těchto frekvencích a každý výrobce si to může dělat a také dělá po svém. Proto také existuje spousta různých protokolů. Naštěstí v tomto frekvenčním pásmu se nepoužívají žádné kanály, vysílá se pouze na jedné frekvenci a nepoužívá se žádné modulace. Tedy buď vysílám nebo ne. Komunikace pak probíhá formou jakési morseovky. Přijímače jsou většinou stavěné na vysokou citlivost, což znamená, že když nikdo nevysílá tak chytáme šum (podobně jako na analogové televizi, nebo rádiu). Je potom na programu, aby rozeznal co je šum a co korektní vysílání. Navíc nikdo na této frekvenci neřídí vysílání, tj. může se klidně stát, že v jeden okamžik vysílá více zařízení a výsledná signál je tudíž nečitelný.

Z těchto důvodů je potřeba data vysílat opakovaně. Nejjednodušší zařízení jen své data několikrát opakovaně vyšlou a tím končí. Složitější mají oboustrannou komunikaci tj. vyšlou data a čekají potvrzení o přijetí, pokud ho nedostanou tak celý proces opakují.

### Simple protokol:

Protokol, který používá většina nejjednodušších zařízení (bezdrátové zásuvky, zvonky, atd..)

data se posílají po jednotlivých bitech:

Bity jsou kódovány následovně:

0-bit

HIGH po dobu 375μs následovaná LOW po dobu 1125μs

| ----- | \_\_\_\_\_ |

1-bit

HIGH po dobu 1125μs následovaný LOW po dobu 375μs

| ----- | \_\_\_\_\_ |

PS: HIGH vysílám, LOW nevysílám

Na začátku odeslaných dat je adresa zařízení pro které mají být data určena, zbytek jsou předávaná data. Celé to tvoří jakýsi paket, který je opakovaně vyslán (cca 7x). Jednotlivé pakety se od sebe oddělují delší prodlevou min. 2000μs kdy se nevysílá.

Např. data v paketu pro ovládání bezdrátových zásuvek fi. Kanlux mají následující význam:

vyslany paket									
-----kod zarizeni-----									
0	10	11	20	21	22	23	24	25 (bit)	
-----kod sady-----			-----kod zasuvky-----			tlacitko ON-		tlacitko OFF-	ukoncuji bit-
ON	OFF	ON	ON	OFF	ON	OFF	OFF	OFF	OFF
11	01	11	11	11	01	00	01	01	01
			press	no-press		bez funkce		vysilaci klid min. 2000µs -	
			01	00		0		bez funkce	
								X	

PS: aby zásuvka zareagovala je potřeba stejný paket poslat bezprostředně min. 3x za sebou

## DOSTUPNÝ HARDWARE

K dostání jsou jednoduché přijímače a vysílače v ceně několika dolarů nebo dražší komplexní kontroléry s vlastním procesorem.

### Příklady jednoduchých přijímačů a vysílačů:

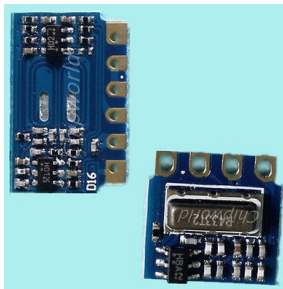
#### Set jednoduchý přijímač a vysílač za cca 20,- Kč

<http://www.ebay.com/itm/1pcs-433Mhz-RF-transmitter-and-receiver-kit-for-Arduino-CA-NEW-/141977599565?hash=item210e863e4d:g:8wEAAOSw6EhUN9s->



přijímač 5V napájení 5V logika (pro raspberryPI je potřeba převodník úrovní)  
vysílač 5V-12V čím víc voltů tím větší vysílací výkon (dosah)

#### Set jednoduchý přijímač a vysílač za cca 45,- Kč

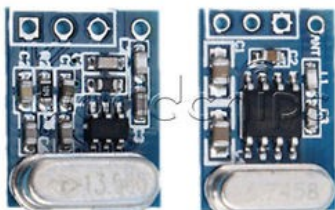


<http://www.ebay.com/itm/433Mhz-MINI-Wireless-Transmitter-and-receiver-Module-Wireless-remote-control/381536934078?trksid=p2047675.c100005.m1851&trkparms=aid%3D2220072%26algo%3DSIC.MBE%26ao%3D2%26asc%3D41451%26meid%3Dd99a4dcd234f400696c82c7f4b1d438b%26pid%3D100005%26rk%3D1%26rkt%3D6%26mehot%3Dag%26sd%3D141977599565>

přijímač 5V napájení 5V logika (pro raspberryPI je potřeba převodník úrovní)  
vysílač 2,6V-12V čím víc voltů tím větší vysílací výkon (dosah)

#### Set jednoduchý přijímač a vysílač za cca 60,- Kč

(433MHZ Transmitter & Receiver Module SYN480R SYN115 ASK Wireless Module)



<http://www.ebay.com/itm/433MHZ-Transmitter-Receiver-Module-SYN480R-SYN115-ASK-Wireless-Module/322056931842?trksid=p2047675.c100011.m1850&trkparms=aid%3D222007%26algo%3DSIC.MBE%26ao%3D1%26asc%3D41451%26meid%3D2aa6bfe766214609a85ec9e5ed0c6747%26pid%3D100011%26rk%3D4%26rkt%3D6%26sd%3D351350062982>

Přijímač 3,3-5,5V, přímo zapojitelné do raspberryPI.  
Vysílač 1,8-3,6V vhodný pro provoz zařízení napájených baterkou.  
**Tyto moduly používáme do řídicí jednotky Seahu SH017.**  
(zapojení do Seahu SH017 viz. dokumentace Hardwarový popis)

## Zapojení jednoduchých přijímačů a vysílačů:

Jednoduché přijímače mají jen 2 piny pro napájení (GND, +napájecí napětí), jeden pin pro příjem dat (0=nebylo detekováno vysílání, 1=bylo detekováno vysílání). Stav na tomto pinu je potřeba průběžně softwarově sledovat a vyhodnocovat. Některé přijímače mají také pin pro připojení externí antény.

Jednoduché vysílače mají také jen piny pro GND, napájení, externí anténu a data pin pomocí kterého lze jednoduše ovládat zda-li modul vysílá či ne (1 - vysílá, 0 – nevysílá).

## Příklady komplexních kontrolérů s vlastním procesorem:

Tyto kontroléry s vlastním procesorem zpracovávají přijatý signál a odchyťávají v přijatém šumu data. Mají zabudovanou podporu mnoha protokolů a zařízení různých výrobců. Pro podporu nových zařízení bývají vybaveny možností upgradu firmwaru. Rozeznanou komunikaci pak předávají nejčastěji v textové podobě nadřazenému počítači. Kontrolér nejen přijímá ale může také vysílat.

### RFXcom USB přijímač-vysílač s vlastním procesorem a vestavěnou podporou mnoha zařízení a protokolu

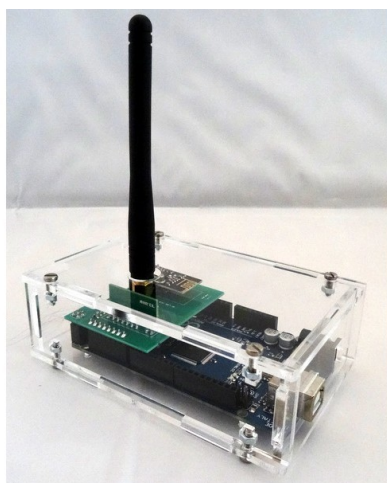


Cena cca 3100,- Kč

<http://www.rfxcom.com>

USB rozhraní pro ovládání nejrozumnějších zařízení na 433MHz. Seznam podporovaných zařízení naleznete na jejich webu. Má přímou podporu pro systém Domoticz.

### RFLink - arduino projekt (freeware pro vývojovou desku 8-bitového počítače s přijímačem a vysílačem).



Cena cca 1100,- za stavebnici.

<http://www.nemcon.nl/blog2/>

Podobné řešení jako výše, ale freeware. Jde o freeware zdarma je plná verze firmware pro vývojovou desku arduino mega. Je také k dispozici část zdrojových kódů, ale jen ta část nutná pro vývoj podpory dalších zařízení.

K počítači se připojuje pomocí USB (ten se chová jako USB → seriál převodník), nebo pomocí ethernetu. 8-bit. procesor zpracovává signál z přijímače a na sériovou linku (nebo síťový port) oznamuje v textové podobě co rozeznal. Obdobně na základě povelů dokáže vysílat. Má vestavěnou podporu mnoha protokolů za zařízení, projekt je aktivní a pravidelně se objevují aktualizace s podporou nových zařízení. Projekt se rozrůstá a už nyní mimo 433MHz podporuje i

moduly a protokoly na 868MHz a 2,4GHz. Má nativní podporu pro systém Domoticz.



PS:

Podarilo se mi RFLink přepsat tak aby běžel přímo na raspberry PI. Zatím jen s volně dostupnými moduly. Podporu ostatních modulů se snažím vyjednat s autorem projektu. Verzi pro raspberryPi naleznete na : <https://github.com/seahu/rflink>

## UTILITY PRO PŘÍJEM A VYSÍLÁNÍ

Programu pro příjem a vysílání existuje celá řada zde se zmíním o utilitách v programovacím jazyce python od Seahu určené pro zkušební účely a utility 433Utils od Suat .zgür v programovacím jazyce C++.

Veškeré tyto utility najdete v adresáři: /opt/seahu/rfx433MHz (ve vašem modulu Seahu SH017).

### SEAHU UTILITY V PYTHONU:

#### *receiver\_433Mhz.py*

jednoduchý prográmeček zobrazující po krátkém časovém intervalu stav z přijímače tj. 1 nebo 0. Velice se mi osvědčil při zkoumání protokolu bezdrátové zásuvky. Pokud se dobře nastaví velikost okna terminálu vznikne obraz šumu nul a jedniček. Při příjmu vysílání jsou pak jasně patrné pravidelné obrazce. Program se ukončí kombinací kláves ctrl+C.

binární šum

ukázka příjmu paketu

#### *record\_433Mhz.py* *název\_souboru* *počet\_zaznamenaných\_sekund*

malý prográmeček, který uloží výše zobrazený záznam zadané doby do zadaného souboru. Vhodné např. pro uložení záznamu např. se stlačením tlačítka na dálkovém ovládači.

#### *play\_433Mhz.py* *název\_souboru*

přehraje zadaný soubor pomocí vysílače. Jednoduše tedy můžete nahrát např. zmáčknutí tlačítka na dálkovém ovládači a pak je kdykoliv přehrát a simulovat tak opětovné zmáčknutí tlačítka na dálkovém ovládači)

***get\_433Mhz.py***

přijímá data data v tom nejjednodušším protokolu a zobrazuje je jako binární číslo, každý paket na samostatný řádek. Program se ukončí kombinací kláves ctrl+C. Podporuje jen simple protokol.

```

Terminal - pi@raspberrypi: /opt/seahu/rfx433MHz
File Edit View Terminal Tabs Help

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 17 13:50:10 2004
pi@raspberrypi:~$ cd /opt/seahu/rfx433MHz/
pi@raspberrypi:/opt/seahu/rfx433MHz$ ./get_433Mhz.py
1111111111000101010100
1111111111000101010100
1111111111000101010100
1111111111000101010100
1111111111000101010100
1111111111000101010001
1111111111000101010001
1111111111000101010001
1111111111000101010001
1111111111000101010001
1111111111000101010111
1111111111000101010100
1111111111000101010100
1111111111000101010100
1111111111000101010100

```

***send\_433Mhz.py binární\_kód***

odvysílá zadaný kód pomocí simple protokolu.

```

Terminal - pi@raspberrypi: /opt/seahu/rfx433MHz
File Edit View Terminal Tabs Help

pi@raspberrypi:/opt/seahu/rfx433MHz$ ./send_433Mhz.py 1111111111000101010100
./send_433Mhz.py:43: RuntimeWarning: This channel is already in use, continuing
anyway. Use GPIO.setwarnings(False) to disable warnings.
GPIO.setup(pin data, GPIO.OUT)
send: 1111111111000101010100
send: 1111111111000101010100
send: 1111111111000101010100
send: 1111111111000101010100
send: 1111111111000101010100
pi@raspberrypi:/opt/seahu/rfx433MHz$

```

## UTILITY Z NINJABLOCKS

zdrojak <https://github.com/ninjablocks/433Utils>

upravený zdroják v /home/pi/433Utils

více na <http://shop.ninjablocks.com/blogs/how-to/7506204-adding-433-to-your-raspberry-pi>

Projekt je společný pro arduino i raspberry PI. Skenování příjmu dat je tam inteligentně vyřešeno pomocí přerušení, což výrazně odlehčuje zátěž procesoru.

Z tohoto projektu používám jen dva programy a to codesend pro odesílání a RFSniffer pro příjem.

Oba tyto programy musí být spouštěny pod superuživatелеm.

***sudo RFSniffer***

přijímá a zobrazuje rozeznaná data s přijímače v binárním formátu za kterým je vedeno číslo protokolu.

```

Terminal - pi@raspberrypi: /opt/seahu/rfx433MHz
Soubor Úpravy Zobrazit Terminál Karty Návoděda

pi@raspberrypi:/opt/seahu/rfx433MHz$ sudo ./RFSniffer
Received 1111111111000101010100 1
Received 1111111111000101010100 1
Received 1111111111000101010100 1
Received 1111111111000101010100 1
Received 1111111111000101010001 1
Received 1111111111000101010001 1

```

***sudo codesend kód\_v\_binárním\_formátu číslo\_protokolu***

odvysílá zadaný kód.

```

Terminal - pi@raspberrypi: /opt/seahu/rfx433MHz
Soubor Úpravy Zobrazit Terminál Karty Návoděda

pi@raspberrypi:/opt/seahu/rfx433MHz$ sudo ./codesend 1111111111000101010001 1
sending code[1111111111000101010001]
pi@raspberrypi:/opt/seahu/rfx433MHz$

```

Za zmínku stojí na těchto utilitách postavený projekt **IoT-433mhz** na adrese:

<https://www.hackster.io/roccomuso/iot-433mhz-597295>

# NAPOJENÍ NA SYSTÉM DOMOTICZ

Způsobu jak ovládat bezdrátová zařízení na 433MHz ze systému domoticz je více. Nejjednodušší jsou řešení, která mají přímou podporu pro domoticz. Tj. Použití kontrolérů RFXcom nebo RFLink. Druhá možnost je použití externích programů spolu s možností v domoticzu vytvořit virtuální (tzv. dummy zařízení) např. vypínač a jemu pak nastavit script, který se má spustit při sepnutí a script, který se má spustit při vypnutí. Dotyčný script může být např. spuštění programu „script://sudo codesend 16234“, a tím se může sepnout např. bezdrátová zásuvka. Nevýhoda je že tento způsob neřeší příjem.

Další možnosti může být použití webového programu od Seahu, kde si z nabídky podporovaných zařízení dotyčné zařízení vyberete, nastavíte mu kód (nebo kód naskenujete), zařízení pojmenujete pro účely použití v systému domoticz a tlačítkem přidat do domoticzu ho necháte v domoticzu vytvořit. Tento webový program zařízení v domoticzu vytvoří podobně jako v předcházející možnosti, jen trochu pohodlněji. Navíc takto vytvořené virtuální zařízení mají On-Off scripty nastavené tak, že veškeré změny prochází tímto webovým programem který si je ukládá a pravidelně je obnovuje znovu vysláním. To má tu výhodu, že např. pokud se zařízení nějakým nedopatřením nesepe při prvním vyslání sepne se při dalším. Další výhodou je v tom že tento webový program řeší i příjem dat z přijímače.

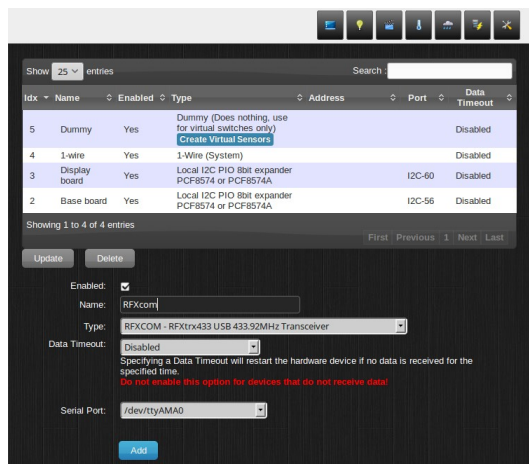
## 1. Nativní podpora kontrolérů RFXcom a RFLink

### Použití USB kontroléru RFXcom

Určitě nejlepší návody naleznete přímo na stránkách výrobce,:

RFXcom: <http://www.rfxcom.com>

Ale v kostce v menu → Setup → Hardware vyhledat dotyčný hardware (tj. „RFXCOM -RFXtrx433 USB 433,92MHz Transceiver“) viz. obr.



RFXcom

Dle svého uvážení pojmenovat (teoreticky si těchto stejných kontrolérů můžete přes USB připojit více, pak se pojmenování hodí). Dále je potřeba vybrat sériový port na kterém komunikují. Pravda je, že kontrolér připojíte pomocí USB kabelu, ale ve skutečnosti se tento kontrolér tváří jako USB sériový port, proto ten výběr sériového portu.

Show 25 entries Search:

Idx	Name	Enabled	Type	Address	Port	Data Timeout
6	RFLink	Yes	RFXCOM - RFXtrx433 USB 433.92MHz Transceiver Version: 0 <a href="#">Set Mode</a>		/dev/ttyAMA0	Disabled
5	Dummy	Yes	Dummy (Does nothing, use for virtual switches only) <a href="#">Create Virtual Sensors</a>			Disabled
4	1-wire	Yes	1-Wire (System)			Disabled
3	Display board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-60	Disabled
2	Base board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-56	Disabled

Showing 1 to 5 of 5 entries

Dále si můžete vybrat mód, což u RFXcom znamená, které zařízení či protokoly chcete aby kontrolér rozeznával.

Device: RFLink

☐ Keelogg  
☐ Home Comfort  
☐ undec on  
☐ ImaginTronix  
☐ Byron SX  
☐ Conrad RSL  
☐ Lighting4  
☐ FineOffset/Viking  
☐ Rubicon  
☐ AE/Blyss

☐ Blind T1/T2/T3/T4  
☐ Blind T0  
☐ ProGuard  
☐ FS20/Legrand CAD  
☐ LaCrosse  
☐ Hideki/UPM  
☐ AD/LightwaveRF  
☐ Mertik

☐ Visonic  
☐ ATI remote  
☐ Oregon Scientific  
☐ Meiantech  
☐ HomeEasy EU  
☐ AC  
☐ ARC  
☐ X10

[Default](#) [Set Mode](#)

[Upgrade Firmware](#)

Pokud kontrolér detekuje vysílání nějakého zařízení a rozpozná ho, tak ho domoticz automaticky přiřadí do seznamu zařízení v menu Setup → Devices, kde si ho můžete pojmenovat a označit k používání.

## Použití USB kontroléru RFLink

Doporučuji prostudovat stránky projektu. Nicméně RFLink a RFXcom jsou si hodně podobná zařízení, s podobným nastavením.

RFLink: <http://www.nemcon.nl/blog2/>

Pokud se vám nechce studovat tak kostce: v menu → Setup → Hardware vyhledat dotýčný hardware (tj. „RFLink Gateway USB“) viz. obr.

Show 25 entries Search:

Idx	Name	Enabled	Type	Address	Port	Data Timeout
5	Dummy	Yes	Dummy (Does nothing, use for virtual switches only) <a href="#">Create Virtual Sensors</a>			Disabled
4	1-wire	Yes	1-Wire (System)			Disabled
3	Display board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-60	Disabled
2	Base board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-56	Disabled

Showing 1 to 4 of 4 entries

[Update](#) [Delete](#)

Enabled: ☒

Name:

Type:

Data Timeout:   
Specifying a Data Timeout will restart the hardware device if no data is received for the specified time.  
 Do not enable this option for devices that do not receive data!

Serial Port:

[Add](#)

RFLink



Dle svého uvážení pojmenovat, vybrat USB sériový port na kterém komunikují a kontrolér přidat.

Idx	Name	Enabled	Type	Address	Port	Data Timeout
6	RFLink	Yes	RFLink Gateway USB Version: <a href="#">Create RFLink Devices</a>		/dev/ttyAMA0	Disabled
5	Dummy	Yes	Dummy (Does nothing, use for virtual switches only) <a href="#">Create Virtual Sensors</a>			Disabled
4	1-wire	Yes	1-Wire (System)			Disabled
3	Display board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-60	Disabled
2	Base board	Yes	Local I2C PIO 8bit expander PCF8574 or PCF8574A		I2C-56	Disabled

Showing 1 to 5 of 5 entries

A to je v podstatě vše. Na rozdíl od RFXcom zde není na výběr, která zařízení a protokoly chceme rozeznávat. RFLink se snaží automaticky rozeznat vše co umí. Pokud rozpozná nějaké zařízení domoticz si ho automaticky přidá do seznamu zařízení v menu Setup → Devices. Navíc nabízí tlačítko pro přidání zařízení. To je taková nouzovka, když se vám ztratí nebo rozbije dálkový ovládač a tím pádem nelze zařízení automaticky vytvořit na základě snímání. Pro přidání takového zařízení je však potřeba zadat příkaz ve správném tvaru spolu s adresou zařízení, více na stránkách projektu.

Manual Create RFLink devices

Command:

OK Cancel

## Použití modifikované verze RFLinku pro přímý běh raspberryPi

Tato verze funguje na raspberryPi jako síťová služba standardně na portu 5050. Místo komunikace na sériovém portu probíhá komunikace po síťovém portu.

### Instalace na raspberryPi

předpokladem je funkční distribuce raspbian. Pak otevřete terminál s příkazovou řádkou a pokračujte dle následujících kroků:

```
Shell:~$ sudo apt-get install wiringpi git
...
Shell:~$ git clone git://github.com/seahu/rflink
...
Shell:~$ cd rflink/Rpi_rflink
...
Shell:~$ make
...
Shell:~$ sudo make install
...
Shell:~$ sudo autostart_on
...
```

tím je dokončena instalace v /etc/rflink.conf si případně upravte číslo portu a čísla pinu TX a RX (tj. čísla pinu na kterých máte napojený datový vodič přijímače a vysílače. Čísla pinu je potřeba zadávat podle

číslování z wiringpi, přehledný popis číslování dostanete zadáním příkazu `gpio readall`. Pak už stačí službu jen spustit.

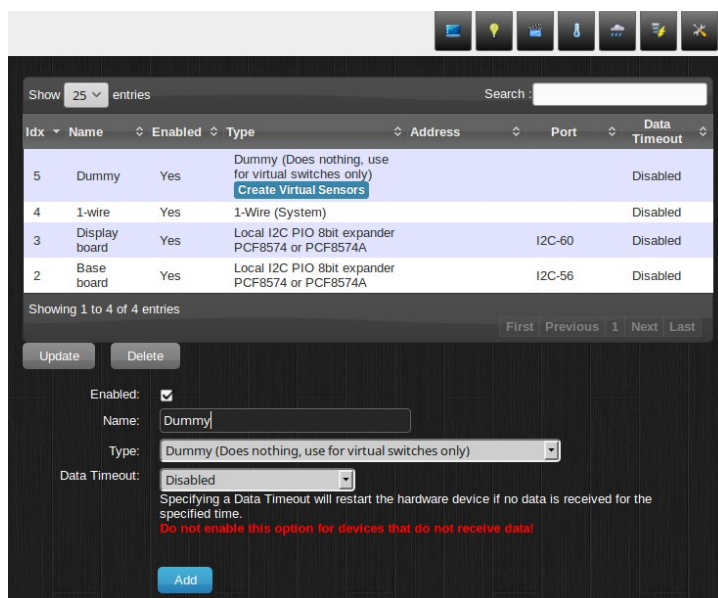
```
Shell:~$ sudo /etc/init.d/start
```

### Použití v domoticzu

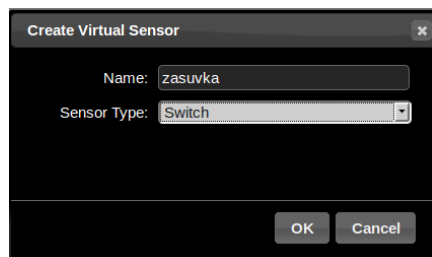
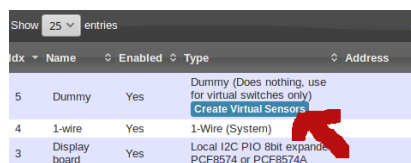
Je stejné jako u předchozí verze RFLinku jen při přidávání hardwaru (menu Setup/Hardware) je potřeba vybrat ze seznamu podporovaných zařízení RFLink Gateway with LAN interface, do kolonky remote address zadat localhost a do portu stejný port jako v konfiguraci (/etc/rflink.conf).

## 2. Použití Dummy (virtuálního) zařízení a externího scriptu (ukázka ovládaní bezdrátové zásuvky)

Nejdříve je potřeba v domoticzu vytvořit pokud ještě nemáte Dummy (virtuální) hardware: menu Setup → Hardware v kolonce type: vybrat Dummy (Does nothing, use for virtual switches only) název (name:) může být libovolný.



Poté u nově vytvořeného hardwaru klikněte na tlačítko „Create Virtual Sensors“ a vytvořte dummy (virtuální switch) dle následující obrázku.



Vyberte typ senzoru „Switch“ libovolně jej pojmenujte a potvrďte OK. Tím je dummy přepínač vytvořen. Zbývá nastavit script, který se má spustit při zapnutí a script při vypnutí.

Jako script použijí program codesend který jsem popisoval výše viz. (UTILITY Z NINJABLOCKS ). Jenže za codesend je potřeba uvést kód, který se má odvíjet a ten je potřeba nejdříve zjistit. To zjistíme programem RFSniffer (s též sady utilit). Oba programy je potřeba spouštět jako superuživatel tj. buď jako root nebo pomocí programu sudo. Nejdříve si naskenujeme kódy tlačítek z dálkového ovládače bezdrátové zásuvky, které zásuvku zapínají a vypínají:

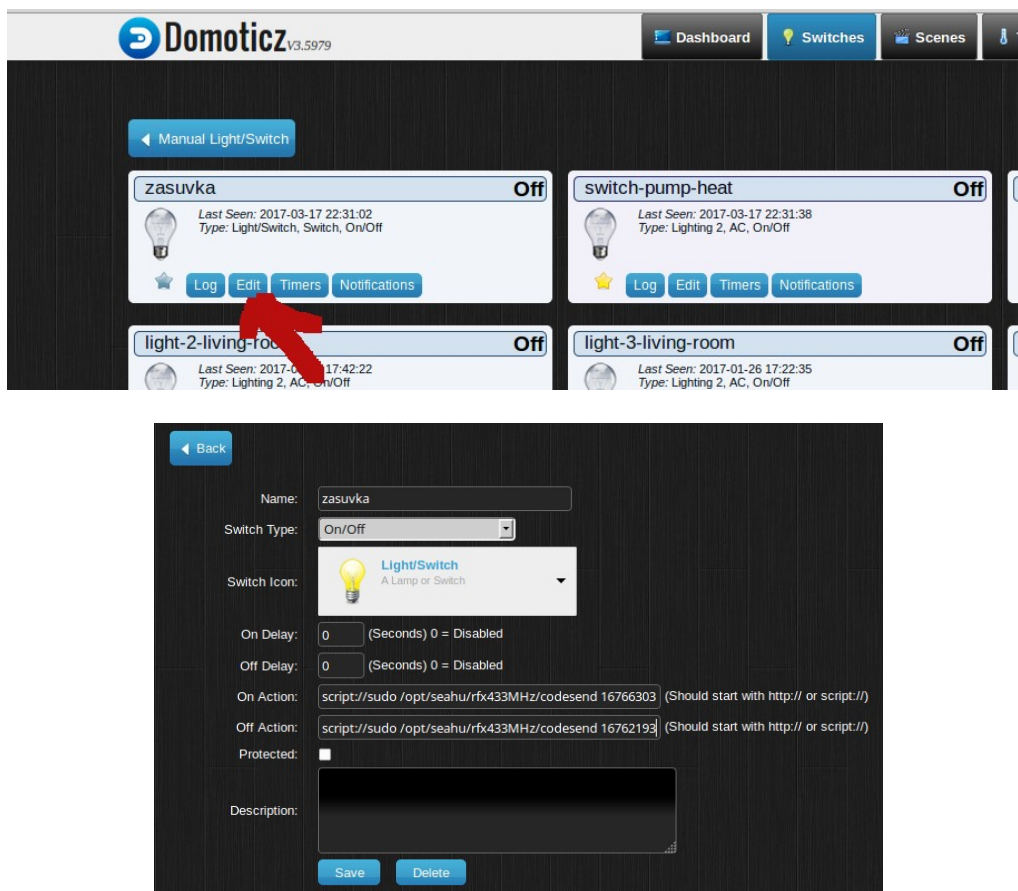
```
Shell:~$ sudo /opt/seahu/rfx433MHz/RFSniffer
Received 16766303
Received 16762193
```

a pomocí kombinace kláves ctrl+C skenování ukončíme.

Že jsme kódy dobře naskenovali si můžeme ověřit jejich odvíjetím, mělo by to zásuvku ovládat.

```
Shell:~$ sudo /opt/seahu/rfx433MHz/codesend 16766303
sending code[16766303]
Shell:~$ sudo /opt/seahu/rfx433MHz/codesend 16762193
sending code[16762193]
```

A právě tyto příkazy vložíme do konfigurace dummy zásuvky. Menu Switches → nově vytvořený switch → Edit .



A samozřejmě uložíme (save). Od teď by již měl přepínač v domoticzu fungovat. Pokud ne podívejte se do spouštěcího scriptu domoticzu pod kterým uživatelem běží (/etc/init.d/domoticz.sh řádek začínající na USERNAME=). Tento uživatel by měl mít možnost spouštět program „codesend“ pomocí sudo bez zadání hesla. To se nastavuje v souboru /etc/sudoer , kde stačí přidat řádek:

```
uživatel_pod_kterým_beží_domoticz ALL=NOPASSWD: /opt/seahu/rfx433MHz/codesend
```

PS: na raspberryPI pro uživatele „pi“ toto není potřeba, ten už má nastavené, že může spouštět cokoli bez zadávání hesla.

Nutno ještě upozornit, že k vysílání kódu dojde jen při změně stavu přepínače, a pokud v té době v blízkém okolí vysílá ještě někdo jiný (klidně i další čidlo), tak zásuvka na vysílání nemusí zareagovat. Tj. Nikdy si nemůžete být jisti že stav v domoticzu odpovídá realitě.

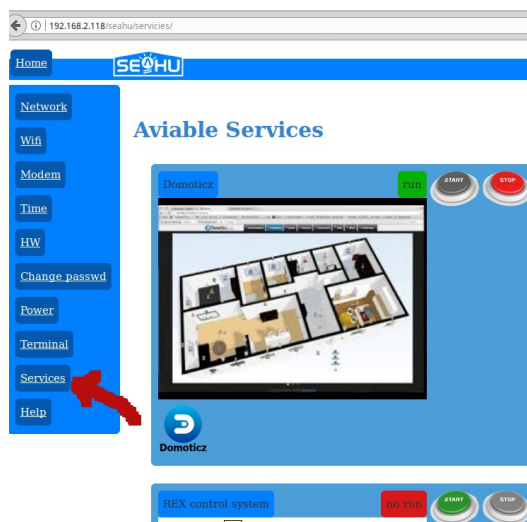
## SEAHU WEBOVÝ PROGRAM JAKO PROSTŘEDNÍK MEZI RFX433 A DOMOTICZKEM

protože výše popsané řešení je příliš komplikované nabízí Seahu jednoduchý PHP program, který to nakonfiguruje za vás a to ještě takovým způsobem, že řeší i příjem vysílání a pravidelnou obnovu stavu registrovaných zařízení (takže i když ne 100%, ale i tak mnohem více můžete důvěřovat, že stav v domoticzu odpovídá realitě). Toto řešení je instalované v řídicím modul Seahu SH017, proto zde nebudu rozebírat jak jej nainstalovat.

Dále musím upozornit, že aktuálně je zde podpora jen bezdrátových zásuvek fi. Kanlux (k dostání v Hornbachu) a obecného spínače a tlačítka.

Nejprve je potřeba službu která toto zastřešuje aktivovat. Zadejte do prohlížeče IP adresu vašeho Seahu SH017 modulu a dostanete se do konfigurace. Přihláste se (pokud jste heslo ještě neměnili tak výchozí heslo je „raspberry“)

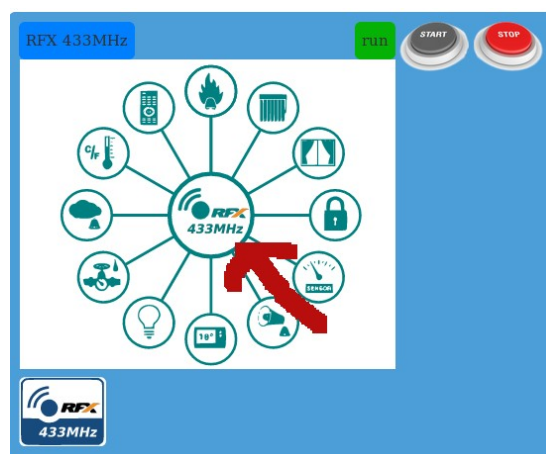
V menu „Services“ zapněte službu RFX433MHz dle následujících obrázků:



*Výběr menu Services*



*Tlačítkem Start aktivovat službu*



*Kliknutím na schématický obrázek služby se dostanete do její konfigurace*



Následuje samotná konfigurace služby. V stlačení tlačítka „Add new device“ se dostanete do formuláře, kde si nejdříve vyberete výrobce zařízení a poté konkrétní typ podporovaného zařízení. To se vám pak zobrazí v seznamu a s něho ho pak můžete přejmenovat (pro účely pojmenování v domoticzu), nastavit mu kódy pomocí DIP přepínače nebo naskenováním, odzkoušet funkcionalitu a jedním tlačítkem přidat do domoticzu. Aby se aktualizovaly stavy tlačítek např. po zmáčknutí na dálkovém ovládači je potřeba stránku aktualizovat. Více viz. následující obrázky:

The screenshot shows the 'Add new device' screen in the SEAHU web interface. It features a blue header with the SEAHU logo and a '433MHz' indicator. Below the header is a 'Add new device' button. The main content area displays three device categories:

- Kanlux APO TM3 socket:** Includes a DIP switch configuration (ON/OFF) and buttons for 'Set code', 'Set Name', 'Add Domoticzu', and 'delete'.
- Kanlux APO TM3 remove control:** Includes a list of buttons (A, B, C, D, E) and buttons for 'Set code', 'Set Name', 'Add Domoticzu', and 'delete'.
- GENERAL general switch:** Includes a 'General Switch ON OFF' configuration and buttons for 'Set code', 'Set Name', 'Add Domoticzu', and 'delete'.

### Ukázka nastavení kódu pomocí DIP přepínače

This screenshot shows the 'Add new device' screen with the 'Start scan ON' and 'Start scan OFF' buttons highlighted. The interface is similar to the previous one, but the 'Set code' buttons are now active, indicating that the code has been successfully scanned and set for the device.

### Ukázka naskenování kódu

This screenshot shows the 'Add new device' screen with the 'Set code' buttons highlighted. The interface is similar to the previous ones, but the 'Set code' buttons are now active, indicating that the code has been successfully scanned and set for the device.

### Ukázka pojmenování tlačítek

## Více informací:

- web raspberryPI projektu: <https://www.raspberrypi.org/>
- web seahu: <http://www.seahu.cz>
- čerpaná inspirace z:
  - <http://www.rfxcom.com>
  - [http://homeeasyhacking.wikia.com/wiki/Home\\_Easy\\_Hacking\\_Wiki](http://homeeasyhacking.wikia.com/wiki/Home_Easy_Hacking_Wiki)
  - <http://shop.ninjablocks.com/blogs/how-to/7506204-adding-433-to-your-raspberry-pi>
  - <https://www.domoticz.com/forum/viewtopic.php?t=7156>
  - [http://www.princetronics.com/how-to-read-433-mhz-codes-w-raspberry-pi-433-mhz-receiver/#Installing\\_433Utils](http://www.princetronics.com/how-to-read-433-mhz-codes-w-raspberry-pi-433-mhz-receiver/#Installing_433Utils)
  - <https://tutorials-raspberrypi.com/let-raspberry-pis-communicate-with-each-other-per-433mhz-wireless-signals/>
  - <https://www.raspberrypi.org/forums/viewtopic.php?f=37&t=66946>
  - <https://www.root.cz/clanky/arduino-na-433-mhz-komunikuje-s-bazenovym-teplomerem/>
  - <https://www.hackster.io/roccomuso/iot-433mhz-597295>

## Napsal:

Ing. Ondřej Lyčka duben 2017

Verze dokumentu: 2.01