

ioBroker - Integration von INKBIRD Sensoren mit 433 MHz RF (Poolsensor IBS-P01R und Temperatur/Feuchtesensor ITH-20R)

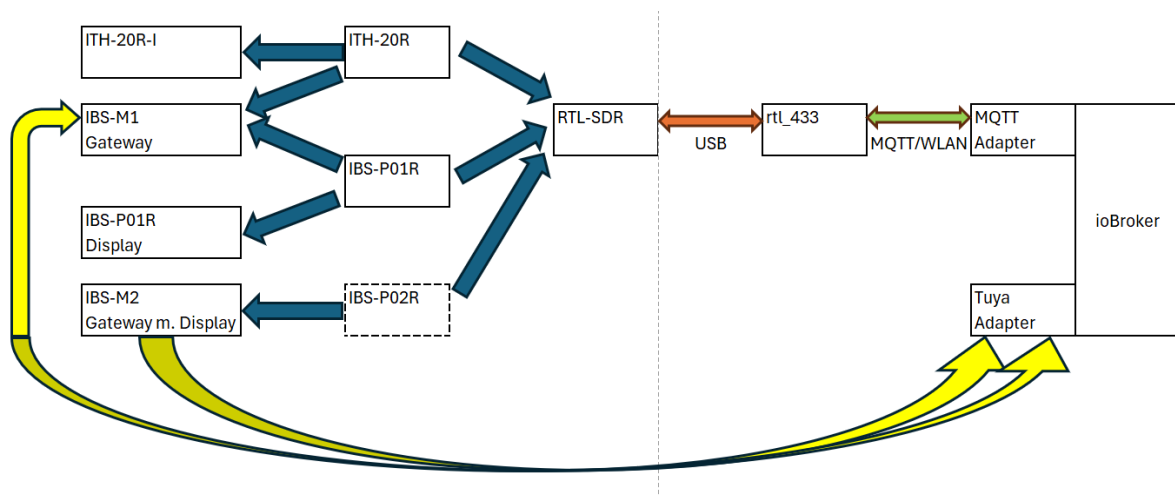
Es gibt bereits eine Integration für die beiden o.g. INKBIRD Sensoren über die INKBIRD Gateways IBS-M1 und IBS-M2 unter Verwendung des Tuya-Adapter im ioBroker. Diese Integration erlaubt z.B. auch die Einbindung von Bluetooth – Sensoren wie IBS-P01B. Die gleichzeitige Verbindung der Gateways mit der INKBIRD – Cloud und Nutzung der entsprechenden Apps ist nicht möglich. Das Verfahren ist hier beschrieben <https://github.com/jurudolph/Inkbird>

Für die o.g. INKBIRD Sensoren mit 433 MHz RF gibt es eine Möglichkeit das RF Signal direkt zu verarbeiten und in ioBroker zu integrieren, ohne INKBIRD Gateway.

Beide Möglichkeiten habe ich im unteren Bild betrachtet. Im linken Teil ist die INKBIRD-„Welt“ dargestellt. Beispielhaft sind hier ein ITH-20R (Temperatur/Feuchte-Sensor) und zwei Poolsensoren IBS-P01R und IBS-P02R abgebildet. Diese verbinden sich über 433 MHz RF (blaue Pfeile) mit den IBS-M1 und IBS-M2 Gateways, sowie mit den Displays ITH-20R-I und IBS-P01R Display.

Bei der ersten Methode verbinden sich die IBS-M1 / IBS-M2 Gateways per WLAN (gelbe Pfeile) mit dem Tuya-Adapter.

Bei der zweiten Methode empfängt ein RTL-SDR (<https://www.rtl-sdr.com/>) die 433 MHz RF Signale der Sensoren direkt. Der RTL-SDR ist über USB (roter Pfeil) an einen Rechner (Linux/Windows/MacOS/Docker) angeschlossen, auf dem rtl_433 (https://github.com/merbanan/rtl_433) läuft. rtl_433 kann mehr als 250 verschiedene RF Protokolle dekodieren (auch INKBIRD) und verschiedenen Output liefern. Das Ausliefern als MQTT-Client (grüner Pfeil) ist einfach und kann den MQTT – Adapter (Broker/Server) am ioBroker bedienen.



Ich habe das Verfahren für die Sensoren ITH-20R und Poolsensor IBS-P01R getestet. Für IBS-P02R gibt es noch keine Testergebnisse. Für die Einbindung des IBS-P03R gibt es ein Github Issue #3003, wobei zu beachten ist, dass der IBS-P03R auf 868 MHz funkt..

Das Ergebnis im ioBroker sieht beispielsweise dann so aus: Die Device ID 40285 zeigt einen ITH-20R mit externem Sensor und die Device ID 59012 zeigt einen Poolsensor IBS-P01R.

Inkbiird-ITh2DR	Inkbiird-ITh2DR								
40285	40285								
battery_ok	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/battery_ok	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 19:07:33.556	0,79		
freq1	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/freq1	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	433,94374		
freq2	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/freq2	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	433,90307		
humidity	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/humidity	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	72,8		
id	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/id	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	40285		
mic	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/mic	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-16 17:57:03.490	CRC		
mod	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/mod	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-16 17:57:03.501	FSK		
noise	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/noise	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	-23,2191		
protocol	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/protocol	mqtt://admin	0x00	good	2024-07-23 20:00:41.277	2024-07-16 17:57:03.490	194		
rsi	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/rsi	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	-0,267014		
sensor_num	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/sensor_num	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-16 17:57:03.490	2		
snr	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/snr	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 20:00:41.319	22,95213		
temperature_2_C	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/temperature_2_C	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-22 22:07:49.404	22,1		
temperature_C	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/temperature_C	mqtt://admin	0x00	good	2024-07-23 20:00:41.319	2024-07-23 08:58:37.977	22		
time	rtl_433/raspberrypi/devices/inkbird-ITh2DR/40285/time	mqtt://admin	0x00	good	2024-07-23 20:00:41.277	2024-07-23 20:00:41.277	2024-07-23 20:00:39.308268		
59012	59012								
battery_ok	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/battery_ok	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-16 18:27:27.508	0,6		
freq1	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/freq1	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-23 20:03:07.654	433,92893		
freq2	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/freq2	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-23 20:03:07.654	433,90486		
humidity	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/humidity	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-16 18:27:27.508	130		
id	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/id	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-16 18:27:27.508	59012		
mic	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/mic	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-16 18:27:27.508	CRC		
mod	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/mod	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-16 18:27:27.508	FSK		
noise	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/noise	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-23 20:03:07.654	-21,7204		
protocol	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/protocol	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-16 18:27:27.508	194		
rsi	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/rsi	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-23 20:03:07.654	-10,9899		
sensor_num	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/sensor_num	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-16 18:27:27.508	3		
snr	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/snr	mqtt://admin	0x00	good	2024-07-23 20:03:07.654	2024-07-23 20:03:07.654	10,73049		
temperature_2_C	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/temperature_2_C	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-16 18:27:27.508	130		
temperature_C	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/temperature_C	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-23 19:28:28.846	28,1		
time	rtl_433/raspberrypi/devices/inkbird-ITh2DR/59012/time	mqtt://admin	0x00	good	2024-07-23 20:03:07.653	2024-07-23 20:03:07.653	2024-07-23 20:03:06.615823		

Technische Umsetzung:

Grundsätzlich ist es möglich rtl_433 auf dem ioBroker Server laufen zu lassen (wie beispielsweise auch beim RTL_433 Adapter von ioBroker https://github.com/phillipssc/ioBroker.rtl_433) und den RTL-SDR über USB daran anzuschließen. In meinem Fall befindet sich der ioBroker im Keller (schlecht für eine Antenne). Daher läuft bei mir rtl_433 auf einem alten Raspberry Pi 1, den ich für Funk optimal positionieren kann. Der Stromverbrauch des Raspi inkl. RTL-SDR liegt bei 3 W.

Nebeneffekte:

Der RTL-SDR fängt natürlich nicht nur die Signale der INKBIRD RF Sensoren auf. Ich empfangen z.B. auch Daten von meiner alten ELV WS 2000 Wetterstation, TFA Temperatur/Feuchte-Sensoren, Intertechno Fernbedienungen, usw. Durch sinnvolle Konfiguration des rtl_433 können unerwünschte Geräte (z.B. von den Nachbarn) ausgefiltert werden.

Da die Gateways nun nicht mehr für die Übertragung zum ioBroker benötigt werden, können sie bei Bedarf wieder mit der INKBIRD Cloud und der App verbunden werden.

Schritt-für-Schritt-Anleitung (TBD)

1. RTL-SDR mit rtl_433 zum Testen installieren
2. ioBroker MQTT Broker/Server Adapter installieren und konfigurieren
3. RTL-SDR mit rtl_433 und MQTT Client einrichten
4. RTL-SDR mit rtl_433 und MQTT Client auf Raspberry Pi installieren, konfigurieren und als Service einrichten
5. Protokollfilter einrichten (optional)