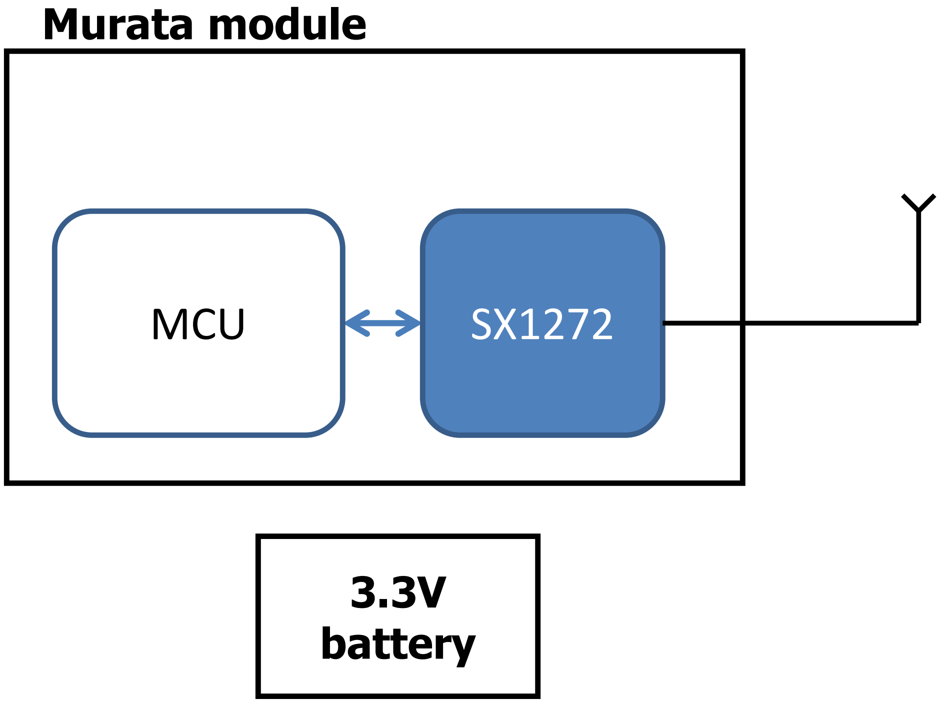
SEMTECH

www.semtech.com

LoRaWAN Relay workshop

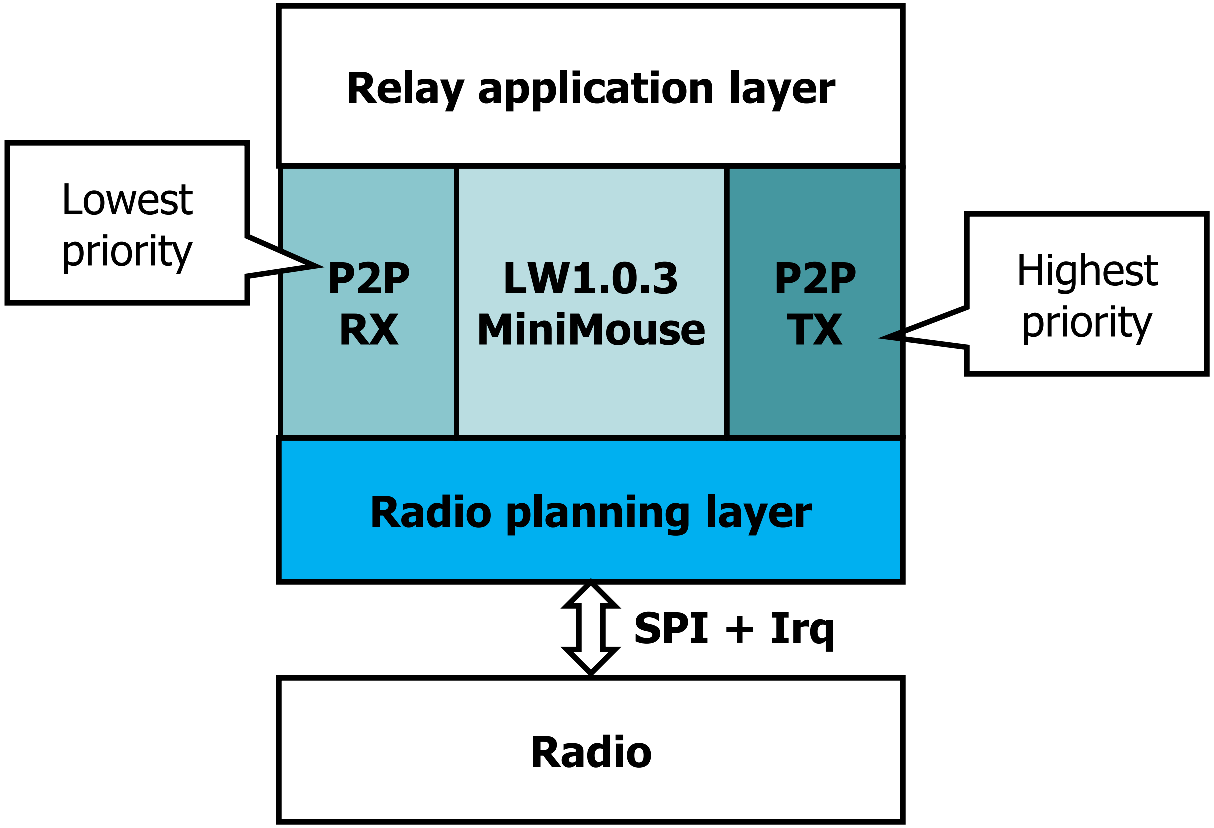
Nsornin & FHolin / Jan-2019

Block diagram



Given the very low average current consumption of the relay , energy harvesting may make sense to make it totally maintenance-free

Relay Software stack



Relay application layer

Lowest priority

P2P RX

LW1.0.3 MiniMouse

P2P TX

Highest priority

Radio planning layer

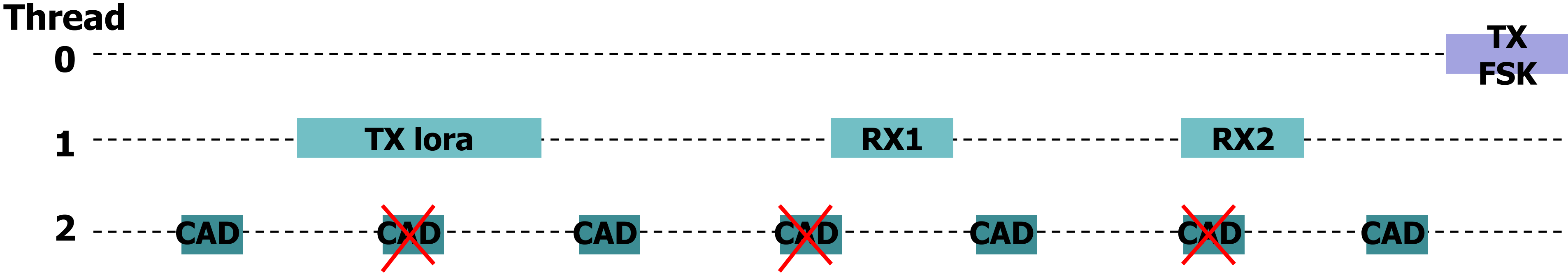
SPI + Irq

Radio

Radio planning layer

- ❑ **Written in C++**
- ❑ **Replaces radio driver**
- ❑ **Allows several threads to use a common radio resource , managing real time, priorities and collisions.**
- ❑ **API exposed to thread is : schedule_radio(modulation, TX/RX, time, duration)**
- ❑ **Threads can be developed and validated independently.**

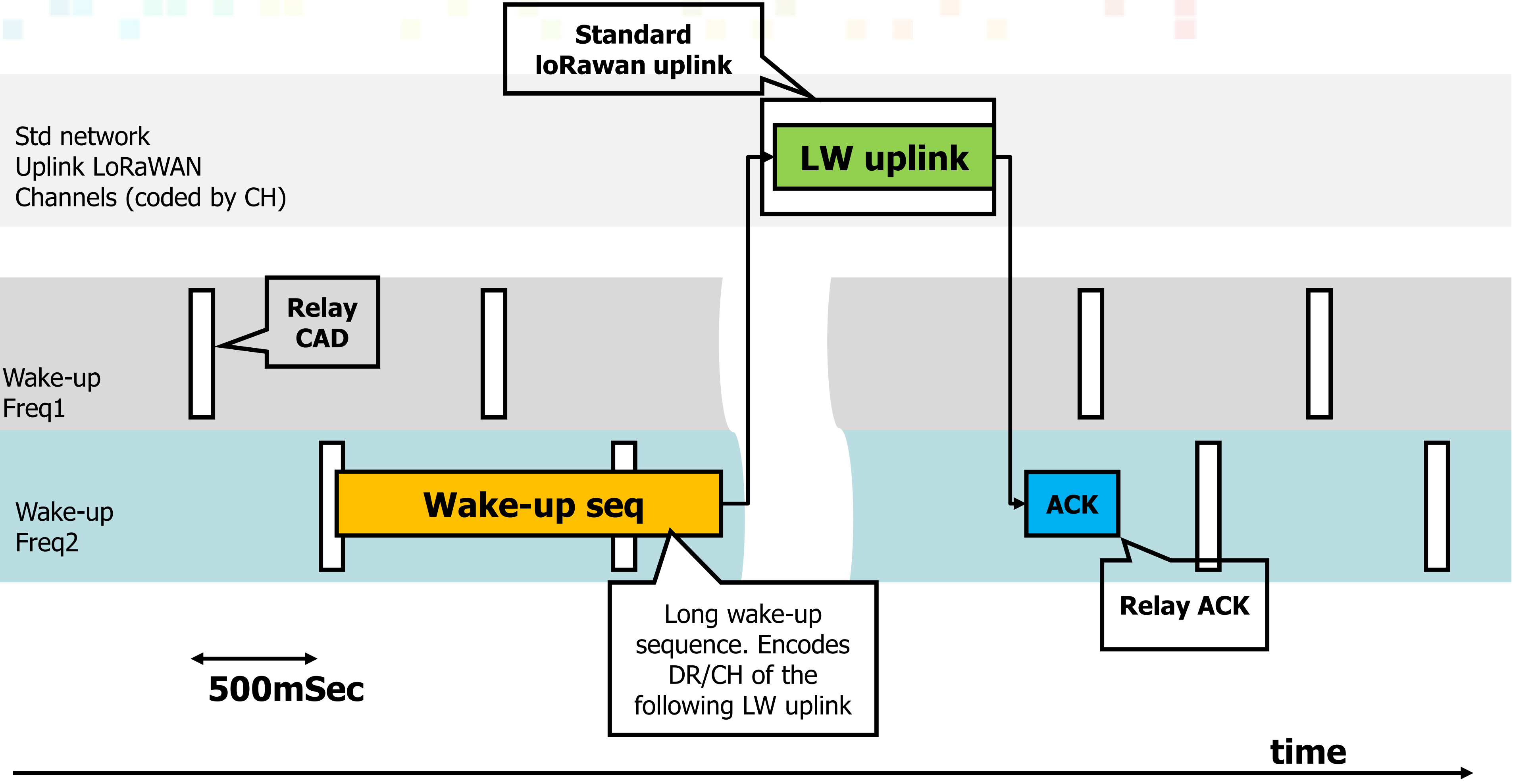
Example: In our relay the P2P stack and the LoRaWAN stacks never have to interact. The LoRaWAN stack is a standard stack that as not been altered in any way.



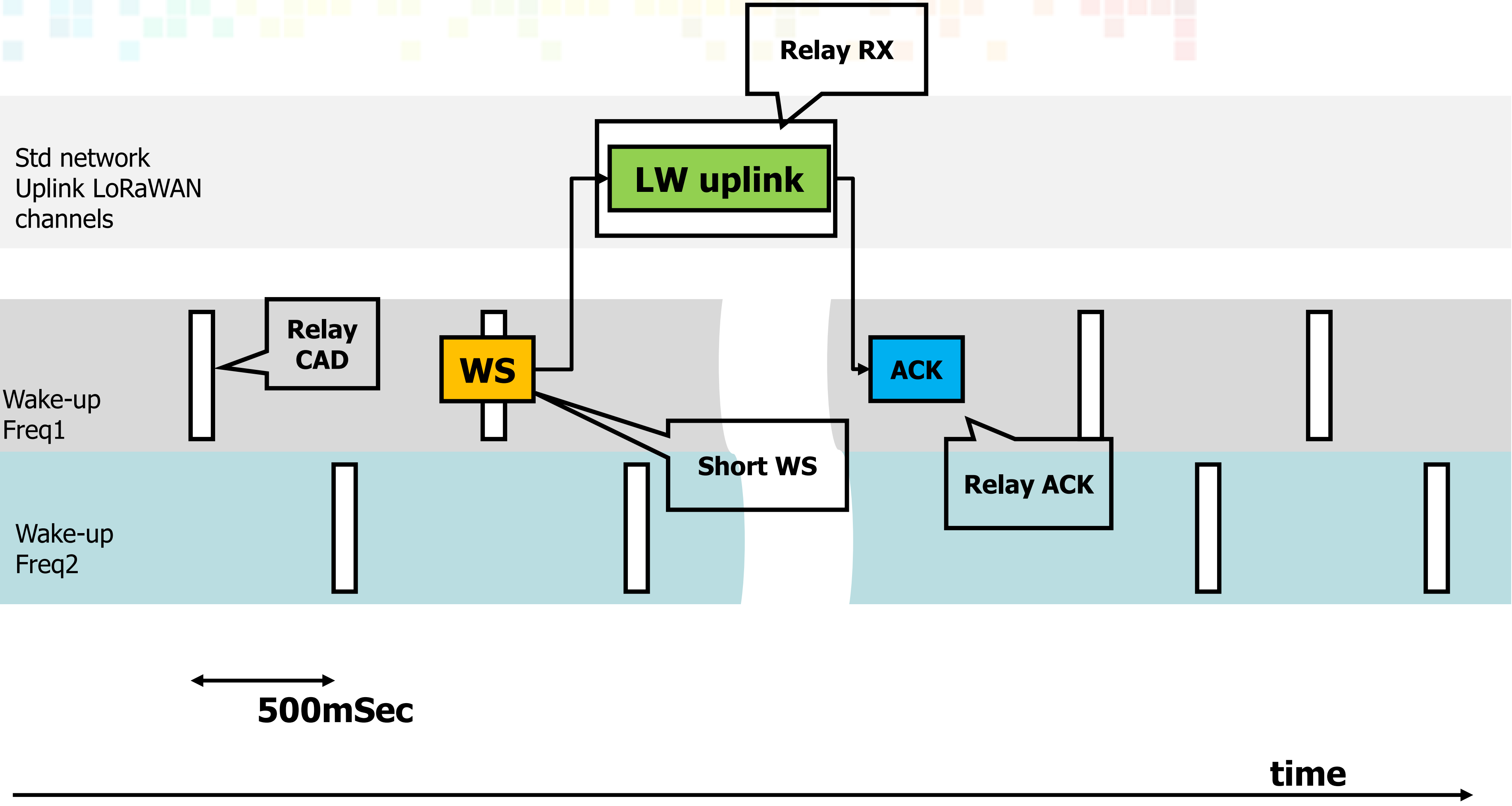
Principle

- ❑ **Relay alternatively scans (CAD) 2 wake-up channels every 500mSec**
- ❑ **End-devices under relay**
  - Prefix their LW uplinks with a wake-up sequence transmitted on one of the 2 wake-up channels
  - Wake-up sequence is initially 1sec long
  - Can use shorter wake-up sequence once synchronized (as short as 60mSec) and depending on data rate used
- ❑ **The wake-up sequence encodes the channel & DR of the actual LoRaWAN uplink**
- ❑ **Relay filters device’s uplink using a device whitelist**
- ❑ **authorized devices’ uplinks are encapsulated and forwarded to the network**

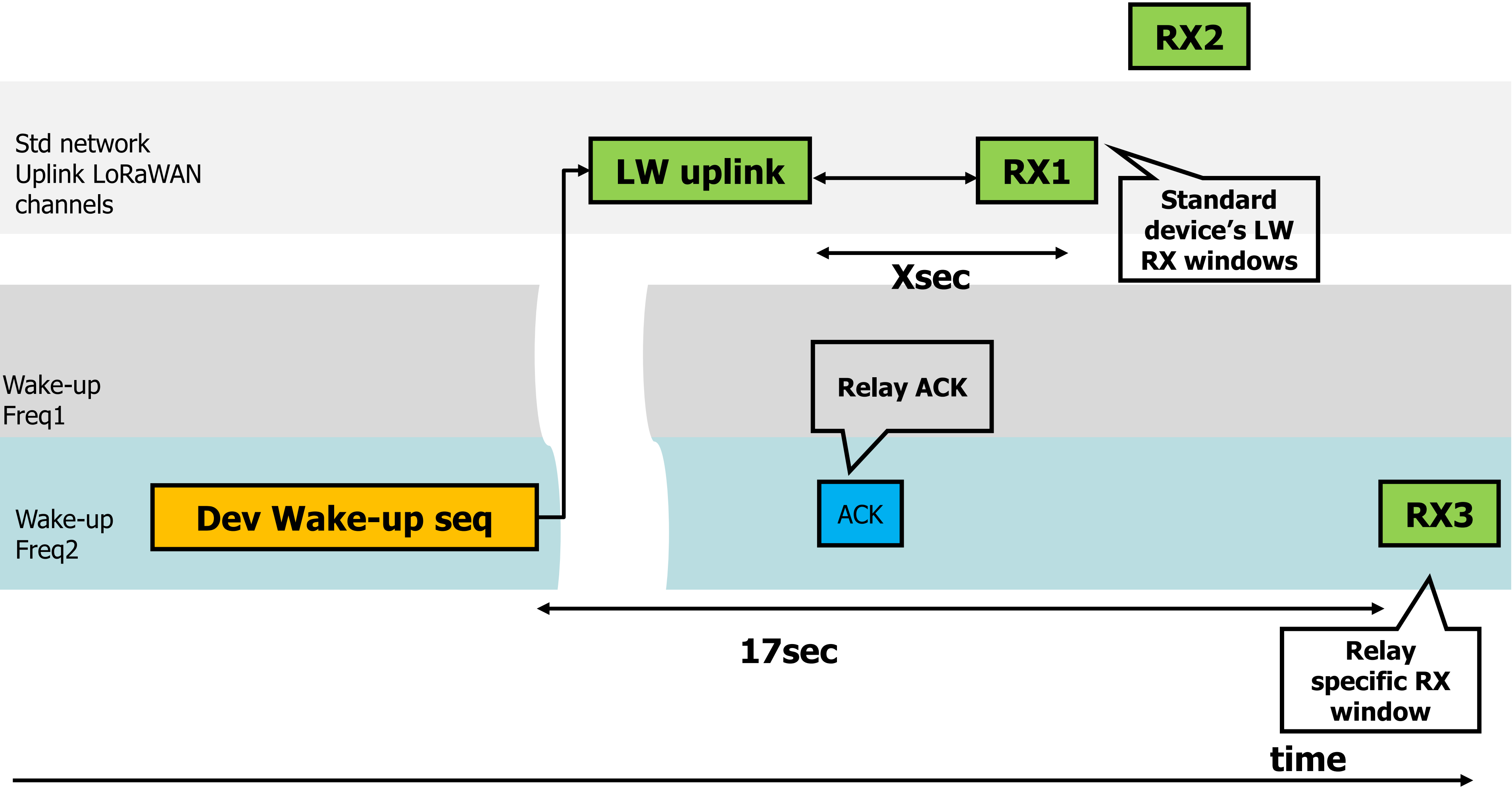
Unsynchronized Device to Relay uplink


Standard loRawan uplink
Std network Uplink LoRaWAN Channels (coded by CH)
LW uplink
Relay CAD
Wake-up Freq1
Wake-up Freq2
Wake-up seq
ACK
Long wake-up sequence. Encodes DR/CH of the following LW uplink
Relay ACK
500mSec
time

Synchronized Device to Relay uplink



Complete device2relay classA transaction



Content of forwarded uplinks

If device is whitelisted, the Relay forwards the uplink encapsulated in an Application payload to the network.

The following field are present:

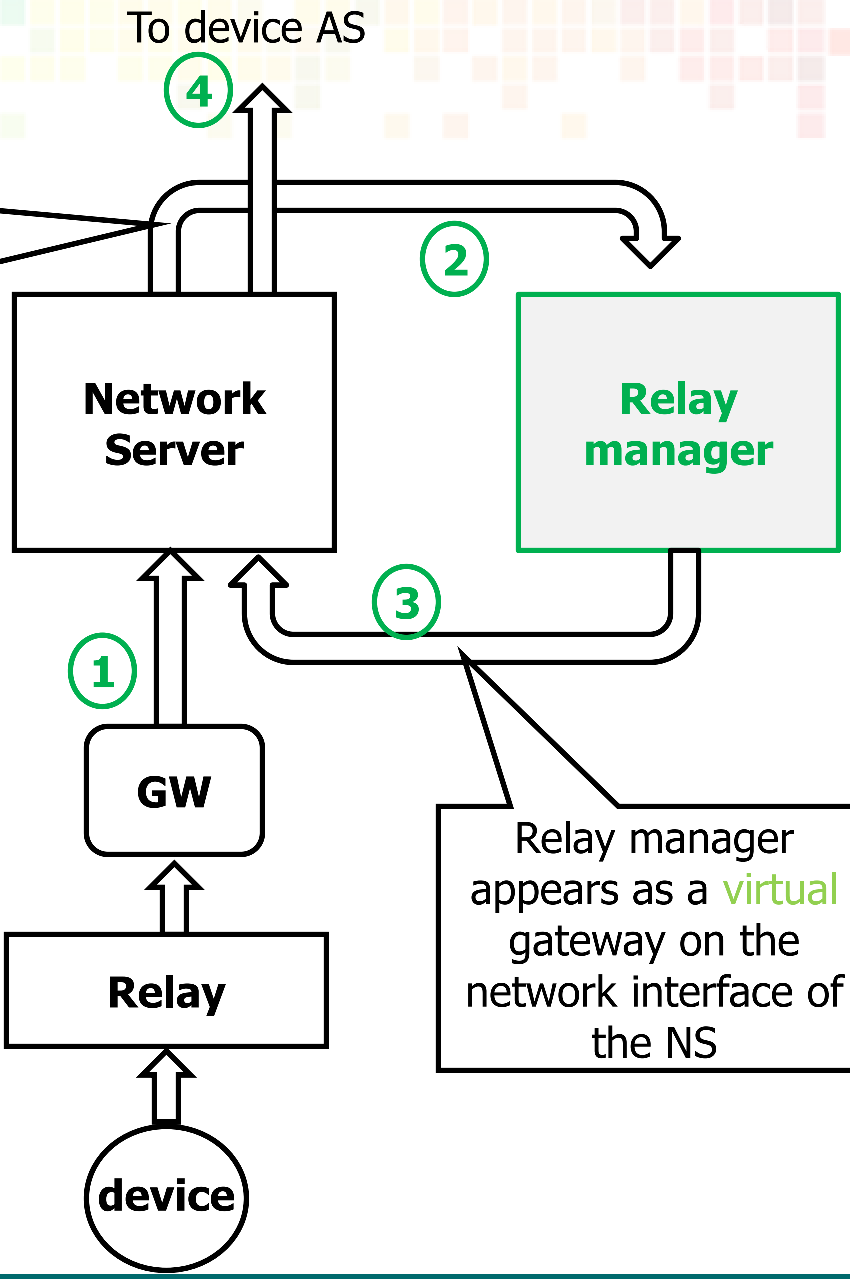
- ❑ **The full device uplink**
- ❑ **The channel on which uplink was received by the relay**
- ❑ **The RSSI/SNR of received device uplink**

Frequency diversity

- ❑ **Relay alternatively scans 2 wake-up channels**
- ❑ **Device uplinks may happen on any channel (at least 8)**
  - ▪ Also brings diversity on RX1 slot of device because RX1 freq = uplink freq
- ❑ **The LoRaWAN uplink is a *standard* LoRaWAN frame and is received by the network standard gateways (if in reach)..**

- ❑ **Keeping frequency diversity is very important to guarantee robust operation in interfered environment !!**

Transparent Network operation



Relay manager appears as a normal application server on the application interface of the NS

Relay manager appears as a virtual gateway on the network interface of the NS

Relay encapsulates device uplink frame

Relay application layer

Periodic Relay status report + remote configuration.

Relay status:

- ❑ List of devEUI trying to join with RSSI
- ❑ List of DevAddr trying to uplink with associated RSSI

Remote configuration:

- ❑ DevEUI whitelist
- ❑ DevAddr whitelist

Easy configuration through a dashboard..

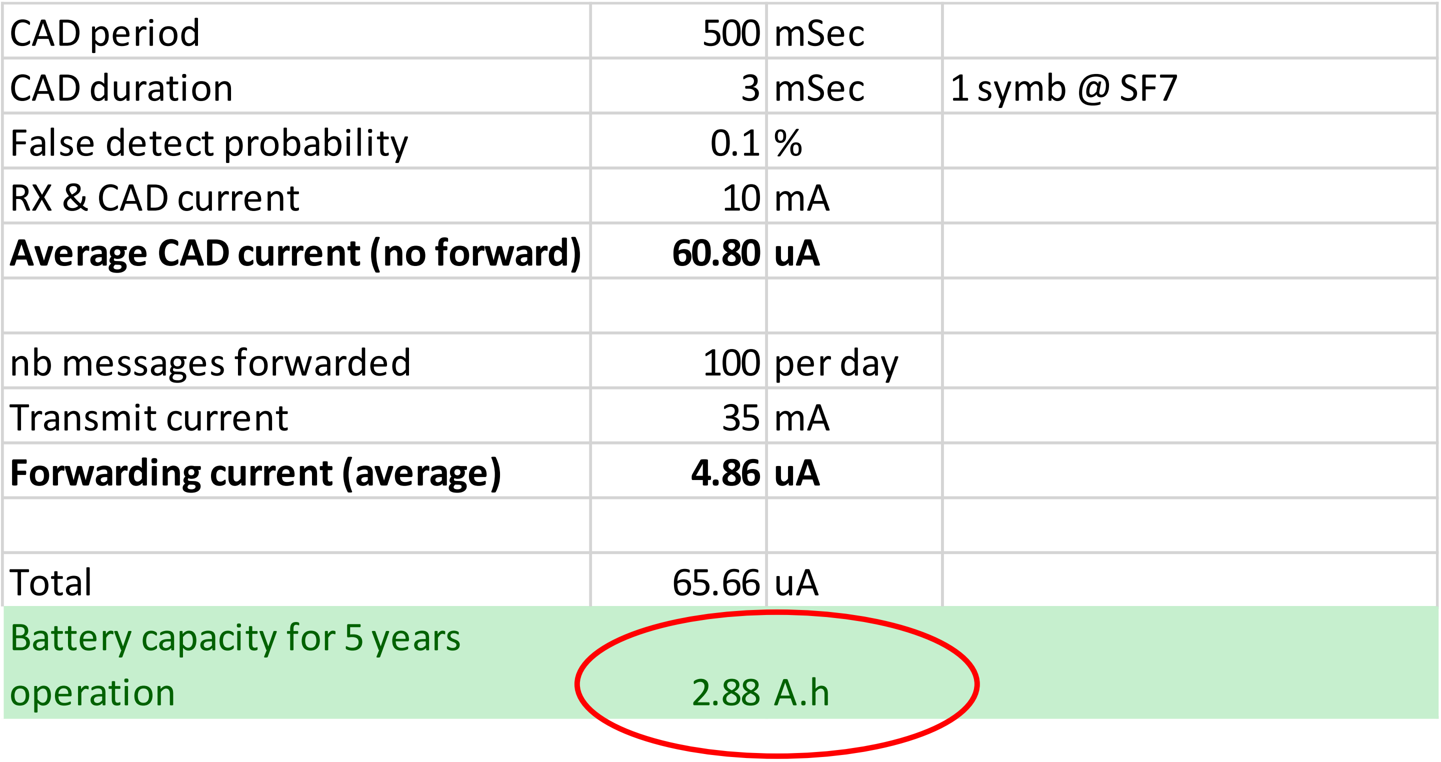
Relay application layer (missing)

On top of that we would like:

❑ **Spectral scan statistics :** useful to allocate best channels locally

❑ **Battery status:** prediction of relay battery life

CAD Relay battery life (SX1272 based)



CAD period	500	mSec	
CAD duration	3	mSec	1 symb @ SF7
False detect probability	0.1	%	
RX & CAD current	10	mA	
Average CAD current (no forward)	**60.80**	**uA**	
nb messages forwarded	100	per day	
Transmit current	35	mA	
Forwarding current (average)	**4.86**	**uA**	
Total	65.66	uA	
Battery capacity for 5 years operation	2.88	A.h	

Device battery life under relay
TX every 30min

message periodicity	30	minutes	
Transmit current	35	mA	
max clock drift	30	ppm	
Wakeup sequence length	**60**	mSec	
Total uplink time on air	**180**	mSec	assuming 30 bytes payload
average transmit current	**3.50**	uA	
RX1 & RX2 current	**0.11**	uA	
Total	**3.61**	uA	
Battery capacity for 5 years operation	**0.16**	**A.h**	

Thanks to synchronization

Device battery life under relay
TX every 360min (6h)

message periodicity	360	minutes	
Transmit current	35	mA	
max clock drift	30	ppm	
Wakeup sequence length	**1000**	mSec	
Total uplink time on air	**1120**	mSec	assuming 30 bytes paylaod
average transmit current	**1.81**	uA	
RX1 & RX2 current	**0.01**	uA	
Total	**1.82**	uA	
Battery capacity for 5 years operation	**0.08**	**A.h**	

Device stack required modification

- ❑ Prefix JoinRequest and LW uplinks with wake-up sequence
- ❑ Open short RX window immediately after LW uplink to catch RelayACK
- ❑ Open a short RX3 slot 17sec after LW transmission

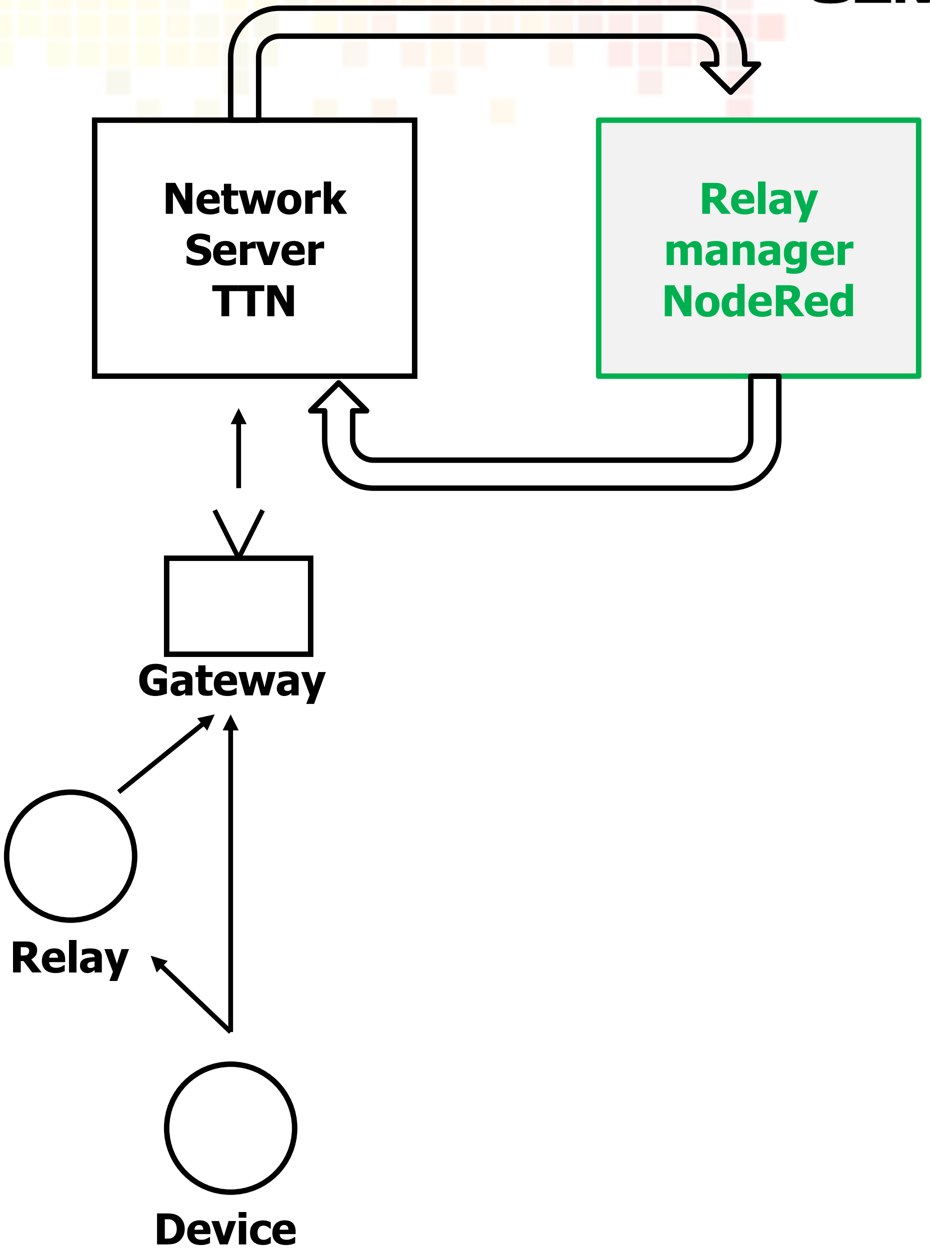
All LoRaWAN MAC commands are supported.

SEMTECH

Demo time !!

What are you going to see ?

Demo setup


Network Server TTN
Relay manager NodeRed
Gateway
Relay
Device

Backup slides

Simultaneous Relay/network operation

Problem statement: connectivity switch

1. **Devices must be able to join through gateway or relay indifferently (gateway preferred when available)**
2. **Device must fall back to relay mode automatically if gateway disappears**
3. **Device must disconnect from relay and connect though gateway when a new gateway appears**

Those 3 scenarios are addressed sequentially in the next 3 slides

1: Devices must be able to join through gateway or relay indifferently (gateway preferred when available)

Problem addressed by following process:

1. Simply transmit Wake-up sequence + joinReq until a JoinAccept is received.

The actual JoinReq LoRaWAN frame is transmitted on the default LoRaWAN network channels therefore can be received by the network gateways and relays at the same time.

Network selects whether device will operate under relay or gateway through the JoinAccept frame CFlist

2: Device must fall back to relay mode automatically if gateway disappears

Easy solution for OTA devices:

1. Device detects that it is disconnected and should rejoin (no DL for an extended period of time or nb of Uls)
2. Device goes back to Join mode and follows Join process
3. Device rejoins through a relay because no gateway is in range

3: Device must disconnect from relay and connect though gateway when a new gateway appears

Can be solved as follow:

1. Device is connected through a relay
2. Relay receives device's uplinks on the same channel than the network channels.

   Note that this channel changes with every uplink, randomly selected by device
3. Network gateway CAN receive that uplink as any other uplink
4. Network detects that device's uplinks are sometime received both though relay and through a gateway
5. Network can decide to send a MAC command to device to enable the network mode and get out of relay mode , or easier ask device to rejoin and connect it back through gateways

Standard Wake-up sequences

There are 2 types of wake-up sequences

1. **JoinRequest wake-up sequence**
2. **Standard uplink wake-up sequence**

The type of wake-up sequence is coded by the Ftype bit in the wake-up sequence header

Wake-up packets are always transmitted in implicit header mode (no PHY header) to minimize overhead.

Wake-up packet’s preamble length is a function of the spreading factor used.

Standard uplink Wake-up sequence

❑ standard Wake-up sequence is composed of 1 or several packets each carrying the following information : 11 bytes

Field	Header	DevAddr	CntDnw	Fcount	CH	MIC
Byte	1	4	1	2	1	2

❑ The CntDnw field indicates the number of micro-packet before the start of the LoRaWAN frame transmission when the wake-up sequence is composed of several wake-up packets

❑ Fcount + MIC = authentication of device / DOS attack mitigation

Header & CH fields

Header byte

Field	DR	RFU	Ftype
bits	7:4	3:1	0

Ftype = 0, standard uplink following
Ftype = 1, JoinRequest following
DR : The data rate index used for the transmission of the following LoRaWAN uplink as defined in the regional param document

CH byte

Field	RFU	ChIndex
bits	7	6:0

ChIndex : the index of the channel used for the transmission of the following standard LoRaWAN uplink as defined in the Regional Param document:
In EU , range 0 to 15
In CH , range 0 to 95
In US , range 0 to 71

MIC computation

Field	Header	DevAddr	CntDnw	Fcount	CH	MIC
Byte	1	4	1	2	1	2

cmac = aes128_cmac(RelaySKey, Header | DevAddr | CntDnw | Fcount | CH | pad16)

MIC = *cmac*[0..1]

JoinRequest Wake-up sequence

- ❑ JoinRequest Wake-up sequence is composed of 1 or several packets each carrying the following information : 11 bytes

Field	Header	JoinEUI	CntDnw	CH
Byte	1	8	1	1

- ❑ The CntDnw field indicates the number of micro-packet before the start of the LoRaWAN frame transmission when the wake-up sequence is composed of several wake-up packets. If the wake-up sequence is composed of a single packet (ex SF12) , that field is zero
- ❑ There is no MIC field for JoinRequest wake-up packets because the RelaySKey has not yet been shared between the relay and the device

Device to relay uplink

- ❑ **Device sends wake-up sequence (WS)**
- ❑ **WS contains the channel & DR that will be used for LoRawan uplink**
- ❑ **Relay receives one of the WS micro-packet, then goes back to sleep until time to receive UL**
- ❑ **Relay switches to appropriate Channel & DR to receive uplink**
- ❑ **If uplink is received without CRC error, Relay sends immediately a RelayACK containing:**
  - HASH of the uplink computed using the device’s RelaySKey (proof of delivery for the device)
  - Timing correction information : realigns device's clock to relay clock

Relay-ACK : synchronization

- The relay provides synchronization information to the device using the RelayACK frame following every successful device-to-relay transmission
- The content of the RelayACK frame is:

Field	MIC	Timing
Byte	2	2

- Timing encodes the timing difference between the last WS packet and the relay’s CAD window.
  - This information allows the device to send a shorter WS next time
  - Device is not obliged to use it, but may simply select to send a long WS each time

Relay-ACK : synchronization

- ❑ MIC field is computed as follow:

Size (bytes)	1	5	4	2	2	1	1
B_0	0x49	5 x 0x00	DevAddr	0x00	Fcnt	0x00	len(LoRaWANmsg)

The MIC is calculated as follows:

$$\boldsymbol{cmac = \mathrm{aes128\_cmac}(\mathrm{RelaySKey}, B_0 \mid msg)}$$
$$\boldsymbol{\mathrm{MIC} = cmac[0..1]}$$

Where msg is the complete LoRaWAN uplink frame

Timing alignment

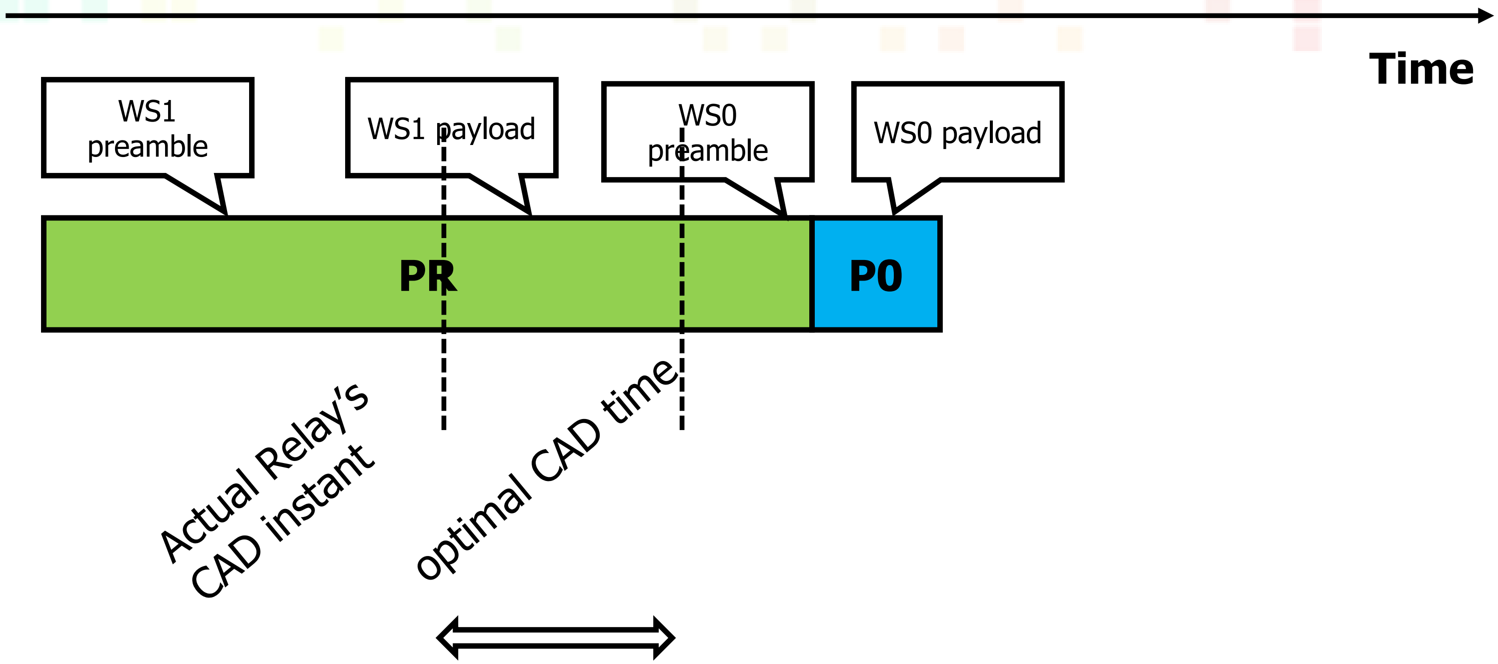
Timing field

Field	RFU	TimingOffset
bits	15:12	11:0

TimingOffset is a 2’s complement signed field encoding the offset in mSec between the moment the wake-up sequence was transmitted compared to the optimal instant.

A positive value means the device transmitted the WS too early, a negative value that device transmitted too late

Timing alignment (cont)



Offset: In that example the device transmits the WS too late, the offset value is negative.

Synchronization benefit

- ❑ Very simple scheme that allows device regularly connecting with relay to use short wake-up sequences
- ❑ Long WS = 1sec
- ❑ After initial handshake and assuming one exchange every hour
  - Clock drift between device & relay < 30ppm
  - Max clock drift during 1 hour = 3600x30ppm = +/- 100mSec
- ❑ Device can use 200mSec wake-up sequence instead of 1sec
- ❑ Divides device’s power consumption by 5

Device-relay link margin

- ❑ **The link margin is limited by the wake-up sequence detection sensitivity which is a function of spreading factor & BW**
- ❑ **Assuming SF7/125kHz for UE / SF9/500kHz for Us**
- ❑ **Detection sensitivity = -123dBm**

- ❑ **So typical device2Relay link margin is = 123 + device_tx_power dB**
  - 143dB for 20dBm TX power
  - 137dB for 14dBm

- ❑ **Higher sensitivity can be reached at the expense of higher CAD current on the relay**