

Program za upravljanje koncentratorima

Pokretanje i zaustavljanje

Za pokretanje programa potrebno je pozicionirati se u mapu /opt/ttn-gateway/bin naredbom

cd /opt/ttn-gateway/bin

i pokrenuti skriptu start_all_concentrators.sh naredbom

sudo bash start_all_concentrators.sh

(i upisati lozinku za korisnika; trenutna lozinka = omegasw).

Ta naredba će kroz sljedeće 3-4 minute otvoriti četiri prozora u kojima se može pratiti promet na svakom od koncentratora.

Program se zaustavlja tako da se zaustave sve skripte koje su u tom trenutku pokrenute klikom na 'x' ili kombinacijom ctrl+c.

Pokretanje samo jednog koncentratora

Jedan koncentrator pokreće se skriptom

/opt/ttn-gateway/bin/{{ime_koncentratora}}.sh (npr. 0_0.sh)

U njoj se radi sljedeće:

Postavlja SPI_DEV_PATH na njega (npr. '/dev/spidev.0.0' ako se želi pokrenuti samo 0.0 koncentrator) na sljedećim skriptama:

- /opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c,
- /opt/ttn-gateway/lora_gateway/libloragw/src/loragw_hal.c,
- /opt/ttn-gateway/lora_gateway/libloragw/src/loragw_spi.native.c,
- /opt/ttn-gateway/lora_gateway/libloragw/src/loragw_reg.c,
- /opt/ttn-gateway/lora_gateway/libloragw/inc/imst_rpi.h.

Nakon toga, potrebno je promijeniti i RESET_PIN u skripti `/opt/ttn-gateway/bin/start.sh` i pokrenuti istu.

Reset pinovi su postavljeni na sljedeći način:

SPIDEV	RPI pin
0.0	23
0.1	25
1.1	22
1.2	24

Program se zaustavlja klikom na 'x' ili kombinacijom tipki 'ctrl+c'

Dodatno

Lozinka za RPI: omegasw

Održavanje sustava

U slučaju problema, treba provjeriti na kojem koncentratoru dolazi do problema. To se može napraviti tako da se poziva jedan po jedan koncentrator i gleda radi li normalno (provjerava testnim LoRa uređajem). Kako se pokreće jedan po jedan koncentrator i što ta skripta radi je opisano prije u *Pokretanje samo jednog koncentratora*.

Najčešće dolazi do izbacivanja jednog koncentratora (jedne SPI veze) pa, nakon što program otvori sva četiri prozora, u sljedećih nekoliko minuta treba provjeriti rade li svi koncentratori normalno. Ako ne, treba zaustaviti sve koncentratore i ponovno pokrenuti program. Ako se takav problem ponavlja, treba isključiti RPi i provjeriti jesu li sve žice i pigtaili dobro spojeni.

Izrada

Potrebna oprema:

- Raspberry Pi 3 Model B
- Koncentrator iC880a (četiri primjerka)
- Antena 868 MHz s SMA Mount konektorom (četiri primjerka)
- Female U.fl na Male SMA pigtail (<https://hr.rsdelivers.com/product/rf-solutions/cba-ufl-smam25/rf-solutions-black-female-ufl-to-male-sma-rg178/1258188>)
- Eksperimentalna pločica i žice

Program je razvijen na temelju postojećeg pisanog od strane The Things Networka koji je radio s jednim koncentratorom budući da je do određenih implementacijskih rješenja u LoRa komunikaciji teško ili nemoguće doći.

Spomenuti program se nalazi na <https://github.com/ttn-zh/ic880a-gateway/wiki>.

Nakon što se pravilno povežu pinovi (prema uputama s githuba), s te stranice treba preuzeti skripte i pokrenuti instalaciju. Svaki put kad se pokrene Raspberry Pi, pokreće se i skripta **/opt/ttn-gateway/bin/start.sh** što je postavljeno u **/lib/systemd/system/ttn-gateway.service**. Ako se želi raditi s više antena i ne želi se koristiti ta opcija, u **/lib/systemd/system/ttn-gateway.service** treba maknuti liniju koja pokreće **start.sh**.

Kao što je rečeno u dokumentu 'Izrada LoRa bazne stanice', za svaku od SPI veza se spajaju tri pina (MISO, MOSI i SPI CLK), a uz njih za svaki koncentrator posebno napajanje, uzemljenje, Chip Select i Reset. Reset pin se postavlja u **/opt/ttn-gateway/bin/start.sh** i može biti bilo koji od slobodnih GPIO pinova. S druge strane, Chip Select je za obje SPI veze definiran. U SPI0 to mogu biti pinovi 24 i 26, a u SPI1 11, 12 i 36.

Podaci o gatewayu i korištenom Network Serveru nalaze se u

/opt/ttn-gateway/bin/local_conf.json

start.sh skripta poziva

/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c

u kojoj se nalazi program (glavni dio programa) za proslijeđivanje paketa primljenih preko antene prema mreži (Network Serveru) ili obrnuto.

Postavljanje pinova i pozivanje start.sh skripte za jedan koncentrator radi se u

/opt/ttn-gateway/bin/{{broj koncentratora (npr. 0_0) }}.sh .

U toj skripti se mijenja naziv koncentratora za skripte koje se kasnije koriste u **/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c**:

- **/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c**
- **/opt/ttn-gateway/ lora_gateway/libloragw/src/loragw_hal.c**
- **/opt/ttn-gateway/ lora_gateway/libloragw/src/loragw_spi.native.c**
- **/opt/ttn-gateway/ lora_gateway/libloragw/src/ loragw_reg.c**

Skripta poziva naredbu **sudo make -f Makefile** iz sljedećih mapa:

- **lora_gateway/libloragw/**
- **lora_gateway/**
- **packet_forwarder/poly_pkt_fwd/**
- **packet_forwarder/**

Na kraju se poziva start.sh skripta koja radi inicijalizaciju i pokreće poly_pkt_fwd.c

/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c

- Glavni dio programa je **int main()** i u njemu se nalazi inicijalizacija i pozivaju dolje navedene funkcije.
- Funkcije:
 - **Parse_SX1301_configuration**
 - **Parse_gateway_configuration**
 - **Thread_up**
 - **Thread_down**
 - **Thread_gps**
 - **Thread_gps**
 - **Concentrator_chosen**

/opt/ttn-gateway/ lora_gateway/libloragw/src/loragw_hal.c

- Skripta sadrži strukture za **pkt_rx** i **pkt_tx** i strukture paketa (p) **lgw_pkt_rx_s** i **lgw_pkt_tx_s**
- Funkcije:
 - Sx125x_write
 - Sx125x_read
 - Setup_sx125x
 - Lgw_start
 - Lgw_stop
 - Lgw_receive
 - Lgw_send

/opt/ttn-gateway/ lora_gateway/libloragw/src/ loragw_reg.c

- Skripta sadrži mapiranje registara u strukturi **lgw_reg_s**
- Funkcije:
 - Lgw_connect
 - Lgw_disconnect
 - Lgw_reg_check
 - Lgw_reg_w
 - Lgw_reg_r

/opt/ttn-gateway/ lora_gateway/libloragw/src/loragw_spi.native.c

- Inicijalizacija SPI veze
- Funkcije:
 - Lgw_spi_open
 - Lgw_spi_close
 - Lgw_spi_w
 - Lgw_spi_r
 - lgw_spi_wb() // multiple byte (burst) write
 - lgw_spi_rb() // multiple byte (burst) read

DEBUG_SPI se postavlja u **lora_gateway/libloragw/inc/config.h

parametri veze RPI-ja i koncentratora su u **lora_gateway/libloragw/inc/imst_rpi.h

LoRa dokumentacija:

'Since multiple gateways can receive the same LoRa RF message from a single end device, the LNS performs data de-duplication and deletes all copies. Based on the RSSI levels of the identical messages, the network server typically selects the gateway that received the message with the best RSSI when transmitting a downlink message because that gateway is the one closest to the end device in question.'

Tijek programa

Init connection	poly_pkt_fwd.c u funkciji <code>int main()</code> poziva funkciju <code>lgw_start()</code> koja se nalazi u lora_gateway/libloragw/src/loragw_hal.c loragw_hal.c u funkciji <code>lgw_start()</code> poziva funkciju <code>lgw_connect()</code> (SPI povezivanje s koncentratorom) koja se nalazi u loragw_reg.c loragw_reg.c u funkciji <code>lgw_connect()</code> poziva funkciju <code>lgw_spi_open()</code> koja se nalazi u lora_gateway/libloragw/src/loragw_spi.native.c
Receive	poly_pkt_fwd.c u funkciji <code>thread_up()</code> poziva funkciju <code>lgw_receive()</code> koja se nalazi u lora_gateway/libloragw/src/loragw_hal.c
Send	poly_pkt_fwd.c u funkciji <code>thread_down()</code> poziva funkciju <code>lgw_send()</code> koja se nalazi u lora_gateway/libloragw/src/loragw_hal.c

Skripta **/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/poly_pkt_fwd.c**

U glavnoj funkciji nakon inicijalizacije koncentratora i SPI veze, program pokreće 2 dretve: **thread_up** i **thread_down**.

Thread_up prima pakete s uređaja i prosljeđuje ih prema Network serveru. U njoj se vrti petlja while sve dok ne dobije od korisnika znak da prestane s radom (CTRL+C). Ako se radi s bazom podataka, funkcija se za svaki primljeni paket povezuje s bazom podataka (baza se nalazi u **/opt/ttn-gateway/packet_forwarder/poly_pkt_fwd/src/sensordata2.db** i može se otvoriti pomoću terminala ili programa DB Browser for SQLite) i provjerava je li se već primio isti paket kako ne bi bilo duplikata na Network Serveru. Bez obzira je li paket duplikat ili ne, vrijednosti paketa se upisuju u bazu podataka s razlikom kolone **sent_on_server** (1 ako je poslano na server, 0 ako nije). Sve kolone koje se upisuju bazu podataka su:

- id
- gateway_eui
- concentrator
- device_eui
- payload
- size
- modulation
- bw
- sf
- rssi
- sent_on_server
- direction (up/down)
- internal_counter
- received_at

Prednost korištenja baze podataka je backup svih podataka (poslanih i primljenih), mogućnost analize LoRa parametara i praćenje prometa podataka po koncentratorima. Nedostatak je što se korištenjem baze podataka mora postaviti uvjet da u jednom trenutku samo jedan koncentrator može pristupiti bazi (zbog ograničenja SQLite3 BP) dok se inače u isto vrijeme mogu provoditi programi s dva koncentratora, odnosno s dvije različite SPI veze. U slučaju da se ipak želi koristiti baza podataka, u funkciji koja odlučuje koji koncentrator u kojem trenutku pristupa SPI vezi (concentrator_chosen), treba podijeliti vrijeme pristupa na 4 (za svaki koncentrator) dok je inače podijeljeno na 2 (jer 2 koncentratora dijele istu SPI vezu).

Za slanje prema uređaju koristi se funkcija **thread_down**. Kod nje je ideja bila da se prilikom slanja naredbe prema uređaju koristi koncentrator, odnosno antena s kojom je imao najbolju vezu. Program za takvo odlučivanje je napisan i zakomentiran budući da je payload koji je primljen s Network Servera, a u kojem se nalazi naredba (poruka) i naziv uređaja prema kojem ona ide, enkriptiran tako da se ne može saznati koji je to uređaj. U slučaju da se do toga ipak dođe, za taj podatak je napravljeno mjesto u bazi podataka (device_eui) i može se pustiti i taj dio koda koji provjerava zadnjih 30 poruka između bazne stanice i tog uređaja i koristi onu antenu (odnosno koncentrator) koja ima najbolji signal to noise ratio. Ako se do tog podatka ipak ne može doći, budući da bazna stanica ionako ne šalje više od jedne poruke u trenutku, a i korištenjem samo onog koncentratora koji ima najbolju vezu s uređajem ne šteti energiju, može se koristiti i dalje bez baze podataka.

Ova funkcija, kao i **thread_up** za svaku SPI vezu, a tako i za svaki koncentrator, radi periodički (kroz funkciju concentrator_chosen). U jednom trenutku su aktivni uređaj 0 veze SPI0 i uređaj 2 veze SPI1, a u drugom uređaj 1 veze SPI0 i uređaj 1 veze SPI1. Vrijeme trajanja jednog perioda je 3 sekunde (određeno empirijski najboljim rezultatima). Gubitak paketa nije problem budući da svaki koncentrator ima buffer koji čuva poruku dok ga se ne odabere za proslijeđivanje Raspberry Piju.