Obsah

Port 502 — Modbus TCP server **1**
Parametry spojení 1
Mapa registrů (setMap) 1
Datový tok 2
Konvence kódování hodnot 2
Rozložení adresního prostoru 2
1000–1203 — data z RTU pro NPA *(zápis klientem)* 2
2000–2092 — předávací místo obchodního měření (TML) 2
2500–2632 — suma rozpadových míst + povely regulace 3
3000 + 200·n — jednotlivá rozpadová místa (lokace n) 6
19000–19184 — TML detail (kompatibilní s Janitza, BE float) 7
10032–10062, 13437–13444 — parametry/firmware TML 7
40000 + 200·id — rozpadová místa z externí RTU (ext_ds) 7
50000 + 200·a — per-device blok (dd_modbusAddr = a, 0--9) 7
Evidence zápisů (kdo, kdy) a stránka Modbus registry 8
Přidání nové hodnoty do mapy 8

Port 502 — Modbus TCP server

Platí pro verzi NPA R04: 2.15.beta12 · Vytvořeno: 2026-06-02 · Aktualizováno: 2026-09-11

> Zdroj: `nmodbusrv.cpp` / `nmodbusrv.h`, třída `NModbuSrv` (singleton `NModbuSrv::GetNModbuSrv()`). Implementace přes `QModbusTcpServer`.

Hlavní strojové rozhraní pro nadřazené systémy (SCADA, RTU provozovatele distribuční soustavy). Stejná mapa registrů je sdílena i s Modbus RTU slave variantou (`QModbusRtuSerialSlave` na `/dev/ttyS2`, aktivní jen při `ns_485_m-s == "ns_485_slave"`).

Parametry spojení

Parametr	Hodnota	Zdroj
Naslouchá na	`0.0.0.0`	pevně
TCP port	`mC2["ns_netPort"]` (typicky **502**)	`/data/NPA_R04.cnf`
Modbus Unit ID	`mC2["ns_netId"]` (výchozí 1)	`/data/NPA_R04.cnf`
ListenOnlyMode	`false` (povolen zápis)	pevně

Hodnoty `ns_netPort` / `ns_netId` lze přečíst i přes JSON-RPC dotaz `get_modbus_srv_param` (viz port-7777-json-rpc.md) — používá to vzdálené párování NPA jednotek přes VPN-ID.

Mapa registrů (`setMap`)

Tabulka	Rozsah	Pozn.
Coils	`0 … G_NUM_NOIL` (= 5)	`globals.h: G_NUM_NOIL = 5`
Discrete Inputs	`0 … 0xE`	
Input Registers	`0 … 60000`	čtení (zrcadlo `NState`)
Holding Registers	`0 … 60000`	čtení + povelové zápisy

Většina hodnot se zapisuje **současně do Input i Holding** registrů (viz `WriteMTUint16` ad.), takže klient je může číst funkcí 0x03 i 0x04. Při aktivním RTU slave se zrcadlí i tam.

Datový tok

- **Výstup (NPA → klient):** funkce `Loop()` (volaná z 50 ms `ReadIO` a 500 ms `Test1s` tiků v `main.cpp`) průběžně přepisuje registry hodnotami z `NState`.
- **Vstup (klient → NPA):** zápis klienta vyvolá signál `dataWritten` → `updateRegistersTCP/RTU()` → `updateRegisters()`, kde se vybrané adresy interpretují jako povely (regulace, blokování, galvanické odpojení) a propíšou do `NState` / `NConfig`.

Konvence kódování hodnot

Pomocné funkce vynucují pořadí registrů a šířku:

Funkce	Typ	Pořadí slov
`WriteMTUint16`	16bit	—
`WriteMTUint32`	32bit	high-word first (BE po slovech)
`WriteMTUint64`	64bit	
`WriteMTFloat`	float32	**BE** (high word na nižší adrese)
`WriteMTFloatLE`	float32	**LE** (slova prohozená)
`WriteMTDoubleLE`	float64	**LE**
`WriteMT2bit`	16bit	dvoubitový stav: bit0 = OFF, bit1 = ON; `0xFFFF` = bez informace
`WriteFloatToIR` / `WriteUint64ToIR` / `WriteInt64ToIR`	jen Input Reg.	

Hodnoty bez dat se zapisují jako `qQNaN()` (float) nebo `0xFFFF` (2bit/uint).

Rozložení adresního prostoru

`1000–1203` — data z RTU pro NPA *(zápis klientem)*

El. i neel. veličiny zapisované nadřazenou RTU. Při zápisu se uloží do `NState` (`MBReg`, značka `iw`) a předají do driveru `rtu_master` (`RtuMeter::SaveMeter`).

`2000–2092` — předávací místo obchodního měření (TML)

„Od adresy 2000 jsou data z předávacího místa obchodního měření pro potřeby RTU.”

Adresa	Veličina	Formát
2000--2007	stavy spínačů TML-QM/QS/QSF/Q1/Q6/QE6/FU1/FA1	2bit (OFF/ON)
2008	TML-H100NAT	uint16
2009--2013	I1, I2, I3	float LE
2015--2021	Us12, Us23, Us31, Us	float LE
2023--2027	U1, U2, U3	float LE
2029--2035	P1, P2, P3, P	float LE
2037--2043	Q1, Q2, Q3, Q	float LE
2045--2051	cosφ L1, L2, L3, Σ (počítáno z P/Q, vydáváno za PF)	float LE
2053	f	float LE
2055	TML-H850T	uint16
2056	SPG_HDO	uint16
2057	PF – power factor |P|/S v % (0--100), 0xFFFF = bez informace (od 2.15.beta12 počítáno z P a S)	uint16

Adresa	Veličina	Formát
2058	Rotate_f	float LE
2060	TML-UPS-BatteryLow – vstupní funkce `TML-UPS-BattLo` (do 2.15.beta11 se hledal neexistující název → vždy bez informace)	uint16
2093 / 2095 / 2097	SOL / T / WS – osvit, teplota, vítr z RTU elektroměru (blok 1000, od 2.15.beta12)	float LE
2061 / 2065 / 2069	E_del_1 / E_del_2 / E_del_3 – dodaná činná energie po fázích [Wh]	double LE (4 reg.)
2073	E_del – dodaná činná energie do DS, suma [Wh]	double LE
2077 / 2081 / 2085	E_con_1 / E_con_2 / E_con_3 – odebraná činná energie po fázích [Wh]	double LE
2089	E_con – odebraná činná energie z DS, suma [Wh]	double LE

Energie se sčítají z přírůstků TML elektroměru (`Device::SaveEnergy`). Nedostupná hodnota se od 2.15.beta9 zapisuje jako **0** (dřív NaN). Energie po fázích poskytují drivery Janitza / BMR PLA33, RTU meter, PRO380, EM300/ET300 (záporné po fázích jen řada ET), SolarEdge meter, NPA slave TML/DS a ext_ds (pokud je PLC plní). U elektroměrů, které je v protokolu nemají (EME319, SMEAN, pulzní, EM330/EM340 pro zápornou energii), se přírůstek sumy **rozpočítá na fáze** podle okamžitých výkonů fází `|P1|:|P2|:|P3|` (bez nich rovným dílem) – hodnoty jsou tedy odhad, v detailu zařízení je příznak `E_del_faze_rozpocitane` / `E_con_faze_rozpocitane`. Teprve když chybí i suma, je 0.

2500–2632 — suma rozpadových míst + povely regulace

Adresa	Veličina	Formát	Směr
2501	systémová chyba (SysErr)	float LE	out
2503	chyba zařízení DS	float LE	out
2505 / 2506	suma stavů DS-QM / DS-FA1	2bit	out
2507	DS-H100NAT	uint16	out
2508	P (suma lokací FVE/BESS/MVE/KGJ/BPS/VTE/DA/HE/TUV/AKU/Wallbox)	float LE	out
2516	Q (tytéž lokace)	float LE	out
2518--2532	Us, U1--U3, I1--I3, f	float LE	out
2534	Cap-proc_BSAE (SOC %) — kapacitně vážený průměr přes BESS zařízení výrobny (výběr podle příznaku zařízení `bess`, který si BESS drivery nastavují samy, ne podle typu místa; bez BESS / bez SOC = 0)	uint16	out
2535	**Globální BESS – nabíjení (Ch_BESS_reg)**	uint16	**in**
2536	Ch_BESS_reg (zrcadlo)	uint16	out

Adresa	Veličina	Formát	Směr
2538	**Globální činný výkon výrobny (SPG, stupeň 0--4 = P0--P4; výchozí 0/0/30/60/100 % dle PDS). Stupeň se přijme jen se zaškrtnutou *Aktivace Px* (`fs_rgP0sw`…`fs_rgP4sw`) v Nastavení výrobny – jinak se povel jen zaloguje a zrcadla 2539 / 3029 / 40031 zůstávají na předchozím stupni**	uint16	**in**
2539	SPG_reg (zrcadlo globálního stupně z 2538; žádná suma z míst 3029+200n ani zařízení 40031+200n)	uint16	out
2540	H796F	uint16	out
2541	**Globální jalový výkon Q (QLC_reg, 1--5 = QL375 / QL185 / Q0 / QC185 / QC375)**	uint16	**in**
2542 / 2543	QLC_reg (zrcadlo) / QLC_Err	uint16	out
2544	**Globální účiník cosφ (SPFG_reg, 1--5 = cos(φ)_L1 / L2 / 1 / C2 / C1)**	uint16	**in**
2545 / 2546	SPFG_reg (zrcadlo) / SPFG_Err	uint16	out
2547	**Globální BESS – vybíjení (Dch_BESS_reg)**	uint16	**in**
2548	Dch_BESS_reg (zrcadlo)	uint16	out
2550	**Blokování zapnutí výrobny – distribuce (H298CBE)**	uint16	**in**
2551	H298CBE_conf (globální souhrn)	uint16	out
2552	**BESS – blokování zapnutí výrobny (H298CBE_BESS)**	uint16	**in**
2553	H298CBE_BESS_conf (globální souhrn)	uint16	out
2554	**Galvanické odpojení zdrojů (GD_Power_supply)**	uint16	**in**
2555	GD_Power_supply (zrcadlo)	uint16	out
2560--2566	offline příznaky (Power, ELM_TML, DSi, DS, BackUp, BESS, Wallbox)	uint16	out
2567	H100IF (globální souhrn)	uint16	out
2568	H835L	uint16	out
2571	Total_stop	uint16	out
2572 / 2576	E_del / E_con (suma typů výroben)	double LE	out
2580--2604	energie BESS, AKU, Wallbox, DSi (E_del/E_con/E_BESS_bu)	double LE	out
2608--2632	Us12/23/31, P1--P3, Q1--Q3, cosφ L1--L3+Σ (DS)	float LE	out
2634	**Power_export_limit** – dynamické omezení rezervovaného výkonu výrobny [W], „−“ dodávka do sítě, „+“ odběr; uplatní se jen s fajfkou `fs_extPlcKwAccept` (výchozí ano)	float (2 reg.)	**in**

Adresa	Veličina	Formát	Směr
2636	**DS_power_value** – globální omezení činného výkonu [%] všech míst; uplatní se jen s fajfkou `fs_extPlcPrcAccept` (výchozí ano); nejnižší % ze všech řídicích funkcí vítězí	uint16	**in**
2510	**Power_export_reserved** – dynamické omezení rezervovaného výkonu z dispečerského řízení [W], „−“ dodávka, „+“ odběr; povel DSŘ, přijímá se s fajfkou `fs_distCmd` (Nastavení výrobny, „Řídící povel Distribuce“, výchozí ano) jako kandidát povoleného přetoku v `RegulacePodlePretoku` (od 2.15.beta12)	float (2 reg.)	**in**
2512	Power_export_reserved_conf – 0 bez požadavku nebo nepřijímá se, 1 požadavek přijat, 2 splněn (měřený přetok ≤ požadavek + 1 % instalovaného výkonu)	uint16	out
2513	SY_avr – měrná výtěžnost [kWh/kWp] = dodaná energie výroben / instalovaný výkon	float LE	out
2515	PF_avr – power factor rozpadových míst |P|/S v %	uint16	out
2556 / 2558	**GD_BESS / GD_Wallbox** – galvanické odpojení BESS / Wallbox; stav pro výstupní funkce `GD_BESS` / `GD_Wallbox` (bez časované sekvence)	uint16	**in**
2557 / 2559	GD_BESS_conf / GD_Wallbox_conf (zrcadlo, −1 bez povelu)	uint16	out
2569 / 2570	DS_Door_open_suma / TML_Door_open_suma – 1 otevřeno (vstupní funkce `DS` / `TML`, 1 = zavřeno; OR přes místa)	uint16	out
2637--2650	F471_conf, Uzad/cos_fi/Qzad/dQdt/Treg/Utol/Tsr_conf – zrcadlo přijatých hodnot U/Q (−1 / NaN bez povelu)	uint16 / float LE	out
2652--2665	**F471, Uzad, cos_fi, Qzad, dQdt, Treg, Utol, Tsr** – regulace U/Q: hodnoty se přijmou, evidují (NEvents) a zrcadlí do _conf; **regulaci U/Q NPA neprovádí**	uint16 / float	**in**
2667 / 2669	**SK: Vypnutí výrobny / Odblokuj vypnutí** – přijato a evidováno, bez akce	uint16	**in**
2668 / 2670	SK potvrzení (zrcadlo)	uint16	out

Adresy se značkou **in** jsou povelové holding registry; jejich změna spustí akci v `updateRegisters()` (nastavení regulačního stupně, uložení stavu, případně časovanou sekvenci galvanického odpojení u 2550/2552).

`3000 + 200·n` — jednotlivá rozpadová místa (lokace `n`)

Pro každou lokaci z `mC2["location"]` blok 200 registrů. Klíčové offsety:

Offset	Veličina	Formát	Směr
+0	typ DS (`lo_typeDs` → kód)	uint16	out
+1 / +2	DS-QM / DS-FA1	2bit	out
+3	DS-H100NAT	uint16	out
+4	P	float LE	out
+8	Q	float LE	out
+10--+24	Us, U1--U3, I1--I3, f	float LE	out
+26	Cap-proc_BSAE (SOC %) — kapacitně vážený průměr přes BESS zařízení místa	uint16	out
+27/+29/+31/+33/+35	Ch_BESS_reg / SPG_reg / QLC_conf / SPFG_reg / Dch_BESS_reg	uint16	out
+30	H796F	uint16	out
+32 / +34	QLC_Err / SPFG_Err	uint16	out
+37 / +38	H298CBE_conf / H298CBE_BESS_conf	uint16	out
+57 / +61 / +65	E_del / E_con / E_BESS_bu – energie místa [Wh]	double LE	out
+69 / +73 / +77	E_del_1 / E_del_2 / E_del_3 – dodaná energie místa po fázích [Wh], 0 = elektroměr neposkytuje (od 2.15.beta9)	double LE	out
+81 / +85 / +89	E_con_1 / E_con_2 / E_con_3 – odebraná energie místa po fázích [Wh] (od 2.15.beta9)	double LE	out
+44	**fs_DS_power_value** – omezení činného výkonu místa [%]; uplatní se jen s fajfkou `lo_extPlcAccept` místa (výchozí ano); s reg. 2636 platí nižší (viz rizeni-vykonu.md 2.5)	uint16	**in**
+46--+51	offline příznaky (FVE/DSi/DS/BackUp/BESS/Wallbox)	uint16	out
+52	H100IF	uint16	out
+53	H835L	uint16	out
+55	**Total_stop**	uint16	**in**
+56	Total_stop (zrcadlo)	uint16	out
+57/+61/+65	E_del / E_con / E_BESS_bu	double LE	out
+95--+119	Us12/23/31, P1--P3, Q1--Q3, cosφ L1--L3+Σ	float LE	out
+6 / +7	Power_export_reserved_conf (= globální 2512) / PF_avr místa |P|/S v %	uint16	out
+39 / +40 / +41	GD_Power_supply_conf / GD_BESS_conf / GD_Wallbox_conf (stav místa, jinak globální)	uint16	out
+42	T_rack – teploměr (dd_type `tme`) přiřazený místu, jinak NaN	float LE	out
+54	DS_Door_open – 1 otevřeno (vstupní funkce DS místa)	uint16	out
+93	SY_avr místa [kWh/kWp] = E_del / lo_insPower	float LE	out
+121	DS-UPS-BatteryLow (vstupní funkce `DS-UPS-BattLo` místa)	uint16	out

Offset	Veličina	Formát	Směr
+122--+135	F471, Uzad … Tsr – zrcadlo globálních U/Q povelů	uint16 / float LE	out
+137 / +138	SK potvrzení (zrcadlo globálních)	uint16	out

Rozsah bloku: `3000 … 3000 + 200·20` (max 20 lokací).

19000–19184 — TML detail (kompatibilní s Janitza, BE float)

Plný profil předávacího měření ve **float BE**: U/Us, I, P, S, Q po fázích a sumy (19000--19052), všechny energetické registry (činné/jalové/zdánlivé, ind./kap., 19054--19108), THD U/I (19110--19120), DeviceId (19176, uint32), RS485 baud (19184, uint32).

10032–10062, 13437–13444 — parametry/firmware TML

CT/VT převody (`MeasurementCurrent/VoltageTransformerPrimary/Secondary0..3`, float BE) a firmware řetězec (8× uint16).

40000 + 200·id — rozpadová místa z externí RTU (ext_ds)

Zrcadlení pro zařízení typu `ext_ds` (`dd_dev_id = id`). Zápis klientem (celý blok) plní `ExtDs::SaveMeter`. NPA do bloku navíc zapisuje regulační povely pro dané zařízení:

Offset	Veličina
+28 / +31 / +34 / +37 / +40	Ch_BESS_reg / SPG_reg / QLC_reg / SPFG_reg / Dch_BESS_reg
+43 / +45	H298CBE / H298CBE_BESS
+67	Total_stop

Z bloku NPA čte (zapisuje PLC): +32 echo stupně SPG_reg, +76 / +80 / +84 E_del / E_con / E_BESS_bu (double LE, s posunem `+200·id` – do 2.15.beta9 se četly chybně vždy ze slotu 0) a od 2.15.beta9 volitelně energie po fázích +88 / +92 / +96 E_del_1..3 a +100 / +104 / +108 E_con_1..3 (double LE, PLC, které je neplní, nechá 0).

Z +70 (`lo_Power`, float LE) se plní jmenovitý výkon měniče `dd_Power`. Kapacitu baterie PLC neposílá – `dd_BESSCapacity` i `dd_Power` lze v editaci zařízení zadat ručně (fajfka „zadat ručně“; kapacita od 2.15.beta16).

Rozsah: `40000 … 40000 + 200·20`.

50000 + 200·a — per-device blok (dd_modbusAddr = a, 0--9)

Každé zařízení s nastaveným `dd_modbusAddr` (0--9) dostane blok 200 registrů. Kompletní elektrický profil:

Offset	Veličina	Formát
+0--+6	Us12, Us23, Us31, Us	float LE
+8--+14	U1, U2, U3, Uf	float LE
+16--+22	I1, I2, I3, I	float LE
+24--+30	P1, P2, P3, P	float LE
+32--+38	Q1, Q2, Q3, Q	float LE
+40--+46	S1, S2, S3, S	float LE
+48--+54	cosφ L1, L2, L3, Σ	float LE

Offset	Veličina	Formát
+56 / +58	f / Rotate_f	float LE
+60--+104	E_del_1/2/3/Σ, E_con_1/2/3/Σ, E1/E2/E3/E_total	double LE
+108--+168	E_app_, *E_react_ind_*, E_react_cap_, *E_react_*	double LE
+172--+182	THD UL1--3, IL1--3	float LE
+184 / +186 / +188	SOL (osvit) / T (teplota) / WS (vítr)	float LE
+190	stav (1 = offline)	uint16

Evidence zápisů (kdo, kdy) a stránka Modbus registry

Od 2.15.beta12 server eviduje **každý zápis do vstupního registru** (`in`): adresu, hodnotu, čas (UTC) a **zdroj** – IP Modbus TCP klienta nebo RTU u sériového slave. IP se zjišťuje přes `QModbusTcpConnectionObserver` (Qt ≥ 5.13: slot readyRead socketu je připojen dřív než interní zpracování Qt, peer je při `dataWritten` znám přesně); na starším Qt (ROCK) záložně přes `QTcpServer` uvnitř `QModbusTcpServer` (`newConnection` → zaháčkování socketů) – při jednom klientovi přesně, při více současných klientech může být přiřazení o jeden požadavek pozadu. Evidenci vede singleton `NEvents` (`nevents.cpp`):

- **Inteligentní dedup:** opakovaný zápis stejné hodnoty do téhož registru se nezapisuje, jen se počítá (×N, čas posledního opakování); při první repetici jedna řádka „hodnota se opakuje", při změně nová řádka s počtem opakování předchozí hodnoty.
- **Soubor** `/log/npa_events.log` (řádek `čas | zdroj | kdo | co | hodnota | poznámka`, rotace při 2 MB do `.bak`); obsahuje i ruční zásahy z webu (`web`, viz `port-8080-http.md`).
- Datové proudy (bloky `1000`–`1203` z RTU a `40000+200·id` z externího PLC) události nevytvářejí – u bloku je jen „poslední zápis kdo/kdy" a počet.
- **Dálkový log:** nové události jdou v dalším paketu `/measuring` v poli `events` (viz `interfaces.md` §7); fronta (max. 200, `events_dropped`) se vyprázdní až po úspěšném odeslání.

Stránka **Modbus registry** (menu, admin+) zobrazuje mapu podle referenční tabulky **NPA-R04_MODBUS-tabulka verze 25.09** (`html/v2/modbus_map.json`, generuje `doc/build_modbus_map.py` z `.ods`; bloky 19000 / 10032 / 13437 / 50000 v tabulce nejsou a jsou v generátoru popsané ručně podle `NModbuSrv::Loop`): bloky jako sbalovací karty, místa a zařízení dynamicky podle konfigurace, sloupce adresa / směr / typ / označení / popis / aktuální hodnota / poslední zápis (jen `in`), filtr, autoobnova, adresy mimo mapu v sekci „Ostatní", dole posledních 200 řádků logu zápisů a odkaz na celý soubor (`events_log`).

Přidání nové hodnoty do mapy

1. V `Loop()` přidat `WriteMT…(adresa, NState…)` na zvolenou adresu (dodržet pořadí slov dle typu — viz tabulka konvencí).
2. Má-li jít o **povel** (zápis klientem), přidat větev `if (address == …)` do `updateRegisters()`.
3. Aktualizovat referenční mapování „NPA-R04_MODBUS-tabulka" i tento dokument a přegenerovat `html/v2/modbus_map.json` (`doc/build_modbus_map.py <tabulka.ods>`).
4. Vstupní registr: přidat volání `NEvents::ModbusWrite(...)` v `updateRegisters()`.