Obsah

Řízení výkonu NPA R04 – uživatelská příručka **1**
Obsah . . . 1
1. Jak řízení funguje (přehled) . . . 2
Rychlá orientace v menu . . . 2
2. Nastavení rozpadového místa . . . 2
2.1 Základní údaje . . . 3
2.1.1 Jmenovitý výkon měniče (dd_Power) . . . 3
2.2 Způsob regulace zařízení (lo_regDev) . . . 4
2.3 Mapování povelů Px na procenta (lo_rgP0…lo_rgP4) . . . 4
2.4 Priorita vypínání (lo_regPwrPriority) . . . 6
2.5 Obchodní flexibilita – řízení činného výkonu z externího PLC . . . 6
2.6 Vypnutí ovládání zařízení (dd_ctrlEnable) . . . 7
2.7 Maximální výkon z rozpadového místa (lo_maxOutPwrEn, lo_maxOutPower) . . . 7
3. Povely a omezení . . . 8
3.0 SoP – Celkový instalovaný výkon . . . 8
3.1 Povely distribučního dispečinku (DŘ) . . . 9
3.2 Omezení přetoku do sítě . . . 9
3.3 Spotové ceny . . . 11
3.4 Rampy – gradient změny činného výkonu . . . 11
3.5 Aktuální povel ve stavu (akt_proc) – co se ukládá a odesílá . . . 12
4. Nastavení regulátorů . . . 12
4.1 Význam konstant . . . 13
4.2 Vzorec regulátoru . . . 13
5. Příklady s grafy . . . 13
5.1 Přesné doregulování vs. okamžitý povel . . . 13
5.2 Měniče s velkým dopravním zpožděním . . . 14
5.3 Co se stane při kmitání (špatné parametry) . . . 15
5.4 Asymetrie i+ / i- – rychlé stažení přetoku, pomalý náběh . . . 15
6. Postup ladění (cookbook) . . . 16
Zdrojové soubory . . . 16

Řízení výkonu NPA R04 – uživatelská příručka

Platí pro verzi NPA R04: 2.15.beta5 · Vytvořeno: 2026-06-17 · Aktualizováno: 2026-08-04

Tento dokument vysvětluje, jak jednotka NPA R04 řídí **činný výkon** výrobny: jak se nastavují **rozpadová místa**, jak se zadávají **povely a omezení** a jak se ladí **regulátory**. Na konci jsou praktické **příklady s grafy** – včetně přesného doregulování na procenta, řízení měničů s velkým dopravním zpožděním a ukázky, kdy regulátor začne kmitat.

> Tato příručka popisuje regulaci **činného výkonu** (modul `NRegulation2`). **Řízení BESS, cos φ a jalového výkonu sem nepatří** – mají samostatné dokumenty (BESS viz `bess.md`, cos φ a jalový výkon samostatně).

Obsah

1. Jak řízení funguje (přehled)
2. Nastavení rozpadového místa
3. Povely a omezení
4. Nastavení regulátorů
5. Příklady s grafy
6. Postup ladění (cookbook)

---

1. Jak řízení funguje (přehled)

Regulační smyčka běží **každých 500 ms** (`NRegulation2::Loop`). V každém kroku:

```
POVELY / OMEZENÍ                  ZPĚTNÁ VAZBA
- povel P0..P4 (DŘ)  -> pomalá    měření výkonu
- přetoky do DS      -> rychlá    výrobny
- spotové ceny, SoP                   |
      |                               |
      v                               |
[ Suma požadovaný výkon ] <-----------+
  vyber přísnější (rychlá / pomalá)
      |
      v
[ PI regulátor  reg1 / reg2 ]
      |
      v
[ rozpad do míst podle priority ]
  Spojitá: pošle přesné %
  Px:      vybere nejvyšší stupeň P0..P4
      |
      v
[ zařízení / měniče ]
```

- **Požadovaný výkon** se počítá ze dvou větví:
  - **pomalý** – z povelu distribuce `P0..P4` (= `PowerLimit` v %),
  - **rychlý** – z omezení **přetoku** do sítě (povolený přetok, SoP rezervovaný výkon, spotové ceny).
- Regulátor použije tu **přísnější** větev: je-li rychlý požadavek ≤ pomalý, řídí **rychlý regulátor přetoků** (reg2), jinak **pomalý regulátor povelů** (reg1).
- Výsledné procento se rozdělí do **rozpadových míst** podle priority a podle toho, zda místo umí **spojité** řízení nebo jen **stupňové povely Px**.

Rychlá orientace v menu

Veškeré nastavení se dělá ve webovém rozhraní jednotky (horní menu). Kde co najdete:

Co nastavujete	Cesta v menu
Vlastnosti místa, způsob řízení, povely Px (`lo_*`)	**Rozpadová místa** → klik na místo → *editace*
SoP rezervovaný výkon/příkon, úrovně P0--P4 (`fs_sopRsrvPwr`, `fs_rgP0…4`)	**Nastavení výrobny**
Omezení přetoku do sítě (`fs_flwRest…`)	**Nastavení výrobny**
Obchodní flexibilita – fajfky *Akceptovat vzdálené řízení* pro reg. 2636 / 2634 (`fs_extPlcPrcAccept`, `fs_extPlcKwAccept`)	**Nastavení výrobny**
Obchodní flexibilita – fajfka pro reg. 3x44 místa (`lo_extPlcAccept`) + souhrn řídicích funkcí	**Rozpadová místa** → *editace*
Konstanty regulátorů činného výkonu (`fs_reg1/2_k/i+/i-`), spotové ceny (`fs_TarifPacLimit…`)	**Nastavení výrobny** → dole *Nastavení placených funkcí*

> Editovat hodnoty placených funkcí (včetně konstant regulátorů) může jen účet s vyšším oprávněním; běžný uživatel je vidí jen ke čtení.

---

2. Nastavení rozpadového místa

> **Menu: Rozpadová místa** → klikněte na konkrétní místo → tlačítko *editace*.

2.1 Základní údaje

Pole	Klíč	Význam
Typ	`lo_typeDs`	regulaci činného výkonu řeší typy **FVE, MVE, KGJ, BPS, VTE, DA, HE** (BESS, TUV, AKU, Wallbox mají vlastní řízení)
Instalovaný výkon	`lo_insPower`	špičkový instalovaný výkon [Wp] — zadává uživatel
Skutečný činný výkon	*(dopočítává se)*	jmenovitý činný výkon měničů [W] = **součet `dd_Power` zařízení v místě**, jen ke čtení

Maximální výkon místa pro regulaci = `min(Σ dd_Power, lo_insPower)` (když je `lo_insPower` zadán), jinak `Σ dd_Power`. Procenta se vždy počítají z tohoto maxima.

> **Změna od verze 2.14 (2026-07-03).** Dřív se „Skutečný činný výkon" zadával ručně a ukládal do configu pod klíčem `lo_Power`. Nyní je to **vlastnost zařízení** (`dd_Power`), výkon místa je jejich součet (`NConfig::GetLocationPower`) a klíč `lo_Power` se při načtení configu z rozpadových míst **maže**.

2.1.1 Jmenovitý výkon měniče (`dd_Power`)

> **Menu: Zařízení** → klik na zařízení → *editace*, pole **Činný výkon měniče [W]**.

Chová se stejně jako kapacita BESS (viz Řízení BESS §5):

- **bez fajfky „zadat ručně"** (výchozí) — pole je jen ke čtení a plní se **hodnotou z Modbusu**; načtená hodnota je vždy vidět i v **Detailu zařízení** (`Vykon menice nacten [W]`).
- **s fajfkou `dd_PowerHand`** — hodnotu zadává uživatel; načtená z Modbusu slouží jen jako informace. Odfajfkováním se uložená hodnota smaže, aby ji převzal Modbus.

Zápis do configu proběhne **jen při změně > 10 %** (aby kolísání čtení nepřepisovalo config při každém cyklu) a **nikdy se nezapisuje nula** ani neplatné čtení.

Odkud se hodnota čte u jednotlivých typů měničů:

Driver	Zdroj hodnoty
GoodWe (`goodwe`, `gw_15_50kw_bess`, `gw_15_50kw_inverter`)	registr 35001 (*Rate Power*)
Solax DataHub 1000	„Instalovaný výkon [W]" hlášený DataHubem
Solax EMS1000	protokol registr jmenovitého výkonu nemá → **štítkový součet dle typu skříně** (řada TRENE), nebo u **X3-AELIO** největší výkon hlášený EMS v registrech 104/105 (běžící maximum, neklesá při deratingu); použitý zdroj je vidět v Detailu zařízení (od 2.15.beta4)
Solinteg (`solinteg`)	registr 10070 (*Inverter Rated Power*) (od 2.15.beta7); některé FW ho nepodporují a vrací 0xFFFFFFFF → výkon se vezme **z názvu modelu** (např. M2HT-50K-150 = 50 kW); použitý zdroj je v Detailu zařízení („Jmenovity vykon zdroj", od 2.15.beta9)
Sinexcel (`sinexcel`)	registr 53231 (*Nominal online capacity* – kapacita online modulů, běžící maximum, aby výkon neklesal při výpadku modulu) (od 2.15.beta8)
SolarEdge	`Max Active Power`
Sungrow Logger1000	`Max. total nominal active power of array` (výkon celého pole)
Sungrow	`Nominal active power`
Deye	`Rated Power`
Sermatec	štítkový výkon podle modelu skříně (30 / 60 / 120 kW, tabulka v driveru)
extds (podřízené NPA/PLC)	činný výkon místa z registru `40070 + 200·n`

Ostatní drivery (např. GRES, SMA) jmenovitý výkon nehlásí — u nich se zadává ručně. Kontrola configu upozorní, když má výkonové zařízení `dd_Power ≤ 0` nebo když je součet za regulované/BESS místo nulový (viz Konfigurace §Kontrola konfigurace).

2.2 Způsob regulace zařízení (`lo_regDev`)

Toto je klíčové rozhodnutí podle toho, **co zařízení v daném místě umí**:

Volba	Klíč	Kdy použít
Spojitá regulace	`Spojita`	Zařízení umí přijmout libovolné % (měniče FVE, KGJ s plynulým řízením…). Regulátor pošle **přesné procento**.
Regulace povely Px	Px	Zařízení **neumí plynulé řízení** – jen pevné stupně (relé, přepínání odboček, starší měniče reagující jen na povelové úrovně). Regulátor vybere **nejvyšší stupeň**, jehož výkon nepřekročí požadované %.
Bez regulace	none	Místo se nereguluje. Vypne se (0 %) jen pokud je požadováno < 100 %.

> **Px = „jen pro zařízení, která neumožňují plynulé řízení“.** Místo plynulého procenta se používá 5 pevných stupňů `P0…P4` (viz dále). Regulátor nikdy nenastaví procento „mezi” stupni – sníží se na nejbližší** nižší nebo rovný** stupeň, aby nedošlo k překročení limitu.

Volba se nabízí jen u míst typu `FVE`, `MVE`, `KGJ`, `BPS`, `VTE`, `DA`, `HE` – tedy u těch, která regulátor řadí do skupin podle priority. U míst typu **BESS** se nezobrazuje: způsob řízení baterií určuje regulace BESS (nabíjecí/vybíjecí limity, peak shaving), ne `lo_regDev`.

2.3 Mapování povelů Px na procenta (`lo_rgP0…lo_rgP4`)

Distribuce posílá povelové úrovně `P0…P4`. Pro každou úroveň se v místě nastaví, **na kolik procent** výkonu se má místo omezit:

Úroveň	Klíč	Význam dle distributorů (FVE/VTE)
P0	`lo_rgP0`	0 % včetně vnořené AKU (EG.D; v NPA volitelně galvanické odpojení `fs_PwrDiscP0`)
P1	`lo_rgP1`	0 % (ČEZ SPG000)
P2	`lo_rgP2`	30 % (ostatní zdroje KGJ/BPS/MVE: ČEZ 50 %, EG.D 50 %)
P3	`lo_rgP3`	60 % (ostatní zdroje: ČEZ 75 %, EG.D 70 %)
P4	`lo_rgP4`	100 % – základní provozní stav, povel ruší předchozí omezení

Globální výchozí hodnoty z továrního nastavení se zobrazují vedle (`fs_rgP0…fs_rgP4`).

Výchozí hodnoty dle směrnic distributorů (od 2.15.beta9) Jedna tabulka `NConfig::DsoDefaultLevels()` (`nconfig.cpp`) platí pro ČEZ Distribuce, EG.D, PREdistribuce (PN KA 508 není veřejná) i VSD. Chybějící nebo **prázdný** klíč se doplní při startu a při každém uložení konfigurace (`NConfig::SeedDsoDefaults`, záznam v logu); na stránce *Nastavení výrobny* jsou hodnoty vidět jako placeholder a tlačítko **Výchozí dle distribuce** je vyplní do polí (uložit je nutné tlačítkem *Uložit*). Zaškrtávátka aktivace stupňů se nemění – které stupně jsou povolené, určuje smlouva o připojení.

Skupina	Klíče	Výchozí hodnoty	Zdroj
P	fs_rgP0…4	0 / 0 / 30 / 60 / 100 %	ČEZd_PI_0038 kap. 4.1.2.1 + VP_D; EG.D DŘT DECE v04 §3.2
Q (% ΣPi)	fs_rgQL1, QL2, Q0, QC2, QC1	+37,5 / +18,5 / 0 / −18,5 / −37,5 %	EG.D §2.8: QL375 = 0,375·Pinst ind., QL185, Q0 (cos φ = 1), QC185, QC375; ±10 %
cos φ	fs_rgCosL1, L2, 1, C2, C1	0,94 / 0,98 / 1,00 / −0,98 / −0,94	ekvivalent Q stupňů při P = Pinst (cos(atan 0,375), cos(atan 0,185)); PPDS 2025 příl. 4 §9.2.1 vyžaduje schopnost 0,90 ind.--0,90 kap.
nabíjení BESS	fs_BESSCh1…4	0 / 25 / 50 / 100 %	EG.D §4.2 a DOB/DSR v04: IP1--IP4, IP4 základní stav

Znaménko v NPA: **+ induktivní** (odběr Q z DS, zpožděný), **− kapacitní** (dodávka Q do DS). EG.D zapisuje induktivní hodnoty se záporným znaménkem – při přenosu hodnot ze smlouvy zkontrolujte orientaci. Vybíjecí stupně fs_BESSdCh0…4 distributoři nedefinují, výchozí hodnoty se pro ně nedoplňují.

Zdroje: ČEZ Distribuce *Požadavky na regulaci výroben* (ČEZd_PI_0038, účinnost 25. 5. 2022, cezdistribuce.cz/file/edee/distribuce/provozni-instrukce-regulace.pdf), EG.D *Podklady pro dispečerské řízení a chránění DECE 100--1000 kW* v04 (egd.cz/…/podklady-pro-dispecerske--rizeni-a-chraneni-dece-od-100kw-do-1000kw-v4_final.pdf), EG.D *Podklady pro dispečerské řízení DOB a DSR od 250 kW* v04, PPDS 2025 příloha 4 (cezdistribuce.cz/file/edee/distribuce/ppds/ppds--2025_priloha-4.pdf). Tlačítkem **Nastavit** u každého stupně lze povel ručně otestovat (setlocationlevelp). Pole **Manuální nastavení výkonu** (lo_rgGlobal, povel „Px") nastaví místo ručně na zadané % (setlocationpower).

Změna procenta u **právě aktivního** stupně (v místě, v *Nastavení výrobny* nebo u zařízení) se projeví **hned po uložení** – nový povel z dispečinku není potřeba. Ručně zadaný výkon (*Manuální nastavení*, povel P_PR0C) uložení nepřepíše (od 2.15.beta9).

Základ procent místa – instalovaný / fyzický výkon (lo_pwrBase, od 2.15.beta17): šoupátko nad sloupcem *Nastavení „P" Rozpadového místa* určuje, z čeho se počítají procenta stupňů **P0--P4 i Manuálního nastavení Px** místa (jedna společná funkce NState::ProcMistaNaFyzicky):

lo_pwrBase	Význam	Povel na měniče
phys (výchozí, chybí = phys)	% fyzického výkonu = součet dd_Power měničů místa	beze změny (dosavadní chování)
inst	% instalovaného výkonu místa (lo_insPower)	% × lo_insPower / fyzický, oříznuto 0--100 %

Bez vyplněného lo_insPower (nebo bez fyzického výkonu) se nepřepočítává. Přepočet se uplatní u zařízení (PozadovanyVykon, Device::DsrStropPx) i v souhrnu místa (PozadovanyVykonLocation,

Přehled); v souhrnu řídicích funkcí místa je u povelu doplněno „z instalovaného výkonu = X % fyzického“. Netýká se stupňů *Nastavení výrobny* (globální povel jde přes PI regulátor), povelů nastavených přímo na zařízení, výstupu PI regulace ani stupňů BESS. Zkoušky dle RfG nesou vlastní základ u povelu (`PowerLimitBase` v `state.cnf`, viz `zkousky-rfg.md`); jakýkoli jiný povel místa tento klíč smaže.

2.4 Priorita vypínání (`lo_regPwrPriority`)

Když je nutné omezit výkon, místa se obsluhují podle priority (0 = nejvyšší přednost zůstat v provozu):

- **Zapíná se od nejnižšího čísla, vypíná se od nejvyššího.**
- Místo s **nižším číslem** priority dostane výkon dřív a **vypíná se jako poslední**.
- Místo s **vyšším číslem** se **omezuje/vypíná jako první**.
- Místa se stejnou prioritou a stejným `lo_regDev` tvoří **skupinu** regulovanou společně; spojitá místa se v rámci stejné priority řadí za stupňová (Px).

2.5 Obchodní flexibilita – řízení činného výkonu z externího PLC

> **Menu:** *Nastavení výrobny* → **Obchodní flexibilita** (globální registry) a *Rozpadová místa* → **nastavit** u místa → **Obchodní flexibilita** (registr místa)

Externí PLC (obchodník, agregátor flexibility) zapisuje povely do **Modbus serveru NPA** (port 502 / RTU slave). Do 2.15.beta11 existovala u místa volba `lo_regExtPLC` (Neaktivní / v % / v kW) – zrušena, nahrazena fajfkami *Akceptovat vzdálené řízení*.

Registr	Klíč ve stavu	Fajfka (config)	Význam
2636	`DS_power_value`	`fs_extPlcPrcAccept`	globální omezení činného výkonu [%] – podřizují se mu **všechna** rozpadová místa
2634	`Power_export_limit` (float, 2 reg.)	`fs_extPlcKwAccept`	dynamické omezení rezervovaného výkonu výrobny (přetoků) [W]; **„−“ dodávka do sítě, „+“ odběr ze sítě**
3000 + 200·i + 44 (3044, 3244, 3444, …)	`fs_DS_power_value` v místě	`lo_extPlcAccept`	omezení činného výkonu i-tého rozpadového místa [%]; i = pořadí místa v seznamu (od 0)

Pravidla:

- Fajfka **chybí nebo `true` = akceptovat** (výchozí). Bez fajfky se hodnota registru **neuplatní** (chová se jako 100 %), zůstává ale uložená a zobrazená – přepnutí fajfky působí okamžitě bez nového zápisu z PLC.
- Z hodnot **2636 a 3x44** platí pro místo **nižší** (`NState::ExtPlcProcLocation`).

- Ze všech řídicích funkcí činného výkonu – **PI regulátor** výrobny, **povely DSŘ Px / PowerLimit** (globální, na místo i na zařízení) a **obchodní flexibilita** – má nejvyšší prioritu **nejnižší hodnota %**. Klamp je v `NState::OmezMistoProc` (společně s limitem přívodu místa 2.7), kterým prochází každý povel v `NState::PozadovanyVykon`, `PozadovanyVykonLocation` i bypass ramp `Device::DsrStropPx` – pokles hodnoty z PLC se proto stáhne rychle bez ohledu na rampy, stejně jako povel Px.
- Ve skupinách regulace (`GroupSetPower`) se klamp uplatní už při rozdělování výkonu – neodebraný výkon **přeteče na další prioritní skupiny** (jako u limitu přívodu).
- **BESS:** omezuje i **vybíjení** (kladný povel, % jmenovitého výkonu); nabíjení bez vlivu.
- **2634** se nepřepočítává na %: je to další kandidát **povoleného přetoku** v regulaci přetoků (`RegulacePodlePretoku`), povolený přetok = −hodnota registru (záporná hodnota = smí dodávat, kladná = musí odebírat ze sítě). Bere se přísnější z SoP rezervovaného výkonu, nastaveného přetoku, spotových cen a této hodnoty.
- Hodnoty registrů **nemají expiraci** – po výpadku PLC platí poslední zapsaná hodnota, dokud ji PLC nezmění nebo se fajfka nevypne.
- Během zkoušek dle RfG (pevné `PowerLimit` míst) klamp platí také; před zkouškou flexibilitu případně vypněte fajfkou.

Zobrazení: *Nastavení výrobny* ukazuje aktuální hodnoty 2636 / 2634; editace místa ukazuje hodnotu 3x44, obě globální a **souhrn řídicích funkcí místa** (PI regulátor `reg_proc`, povel DSŘ na místo, flexibilita `extplc_proc`, max. výkon z místa, výsledný povel) – vítězná (nejnižší) položka je zvýrazněná. *Přehled* v poli „ModBus řízení externím PLC“ ukazuje `kW, %` z 2634 / 2636, s příznakem „(vyp.)“ při vypnuté fajfce.

2.6 Vypnutí ovládání zařízení (`dd_ctrlEnable`)

Menu: *Detekce zařízení* → **nastavit** u zařízení → **Ovládání zařízení**

Volba je u každého ovladatelného zařízení (měnič, BESS, podřízená NPA, modul s reléovými výstupy). Čistě měřicí zařízení (elektroměry) ji nemají.

Volba	Klíč	Význam
Povoleno	chybí nebo `true`	**Výchozí stav.** Zařízení se účastní regulace a NPA do něj zapisuje řídicí registry.
Vypnuto (jen čtení)	`false`	Zařízení je vyřazeno z regulace a NPA do něj **nezapíše žádný registr**.

Co „vypnuto” znamená v běhu:

- `PozadovanyVykon()` i `PozadovanyVykonBESS()` vrátí `NaN`, takže zařízení nedostane povel;
- **všechny** zápisy zařízení jsou zablokovány společně v `DeviceModbus::WriteReg*` – tedy i cos φ, jalový výkon, start/stop, přepínání režimů a reléové výstupy, ne jen činný výkon;
- u BESS zařízení navíc vypadne z rozdělování společného povelu mezi baterie (`BessSharePower`), aby o svůj podíl neubíralo ostatním;
- **měření běží dál** – hodnoty se čtou, ukládají do stavu, zobrazují a posílají na Modbus i do dálkového logu. Vypnutí se týká výhradně zápisů.

Stav se vypisuje v detailu zařízení („Ovládání vypnuto – zařízení je jen ke čtení…”) a v měření jako položka `ovladani`.

Typické použití: měnič v servisu, zařízení řízené cizím systémem, nebo dočasné vyřazení jednoho kusu z regulace bez mazání z konfigurace.

2.7 Maximální výkon z rozpadového místa (`lo_maxOutPwrEn`, `lo_maxOutPower`)

Menu: *Rozpadová místa* → **nastavit** u místa → **Max. výkon z rozpadového místa**

Omezení výkonu tekoucího **ven** z rozpadového místa – typicky kvůli poddimenzovanému kabelovému přívodu do budovy. Zadává se v kW + zapínací fajfka (výchozí **vypnuto**); v konfiguraci se hodnota ukládá ve W (`lo_maxOutPower`, jako `lo_insPower`).

Sémantika:

- Výstup místa se měří **DS elektroměrem místa** (zařízení s `dd_meas="DS"` v místě, kladné = výstup z místa). Zdroje v místě smí dál **krýt vlastní spotřebu místa** – omezuje se jen to, co teče ven: povolená výroba = aktuální výroba + (limit − měřený výstup). Strop se přepočítává každý tick regulace (500 ms); dolů okamžitě (ochrana kabeláže), nahoru omezeně (`NR_MAXOUT_UP_PROC_TICK`, tlumení – měření se opožďuje za povelem). Limit je suma tří fází, per-fázově se neřeší.
- **Bez DS měření v místě** není zpětná vazba – limit se uplatní **staticky na výrobu** (vlastní spotřeba místa se nekryje) a konfigurace hlásí varování. Na sítích, kde jsou jako DS označené samotné měniče (výstup ≈ výroba), vyjde totéž statické chování – neoznačujte ale elektroměr **i** měniče zároveň (`GetDSFloat` by sčítal obojí).
- **Ostatní místa nejsou dotčena**: neodebraný výkon zůstává v rozpočtu výrobny a přeteče na další prioritní skupiny (stejné chování jako override `fs_DS_power_value`). Uvnitř jedné skupiny (stejná priorita) se ušetřený výkon nedělí – přeteče až do další skupiny.
- Limit **přežije i povely DŘ/Modbus** (`Px`, `PowerLimit` per-zařízení i per-místo) a místa „Bez regulace" (`lo_regDev="none"`) – je zapojen jako klamp v `NState::PozadovanyVykon` a zrcadlí se do bypassu ramp (`DsrStropPx`), takže pokles stropu se stáhne rychle bez ohledu na rampy.
- **BESS v omezeném místě:** vybíjení (kladný povel) omezuje strop výstupu. **Nabíjení** (záporný povel) omezuje tentýž limit v opačném směru – odběr z objektu do místa přes stejný přívod: povolené nabíjení = aktuální nabíjení BESS + (limit − měřený odběr). Strop nabíjení se publikuje jako `maxin_cap_proc` [% jmenovitého výkonu BESS místa] (uniformní procento na všech BESS místa dá v součtu přesně povolený výkon), stejné tlumení jako u výstupu (dolů okamžitě, nahoru `NR_MAXOUT_UP_PROC_TICK`). Bez DS měření staticky na hodnotu limitu; bez BESS v místě se nepublikuje. Ostatní zátěže (wallbox, TUV, …) limit neřídí. Pokles stropu nabíjení spouští bypass rampy nabíjení (`dd_rampChIgnDsr`) stejně jako povel Chx. Platí i pro místa typu BESS (mimo prioritní skupiny).
- Aktuální strop je vidět v ladicím výpisu Regulace2 (řádek `maxout …`) a v živém stavu místa jako `maxout_cap_proc` [%].

---

3. Povely a omezení

Menu: Nastavení výrobny (spotové ceny jsou dole v *Nastavení placených funkcí*).

Zdroje požadovaného výkonu:

3.0 SoP – Celkový instalovaný výkon

Pole	Klíč	Význam
SoP – Celkový instalovaný výkon	`fs_sopTotalPwr`	celkový instalovaný výkon výrobny dle Smlouvy o připojení [W] – **jen ke čtení**, dopočítává se automaticky

Jak se hodnota počítá nyní: pole je v menu zablokované (`disabled`) a nezadává se ručně. Hodnota se dopočítává v C++ a neukládá jako **součet „Instalovaný výkon" (`lo_insPower`) všech rozpadových míst** (s fallbackem na skutečný činný výkon místa, tj. Σ `dd_Power`, pokud je `lo_insPower` = 0).

> **Změna od verze 2.13.beta18 (2026-06-11).** Dříve šlo o **ručně editovatelnou** položku: uživatel zadával celkový instalovaný výkon do pole a hodnota se ukládala do configu pod klíčem `fs_sopTotalPwr`. Regulace ji ale nikdy nečetla – pro výpočty vždy

používala `GetInstalovanyVykon()` (součet `lo_insPower`), takže ručně zadaná hodnota mohla být s realitou nekonzistentní. Od beta18 je pole jen ke čtení a ukládaný klíč `fs_sopTotalPwr` se navíc **při načtení configu z configu maže** (hodnota se vždy dopočítá), aby v něm nezůstávala zastaralá ručně zadaná data.

3.1 Povely distribučního dispečinku (DŘ)

Povel chodí z dispečinku po sběrnici/Modbusu, v menu se nezadává. Mapování úrovní na % se nastavuje u jednotlivých míst (viz 2.3).

- Povel `P0…P4` → `PowerLimit` v % → **pomalý** požadovaný výkon = `PowerLimit% × instalovaný` výkon.
- Mapování úrovně na % řeší každé místo přes `lo_rgP0…4` (viz 2.3).
- `PowerLimit` je kopie procenta aktivního stupně; po každém uložení konfigurace se přepočítá z aktuální hodnoty, takže úprava `fs_rgP*` / `lo_rgP*` / `dd_regPwrP*` zabere okamžitě (od 2.15.beta9).

3.2 Omezení přetoku do sítě

Menu: Nastavení výrobny → *Nastavení hodnot omezení činného výkonu výrobny* a *SoP*.

Pole	Klíč	Význam
Systémové řízení přetoků	`fs_flwRestsw`	zapnutí regulace přetoku (`true`/`false`)
Povolený přetok	`fs_flwRest`	maximální dovolený přetok do DS [W]
Celkový rezervovaný výkon (SoP)	`fs_sopRsrvPwr`	**smluvní maximum** přetoku – nejvyšší priorita, aplikuje se **vždy** (i při vypnutém `fs_flwRestsw`)
SoP jako ¼h průměr (podle NTP)	`fs_sopRsrvPwrQ`	chybí nebo `true` = **výchozí**: SoP platí pro **čtvrthodinový průměr** přetoku (tak měří distributor). Z energie `E_del` TML od začátku čtvrthodiny (900 s od epochy UTC) se počítá povolený okamžitý přetok = (SoP/4 · 0,98 − vyvezeno) / zbývající čas (`NQuarterLimit`, `NR_SOP_Q_MARGIN`); může dočasně SoP překročit a klesá k 0 při vyčerpání. Po startu v běžící čtvrthodině se do její první celé čtvrthodiny počítá s okamžitým limitem; bez energie `E_del` TML platí okamžitý limit. `false` = SoP jako okamžitý limit (chování do 2.15.beta11). Stav je vidět v Nastavení výrobny u fajfky a ve výpisu Regulace
Měření	`fs_mRs`	`fs_measPhase` = regulace po fázích (najde fázi s největším přetokem), jinak podle sumy P
Řídící povel obchodníka / dodavatele energie	`fs_contractCmd`, `fs_contractCmdPlimit`	ruční limit přetoku [W] (od 2.15.beta12): platí s fajfkou; je-li přiřazen binární vstup s funkcí `Power_export_setting`, jen při jeho hodnotě 1 (bez vstupu trvale). Výstupní funkce `Power_export_setting` = 1, když limit právě platí. Přehled: „Řídící povel obchodníka“ [kW]

Pole	Klíč	Význam
Řídící povel Distribuce	`fs_distCmd`	přijímat Modbus registr **2510** `Power_export_reserved` [W] („−“ dodávka, „+“ odběr) jako dynamické omezení přetoku z dispečinku (chybí nebo `true` = přijímat); potvrzení 2512 = 0 bez požadavku / 1 přijat / 2 splněn. Hodnota je v Nastavení výrobny jen ke čtení

Přetok = záporná spotřeba na hlavním elektroměru (TML). Regulátor drží přetok na/pod povolenou hodnotou → **rychlá** větev.

Sledování výroby (bez skoku po mraku). Uplatní se jen v rychlé větvi, a to jen když přetok opravdu omezuje, tj. požadavek rychlé větve je pod instalovaným výkonem (od 2.15.beta15; bez omezení strop jen brzdil výrobu). Povel měničů smí předběhnout skutečnou výrobu jen o rezervu náběhu 2 % jmenovitého výkonu za tick 0,5 s (4 %/s, `NR_TRACK_UP_PROC`, bez konfigurace). Strop se počítá každý tick znovu ze skutečné výroby (`NRegulation2::VyrobniStrop`), takže se při mraku nesčítá: PI se nevytočí na 100 % a po odkrytí měniče nabíhají řízeně místo skoku = přetok. Protože se povel dělí měničům rovnoměrně v %, strop se počítá v % podle měniče s **nejvyšším poměrem**: měnič **na povelu** (P ≥ 95 % povelu, kritérium jako kompenzace výpadku části výrobny, `KompenzacniRezerva`) přispívá povelem, měnič **pod povelem** (mrak, vada) poměrem P/Pn; k maximu se přičte rezerva a výsledek se vynásobí součtem jmenovitých výkonů. Slabý nebo vadný měnič tak nepřiškrtí zdravé (do 2.15.beta14 se sčítaly W, strop dával průměrný poměr a povel kmital pilou, např. 42 → 57 %). Strop se slučuje s oknem ramp (`NastavTrack0kno`, integrál nad oknem se mrazí). Ladicí výpis Regulace: řádek `vyrobni strop` (včetně rozhodujícího měniče), stav `regulace/vyrobni_strop` [W], 0 = neaktivní.

Kde je řetěz vidět. Záložka webu **Tok energie** (`tok_energie.html`, od 2.15.beta15) kreslí toky síť → objekt → vlastní spotřeba objektu / rozpadová místa → zařízení a pod tím průchod regulací: kandidáty povoleného přetoku (vítěz zvýrazněn), povel distribuce, aktivní větev, výstup PI, výrobní strop, strop distribuce + kompenzaci, skupiny a omezení míst, BESS a cos φ. Nahoře souhrn, co právě omezuje (`NRegulation2::TokJson`, pole `omezuje`), a zařízení s rampou, nepotvrzeným povelem nebo trvalým deficitem.

Stav povoleného přetoku. Výsledek `RegulacePodlePretoku` (nejpřísnější z nastavení výrobny, spotových cen, SoP, PLC 2634, povelu obchodníka a DSR 2510) se publikuje jako `regulace/pretok_povoleny` [W, + = smí dodávat; při regulaci po fázích limit na fázi], `null` = přetoky se neregulují. Graf TML (ikona u řádku TML v Přehledu) ho kreslí jako **nastavený výkon** = −přetok v konvenci TML (+ = odběr), spolu s celkovou výrobou (`GetPositiveVal("DS", "P")` = `mAktualniVykonElektrarny`) a vlastní spotřebou (výroba + P TML = `mVlastniSpotreba`), obě aktuální i z energií za 1/4 h (`NHistory`, klíč VYR).

Nulový přetok a maximum výroby. Při `fs_flwRest` = 0 je cíl regulace přesně vlastní spotřeba, takže každé kolísání zátěže je z poloviny přetok a odběr nikdy dřívější přetok „neodečte” (elektroměr počítá dodávku a odběr zvlášť). Co pomáhá: sledování výroby výše (žádný skok po mraku), rampy zařízení (nahoru pomalu, dolů bypass), konstanty `fs_reg2` podle zpoždění smyčky (bess.md 4.7) a případně malý záporný povolený přetok (rezerva do odběru) úměrný kolísání spotřeby. Čtvrthodinový průměr SoP při nule nic nekompenzuje.

BESS a nulový přetok. Ve výrobně s `fs_flwRest` = 0 rychlá větev omezí střídače dřív, než se přebytek stihne uložit do baterie – režim „Nabíjecí limit přetoků“ (bess.md 2.1) totiž čeká na přetok, který nikdy nenastane. Pro tyto výrobny slouží režim **„Přednostní nabíjení BESS z omezené výroby (saldo)“** (`fs_saldoBESS`, od 2.15.beta9): když rychlá větev drží povel skupin FVE pod 98 % (a nejde jen o ořez výrobním stropem – mrak, vadný měnič), baterie po rampě (`fs_saldoBESSRamp`, výchozí 120 s na 0→100 %) zvyšuje nabíjení, tím roste vlastní spotřeba a regulátor přetoku sám povolí střídačům víc. Strop FVE se nezvedá, přetok do sítě nevzniká. Podrobně bess.md 2.2.

3.3 Spotové ceny

Menu: Nastavení výrobny → *Nastavení placených funkcí* (SPOT ceny).

Pole	Klíč	Význam
Tarifní limit	`fs_TarifPacLimit`	zapnutí řízení podle spotových cen
Cena nákupu (distribuce)	`fs_TarifPacLimitRate`	[Kč/MWh]
Cena prodeje (obchodník)	`fs_TarifRedemptionPacLimitRate`	[Kč/MWh]
Regulovat na	`fs_TarifPacLimitValue`	cílový přetok při nevýhodné ceně

Při nevýhodné výkupní ceně se přetok omezí (nebo se při záporných cenách výkon stáhne na 0 – „vyplatí se i mařit"). Bere se vždy **přísnější** z nastaveného přetoku a přetoku podle cen.

3.4 Rampy – gradient změny činného výkonu

Menu: Zařízení → *nastavit* (u zdroje / BESS), nebo rámeček **Hromadné nastavení ramp** pod seznamem zařízení.

Rampa omezuje, jak rychle se smí **měnit povel** činného výkonu poslaný do zařízení (předpis PPDS na gradient změny výkonu). Zadává se jako *x % za y min* jmenovitého výkonu zařízení (`dd_Power`) a zapíná se fajfkou; bez fajfky je změna bez omezení.

Položka	Klíče	Platí pro
Gradient zvyšování / snižování činného výkonu zdrojů	`dd_rampUp{Pct,Min,En}`, `dd_rampDn{Pct,Min,En}`	zdroje (měniče FVE, KGJ…)
Ignorovat rampy jen pro povely Px z DSŘ	`dd_rampIgnDsr`	zdroje
Gradient zvyšování / snižování – BESS vybíjení	`dd_rampDchUp{…}`, `dd_rampDchDn{…}`	BESS
Ignorovat rampy jen pro povely Dchx z DSŘ	`dd_rampDchIgnDsr`	BESS
Gradient zvyšování / snižování – BESS nabíjení	`dd_rampChUp{…}`, `dd_rampChDn{…}`	BESS
Ignorovat rampy jen pro povely Chx z DSŘ	`dd_rampChIgnDsr`	BESS

Jak to funguje:

- Rampa sedí až **za regulátorem** – PI regulátor počítá cíl stejně jako dosud, driver dostane hodnotu, která se k cíli blíží nastavenou rychlostí. V detailu zařízení je stav vidět v řádku *Rampa zdroje / Rampa BESS* (`cil … %, povel … %`).
- Aby se regulátor při pomalé rampě „nevytočil" a po jejím dojetí nepřestřelil, sleduje PI rampu (*tracking anti-windup*): smí ji předběhnout nejvýš o 2 % instalovaného výkonu. V debug výpisu Regulace2 je to řádek *tracking okno PI …*. Bez aktivních ramp se nic nemění.
- Po startu NPA (nebo po obnovení povelu) rampa **startuje z aktuálně měřeného výkonu**, takže povel neskočí.
- U BESS je + vybíjení a − nabíjení. Přechod přes nulu se dělí: nejdřív *snižování* jednoho režimu k nule, potom *zvyšování* druhého – každé svou rychlostí.
- **Ignorovat rampy jen pro povely z DSŘ:** když přijde z dispečinku povel k omezení (pokles stupně Px, resp. Dchx / Chx), provede se snížení **bez rampy**, tj. stávající maximální rychlostí PI regulátoru. Ostatní změny (přetok, spotové ceny, návrat nahoru) rampě podléhají dál. Výchozí stav: bez fajfky – rampy platí i pro povely z DSŘ.
- **Hromadné nastavení ramp** zapíše stejné hodnoty do všech zdrojů (položky zdrojů) a všech BESS (položky BESS). Nad formulářem je informace, zda mají všechna zařízení stejné nastavení, nebo upozornění *různé nastavení na zařízeních* (s výčtem, která se liší).

Pomalá rampa prodlužuje reakci na přetok: po dobu sjíždění výkonu trvá vyšší dodávka do sítě. Pro místa s tvrdým limitem přetoku zvažte rampu snižování vypnout nebo nastavit rychlejší než rampu zvyšování.

Jalový výkon / cos φ a BESS mají vlastní regulátory (reg3, reg4) a jsou popsány v samostatných dokumentech (BESS viz `bess.md`). V této příručce je neřešíme.

3.5 Aktuální povel ve stavu (`akt_proc`) – co se ukládá a odesílá

Sloupec **Aktuálně [%]** v Přehledu (Rozpadová místa, Měniče, BESS) se ukládá do živého stavu (`NState::mM`) každého rozpadového místa a každého ovladatelného zařízení, takže je i v datech odesílaných do cloudu (`m[<uuid>]`, viz `interfaces.md` §7) a v diagnostice (Info → *Kopírovat diagnostiku*). Regulace počítá povel **po fázích**, proto se ukládá vektor i přepočtený skalár:

Klíč	Kde	Význam
`pozadovane_proc`, `pozadovane_proc_1..3`	místo	výstup regulace pro místo (skalár + fáze L1..L3) **před** povely DŘ / Modbus (`PowerLimit`, stupeň Px)
`akt_proc`, `akt_proc_1..3`	místo	výsledný povel místa = hodnota v Přehledu (včetně `PowerLimit` / stupně Px); fáze = tvar z regulace přeškálovaný na skalár
`akt_proc`, `akt_proc_1..3`	zdroj (`pwr`)	povel skutečně posílaný do měniče **po rampě**; měnič s jedním setpointem dostane `akt_proc`, fáze nesou tvar, který chce regulace
`akt_proc_bess`, `akt_proc_bess_1..3`	BESS (`bess`)	povel BESS po rampě, + vybíjení / − nabíjení; fáze přímo z regulace BESS (např. režim saldo); hybridní měnič (FVE + baterie) má oba klíče
`<klíč>_timed`	místo i zařízení	`{max_v, max_t, min_v, min_t}` – extrémy s časem (UTC) od posledního odeslání do cloudu, stejně jako u `P`, `I`, …

Konvence: fáze jsou v **% jmenovitého výkonu fáze** (třetina `dd_Power`, resp. výkonu místa), vyvážený povel 60 % = `60 / 60 / 60`; skalár je průměr fází. Bez povelu (vypnuté ovládání `dd_ctrlEnable`, chybějící regulace) se klíče **odstraní**; u zařízení zmizí i při výpadku komunikace (stav se maže). KGJ Tedom se řídí stupněm, ne procenty, klíče nemá. Detail zařízení ukazuje cíl regulace a posílaný povel v řádku *Pozadovany vykon (cil regulace)*.

---

4. Nastavení regulátorů

Menu: Nastavení výrobny → dole *Nastavení placených funkcí* → tabulka *Nastavení regulátoru*. Konstanty mají u sebe nápovědu (najetím myší na pole). Editace jen pro účet s vyšším oprávněním.

Činný výkon řídí **dva** PI regulátory (další dva – reg3 pro cos φ a reg4 pro BESS – mají samostatné dokumenty):

#	Název v UI	Klíče	Co reguluje
1	Regulátor pro povely Px z DŘ	`fs_reg1_k`, `fs_reg1_i+`, `fs_reg1_i-`	pomalá regulace podle povelů distribuce

#	Název v UI	Klíče	Co reguluje
2	Regulátor pro přetoky do DS	`fs_reg2_k`, `fs_reg2_i+`, `fs_reg2_i-`	rychlá regulace přetoku

4.1 Význam konstant

Každý regulátor má tři konstanty:

- **`k` – proporcionální konstanta (P).** Působí **okamžitě** úměrně regulační odchylce.
  - `k = 0` → **P složka vyřazena**; odchylku koriguje **jen integrační složka** (nad rámec stále přítomné dopředné vazby na žádanou hodnotu, viz 4.2). Výchozí stav.
  - Malé k (řádově 0,001--0,01) už má silný vliv (odchylka je ve wattech, tj. velká čísla).
  - Příliš velké k → překmity až nestabilita.
- **`i+` – integrační konstanta pro stoupání** (odchylka > 0, tj. měření je **pod** žádanou hodnotou → výkon **roste**).
- **`i-` – integrační konstanta pro klesání** (odchylka < 0, tj. měření je **nad** žádanou hodnotou → výkon **klesá**).
  - Integrační konstanta je **časová konstanta**: čím **menší** číslo, tím **rychlejší** přibližování (jednotky = velmi rychlé, ale hrozí překmit; desítky = pomalé a stabilní).
  - `i = 0` → **integrační složka vyřazena a integrátor vynulován**.

  Asymetrie `i+` ≠ `i-` umožňuje např. **rychle stáhnout** přetok a **pomalu najíždět** zpět. Výchozí regulátor přetoků má `i-` = 5 (rychlý pokles) a `i+` = 20 (pomalý nárůst).

4.2 Vzorec regulátoru

Regulátory běží v každém kroku (0,5 s). Regulace činného výkonu (reg1, reg2) má tvar s **dopřednou vazbou** (výstup startuje na žádané hodnotě a PI ji jen koriguje):

```
odchylka      = žádaná - měřená
integrátor   += odchylka / i+     (když odchylka > 0)
integrátor   += odchylka / i-     (když odchylka < 0)

výstup = žádaná + (žádaná / y_max) * ( k * odchylka + integrátor )
```

y_max = maximální výkon na fázi (instalovaný výkon / 3).

Důležité vlastnosti:

- **Integrátor odstraní trvalou odchylku** – akumuluje se, dokud měření ≠ žádaná hodnota. Proto i při `k = 0` regulátor **dojede přesně** na cílovou hodnotu (přesné doregulování).
- **Při `i+` = `i-` = 0** je integrace vyřazena, takže `výstup = žádaná` – na výstup jde **přímo procento z povelu** a odchylka se **vůbec nedoreguluje** (okamžitý povel, viz 5.1 B).
- Při `žádaná = 0` (vypnutá regulace) a při nulovém měření se **integrátor resetuje** (jinak by „nafoukl" a po zapnutí způsobil překmit).
- **Anti-windup**: při saturaci výstupu (na y_min/y_max) se integrátor po fázích zmrazí.

---

5. Příklady s grafy

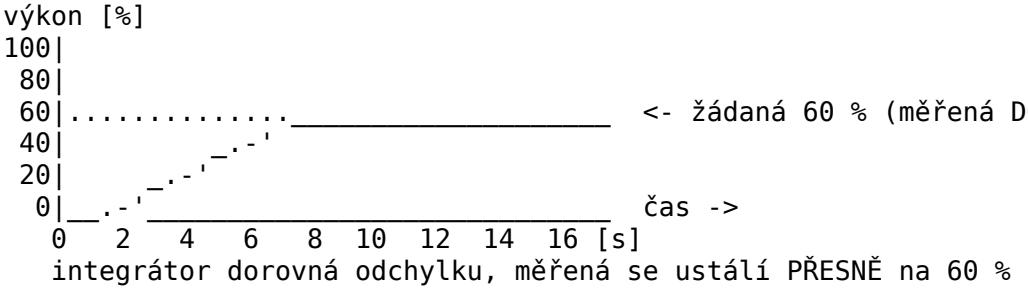
Grafy jsou ilustrační (perioda kroku 0,5 s). Žádaná hodnota = **60 %**, místo se rozbíhá z 0.

5.1 Přesné doregulování vs. okamžitý povel

Podle nastavení integračních konstant se regulátor chová **dvěma zcela odlišnými způsoby**. Rozdíl je v tom, **zda se vůbec řeší odchylka** mezi žádanou a skutečně naměřenou hodnotou.

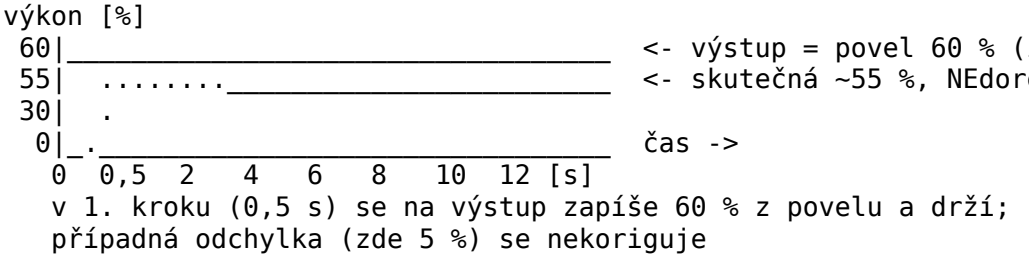
A) Přesné doregulování (zpětnovazební) Regulátor **sleduje odchylku** a integrační složkou ji postupně vynuluje – skutečně naměřený výkon **přesně dojede** na žádané procento (dorovná i ztráty, derating, nepřesnost měničů). Je o něco **pomalejší**, protože musí doregulovat.

```
fs_reg1_k  = 0      (P vyřazeno)
fs_reg1_i+ = 20     (doregulování nahoru)
fs_reg1_i- = 20     (doregulování dolů)
```



B) Okamžitý povel (rychlý, bez doregulování odchylky) Vyřaď i integrační složku (vše na 0). Pak `k·odchylka + integrátor = 0` a podle vzorce platí **výstup = žádaná**. Na výstup se v **prvním kroku ihned přepíše procento z povelu** a drží – **odchylka se vůbec neřeší** (žádná zpětná vazba). Nejrychlejší možná reakce, ale skutečně naměřená hodnota se **nedoreguluje** (zůstane, jaká vyjde).

```
fs_reg1_k  = 0      (P vyřazeno)
fs_reg1_i+ = 0      (integrace vyřazena)
fs_reg1_i- = 0      (integrace vyřazena)
```



> **Shrnutí:** `i+`/`i-` > 0 → **přesné doregulování** (měřená dojede na cíl, pomalejší); `i+` = `i-` = 0 → **okamžitý povel** (na výstup se ihned zapíše % z povelu, odchylka se neřeší). Proporcionální k nech v obou případech na 0.

5.2 Měniče s velkým dopravním zpožděním

Problém: měnič na změnu žádané hodnoty zareaguje až za několik sekund (dopravní zpoždění / mrtvá doba). Agresivní regulátor „nevidí" efekt svého zásahu, přidá ještě víc – a **přestřelí**.

Řešení: regulátor **zpomalit** – integrační konstanta musí být **výrazně větší** než dopravní zpoždění. P složku držet na nule:

```
fs_reg2_k = 0       (žádné okamžité skoky)
fs_reg2_i+ = 40     (pomalé, opatrné najíždění)
fs_reg2_i- = 30
```

```
ŠPATNĚ - rychlý regulátor (i+ = 3) na měniči se zpožděním 3 s:
výkon [%]
 90|      /\        /\
 75|     /  \      /  \      /\
 60|..../....\..../....\..../..\.....  <- žádaná 60 % (překmity)
 45|   /      \  /      \  /    \
 30|  /        \/        \/      \
  0|_/_____________________________  čas ->

SPRÁVNĚ - pomalý regulátor (i+ = 40) na stejném měniči:
výkon [%]
 60|............._____________________  <- žádaná 60 % (hladké)
 45|        _.-'
 30|     _.-'
 15|   _.'
  0|_.'_____________________________  čas ->
   0   2   4   6   8  10  12  14 [s]
```

Pravidlo: čím větší dopravní zpoždění měniče, tím **větší** `i+`/`i-` (pomalejší regulátor). Orientačně volte integrační konstantu několikanásobek mrtvé doby.

5.3 Co se stane při kmitání (špatné parametry)

Když je `k` příliš velké nebo `i` příliš malé vzhledem k dynamice soustavy (a zpoždění), regulátor se dostane do **trvalého kmitání** až nestability:

```
fs_reg2_k = 0,05   <- příliš velké
fs_reg2_i+ = 2     <- příliš rychlé

výkon [%]
100|    /\        /\        /\        /\
 80|   /  \      /  \      /  \      /  \
 60|../....\..../....\..../....\..../....\.  <- žádaná 60 %
 40| /      \  /      \  /      \  /      \
 20|/        \/        \/        \/        \
  0|_____________________________________  čas ->
   0   2   4   6   8  10  12  14  16 [s]
   amplituda neklesá -> systém NEDOJEDE, kmitá kolem žádané
```

Projevy kmitání: výkon i přetok „pendlují", měniče přepínají nahoru/dolů, na elektroměru oscilace.
Náprava:

1. Sniž `k` (ideálně na `0`).
2. Zvětši `i+`/`i-` (zpomalení integrace).
3. Pokud kmitání přetrvává, zkontroluj dopravní zpoždění měničů (viz 5.2).

```
Postupné tlumení po opravě (k=0, i+ zvětšeno z 2 na 25):
výkon [%]
 75|   /\
 60|../..\._.....____________________  <- žádaná 60 % (kmit se utlumí)
 45| /    V
 30|/
  0|_____________________________  čas ->
```

5.4 Asymetrie i+ / i- – rychlé stažení přetoku, pomalý náběh

Pro regulaci přetoku se hodí **rychle stáhnout** výkon (bezpečnost – nepřekročit limit) a **pomalu najíždět** zpět (klid sítě):

```
fs_reg2_k  = 0
fs_reg2_i+ = 20   (pomalý nárůst)
fs_reg2_i- = 5    (rychlý pokles)

výkon [%]
 80|_____                              <- výroba běží
 60|     \      <- PŘETOK! rychlé stažení (i- = 5)
 40|      \___
 20|          '------____              <- pomalé najíždění (i+ = 20)
  0|                    '-------____   čas ->
   0   2   4   6   8  10  12 [s]
```

---

6. Postup ladění (cookbook)

1. **Začni od továrního nastavení** – `k = 0`, integrace v desítkách (reg1 `i± = 20`, reg2 `i+ = 20`, `i- = 5`). Tento stav je stabilní a dojede přesně.
2. **Příliš pomalá reakce?** Postupně **zmenšuj** `i+`/`i-` (např. 20 → 10). Po každé změně sleduj překmit.
3. **Objeví se překmit/kmitání?** Vrať `i` zpět na vyšší hodnotu a/nebo nech `k = 0`. k zvyšuj jen výjimečně a po malých krocích (0,001 → 0,005 → 0,01).
4. **Měniče reagují se zpožděním?** Drž `k = 0` a `i` nastav na **několikanásobek** mrtvé doby (viz 5.2).
5. **Přetoky musí být přísné?** Nastav `i-` (pokles) výrazně menší než `i+` (nárůst) – viz 5.4.
6. **Zařízení neumí plynulé řízení?** Přepni místo na `lo_regDev = Px` a vyplň `lo_rgP0…4`.

Situace	k	i+	i-
Přesné doregulování (měřená dojede na cíl)	0	20	20
Okamžitý povel (ihned zapíše %, neřeší odchylku)	0	0	0
Rychlejší doregulování (svižné měniče)	0	8--12	8--12
Velké dopravní zpoždění měničů	0	30--60	30--60
Přísné přetoky (rychle dolů, pomalu nahoru)	0	20	5
⚠ Nestabilní (NEPOUŽÍVAT)	≥ 0,05	≤ 2	≤ 2

Zlaté pravidlo: pro přesnost a stabilitu nech `k = 0` a laď **jen** integračními konstantami `i+`/`i-`. Proporcionální složku přidávej jen, když potřebuješ rychlejší okamžitou reakci a víš, že soustava nemá velké zpoždění.

Zdrojové soubory

Část	Soubor / symbol
Hlavní smyčka regulace	`nregulation2.cpp` → `NRegulation2::Loop()`
PI regulátor (činný výkon)	`npicontroller.cpp` → `updateParam()`
Rozpad do míst, priorita, Px	`nregulation2.cpp` → `GetLocationPrioritySort()`, `GroupSetPower()`
Povely distribuce / přetoky / SoP / spotové ceny	`nregulation2.cpp` → `RegulacePodleDistribuce()`, `RegulacePodlePretoku()`, `RegulacePodleSpotCen()`
Regulace BESS (režimy 1--5, saldo = 5, míra omezení FVE `mPvCurtailProc`)	`nregulation2.cpp` → `BessRegulation()`, `GroupSetPower()`; konstanty `NR_SALDO_*` v `nregulation2.h`
Výchozí konstanty regulátorů	`nregulation2.h` → `NR_CONST[]`
UI – nastavení místa	`html/v2/location_edit.html`

Část	Soubor / symbol
UI – nastavení regulátorů	`html/v2/factory_settings_edit_paid_features.html`