

SRNE TOU arbitrage: plan (#67)

“ The plan to turn on time-of-use battery arbitrage on the Sezibwa/Nansana SRNE inverter (logger issue #67): charge the battery from the grid off-peak, discharge it to the load at peak, using the SRNE charge/discharge SCHEDULE registers written through the guarded write path we proved with #53. Compiled 2026-08-11.

The gap (why we are leaving money on the table)

The inverter already CHARGES off-peak (charge window is set to 00:00-06:00) but it does NOT discharge at peak - the discharge time-windows are all 0 (off). Live-confirmed 2026-08-10: at peak the battery sat idle on grid at ~79% while ~1.3 kW of load was served straight from the grid. So we pay the peak grid rate for load we could have covered with cheap off-peak energy already sitting in the battery. Turning on the discharge window closes that.

The economics (Code 10.2, what NFE pays the utility)

Period	Window	Rate (UGX/kWh)
Off-peak	00:00-06:00	343.20
Shoulder	06:00-18:00	577.11
Peak	18:00-24:00	666.50 (Q3 2026)

Arbitrage = buy at 343 off-peak, avoid ~666 at peak = ~320 UGX/kWh gross, minus battery wear. Usable battery ~10.49 kWh; discharging ~5-7 kWh at peak while holding a reserve gives a gross saving in the low thousands of UGX per day. The daily energy/cash summary (#26, already live) already computes charged kWh + buy cost, discharged kWh + value, wear, and net margin - so once this is on we MEASURE the real number, not guess.

The levers (SRNE registers, decoded firmware v925, RW)

Written through the same guarded path as #53 (SafeWriteHandler: default-off, verified-machine-state gate, one-shot FC16, readback-verified). Time encoding = hour*256 + minute (e.g. 18:00 = 4608, 06:00 = 1536).

Function	Register(s)	Live (2026-08-09)	Target
Charge window 1	0xE026 / 0xE027	00:00-06:00	keep (off-peak)
Charge windows 2/3	0xE028/E029, 0xE02A/E02B	unset	optional
Discharge windows 1/2/3	0xE02D-0xE032	ALL 0 (off)	window 1 = 18:00-24:00 (peak)
Output priority	0xE204	3	confirm = battery-first in-window
Charge priority	0xE20F	tbd	grid, off-peak
Switch-to-line SoC (reserve floor)	0xE01F	10	raise to protect reserve
Switch-to-battery SoC	0xE020	90	tune
Discharge cutoff SoC	0xE00F	5	hard floor
Inverter RTC	set 0xE034-0xE036 / read 0x020C-0x020E	must be correct	verify, set if drifted

NOT available on this firmware (confirmed Illegal-data-address): real-time active-power dispatch (0xE047-0xE04D, 0xE400-0xE402). We do NOT need it - scheduling covers TOU arbitrage.

Safety framing: reserve vs savings (the core tension)

The site's PRIMARY mission is backup during grid outages. Discharging at peak spends reserve exactly in the evening window when outages are most likely. So arbitrage must never drain below a backup reserve. Policy: hold $\geq 50\%$ reserve until 30 days of outage data are collected (per CUSTOM_CODE/CLAUDE.md), then revisit. The discharge must STOP at the reserve floor, not run down to the discharge-cutoff. Phase 1 confirms exactly which register enforces that floor on this firmware (the switch-to-line SoC band vs a per-window reserve). RTC accuracy is critical - a wrong clock means the windows fire at the wrong time.

The build (mirrors #53, which is proven)

Phase 1 - Prove the schedule-write path (DAYTIME, no customer disruption):

- Fresh bus-quiet mbpoll read of every schedule/RTC/reserve register above = current ground truth (stop edge + mbpoll, 5s settle, read, restart - the method from #53).
- Extend the SRNE OpenEMS component to expose the schedule + RTC + reserve registers as guarded-writable settings, reusing the SafeWriteHandler infra from #53.
- Safe live test: write an UNUSED, far-from-now discharge window (e.g. a mid-morning slot), verify the readback, restore it to 0. Verify RTC reads wall-clock; set if off. This proves writes land and read back without ever activating a real discharge in front of customers.
- Tests to the 90% coverage gate; ADR; PR; git -> deploy.

Phase 2 - Program the arbitrage schedule (DAYTIME, supervised):

- Through the guarded write config (gateway config -> deploy, no box edits): set discharge window 1 = 18:00-24:00, confirm charge window 1 = 00:00-06:00, set the reserve floor to protect 50%, confirm output/charge priority.
- Watch one full cycle live: does it discharge at peak (battery carries the load, grid draw drops)? does it stop at the reserve floor? does it recharge off-peak overnight?

Phase 3 - Measure + tune:

- The daily #26 summary reports the actual arbitrage margin each morning. Tune the window edges and reserve against real margin + the outage log (the 30-day rule before dropping reserve below 50%).

Guardrails (all carried from tonight's #53 discipline)

Live-test before commission. Daytime writes only. Guarded write path (default-off, verified-state gate, one-shot, readback-verified). git -> deploy, never hand-edit the box. Bus-quiet reads (stop edge + mbpoll + 5s). Reserve-first: backup mission beats savings.

Open questions Phase 1 must answer

1. Which register actually enforces the reserve floor on v925 (switch-to-line SoC vs a per-window reserve field)?
 2. Is the inverter RTC accurate right now? (windows fail silently if not)
 3. Output-priority (0xE204=3) semantics on this firmware - does it put battery first inside a discharge window while grid is present?
-

Revision #1

Created 2026-08-10 19:53:37 UTC by hillary.arinda

Updated 2026-08-10 19:53:37 UTC by hillary.arinda