

Sorveglianza Agnostica Multivariabile



Framework di analisi composita

Release 26.02.06

Copyright 2026

Tutti i diritti riservati

In copertina e nella premessa le immagini sono generate da AI

Release documento 260206



260206

Indice

Premessa Digitale e Letteraria

1. Score di scostamento

- 1.1 Esempio di scores da PWA "IceSense"
- 1.2 Differenza di media (Z-score)
- 1.3 Trend (pendenza)
- 1.4 Varianza (stabilità)
- 1.5 Forma della curva (RMSE)
- 1.6 Calcolo dello Score Finale
- 1.7 Modello di determinazione della baseline
- 1.8 Granularità dei dati e scalabilità degli score
- 1.9 Tempo di refresh dello score

2. Rilevamento deterministico

- 2.1 Invio dati all'assistenza tecnica
- 2.2 Invio dati all'AI

3. Sorveglianza complementare: Snapshot System

- 3.1 Funzioni utilizzate

4. Sorveglianza complementare: Snapshot Service

- 4.1 Calcolo e significato Service Score

5. Sorveglianza complementare: Snapshot Functions

- 5.1 Esempio di Snapshot Functions
- 5.2 Score MCU

6. Score Batteria (per unità con backup di energia)

- 6.1 Score finale e visualizzazione
- 6.2 Composizione dello Score Batteria
- 6.3 Formule di Calcolo

7. Applicazione del Modello di Sorveglianza alle unità di processo

- 7.1 Definizioni formali degli score specifici

8. Specifica tecnica per implementazione firmware

- 8.1 Obiettivo
- 8.2 Struttura del ciclo
- 8.3 Acquisizione dati
- 8.4 Normalizzazione temporale
- 8.5 Costruzione baseline
- 8.6 Aggiornamento della baseline
- 8.7 Snapshot tecnici
- 8.8 Risultato atteso

9. Esempi di Interfaccia con display EBTN

Premessa Digitale



Inquadra il QR per un'anteprima
multimediale del contenuto

Premessa Letteraria

La **sorveglianza agnostica** multivariabile nasce dall'esigenza di analizzare il comportamento reale dell'impianto senza ricorrere a soglie fisse, regole predefinite o modelli dipendenti dall'impianto. Nei sistemi fisici complessi non è possibile definire limiti universali o prevedere in modo deterministico l'insorgere di un guasto: ciò che invece è sempre osservabile è l'allontanamento progressivo di una variabile dal proprio comportamento storico. Il modello adottato si basa esattamente su questo principio: misurare lo scostamento dalla normalità statistica, non predire il guasto.

Lo score di scostamento fornisce un indicatore quantitativo, stabile e riproducibile dell'evoluzione delle grandezze monitorate, indipendentemente dal tipo di impianto, dalla variabile, dal contesto operativo e determina il livello di attenzione.

A differenza della sorveglianza agnostica, **l'analisi deterministica** estrae in tempo reale gli eventi di guasto, allerta localmente i soggetti operativi, e se l'unità è connessa invia al cloud lo stato di allarme. Ogni evento deterministico è quindi finalizzato a raggiungere rapidamente chi deve intervenire sul campo.

Tuttavia, uno scostamento numerico o uno stato di allarme non è sufficiente per avere un'immagine completa di tutte le funzionalità: serve un contesto tecnico che descriva lo stato del controllore, del firmware, e dell'applicazione. Per questo motivo il sistema integra un secondo elemento di analisi: **gli snapshot (Functions, System, Service)**, ovvero report tecnici strutturati che raccolgono e organizzano le informazioni settoriali necessarie per interpretare il funzionamento globale. Analisi agnostica, deterministica e snapshot costituiscono tre livelli complementari.

Gli score rappresentano la memoria tecnica dell'impianto: una lettura storica e progressiva del suo comportamento, costruita nel tempo attraverso l'osservazione delle grandezze fisiche e delle loro deviazioni dalla normalità statistica. Gli snapshot, al contrario, sono una fotografia istantanea, generata su richiesta, che cattura lo stato presente del controllore, dell'applicazione e dell'hardware. Se gli score raccontano l'evoluzione, gli snapshot raccontano il momento: insieme permettono di comprendere non solo ciò che l'unità è stata, ma ciò che è ora, offrendo una visione completa e coerente del suo funzionamento reale.

Il modello di sorveglianza mantiene la propria coerenza anche quando viene applicato a sistemi termici discontinui, come unità di processo (forni professionali). In questi casi la dinamica non è continua come nelle unità di conservazione, ma segmentata in fasi operative distinte. La baseline non è una curva unica, ma un insieme di firme termiche normalizzate, una per ciascuna fase. La logica statistica rimane invariata: ciò che conta non è il valore assoluto, ma la coerenza della dinamica rispetto al comportamento storico.

Allo stesso modo, la componente deterministica continua a intercettare gli eventi di guasto certo, mentre gli snapshot forniscono il contesto tecnico istantaneo. Il modello

rimane agnostico: cambia la natura del processo fisico, non il metodo con cui viene analizzato. Questa adattabilità consente di utilizzare lo stesso linguaggio tecnico per unità di conservazione, forni di cottura e, più in generale, per qualunque sistema fisico caratterizzato da una dinamica misurabile nel tempo.

A questi tre livelli si aggiunge **l'intelligenza artificiale**, che non esegue calcoli tecnici ma svolge una funzione di interpretazione linguistica. L'AI riceve un prompt strutturato, derivato dai dati filtrati e consolidati, e restituisce una spiegazione chiara e coerente delle anomalie rilevate, utile soprattutto per l'utente non tecnico. L'AI non sostituisce la diagnostica, ma la rende leggibile e immediatamente utilizzabile.

Il sistema di sorveglianza nel suo complesso crea un linguaggio tecnico comune tra tutte le figure coinvolte: l'utilizzatore sul campo, lo sviluppatore software, chi scrive il firmware, chi progetta l'hardware e il tecnico termodinamico che interpreta le grandezze fisiche

Ogni elaborato parte dal dispositivo fisico, passa attraverso l'utente tramite connessione NFC o tramite cloud, raggiunge le competenze software, firmware, hardware e termodinamiche, e ritorna sul campo sotto forma di interventi mirati. Il sistema non si limita a descrivere un singolo evento, ma contribuisce a individuare pattern, condizioni ricorrenti e anomalie utili al miglioramento continuo dell'intera flotta. L'elaborazione avviene interamente nel controllore. Per le unità connesse al cloud riceve solo indicatori già calcolati, evitando qualsiasi carico computazionale. La piattaforma funge da vettore di distribuzione e correlazione, non da motore di analisi.

1. Score di scostamento

Lo score è calcolato attraverso la somma di quattro controlli statistici, ciascuno progettato per individuare un tipo specifico di cambiamento nei dati analizzati rispetto a una baseline storica. Il modello adottato presenta caratteristiche fondamentali che ne garantiscono l'universalità:

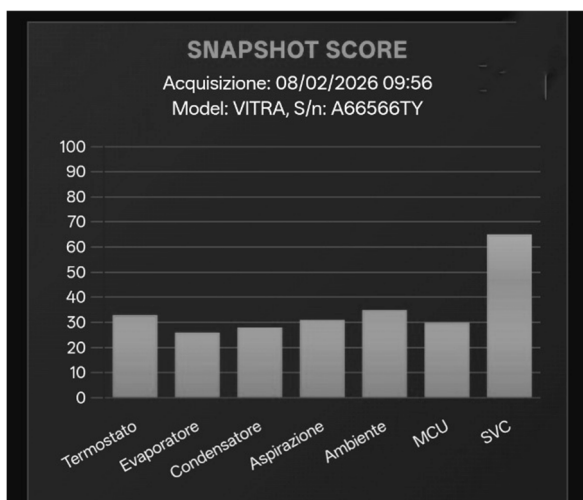
- Non richiede configurazioni specifiche per l'impianto
- Non dipende da limiti preimpostati o soglie fisse
- Si adatta a qualunque variabile fisica o elettrica, incluse quelle ricavate dal monitoraggio della MCU del controllore
- Mantiene coerenza e riproducibilità indipendentemente dal contesto operativo

Lo score non produce allarmi, ma segnali di attenzione: un indicatore oggettivo, stabile e trasversale, che permette di concentrare l'analisi sulle variabili realmente significative, migliorando la capacità di diagnosi precoce e riducendo il rischio di interpretazioni soggettive.

Per il modello di studio (PWA "IceSense") si è scelto un campionamento a bassa granularità: 1 campione ogni ora, pari a 24 campioni al giorno.

1.1 Esempio di scores da PWA "IceSense"

La variabile con maggior punteggio è, in questo caso, il condensatore; per il punteggio restituito (circa 70) merita attenzione; successivamente si potrebbe presentare un guasto che coinvolge lo scambio termico dell'unità attenzionata.



1.2. Differenza di media (Z-score)

Il primo controllo valuta quanto la media dei dati attuali si discosta dalla media della baseline. Maggiore è la differenza tra le due medie rispetto alla variabilità storica, più aumenta il contributo allo score. Il valore massimo attribuibile è pari a 100.

Formula:

$$Z = (\mu_{\text{current}} - \mu_{\text{baseline}}) / \sigma_{\text{baseline}}$$

Variabili:

- μ_{current} : media degli n valori attuali
- μ_{baseline} : media degli n valori della baseline
- σ_{baseline} : deviazione standard degli n valori della baseline

1.3. Trend (pendenza)

Il secondo controllo verifica se negli ultimi N punti esiste una tendenza chiara e significativa (in crescita o in diminuzione). La pendenza contribuisce allo score solo se supera una soglia minima, evitando così falsi allarmi. Più la pendenza è marcata, maggiore sarà il contributo.

Formula della pendenza:

$$\text{slope} = [N \cdot \sum(x_i \cdot y_i) - (\sum x_i)(\sum y_i)] / [N \cdot \sum(x_i^2) - (\sum x_i)^2]$$

Variabili:

- N = numero dei campioni
- x_i : indice del punto (da 1 a N)
- y_i : valore corrente al punto i

1.4. Varianza (stabilità)

Il terzo controllo confronta la variabilità dei dati attuali con quella della baseline. Se la variabilità si discosta significativamente dal riferimento storico, viene aggiunto un contributo allo score.

Formula: $R = \sigma_{\text{current}} / \sigma_{\text{baseline}}$

Variabili:

- σ_{current} : deviazione standard degli n valori attuali
- σ_{baseline} : deviazione standard degli n valori della baseline

Condizioni di attivazione: $R < 0.4$ oppure $R > 2.5$

Scostamento considerato: $|R - 1|$

Nota importante

I valori 0.4 e 2.5 definiscono l'intervallo di variabilità statisticamente coerente: al di sotto di 0.4 la dispersione si riduce a meno della metà, mentre oltre 2.5 risulta più che raddoppiata. In entrambi i casi la dinamica del segnale cambia natura rispetto alla baseline.

Il contributo allo score deriva dallo scostamento dalla normalità ($|R - 1|$), non dal superamento di una soglia operativa. In questo modo il modello rimane completamente agnostico: interpreta questi valori come indicatori matematici di quanto la variabilità attuale si discosti da quella storica, indipendentemente dal tipo di variabile o dall'impianto.

1.5. Forma della curva (RMSE)

Il quarto controllo confronta punto per punto la forma delle due curve (baseline e attuale), calcolando quanto la forma complessiva è cambiata. Un maggiore scostamento della curva attuale rispetto alla baseline aumenta lo score.

Formula RMSE:

$$\text{RMSE} = \sqrt{\left[(1 / N) \cdot \sum (y_{i,\text{current}} - y_{i,\text{baseline}})^2 \right]}$$

Normalizzazione: $\text{RMSE norm} = \text{RMSE} / \sigma_{\text{baseline}}$

Variabili:

- $y_{i,\text{current}}$: valore corrente al punto i
- $y_{i,\text{baseline}}$: valore baseline al punto i
- N = numero dei campioni
- σ_{baseline} : deviazione standard della baseline

1.5.1 Componente deterministica dello Score

In analogia con le variabili termodinamiche, allo Score viene aggiunta anche una componente deterministica, che intercetta condizioni non riconducibili a un'analisi statistica. Si tratta di eventi certi di malfunzionamento, rilevati tramite soglie fisse e ritardi temporali predefiniti:

- Alta temp. termostato (sopra soglia oltre il tempo di ritardo predefinito)
- Bassa temp. termostato (sotto soglia oltre il tempo di ritardo predefinito)
- Bassa temp. di evaporatore (sotto soglia predefinita con ritardo di 5 sec)
- Alta temp. di condensatore (sopra soglia predefinita con ritardo di 5 sec)
- Bassa temp. di aspirazione (sotto soglia predefinita con ritardo di 5 sec)
- Alta temp. di aspirazione (sopra soglia predefinita con ritardo di 5 sec)

Questi eventi generano un incremento diretto dello Score, poiché rappresentano condizioni di guasto certo e non interpretabili come semplici deviazioni statistiche.

1.6. Calcolo dello Score Finale

Ogni controllo statistico produce un contributo proporzionale allo scostamento rilevato, ma ciascun contributo viene pesato e poi limitato così che nessun singolo fenomeno possa saturare lo score. La somma dei contributi è a sua volta vincolata a un massimo di 100, mantenendo lo score sempre stabile e confrontabile. Un ammorbidimento finale riduce la sensibilità ai picchi e rende la crescita più progressiva, così lo score riflette l'allontanamento dalla normalità senza diventare eccessivamente reattivo.

Formula finale: $SCO = (\text{score grezzo} / 100)^{0.55} \times 100$

La baseline mantiene sempre gli N valori originali, essenziali per i confronti futuri e per la coerenza del metodo di analisi. L'esponente 0.55 è stato scelto per ottenere una curva di crescita progressiva ma non piatta.

1.7. Modello di determinazione della baseline

La baseline viene generata automaticamente utilizzando solo i periodi di funzionamento stabile, esclusi aperture porta, sbrinamenti e stati speciali. Il controllore identifica queste condizioni in tempo reale, garantendo che la baseline rifletta esclusivamente il comportamento normale dell'impianto.

Una volta definita, la baseline non viene riscritta continuamente, ma aggiornata in modo lento e controllato: i nuovi dati vengono integrati solo quando il sistema si trova di nuovo in condizioni operative stabili, mentre i campioni acquisiti durante aperture porta, carichi di prodotto caldo o sbrinamenti sono esclusi dall'apprendimento. In questo modo la baseline segue gradualmente l'evoluzione naturale dell'impianto (stagionalità, invecchiamento), senza inseguire i disturbi operativi.

Quando lo score indica uno scostamento prolungato e significativo dalla normalità, la baseline viene mantenuta ferma e utilizzata come riferimento per il calcolo degli score, evitando che il modello si adatti ad uno stato degradato. Dopo interventi strutturali sull'impianto o modifiche rilevanti, il sistema è in grado di ricostruire automaticamente una nuova baseline a partire dai nuovi periodi di funzionamento stabile, senza richiedere configurazioni manuali. In sintesi, la baseline non è un valore fissato a priori, ma un riferimento dinamico, appreso dal comportamento reale dell'impianto e continuamente mantenuto coerente grazie alle informazioni operative fornite dal controllore.

1.8. Granularità dei dati e scalabilità degli score

Nel modello concettuale usato per spiegare il funzionamento dello score, abbiamo adottato una rappresentazione semplificata basata su 24 campioni orari. È una scelta pensata per rendere immediata la comprensione della logica del metodo, senza dover gestire fin da subito la complessità operativa dei dati reali.

Nella pratica, però, i controllori operano con una granularità molto più fine: fino a 2.880 campioni al giorno, uno ogni 30 secondi. Questa densità informativa elevata non

rappresenta un vincolo, ma un vantaggio: lo score è un metodo completamente agnostico che si adatta automaticamente alla quantità di dati disponibili. Non dipende dalla frequenza di campionamento, non richiede soglie fisse né regole specifiche per impianto, e non ha bisogno di tarature manuali.

La granularità elevata migliora significativamente la qualità statistica delle medie, delle varianze, dei trend e della forma della curva. In altre parole, più dati arrivano, più lo score diventa robusto, stabile e affidabile, senza diventare eccessivamente reattivo o sensibile al rumore. Questa caratteristica lo rende particolarmente efficace in due contesti operativi molto diversi.

Nel caso dei controllori che comunicano via NFC, dove ogni scambio è limitato a circa 1Kb, non è possibile trasferire grandi serie temporali. Lo score risolve il problema alla radice: tutta l'elaborazione avviene localmente e ciò che viene trasmesso è solo il risultato finale, leggero e compatibile con i vincoli fisici della tecnologia NFC. L'operatore riceve un'informazione sintetica ma estremamente significativa, senza saturare la banda.

All'estremo opposto, nei sistemi connessi al cloud, la granularità elevata potrebbe trasformarsi in un problema di carico computazionale se fosse la piattaforma a dover elaborare i dati di ogni dispositivo. Il nostro approccio evita questo rischio: il calcolo dello score avviene sul controllore, non nel cloud. La piattaforma riceve solo indicatori già elaborati, senza dover eseguire regressioni, RMSE o analisi statistiche per migliaia di device. Il risultato è una scalabilità naturale, con costi computazionali minimi e prestazioni costanti anche su flotte molto numerose.

La combinazione tra alta granularità reale, calcolo locale, output leggero e totale agnosticità rispetto alle grandezze fisiche ed elettriche, incluse quelle ricavabili dalla MCU del controllore, rende lo score uno strumento affidabile, universale e industrialmente sostenibile.

Non predice il guasto, ma emula in modo credibile l'allontanamento dalla normalità, offrendo un vantaggio competitivo reale sia nei sistemi a bassissima banda sia nelle architetture cloud ad alta densità di dispositivi.

1.9 Tempo di refresh dello score

Nel modello concettuale è stato utilizzato un campionamento orario per semplificare la spiegazione del metodo. Nella pratica operativa, tuttavia, i controllori acquisiscono i dati con maggiore frequenza. In un contesto ad alta granularità, il tempo di refresh dello score assume un ruolo fondamentale per garantire stabilità, coerenza e robustezza statistica.

Lo score non viene ricalcolato a ogni campione: un aggiornamento così frequente renderebbe l'indicatore eccessivamente reattivo al rumore e alle oscillazioni fisiologiche del sistema. L'obiettivo del modello è invece misurare l'allontanamento progressivo dalla normalità, non le variazioni istantanee.

Per questo motivo il calcolo dello score avviene su una finestra mobile di N campioni confrontando la dinamica attuale con la baseline storica. L'aggiornamento dello score viene eseguito a intervalli regolari, tipicamente ogni 5–10 minuti, corrispondenti a 10–20 campioni. Questa frequenza rappresenta un equilibrio ottimale: sufficientemente elevata da cogliere evoluzioni significative, ma abbastanza lenta da filtrare il rumore e mantenere la stabilità dell'indicatore.

In sintesi la granularità fine migliora la qualità statistica dei controlli, mentre il refresh periodico evita reazioni impulsive e garantisce uno score affidabile, coerente, industrialmente utilizzabile e commercialmente onesto.

2. Rilevamento deterministico

Accanto al modello statistico degli score, il sistema integra un secondo criterio dedicato agli eventi improvvisi, cioè alle condizioni che richiedono una sorveglianza tecnica immediata (guasti effettivi). In questi casi, il controllo è deterministico, basato su limiti e range predefiniti dal costruttore. Ad ogni campionamento, il sistema verifica se i valori misurati rientrano nel range operativo previsto. Le variabili analizzate includono:

- Temperatura **termostato**: anomalia se il valore è fuori dal range previsto
- Temperatura **uscita evaporatore**: anomalia se il valore è fuori dal range previsto
- Temperatura **uscita condensatore**: anomalia se il valore supera la soglia max
- Temperatura di **aspirazione**: anomalia se il valore è fuori dal range previsto
- Temperatura **ambiente**: variabile esplicativa

Le anomalie consecutive relative alla stessa variabile vengono filtrate, mantenendo solo il valore estremo per evitare duplicazioni inutili.

2.1 Invio dati all'assistenza tecnica

E' il vettore preferibile per una risposta umana e competente; ovviamente necessaria per gli interventi sul luogo.

2.11 Esempio di snippet di guasto fornito dal controllore:

Acquisizione: 01/02/2026 09:50

Unità in esercizio: Via Severino Grattoni, 12 – Roma (Italy)

Meteo locale 14°- 93%

Model: VITRA, S/n: A66566TY Installato: 10/12/2025

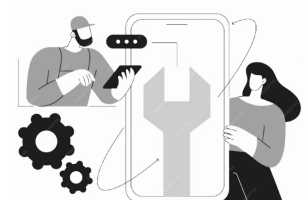
[CONFIGURAZIONE IMPIANTO]

Setpoint: 2°C | Isteresi: 4°C Limiti allarme: -5°C / 10°C

[ESTRAZIONE DATI INLINE]

Ora 9:45: Condensatore 67.8°C **SURRISCALDATO** (limite: 45°C) | Termostato: 4.0°C | Evaporatore: -13.5°C | Condensatore: 67.8°C | Aspirazione: -9.6°C | Ambiente: 27.5°C | Porta aperta: no | Sbrinamento: no |

} Intestazione report



2.2 Invio dati all'AI

Ed è a questo punto che può intervenire l'AI per dare un contributo informativo all'utenza spesso disorientata da allarmi e codici di guasto esposti dal controllore e difficilmente interpretabili. L'elenco così consolidato, e formattato come prompt efficace, viene trasmesso all'AI insieme ai valori delle altre variabili, in modo da fornire gli elementi di correlazione necessari.

L'intelligenza artificiale non esegue alcun calcolo tecnico: si limita a produrre una risposta testuale, descrivendo il problema rilevato, le possibili cause e le conclusioni operative. In altre parole, permette di individuare con chiarezza la causa del problema, spesso espressa tramite codici criptici e poco intuitivi utilizzati dai vari controllori presenti sul mercato. La stessa informazione può essere trasmessa al servizio di assistenza tecnica sia tramite Cloud sia dal telefono che l'ha acquisita via NFC, con l'obiettivo di consentire la diagnosi e la risoluzione del guasto da remoto, quando possibile, oppure di attivare un intervento in loco.

In sintesi, il rilevamento delle anomalie è deterministico e basato su regole fisse, mentre l'AI interviene solo per l'interpretazione linguistica e la presentazione del risultato. Questo approccio completa il quadro della sorveglianza tecnica.

2.21 Richiesta API DeepSeek

Le richieste all'API DeepSeek seguono un formato standard che prevede l'utilizzo del metodo HTTP POST verso un endpoint dedicato. Ogni chiamata include l'header Content-Type: application/json e l'autorizzazione tramite Bearer Token. Il corpo della richiesta contiene la selezione del modello (deepseek-chat), i parametri di configurazione (max_tokens e temperature) e la sequenza di messaggi, in cui il contenuto del prompt viene inserito come messaggio utente.

POST <https://api.deepseek.com/v1/chat/completions>

Headers:

Content-Type: application/json

Authorization: Bearer

Body:

```
{
  "model": "deepseek-chat",
  "max_tokens": ,
  "temperature": ,
  "messages": [{ "role": "user", "content": " " }]
}
```

Obiettivo: eseguire la diagnosi di un frigorifero professionale TN ventilato a temperatura positiva, partendo da dati forniti. Parametri API: max_tokens: 400, temperature: 0.

2.22 Formato del Prompt

Il prompt è redatto in modo che l'AI assuma il ruolo di tecnico frigorista. Vengono fornite dettagliate informazioni sulla configurazione dell'impianto (tipologia, setpoint, isteresi, limiti di allarme, range normali, valori fuori limite). Le istruzioni specificano che la risposta deve basarsi esclusivamente su queste informazioni, senza introdurre dati non presenti, e fornire indicazioni sulle possibili cause di anomalie. Il prompt deve essere chirurgico al fine di evitare allucinazioni da parte dell'AI.

La risposta viene articolata in tre capitoli: ANALISI (riassunto dei parametri e anomalie), PROBLEMA RILEVATO (spiegazione tecnica della causa), CONCLUSIONI (consiglio operativo), con un massimo di 6 righe complessive per non appesantire il risultato.

2.23 Snippet di guasto filtrato

Ora 9:45: Condensatore 67.8°C SURRISCALDATO (limite: 45°C) | Termostato: 4.0°C | Evaporatore: -13.5°C | Condensatore: 67.8°C | Aspirazione: -9.6°C | Ambiente: 27.5°C | Porta aperta: no | Sbrinamento: no |

La risposta è un testo libero strutturato in tre sezioni.

2.24 Risposta reale dell'AI

[ANALISI]

Anomalia rilevata: Condensatore a 67.8°C, surriscaldato oltre il limite normale (45°C). Tutti gli altri parametri (evaporatore, aspirazione) sono nei range.

[PROBLEMA RILEVATO]

Alta temperatura di condensazione. Possibili cause: sporcizia su condensatore, ventilatore condensatore guasto o difettoso, carica refrigerante eccessiva.

[CONCLUSIONI]

Intervento urgente: pulizia del condensatore e verifica funzionamento del ventilatore. Controllare un eventuale sovraccarico di gas.

2.25 Costi AI

Costo Token (in+out) DeepSeek: 0.45 euro / [milione token]

Token scambiati: 753.

Costo in euro della singola transazione: 0,00033885 euro.

Come risulta dai calcoli, il costo è assolutamente sostenibile.

3. Sorveglianza complementare: Snapshot System

Lo Snapshot System (SSY) è progettato come componente nativo di diagnostica professionale: raccoglie in un unico report lo stato dell'applicazione, della rete, del dispositivo e del controllore, con l'obiettivo di rendere immediata la comunicazione tra chi utilizza l'applicazione sul campo e chi sviluppa il software in azienda. La figura tipica non è uno sviluppatore: è un professionista di altro settore che utilizza l'unità per conservare, trasformare i prodotti, con la possibilità di segnalare eventuali anomalie in pochi minuti. Lo Snapshot System gli permette di farlo senza interpretare log o messaggi tecnici, generando automaticamente un report strutturato con contesto, modello, installazione, anomalie rilevate, richieste di rete recenti, errori JavaScript, informazioni sul dispositivo e log significativi, pronto per essere copiato e inviato all'assistenza con un solo gesto.

Le informazioni raccolte dal termostato, possono essere lette dall'utente tramite NFC oppure inviate automaticamente al cloud, se connesso, dove diventano disponibili allo sviluppatore del software. L'azienda riceve una fotografia tecnica fedele del comportamento reale dell'unità, senza dipendere da interpretazioni soggettive.

Per lo sviluppatore, questo non rappresenta un "debug in produzione" nel senso tradizionale, perché il codice è già stato testato e validato prima della distribuzione. Tuttavia, il comportamento reale dipende da variabili che in laboratorio non sono completamente riproducibili: modelli di telefono diversi, browser differenti, versioni del sistema operativo, qualità della rete, condizioni ambientali, configurazioni specifiche dell'unità. Lo Snapshot System colma esattamente questo divario fornendo una diagnostica contestuale: una fotografia tecnica precisa del contesto in cui si manifesta un problema. Invece di ricevere segnalazioni generiche come "non funziona" o "si blocca", lo sviluppatore ottiene un report strutturato con errori JavaScript completi di stack trace, richieste HTTP con esito e tempi, stato della connessione e informazioni sul dispositivo.

Questo riduce drasticamente il tempo necessario per capire se l'anomalia è legata alla rete, a un modello specifico di telefono, a una configurazione particolare dell'unità, a un comportamento limite del firmware o a un problema hardware. Non si tratta di rifare il debug, ma di collegare rapidamente il problema al suo contesto reale, con dati oggettivi e immediatamente utilizzabili.



3.1 Funzioni utilizzate

- generateSystemSnapshot: crea il report completo
- copySystemSnapshot: copia il report negli appunti
- switchAITab: attiva la generazione quando si apre il tab System.

La diagnostica si basa su tre meccanismi di intercettazione:

- interceptConsoleLogs per log e warning
- interceptJSErrors per errori JavaScript
- interceptNetworkRequests per le chiamate HTTP. trimInterceptedLogs mantiene i log entro un limite controllato

Le funzioni helper forniscono i dati necessari:

- getNetworkRequests per le richieste recenti
- getDeviceInfo e getDeviceInfoCached per il contesto hardware/software
- getPhoneModel e getDeviceInfoFromUserAgentData per identificare il dispositivo
- getOsDisplayString per la formattazione del sistema operativo

Le funzioni di ottimizzazione garantiscono fluidità e immediatezza:

- enrichDeviceInfoAtStartup prepara i dati del dispositivo
- generateSystemSnapshotInWorker elabora i log in background
- generateSystemSnapshotLazy genera il report solo al primo accesso

Lo Snapshot Functions utilizza due funzioni:

- generateFunctionsSnapshot per il report funzionale del controllore
- copyFunctionsSnapshot per la condivisione

3.11 Esempio di Snapshot System restituito dal controllore:

Acquisizione: 31/01/2026 17:00

Unità in esercizio: Via Severino Grattoni, 12 – Roma – Meteo locale 13°- 75%

Model: VITRA, S/n: A66566TY – Installato: 10/12/2025

[NETWORK]

17:00:22 GET <https://nominatim.openstreetmap.org/reverse> → 200

17:00:24 GET <https://api.open-meteo.com/v1/forecast> → 200

[JAVASCRIPT]

17:00:33 Test errore JavaScript:

at console.error (<https://iceeruda.pages.dev/:3842:14>) at toggleTestErrors
(<https://iceeruda.pages.dev/:4499:15>)

17:00:33 Test console.error:

errore di validazione dati at console.error (<https://iceeruda.pages.dev/:3842:14>) at toggleTestErrors
(<https://iceeruda.pages.dev/:4502:13>)

17:00:33 Unhandled Promise Rejection

[APP FLAGS]

API scaduta: SI', Errore NFC scrittura: SI'

[DEVICE]

OPPO Reno 15F / Android 16 – Viewport: 1280×1024 | Network: 5G online

4. Sorveglianza complementare: Snapshot Service

Lo Snapshot Service (SVC) è il modulo incaricato di monitorare l'usura progressiva dei componenti dell'unità, con l'obiettivo di supportare la manutenzione programmata. A differenza degli score agnostici e dei controlli deterministici, che si concentrano sul comportamento anomalo o sul guasto, il Service valuta la distanza da una soglia di intervento basata su parametri di vita utile.

Nel contesto delle unità di conservazione, lo Snapshot Service si concentra su componenti critici come i relè di potenza, il compressore e la batteria tampone. Il sistema registra il numero di cicli di commutazione dei relè, le ore di funzionamento del compressore, i cicli di sbrinamento

Questi dati vengono confrontati con i valori nominali di progetto per stimare la vita residua dei componenti. Il Service Score viene aggiornato in tempo reale e può essere visualizzato come barra di avanzamento o soglia numerica. Quando il punteggio supera un valore critico, il sistema suggerisce un intervento di manutenzione, evitando guasti improvvisi e ottimizzando la pianificazione degli interventi.

Il Service Score è un indicatore numerico (0–100) che rappresenta la percentuale di usura cumulata dell'unità rispetto alla sua vita utile stimata. A differenza degli score agnostici, non misura scostamenti statistici, ma aggrega metriche lente e progressive, normalizzate rispetto ai valori nominali di progetto.

4.1 Calcolo e significato Service Score

Il Service Score rappresenta un indicatore sintetico e affidabile per la manutenzione predittiva degli impianti. Ogni componente monitorato contribuisce al punteggio finale in base alla propria importanza e alla frequenza con cui necessita di sostituzione. Il calcolo avviene localmente nel controllore e viene aggiornato in modo incrementale, garantendo la memorizzazione persistente dei dati.

4.1.1 Formula Generale

Il punteggio SVC viene calcolato secondo la seguente formula:

$$SVC = \min \{ 100, [(w_1 \cdot R_1 + w_2 \cdot R_2 + w_3 \cdot R_3 + \dots + w_n \cdot R_n) / \sum w_i] \}$$

Dove:

- SVC = Service Score finale (range 0–100)
- w_i = peso assegnato al componente i (relè, compressore, batteria, ecc.)
- R_i = rapporto di usura del componente i , calcolato come:
- $R_i = \min \{ 1, (\text{valore_attuale} / \text{valore_nominale}) \}$

4.1.2 Esempio di Componenti e Pesì

Componente	Parametro monitorato	Valore nominale	Peso (wi)
Relè di potenza	Cicli di commutazione	50.000 cicli	0.45
Compressore	Ore di funzionamento	35.000 ore	0.35
Cicli di sbrinamento	Numero cicli	1.200	0.10
Stress termico	Ore > 45°C PCB	300 ore	0.10

4.1.3 Esempio di Calcolo

Considerando i seguenti valori rilevati per ciascun componente:

- Relè: 50.000 cicli $\rightarrow R_1 = 0.5$
- Compressore: 35.000 ore $\rightarrow R_2 = 0.4$
- Sbrinamenti: 1.200 $\rightarrow R_4 = 0.4$
- Stress termico: 300 ore $\rightarrow R_5 = 0.3$

L'applicazione della formula genera il seguente punteggio:

$$SVC = (0.45 \cdot 0.5 + 0.35 \cdot 0.4 + 0.10 \cdot 0.4 + 0.10 \cdot 0.3) / 1 = 0.225 + 0.14 + 0.04 + 0.03 = \mathbf{0.435}$$

4.1.4 Interpretazione e Utilizzo del Service Score

Il Service Score cresce in modo continuo e viene limitato al valore massimo di 100. Superando determinate soglie operative (ad esempio, 80), il sistema può suggerire interventi programmati. È importante sottolineare che il Service Score non genera allarmi, ma funge da strumento di supporto per la manutenzione predittiva. Il modello di calcolo è completamente agnostico: i valori nominali possono essere configurati in base alle esigenze di ciascun impianto, senza modificare la logica della formula. Il punteggio può essere visualizzato nell'interfaccia utente, trasmesso via NFC o inviato al cloud come metrica per la gestione della flotta.

4.1.5 Esempio di Snapshot Service restituito dal controllore

Acquisizione: 07/02/2026 21:31

Unità in esercizio:

Via Federico Menabrea - Roma

Meteo locale 12° - 99%

Model: VITRA, S/n: A66566TY

Installato: 10/12/2025

[ESTRAZIONE DATI IN LINE]

Relè U1 Cicli 45.233

Relè U2 Cicli 10.009

Relè U3 Cicli 32.262

Compressore ore 35.000

Cicli di sbrinamento n° 1.200

Alta temperatura PCB ore 300

SCORE = 44



5. Sorveglianza complementare: Snapshot Functions

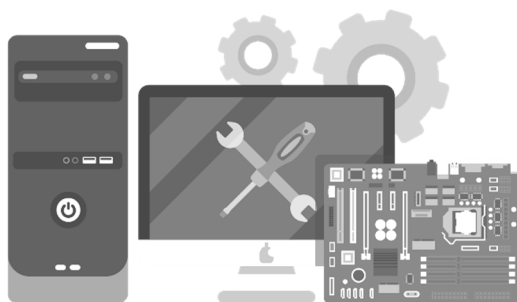
Lo Snapshot Functions (SFN) è il sistema con cui il controllore registra e rende disponibili le informazioni diagnostiche per la parte hardware e firmware; elabora i dati, li memorizza, li trasferisce all'app tramite protocollo NFC, oppure al Cloud se connesso. In questo modo i tecnici hardware/firmware possono analizzare i dati ricevuti.

Il termostato registra: interruzioni e spegnimenti, errori I2C, riavvii MCU, tensioni 3V3/12VDC, temperatura PCB, integrità firmware (checksum, watchdog, heap/stack, allocazioni fallite). Queste informazioni permettono al tecnico di valutare rapidamente l'affidabilità dell'unità, individuare componenti usurati, diagnosticare sensori guasti, riconoscere un'instabilità del firmware, verificare problemi di alimentazione o surriscaldamento e capire come intervenire per la risoluzione del problema. Il report può essere copiato e condiviso, evitando descrizioni verbali e permettendo confronti nel tempo.

Questo assicura un debug remoto, l'identificazione di problemi ricorrenti, l'ottimizzazione del codice firmware e una validazione hardware in condizioni reali.

Requisiti principali: memoria non volatile e aggiornamento continuo delle metriche lato termostato; formato dati standard (JSON/NDEF); lettura NFC e parsing lato app; endpoint di ricezione e archiviazione lato Cloud.

In sintesi, il tecnico ottiene una diagnostica immediata senza strumenti aggiuntivi; In questo modo l'utilizzatore beneficia di interventi più rapidi, maggiore qualità del prodotto e tracciabilità delle anomalie.



5.11 Esempio di Snapshot Functions restituito dal controllore

Acquisizione: 31/01/2026 15:51

Unità in esercizio: Via Severino Grattoni, 12 - Roma

Meteo locale 13° - 75%

Model: VITRA, S/n: A66566TY - Installato: 10/12/2025

[UTILIZZO]

Interruzioni rete: 21

Spegnimenti: 5

Integrità mappe: si

[ERRORI I2C]

Tipo ultimo errore: nack su indirizzo

Dispositivo coinvolto: 0x48

Reset bus I2C: 2

Errori totali sessione: 5

Stato linee in errore: SDA=LOW, SCL=HIGH

[RIAVVIO MCU]

Tipo ultimo reset: watchdog

Registri di fault: 0xDEADBEEF

Stack pointer al crash: 0x3FFB7A10

Reboot totali: 3

[ALIMENTAZIONE 3V3]

Tensione attuale: 3280 mV

Tensione minima: 3190 mV

Tensione massima: 3310 mV

[ALIMENTAZIONE 12VDC]

Tensione attuale: 12040 mV

Tensione minima: 11890 mV

Tensione massima: 12110 mV

[MCU]

Temperatura PCB: 47 °C

[FIRMWARE]

Checksum firmware: OK (0xA3F2)

Watchdog: attivo, refresh OK

Stato task RTOS: 0x07

Heap libero: 18240 bytes

Stack libero: 4096 bytes

Allocazioni fallite: 0

Firmware release: 48901-56744

.

5.2 Score MCU

Lo Score MCU rappresenta un indicatore sintetico, espresso su una scala da 0 a 100, ideato per misurare la stabilità hardware e firmware del controllore. A differenza degli score termodinamici, questo parametro non utilizza una baseline e non è influenzato dalla dinamica fisica dell'impianto. Il valore dello Score MCU viene calcolato localmente dal termostato, sfruttando i dati raccolti tramite Snapshot Functions, che includono reboot, reset da watchdog, errori I2C, stato delle linee SDA/SCL e oscillazioni delle alimentazioni 3V3/12VDC.

Lo Score MCU misura la stabilità hardware e firmware del controllore. Diversamente dagli score termodinamici, non utilizza una baseline e non dipende dalla dinamica fisica dell'impianto: riflette la robustezza del controllore e dei suoi componenti.

Il valore, normalizzato su una scala da 0 a 100, segue gli stessi livelli di attenzione definiti per gli score agnostici. Lo Score MCU sarà sempre visibile nell'interfaccia utente accanto alle cinque barre termodinamiche e allo Score Batteria e permetterà di identificare in modo rapido condizioni di instabilità hardware o firmware che potrebbero influenzare il funzionamento dell'unità.

5.2.1 Formula di Calcolo dello Score MCU

Il calcolo dello Score MCU si basa su quattro contributi principali, ciascuno relativo a una specifica area di funzionamento del controllore:

1. Reboot e Watchdog:
 - 0 reboot: 0 punti
 - 1 reboot: +10 punti
 - 2 reboot: +20 punti
 - 3 o più reboot: +35 punti
2. Watchdog reset: +40 punti (sovrascrive il punteggio precedente)
3. Errori I2C: 1–3 errori: +10 punti, 4–10 errori: +20 punti, >10 errori: +30 punti
4. SDA/SCL bloccate: +40 punti
5. Alimentazioni (3V3 / 12VDC):
 - Oscillazione inferiore al 3%: 0 punti
 - Oscillazione tra 3% e 7%: +10 punti
 - Oscillazione superiore al 7%: +20 punti
 - Undervoltage: +30 punti
6. Vita relè:
 - Tutti con vita residua superiore al 60%: 0 punti
 - Almeno uno con vita residua inferiore al 40%: +10 punti
 - Almeno uno con vita residua inferiore al 20%: +20 punti

Il punteggio grezzo è la somma di tutti i contributi, fino a un massimo di 120. La normalizzazione avviene tramite la seguente formula: $MCU_SCORE = \min(100, (score_grezzo / 120) \times 100)$.

6. Score Batteria (per unità con backup di energia)

Lo Score Batteria è un indicatore composito progettato per monitorare in modo affidabile lo stato della batteria di backup, integrando due dimensioni complementari: l'invecchiamento certo (AGING) e lo scostamento elettrico (DRIFT). Questa architettura consente di distinguere tra il naturale decadimento prestazionale dovuto al tempo e l'emergere di anomalie elettriche, mantenendo al contempo un criterio oggettivo per la sostituzione programmata.

La componente AGING rappresenta l'usura temporale della batteria. Cresce in modo lineare dalla data di installazione fino a raggiungere il valore 100 al ventiquattresimo mese, indipendentemente dal comportamento elettrico. Questo andamento riflette la politica di sostituzione programmata: dopo due anni la batteria viene considerata a fine vita utile, anche in assenza di segnali di degrado elettrico. È quindi possibile che, pur in presenza di un DRIFT minimo, lo Score complessivo raggiunga comunque il valore massimo per effetto della sola componente AGING.

La componente DRIFT misura la deviazione del comportamento elettrico rispetto alla baseline storica della batteria. Utilizza lo stesso modello agnostico impiegato per le variabili termodinamiche, analizzando media, trend, varianza e forma della curva di tensione. In questo modo è possibile rilevare instabilità o anomalie che non dipendono dall'età della batteria ma dal suo comportamento operativo.

In analogia con le variabili termodinamiche, allo Score viene aggiunta anche una componente deterministica, che intercetta condizioni non riconducibili a un'analisi statistica:

- Batteria sconnessa,
- Mancato raggiungimento della soglia di ricarica dopo il tempo previsto dalla funzione di mantenimento,
- Tensione insufficiente al momento dell'inserimento della batteria.

Questi eventi generano un incremento diretto dello Score, poiché rappresentano condizioni di malfunzionamento certo.

6.1 Score finale e visualizzazione

Lo Score Batteria è ottenuto tramite una combinazione pesata di AGING e DRIFT ed è rappresentato nell'interfaccia come una barra dedicata. Il risultato è un criterio di valutazione trasparente, riproducibile e coerente con le esigenze di affidabilità del sistema.



6.2 Composizione dello Score Batteria

AGING: misura l'invecchiamento certo della batteria, che raggiunge il valore massimo di 100 al termine della vita utile prevista (24 mesi).

DRIFT: valuta lo scostamento elettrico rispetto alla baseline storica, analizzando la tensione tramite modelli statistici agnostici.

Deterministico: segnala condizioni certe di malfunzionamento (sconnessione, mancata ricarica, tensione insufficiente all'inserimento).

La batteria ha una durata massima di 24 mesi. Anche in assenza di anomalie elettriche, al compimento di questo periodo lo Score Batteria raggiunge 100, segnalando la necessità di una sostituzione programmata.

6.3 Formule di Calcolo

6.3.1 Componente AGING

- t = mesi trascorsi dall'installazione (0-24)
- $AGING = \min(100, (t / 24) \times 100)$

Esempi: 0 mesi = 0; 12 mesi = 50; 24 mesi = 100 (sostituzione raccomandata)

6.3.2 Componente DRIFT

Controlli standard applicati:

- Differenza di media (Z-score)
- Trend (pendenza)
- Varianza ($\sigma_{current} / \sigma_{baseline}$)
- Forma della curva (RMSE_norm)

Formula di normalizzazione:

- $DRIFT = (score_grezzo / 100)^{0.55} \times 100$

6.3.3 Componente deterministica

Valori incrementali applicati in caso di:

- Batteria sconnessa
- Mancata ricarica entro il tempo previsto
- Tensione insufficiente all'inserimento

Questi eventi portano lo Score verso 100 in modo immediato e non statistico.

6.3.4 Combinazione finale

Peso consigliato:

- AGING = 70%
- DRIFT = 30%

Formula: $BATTERY_SCORE = \min(100, 0.7 \times AGING + 0.3 \times DRIFT + \text{deterministico})$

7. Applicazione del Modello di Sorveglianza alle unità di processo

L'applicazione del modello di sorveglianza agnostica ad un'unità di processo, quale un forno professionale, richiede di spostare il concetto di "normalità" dal funzionamento continuo dell'impianto alla ripetibilità delle fasi che compongono una ricetta. Ogni ciclo di cottura, pur variando nei valori assoluti, segue una struttura riconoscibile: preriscaldamento, carico, cottura principale e fase finale. Il firmware deve quindi osservare queste fasi, non l'intero ciclo come un blocco unico, e costruire per ciascuna una baseline basata su cicli riusciti.

Durante l'uso reale, il forno registra alcuni cicli completati senza allarmi critici e li normalizza sul tempo della fase, mappandoli su un asse 0–100%. Da questi dati vengono ricavati media, deviazione standard e forma della curva. La baseline non è una curva rigida ma un corridoio statistico ($\text{media} \pm k \cdot \sigma$)* che rappresenta la variabilità normale della ricetta, inclusa la pezzatura del prodotto. Questo approccio evita di confondere un carico pesante con un forno degradato: finché la dinamica reale rimane nel corridoio, il comportamento è considerato coerente.

Durante l'esecuzione di un ciclo, il firmware campiona temperatura camera, eventuale temperatura al cuore, duty delle resistenze, ventilazione, umidità iniettata. Ogni campione viene associato al punto corrispondente della fase normalizzata e confrontato con la baseline. Lo scostamento viene misurato attraverso i quattro indicatori canonici precedentemente descritti: **differenza di media (Z-score), pendenza (trend), varianza e forma della curva (RMSE)**. Questi contributi generano uno score per fase, che viene poi combinato in uno score complessivo del ciclo. Un ammorbidimento finale stabilizza la dinamica ed evita reazioni impulsive.

In parallelo al modello statistico, il firmware applica controlli deterministici per gli eventi di guasto certo: sovratemperatura camera, sonde fuori range, errori di sicurezza ed altro. Questi eventi incrementeranno direttamente lo score e verranno registrati senza interpretazione statistica.

Al termine del ciclo, se non sono presenti anomalie critiche e lo score è sotto una soglia di degrado, i dati normalizzati vengono integrati nella baseline tramite un aggiornamento lento. In caso contrario, il ciclo viene scartato per evitare che il sistema apprenda comportamenti anomali. Questo garantisce che la baseline rimanga un riferimento affidabile nel tempo.

Il firmware genera gli snapshot tecnici (System, Functions e Service) per la diagnostica hardware/firmware e la manutenzione programmata. L'analisi operativa del ciclo (score, eventi, performance termiche) è gestita dai sistemi di sorveglianza agnostica e deterministica. Lo snapshot è leggibile dall'app, trasferibile via NFC e archiviabile nel cloud se l'unità è connessa, fornendo al service e all'R&D una fotografia tecnica completa del ciclo.

Il risultato è un sistema che distingue in modo affidabile la normale variabilità del prodotto da un reale degrado del forno, non richiede configurazioni manuali, non utilizza soglie fisse e rimane completamente agnostico rispetto al tipo di alimento. La logica è industrialmente realizzabile: segmentazione del ciclo, normalizzazione temporale, corridoio statistico, calcolo locale dello score e snapshot tecnici integrati.

Significato di $\text{media} \pm k \cdot \sigma$

Quando diciamo $\text{media} \pm k \cdot \sigma$, stiamo definendo un **corridoio statistico** entro cui un comportamento è considerato normale.

- **media** = il valore medio osservato in una fase della ricetta
- **σ (sigma)** = la deviazione standard
- **k** = un fattore che decide quanto largo deve essere il corridoio (tipico tra 1 e 3).

In pratica:

- **$\text{media} + k \cdot \sigma$** = limite superiore del comportamento normale
- **$\text{media} - k \cdot \sigma$** = limite inferiore del comportamento normale

Tutto ciò che sta **dentro** questo intervallo è considerato **coerente** con la ricetta. Tutto ciò che sta **fuori** indica **scostamento**.

7.1 Definizioni formali degli score specifici

SC_RICETTA – Score di Cottura

Indice numerico (0–100) che misura lo scostamento della cottura reale rispetto alla baseline statistica della ricetta. È calcolato confrontando, per ciascuna fase del ciclo, la dinamica termica misurata (temperatura camera, eventuale temperatura al cuore, duty resistenze e ventilazione) con la baseline normalizzata ($\text{media} \pm k \cdot \sigma$). Lo score di fase è ottenuto combinando Z-score, trend, varianza e RMSE; lo score di ricetta è la somma pesata degli score di fase, con ammorbidimento finale. Valori elevati indicano comportamento anomalo rispetto alla firma termica attesa della ricetta.

SC_FORNO – Score di Prestazione Termica

Indice numerico (0–100) che rappresenta lo stato termico del forno nel tempo. È calcolato come aggregazione (media o mediana pesata) degli SC_RICETTA degli ultimi N cicli significativi. Riflette la capacità del forno di raggiungere e mantenere i setpoint, recuperare dopo il carico e garantire stabilità termica. Valori crescenti indicano degrado progressivo di resistenze, ventilazione, sensori o dispersioni termiche.

SC_ALLARMI – Score Deterministico di Guasto

Indice numerico (0–100) generato esclusivamente da eventi deterministici rilevati dal firmware. Ogni evento certo (sovratemperatura camera oltre soglia e ritardo, sensori fuori range, resistenze non funzionanti, errori porta/protezioni, errori ventilazione) produce un incremento definito dello score. Non utilizza analisi statistica. Valori elevati indicano la presenza di guasti reali che richiedono intervento.

SC_MCU – Score di Affidabilità Hardware/Firmware

Indice numerico (0–100) che misura la qualità operativa del forno dal punto di vista elettronico e firmware. È calcolato aggregando: riavvii MCU, watchdog, errori I2C/bus, tensioni anomale, temperatura PCB, errori di memoria, integrità firmware.

8. Specifica tecnica per implementazione firmware

Sorveglianza Agnostica per Ricette Termiche Discontinue

8.1. Obiettivo

Implementare nel firmware un sistema di sorveglianza basato su analisi statistica delle fasi di cottura, capace di distinguere la normale variabilità del carico (pezzatura) da un degrado reale del forno. Il sistema deve operare localmente, senza dipendenze dal cloud, e produrre uno score di ciclo e uno snapshot tecnico.

8.2. Struttura del ciclo

Il firmware deve riconoscere automaticamente le seguenti fasi:

- Preriscaldamento (fino al raggiungimento del setpoint)
- Carico (porta aperta → chiusura → assestamento)
- Cottura principale (mantenimento setpoint e vapore)
- Fase finale (asciugatura/mantenimento/raffreddamento)

La transizione tra fasi è determinata da eventi misurabili.

8.3. Acquisizione dati

Il firmware deve campionare con frequenza costante (es. 10 s):

- Temperatura camera
- Eventuale temperatura sonda al cuore
- Duty resistenze (0–100%)
- Duty ventilazione
- Stato vapore (on/off o n%)
- Eventi: porta, pause, errori

I dati vanno memorizzati in RAM per il ciclo corrente e in FLASH per la baseline.

8.4. Normalizzazione temporale

Ogni fase ha durata variabile. Il firmware deve mappare i campioni su un asse normalizzato 0–100% della fase. Es.: 50 punti per fase, indipendenti dalla durata reale.

8.5. Costruzione baseline

Per ogni ricetta, il firmware deve mantenere una baseline composta da:

media_i per ogni punto normalizzato

deviazione standard σ_i

curva media per RMSE

La baseline si costruisce da cicli "buoni", definiti come:

- nessun allarme critico
- nessuna interruzione anomala
- score ciclo < soglia_degrado (es. 60)

I dati dei cicli buoni vengono integrati tramite media mobile lenta (EMA o media pesata).

8.5.1. Corridoio statistico

Per ogni punto normalizzato:

- $\text{limite_sup} = \text{media_i} + k \cdot \sigma_i$
- $\text{limite_inf} = \text{media_i} - k \cdot \sigma_i$
(k tipico: 1.5–2.5)

Il corridoio rappresenta la variabilità normale della ricetta, inclusa la pezzatura.

8.5.2. Calcolo score di fase

Per ogni fase, il firmware deve calcolare quattro contributi:

a) **Differenza di media (Z-score)**

$$Z = (\text{media_attuale} - \text{media_baseline}) / \sigma_baseline$$

b) **Trend (pendenza)**

Calcolo della pendenza su finestra scorrevole; penalizzare salite lente o instabilità.

c) **Varianza**

Confronto $\text{varianza_attuale} / \text{varianza_baseline}$; penalizzare oscillazioni anomale.

d) **RMSE** Confronto forma curva attuale vs curva media baseline.

Ogni contributo produce un valore 0–100, pesato e limitato.

8.5.3. Score di ciclo

Lo score di ciclo è la somma pesata degli score di fase.

Applicare ammorbidimento finale:

$$\text{SCO} = (\text{score_grezzo} / 100)^{0.55} \times 100$$

8.5.4. Componente deterministica

Eventi deterministici possono incrementare direttamente lo score o esporre il segnale di allarme e il relativo codice di guasto.

- sovratemperatura camera oltre soglia e ritardo
- sensori fuori range
- resistenze non funzionanti (tempo salita > limite assoluto)
- errori porta/protezioni
- errori ventilazione, ecc.

Ogni evento genera:

- codice evento
- incremento score (**non necessariamente**)
- registrazione nello snapshot

8.55 Compatibilità con attuatori a modulazione continua

Il modello di sorveglianza rimane valido anche per unità dotate di ventilazione a inverter e resistenze modulate tramite SSR. La natura agnostica dello score non dipende dal tipo di attuatore, ma esclusivamente dalla dinamica statistica delle variabili monitorate.

I valori reali di comando (RPM effettivi per la ventilazione, duty effettivo per le resistenze) vengono integrati nella baseline con la stessa logica utilizzata per i sistemi a step discreti. La maggiore granularità del segnale è assorbita automaticamente dalla finestra mobile e dal tempo di refresh dello score.

Non sono richiesti adattamenti strutturali: la baseline continua a rappresentare il comportamento normale della ricetta, mentre eventuali scostamenti persistenti o variazioni di forma vengono rilevati con la metodologia statistica già definita.

8.6. Aggiornamento della baseline

A fine ciclo:

- se ciclo buono → aggiornare baseline
- se score alto o allarmi → NON aggiornare
- mantenere massimo N cicli baseline per ricetta (es. 20–50)

8.7. Snapshot tecnici

Il firmware deve generare tre tipologie di snapshot, coerenti con il modello generale del sistema di sorveglianza:

SNAPSHOT SYSTEM (SSY)

- Stato connessione NFC/Cloud
- Richieste di rete recenti (se connesso)
- Errori applicazione (se presente interfaccia utente)

SNAPSHOT FUNCTIONS (SFN)

- Riavvii MCU, watchdog, errori I2C
- Temperatura PCB, tensioni alimentazione
- Checksum firmware, heap/stack libero
- Allocazioni fallite, stato task RTOS

SNAPSHOT SERVICE (SVC)

- Cicli commutazione relè (attuale/nominale)
- Ore funzionamento resistenze (attuale/nominale)
- Ore funzionamento motore ventilazione (attuale/nominale)
- Numero cicli cottura completati (attuale/nominale)
- Service Score calcolato (0–100)

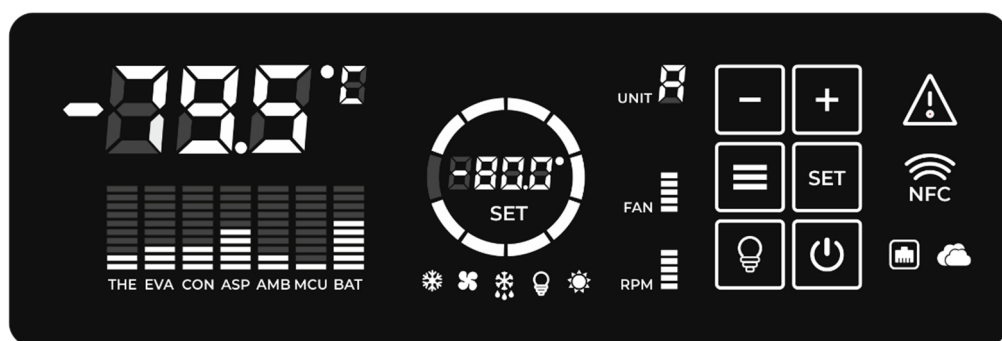
Tutti gli snapshot devono essere: – Leggibili dall'app – Trasferibili via NFC – Inviabili al cloud (se connesso) – Memorizzati in formato standard (JSON/NDEF). I dati operativi del ciclo (score, tempi, eventi) sono già disponibili attraverso il sistema di analisi agnostica e deterministica e non richiedono uno snapshot dedicato.

8.8. Risultato atteso

Il sistema deve: distinguere carico pesante da forno degradato, non richiedere configurazioni manuali, non usare soglie fisse, adattarsi all'uso reale, fornire score affidabile e riproducibile, produrre snapshot tecnici utili a service, R&D e Chef.

Concludendo: il risultato è un sistema di sorveglianza che non si limita a rilevare anomalie, ma le rende comprensibili, contestualizzate e utili, migliorando la qualità dell'intervento e l'affidabilità dell'intera flotta.

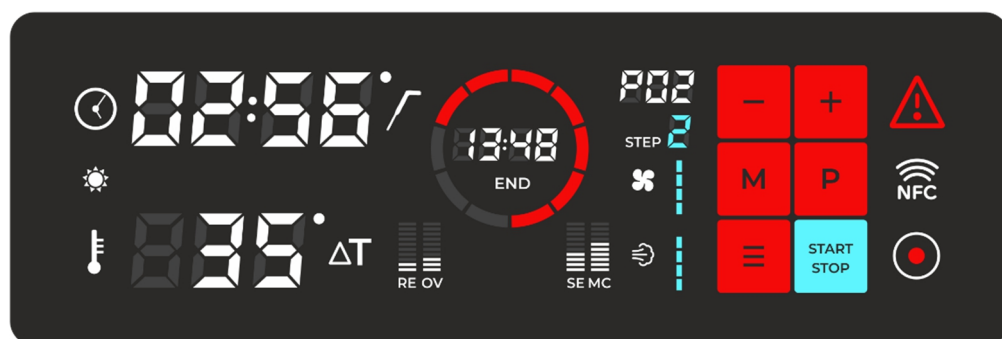
9. Esempi di Interfaccia con display EBTN



Pannello di controllo per unità TN, BT, FREEZER, BANCHE BIO

Formato 180x60mm

Area visibile Display 180x35mm



Pannello di controllo per Forni Professionali

Formato 180x60mm

Area visibile Display 180x35mm

Note

