

MANUALE UTENTE

MULTIPROTOCOL “KEY-C” GATEWAYS SERIES

MODBUS TO CLOUD (MQTT/HTTP) GATEWAYS



SENECA S.r.l.

Via Austria 26 – 35127 – Z.I. - PADOVA (PD) - ITALY

Tel. +39.049.8705355 – 8705355 Fax +39 049.8706287

+

www.seneca.it

ORIGINAL INSTRUCTIONS

ATTENZIONE

SENECA non garantisce che tutte le specifiche e/o gli aspetti del prodotto e del firmware, ivi incluso, risponderanno alle esigenze dell'effettiva applicazione finale pur essendo, il prodotto di cui alla presente documentazione, rispondente a criteri costruttivi secondo le tecniche dello stato dell'arte.

L'utilizzatore si assume ogni responsabilità e/o rischio segnatamente alla configurazione del prodotto per il raggiungimento dei risultati previsti in relazione all'installazione e/o applicazione finale specifica.

SENECA, previ accordi al caso di specie, può fornire attività di consulenza per la buona riuscita dell'applicazione finale, ma in nessun caso può essere ritenuta responsabile per il buon funzionamento della stessa.

Il prodotto SENECA è un prodotto avanzato, il cui funzionamento è specificato nella documentazione tecnica fornita con il prodotto stesso e/o scaricabile, anche in un momento antecedente all'acquisto, dal sito internet www.seneca.it.

SENECA adotta una politica di continuo sviluppo riservandosi, pertanto, il diritto di effettuare e/o introdurre - senza necessità di preavviso alcuno - modifiche e/o miglioramenti su qualsiasi prodotto descritto nella presente documentazione.

Il prodotto quivi descritto può essere utilizzato solo ed esclusivamente da personale qualificato per la specifica attività ed in conformità con la relativa documentazione tecnica avendo riguardo, in particolare modo, alle avvertenze di sicurezza.

Il personale qualificato è colui che, sulla base della propria formazione, competenza ed esperienza, è in grado di identificare i rischi ed evitare potenziali pericoli che potrebbero verificarsi nell'utilizzo di questo prodotto.

I prodotti SENECA possono essere utilizzati esclusivamente per le applicazioni e nelle modalità descritte nella documentazione tecnica relativa ai prodotti stessi.

Al fine di garantire il buon funzionamento e prevenire l'insorgere di malfunzionamenti, il trasporto, lo stoccaggio, l'installazione, l'assemblaggio, la manutenzione dei prodotti SENECA devono essere eseguiti nel rispetto delle avvertenze di sicurezza e delle condizioni ambientali specificate nella presente documentazione.

La responsabilità di SENECA in relazione ai propri prodotti è regolata dalle condizioni generali di vendita scaricabili dal sito www.seneca.it.

SENECA e/o i suoi dipendenti, nei limiti della normativa applicabile, non saranno in ogni caso ritenuti responsabili di eventuali mancati guadagni e/o vendite, perdite di dati e/o informazioni, maggiori costi sostenuti per merci e/o servizi sostitutivi, danni a cose e/o persone, interruzioni di attività e/o erogazione di servizi, di eventuali danni diretti, indiretti, incidentali, patrimoniali e non patrimoniali, consequenziali in qualsiasi modalità causati e/o cagionati, dovuti a negligenza, imprudenza, imperizia e/o altre responsabilità derivanti dall'installazione, utilizzo e/o impossibilità di utilizzo del prodotto.

CONTACT US

Technical support	supporto@seneca.it
Product information	commerciale@seneca.it

Document revisions

DATE	REVISION	NOTES	AUTHOR
14/02/2025	0	Prima revisione	MM
24/02/2025	1	Aggiunto capitolo sul significato dei led	MM
01/07/2025	2	Aggiunte regole logiche Aggiunto datalogger Aggiunto Cloud Seneca Cloudbox 2 Riscrittura del documento	MM
06/10/2025	3	Modificato calcolo per numero di log in flash. Aggiunta la scalatura dei TAG Aggiunta la gestione dei gruppi di campionamento Aggiunta la modifica da Cloud del tempo di campionamento dei gruppi Modificato Schema a blocchi di funzionamento	MM
17/03/2026	4	Fix funzionamento delle azioni delle regole logiche con period = 0 Allineato al firmware rev 119	MM

Questo documento è di proprietà di SENECA srl.
La duplicazione e la riproduzione sono vietate, se non autorizzate.

INDICE

1.	DESCRIZIONE	6
1.1.	Protocolli MODBUS, MQTT, HTTP	6
1.2.	CARATTERISTICHE DELLE PORTE DI COMUNICAZIONE DELLA SERIE "KEY"	6
2.	REVISIONE HARDWARE DEL DISPOSITIVO	7
3.	TECNOLOGIA FLEX PER IL CAMBIO DI PROTOCOLLO	8
3.1.	CAMBIO DEI PROTOCOLLI CON IL SOFTWARE SENECA DISCOVERY DEVICE	9
4.	SIGNIFICATO DEI LED	10
4.1.	LED MODELLO Z-KEY-C	10
4.2.	LED MODELLO R-KEY-LT-C	11
4.3.	LED MODELLO Z-KEY-2ETH-C	12
5.	PORTA ETHERNET	13
6.	AGGIORNAMENTO FIRMWARE	13
7.	ACQUISIZIONE ED ELABORAZIONE DEI DATI, INVIO DI DATI E ALLARMI	14
7.1.	IL DATA BUS E I PROTOCOLLI INDUSTRIALI MODBUS	15
7.1.1.	PROTOCOLLI MODBUS	15
7.2.	TAGS DATA SHARED MEMORY (MEMORIA CONDIVISA)	16
7.3.	IL DATALOGGER E LA CACHE	16
7.4.	ELABORAZIONE DEI TAG: LE REGOLE LOGICHE	17
7.5.	CONNESSIONE AI CLOUD TRAMITE TECNOLOGIA "EASY CLOUD"	17
7.6.	INVIO DI ALLARMI AI CLOUD	17
8.	MODALITA' DI FUNZIONAMENTO	18
8.1.	MODBUS MASTER / CLIENT TO CLOUD	18
8.2.	DIAGNOSTICA SEMPLIFICATA DEI TAG	19
8.3.	DIAGNOSTICA ESTESA DEI TAG	19
9.	WEBSERVER DEI GATEWAY "-C"	21
9.1.	GUIDA PASSO PASSO PER IL PRIMO ACCESSO AL WEBSERVER	21
10.	CONFIGURAZIONE DEL DISPOSITIVO DA WEBSERVER	22
10.1.	PAGINA "SETUP"	22
10.1.1.	PARAMETRI DI CONFIGURAZIONE GENERALI	23
10.1.	PAGINA "DATALOGGER"	26
10.2.	PAGINA "SETUP TAG"	26
10.2.1.	Vista in tempo reale del Modbus Gateway: LA PAGINA "STATUS"	32
10.3.	PAGINA "SETUP TEXT MESSAGE" (ALLARMI)	33

10.4.	PAGINA "SETUP TIMER"	33
10.5.	PAGINA "SETUP RULES"	35
10.5.1.	RULE CONFIGURATION.....	35
10.5.2.	IF CONDITION: TYPE.....	36
10.5.3.	IF CONDITION OPERATOR.....	39
10.5.4.	THEN/ELSE ACTION.....	41
10.6.	PAGINA "CLOUD SETUP"	46
10.6.1.1.	ESEMPIO	50
10.6.2.	DIREL ADM4.0.....	51
10.6.1.	CLOUDBOX 2 SENECA.....	52
10.7.	PAGINA "FIRMWARE UPDATE"	52
10.8.	PAGINA "UTC TIME SETUP".....	53
10.9.	PAGINA "CERTIFICATE/DATABASE UPDATE"	53
10.10.	PAGINA "SERIAL TRAFFIC MONITOR"	53
11.	SCRITTURE DA CLOUD VERSO IL DISPOSITIVO	55
11.1.	SCRIVERE TAG DAL CLOUD AL DISPOSITIVO VIA MQTT (CLOUD GENERIC).....	55
12.	MODIFICA DEL TEMPO DI CAMPIONAMENTO DA MQTT	57
13.	RIPRISTINO DEL DISPOSITIVO ALLA CONFIGURAZIONE DI FABBRICA	58
14.	TEMPLATE EXCEL.....	59
15.	INSTALLAZIONE DI PIÙ DISPOSITIVI IN UNA RETE UTILIZZANDO IL "DHCP FAIL ADDRESS".....	59
16.	IL CAVO RS232 DB9	59
17.	PROTOCOLLI MODBUS DI COMUNICAZIONE SUPPORTATI	60
17.1.	Codici funzione Modbus supportati	60
18.	INFORMAZIONI SUI REGISTRI MODBUS	61
18.1.	NUMERAZIONE DEGLI INDIRIZZI MODBUS "0 BASED" O "1 BASED"	61
18.2.	NUMERAZIONE DEGLI INDIRIZZI MODBUS CON CONVENZIONE "0 BASED"	62
18.3.	NUMERAZIONE DEGLI INDIRIZZI MODBUS CON CONVENZIONE "1 BASED" (STANDARD)	62
18.4.	CONVENZIONE DEI BIT ALL'INTERNO DI UN REGISTRO MODBUS HOLDING REGISTER	63
18.5.	CONVENZIONE DEI BYTE MSB e LSB ALL'INTERNO DI UN REGISTRO MODBUS HOLDING REGISTER	63
18.6.	RAPPRESENTAZIONE DI UN VALORE A 32 BIT IN DUE REGISTRI MODBUS HOLDING REGISTER CONSECUTIVI	64
18.7.	TIPI DI DATO FLOATING POINT A 32 BIT (IEEE 754)	65

1. DESCRIZIONE

I prodotti Z-KEY-C, R-KEY-LT-C, Z-KEY-2ETH-C permettono di acquisire i dati da bus seriali o ethernet basati su protocolli Modbus e di inviarli ai cloud con il protocollo MQTT(s) o http(s).

È anche supportata la scrittura da cloud verso Modbus.

1.1. PROTOCOLLI MODBUS, MQTT, HTTP



I protocolli Modbus supportati sono:

Modbus RTU Master

Modbus RTU Slave

Modbus ASCII Master

Modbus ASCII Slave

Modbus TCP-IP Server

Modbus TCP-IP Client

Per ulteriori informazioni su questi protocolli, consultare il sito web delle specifiche Modbus:

<http://www.modbus.org/specs.php>



Il protocollo MQTT supportato è la versione 3.1.1



Il protocollo HTTP per la pubblicazione dei tag su cloud si basa su API Rest



Il protocollo TLS supportato è la versione 1.2



Certificati delle chiavi secondo standard X.509

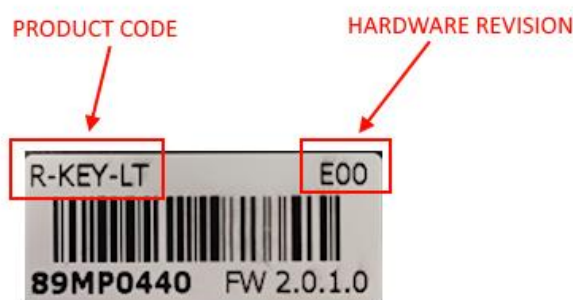
1.2. CARATTERISTICHE DELLE PORTE DI COMUNICAZIONE DELLA SERIE "KEY"

PRODOTTO	NR PORTE ETHERNET	NR PORTE SERIALI	PORTE SERIALI ISOLATE
Z-KEY-C	1	2	Sì, entrambe le porte
R-KEY-LT-C	1	1	NO
Z-KEY-2ETH-C	2	2	Sì, entrambe le porte

2. REVISIONE HARDWARE DEL DISPOSITIVO

In un'ottica di miglioramento continuo Seneca aggiorna e rende sempre più sofisticato l'hardware dei suoi dispositivi. È possibile conoscere la revisione hardware di un prodotto tramite l'etichetta posta nel fianco del dispositivo.

Un esempio di etichetta del prodotto R-KEY-LT è il seguente:



Nell'etichetta è anche riportata la revisione di firmware presente nel dispositivo (in questo caso 2.0.1.0) al momento della vendita, la revisione hardware (in questo caso) è la E00.

Per migliorare le prestazioni o per estendere le funzionalità Seneca consiglia di aggiornare il firmware all'ultima versione disponibile (si veda nel sito www.seneca.it la sezione dedicata al prodotto).

3. TECNOLOGIA FLEX PER IL CAMBIO DI PROTOCOLLO



I dispositivi della serie KEY, a partire dalla revisione hardware indicata nella tabella seguente, includono la tecnologia Flex.

GATEWAY	TECNOLOGIA FLEX SUPPORTATA DALLA REVISIONE HARDWARE
Z-KEY	"G00"
R-KEY-LT	"E00"
Z-KEY-2ETH	"C00"

Flex permette di cambiare a piacimento la combinazione dei protocolli di comunicazione industriale supportati dai gateway tra un elenco di quelli disponibili, lo sviluppo è in continuo aggiornamento, per una lista esaustiva fare riferimento alla pagina:

<https://www.seneca.it/flex/>

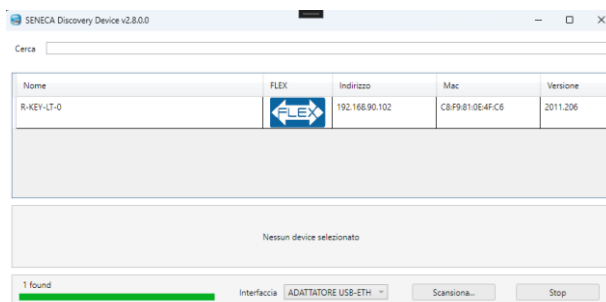
Alcuni esempi di protocolli supportati sono:



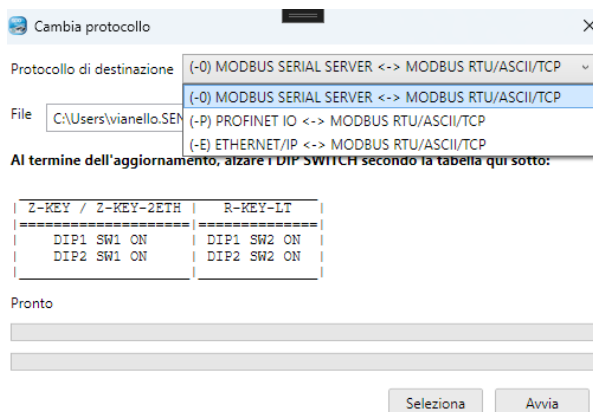
Il gateway diventa quindi "universale" e compatibile con i sistemi Siemens oppure Rockwell oppure Schneider etc...senza la necessità di acquistare hardware differenti.

3.1. CAMBIO DEI PROTOCOLLI CON IL SOFTWARE SENECA DISCOVERY DEVICE

Dalla revisione 2.8 il software Seneca Discovery Device individua i dispositivi che supportano la tecnologia "Flex":



Ad esempio nel caso in figura è possibile premere il pulsante "Cambio Protocollo" e selezionare il protocollo di destinazione tra quelli in elenco:



Alla fine dell'operazione portare (solo alla prima accensione) i dip 1 e 2 a "ON" per forzare il dispositivo a default (vedi anche il capitolo "RIPRISTINO DEL DISPOSITIVO ALLA CONFIGURAZIONE DI FABBRICA").

Fare sempre riferimento al manuale user del protocollo di comunicazione installato nel dispositivo scaricandolo dal sito Seneca.

4. SIGNIFICATO DEI LED

I dispositivi sono dotati di led il cui significato è il seguente:

4.1. LED MODELLO Z-KEY-C

LED	STATO
PWR	Acceso fisso: dispositivo alimentato e indirizzo IP impostato
	Lampeggiante: indirizzo IP non ancora impostato
	Spento: dispositivo non alimentato
COM	Acceso fisso: nessun errore di connessione al cloud
	Lampeggiante: errore di connessione al cloud (per maggiori dettagli sull'errore fare riferimento alla pagina status del webserver)
	Spento: dispositivo non alimentato
TX1	Lampeggiante: trasmissione dati su porta seriale #1
	Spento: nessuna trasmissione su porta seriale #1
RX1	Lampeggiante: ricezione dati su porta seriale #1
	Acceso fisso: verificare il cablaggio della porta seriale #1
	Spento: nessuna ricezione su porta seriale #1
TX2	Lampeggiante: trasmissione dati su porta seriale #2
	Spento: nessuna trasmissione su porta seriale #2
RX2	Lampeggiante: ricezione dati su porta seriale #2
	Acceso fisso: verificare il cablaggio della porta seriale #2
	Spento: nessuna ricezione su porta seriale #2
ETH ACT (VERDE)	Lampeggiante: presenza di dati sulla porta ethernet
	Acceso fisso: porta ethernet connessa ma nessuna presenza di dati
	Spento: verificare il cablaggio della porta ethernet

ETH LNK (GIALLO)	Acceso fisso: cavo ethernet connesso Spento: verificare il cablaggio della porta ethernet
---------------------	--

4.2. LED MODELLO R-KEY-LT-C

LED	STATO
PWR	Acceso fisso: dispositivo alimentato e indirizzo IP impostato Lampeggiante: indirizzo IP non ancora impostato Spento: dispositivo non alimentato
COM	Acceso fisso: nessun errore di connessione al cloud Lampeggiante: errore di connessione al cloud (per maggiori dettagli sull'errore fare riferimento alla pagina status del webserver) Spento: dispositivo non alimentato
TX	Lampeggiante: trasmissione dati su porta seriale Spento: nessuna trasmissione su porta seriale
RX	Lampeggiante: ricezione dati su porta seriale Acceso fisso: verificare il cablaggio della porta seriale Spento: nessuna ricezione su porta seriale
ETH ACT (VERDE)	Lampeggiante: presenza di dati sulla porta ethernet Acceso fisso: porta ethernet connessa ma nessuna presenza di dati Spento: verificare il cablaggio della porta ethernet
ETH LNK (GIALLO)	Acceso fisso: cavo ethernet connesso Spento: verificare il cablaggio della porta ethernet

4.3. LED MODELLO Z-KEY-2ETH-C

LED	STATO
PWR	Acceso fisso: dispositivo alimentato e indirizzo IP impostato Lampeggiante: indirizzo IP non ancora impostato Spento: dispositivo non alimentato
COM	Acceso fisso: nessun errore di connessione al cloud Lampeggiante: errore di connessione al cloud (per maggiori dettagli sull'errore fare riferimento alla pagina status del webserver) Spento: dispositivo non alimentato
TX1	Lampeggiante: trasmissione dati su porta seriale #1 Spento: nessuna trasmissione su porta seriale #1
RX1	Lampeggiante: ricezione dati su porta seriale #1 Acceso fisso: verificare il cablaggio della porta seriale #1 Spento: nessuna ricezione su porta seriale #1
TX2	Lampeggiante: trasmissione dati su porta seriale #2 Spento: nessuna trasmissione su porta seriale #2
RX2	Lampeggiante: ricezione dati su porta seriale #2 Acceso fisso: verificare il cablaggio della porta seriale #2 Spento: nessuna ricezione su porta seriale #2
ET1	Lampeggiante: presenza di dati sulla porta ethernet #1 Acceso fisso: porta ethernet #1 connessa ma nessuna presenza di dati Spento: verificare il cablaggio della porta ethernet #1
ET2	Lampeggiante: presenza di dati sulla porta ethernet #2 Acceso fisso: porta ethernet #2 connessa ma nessuna presenza di dati Spento: verificare il cablaggio della porta ethernet #2

5. PORTA ETHERNET

La configurazione di fabbrica della porta ethernet è:

IP STATICO: 192.168.90.101

SUBNET MASK: 255.255.255.0

GATEWAY: 192.168.90.1

Non devono essere inseriti più dispositivi sulla stessa rete con lo stesso ip statico.



ATTENZIONE!

**NON CONNETTERE 2 O PIU' DISPOSITIVI CON LA CONFIGURAZIONE DI FABBRICA SULLA STESSA RETE ETHERNET PENA IL NON FUNZIONAMENTO DEL DISPOSITIVO
(CONFLITTO DI INDIRIZZI IP 192.168.90.101)**

6. AGGIORNAMENTO FIRMWARE

Al fine di migliorare, aggiungere o ottimizzare le funzionalità del prodotto, Seneca rilascia dei firmware aggiornati sulla sezione del dispositivo nel sito internet www.seneca.it

L'aggiornamento firmware viene effettuato tramite i tool Seneca oppure tramite il webserver.



ATTENZIONE!

PER NON DANNEGGIARE IL DISPOSITIVO NON TOGLIERE ALIMENTAZIONE DURANTE L'OPERAZIONE DI AGGIORNAMENTO DEL FIRMWARE.



ATTENZIONE!

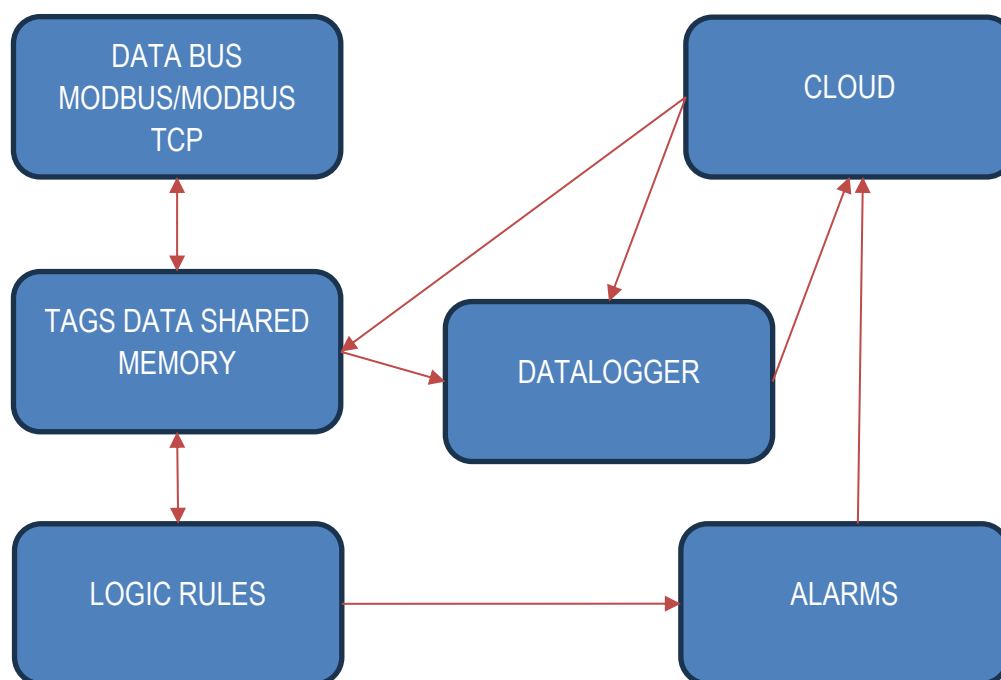
L'AGGIORNAMENTO FIRMWARE CANCELLERA' I DATI ACQUISITI DAL DATALOGGER, SALVARE I DATI PRIMA DI EFFETTUARE L'OPERAZIONE DI AGGIORNAMENTO.

7. ACQUISIZIONE ED ELABORAZIONE DEI DATI, INVIO DI DATI E ALLARMI

I dispositivi della serie "KEY-C" permettono di acquisire dati dai bus (tramite i protocolli di comunicazione industriale modbus RTU/ASCII e modbus TCP-IP), questi dati sono salvati in una memoria condivisa (shared) e possono essere elaborati tramite scalature oppure tramite le regole logiche. Una volta elaborati i dati questi sono salvati nella memoria interna e possono essere estratti tramite il webserver oppure inviati verso i cloud o server FTP/Email etc...

Gli allarmi sono generati dalle regole logiche e possono essere inviati anch'essi ai cloud.

Si faccia riferimento al seguente al seguente schema a blocchi:



L'acquisizione dei dati (Tag) nei bus (Data Bus) avviene attraverso i protocolli industriali Modbus.

Questi dati confluiscono nella memoria condivisa (Tags Data Shared Memory), in questa memoria le regole logiche (Logic Rules) eseguono le elaborazioni dei dati.

Il datalogger acquisisce i dati elaborati dalla Shared Memory, li archivia e li spedisce al cloud.

Le regole logiche generano allarmi che possono essere inviati al Cloud in tempo reale.

Il Cloud può accedere e quindi scrivere i dati già elaborati nella memoria condivisa (Shared Memory) e può cambiare il tempo di campionamento dei gruppi nel datalogger.

Di seguito analizzeremo i principali componenti dello schema a blocchi.

7.1. IL DATA BUS E I PROTOCOLLI INDUSTRIALI MODBUS

I dati risiedono in dispositivi esterni e devono essere connessi tramite protocolli industriali.

Il dispositivo include il protocollo modbus RTU/ASCII master e Modbus TCP-IP client in modo da potersi connettere con i più svariati produttori di terze parti. La memoria shared è anche accessibile dall'esterno sempre tramite protocollo Modbus.

7.1.1. PROTOCOLLI MODBUS



Modbus è nato come protocollo di comunicazione seriale da Modicon (azienda ora parte del gruppo Schneider Electric) per mettere in comunicazione i propri controllori logici programmabili (PLC). È diventato uno standard de facto nella comunicazione di tipo industriale, ed attualmente è uno dei protocolli di connessione più diffusi al mondo fra i dispositivi elettronici industriali. Oltre alla versione seriale i dispositivi Seneca supportano anche quella basata su Ethernet.

I protocolli Modbus supportati sono:

- Protocollo Modbus RTU Master
- Protocollo Modbus RTU Slave
- Protocollo Modbus TCP-IP Client
- Protocollo Modbus TCP-IP Server

Per maggiori informazioni si faccia riferimento al sito:

<https://modbus.org/>

Grazie a questi protocolli è possibile acquisire le variabili in memoria direttamente da dispositivi esterni Modbus RTU slave o Modbus TCP-IP server.

7.2. TAGS DATA SHARED MEMORY (MEMORIA CONDIVISA)

I dati acquisiti dai bus confluiscono nella memoria condivisa, questa memoria è accessibile dall'esterno del dispositivo con i protocolli Modbus.

Ogni dato è individuato da un nome mnemonico e da un tipo (intero, a virgola mobile etc...), così caratterizzato prende il nome di "Tag".

Su questi Tag è possibile effettuare vari tipi di elaborazioni come vedremo più avanti nel manuale.

7.3. IL DATALOGGER E LA CACHE

I Gateway della serie "KEY-C" Seneca includono un datalogger che permette di gestire fino a 300 variabili contemporaneamente (TAG). È anche possibile scalare ciascuna variabile ed effettuare ulteriori elaborazioni con le regole logiche. I dati acquisiti dal datalogger possono poi essere inviati ai diversi cloud o salvati nella memoria interna (cache).

I log della cache sono memorizzati nella memoria flash, quindi, se il trasferimento dei dati al cloud temporaneamente fallisce, questo può essere trasferito con successo in un secondo momento recuperando i dati da questa memoria interna.

Quando viene raggiunto il limite del numero di log nella cache si verifica una "rotazione", cioè i dati più vecchi vengono sovrascritti dai nuovi.

La memoria a disposizione per la cache dei log è suddivisa in 464 settori da 4Kbyte, 1 settore è utilizzato per la rotazione e quindi sono utilizzabili solo i primi 463 settori.

Il numero di campioni che è possibile salvare in memoria (NS) è dato dalla relazione:

$$NS = 463 * (DIV(\frac{4096}{10 * N_{TAG} + 22}))$$

Dove:

DIV è la divisione intera

N_{TAG} è il numero di TAG configurati

Ad esempio impostando 100 TAG si ha:

$$NS = 463 * \left(DIV \left(\frac{4096}{(10 * 100) + 22} \right) \right) = 463 * 4 = 1852$$

La durata massima temporale del buffer delle acquisizioni dipende anche dal tempo di acquisizione, per esempio fare riferimento alla tabella:

NUMERO DI TAG	NUMERO CAMPIONI	DI	TEMPO ACQUISIZIONE [s]	DI	DURATA DEL BUFFER CACHE [h]
----------------------	------------------------	-----------	-------------------------------	-----------	------------------------------------

1	59264	30	493 ore
10	15279	30	127 ore
100	1852	30	15 ore
1	59264	60	987 ore
10	15279	60	254 ore
100	1852	60	30 ore

I dati della cache possono anche essere esportati dal webserver in formato CSV.
Quando viene eseguito un reboot i dati nella cache sono mantenuti.

ATTENZIONE !

Quando viene eseguito un aggiornamento firmware i dati della cache vengono cancellati!

ATTENZIONE !

Quando viene modificata la struttura dei TAG (aggiunto/cancellato/modificato un TAG) i dati della cache vengono cancellati!

7.4. ELABORAZIONE DEI TAG: LE REGOLE LOGICHE

I dispositivi della serie "KEY-C" hanno la possibilità di definire un insieme di regole che realizzeranno un programma. Non è necessario conoscere un linguaggio di programmazione poiché le regole sono del tipo: "Se avviene questo evento allora esegui questa operazione, altrimenti quest'altra".

Per maggiori informazioni fare riferimento ai rispettivi capitoli del presente manuale.

7.5. CONNESSIONE AI CLOUD TRAMITE TECNOLOGIA "EASY CLOUD"

La tecnologia "Easy Cloud" si basa sul protocollo MQTT e permette la connessione bidirezionale con i principali cloud disponibili.

7.6. INVIO DI ALLARMI AI CLOUD

Per le logiche dell'allarmistica dei TAG si utilizzano le regole logiche (attraverso l'azione di invio di messaggi), un allarme è composto da un testo e da alcuni placeholder che possono includere alcune informazioni del tag e della macchina.

Per maggiori informazioni fare riferimento al capitolo 10.3.

8. MODALITA' DI FUNZIONAMENTO

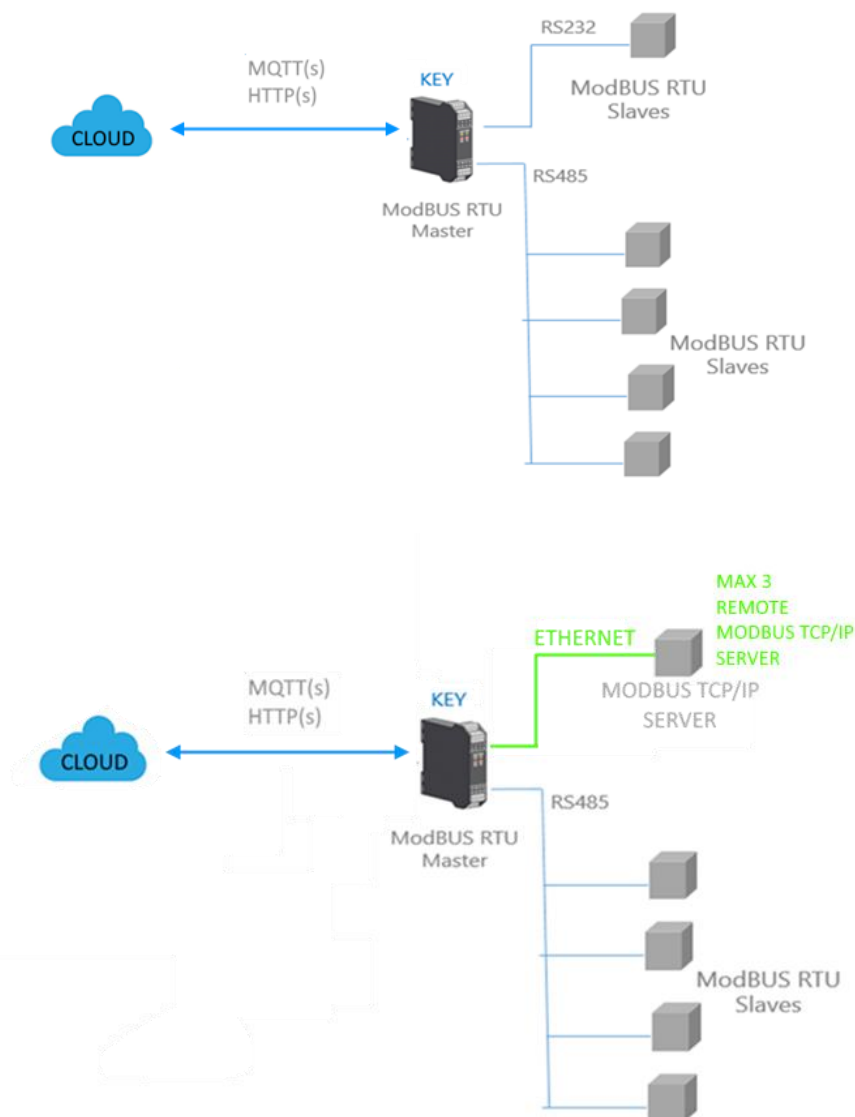
Il Gateway funziona nella modalità:

MODBUS SERIALE-ETHERNET MASTER/CLIENT TO CLOUD

8.1. MODBUS MASTER / CLIENT TO CLOUD

È possibile inviare dati da I/O di tipo Modbus RTU/ASCII Slave e/o TCP Server remoti verso un cloud (e vice versa).

Qui sotto alcuni esempi di connessione possibili:



Il Gateway, nella parte di campo funziona come un dispositivo Modbus master / Modbus Client e dall' altra parte come un client verso il broker MQTT o server HTTP tramite ethernet.

Le richieste Modbus (comandi di lettura o scrittura) vengono configurate nel dispositivo gateway.

Oltre ai dispositivi seriali è anche possibile connettere fino Nr 3 Modbus TCP-IP server remoti.

È anche possibile scrivere i TAG (e quindi i registri Modbus) dal cloud.

Il Gateway contemporaneamente attiva sempre un modbus TCP-IP server per poter accedere alla memoria shared dei tag.

8.2. DIAGNOSTICA SEMPLIFICATA DEI TAG

La diagnostica dei tag è visualizzabile tramite registri Modbus.

Il primo indirizzo Modbus, da cui parte la diagnostica semplificata, è di default il 49001 (Holding Register 9000).

Ogni bit rappresenta un tag con il seguente significato:

1 = TAG OK

0 = TAG FAIL

Il bit meno significativo è lo stato del tag nr 1

Il successivo è lo stato del tag nr 2 e così via...

Per esempio la lettura dei seguenti registri:

49001 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1

49002 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1

Significa: TAG 1, TAG 4, TAG17, TAG 18 , TAG 19, TAG 20 OK, tutti gli altri in FAIL.

All'avvio tutti i tag sono in stato di fail (tutti a 0).

8.3. DIAGNOSTICA ESTESA DEI TAG

Quando un tag è in stato di errore è possibile avere maggiori informazioni utilizzando la diagnostica estesa.

La diagnostica estesa riserva 1 byte per ciascun tag (poiché il limite è di 300 tag, ci sono 300 byte = 150 registri Modbus per la diagnostica estesa).

Questa diagnostica si trova alla fine della diagnostica semplificata (indirizzo Modbus di partenza di default è il 49033, Holding register 32).

Ogni registro Modbus contiene 2 tag, quindi ad esempio:

49033 TAG02_TAG01

49034 TAG04_TAG03

...

49182 TAG300_TAG299

49183 LAST_LOOP_TIME_COM1 [x1 ms]

49184 LAST_LOOP_TIME_COM2 [x1 ms]

Il significato del byte di diagnostica avanzata è:

VALORE BYTE	SIGNIFICATO	NOTE
0	OK	Il tag è correttamente letto/scritto
1	TIMEOUT	La risposta del tag è in timeout, ma verrà interrogato di nuovo
2	DELAYED	Troppi fail, il polling del tag è ritardato (il tag sarà interrogato nuovamente dopo il tempo di quarantena configurato)
3	EXCEPTION	Risposta di eccezione del Modbus ma il tag verrà interrogato di nuovo
4	CRC ERRORE	Risposta di eccezione del Modbus CRC ma il tag verrà interrogato di nuovo

Per esempio:

49033 0x0000

49034 0x0002

Significa che:

I TAG 1 e 2 sono OK (0x00 e 0x00)

Il TAG 03 è in stato di ritardo (0x02)

Il TAG 4 è OK (0x00)

LAST_LOOP_TIME_COMx è un registro che contiene l'ultimo tempo di interrogazione di tutti i tag seriali (in quanti di 10 ms) quindi, per esempio:

49183 25

49184 42

Significa che il loop della seriale 1 è stato di 250ms, il loop della seriale 2 è stato di 420ms.

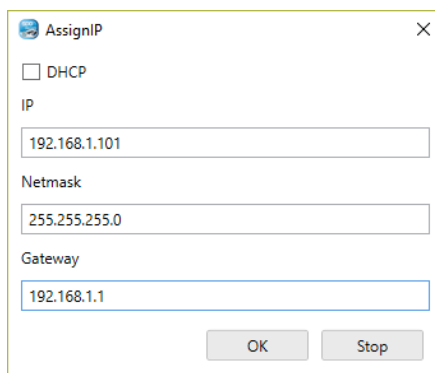
9. WEBSERVER DEI GATEWAY "-C"

9.1. GUIDA PASSO PASSO PER IL PRIMO ACCESSO AL WEBSERVER

PASSO 1: ALIMENTARE IL DISPOSITIVO E COLLEGARE LA PORTA ETHERNET

PASSO 2 SOFTWARE SENECA DISCOVERY DEVICE

Se è necessario cambiare l'indirizzo IP del dispositivo (default 192.168.90.101), lanciare il software Seneca Discovery Device ed eseguire lo SCAN, selezionare il dispositivo e premere il pulsante "Assign IP", impostare una configurazione compatibile con il proprio PC, ad esempio:



Confermare con OK. Ora il dispositivo è raggiungibile via ethernet dal proprio pc.

PASSO 3 ACCESSO AL WEBSERVER DI CONFIGURAZIONE

Inserire le credenziali di accesso:

user: admin

password: admin



ATTENZIONE!

**I WEB BROWSER DI CUI È STATA TESTATA LA COMPATIBILITÀ CON IL WEBSERVER DEL
DISPOSITIVO SONO:**

MOZILLA FIREFOX E GOOGLE CHROME.

NON È, QUINDI, ASSICURATO IL FUNZIONAMENTO CON ALTRI BROWSER

10. CONFIGURAZIONE DEL DISPOSITIVO DA WEBSERVER



ATTENZIONE!

I WEB BROWSER DI CUI È STATA TESTATA LA COMPATIBILITÀ CON IL WEBSERVER DEL DISPOSITIVO SONO:

MOZILLA FIREFOX E GOOGLE CHROME.

NON È, QUINDI, ASSICURATO IL FUNZIONAMENTO CON ALTRI BROWSER



ATTENZIONE!

DOPO IL PRIMO ACCESSO CAMBIARE USER NAME E PASSWORD AL FINE DI IMPEDIRE L'ACCESSO AL DISPOSITIVO A CHI NON È AUTORIZZATO.



ATTENZIONE!

SE I PARAMETRI DI ACCESSO AL WEBSERVER SONO STATI SMARRITI, PER ACCEDERE AL WEBSERVER, È NECESSARIO EFFETTUARE LA PROCEDURA DI RISPRISTINO ALLA CONFIGURAZIONE DI FABBRICA

10.1. PAGINA "SETUP"

Nessun file selezionato

	CURRENT	UPDATED
ETHERNET DHCP	Disabled	Disabled ▾
ETHERNET STATIC IP	192.168.90.101	192.168.90.101
ETHERNET STATIC IP MASK	255.255.255.0	255.255.255.0
ETHERNET STATIC GATEWAY	192.168.90.1	192.168.90.1
WORKING MODE	MODBUS GATEWAY ON PORT#1	MODBUS GATEWAY ON PORT#1 ▾
TIMEOUT RESPONSE MODE	NONE	NONE ▾
TCP/IP PORT	502	502

La prima colonna rappresenta il nome del parametro, la seconda colonna "current" è il valore corrente del parametro. L'ultima colonna "updated" è utilizzata per modificare la configurazione corrente.

Quando una configurazione è stata inserita è necessario confermarla con il pulsante "APPLY", a questo punto la nuova configurazione è operativa.

Se si desidera ripristinare i parametri di default, cliccare sul pulsante "FACTORY DEFAULT".

10.1.1.PARAMETRI DI CONFIGURAZIONE GENERALI

I parametri di configurazione generale sono spiegati di seguito:

DHCP

Disattivato: Viene impostato una Configurazione di Rete statica

Attivato: L'indirizzo IP, la maschera IP e l'indirizzo del gateway sono ottenuti dal server DHCP.

L'indirizzo del gateway può essere individuato dal software Seneca Discovery Device.

ETHERNET STATIC IP

Indirizzo IP statico quando il DHCP è disabilitato

ETHERNET STATIC IP MASK

Maschera quando il DHCP è disabilitato

ETHERNET STATIC GATEWAY

Indirizzo del gateway quando il DHCP è disabilitato

TCP/IP PORT

Porta TCP-IP per protocollo Modbus TCP-IP Server (È possibile collegare al gateway fino ad un massimo di 8 client)

PORT#n MODBUS PROTOCOL

Seleziona il protocollo seriale Modbus RTU o ASCII

PORT#n BAUDRATE

Seleziona il baudrate della porta seriale

PORT#n BIT

Seleziona il numero di bit per la comunicazione seriale.

PORT#n PARITY

Seleziona il tipo di parità della porta seriale (Nessuna, Pari o Dispari)

PORT#n STOP BITS

Imposta il numero di bit di stop della porta (1 o 2), si noti che se la parità è impostata, può essere utilizzato solo 1 bit di stop.

PORT#n TIMEOUT [ms]

Imposta il tempo di attesa per una risposta dal dispositivo seriale modbus slave, dopo questo tempo senza alcuna risposta si avrà un TIMEOUT.

PORT#n DELAY BETWEEN POLLS [ms]

Imposta la pausa tra due richieste Modbus master seriali successive.

PORT#n WRITING RETRIES

Imposta il numero di tentativi di scrittura sul (o sui) TAG prima di impostare lo stato di FAIL.

PORT#n MAX READ NUM

Imposta il Massimo numero di registri che possono essere letti con le funzioni di lettura multipla (il gateway ottimizzerà le letture con al massimo questo numero di registri). Va regolato in base al massimo numero di registri che si possono essere letti contemporaneamente dal dispositivo slave.

PORT#1 MAX WRITE NUM

Imposta il Massimo numero di registri che possono essere scritti con le funzioni di scrittura multipla (il gateway ottimizzerà le scritture con al massimo questo numero di registri).

WEB SERVER PORT

Imposta la porta TCP-IP per il Webserver.

WEB SERVER AUTHENTICATION USER NAME

Imposta il nome utente per l'accesso al Webserver (se nome utente e password sono lasciati vuoti non è necessaria alcuna autenticazione per l'accesso al Webserver)

WEB SERVER AUTHENTICATION PASSWORD

Imposta la password per l'accesso al Webserver (se nome utente e password sono lasciati vuoti non è necessaria alcuna autenticazione per l'accesso al Webserver)

**ATTENZIONE!**

CAMBIARE NEL WEBSERVER IL NOME UTENTE E LA PASSWORD DI DEFAULT PER LIMITARNE L'ACCESSO.

**ATTENZIONE!**

SE SI LASCIANO VUOTE LE DUE CASELLE DI TESTO DEI PARAMETRI RELATIVI ALL'AUTENTICAZIONE, QUESTA VIENE TOLTA.

WEBSERVER HTTPS

Forza il webserver ad usare il protocollo https sicuro invece che quello http

IP CHANGE FROM DISCOVERY

Imposta per impostare se un utente è autorizzato a modificare la configurazione IP dal software "Seneca Discovery Device".

DIAGNOSTIC REGISTERS MAPPING

Imposta il tipo di registro che conterrà la diagnostica semplificata ed avanzata. È possibile selezionare tra holding registers o input registers.

DIAGNOSTIC REGISTER START ADDRESS

Imposta l'indirizzo di partenza per i registri di diagnostica (default offset 9000 -> 49001 in caso di holding registers o 39001 in caso di input registers)

PORT #n TAGS QUARANTINE [s]

Quando un TAG è in FAIL questo viene messo in quarantena e non viene più interrogato per il tempo impostato.

MODBUS TCP-IP CLIENT

Abilita o meno i client Modbus TCP-IP, il gateway può collegarsi ad un massimo di 3 server Modbus TCP-IP.

TCP-IP PORT SERVER #n (Solo se attivo il Modbus TCP-IP client)

Utilizzato per impostare la porta #n del server TCP-IP

TCP-IP ADDRESS SERVER #n (Solo se attivo il Modbus TCP-IP client)

Utilizzato per impostare l'indirizzo IP del server #n

MODBUS TCP-IP CLIENT TIMEOUT [ms] (Solo se attivo il Modbus TCP-IP client)

Utilizzato per impostare il timeout di connessione per i client Modbus TCP-IP.

MODBUS TCP-IP CLIENT DELAY BETWEEN POLLS [ms] (Solo se attivo il Modbus TCP-IP client)

Imposta la pausa tra due richieste Modbus TCP-IP client successive.

MODBUS TCP-IP CLIENT WRITING RETRIES (Solo se attivo il Modbus TCP-IP client)

Imposta il numero di tentativi di scrittura sul (o sui) TAG prima di impostare lo stato di FAIL.

MODBUS TCP-IP CLIENT MAX READ NUM (Solo se attivo il Modbus TCP-IP client)

Imposta il Massimo numero di registri che possono essere letti con le funzioni di lettura multipla (il gateway ottimizzerà le letture con al massimo questo numero di registri).

MODBUS TCP-IP CLIENT MAX WRITE NUM (Solo se attivo il Modbus TCP-IP client)

Imposta il Massimo numero di registri che possono essere scritti con le funzioni di scrittura multipla (il gateway ottimizzerà le scritture con al massimo questo numero di registri).

SYNC CLOCK WITH INTERNET TIME

Permette di attivare l'aggiornamento della data/ora tramite la connessione ai server NTP ([RFC 5905](https://tools.ietf.org/html/rfc5905)).

ATTENZIONE!

AD OGNI SPEGNIMENTO IL DISPOSITIVO DEVE POTER RECUPERARE LA DATA / ORA DA UN SERVER NTP ALTRIMENTI QUESTA SARA' PRESA DALL'ULTIMO LOG ACQUISITO

NTP SERVER 1 ADDRESS

È l'indirizzo IP del primo server NTP (ad esempio 193.204.114.232 per l'NTP dell'INRIM)

NTP SERVER 2 ADDRESS

È l'indirizzo IP del secondo server NTP (nel caso il primo non risponda)

ATTENZIONE!

SI RICORDA CHE I SERVER NTP UTILIZZANO LA PORTA UDP 123 (CHE DEVE QUINDI ESSERE APERTA NELLA CONFIGURAZIONE DELLA RETE UTILIZZATA)
WATCHDOG ENABLE

WATCHDOG ENABLE

Abilita o no il riavvio a tempo del gateway.

WATCHDOG TIMEOUT [ore]

Imposta il tempo in ore dopo il quale il gateway effettuerà un riavvio forzato (solo in caso il parametro WATCHDOG ENABLE sia attivato).

10.1. PAGINA "DATALOGGER"

In questa pagina è possibile selezionare il tempo di acquisizione del datalogger e di scaricare i log dalla memoria internadel dispositivo in formato testo CSV.

È possibile definire fino ad 11 gruppi di TAG, ognuno di questi gruppi può inviare i propri TAG a tempi differenti (multipli del tempo di log).

10.2. PAGINA "SETUP TAG"

Nella modalità Modbus Tags Gateway è necessario definire i tag (ovvero le variabili) Modbus, per far questo è possibile utilizzare:

- *Il webserver*
- *Un template excel*

In questo capitolo verrà spiegata la configurazione del tag dal webserver.

Per editare i TAG tramite webserver accedere alla sezione "Setup tag" del menù di navigazione:

R-KEY-LT-C-HWE Setup TAG Firmware Version : 2027_107

[Select file](#) | Nessun file selezionato

[Load Tag File](#)

[Save to file current configuration](#)

[APPLY](#)

[ADD](#) | [CLONE](#) | [DELETE](#) | [MOVE UP](#) | [MOVE DOWN](#)

PRESS "CTRL" KEY TO SELECT MORE ROWS
MODBUS ADDRESS ARE 1-BASED (1=40001/20001...)
HOVER TO THE INPUT BOX TO SEE EQUIVALENT SENECA ADDRESS

GATEWAY TAG NR	GATEWAY MODBUS START REQUEST	GATEWAY TAG NAME	TARGET MODBUS DEVICE	TARGET RESOURCE	TARGET CONNECTED TO	TARGET MODBUS STATION ADDRESS	TARGET MODBUS REQUEST TYPE	TARGET MODBUS START REGISTER	TARGET REGISTER DATA TYPE	TAG INTERNAL INITIAL VALUE	GROUP TYPE	SCALE M FACTOR	SCALE Q FACTOR
1	1	TEST1	CUSTOM		PORT#1	1	HOLDING REGISTER	1	16BIT UNSIGNED	0	None	1.0	0.0
2	2	TEST2	CUSTOM		PORT#1	2	HOLDING REGISTER	1	16BIT UNSIGNED	0	None	1.0	0.0
3	3	TEST3	CUSTOM		PORT#1	3	HOLDING REGISTER	1	16BIT UNSIGNED	0	None	1.0	0.0
4	4	TEST4	CUSTOM		PORT#1	4	HOLDING REGISTER	1	16BIT UNSIGNED	0	None	1.0	0.0

Selezionando con il mouse una riga (questa diventerà gialla come in figura) e sarà possibile clonarla, cancellarla o spostarla.

È possibile selezionare più righe cliccando con il mouse la riga e mantenendo premuto il tasto CTRL.

GATEWAY TAG NR

È il numero del tag, è possibile definire un massimo di 300 tag.

GATEWAY MODBUS START ADDRESS

Imposta l'indirizzo della posizione di memoria del Gateway in cui è salvato il TAG, questi registri sono accessibili sia da Modbus seriale che da Modbus TCP-IP.

GATEWAY TAG NAME

Imposta il nome mnemonico del tag (verrà visualizzato nella pagina "STATUS" e sarà utilizzato nelle regole logiche)

TARGET MODBUS DEVICE

Seleziona il modello di Modbus RTU slave da database dei dispositivi Seneca o seleziona "custom" se non si utilizza uno slave Modbus RTU Seneca.

TARGET RESOURCE

Se si utilizza un Seneca Modbus RTU Slave seleziona il nome della risorsa dal database Seneca.

TARGET CONNECTED TO

Selezionare a quale porta seriale del gateway è collegato il dispositivo slave modbus rtu.

(nel caso di R-KEY-LT è disponibile solo la porta COM 1).

Oltre alle porte seriali è possibile recuperare i dati da un massimo di 3 server Modbus TCP-IP remoti.

Un tag può anche essere non legato ad una acquisizione modbus ma essere una variabile interna che verrà utilizzata nelle regole logiche, in questo caso selezionare "INTERNAL".

TARGET MODBUS STATION ADDRESS

Definisce il Modbus Station Address (chiamato anche indirizzo del nodo Modbus) del dispositivo slave.

TARGET MODBUS REQUEST TYPE

Selezionare il tipo di registro Modbus:

Coil
Discrete Input
Holding Register
Input Register

TARGET MODBUS START REGISTER ADDRESS

Definisce il registro di partenza del TAG da acquisire dello slave Modbus RTU.

TARGET REGISTER DATA

Selezionare il tipo di variabile del TAG:

16 BIT SIGNED: variabile a 16 bit da -32768 a +32767

16 BIT UNSIGNED: variabile a 16 bit da 0 a 65535

32 BIT SIGNED MSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word più significativa, può assumere valori da -2147483648 a +2147483647

32 BIT UNSIGNED MSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word più significativa, può assumere valori da 0 a 4294967295

32 BIT SIGNED LSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word meno significativa, può assumere valori da -2147483648 a +2147483647

32 BIT UNSIGNED LSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word meno significativa, può assumere valori da 0 a 4294967295

32 BIT REAL MSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word più significativa, valore a virgola mobile a singola precisione (IEEE 758-2008)

32 BIT REAL LSW: 2 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word meno significativa, valore a virgola mobile a singola precisione (IEEE 758-2008)

BIT: 1 Coil booleano o Discrete Input, valore true o false.

64 BIT SIGNED MSW 4 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word più significativa, può assumere valori da -9223372036854775808 a +9223372036854775807

64 BIT SIGNED LSW 4 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word meno significativa, può assumere valori da -9223372036854775808 a +9223372036854775807

64 BIT UNSIGNED MSW 4 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word più significativa, può assumere valori da 0 a 18446744073709551615

64 BIT UNSIGNED LSW 4 registri modbus il cui registro Modbus con l'indirizzo inferiore contiene la word meno significativa, può assumere valori da 0 a 18446744073709551615

16 BIT SIGNED SCALED TO REAL MSW legge un registro 16 bit signed da modbus e lo converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: TAG SCALED = (TAG * "SCALE M FACTOR")+"SCALE Q FACTOR".

16 BIT SIGNED SCALED TO REAL LSW legge un registro 16 bit signed da modbus e lo converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: TAG SCALED = (TAG * "SCALE M FACTOR")+"SCALE Q FACTOR".

16 BIT UNSIGNED SCALED TO REAL MSW legge un registro 16 bit unsigned da modbus e lo converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag

prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

16 BIT UNSIGNED SCALED TO REAL LSW legge un registro 16 bit unsigned da modbus e lo converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT SIGNED MSW SCALED TO REAL MSW legge due registri interpretandoli come 32 bit signed da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT SIGNED MSW SCALED TO REAL LSW legge due registri interpretandoli come 32 bit signed da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT UNSIGNED MSW SCALED TO REAL MSW legge due registri interpretandoli come 32 bit unsigned da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT UNSIGNED MSW SCALED TO REAL LSW legge due registri interpretandoli come 32 bit unsigned dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT SIGNED LSW SCALED TO REAL MSW legge due registri interpretandoli come 32 bit signed dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT SIGNED LSW SCALED TO REAL LSW legge due registri interpretandoli come 32 bit signed dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT UNSIGNED LSW SCALED TO REAL MSW legge due registri interpretandoli come 32 bit unsigned dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

32 BIT UNSIGNED LSW SCALED TO REAL LSW legge due registri interpretandoli come 32 bit unsigned dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT SIGNED MSW SCALED TO REAL MSW legge 4 registri interpretandoli come 64 bit signed dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT SIGNED MSW SCALED TO REAL LSW legge 4 registri interpretandoli come 64 bit signed dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT UNSIGNED MSW SCALED TO REAL MSW legge 4 registri interpretandoli come 64 bit unsigned dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT UNSIGNED MSW SCALED TO REAL LSW legge 4 registri interpretandoli come 64 bit unsigned dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT SIGNED LSW SCALED TO REAL MSW legge 4 registri interpretandoli come 64 bit signed dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT SIGNED LSW SCALED TO REAL LSW legge 4 registri interpretandoli come 64 bit signed dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT UNSIGNED LSW SCALED TO REAL MSW legge 4 registri interpretandoli come 64 bit unsigned dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $\text{TAG SCALED} = (\text{TAG} * \text{"SCALE M FACTOR"}) + \text{"SCALE Q FACTOR"}$.

64 BIT UNSIGNED LSW SCALED TO REAL LSW legge 4 registri interpretandoli come 64 bit unsigned dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore

contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $TAG\ SCALED = (TAG * "SCALE\ M\ FACTOR") + "SCALE\ Q\ FACTOR"$.

32 BIT REAL MSW SCALED TO REAL MSW legge 2 registri interpretandoli come real dove l'indirizzo inferiore contiene la word più significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $TAG\ SCALED = (TAG * "SCALE\ M\ FACTOR") + "SCALE\ Q\ FACTOR"$.

32 BIT REAL LSW SCALED TO REAL MSW legge 2 registri interpretandoli come real dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word più significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $TAG\ SCALED = (TAG * "SCALE\ M\ FACTOR") + "SCALE\ Q\ FACTOR"$.

32 BIT REAL LSW SCALED TO REAL LSW legge 2 registri interpretandoli come real dove l'indirizzo inferiore contiene la word meno significativa da modbus e li converte in un real dove l'indirizzo inferiore contiene la word meno significativa. Se si desidera è anche possibile scalare il tag prima della conversione secondo la formula: $TAG\ SCALED = (TAG * "SCALE\ M\ FACTOR") + "SCALE\ Q\ FACTOR"$.

N.B. Questo campo viene compilato automaticamente se nel campo "TARGET MODBUS DEVICE" è stato selezionato un dispositivo slave Seneca.

ATTENZIONE!

***Tutti i valori a 32 bit sono memorizzati in 2 registri consecutivi, ad esempio:
Il Totalizzatore TAG 1 in tipo MSW unsigned a 32 bit è memorizzato negli indirizzi 40016 e 40017:
La parola più significativa è il 40016, quella meno significativa è il 40017.
Quindi il valore a 32bit si ottiene dalla seguente relazione:
 $Totalizer1 = (40017) + (Reg\ (40016) \times 65536)$***

Allo stesso modo tutti i valori a 64 bit sono memorizzati in 4 registri consecutivi.

INTERNAL TAG INITIAL VALUE

Seleziona il valore iniziale del Tag di tipo internal (cioè è un Tag che non proviene da Modbus ma definito come variabile interna per le regole logiche).

GROUP TYPE

Seleziona il gruppo associato al TAG

SCALE M FACTOR / SCALE Q FACTOR

Seleziona la scalatura da applicare al TAG secondo la formula:

$$Scaled\ Tag = (TAG * M) + Q$$

La scalatura non è disponibile per tutti i TARGET REGISTER DATA.

10.2.1. Vista in tempo reale del Modbus Gateway: LA PAGINA "STATUS"

Una volta che i TAG sono configurati è possibile visualizzare in tempo reale lo stato della comunicazione Modbus, dalla sezione Status del menù di navigazione.

La visualizzazione in tempo reale mostrerà la corrente configurazione di rete, la modalità di funzionamento e le informazioni sui TAGS.



R-KEY-LT-HW3 **Status** **Firmware Version : 1800_123**

Status	DHCP : Disabled
Setup	ACTUAL IP ADDRESS : 192.168.90.101
Setup TAG	ACTUAL IP MASK : 255.255.255.0
Firmware Update	ACTUAL GATEWAY ADDRESS: 192.168.90.1
Serial Traffic Monitor	ACTUAL MAC ADDRESS: c8-f9-81-0e-1e-11
	WORKING MODE: Modbus Tags Gateway Ethernet to Serial (PORT#1 MASTER)
	PORT#1 LOOP TIME [ms]: 601
	PORT#2 LOOP TIME [ms]: 0

REBOOT

Page : 1/10 PREVIOUS PAGE NEXT PAGE

GATEWAY TAG NR	GATEWAY TAG NAME	GATEWAY MODBUS START REGISTER	TAG DATA TYPE	TAG VALUE	TAG READING STATUS	
1	BICI	40001	16BIT UNSIGNED	0	DELAYED	CHANGE

Le informazioni sui Tag includono: Il nome, l'indirizzo Modbus del Gateway, il valore e lo stato.

Per il campo stato vale la seguente legenda:

OK = TAG privo di errori

FAIL_TO = Timeout Lettura del TAG

DELAYED = Raggiunto il numero di retry impostato, il polling del tag è ritardato (il tag sarà interrogato nuovamente dopo il tempo di quarantena configurato)

EXC = risposta di eccezione del protocollo Modbus

10.3. PAGINA "SETUP TEXT MESSAGE" (ALLARMI)

Qui è possibile definire il testo degli allarmi che saranno spediti verso il cloud.

Il testo del messaggio può contenere solo caratteri ASCII.

È possibile utilizzare la sintassi {NOME_TAG} per includere nel testo il valore attuale di un tag.

Ad esempio il testo del messaggio:

"LIVELLO ACQUA ={LEVEL} m"

Fornirà un testo con il valore del tag riportato come testo, se il tag "LEVEL" vale 1.232 si avrà:

LIVELLO ACQUA = 1.232 m

Questa sintassi può essere utilizzata più di una volta nel testo di un messaggio.

Ogni messaggio ha un campo ID che è usato per associare il messaggio all'allarme nelle regole logiche.

Oltre a questo è possibile aggiungere anche i seguenti placeholder:

Legenda for message code Format	
Format	Meaning
%c	device Client ID
%m	device MAC Address
%M	device MAC Address without dot separator
%d	date-time
%t	timestamp (number of seconds since the "epoch")
%u	timestamp (number of milliseconds since the "epoch")

La pubblicazione dei messaggi di allarme avviene nel topic indicato nel campo:

"Cloud Topic to send message" della sezione setup cloud.

10.4. PAGINA "SETUP TIMER"

Questa sezione consente di definire i timer da utilizzare nelle regole logiche.

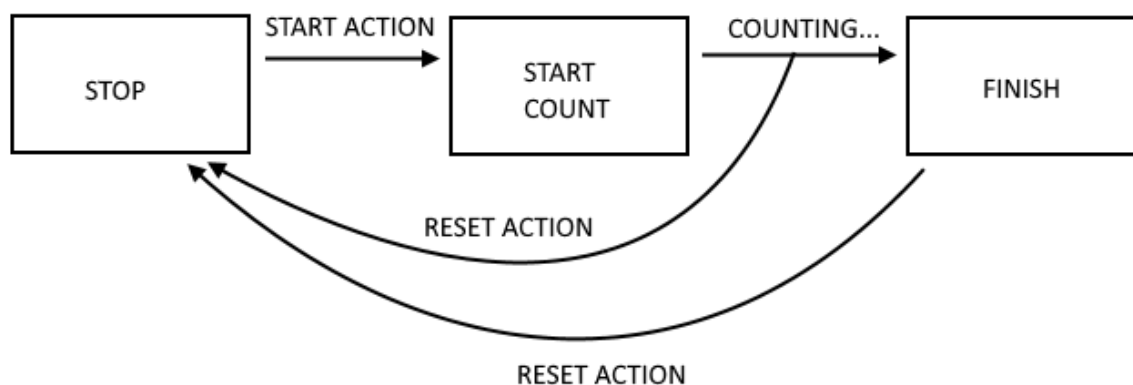
L'ID rappresenta il mnemonico del timer che deve essere utilizzato nelle regole.

"Enable" seleziona se il timer è attivo o meno.

"Duration" è il valore di attivazione in [ms].

Nota

I timer per impostazione predefinita sono in modalità di stop, necessitano di un'azione per l'avvio e di un'azione per il ripristino, secondo lo schema seguente:



10.5. PAGINA "SETUP RULES"

In questa sezione è possibile definire un insieme di regole logiche che realizzeranno un programma.

È importante notare che le scritture dei Tag avvengono "During execution" cioè la scrittura dei tag è eseguita subito dopo aver eseguito l'azione di scrittura.

Per configurare una regola, sono disponibili i seguenti parametri:

10.5.1. RULE CONFIGURATION

ID

Ordine di esecuzione della regola (1 = Prima regola ad essere eseguita)

Enabled

Indica se la regola è abilitata oppure se deve essere esclusa dall'esecuzione

Description

Descrizione testuale mnemonica della regola

Period [ms]

Se il valore è = 0, le azioni vengono eseguite in modalità "one time" o "repeat" (in questo caso verranno eseguite alla massima velocità possibile).

Nella modalità "one time" l'azione verrà eseguita solo se c'è una modifica nel risultato della condizione (if o else) (cioè su cambio di stato da false a true).

Nella modalità "repeat" l'azione verrà eseguita ad ogni loop cercando di rispettare la tempistica indicata.

Se il valore è > 0 le azioni vengono eseguite solo in modalità "repeat".

ATTENZIONE!

Utilizzare valori di periodo adeguati per le azioni di invio tramite MQTT/HTTP!

Se Period > 0 le azioni vengono sempre eseguite in modalità "repeat"

Le azioni di invio di messaggi sono eseguite in modalità one time (solo una volta su passaggio della condizione da false a true) se period = 0, sono ripetute in modalità repeat se period > 0.

10.5.2. IF CONDITION: TYPE

Questa sezione definisce il tipo di condizione, sono possibili i seguenti tipi:

None

Nessuna condizione da valutare

Always True

La condizione If è sempre vera.

Nota che la regola viene eseguita solo una volta se Period è = 0 ms o se il campo "ACTION MODE" dell'azione è in modalità "One Time".

Se è necessario eseguire una regola ad ogni ciclo, è necessario mettere le azioni in "repeat mode".

Se è necessario eseguire una regola a tempo (ogni x ms), è necessario impostare Period > 0ms.

Always False

La condizione If è sempre falsa.

Può essere utilizzata per fini di debug (bloccando la condizione in modo veloce).

Digital Tag

La condizione dipende dallo stato di un tag digitale:

Campo	Significato
Tag	Seleziona il tag che deve essere utilizzato per la condizione
Operator	Può valere solo "="
Tag / Constant value	Seleziona se il confronto è tra un altro tag digitale o un valore booleano costante (TRUE o FALSE)

Analog Tag

La condizione dipende da un confronto con un TAG analogico

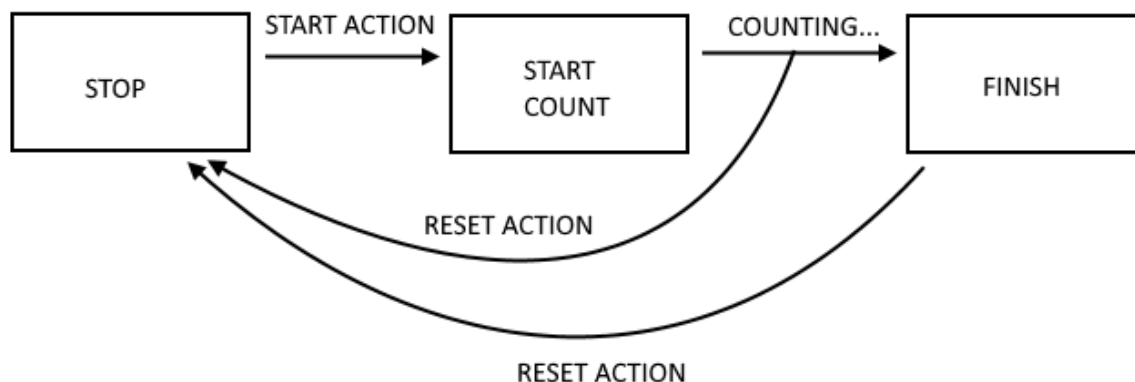
Campo	Significato
Tag	Seleziona il tag che deve essere utilizzato per la condizione
Operator	Può valere: "="
	">"
	"<"
	">="
	"<="
Tag / Constant value	Seleziona se il confronto è tra un altro tag analogico o un valore costante

Timer

La condizione dipende dallo stato del timer selezionato

Campo	Significato
ID	Selezionare l'ID del timer da utilizzare
Expired	Può essere: "OFF" o "ON" Con "ON" la condizione è vera solo allo scadere del timer (stato FINISH). Con "OFF" la condizione è vera fino a quando il timer non è in STOP o COUNTING. Quando il timer è nello stato FINISH la condizione diventa falsa.

Il funzionamento del Timer è rappresentato nello schema seguente:



Scheduler

La condizione dipende dallo scheduler (calendario) impostato:

Campo	Significato
Type	Può valere: Every Day, Every week, Every Month Every Day: la condizione è vera ogni giorno all'ora e minuti configurati Every Week: la condizione è vera una volta a settimana il giorno della settimana selezionato alle ore e minuti selezionati Every Month: la condizione è vera una volta al mese il giorno del mese selezionato alle ore e minuti selezionati
Day	Se il tipo è Weekly stabilisce il giorno della settimana: 0 = Domenica 1 = Lunedì 2 = Martedì 3 = Mercoledì 4 = Giovedì 5 = Venerdì 6 = Sabato Se il tipo è Monthly: Seleziona il giorno del mese da 1 a 31
Hour	Ore
Minute	Minuti

Rule Status

La condizione dipende dall'abilitazione o no di una regola:

Campo	Significato
ID	Seleziona l'ID della regola
Enabled	Seleziona tra "enabled" o "disabled" Se "Enabled" la condizione è VERA se la regola selezionata è abilitata. Se "Disabilitato" la condizione è VERA se la regola selezionata è disabilitata.

Bitmask

La condizione dipende dalla mascheratura di un tag con una costante esadecimale:

Campo	Significato
Tag	Seleziona il tag a cui applicare la maschera di bit da un elenco contenente tutti i tag con tipo di dato "16Bit Unsigned"
Mask	La maschera di bit rappresentata come una stringa di 4 cifre esadecimali

La condizione di mascheratura "Bitmask" è VERA se l'operazione AND bit per bit tra il Tag e la Maschera dati è diversa da 0; FALSO altrimenti.

Esempio:

Tag=0x1233 (esadecimale) = 0b 0001 0010 0011 0011 (binario)

Mask=0x8001 (esadecimale) = 0b 1000 0000 0000 0001 (binario)

Significa che la maschera analizza il bit0 (meno significativo) e il bit 15 (più significativo) del Tag.

L' AND bit a bit fornisce:

0001 0010 0011 0011

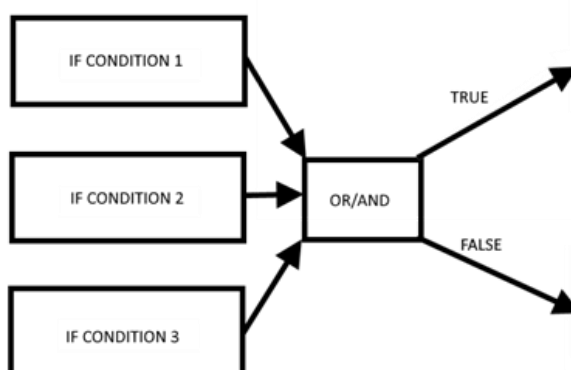
1000 0000 0000 0001

0000 0000 0000 0001

Per cui la condizione è VERA.

10.5.3. IF CONDITION OPERATOR

Le "condizioni IF" possono essere combinate insieme in logica "OR" o "AND", in pratica:



Le "condizioni IF" legate assieme da "OR" assumono lo stato TRUE se almeno una delle condizioni è vera.

Le "condizioni IF" legate assieme da "AND" assumono lo stato TRUE solo se tutte sono vere.

Più in dettaglio seguono la seguente tabella:

IF CONDITION 1	IF CONDITION 2	IF CONDITION 3	"OR"	"AND"
FALSE	FALSE	FALSE	FALSE	FALSE

FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE
TRUE	FALSE	TRUE	TRUE	FALSE
TRUE	TRUE	FALSE	TRUE	FALSE
TRUE	TRUE	TRUE	TRUE	TRUE

10.5.4.THEN/ELSE ACTION

In questa sezione è possibile definire l'azione che deve essere eseguita nel caso le condizioni diano come risultato TRUE (azione THEN) o FALSE (azione ELSE).

NONE

Nessuna azione da eseguire

Digital Tag

Esegue una scrittura su un Tag di tipo digitale.

Campo	Significato
Action Mode	Permette di selezionare tra "One Time" o "Repeat". Con "One Time" l'azione viene eseguita solo se c'è un cambiamento nel risultato da false a true della condizione IF o ELSE Con "Repeat" l'azione viene eseguite ad ogni loop (o al tempo impostato da period).
Destination Tag	È il tag in cui viene copiato il risultato TRUE/FALSE calcolato
Operator	È l'operatore booleano da utilizzare, selezionato tra =, NOT, OR ecc ...
Source Tag 1 / Constant value 1	Seleziona il primo tag da utilizzare nel calcolo booleano. È possibile anche usare una costante booleana
Source Tag 2 / Constant value 2	Selezionare il secondo Tag se l'operatore necessita di 2 input (Ad esempio operatore "OR"). È anche possibile utilizzare una costante booleana

Analog Tag

Esegue una scrittura su un Tag di tipo analogico.

Campo	Significato
Action Mode	Selezionare tra "One Time" o "Repeat". Con "One Time" l'azione viene eseguita solo se c'è un cambiamento nel risultato da false a true della condizione IF o ELSE Con "Repeat" l'azione viene eseguite ad ogni loop (o al tempo impostato da period).
Destination Tag	È il tag in cui viene copiato il risultato calcolato
Operator	È l'operatore matematico da utilizzare, è possibile selezionare tra: "="
	copia il tag di origine 1 oppure il valore costante 1 nel tag di destinazione

	Esempio: Tag di destinazione = Tag di origine 1 Oppure Tag di destinazione = valore costante 1 "+" = Somma al tag di destinazione il valore del tag di origine1 oppure il valore costante 1 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag di destinazione + Tag di origine 1 "-" = Sottrae al tag di destinazione il valore del tag di origine1 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag di destinazione - Tag di origine 1 "*" = Moltiplica il tag di destinazione per il valore di tag di origine 1 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag di destinazione * Tag di origine 1 "/" = Divide il tag di destinazione con il valore di tag di origine 1 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag di destinazione / Tag di origine 1 "%" = Calcola il resto della divisione dal tag di destinazione e il valore del tag di origine1 e copia il risultato nel tag di destinazione. (Notare che 53% 7 = 4) Esempio: Tag di destinazione = Tag di destinazione% Tag di origine1 "abs" Calcola il valore assoluto di Source Tag 1 o Constant value 1 e copia il risultato nel Destination Tag (Notare che abs (-4) = 4)
--	---

	Esempio: Tag di destinazione = abs (Tag sorgente 1) "Sqrt" Calcola il valore della radice quadrata del tag sorgente 1 o valore costante 1 e copia il risultato nel tag di destinazione. (Notare che $\text{sqrt}(9) = \sqrt{9} = 3$) Esempio: Tag di destinazione = sqrt (tag di origine 1) "Sqr" Calcola il valore quadrato del tag di origine 1 o valore costante 1 e copia il risultato nel tag di destinazione. (Notare che $\text{sqr}(3) = 3^2 = 9$) Esempio: Tag di destinazione = sqr (tag di origine 1) "Log" Calcola il logaritmo decimale del tag sorgente 1 o valore costante 1 e copia il risultato nel tag di destinazione. (Notare che $\log(3) = 0,4771212$) Esempio: Tag di destinazione = log (tag di origine 1) "Ln" Calcola il logaritmo naturale del tag di origine 1 o valore costante 1 e copia il risultato nel tag di destinazione. (Notare che $\ln(3) = 1.09861228867$) Esempio: Tag di destinazione = ln (Tag sorgente 1) "Exp" Calcola il numero di Eulero elevato a Source Tag 1 o Constant value 1 e copia il risultato nel Destination Tag. Si noti che: $\ln(\exp 3) = 3$ Esempio: Tag di destinazione = scadenza (tag di origine 1) "+"
--	--

	Somma il Source Tag 1 o Constant value 1 Con il valore di Source Tag 2 o Constant value 2 e copia il risultato nel Destination Tag. Esempio: Tag di destinazione = Tag sorgente 1+ Tag sorgente 2 "_" Sottrae il tag sorgente 1 o valore costante 1 con il valore del tag sorgente 2 o valore costante 2 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag di origine 1- Tag di origine 2 "**" Moltiplicare il tag di origine 1 o valore costante 1 con il valore di tag di origine 2 o valore costante 2 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag sorgente 1 * Tag sorgente 2 "/" Divide il tag di origine 1 o valore costante 1 con il valore di tag di origine 2 o valore costante 2 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = Tag sorgente 1 / Tag sorgente 2 "%" Calcola il resto della divisione tra il tag sorgente 1 o valore costante 1 e il valore del tag sorgente 2 o valore costante 2 e copia il risultato nel tag di destinazione. (Notare che 53% 7 = 4) Esempio: Tag di destinazione = Tag sorgente 1% Tag sorgente 2 "Pow" Calcola il valore Source Tag1 o Constant 1 elevato alla potenza del Source Tag2 / Constant value 2 e copia il risultato nel tag di destinazione. Esempio: Tag di destinazione = (Source Tag1) ^ (Source Tag2)
Source Tag 1 / Constant value 1	Seleziona il tag da utilizzare come ingresso 1 per l'operatore utilizzato. È possibile utilizzare anche usare un valore costante.

Source Tag 2 / Constant value 2	Seleziona il tag da utilizzare come ingresso 2 nel calcolo se l'operatore necessita di 2 ingressi. Puè anche essere utilizzato un valore costante.
---------------------------------	---

Timer

È possibile selezionare l'azione da eseguire nel timer selezionato.

Campo	Significato
Id	Seleziona il timer tra quelli configurati
Action	Seleziona il tipo di azione da eseguire nel timer selezionato. "Start" esegue l'azione di avvio del timer selezionato "Reset" esegue l'azione di reset del timer allo stato di stop

Rule Status

L'azione abilita o disabilita una regola.

Campo	Significato
Id	Seleziona la regola
Enable	Seleziona se l'azione deve o no abilitare la regola selezionata: "OFF" disabilita la regola selezionata "ON" abilita la regola selezionata

Bitmask

Questa azione permette di portare al valore 1 o al valore 0 un numero configurabile di bit di un determinato tag.

Campo	Significato
Action Mode	Seleziona tra "One Time" o "Repeat". Con "One Time" l'azione viene eseguita solo se c'è un cambiamento nel risultato da false a true della condizione IF o ELSE Con "Repeat" l'azione viene eseguite ad ogni loop (o al tempo impostato da period).
Destination Tag	È il tag in cui viene copiato il risultato dell'azione, il tag deve essere di tipo "16 bit unsigned"
Source Tag	Seleziona il tag da utilizzare nel calcolo. È possibile anche inserire il source tag ed il destination tag uguali in modo da eseguire l'azione sullo stesso TAG. Il tag deve essere di tipo "16 bit unsigned"

Mask	È la maschera in formato esadecimale che permette la mascheratura dei bit da controllare.
Action	È possibile scegliere tra "Set" ovvero porta ad 1 i bit, oppure "Reset" ovvero porta a 0 i bit.

Send Message

Esegue una scrittura su un Tag di tipo stringa.

Campo	Significato
Message	È l'ID del messaggio da inviare via MQTT/HHTP

10.6. PAGINA "CLOUD SETUP"

In questa pagina è possibile configurare la connessione verso il cloud dei Tag configurati.

CLOUD PROTOCOL

Seleziona il protocollo da usare tra MQTT e http

CUSTOM CLOUD

Seleziona se utilizzare o no i campi preconfigurati per un cloud specifico.

Attualmente è possibile configurare:

None: Tramite la configurabilità del dispositivo è possibile virtualmente connettersi ad ogni cloud

Direl: Imposta il dispositivo per la connessione con il cloud Direl ADM

Per aggiungere alla lista altri cloud è possibile formulare una richiesta a Seneca.

CLOUD SERVER ADDRESS

Seleziona l'indirizzo del cloud a cui ci si deve connettere

CLOUD SERVER PORT

Seleziona la porta del server

MQTT CLIENT ID/HTTP PATH

Definisce il Client ID usato nel protocollo MQTT o la path di pubblicazione sul server HTTP

MQTT WEBSOCKET

Permette di attivare la comunicazione MQTT tramite Websockets

MQTT KEEP ALIVE INTERVAL [s]

Questo parametro definisce il Keep alive il quale assicura che la connessione tra il broker e il client sia ancora aperta e che il broker e il client siano consapevoli di essere connessi. Quando il client stabilisce una connessione al broker, comunica al broker un intervallo di tempo in secondi. Questo intervallo definisce il periodo di tempo massimo durante il quale il broker e il client possono non comunicare tra loro.

MQTT CLEAN SESSION

Questo parametro definisce la "clean session". Quando il flag di clean session è impostato su true, il client non desidera una sessione persistente. Se il client si disconnette per qualsiasi motivo, tutte le informazioni e i messaggi accodati da una precedente sessione vengono persi.

MQTT MESSAGE RETAIN

Normalmente se un publisher pubblica un messaggio su un topic a cui nessuno è sottoscritto, il messaggio viene semplicemente scartato dal broker. Tuttavia il publisher può dire al broker di conservare l'ultimo messaggio di quel topic

MQTT QUALITY OF SERVICE [QOS]

Questo parametro definisce il QOS del protocollo MQTT.

Può essere selezionato tra

QOS 0 (solo una volta, senza ack)

QOS 1 (almeno una volta, con ack)

QOS 2 (solo una volta, con ack e rinvio)

CLOUD AUTHENTICATION

Questo parametro definisce se deve essere utilizzata l'autenticazione con utente / password per l'accesso al cloud

CLOUD AUTHENTICATION USER

Username del broker o server

CLOUD AUTHENTICATION PASSWORD

Password del broker o server

CLOUD SSL/TLS

Definisce se attivare il protocollo di sicurezza SSL/TLS 1.2

CLOUD CLIENT CERTIFICATE REQUIRED

Definisce se è necessario gestire i certificati x.509 per la connessione SSL/TLS

CLOUD CLIENT CERTIFICATE VALIDITY CHECK

Se attivato verifica che i certificati siano validi

CLOUD LOG ON CHANGE

Aggiorna i valori sul broker mqtt o sul server http solo su cambiamento e non più a tempo

CLOUD PUBLISH MULTIPLE TAGS

Per il protocollo MQTT questo parametro definisce se la publish contiene più tag o se il dispositivo deve inviare una publish per ciascun tag.

Per il protocollo HTTP questo parametro definisce se la post contiene più tag o se il dispositivo deve inviare una post per ciascun tag.

CLOUD PUBLISH TOPIC FOR LOGS

Seleziona il nome del topic per i log utilizzando la seguente tabella:

%c	Device Client ID
----	------------------

%m	Device MAC Address
%j[field]	Aggiunge i doppi apici " a [field]. I doppi apici rappresentano una stringa in JSON

Ad esempio:

Se:

Device Client ID = Padova13

Publish Topic for Logs = seneca/%c/data

Si avrà che i log dei dati sono inviati al topic: Seneca/Padova13/data

CLOUD PUBLISH PAYLOAD FOR LOGS

Seleziona il formato che deve essere utilizzato per il payload utilizzando la seguente tabella:

%c	Device Client ID
%m	Device MAC Address
%d	data-ora
%t	timestamp (numero di secondi dal 01/01/1970)
%tms	timestamp (numero di millisecondi dal 01/01/1970)
%b	bulk (formato specificato in "Publish Bulk Format")
%n	Nome del tag (solo per "Publish Bulk Format")
%i	Id univoco della variabile
%v	Valore del tag (solo in "Publish Bulk Format")
%j[field]	Aggiunge i doppi apici " a [field]. I doppi apici rappresentano una stringa in JSON

Nota: il placeholder %i aggiunge un ID univoco alla variabile da pubblicare secondo l'ordine del TAG (vedi pagina Tag view)

CLOUD PUBLISH BULK FORMAT

Seleziona il formato per il "bulk mode" secondo la seguente tabella:

%c	Device Client ID
%m	Device MAC Address
%d	Data/ora
%t	timestamp (numero di secondi dal 01/01/1970)
%n	Nome del tag (solo per "Publish Bulk Format")
%v	Valore del tag (solo in "Publish Bulk Format")
%j[field]	Aggiunge i doppi apici " a [field]. I doppi apici rappresentano una stringa in JSON

CLOUD SUBSCRIBE TOPIC FOR COMMANDS

Per scrivere un tag tramite MQTT, il dispositivo deve ricevere una PUBLISH dal cloud stesso con il formato indicato in questo campo vedi capitolo "SCRITTURE DA CLOUD VERSO IL DISPOSITIVO".

SITE
SPACE
MACHINERY

Rappresentano tre campi testo che possono essere utilizzati dai cloud per identificare il dispositivo.

CLOUD TOPIC TO SEND MESSAGE

E' il topic su cui verranno pubblicati i messaggi di allarme definiti nella pagina "Setup Text Messages". La logica di invio degli messaggi di testo (allarmi) è definita nelle regole logiche.

Alla fine della pagina è possibile caricare i certificati e la chiave per la connessione al server MQTT in formato .pem

MQTT CA CERTIFICATE :

NOT PRESENT

Nessun file selezionato

MQTT CLIENT CERTIFICATE

NOT PRESENT

Nessun file selezionato

MQTT CLIENT CERTIFICATE PRIVATE KEY :

NOT PRESENT

Nessun file selezionato

10.6.1.1. ESEMPIO

Supponiamo di voler inviare i log di due TAG: tag1 e tag2, con la seguente configurazione:

Client ID = "Test"

Publish Topic for Logs = seneca/%c/data

Publish Payload for Logs = {"type": "data", "message": {"device": "%jc", "date": "%t", "signals": [%b]}}

Publish Bulk Format = {"name": "%jn", "value": "%v", "valid" : %i}

Si otterrà: sul topic "Seneca/Test/data" il seguente Payload:

```
{
  "type": "data",
  "message": {
    "device": "Test",
    "date": "1750942723",
    "signals": [
      {
        "name": "tag1",
        "value": 1234,
        "valid": 1
      },
      {
        "name": "tag2",
        "value": 5678,
        "valid": 1
      }
    ]
  }
}
```

10.6.2.DIREL ADM4.0

I parametri per il cloud di Direl (<https://www.direl.it/>) sono i seguenti:

Campo	Significato
Enable	Abilita o no la connessione con il cloud Direl ADM4.0
Username for Commands	È la username per l'accesso in scrittura dal cloud verso il dispositivo
Password for Commands	È la password per l'accesso in scrittura dal cloud verso il dispositivo

10.6.1. CLOUDBOX 2 SENECA

I parametri per la connessione sono:

Campo	Significato
Enable	Abilita o no la connessione con il cloud Onboard
Username	È la username per l'accesso al cloud
Password	È la password per l'accesso al cloud

CLOUD SUBSCRIBE TOPIC FOR COMMANDS

Per scrivere un tag tramite MQTT, il dispositivo deve ricevere una PUBLISH dal cloud stesso con il formato indicato in questo campo.

MQTT CA CERTIFICATE FILE (.pem)

File che rappresenta il Root CA Certificate in formato .pem

MQTT/HTTP SERVER CERTIFICATE FILE (.pem)

File che rappresenta il Client Certificate in formato .pem

MQTT CLIENT PRIVATE KEY FILE (.pem)

File che rappresenta la chiave del Client in formato .pem

10.7. PAGINA "FIRMWARE UPDATE"

Permette di aggiornare il firmware del dispositivo.



ATTENZIONE!

**PER NON DANNEGGIARE IL DISPOSITIVO NON TOGLIERE ALIMENTAZIONE DURANTE
L'OPERAZIONE DI AGGIORNAMENTO DEL FIRMWARE.**



ATTENZIONE!

**L'AGGIORNAMENTO FIRMWARE CANCELLERÀ I DATI ACQUISITI DAL DATALOGGER, SALVARE
I DATI PRIMA DI EFFETTUARE L'OPERAZIONE DI AGGIORNAMENTO.**

10.8. PAGINA "UTC TIME SETUP"

Permette di impostare la data ora.



ATTENZIONE!

AD OGNI SPEGNIMENTO IL DISPOSITIVO DEVE POTER RECUPERARE LA DATA / ORA DA UN SERVER NTP ALTRIMENTI QUESTA SARA' PRESA DALL'ULTIMO LOG ACQUISITO

10.9. PAGINA "CERTIFICATE/DATABASE UPDATE"

In questa pagina è possibile caricare nel dispositivo i certificati X.509 per il webserver (se attivata la modalità https) e aggiornare il database dei dispositivi Seneca.

10.10. PAGINA "SERIAL TRAFFIC MONITOR"

La pagina Serial Traffic Monitor del webserver mostra i pacchetti seriali che il gateway sta ricevendo e trasmettendo per il debug della linea:

START/STOP TRAFFIC MONITOR ENABLED		
116	RECEIVE	01 03 00 00 00 01 84 0a
14	SEND	01 03 02 12 34 b5 33
114	RECEIVE	01 03 00 00 00 01 84 0a
16	SEND	01 03 02 12 34 b5 33
112	RECEIVE	01 03 00 00 00 01 84 0a
18	SEND	01 03 02 12 34 b5 33
109	RECEIVE	01 03 00 00 00 01 84 0a
11	SEND	01 03 02 12 34 b5 33
117	RECEIVE	01 03 00 00 00 01 84 0a
13	SEND	01 03 02 12 34 b5 33
115	RECEIVE	01 03 00 00 00 01 84 0a
15	SEND	01 03 02 12 34 b5 33
113	RECEIVE	01 03 00 00 00 01 84 0a
17	SEND	01 03 02 12 34 b5 33
110	RECEIVE	01 03 00 00 00 01 84 0a
20	SEND	01 03 02 12 34 b5 33
108	RECEIVE	01 03 00 00 00 01 84 0a
12	SEND	01 03 02 12 34 b5 33
116	RECEIVE	01 03 00 00 00 01 84 0a
14	SEND	01 03 02 12 34 b5 33
114	RECEIVE	01 03 00 00 00 01 84 0a
16	SEND	01 03 02 12 34 b5 33
111	RECEIVE	01 03 00 00 00 01 84 0a
19	SEND	01 03 02 12 34 b5 33
109	RECEIVE	01 03 00 00 00 01 84 0a

La prima colonna è il ritardo in millisecondi dall'ultimo pacchetto, la seconda colonna è il verso del pacchetto (ricevuto da o trasmesso a), l'ultima colonna è il contenuto del pacchetto in formato esadecimale. Viene visualizzato solo il flusso ModBUS seriale.

Il Traffic Monitor mostra tutti i pacchetti ricevuti dalla linea seriale, ad esempio se si tratta di uno slave seriale con una risposta errata del Modbus:

3870	SEND	01 03 00 00 00 0a c5 cd
130	RECEIVE	fe fe ff df bc cf bc 9e cf f0 3e 7c bc bc ce 3e cf ce 3c df 8e 8f cf ee ce ce ce bc ce c7 c7 87 be 9e bc bc 9f 3e 3c bc bc 3e bc 8e c7 3c cf 9f be ef bc 01 03 14 42 00 08 7c 00 0b 00 01 00 01 00 00 04 00 c3 48 00 00 44 22 b8 5d

Il Traffic Monitor visualizzerà anche i pacchetti difettosi in giallo (per esempio un master seriale con baud rate errato):

18	SEND	01 03 02 12 34 b5 33
988	RECEIVE	01 03 00 00 00 01 84 0a
12	SEND	01 03 02 12 34 b5 33
20990	INVALID RECEIVE	20 e0 20 e0 20 e0 20 e0
14994	INVALID RECEIVE	20 e0 20 e0 20 e0 20 e0
14100	INVALID RECEIVE	20 e0 20 e0 20 e0 20 e0
14897	INVALID RECEIVE	20 e0 20 e0 20 e0 20 e0

11. SCRITTURE DA CLOUD VERSO IL DISPOSITIVO

11.1. SCRIVERE TAG DAL CLOUD AL DISPOSITIVO VIA MQTT (CLOUD GENERIC)

Tramite la configurazione MQTT è possibile scrivere i TAG in due modalità fondamentali:

Nella prima nel payload non compare il nome del tag, nella seconda il nome del tag è esplicitato nel payload.

Per scrivere un tag senza esplicitare il suo nome nel payload bisogna eseguire una sottoscrizione al topic:

```
seneca/<ClientID>/info/#
```

dove <ClientID> è appunto il client id configurato

Verrà poi ricevuta dal dispositivo una publish con topic:

```
seneca/<ClientID>/info/<nome tag>
```

e payload:

```
{"val": <valore tag>}
```

oppure

```
{"value": <valore tag>}
```

Ad esempio:

facendo la publish al topic:

```
seneca/<ClientID>/info/Pippo
```

con payload:

```
{"val": 1234}
```

Si scrive il valore decimale 1234 nel Tag di nome "Pippo" (attenzione al case sensitive).

Per scrivere un tag esplicitando il nome nel payload bisogna eseguire una sottoscrizione al topic:

```
seneca/<ClientID> /info
```

Verrà poi ricevuta dal dispositivo una publish con topic:

seneca/<ClientID> /info

e payload:

```
{"tags": [{"<nome tag>": <valore tag>}]}
```

Ad esempio:

```
{"tags": [{"Pippo_fp": 123.46}]}
```

Scrivo nel tag "Pippo_fp" il valore floating point 123,46

Oppure è possibile invece che definire il nome del tag utilizzare l'ID (numero che compare nella colonna Vid dei Tag (vedi pagina web di configurazione Tag setup):

```
{"tags_id": [{"<(vid+1)>": <valore tag>}]}
```

Ad esempio:

```
{"tags_id": [{"25": 789}]}
```

Scrivo nel tag con vid = 24 il valore intero decimale 789

È anche possibile scrivere più di un tag contemporaneamente con le sintassi:

```
{"tags": [{"<nome tag1>": <valore tag1>}, {"<nome tag2>": <valore tag2>}, .... ] }
```

Oppure:

```
{"tags_id": [{"<(vid tag1)+1>": <valore tag1>}, {"<(vid tag2)+1>": <valore tag2>}, .... ] }
```

Ad esempio:

```
{"tags": [{"Pippo": 1234}, {"Pippo_fp": 123.46}]}
```

```
{"tags_id": [{"25": 1234}, {"26": 123.46}]}
```

Scrivono entrambi i tag contemporaneamente.

12. MODIFICA DEL TEMPO DI CAMPIONAMENTO DA MQTT

È possibile modificare da MQTT il tempo di invio dei vari gruppi di invio tramite il seguente comando in formato json:

```
{"cmd_exec": [{"<nome gruppo>": <valore intero>}]}
```

dove "<nome gruppo>" può essere:

"grp_none" : per il gruppo NONE

"grp_input" : per il gruppo INPUT

"grp_alarm" : per il gruppo ALARM

"grp_status" : per il gruppo STATUS

"grp_cmd" : per il gruppo COMMAND

"grp_temp" : per il gruppo TEMPERATURE

"grp_hr" : per il gruppo HUMIDITY

"grp_weight" : per il gruppo WEIGHT

"grp_v" : per il gruppo VOLTAGE

"grp_i" : per il gruppo CURRENT

"grp_other" : per il gruppo OTHER

mentre <valore intero> è un numero intero positivo maggiore o uguale a 1

e inferiore a 65536.

Il comando json va scritto nel topic impostato sulla configurazione aggiungendo la stringa "/info".

Il comando è valido solo se si seleziona il tipo di CLOUD Generic.

13. RIPRISTINO DEL DISPOSITIVO ALLA CONFIGURAZIONE DI FABBRICA

La configurazione di fabbrica riporta tutti i parametri a default.

Per ripristinare il dispositivo alla configurazione di fabbrica è necessario seguire la seguente procedura:

Z-KEY-C / Z-KEY-2ETH-C:

- 1) Togliere alimentazione al dispositivo
- 2) Portare i dip switch 1 e 2 ad ON
- 3) Alimentare il dispositivo per almeno 10 secondi
- 4) Togliere alimentazione al dispositivo
- 5) Portare i dip switch 1 e 2 ad OFF
- 6) Al prossimo riavvio il dispositivo avrà caricata la configurazione di fabbrica

R-KEY-LT-C:

- 1) Togliere alimentazione al dispositivo
- 2) Portare i dip switch 1 e 2 di SW2 ad ON
- 3) Alimentare il dispositivo per almeno 10 secondi
- 4) Togliere alimentazione al dispositivo
- 5) Portare i 2 dip switch di SW2 ad OFF
- 6) Al prossimo riavvio il dispositivo avrà caricata la configurazione di fabbrica

14. TEMPLATE EXCEL

È disponibile un template Microsoft Excel™ per creare un file .bin da importare nel gateway o vice versa. Il modello può essere liberamente scaricato dal sito web Seneca.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	MODBUS TCP/IP			SERIAL MODBUS RTU					Export CGI file... Import CGI file... 				
	TAG NR	GATEWAY TAG NAME	GATEWAY MODBUS TCP/IP REGISTER ADDRESS	TARGET MODBUS RTU REGISTER TYPE	TARGET MODBUS RTU DATA TYPE	TARGET CONNECTED TO SERIAL PORT NR	TARGET MODBUS RTU START REGISTER	TARGET MODBUS RTU SLAVE ADDRESS	SENECA Z-KEY TAGS TEMPLATE FOR GATEWAY MODE. E				
2													
3	1	TAG1	1	HOLDING REGISTER	UINT16	#1	3	2					
4	2	TAG2	2	HOLDING REGISTER	UINT16	#1	4	2					
5	3	TAG3	3	HOLDING REGISTER	UINT16	#1	5	2					
6	4	TAG4	5	HOLDING REGISTER	UINT16	#1	6	2					
7	5	TAG5	7	HOLDING REGISTER	UINT16	#1	7	2					
8	6	TAG6	8	HOLDING REGISTER	UINT16	#1	8	2					
9	7	TAG7	9	HOLDING REGISTER	UINT16	#1	9	2					
10	8	TAG8	10	HOLDING REGISTER	UINT16	#1	10	2					
11	9	TAG9	1	COIL	BIT	#1	1	3					
12	10	TAG10	2	COIL	BIT	#1	2	3					
13	11	TAG11	3	COIL	BIT	#1	3	3					
14	12	TAG12	4	COIL	BIT	#1	4	3					
15	13	TAG13	5	COIL	BIT	#1	5	3					
16	14	TAG14	6	COIL	BIT	#1	6	3					
17	15	TAG15	7	COIL	BIT	#1	7	3					
18	16	TAG16	8	COIL	BIT	#1	8	3					
19	17	TAG17	14	HOLDING REGISTER	INT16	#1	13	4					
20	18	TAG18	15	HOLDING REGISTER	INT16	#1	14	4					
21	19	TAG19	16	HOLDING REGISTER	INT16	#1	15	4					
22	20	TAG20	17	HOLDING REGISTER	INT16	#1	16	4					
23	21	TAG21	1	DISCRETE INPUT	BIT	#1	1	5					
24	22	TAG22	2	DISCRETE INPUT	BIT	#1	2	5					
25	23	TAG23	3	DISCRETE INPUT	BIT	#1	3	5					

15. INSTALLAZIONE DI PIÙ DISPOSITIVI IN UNA RETE UTILIZZANDO IL "DHCP FAIL ADDRESS".

Quando Il Gateway è configurato con il DHCP attivato ma non riceve la configurazione del DHCP server entro 2 minuti allora assume un indirizzo di fail.

Questo indirizzo di fail è 169.254.x.y dove x.y sono gli ultimi due valori dall'indirizzo MAC.

In questo modo se si forza a DHCP tutti i dispositivi si può installare in rete anche se non c'è un server DHCP attivo.

Quando l'indirizzo di fail è stato attivato (il led relativo smette di lampeggiare), è possibile lanciare il software "Seneca Discovery Device" e forzare l'indirizzo IP che si preferisce a tutti i dispositivi.

16. IL CAVO RS232 DB9

Il CAVO DB9 CAVO RS232 può essere ottenuto da Seneca (può essere acquistato anche dal sito web di e-commerce www.seneca.it) per il collegamento con un dispositivo DB9 RS232.

17. *PROTOCOLLI MODBUS DI COMUNICAZIONE SUPPORTATI*

I protocolli di comunicazione Modbus supportati sono:

- Modbus RTU/ASCII master/slave (dalle porte seriali #1 e #2)
- Modbus TCP-IP Client (dalla porta Ethernet), massimo 10 Server Modbus TCP-IP remoti

Per ulteriori informazioni su questi protocolli, consultare il sito Web:

<http://www.modbus.org/specs.php>.

17.1. *CODICI FUNZIONE MODBUS SUPPORTATI*

Sono supportate le seguenti funzioni Modbus:

- Read Coils (function 1)
- Read Discrete Inputs (function 2)
- Read Holding Registers (function 3)
- Read Input Registers (function 4)
- Write Single Coil (function 5)
- Write Single Register (function 6)
- Write multiple Coils (function 15)
- Write Multiple Registers (function 16)



ATTENZIONE!

Tutte le variabili a 32 bit sono contenute in 2 registri Modbus consecutivi

Tutte le variabili a 64 bit sono contenute in 4 registri Modbus consecutivi

18. INFORMAZIONI SUI REGISTRI MODBUS

Nel seguente capitolo vengono usate le seguenti abbreviazioni:

MS	Most Significant
LS	Least Significant
MSBIT	Most Significant Bit
LSBIT	Least Significant Bit
MMSW	"Most" Most Significant Word (16bit)
MSW	Most Significant Word (16bit)
LSW	Least Significant Word (16bit)
LLSW	"Least" Least Significant Word (16bit)
RO	Read Only
UNSIGNED 16 BIT	Registro intero senza segno che può assumere valori da 0 a 65535
SIGNED 16 BIT	Registro intero con segno che può assumere valori da -32768 a +32767
UNSIGNED 32 BIT	Registro intero senza segno che può assumere valori da 0 a 4294967296
SIGNED 32 BIT	Registro intero con segno che può assumere valori da -2147483648 a 2147483647
UNSIGNED 64 BIT	Registro intero senza segno che può assumere valori da 0 a 18.446.744.073.709.551.615
SIGNED 64 BIT	Registro intero con segno che può assumere valori da -2 ⁶³ a 2 ⁶³ -1
FLOAT 32 BIT	Registro a virgola mobile a 32 bit, a precisione singola (IEEE 754) https://en.wikipedia.org/wiki/IEEE_754
BIT	Registro booleano, che può assumere i valori 0 (false) o 1 (true)

18.1. NUMERAZIONE DEGLI INDIRIZZI MODBUS "0 BASED" O "1 BASED"

I registri Holding Register secondo lo standard ModBUS sono indirizzabili da 0 a 65535, esistono 2 diverse convenzioni per la numerazione degli indirizzi: la "0 BASED" e la "1 BASED".

Per maggiore chiarezza Seneca riporta le proprie tabelle dei registri in entrambe le convenzioni.



ATTENZIONE!

**LEGGERE ATTENTAMENTE LA DOCUMENTAZIONE DEL DISPOSITIVO MASTER MODBUS
AL FINE DI CAPIRE QUALE DELLE DUE CONVENZIONI IL COSTRUTTORE HA DECISO DI
UTILIZZARE.**

SENECA, PER I SUOI PRODOTTI, UTILIZZA LA CONVENZIONE "1 BASED"

18.2. NUMERAZIONE DEGLI INDIRIZZI MODBUS CON CONVENZIONE "0 BASED"

La numerazione è del tipo:

INDIRIZZO MODBUS HOLDING REGISTER (OFFSET)	SIGNIFICATO
0	PRIMO REGISTRO
1	SECONDO REGISTRO
2	TERZO REGISTRO
3	QUARTO REGISTRO
4	QUINTO REGISTRO

Per cui il primo registro si trova all'indirizzo 0.

Nelle tabelle che seguono questa convenzione è indicata con **"OFFSET INDIRIZZO"**.

18.3. NUMERAZIONE DEGLI INDIRIZZI MODBUS CON CONVENZIONE "1 BASED" (STANDARD)

La numerazione è quella stabilita dal consorzio Modbus ed è del tipo:

INDIRIZZO MODBUS HOLDING REGISTER 4x	SIGNIFICATO
40001	PRIMO REGISTRO
40002	SECONDO REGISTRO
40003	TERZO REGISTRO
40004	QUARTO REGISTRO
40005	QUINTO REGISTRO

Questa convenzione può essere indicata con **"INDIRIZZO 4x"** poiché viene aggiunto un 40000 all'indirizzo in modo che il primo registro ModBUS sia 40001.

È anche possibile una ulteriore convenzione dove viene omissso il numero 4 davanti all'indirizzo del registro:

INDIRIZZO MODBUS HOLDING SENZA 4x	SIGNIFICATO
1	PRIMO REGISTRO
2	SECONDO REGISTRO
3	TERZO REGISTRO
4	QUARTO REGISTRO
5	QUINTO REGISTRO

18.4. CONVENZIONE DEI BIT ALL'INTERNO DI UN REGISTRO MODBUS HOLDING REGISTER

Un registro ModBUS Holding Register è composto da 16 bit con la seguente convenzione:

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Ad esempio, se il valore del registro in decimale è

12300

il valore 12300 in esadecimale vale:

0x300C

l'esadecimale 0x300C in valore binario vale:

11 0000 0000 1100

Quindi, usando la convenzione di cui sopra otteniamo:

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	1	0	0	0	0	0	0	0	0	1	1	0	0

18.5. CONVENZIONE DEI BYTE MSB E LSB ALL'INTERNO DI UN REGISTRO MODBUS HOLDING REGISTER

Un registro ModBUS Holding Register è composto da 16 bit con la seguente convenzione:

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Si definisce Byte LSB (Least Significant Byte) gli 8 bit che vanno da Bit 0 a Bit 7 compresi, si definisce Byte MSB (Most Significant Byte) gli 8 bit che vanno da Bit 8 a Bit 15 compresi:

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BYTE MSB								BYTE LSB							

18.6. *RAPPRESENTAZIONE DI UN VALORE A 32 BIT IN DUE REGISTRI MODBUS HOLDING REGISTER CONSECUTIVI*

La rappresentazione di un valore a 32 bit nei registri Holding Register in ModBUS è fatta utilizzando 2 registri consecutivi Holding Register (un registro Holding Register è da 16 bit). Per ottenere il valore a 32 bit è necessario leggere quindi due registri consecutivi:

Ad esempio se il registro 40064 contiene i 16 bit più significativi (MSW) mentre il registro 40065 i 16 bit meno significativi (LSW) il valore a 32 bit si ottiene componendo i 2 registri:

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
40064 MOST SIGNIFICANT WORD															

BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT	BIT
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
40065 LEAST SIGNIFICANT WORD															

$$Value_{32bit} = Register_{LSW} + (Register_{MSW} * 65536)$$

Nei registri di lettura è possibile scambiare il word più significativo con quello meno significativo quindi è possibile ottenere il 40064 come LSW e il 40065 come MSW.

