



Indice

Introduzione	4
1 Accesso all'XPANGO Driver Editor	5
2 I Template Modbus e Siemens S7	6
2.1 Il Template MODBUS.....	6
2.1.1 General.....	6
2.1.2 Variables.....	8
2.1.2.1 Variable name	8
2.1.2.2 Section name.....	10
2.1.2.3 Area	11
2.1.2.4 Address.....	12
2.1.2.5 Bit of word.....	13
2.1.2.6 Data Type	14
2.1.2.7 Signed	14
2.1.2.8 Access Type	15
2.1.2.9 Measurement Unit	15
2.1.2.10 Scale.....	15
2.1.2.11 Offset Minimum e Maximum.....	16
2.1.2.12 Alarmed	17
2.1.2.13 Decode	17
2.1.2.14 Totalizer	17
2.1.2.15 Preselected.....	18
2.1.3 Digital Alarms	18
2.1.4 Analog Alarms	20
2.1.5 Virtual Variables	21
2.2 Il Template Modbus: esempio di compilazione.....	22
2.3 Il Template Siemens S7.....	26
2.3.1 General.....	28
2.3.2 Variables.....	29
2.3.2.1 Area	29
2.3.2.2 Data Block Number.....	30
2.3.2.3 Data Type	31
2.4 Il Template Siemens S7: esempio di compilazione	31
3 L'XPANGO Driver Editor.....	36
4 I Indicazioni e suggerimenti.....	38
5 Contatti.....	39

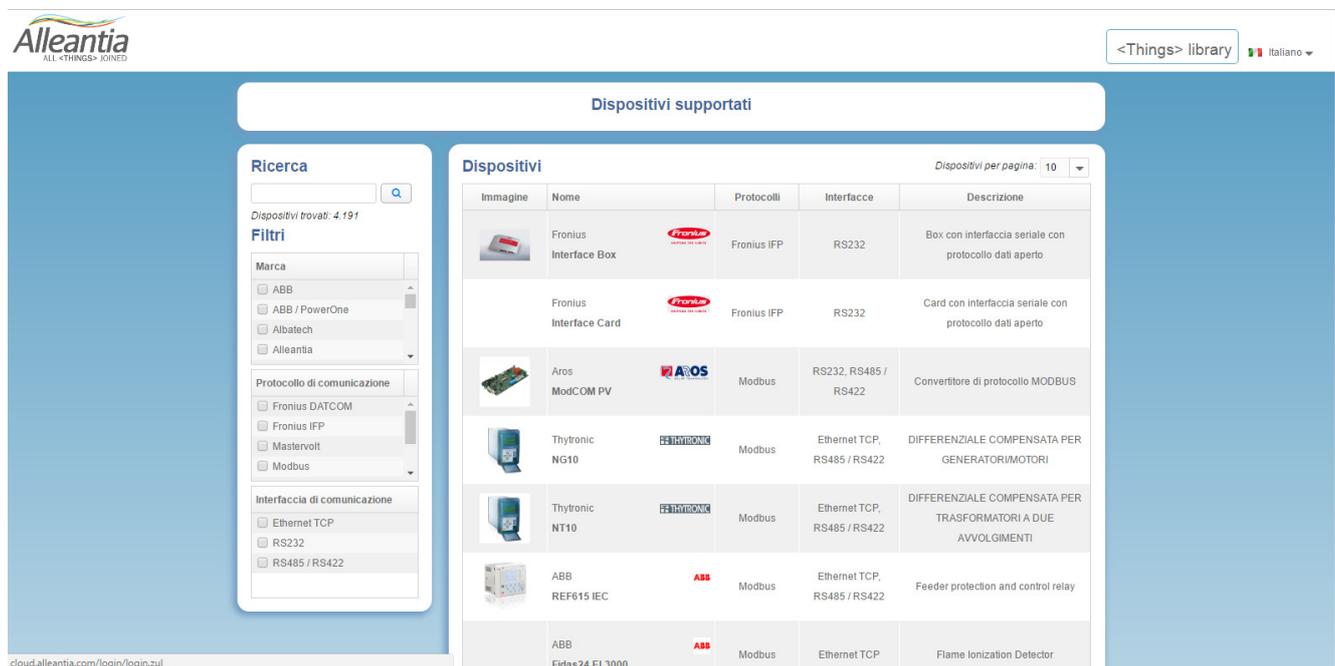
Introduzione

La piattaforma software di Alleantia è l'unica soluzione Industrial Internet of Things che permette la comunicazione in pochi secondi con qualsiasi dispositivo industriale per acquisizione dati e invio parametri, e quindi utilizzare le informazioni in diversi contesti e applicazioni in modo trasparente da protocolli e settaggi dei dispositivi connessi.

Per questo, si basa sul concetto di 'driver', dove per ogni tipologia di dispositivo viene creato un file XPANGO sulla base delle specifiche del dispositivo stesso. I driver XPANGO descrivono la semantica delle informazioni trasmesse e ricevibili dal dispositivo, e sono utilizzati dai sistemi di Alleantia IoT SCADA Server e IoT Gateway Server per realizzare la comunicazione, tramite linea seriale RS232 – 422 – 485 o Ethernet, con tutti i dispositivi che supportano i largamente adottati protocolli di comunicazione Modbus, Siemens S7, Focas e altri numerosi protocolli proprietari.

All'acquisto, numerosi si driver XPANGO sono già presenti nei sistemi ISS e IGS e vengono utilizzati per la configurazione dei dispositivi di un qualsiasi impianto gestito dal nostro dispositivo. (Vedi par. 5.2.2.1 Manuale d'installazione e Uso IOT SCADA Server). Driver aggiuntivi sono poi messi a disposizione degli utilizzatori dei sistemi Alleantia nella Library of Things (<http://cloud.alleantia.com/info/products.zul>), da cui è possibile scaricare aggiornamenti disponibili periodicamente. Inoltre, è possibile per ogni utilizzatore creare i propri drivers; ciò è particolarmente necessario per dispositivi programmabili quali i Programmable Logic Controllers (PLC). Per questo scopo è stato creato l'XPANGO driver editor, di cui al presente manuale d'uso.

Il manuale è completato da tutorial online disponibile su Vimeo all'indirizzo <https://vimeo.com/alleantia>.



The screenshot displays the 'Dispositivi supportati' (Supported Devices) page on the Alleantia cloud platform. The interface includes a search bar, a filter sidebar, and a main table of devices. The filter sidebar on the left allows users to search and filter devices by brand (ABB, Fronius, Aros, Thytronic), communication protocol (Fronius DATCOM, Fronius IFP, Mastervolt, Modbus), and communication interface (Ethernet TCP, RS232, RS485 / RS422). The main table lists the following devices:

Immagine	Nome	Protocolli	Interfacce	Descrizione
	Fronius Interface Box	Fronius IFP	RS232	Box con interfaccia seriale con protocollo dati aperto
	Fronius Interface Card	Fronius IFP	RS232	Card con interfaccia seriale con protocollo dati aperto
	Aros ModCOM PV	Modbus	RS232, RS485 / RS422	Convertitore di protocollo MODBUS
	Thytronic NG10	Modbus	Ethernet TCP, RS485 / RS422	DIFFERENZIALE COMPENSATA PER GENERATORI/MOTORI
	Thytronic NT10	Modbus	Ethernet TCP, RS485 / RS422	DIFFERENZIALE COMPENSATA PER TRASFORMATORI A DUE AVVOLGIMENTI
	ABB REF615 IEC	Modbus	Ethernet TCP, RS485 / RS422	Feeder protection and control relay
	ABB Fidas24 EL3000	Modbus	Ethernet TCP	Flame Ionization Detector

Figura 1 – Library of Things di Alleantia disponibile all'indirizzo <http://cloud.alleantia.com/info/products.zul>

1 Accesso all'XPANGO Driver Editor

Per avere accesso all'XPANGO Driver editor bisogna innanzitutto collegarsi al sito <http://cloud.alleantia.com> ed effettuare il login inserendo email e password negli appositi campi. Per il primo accesso è necessario registrarsi premendo sull'apposito link. È possibile eseguire l'iscrizione collegandosi attraverso i più comuni social network (Facebook, LinkedIn, Twitter) oppure inserendo i propri dati premendo sul link *i tuoi dati*. Nel secondo caso, bisogna compilare il form inserendo tutti i campi obbligatori, spuntare le caselle relative all'accettazione dei termini di servizio e della privacy policy e inserire il codice di verifica. Infine, facendo click sul tasto "Iscriviti" riceverete, all'indirizzo mail inserito durante l'iscrizione, un messaggio contenente un link per la conferma dell'indirizzo mail.

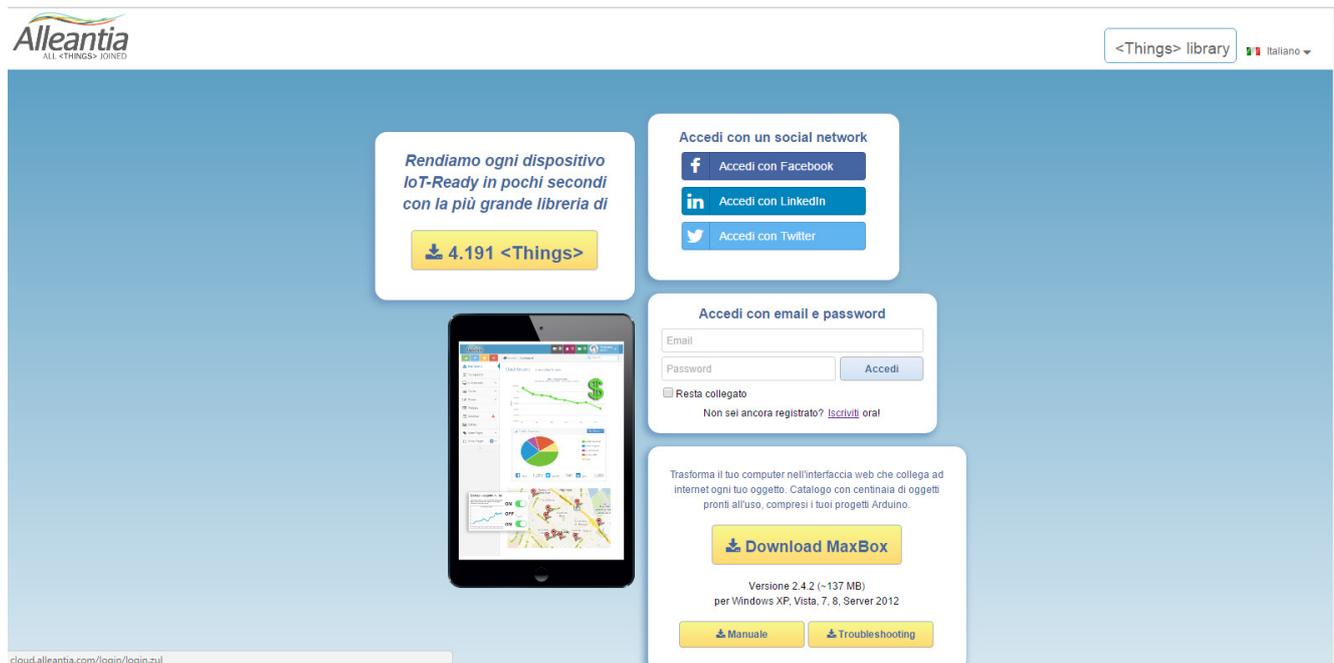


Figura 2 - Pagina di accesso al cloud di Alleantia

Una volta completata la registrazione e inseriti i dati di accesso, è possibile creare il driver di un qualsiasi dispositivo accedendo all'XPANGO Driver Editor tramite il tasto "Crea <Things>" collocato in alto a destra. Si aprirà quindi la pagina <http://cloud.alleantia.com/xmod/convert.zul>. Da questa è possibile scaricare i template Modbus e Siemens S7 in formato Excel.

2 I Template Modbus e Siemens S7

I template sono fogli di calcolo preimpostati che hanno il compito di facilitare il lavoro dell'utente che intende creare un driver XPANGO a partire dalle informazioni contenute all'interno dei registri delle mappe Modbus. Quest'ultime vengono, di solito, fornite dalle stesse case costruttrici o dal programmatore del PLC.

La struttura dei template è semplice e intuitiva e pertanto adatta ad utenti di qualsiasi esperienza e formazione. Una volta comprese le peculiarità di ciascuna delle tipologie di dati presenti all'interno delle Modbus table non resterà che copiare ciascun indirizzo di comunicazione all'interno del template.

Dalla pagina di Editor sarà possibile scaricare due tipologie di template: il template Modbus ed il template Siemens S7.

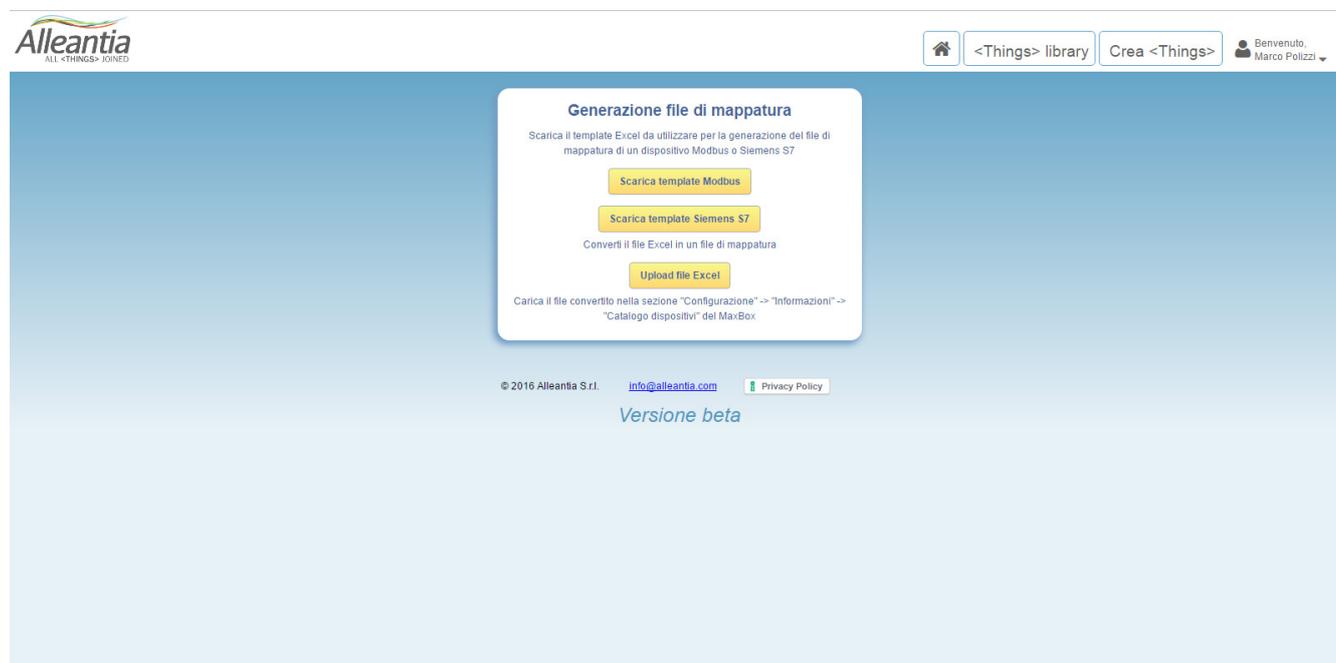


Figura 3 - XPANGO Driver Editor di Alleantia disponibile all'indirizzo <http://cloud.alleantia.com/xmod/convert.zul>

I due template differiscono principalmente per la diversa modalità con cui i costruttori forniscono gli indirizzi di comunicazione. Il protocollo di comunicazione Modbus oltre ad essere ormai uno standard de facto nella comunicazione industriale e il linguaggio di connessione più diffuso al mondo. Il protocollo Siemens S7 è stato sviluppato dall'omonima casa costruttrice appositamente per la comunicazione con i PLC prodotti dall'azienda.

2.1 Il Template MODBUS

Questo template deve essere usato per ottenere il driver XPANGO di un qualsiasi dispositivo indipendentemente dalla sua tipologia o dalla particolare costruttore.

2.1.1 General

Come prima cosa, dopo aver aperto il file, si può procedere alla compilazione del tab "General".

2 | Template Modbus e Siemens S7

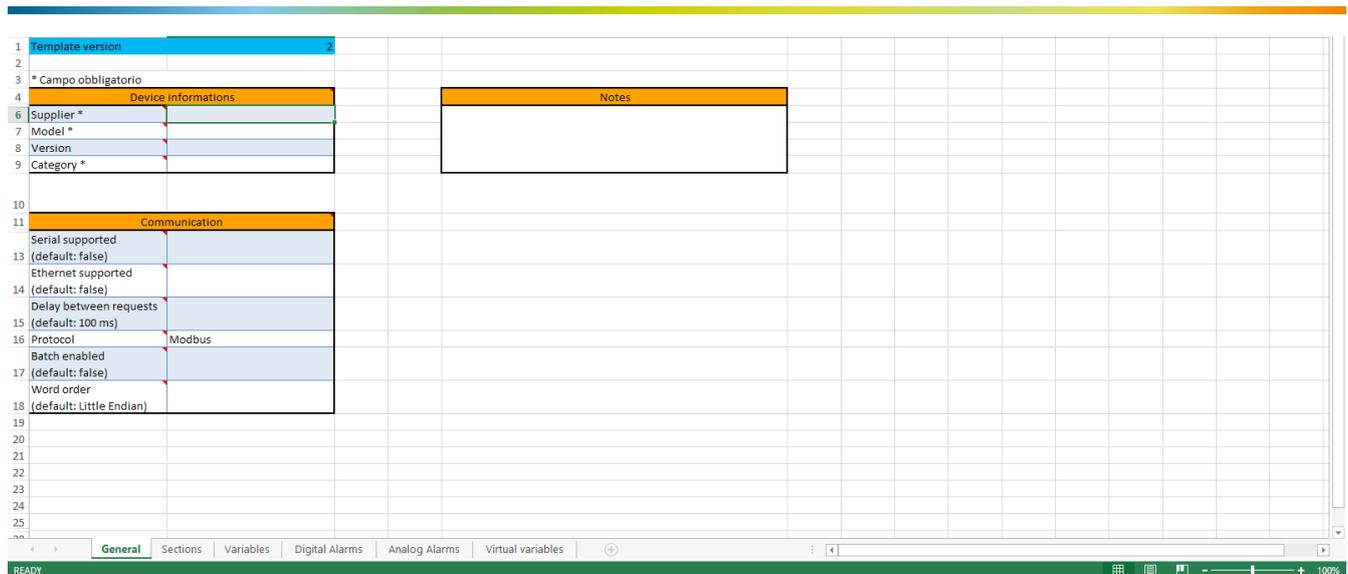


Figura 4 – Tab General del template Modbus

Il tab contiene 3 tabelle distinte: *Device informations*, *Communication* e *Notes*.

Nella tabella *Device Informations* andremo ad inserire le informazioni relative al dispositivo di cui vogliamo ottenere il driver. È essenziale compilare tutti i campi obbligatori contrassegnati dall'asterisco. Inseriamo il nome del costruttore, il modello del dispositivo e scegliamo una categoria tra quelle che appaiono cliccando sulla freccetta menù che viene visualizzata una volta selezionata la cella in questione (cella B9).

Nella tabella *Communication* inseriamo tutte le informazioni relative alle modalità di comunicazione del dispositivo. Nella prima riga specifichiamo se il dispositivo è provvisto o no di porta seriale RS485, RS232. In caso affermativo, cliccando sulla freccetta selezioneremo "TRUE" dal menu a tendina che si apre. Se il dispositivo non è provvisto di porta seriale selezioneremo "FALSE" oppure, più semplicemente, lasceremo il campo vuoto poiché FALSE è il valore di default.

Quanto detto prima vale anche per la seconda riga della tabella *Communication*. Qui bisogna specificare se il dispositivo è dotato di porta Ethernet. Come prima metteremo TRUE in caso affermativo o lasceremo vuoto il campo se il dispositivo non ha la porta Ethernet.

Nella terza riga possiamo impostare il tempo di ritardo tra una richiesta al dispositivo e la successiva. Lasciando il campo vuoto imposteremo tale tempo al valore di default paria a 100 ms.

Dalla penultima riga possiamo abilitare o disabilitare la funzione di *Batch data Transmission* ovvero la trasmissione di un blocco continuo contenente più di un dato. In tal modo è possibile inviare più indirizzi ed acquisirli (read) o modificarli (write) in una sola trasmissione. Tale procedura, detta spesso *Batch Optimization* ha lo scopo di aumentare la velocità di comunicazione rispetto al caso normale in cui inviamo un solo dato alla volta.

Infine, la tabella va completata specificando il *Word order*, parametro essenziale ai calcolatori per immagazzinare in memoria dati con dimensione superiore al byte (1 Byte= 8 bit). Detto questo specificare questo campo è importante qualora si trasmettano dati di tipo *double word* o *quad word* ovvero dati composti da 2 o 4 Word (1 Word= 2 byte=16 bit). Aprendo il menu a tendina possiamo selezionare o la codifica di tipo *Big Endian* oppure quella di tipo *Little Endian*. Scegliendo la codifica *Big Endian* la memorizzazione del dato comincerà

2 I Template Modbus e Siemens S7

dal byte “più significativo” e finirà con quello “meno significativo” mentre, nel caso di codifica *Little Endian* avverrà esattamente il contrario.

Ad esempio, volendo rappresentare il numero esadecimale 0x01234567 in formato Double Word (32 bit) avremo la seguente rappresentazione nei due formati (le due cifre in grassetto rappresentano il byte più significativo):

	Little endian				Big endian			
	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+
	0x67	0x45	0x23	0x01	0x01	0x23	0x45	0x67
	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+	+-----+
Indirizzo di memoria:	0	1	2	3	0	1	2	3

Come si può vedere dalla figura sopra, la codifica Little Endian memorizza il numero esadecimale inserendo le cifre più significative (0x01) sull'indirizzo di memoria più grande (3) mentre la codifica Big endian fa esattamente l'opposto; le cifre più significative saranno salvate all'indirizzo di memoria 0 mentre quelle via via meno significative (23, 45, 67) negli indirizzi di memoria successivi.

Tra i due sistemi di codifica il più utilizzato è il little endian. È bene accertarsi riguardo all'ordinamento effettivamente adottato. A tal proposito è consigliato rivolgersi ai manuali tecnici del dispositivo in questione. Informazioni circa la tipologia del dispositivo e le componenti che lo compongono possono essere inserite nella tabella *Notes*. La sua compilazione è del tutto facoltativa.

2.1.2 Variables

Una volta completata la tab General si può passare al riempimento dei campi contenuti all'interno della tab “Variables”. Durante la nostra comunicazione con il dispositivo possiamo “interrogarlo” sullo stato di tutte le variabili che andremo a registrare nella seguente tab. Ciascuna variabile è innanzitutto individuata da un codice identificativo univoco rappresentato dai numeri della colonna ID le cui celle non possono essere modificate in alcun modo.

2.1.2.1 Variable name

All'interno delle celle della colonna *Variable Name* verrà inserito il nome della variabile in questione. Innanzitutto, il nome deve identificare la variabile in maniera univoca. Infatti, nel caso in cui due variabili appartenenti alla stessa sezione abbiano identica *Variable Name*, la procedura di mappatura del driver XPANGO non andrà a buon fine.

2 I Template Modbus e Siemens S7

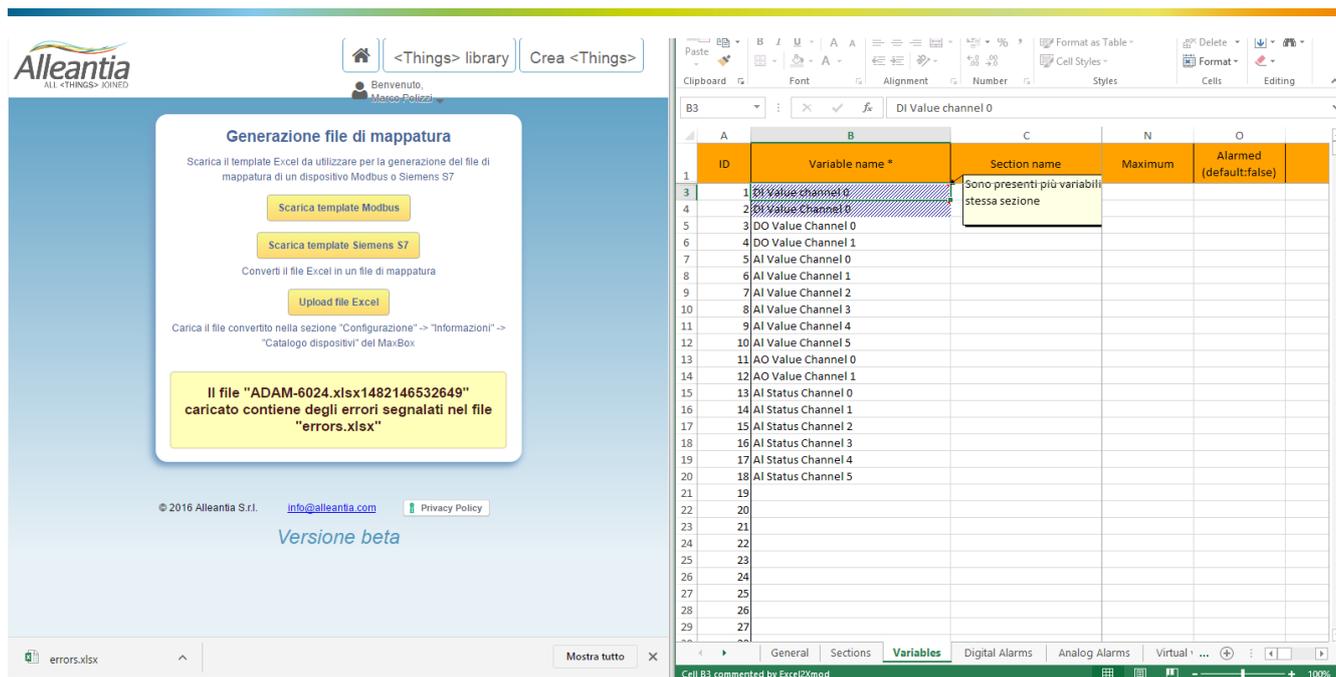


Figura 5 – Tab General del template Modbus: errore dovuto alla presenza di più variabili con lo stesso nome

Detto questo, se due variabili hanno lo stesso nome possiamo eliminare l'errore seguendo due strade. La prima è quella di differenziarle inserendo alla fine del loro nome due numeri (o lettere) diversi. La seconda è quella di inserire le due variabili con identica *Variable Name* in due differenti sezioni (Vedi 2.2.2.2).

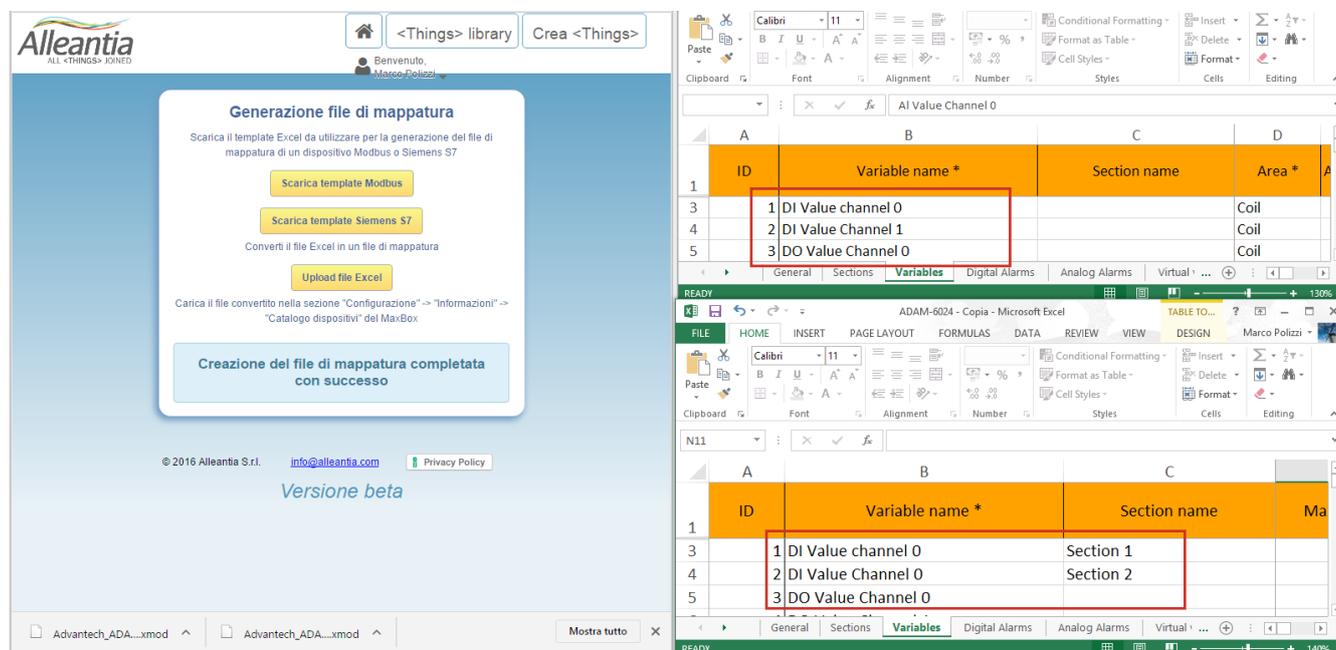


Figura 6 - Tab General del template Modbus: risoluzione dell'errore dovuto alla presenza di più variabili con lo stesso nome

Il nome della variabile deve inoltre identificarla con chiarezza poiché il campo inserito in variable name sarà poi visualizzato nell'interfaccia grafica dell'IoT-SCADA una volta caricato il driver XPANGO e configurato il dispositivo.

2 | Template Modbus e Siemens S7

2.1.2.2 Section name

È possibile raggruppare le variabili in più sezioni definendo per ciascuna di esse una *Section name*. Tale suddivisione sarà riportata nell'interfaccia grafica dell'IOT SCADA Server una volta caricato il driver e configurato il dispositivo.

Le sezioni in cui vogliamo raggruppare le diverse variabili non vanno inserite direttamente nella colonna *Section name*. Prima bisogna spostarsi nel tab *Sections* composto da due colonne: una contenente l'ID identificativo della sezione e l'altra contenente il nome della sezione stessa.

Una volta specificati i nomi delle sezioni di nostro interesse e ritornando al tab *Variables*, possiamo specificare la sezione entro cui inserire una data variabile selezionandola dal menu a tendina creatosi nelle celle della colonna *Section name* dopo aver creato le diverse sezioni. Il menu a tendina contiene le sezioni specificate nel tab *Sections* elencate in ordine crescente secondo l'ID di sezione.

La suddivisione delle variabili in più sezioni non è essenziale alla buona riuscita del processo di mappatura del file .xmod. Essa può risultare comoda all'utente che in tal modo può più facilmente gestire le numerose variabili di cui è costituita una mappa Modbus.

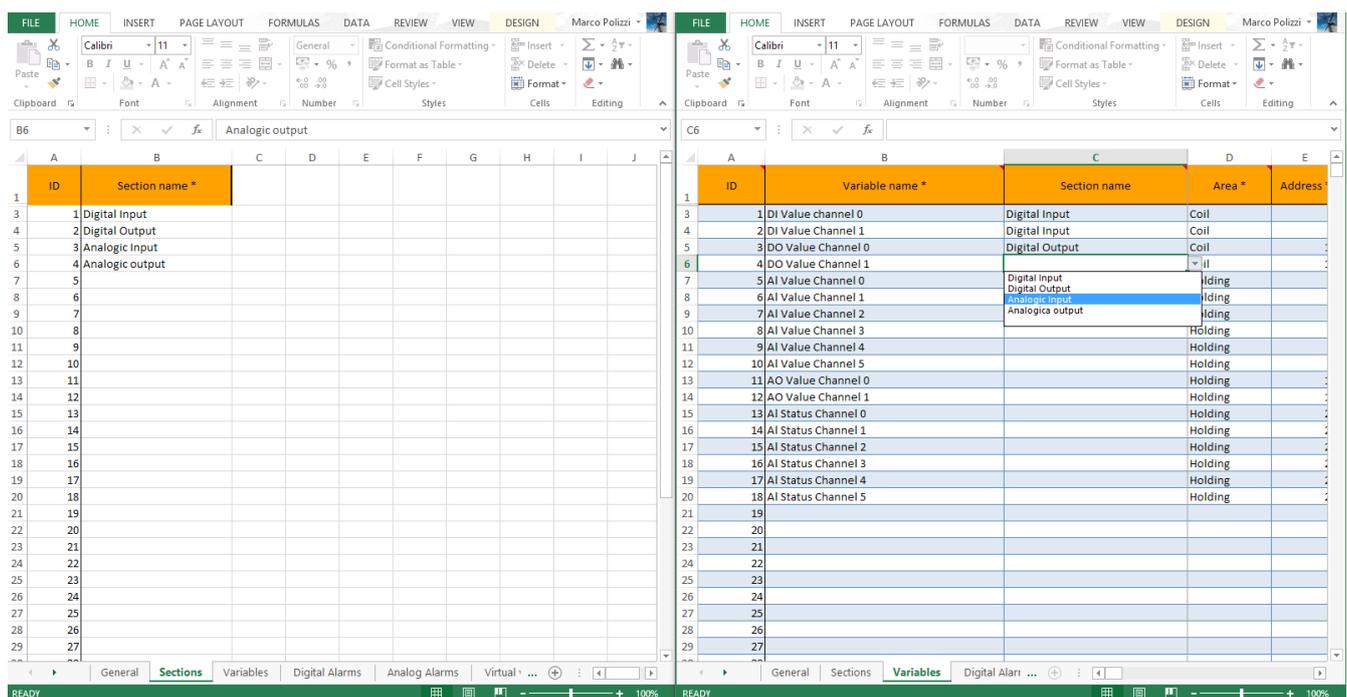


Figura 7 - Tab Sections del template Modbus.

2 I Template Modbus e Siemens S7

2.1.2.3 Area

All'interno della seguente colonna va inserita l'area di memoria, o registro, entro cui verrà inserita la variabile in oggetto. Il protocollo Modbus suddivide i dati oggetto della comunicazione in più registri. L'appartenenza di una variabile ad un dato registro dipenderà dalla tipologia di informazioni che la variabile trasporta con sé. Esistono 4 tipi di registri:

1. Coil (o Discrete Output);
2. Discrete Input;
3. Input Register;
4. Holding Register.

I **Coils** sono registri ad 1 bit, e vengono utilizzati per controllare uscite discrete e possono essere registri operanti sia in lettura che in scrittura.

I **Discrete Input** sono anch'essi registri a 1 bit; in essi vengono memorizzati solo dati in input e quindi sono dei registri di sola lettura. Essendo dei registri ad 1 bit, i registri coils e i registri discrete input contengono esclusivamente dati di tipo booleano.

Gli **Input Registers** e gli **Holding Registers** sono entrambi registri a 16 bit adatti quindi a contenere dati delle dimensioni di una word. I primi accolgono solo dati di input e quindi sono registri di tipo read. I secondi sono invece dei registri universali in quanto possono contenere dati di tipo input, output, dati di configurazione e tanto altro. Inoltre sono registri del tipo Read & Write.

Molti costruttori indicano la tipologia di registro cui un dato è destinato nelle tabelle Modbus da loro stessi fornite. Qualora questa informazione non sia disponibile possiamo capire in che area di memoria verranno salvati i dati della variabile in questione esaminando il suo indirizzo Modbus. Quasi sempre, i costruttori rispettano la convenzione Modicon elaborata agli inizi dagli inventori stessi del linguaggio Modbus. Tale convenzione fornisce un range di indirizzi Modbus per ogni tipologia di registro. Quindi per capire a quale tipo di registro è destinata la variabile basta capire in quale intervalli di indirizzi si inserisce il suo Modbus Address. Infine, è utile sottolineare che il campo Area va compilato per ogni variabile contenuta nel template altrimenti l'XPANGO drive editor non riuscirà ad eseguire la mappatura.

Modbus Data types e Address Space			
Registro	Accesso	Spazio Occupato	Indirizzi
<i>Coil</i>	Read & Write	1 bit	00001 - 09999
<i>Discrete Input</i>	Read Only	1 bit	10001 - 19999
<i>Input Register</i>	Read Only	16 bits	30001 - 39999
<i>Holding Register</i>	Read & Write	16 bits	40001 - 49999

Tabella 1 - Range degli indirizzi dei diversi Registri Modbus, modalità di accesso a ciascuno di essi, tipi di dati supportati e loro dimensione

2 I Template Modbus e Siemens S7

2.1.2.4 Address

All'interno della colonna *Address* va inserito l'indirizzo Modbus entro cui è inserito il valore della variabile che intendiamo leggere o modificare attraverso la comunicazione seriale con il dispositivo. Ciascuna variabile ha un indirizzo diverso dalle altre in maniera tale da poter essere sicuri di "chiedere" al dispositivo informazioni relative esclusivamente alla variabile in questione. Non vi è una modalità universalmente adottata per fornire gli indirizzi Modbus. Alcuni costruttori adottano la convenzione Modicon, altri forniscono l'indirizzo in valore Esadecimale altri adottano modi propri di fornirli fornendo comunque, nei manuali tecnici dei dispositivi, le informazioni necessarie ad interpretarli.

Il Modbus Address può assumere valori compresi tra 0 e 65535, quindi tra 0 e FFFF in notazione esadecimale. A tal proposito, è utile precisare che l'indirizzo va inserito all'interno del template esclusivamente in formato decimale quindi, qualora il costruttore ci indichi gli indirizzi in notazione esadecimale è necessario prima convertirli. La prima cifra dell'indirizzo indica la tipologia di registro cui il dato appartiene (0=Coil, 1= Discrete Input, 3=Input, 4=Holding, vedi par. 2.1.2.3).

L'indirizzo vero e proprio, da inserire all'interno della colonna, è rappresentato da tutte le cifre separate dalla prima da uno o più zeri. Ad esempio nella *Figura 9* è riportata la Mappa Modbus estratta dal manuale del dispositivo ADAM-6024 di Advantech, la variabile *DI Value Channel 0* ha indirizzo 00001; quindi si tratta di una variabile il cui valore appartiene al registro Coil (0X) e ha indirizzo 1. Mentre, la variabile *AI Value Channel 0* ha indirizzo 40001; quindi si tratta di una variabile il cui valore appartiene al registro Holding (4X) e ha indirizzo 1.

Quindi selezioneremo rispettivamente Coil e Holding nella colonna Area, inseriremo 1 per entrambe le variabili nella colonna *Address*.

2 | Template Modbus e Siemens S7

B.2.4 ADAM-6024

12-Ch Universal I/O Module

Address (0X):

Address (0X)	Channel	Description	Attribute
00001	0	DI Value	Read
00002	1		Read

00017	0	DO Value	R/W
00018	1		R/W

Address (4X):

Address (4X)	Channel	Description	Attribute
40001	0	AI Value	Read
40002	1		Read
40003	2		Read
40004	3		Read
40005	4		Read
40006	5		Read

Figura 8 - Mappa Modbus del Dispositivo ADAM-6024 di Advantech

2.1.2.5 Bit of word

Il campo *Bit of word* definisce il bit del registro entro cui sarà memorizzato un dato. Tale informazione è necessaria qualora si voglia ad esempio rappresentare dati di tipo booleano (1 bit) all'interno di registri di tipo Holding o Input Register, ovvero all'interno di registri a 16 bit. Infatti, non ha senso specificare il bit of word di un dato memorizzato all'interno di un registro coil o un registro Discrete Input (il bit of word sarà sempre 0 in questo caso, ovvero il primo e unico bit!).

In un registro holding, che ricordiamo può ospitare un'intera word, il primo bit è il numero 0 mentre l'ultimo è il numero 15. Nel caso in cui si vogliono inserire più variabili booleane all'interno di un registro a 16 bit, queste avranno uguale *Address* ma si distingueranno per il valore del loro *bit of word*.

A titolo di esempio si riporta l'estratto della mappa Modbus del dispositivo ADAM-6224 di Advantech. La segnalazione sullo stato degli input digitali (*DI Event Status*) occuperà un indirizzo di tipo Holding per ogni input digitale presente (Channel da 0 a 3). Dalla nota fornita dal costruttore si capisce come al primo bit della word (bit 0) compete la segnalazione di tipo booleano *Unreliable DI Value*, al secondo bit della word (bit 1) la segnalazione booleana *Safety Value Triggered* e così via. Un esempio su come sarà compilato il template è riportato in *Figura 10*.

2 | Template Modbus e Siemens S7

ID	Variable name *	Section name	Area *	Address *	Bit of word	Data type *	Signed (default: false)	Access type (default: Read)
1	DI Event Status Channel 0 - UART Timeout Triggered		Holding	111	0	Boolean		Read
2	DI Event Status Channel 0 - Safety Value Triggered		Holding	111	1	Boolean		Read
3	DI Event Status Channel 0 - Startup Value Triggered		Holding	111	2	Boolean		Read

Address (0X):			
Address (0X)	Channel	Description	Attribute
Note:			
1. The value definition of DI Event Status			
	Bit	Description	
	0	Unreliable DI value (UART Timeout)	
	1	Safety Value triggered	
	2	Startup Value triggered	
40111	0	DI Event Status	Read
40112	1		Read
40113	2		Read
40114	3		Read

Figura 9 – Template Modbus del dispositivo ADAM-6024 (in alto) ed estratto della mappa Modbus dalla quale sono stati ricavati i dati per la sua compilazione

2.1.2.6 Data Type

Nella colonna *Data Type* del Modbus template va specificato il tipo di dato con cui verrà espresso il valore della variabile in oggetto. La cella contiene un menu a tendina dalla quale sarà possibile selezionare tra i seguenti formati di rappresentazione: *Boolean*, *Integer*, *Double Integer*, *Quad Integer*, *Float*, *Double Float*, *BCD*, *Double BCD*. Si rinvia l'utente a documenti specialistici per maggiori approfondimenti riguardo alle caratteristiche di ognuno dei tipi di formato dati citati.

In registri di tipo *Coil* o *Discrete Input* capaci di accogliere un solo bit, il *Data Type* non può che essere booleano poiché un solo bit è sufficiente per tale rappresentazione. I registri di tipo *Holding* capaci di accogliere un'intera word invece, ospitano dati di tipo *Integer* o di tipo *float* poiché hanno una dimensione di 16 bit. Dati del tipo *Double Integer*, *Double Float* vengono disposti su 2 registri di memoria a 16 bit (*Holding* ad esempio) mentre dati del tipo *Quad Integer* su ben 4 registri (ognuno a 16 bit).

2.1.2.7 Signed

Il campo *Signed* può assumere soltanto due valori, TRUE o FALSE, impostabili dal menu a tendina di cui la cella è provvista. Se il numero che rappresenterà la variabile è un numero con segno il campo *Signed* assumerà il valore TRUE, in caso contrario possiamo selezionare il valore FALSE oppure lasciare vuoto il campo (Default: FALSE).

2 | Template Modbus e Siemens S7

Si noti che il campo va settato in FALSE per i valori inseriti nel registro Coil o Discrete Input. Le variabili contenute nei registri Coil non possono essere rappresentate con segno poiché la rappresentazione del segno stesso, che richiede un bit di memoria, occuperebbe da sola l'intero registro rendendo impossibile la memorizzazione della variabile. È necessario settare VERO per valori di tipo Float poiché si tratta di numeri comunque provvisti di segno.

2.1.2.8 Access Type

Nella colonna *Access Type* definiamo il tipo di accesso alla variabile. Vi sono tre tipi di accessi possibili: in scrittura (*Write*), in lettura (*Read*) oppure in entrambe (*Read/Write*). Gli accessi in semplice lettura o in semplice scrittura permettono rispettivamente, di leggere o modificare il valore di una variabile. La terza modalità di accesso permette sia di leggere il valore della grandezza sia di modificarlo.

L'*Access Type* da impostare è individuato, per prima cosa, dal registro in cui il dato è inserito (Vedi par. 2.1.2.3). Se il dato è inserito in un registro di tipo R/W spetterà a noi decidere se permettere o meno all'utente di leggere o modificare una variabile, o ancora, di poter compiere entrambe le azioni.

L'attributo da dare alla variabile è di solito indicato nelle mappe Modbus dei costruttori.

2.1.2.9 Measurement Unit

Nella presente colonna, la cui compilazione è del tutto facoltativa ai fini della buona riuscita della mappatura del driver XPANGO, va inserita l'unità di misura della grandezza trasportata dalla variabile in questione. L'unità di misura qui specificata, sarà poi visualizzata nell'interfaccia dell'IoT-SCADA insieme al valore della variabile in corso di monitoraggio. Per le tensioni, le correnti e le potenze si utilizzano, di solito, il Volt, l'Ampere ed il kilowatt (a seconda delle potenze in gioco). Molto spesso le unità di misura con cui il dispositivo visualizza le grandezze sono indicate dai costruttori, tuttavia non è necessario attenersi alle loro indicazioni.

Ad esempio, il dispositivo può fornirci il valore della potenza attiva in uscita da un inverter in W. Parlando di dispositivi di vari kW di potenza può capitare che venga visualizzato un valore molto grande e quindi di non comoda gestione. Se, per comodità, vogliamo che la potenza attiva in uscita dall'inverter sia espressa in kW possiamo indicare tale unità di misura nella colonna *Measurement Unit* ed un fattore moltiplicativo pari a 0,001 nella colonna *Scale* (Vedi par. 2.1.2.9).

2.1.2.10 Scale

Nella seguente colonna vanno indicati i fattori moltiplicativi da applicare al valore della variabile monitorata prima di visualizzarla all'utente.

La compilazione di questo campo risulta necessaria qualora vogliamo che l'utente visualizzi il dato misurato dal dispositivo in un'unità di misura da noi stabilita e non decisa dal costruttore.

A titolo di esempio riportiamo in *Figura 11* un estratto della mappa Modbus del contatore di energia modello S504C-6-MOD-MID della Seneca.

2 I Template Modbus e Siemens S7

Parametro	Cod. di funzione (Hex)	Segno	INTERO			IEEE		
			Registro (Hex)	Word	U. M.	Registro (Hex)	Word	U. M.
VALORI DEI CONTATORI TOTALI								
+kWh1 • Energia attiva importata fase 1	03 / 04		0100	3	0.1 Wh	1100	2	Wh
+kWh2 • Energia attiva importata fase 2	03 / 04		0103	3	0.1 Wh	1102	2	Wh
+kWh3 • Energia attiva importata fase 3	03 / 04		0106	3	0.1 Wh	1104	2	Wh
+kWhΣ • Energia attiva importata di sistema	03 / 04		0109	3	0.1 Wh	1106	2	Wh

Figura 10 - Mappa Modbus del contatore di energia S504C-6-MOD-MID di Seneca

Guardando ai valori degli indirizzi forniti secondo la standardizzazione IEEE, si nota come l'Energia attiva importata sulle 3 fasi sia espressa in Wh. Se volessimo visualizzare la variabile in kWh basta inserire un fattore moltiplicativo pari a 0,001. Se volessimo visualizzarla in MWh il fattore da applicare è 1000.

ID	Variable name *	Section name	Measurement unit	Scale
31	+kWh1 • Energia attiva importata fase 1		Wh	kWh 0,001
32	+kWh2 • Energia attiva importata fase 2		Wh	kWh 0,001
33	+kWh3 • energia attiva importata fase 3		Wh	MW 1.000,0

Figura 11 - Template Modbus del contatore di energia S504C-6-MOD-MID di Seneca: unità di misura e fattori di scala applicati

2.1.2.11 Offset Minimum e Maximum

I fattori inseriti nella colonna *Offset* vengono sommati al valore della variabile misurata dal dispositivo prima di presentarla all'utente. I campi *Minimum* e *Maximum* vanno completati quando sono noti i valori massimo e minimo che la variabile assumerà. Questi due valori saranno visualizzati nell'interfaccia dell'IoT-SCADA sulla riga corrispondente alla variabile considerata sotto le voci *min* e *max*.

Per dare un'idea di come può essere utile settare questi parametri facciamo un esempio.

Immaginiamo di poter comunicare con un piranometro il cui segnale in uscita è una corrente proporzionale all'irraggiamento misurato dal dispositivo. Di notte, con irraggiamento pari a 0 W/m² la corrente in uscita dal piranometro sarà pari a 4 mA mentre la corrente corrispondente al massimo irraggiamento misurabile (1000 W/m² ad esempio) è pari a 20 mA.

Per far sì che l'utente legga direttamente il valore dell'irraggiamento in W/m² e non il valore della corrente I in uscita dal piranometro dovremmo implementare la seguente relazione:

$$\text{irraggiamento} = \frac{(I - 4)}{16} * 1000 \quad \left[\frac{W}{m^2} \right]$$

In questo caso, per visualizzare al lettore il valore dell'irraggiamento potremmo impostare un fattore *Scale* pari a 62,5 (1000/16) e un fattore di *Offset* pari a -4. I valori minimo e massimo che la variabile potrà assumere sono rispettivamente 0 W/m² e 1000 W/m².

2 I Template Modbus e Siemens S7

Si noti che è possibile impostare tali parametri direttamente dall'interfaccia dell'IOT SCADA Server (Vedi *Manuale d'installazione e uso IoT SCADA Server – par. 5.2.3*).

2.1.2.12 Alarmed

I campi della colonna *Alarmed* sono di tipo booleano quindi assumono solamente i due valori TRUE e FALSE. Il campo sarà impostato su TRUE se alla variabile in questione è associato un allarme, in caso contrario daremo il valore FALSE o lasceremo vuoto il campo.

Una volta settato TRUE nella cella *Alarmed* di un dato di tipo Booleano, il nome della variabile sarà copiato all'interno del tab *Digital Alarms*; per tutti gli altri tipi di dati la copia sarà effettuata nel tab *Analog Alarms*. Per maggiori dettagli si rinvia il lettore ai paragrafi 2.1.2 e 2.1.4 del presente manuale.

2.1.2.13 Decode

All'interno della colonna *Decode* è possibile creare una corrispondenza tra il valore assunto dalla variabile e un valore numerico o alfanumerico da visualizzare all'utente. Per fare ciò bisogna inserire all'interno della cella informazioni del tipo *nome variabile 1 = valore utente 1, nome variabile 2 = valore utente 2* e così via. Ogni campo va separato con una virgola. In tal modo quando la variabile assumerà il valore indicato l'utente visualizzerà il corrispondente valore o messaggio.

2.1.2.14 Totalizer

I campi della colonna *Totalizer* saranno impostati su TRUE se le variabili in questione sono delle grandezze crescenti nel tempo e si desidera farne un report, ovvero un'analisi dell'andamento della misura su base giornaliera o mensile. Ad esempio, ha senso attivare tale funzione per grandezze cumulabili (monotone crescenti) come l'energia prodotta mentre è inutile applicarla a grandezze come la potenza prodotta, poiché quest'ultima può anche diminuire durante il funzionamento di un impianto.

Oltre al report su misura singola è possibile confrontare le differenze tra due o più grandezze totalizzatore su un dato arco temporale oppure confrontare la stessa misura su archi temporali distinti. Per maggiori dettagli a riguardo rimandiamo al *par. 6.5 del Manuale d'installazione ed uso IoT SCADA Server*.

Ad esempio, settiamo su TRUE la colonna *Totalizer* sulle variabili *Energia in ingresso al sistema* ed *Energia in uscita al sistema* nel Modbus template del contatore di energia S504C-6-MOD-MID di Seneca.

ID	Variable name *	Section name	Totalizer (default: false)	Preselected (default: true)
34	+kWhΣ • System imported active energy 1		TRUE	TRUE
35	-kWh1 • Phase 1 exported active energy 1			TRUE
36	-kWh2 • Phase 2 exported active energy 1			TRUE
37	-kWh3 • Phase 3 exported active energy 1			TRUE
38	-kWhΣ • System exported active energy 1		TRUE	TRUE

Figura 12 - Template Modbus del contatore di energia S504C-6-MOD-MID di Seneca: variabili totalizer

2 | Template Modbus e Siemens S7

Accedendo all'interfaccia dell'IoT SCADA Server all'indirizzo IP 192.168.1.29, caricando correttamente l'XPANGO nel catalogo dispositivi e configurandolo attraverso la sezione "Gestione Dispositivi Impianto" (Vedi Manuale d'installazione ed uso IoT SCADA Server – par.5) ci spostiamo sulla tab *Report*. Cliccando  tasto della tabella Parametri del report andremo a scegliere le variabili del dispositivo di cui effettuare il report (Vedi Manuale d'installazione ed uso IoT SCADA Server – par.6.5). Le uniche variabili selezionabili sono quelle in cui il campo *Totalizer* è stato precedentemente impostato su TRUE.

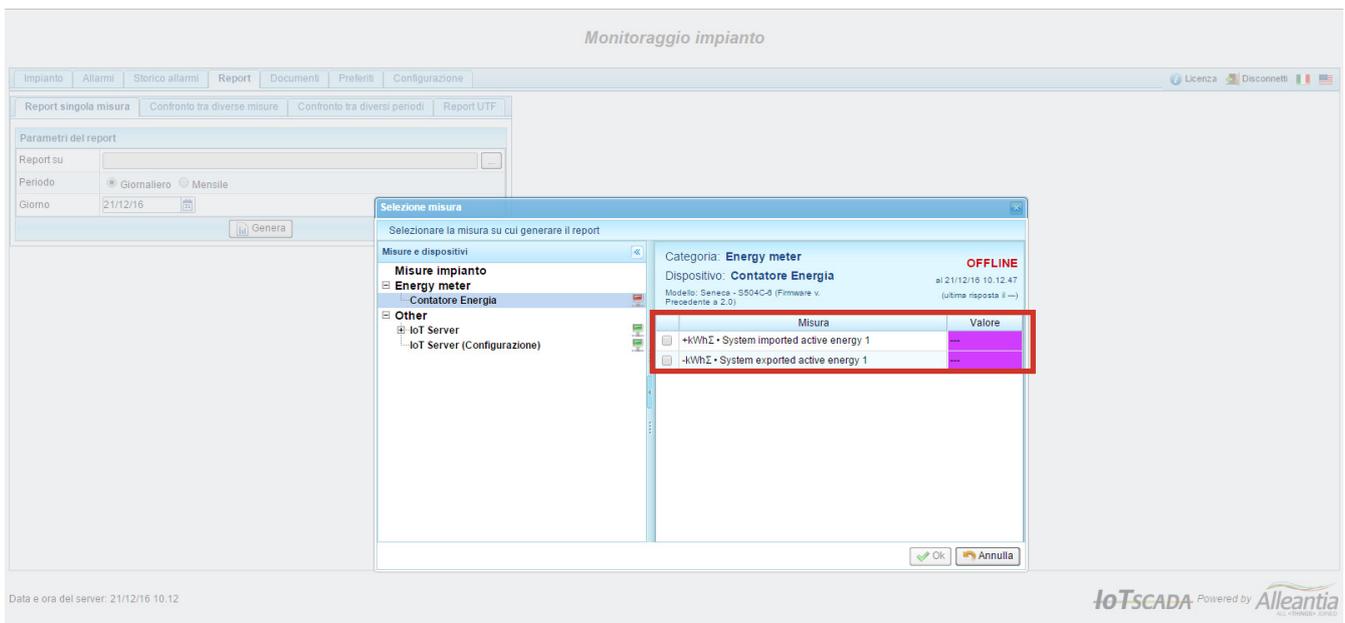


Figura 13 – Interfaccia dell'IoT SCADA Server: lista delle variabili per le quali è possibile generare un report

Si sottolinea come sia importante selezionare le grandezze che per loro natura sono delle funzioni monotone crescenti, e quindi su cui è eseguibile un report, direttamente durante la compilazione del template Modbus.

2.1.2.15 Preselected

Durante il monitoraggio del dispositivo può risultare utile nascondere dalla visualizzazione dell'interfaccia alcune delle variabili inserite all'interno del Modbus così da ridurre il numero delle variabili stesse. È possibile fare ciò impostando il valore FALSE nella cella *Preselected* della riga relativa alla variabile considerata. Qualora si decida di visualizzare le variabili precedentemente deselezionate basta settare il campo in TRUE. Quest'ultima operazione è eseguibile direttamente dall'interfaccia dell'IoT-SCADA Server (Vedi *Manuale d'installazione e uso IoT SCADA Server – par. 5.2.3*). Si noti che il valore di default del campo *Preselected* è TRUE.

2.1.3 Digital Alarms

Nel tab *Digital Alarms* compariranno tutte le variabili di tipo booleano per le quali il campo *Alarmed* del tab *General* verrà impostato su TRUE (Vedi *par 2.1.2.12*).

2 | Template Modbus e Siemens S7

ID	Variable name *	Section name	Alarmed (default: false)
1	DI Value channel 0	Digital Input	TRUE
2	DI Value Channel 1	Digital Input	
3	DO Value Channel 0	Digital Output	TRUE
4	DO Value Channel 1	Digital Output	
5	AI Value Channel 0	Analogic Input	
6	AI Value Channel 1	Analogic Input	
7	AI Value Channel 2	Analogic Input	
8	AI Value Channel 3	Analogic Input	
9	AI Value Channel 4	Analogic Input	
10	AI Value Channel 5	Analogic Input	
11	AO Value Channel 0	Analogic Output	
12	AO Value Channel 1	Analogic Output	
13	AI Status Channel 0		
14	AI Status Channel 1		
15	AI Status Channel 2		
16	AI Status Channel 3		
17	AI Status Channel 4		
18	AI Status Channel 5		
19			
20			
21			
22			
23			
24			
25			
26			
27			

ID	Variable name	Section name	Alarm condition *
1	DI Value channel 0	Digital Input	
2			
3	DO Value Channel 0	Digital Output	
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			

Figura 14 – Tab Variables (a destra) e Digital Alarms (a sinistra) del template Modbus

A questo punto la *number ID* delle variabili considerate sarà evidenziato in Rosso mentre i campi *Variable name* e *Section name* del tab *Digital Alarms* saranno compilati in maniera automatica, nelle righe corrispondenti all'ID delle variabili selezionate.

Gli allarmi digitali sono di tipo booleano e in quanto tali possono notificare solamente due condizioni, TRUE e FALSE. A tal proposito bisogna modificare il campo della colonna *Alarm Condition*, nella quale verrà specificato il valore (TRUE o FALSE) che la variabile dovrà assumere affinché la condizione di allarme sia notificata.

Portando come esempio il Modbus template del dispositivo ADAM – 6024 di Advantech, vogliamo segnalare con un allarme la condizione in cui l'ingresso digitale 0 non è funzionante o è inattivo. In questo caso bisogna impostare il campo *Alarm Condition* della variabile *DI Value Channel 0* su FALSE. Così facendo l'utente sarà avvisato qualora la variabile booleana scelta assumerà, per una qualunque ragione, un valore pari a zero.

Oltre alla condizione di allarme è necessario compilare i campi *Delay ON* e *Delay OFF*.

Nel campo *Delay ON* va inserito l'intervallo di tempo, in millisecondi, trascorso il quale l'allarme sarà notificato. Ad esempio, Inserendo nel campo *Delay ON* il valore 100, dal momento in cui il valore della variabile monitorata assumerà il valore impostato nella colonna *Alarm Condition* e l'istante in cui l'allarme sarà notificato passeranno esattamente 100 ms. Tale accorgimento è utile al fine di evitare la notifica di "falsi allarmi". Se si decidesse di impostare tale tempo a 0 ms, l'utente riceverà una segnalazione ogni qualvolta il valore della variabile sarà uguale a quello impostato nella *Alarm Condition* anche se si tratta di un piccolo transitorio e non di una vera anomalia. Settando un tempo di delay ON "ragionevole" è possibile aumentare la probabilità che il dispositivo segnali esclusivamente le condizioni di malfunzionamento del componente monitorato.

Al contrario, per essere sicuri che un'anomalia si sia effettivamente estinta è necessario prevedere un certo intervallo di tempo tra l'istante in cui il valore della variabile monitorata ritorna al suo valore normale (diverso da quello impostato nella *Alarm Condition*) e il momento in cui la notifica di allarme viene ritirata. Tale intervallo di tempo va inserito, in millisecondi, nella colonna *Delay OFF*.

Infine, nella colonna *Alarm Description* è possibile inserire una stringa per descrivere la tipologia di allarme notificato e, eventualmente, dare indicazioni sulle cause che hanno generato l'anomalia. Il messaggio contenuto nella cella in questione sarà visualizzato all'utente al momento della notifica dell'allarme nell'interfaccia dell'IoT-SCADA Server.

2 | Template Modbus e Siemens S7

2.1.4 Analog Alarms

A differenza degli allarmi descritti nel paragrafo precedente, gli allarmi analogici hanno una logica un po' più sofisticata. Un allarme analogico confronta due valori attraverso un operatore di allarme (=, !=, >, <, etc.) e viene notificato qualora la condizione specificata si realizza. Poiché il confronto avviene su variabili che possono assumere valori diversi da 0 o 1, all'interno del tab *Analog Alarms* compariranno tutte le variabili non booleane per le quali il campo *Alarmed* è stato settato su TRUE.

Figura 15 - Tab Variables (a destra) e Analog Alarms (a sinistra) del template Modbus

Come nel caso degli allarmi digitali, anche qui i campi *Variable name* e *Section name* vengono compilati in automatico e l'ID delle variabili in questione viene evidenziato in rosso nel tab *General*.

In questo caso l'alarm condition viene impostata modificando i campi delle due colonne *Alarm Operator* e *Reference Value*. Nel primo campo va inserito un operatore di confronto, selezionabile dal menu a tendina di cui è provvista la cella, mentre nel secondo campo va inserito il valore che sarà confrontato con la variabile monitorata.

Ad esempio, se vogliamo che l'allarme sia notificato quando la variabile *AI Value Channel 0* assumerà un valore diverso da 10, sceglieremo il simbolo di "!=" nella colonna *Alarm Operator* ed inseriremo il valore 10 nella colonna *Reference Value*.

ID	Variable name	Section name	Alarm operator *	Reference value *	Delay ON (default: 0 ms)	Delay OFF (default: 0 ms)	Alarm description *
1							
2							
3							
4							
5							
6							
7	AI Value Channel 0	Analogic Input	!=	10,0			
8	AI Value Channel 1	Analogic Input	<				
9			<=				
10			=				
11			>=				
12			>				
13							

Figura 16 - Tab Analog Alarms: tipi di Alarm operator settabili.

Analogamente a quanto detto per gli allarmi digitali, anche per gli *Analog Alarms* è possibile impostare un tempo di *Delay ON*, un tempo di *Delay OFF* e visualizzare all'utente una stringa con la descrizione dell'allarme notificato (Vedi par 2.1.3).

2 | Template Modbus e Siemens S7

2.1.5 Virtual Variables

All'interno del tab *Virtual Variables* è possibile creare delle variabili personalizzate a partire dalle grandezze proprie del dispositivo inserite all'interno del tab *Variables*.

Oltre al nome della variabile virtuale e la sezione di appartenenza, va inserito la tipologia di dato cui la variabile appartiene. Dal menu a tendina della colonna *Data Type* è possibile scegliere tra due formati: booleano e numerico.

All'interno della colonna *Expression* va inserita l'espressione numerica grazie alla quale si otterrà la variabile virtuale a partire dalle variabili reali fornite dal dispositivo. L'espressione va inserita richiamando le variabili contenute nel tab *Variables* con l'espressione "\$ID variabile". La formula non deve essere preceduta dal segno = come quando si implementano le formule di un file Excel.

Il resto delle colonne presenta le stesse caratteristiche di quelle presenti nel tab *Variables* (vedi par 2.1.2).

A titolo di esempio, mettiamo di voler verificare che la variabile *System Active Power* fornita dal contatore di energia Seneca S504C-6-MOD-MID sia calcolata effettivamente in maniera corretta. A tal proposito creiamo una variabile virtuale che chiamiamo *Potenza attiva calcolata* che sarà data dalla formula:

$$\text{Potenza attiva calcolata} = I * V * \cos \varphi$$

Dove I è la corrente del sistema, V la tensione del sistema e $\cos \varphi$ il fattore di potenza.

La Formula verrà implementata nella cella della colonna *Expression* del seguente modo:

$$\$7 * \$12 * \$16$$

Guardando al tab *Variables* si può notare come agli ID 7, 12 e 16 corrispondano le variabili ($V\Sigma$ • System voltage), ($A\Sigma$ • System current) e ($PF\Sigma$ • System power factor).

ID	Variable name *	Section name	Maximum
1	V1 • L-N voltage phase 1		
2	V2 • L-N voltage phase 2		
3	V3 • L-N voltage phase 3		
4	V12 • L-L voltage line 12		
5	V23 • L-L voltage line 23		
6	V31 • L-L voltage line 31		
7	VΣ • System voltage		
8	A1 • Phase 1 current		
9	A2 • Phase 2 current		
10	A3 • Phase 3 current		
11	AN • Neutral current		
12	AΣ • System current		
13	PF1 • Phase 1 power factor		
14	PF2 • Phase 2 power factor		
15	PF3 • Phase 3 power factor		
16	PFΣ • System power factor		
17	P1 • Phase 1 active power		
18	P2 • Phase 2 active power		
19	P3 • Phase 3 active power		
20	PΣ • System active power		
21	S1 • Phase 1 apparent power		
22	S2 • Phase 2 apparent power		
23	S3 • Phase 3 apparent power		
24	SΣ • System apparent power		
25	Q1 • Phase 1 reactive power		
26	Q2 • Phase 2 reactive power		
27	Q3 • Phase 3 reactive power		

ID	Variable name *	Expression *
10001	Potenza Attiva Calcolata	\$7*\$12*\$16
10002		
10003		
10004		
10005		
10006		
10007		
10008		
10009		
10010		
10011		
10012		
10013		
10014		
10015		
10016		
10017		
10018		
10019		
10020		
10021		
10022		
10023		
10024		
10025		
10026		
10027		

Figura 17 – Tab Virtual Variables: implementazione dell'espressione analitica della variabile virtuale

2 | Template Modbus e Siemens S7

Se il valore della Potenza attiva calcolata e quella fornita dal dispositivo coincidono il contatore sta operando correttamente.

La variabile virtuale creata sarà visibile insieme alle variabili virtuali nell'interfaccia dell'IoT SCADA Server.

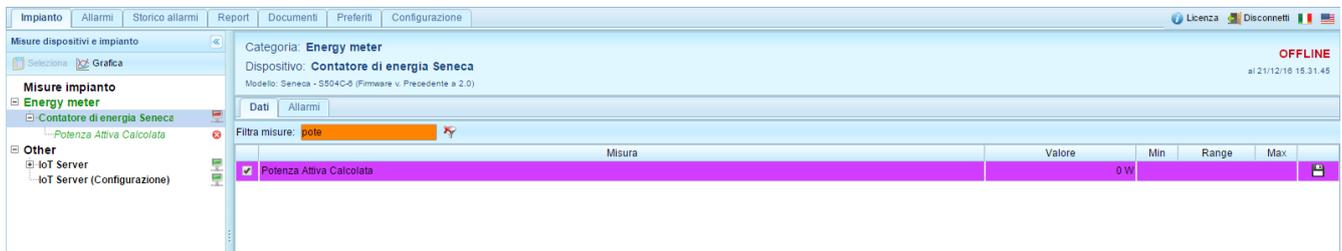


Figura 18 – Visualizzazione della variabile virtuale creata nell'interfaccia dell'IoT Scada Server

2.2 Il Template Modbus: esempio di compilazione

A titolo di esempio si riporta la compilazione del Template Modbus necessaria per la creazione del driver XPANGO del contatore di Energia modello S504C di Seneca.

Per prima cosa bisogna scaricare la mappa Modbus del dispositivo in questione reperibile sul sito ufficiale dell'azienda all'indirizzo <https://www.seneca.it/linee-di-prodotto/energia-e-misure-elettriche/contatori-di-energia-serie-s500/s504c-6>. Dalla sezione Download possiamo scaricare il file **Manuale di comunicazione ModBUS LAN.pdf** contenente la mappa Modbus del dispositivo.

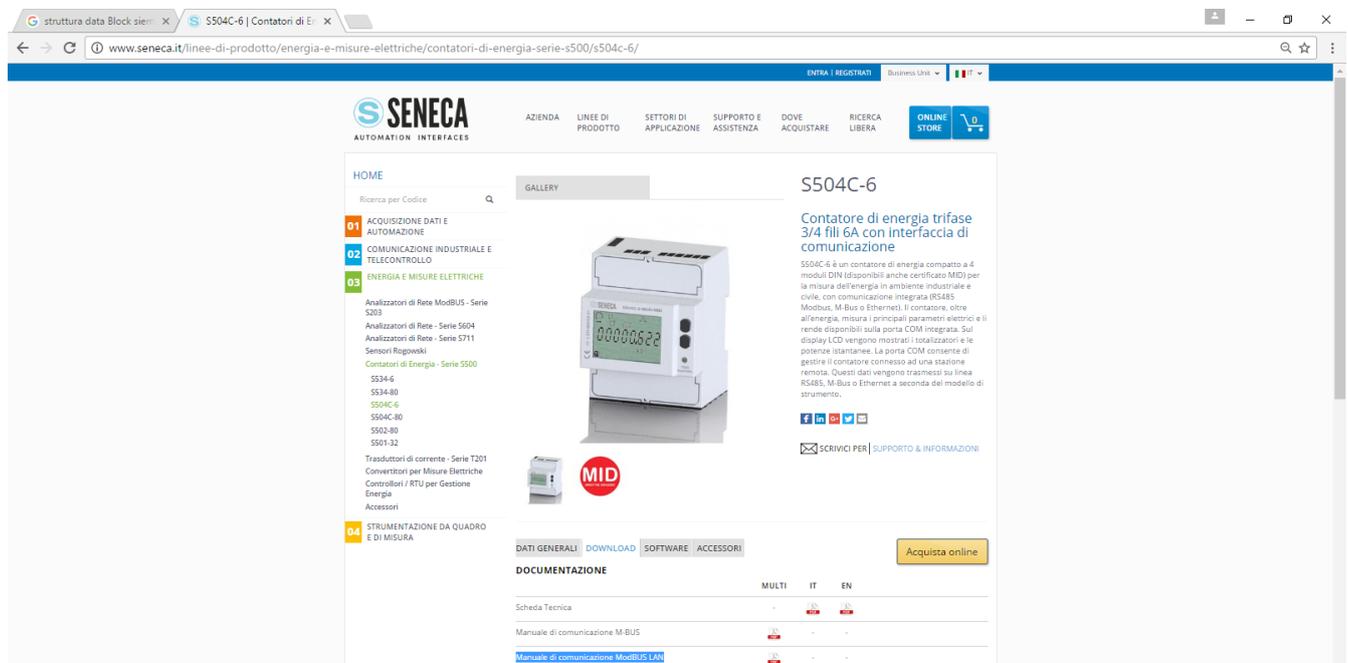


Figura 19 - Pagina Web del sito ufficiale di Seneca dalla quale è possibile scaricare la mappa Modbus del contatore di Energia S504C-6-MOD-MID

2 | Template Modbus e Siemens S7

La mappa Modbus del dispositivo fornita dal costruttore si presenta in forma tabellata. Il costruttore fornisce due set di registri numerati con 0 e 1 rispettivamente. Il Set di registri 1 è disponibile solo per contatori con MODBUS integrato, contatori con ETHERNET integrata o moduli RS485 con versione firmware 2.00 o superiore, mentre il Set 0 è valido per qualsiasi dispositivo. Dato il carattere “universale” del set di registri 0 sarà questo che utilizzeremo nell’esempio.

6. Set di registri 0

6.1 Registri di LETTURA (Codice di funzione \$01/\$03/\$04)

Parametro	Cod. di funzione (Hex)	Segno	INTERO			IEEE		
			Registro (Hex)	Word	U. M.	Registro (Hex)	Word	U. M.
VALORI IN TEMPO REALE								
V1 • Tensione L-N fase 1	03/04		0000	2	mV	1000	2	V
V2 • Tensione L-N fase 2	03/04		0002	2	mV	1002	2	V
V3 • Tensione L-N fase 3	03/04		0004	2	mV	1004	2	V
V12 • Tensione di linea L-L 12	03/04		0006	2	mV	1006	2	V
V23 • Tensione di linea L-L 23	03/04		0008	2	mV	1008	2	V
V31 • Tensione di linea L-L 31	03/04		000A	2	mV	100A	2	V
VΣ • Tensione di sistema	03/04		000C	2	mV	100C	2	V
A1 • Corrente fase 1	03/04	X	000E	2	mA	100E	2	A
A2 • Corrente fase 2	03/04	X	0010	2	mA	1010	2	A
A3 • Corrente fase 3	03/04	X	0012	2	mA	1012	2	A
AN • Corrente di neutro	03/04	X	0014	2	mA	1014	2	A
AΣ • Corrente di sistema	03/04	X	0016	2	mA	1016	2	A
PF1 • Fattore di potenza fase 1	03/04	X	0018	1	-	1018	2	-
PF2 • Fattore di potenza fase 2	03/04	X	0019	1	-	101A	2	-
PF3 • Fattore di potenza fase 3	03/04	X	001A	1	-	101C	2	-
PFΣ • Fattore di potenza di sistema	03/04	X	001B	1	-	101E	2	-
P1 • Potenza attiva fase 1	03/04	X	001C	3	mW	1020	2	W

Figura 20 – Estratto mappa Modbus del contatore di energia S504C-6-MOD-MID di Seneca (Set di registri 0)

Nella colonna *Parametro* vengono forniti i nomi delle variabili che è possibile monitorare.

Nella seconda colonna si riporta il *Codice di funzione* espresso in esadecimale. Il Codice di funzione ci dà indicazione sulla tipologia di registro ovvero, se si tratta di un registro di lettura (Cod. \$01/\$03/\$04) o di scrittura (Cod. \$10).

Nella colonna *Segno* sarà riportata una “X” se la variabile può assumere valori negativi, mentre nel caso in cui potrà assumere solo valori positivi il campo sarà vuoto.

Il costruttore fornisce due categorie di indirizzi Modbus. La prima categoria utilizza una rappresentazione dei valori in numeri interi, la seconda utilizza una rappresentazione in virgola mobile o float a 32 bit secondo lo standard di rappresentazione IEEE. Scegliamo la seconda poiché è di carattere universale essendo una rappresentazione di tipo standard.

Ciascuna delle due categorie è suddivisa in 3 sotto-colonne. La prima fornisce il numero, in esadecimale, del registro Modbus attraverso cui è possibile ricavare i valori assunti dalla variabile in questione, la seconda ci dà la capienza del registro in word (1 word=16 bit) e la terza fornisce l’unità di misura della grandezza monitorata. Detto questo è possibile passare alla compilazione del template Modbus. Nello specifico ci riferiremo al contatore di energia S504C-6-MOD-MID dotato di porta RS485 integrata.

2 | Template Modbus e Siemens S7

Nel tab *General* inseriremo le informazioni sul modello in questione e la modalità di comunicazione compilando i campi secondo quanto riportato nella tabella 2.

Device informations	
Supplier *	Seneca
Model *	S504C-6
Version	Firmware v. Precedente a 2.0
Category *	Energy meter
Communication	
Serial supported (default: false)	VERO
Ethernet supported (default: false)	VERO
Delay between requests (default: 100 ms)	
Protocol	Modbus
Batch enabled (default: false)	
Word order (default: Little Endian)	

Tabella 2 - Esempio di compilazione del tab *General* del template Modbus del contatore di energia S504C-6-MOD-MID di Seneca

È possibile comunicare con il contatore S504C-5-MOD-MID esclusivamente attraverso porta seriale (Serial Supported=TRUE) essendo sprovvisto di porta Ethernet per comunicazione tramite rete LAN (Ethernet Supported=FALSE). Nel nostro caso il campo *Delay between requests* non è stato compilato tuttavia, se intendiamo impostare un intervallo temporale diverso da quello di default pari a 100 ms, è possibile inserire il tempo che intercorre tra una richiesta e la successiva specificandone il valore in ms. La funzione di Batch è stata disabilitata (Batch enabled=FALSE).

Fatto questo possiamo passare alla compilazione del tab *Variables* cominciando dalla prima variabile fornita nella mappa "V1 • Tensione L-N fase 1".

Come **nome della variabile** possiamo copiare quello specificato nella colonna parametro della mappa Modbus.

L'**area** del registro che codifica per la variabile considerata, come vedremo meglio in seguito, non è specificata con chiarezza, tuttavia notiamo che la variabile viene memorizzata su un registro a 2 word (32 bit) quindi, di certo, non si tratta di registri Coil o Discrete Input la cui dimensione è pari a 1 bit. Detto ciò si tratta di registri di tipo Holding o Input.

Ricordando che la prima tipologia di registri è di tipo Read/Write mentre la seconda contiene esclusivamente dati di tipo Read (Vedi par. 2.1.2.3), possiamo dedurre che si tratta di un Input Register in quanto il codice di funzione della variabile, pari a \$03/\$04, codifica per un registro di lettura. Tuttavia è possibile specificare

2 | Template Modbus e Siemens S7

Holding come area senza compromettere il funzionamento del driver XPANGO, poichè i registri di tipo Holding possono contenere qualsiasi tipologia di dati, siano essi di tipo R, W o R/W.

Di solito, è possibile comprendere in maniera immediata l'area del registro dal valore dell'indirizzo Modbus della variabile da monitorare. Convertendo il valore dell'indirizzo da esadecimale (HEX=1000) a decimale troviamo il valore 4096. Secondo lo standard Modbus, un indirizzo con valore compreso tra 0 e 9999 è di tipo coil (vedi *par. 2.1.2.4*) ma ciò non concilia con il fatto che la variabile *V1 • Tensione L-N fase 1* è memorizzata su 32 bit. Da ciò si capisce che gli indirizzi non sono forniti secondo tale standard e di conseguenza non è stato possibile ricavare da questi l'area di memoria della quale il registro fa parte. Il valore convertito in decimale va inserito all'interno della colonna *Address* nella riga corrispondente alla variabile considerata.

Il campo *Bit of word* va lasciato vuoto in quanto è necessario specificare il bit da indirizzare soltanto nel caso in cui, ad esempio, bisogna memorizzare più dati di tipo boolean all'interno di registri a 16 bit.

Il *Data Type* è Float poichè lo standard IEEE utilizza la rappresentazione in virgola mobile.

Nella colonna *Signed* selezioneremo TRUE o FALSE a seconda dell'indicazione fornita nella colonna *Segno* della mappa Modbus. Nel caso della prima variabile impostiamo FALSE.

I campi della colonna *Access type* possono essere lasciati vuoti in quanto sono impostati di default su Read e nel questo caso le variabili sono di sola lettura (cod. \$03/\$04).

L'unità di misura, in questo caso Volt (V), verrà specificata nella colonna *Measurement Unit*.

È possibile lasciare vuote tutte le altre colonne poiché nessuna informazione aggiuntiva è fornita dal costruttore. La colonna *Alarmed* va impostata su TRUE se vogliamo associare un allarme alla grandezza considerata, mentre le colonne *Totalizer* e *Preselected* sono impostate su valori di default che vanno bene per la variabile in questione (vedi *par. 2.1.2.12, 2.1.2.14, 2.1.2.15*) in quanto non si tratta di una grandezza rappresentabile da una funzione monotona crescente (grandezza non cumulabile) e vogliamo inoltre che il suo valore compaia nell'interfaccia dell'IOT SCADA Server.

Quanto detto sopra sarà fatto per tutte le variabili contenute all'interno della mappa modbus. Le variabili così elencate all'interno del file Excel compariranno nell'interfaccia dell'IOT SCADA Server una volta caricato il file XPANGO al suo interno e configurato correttamente il dispositivo con il quale si intende comunicare.

2 | Template Modbus e Siemens S7

Monitoraggio impianto

Impianto | Allarmi | Storico allarmi | Report | Documenti | Preferiti | Configurazione

Misure dispositivi e impianto

Categoria: **Energy meter**
Dispositivo: **Seneca S504C-6 o S534-6**
Modello: Seneca - S504C-6 (Firmware v. Precedente a 2.0)

OFFLINE
al 09/01/17 18.03.15
(ultima risposta 0 --)

Dati | Allarmi

Filtra misure:

Misura	Valore	Min	Range	Max
+KVAH1-C • Phase 1 imported leading apparent energy 1	---	---	---	---
+KVAH1-L • Phase 1 imported lagging apparent energy 1	---	---	---	---
+KVAH2-C • Phase 2 imported leading apparent energy 1	---	---	---	---
+KVAH2-L • Phase 2 imported lagging apparent energy 1	---	---	---	---
+KVAH3-C • Phase 3 imported leading apparent energy 1	---	---	---	---
+KVAH3-L • Phase 3 imported lagging apparent energy 1	---	---	---	---
+KVAHZ-C • System imported leading apparent energy 1	---	---	---	---
+KVAHZ-C • System imported leading apparent energy 4	---	---	---	---
+KVAHZ-L • System imported lagging apparent energy 1	---	---	---	---
+KVAHZ-L • System imported lagging apparent energy 4	---	---	---	---
+kvarh1-C • Phase 1 imported leading reactive energy 1	---	---	---	---
+kvarh1-L • Phase 1 imported lagging reactive energy 1	---	---	---	---
+kvarh2-C • Phase 2 imported leading reactive energy 1	---	---	---	---
+kvarh2-L • Phase 2 imported lagging reactive energy 1	---	---	---	---
+kvarh3-C • Phase 3 imported leading reactive energy 1	---	---	---	---
+kvarh3-L • Phase 3 imported lagging reactive energy 1	---	---	---	---
+kvarhZ-C • System imported leading reactive energy 1	---	---	---	---
+kvarhZ-C • System imported leading reactive energy 4	---	---	---	---
+kvarhZ-L • System imported lagging reactive energy 1	---	---	---	---
+kvarhZ-L • System imported lagging reactive energy 4	---	---	---	---

Data e ora del server: 09/01/17 18.03

IoTSCADA Powered by Alleantia
ALL THINGS JOINED

Figura 21 – Interfaccia dell’IoT SCADA Server: Elenco variabili del dispositivo configurato

Inserire immagine dal manuale SENECA 504, della tabella modbus creata (con indicazione delle variabili non inserite) e della immagine dalla interfaccia del sistema ISS (quando sarà live).

2.3 Il Template Siemens S7

Questo template è stato elaborato per ottenere il driver XPANGO dei PLC Siemens serie S7.

Di seguito non descriveremo nuovamente le colonne del file Excel che, per tipologia di dati contenuti e funzionalità, risultano identiche a quelle presenti nel template Modbus (Vedi *par. 2.1*). Adottando infatti un proprio protocollo di comunicazione, Il template Siemens S7 presenta alcune differenze rispetto al template precedente.

Prima di iniziare il collegamento con un dispositivo Siemens S7, verificare se siano abilitati gli elementi del TIA elencati nei paragrafi successivi:

S7 1200/1500

Un dispositivo esterno può accedere alla CPU S7 1200 / 1500 utilizzando il protocollo S7 “base”, funzionando solo come HMI, cioè sono consentiti solo i trasferimenti di dati di base.

In particolare per accedere ad un DB sono necessarie alcune impostazioni aggiuntive sul lato plc.

1. È possibile accedere solo ai DB globali.
2. L’accesso ottimizzato al blocco deve essere disattivato.
3. Il livello di accesso deve essere “pieno” e il “meccanismo di connessione” deve consentire GET / PUT.

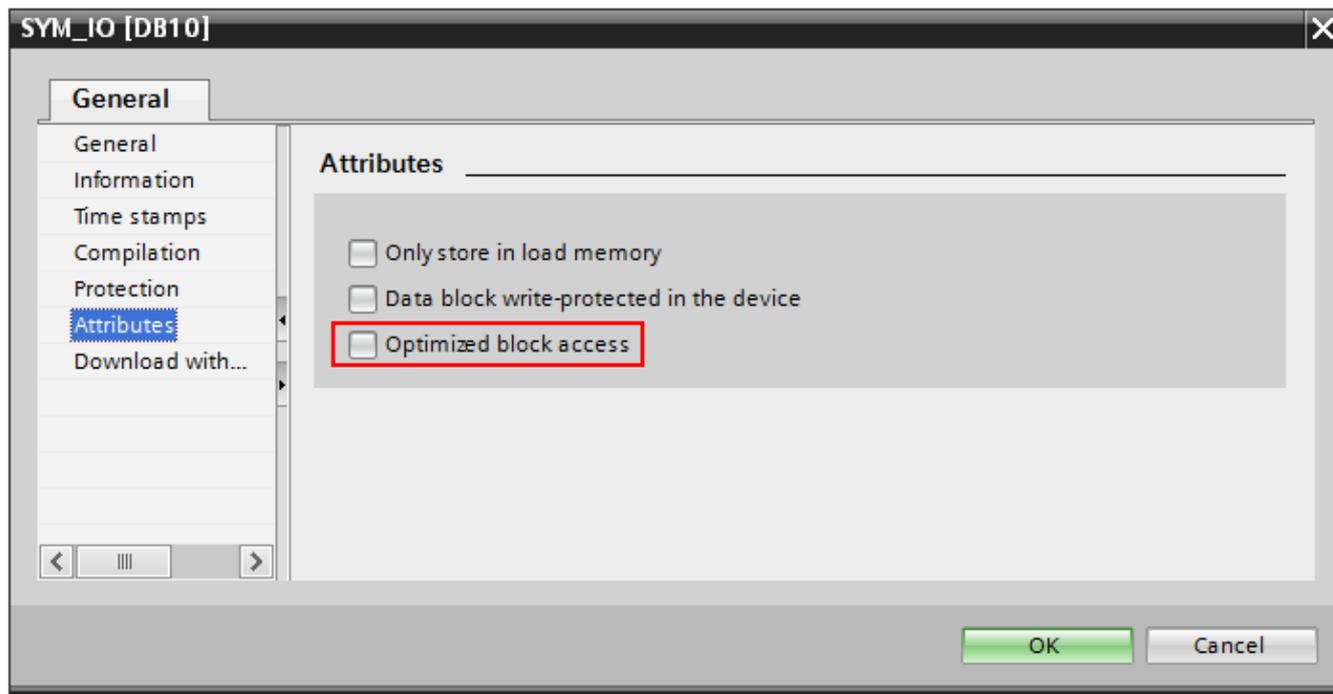
Vediamo queste impostazioni in TIA Portal.

2 | Template Modbus e Siemens S7

Proprietà DB

Selezionare il DB nel riquadro di sinistra sotto “Blocchi di programma” e premere Alt-Invio (o nel menu contestuale selezionare “Proprietà ...”)

De-seleziona l’accesso al blocco ottimizzato (per impostazione predefinita è selezionato).

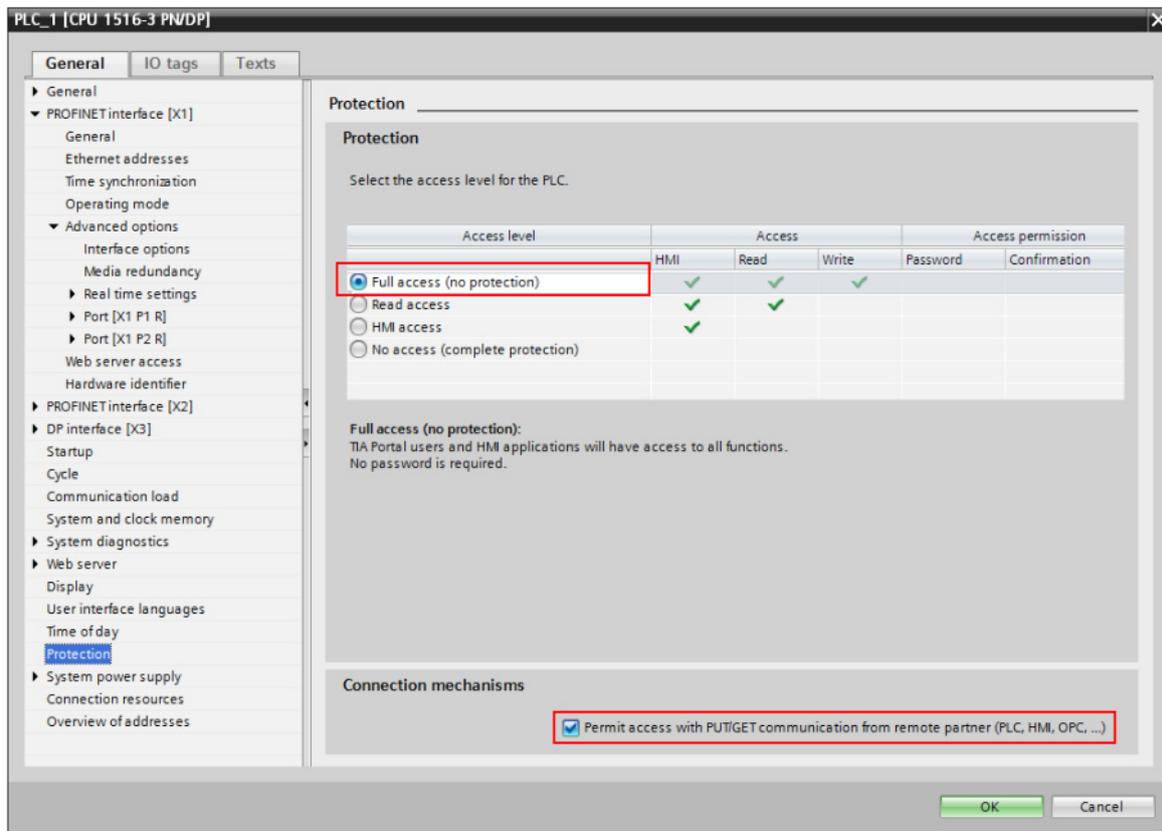


Protezione

Seleziona il progetto CPU nel riquadro di sinistra e premi Alt-Invio (o nel menu contestuale seleziona “Proprietà ...”)

Nella voce Protezione, selezionare “Accesso completo” e spuntare “Autorizza l’accesso con PUT / GET”
Come in figura.

2 | Template Modbus e Siemens S7



2.3.1 General

La Prima differenza sta nel campo *Category* posto all'interno del tab *General*. Qui, possiamo scegliere tra 3 categorie di dispositivi: *PLC*, *Numerical control* e *Other*. Sceglieremo la prima categoria se si tratta di Controllori a Logica Programmata (PLC), la seconda se si tratta di Macchine a Controllo Numerico (CNC) e la terza se il dispositivo non rientra tra le due categorie precedenti.

Il *Delay Between Request* è impostato di Default a 10 ms, quindi questo sarà il tempo che intercorre tra una richiesta e la successiva qualora il campo venga lasciato vuoto.

Infine nel template S7 non è possibile abilitare la funzione di *Batch data transmission* né scegliere tra i due *word order* Big e Little Endian poiché i PLC Siemens S7 utilizzano sempre l'ordinamento Big Endian.

2 | Template Modbus e Siemens S7

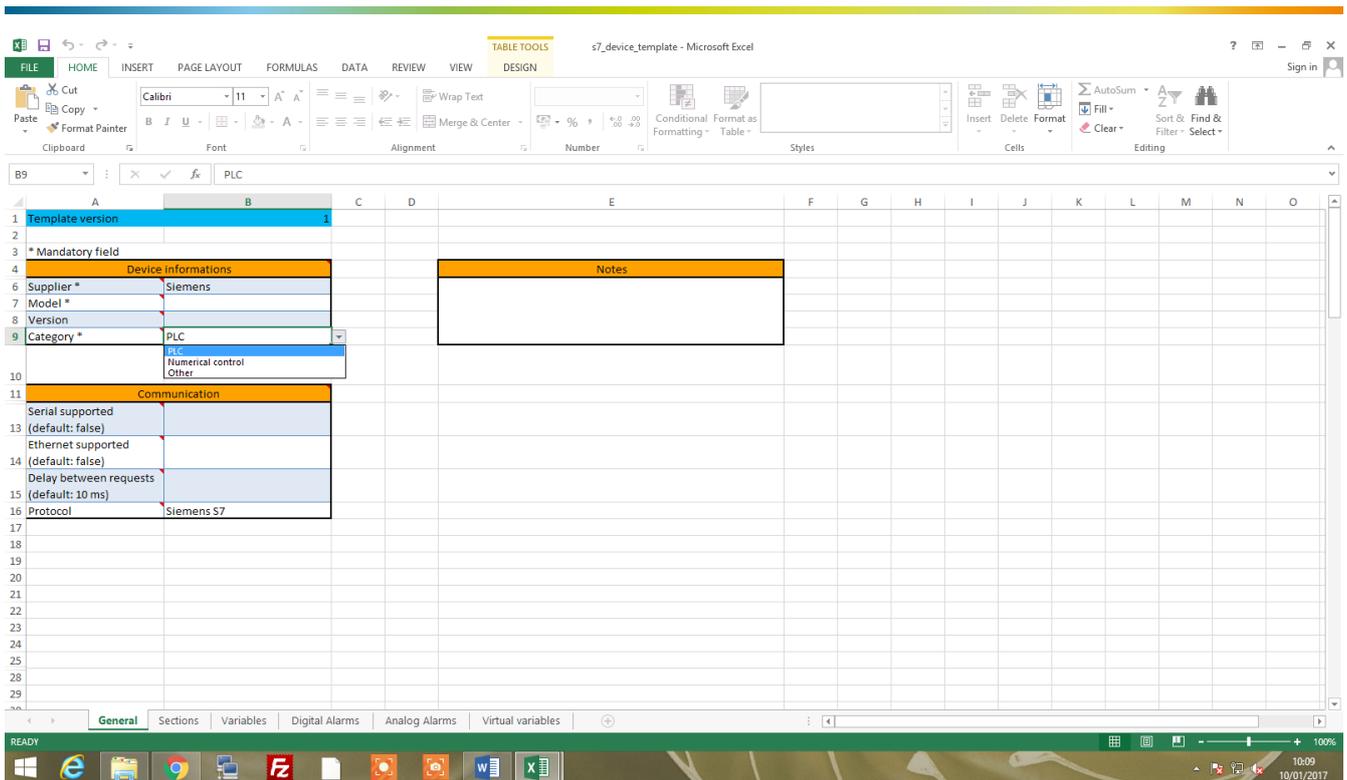


Figura 22 – Tab General del template Siemens S7

2.3.2 Variables

La struttura del tab *Variables* è del tutto identica a quella dell'analogo presente nel template Modbus. Anche qui, ciascuna variabile ha un codice identificativo univoco rappresentato dai numeri della colonna ID le cui celle non possono essere modificate.

In più vi sono delle differenze nei menu a tendina delle colonne *Area* e *Data Type* ed è presente una nuova colonna nominata *Data Block Number*. Di conseguenza saranno queste le colonne che saranno descritte nei paragrafi a seguire.

2.3.2.1 Area

Nella seguente colonna va specificata l'area di memoria entro cui è memorizzata la variabile. La memoria di un PLC Siemens è suddivisa in diverse aree ciascuna delle quali contiene gruppi di variabili con una funzione specifica. Il menu a tendina del template Siemens S7 individua 6 diverse aree di memoria:

- Input (IPI);
- Output (IPU);
- Marker;
- Timer;
- Counter;
- Data Block.

Nelle aree di memoria *IPI* ed *IPU* vengono memorizzati i valori degli ingressi logici all'inizio del ciclo di calcolo e delle uscite logiche alla fine del ciclo di calcolo, rispettivamente.

2 I Template Modbus e Siemens S7

L'area dei *Marker* è l'area destinata alle variabili utilizzate prevalentemente a bit e può essere programmata. All'interno dell'area *Timer* vengono inseriti i temporizzatori ovvero degli elementi capaci di conteggiare il tempo attraverso degli incrementi dt (dt=1ms/10ms/100ms).

I contatori vengono inseriti all'interno dell'area *Counter* e il loro compito è quello di effettuare un conteggio degli eventi relativi ad un segnale esterno. Vi sono contatori che contano solo in avanti, solo all'indietro e contatori bidirezionali.

Infine troviamo l'area *Data Block Globali* che è quella di maggiore interesse ai fini della comunicazione seriale con un PLC. All'interno di quest'area infatti, si trovano le variabili dichiarate dal programmatore del PLC e che potremmo leggere e monitorare una volta creato il relativo driver XPANGO.

2.3.2.2 Data Block Number

L'area *Data Block Globali* di un PLC è suddivisa in una serie di "scompartimenti" detti appunto *Data Block*. Il *Data Block* è una porzione di memoria che può accogliere dati di qualsiasi tipo e dimensione (bit, byte, word, etc.).

Le dimensioni fisiche di un *Data Block* non sono predefinite ma possono essere scelte dal programmatore. Generalmente hanno una dimensione di 1024 bit nei PLC Siemens S7-300 e S7-400. La dimensione totale è suddivisa in tanti *Data byte* di 8 bit ciascuno. Quindi nel caso dei PLC citati ciascun *Data Block* conterrà 128 *Data byte*.

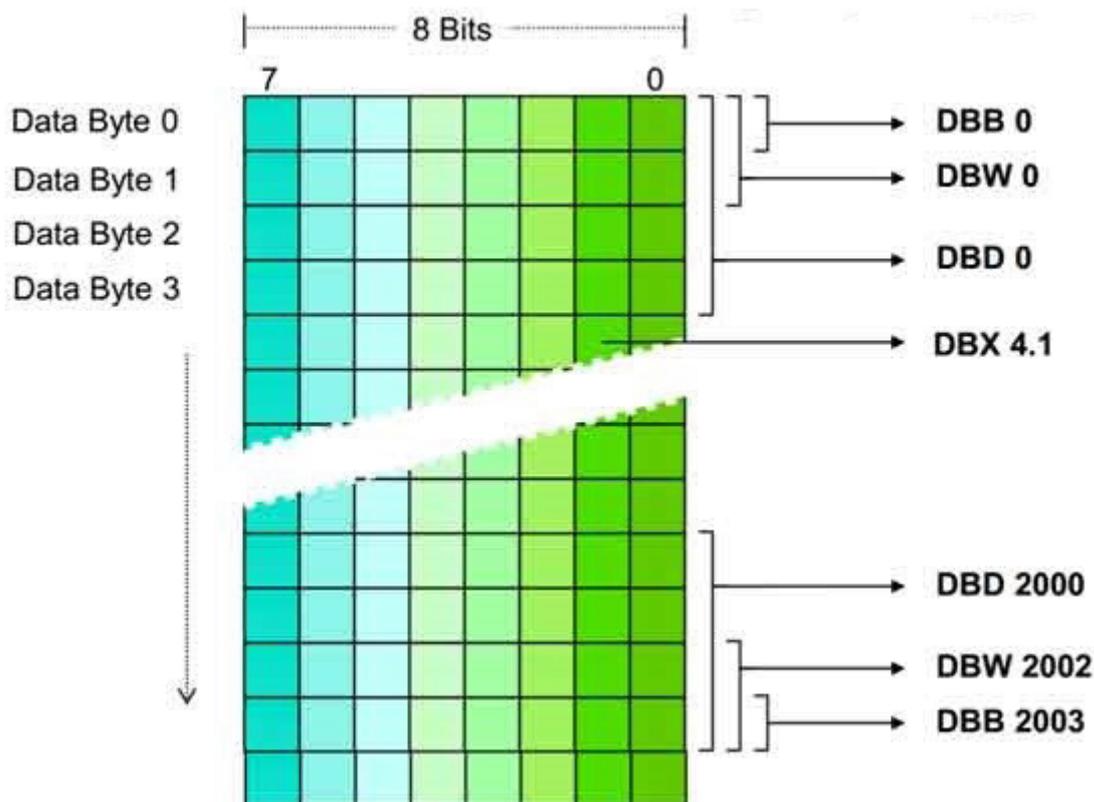


Figura 23 - Struttura della memoria di un PLC

2 I Template Modbus e Siemens S7

Ciascuna porzione di memoria contenente i dati delle variabili, è identificata univocamente da un numero. Tale numero identificativo va specificato nella colonna *Data Block Number*, ovvero va specificato il *Data Block* entro cui la variabile considerata è contenuta.

È utile sottolineare come il campo oggetto del presente paragrafo va lasciato vuoto qualora la variabile non si trovi all'interno dell'area di memoria *Data Block Globali*.

2.3.2.3 Data Type

La colonna *Data Type* del template Siemens S7 è provvista di un menu a tendina dalla quale è possibile selezionare il tipo di dato che esprime una data variabile. I Data type *Bool*, *Int* e *Dint* sono analoghi ai *Boolean*, *Integer* e *Double Integer* presenti nel Template Modbus. Le restanti tipologie sono del tutto nuove e possono essere così descritte:

- *Byte*: numero in formato esadecimale (8 bit);
- *Char*: carattere definito secondo lo standard ASCII (1 bit);
- *Word*: numero decimale senza segno (16 bit);
- *Double Word*: Numero decimale senza segno (32 bit)
- *Real*: numero a virgola mobile definito secondo lo standard IEEE (32 bit)
- *String*: Insieme di caratteri alfanumerici;
- *Date*: data in formato Standard IEC ad intervalli di un giorno.

2.4 Il Template Siemens S7: esempio di compilazione

Come esempio di compilazione riportiamo il Template Siemens S7 di un PLC Simatic S7-300. La tabella contenente le diverse variabili da monitorare è stata estratta dal software di programmazione Step 7 di Siemens.

2 | Template Modbus e Siemens S7

SINOTTICO	SIMBOLO	INDIRIZZO	TIPO	COMMENTO	NOTE
	FAULT_01	DB70.DBW0	INT	WORD ALLARMI 01	
	FAULT_02	DB70.DBW2	INT	WORD ALLARMI 02	
	FAULT_03	DB70.DBW4	INT	WORD ALLARMI 03	
	FAULT_04	DB70.DBW6	INT	WORD ALLARMI 04	
	FAULT_05	DB70.DBW8	INT	WORD ALLARMI 05	
	FAULT_06	DB70.DBW10	INT	WORD ALLARMI 06	
	FAULT_07	DB70.DBW12	INT	WORD ALLARMI 07	
	FAULT_08	DB70.DBW14	INT	WORD ALLARMI 08	
	FAULT_09	DB70.DBW16	INT	WORD ALLARMI 09	
	FAULT_10	DB70.DBW18	INT	WORD ALLARMI 10	
PT101	SEND_A.VCP1	DB70.DBD24	DINT	PRESSIONE VUOTO #1	3 decimali
PT201	SEND_A.VCP2	DB70.DBD28	DINT	PRESSIONE VUOTO #2	3 decimali
PT301	SEND_A.VCP3	DB70.DBD32	DINT	PRESSIONE VUOTO #3	3 decimali
LT101	SEND_A.LEV1	DB70.DBD36	DINT	LIVELLO CAMERA DI FLASH #1	1 decimale
LT201	SEND_A.LEV2	DB70.DBD40	DINT	LIVELLO CAMERA DI FLASH #2	1 decimale
LT301	SEND_A.LEV3	DB70.DBD44	DINT	LIVELLO CAMERA DI FLASH #2	1 decimale
TT101	SEND_A.COT1	DB70.DBD48	DINT	TEMPERATURA CAMERA DI FLASH #1	1 decimale
TT201	SEND_A.COT2	DB70.DBD52	DINT	TEMPERATURA CAMERA DI FLASH #2	1 decimale
TT301	SEND_A.COT3	DB70.DBD56	DINT	TEMPERATURA CAMERA DI FLASH #3	1 decimale
TT102	SEND_A.RCT1	DB70.DBD60	DINT	TEMPERATURA RICIRCOLO #1	1 decimale
TT202	SEND_A.RCT2	DB70.DBD64	DINT	TEMPERATURA RICIRCOLO #2	1 decimale
TT302	SEND_A.RCT3	DB70.DBD68	DINT	TEMPERATURA RICIRCOLO #3	1 decimale
TT103	SEND_A.ST1	DB70.DBD72	DINT	TEMPERATURA VAPORE #1	1 decimale
TT203	SEND_A.ST2	DB70.DBD76	DINT	TEMPERATURA VAPORE #2	1 decimale
TT303	SEND_A.ST3	DB70.DBD80	DINT	TEMPERATURA VAPORE #3	1 decimale
TT104	SEND_A.DT1	DB70.DBD84	DINT	TEMPERATURA DISTILLATO #1	1 decimale
TT204	SEND_A.DT2	DB70.DBD88	DINT	TEMPERATURA DISTILLATO #2	1 decimale
TT304	SEND_A.DT3	DB70.DBD92	DINT	TEMPERATURA DISTILLATO #3	1 decimale
TT105	SEND_A.DTT1	DB70.DBD96	DINT	TEMPERATURA SERBATOIO DISTILLATO #1	1 decimale
TT205	SEND_A.DTT2	DB70.DBD100	DINT	TEMPERATURA SERBATOIO DISTILLATO #2	1 decimale
TT305	SEND_A.DTT3	DB70.DBD104	DINT	TEMPERATURA SERBATOIO DISTILLATO #3	1 decimale
TT106	SEND_A.VCT1	DB70.DBD108	DINT	TEMPERATURA SERBATOIO DEL VUOTO #1	1 decimale
TT206	SEND_A.VCT2	DB70.DBD112	DINT	TEMPERATURA SERBATOIO DEL VUOTO #2	1 decimale
TT306	SEND_A.VCT3	DB70.DBD116	DINT	TEMPERATURA SERBATOIO DEL VUOTO #3	1 decimale

Figura 24 – Elenco degli indirizzi delle variabili di un PLC Siemens Simatic S7-300 estratto dal software Step 7

Le informazioni utili alla compilazione del template Excel sono raccolte nelle colonne INDIRIZZO, TIPO, COMMENTO e NOTE.

All'interno della colonna COMMENTO è inserito il nome della variabile che andiamo a monitorare quindi copieremo tale informazione nella colonna Variable name del template.

Gli indirizzi di comunicazione sono contenuti all'interno della colonna INDIRIZZO. Questi vengono forniti attraverso una rappresentazione tipica del protocollo di comunicazione Siemens S7. Per facilitare la comprensione lo analizziamo in ogni sua parte:

DB70.DBW0

I Primi 4 caratteri dell'indirizzo forniscono il *Data Block Number* del *Data Block* entro cui è memorizzata la variabile *WORD ALLARMI 01*.

Gli ultimi 4 caratteri, separati da un punto dai precedenti, danno informazione circa l'area di memoria entro cui la variabile è contenuta, le dimensioni del dato tramite cui la variabile è espressa e il numero del *Data byte* all'interno del *Data Block* che accoglie il dato.

Ciascun'area di un PLC ha infatti un proprio prefisso identificativo rappresentato dalle prime 2 lettere (DB, in questo caso). La terza lettera invece si riferisce alla dimensione del dato (Vedi la *Tabella 3*).

2 I Template Modbus e Siemens S7

Area di memoria	Dimensione del dato	Prefisso	Area di memoria	Dimensione del dato	Prefisso
Immagine di processo degli ingressi	Bit	E	Blocchi dati di istanza	Blocco	DI
	Byte	EB		Bit	DIX
	Word	EW		Byte	DIB
	Double	ED		Word	DIW
Immagine di processo delle uscite	Bit	A	Stack dei dati locali	Double	DID
	Byte	AB		Bit	L
	Word	AW		Byte	LB
	Double	AD		Word	LW
Area merker	Bit	M		Double	LD
	Byte	MB	Area di memoria		Prefisso
	Word	MW	Area Timer		T
	Double	MD	Area Contatori		Z
Blocchi dati globali	Blocco	DB			
	Bit	DBX			
	Byte	DBB			
	Word	DBW			
	Double	DBD			

Tabella 3 - Prefissi utilizzati per riportare gli indirizzi delle variabili di un PLC Siemens S7

Quindi, nel nostro caso, l'indirizzo fornito si riferisce ad una variabile nominata *WORD ALLARMI 01* contenuta nell'area di memoria *Data Block Globali* e rappresentata da un dato di dimensioni pari a una word, contenuto all'interno del primo *Data byte* (DBWO) del *Data Block n° 70* (DB70).

Quest'ultima informazione può essere inoltre dedotta notando come la variabile successiva non sia contenuta all'interno del Data byte n° 1 ma all'interno del n°2. Di conseguenza la variabile *WORD ALLARMI 01* occupa da sola i data byte numero 0 e 1.

Conoscere la dimensione fisica di un dato tuttavia non è sufficiente ad identificare completamente la tipologia del dato stesso. Ad esempio, sia le Double Word (DW) che i Double Integer (DInt) occupano 4 byte. All'interno della colonna *Data type* del template Siemens S7 andremo ad inserire l'informazione contenuta all'interno della colonna *TIPO* della tabella degli indirizzi. Nel nostro caso, la variabile in considerazione è espressa da un dato di tipo Integer (Int).

ID	Variable name *	Section name	Area *	Data Block Number	Address *	Bit	Data type *	Access type (default: Read)	Measurement unit	Scale
1	WORD ALLARMI 01		Data Block	70	0		Int			

Figura 25 – Esempio di inserimento di una variabile a 16 bit all'interno del template Siemens S7

2 | Template Modbus e Siemens S7

Nella colonna *NOTE* della tabella vengono dati i fattori di scala per cui moltiplicare il valore effettivo della variabile prima di visualizzarla all'utente. Alla dicitura "3 decimali" ad esempio, corrisponde un fattore di scala pari a 0,001 (1 decimali = 0,1, 2 decimali = 0,01). Come per il template Modbus, il fattore di scala sarà inserito all'interno della colonna *Scale*.

ID	Variable name *	Section name	Area *	Data Block Number	Address *	Bit	Data type *	Access type (default: Read)	Measurement unit	Scale	Offset	Minimum	Maximum
1	WORD ALLARMI 01		Data Block	70	0		Int						
2	WORD ALLARMI 02		Data Block	70	2		Int						
3	WORD ALLARMI 03		Data Block	70	4		Int						
4	WORD ALLARMI 04		Data Block	70	6		Int						
5	WORD ALLARMI 05		Data Block	70	8		Int						
6	WORD ALLARMI 06		Data Block	70	10		Int						
7	WORD ALLARMI 07		Data Block	70	12		Int						
8	WORD ALLARMI 08		Data Block	70	14		Int						
9	WORD ALLARMI 09		Data Block	70	16		Int						
10	WORD ALLARMI 10		Data Block	70	18		Int						
11	PRESSIONE VUOTO #1		Data Block	70	24		DInt			0,001			
12	PRESSIONE VUOTO #2		Data Block	70	28		DInt			0,001			
13	PRESSIONE VUOTO #3		Data Block	70	32		DInt			0,001			
14	LIVELLO CAMERA DI FALSH #1		Data Block	70	36		DInt			0,1			
15	LIVELLO CAMERA DI FALSH #2		Data Block	70	40		DInt			0,1			
16	LIVELLO CAMERA DI FALSH #3		Data Block	70	44		DInt			0,1			
17	TEMPERATURA CAMERA DI FLASH #1		Data Block	70	48		DInt			0,1			
18	TEMPERATURA CAMERA DI FLASH #2		Data Block	70	52		DInt			0,1			
19	TEMPERATURA CAMERA DI FLASH #3		Data Block	70	56		DInt			0,1			
20	TEMPERATURA RICIROCOLO #1		Data Block	70	60		DInt			0,1			
21	TEMPERATURA RICIROCOLO #2		Data Block	70	64		DInt			0,1			
22	TEMPERATURA RICIROCOLO #3		Data Block	70	68		DInt			0,1			
23	TEMPERATURA VAPORE #1		Data Block	70	72		DInt			0,1			
24	TEMPERATURA VAPORE #2		Data Block	70	76		DInt			0,1			
25	TEMPERATURA VAPORE #3		Data Block	70	80		DInt			0,1			
26	TEMPERATURA DISTILLATO #1		Data Block	70	84		DInt			0,1			
27	TEMPERATURA DISTILLATO #2		Data Block	70	88		DInt			0,1			
28	TEMPERATURA DISTILLATO #3		Data Block	70	92		DInt			0,1			
29	TEMPERATURA SERBATOIO DISTILLATO #1		Data Block	70	96		DInt			0,1			
30	TEMPERATURA SERBATOIO DISTILLATO #2		Data Block	70	100		DInt			0,1			
31	TEMPERATURA SERBATOIO DISTILLATO #3		Data Block	70	104		DInt			0,1			
32	TEMPERATURA SERBATOIO DEL VUOTO #1		Data Block	70	108		DInt			0,1			
33	TEMPERATURA SERBATOIO DEL VUOTO #2		Data Block	70	112		DInt			0,1			
34	TEMPERATURA SERBATOIO DEL VUOTO #3		Data Block	70	116		DInt			0,1			

Figura 26 – Esempio di compilazione del template Siemens S7: fattori di scala

È Interessante soffermarci sull'indirizzo di un dato di tipo Boolean:

DB70, DBX190.1

In questo caso, per identificare univocamente la posizione del dato in memoria, è necessario specificare il numero del bit che contiene il dato. Questa informazione è data dalla cifra dopo il punto che, per ovvie ragioni, potrà assumere valori che vanno da 0 a 7 (anche qui, l'enumerazione dei bit parte dallo 0). Tale valore sarà inserito all'interno della colonna *Bit* del template Siemens S7.

ID	Variable name *	Section name	Area *	Data Block Number	Address *	Bit	Data type *	Access type (default: Read)	Measurement unit	Scale
89	LIVELLO MINIMO SERBATOIO DEL VUOTO #1		Data Block	70	190	1	Bool			

Figura 27 - Esempio di inserimento di una variabile booleana all'interno del template Siemens S7

Riassumendo, il seguente indirizzo si riferisce alla variabile *LIVELLO MINIMO SERBATOIO DEL VUOTO #1* contenuta nell'area *Data Block Globali* e rappresentata da un dato di dimensione pari a 1 bit (DBX) contenuto all'interno del *Data Block n° 70* (DB70) e precisamente nel 2° bit del 191° *Data byte* (190.1).

Conoscere la dimensione del dato in questo caso, è una condizione sufficiente per identificare il tipo di dato poiché tra i data type contenuti all'interno di un PLC, il Boolean è l'unico ad avere dimensione pari a 1 bit. Tuttavia, la colonna *TIPO* fornisce ancora indicazioni a riguardo (Bool).

2 | Template Modbus e Siemens S7

Le variabili così elencate all'interno del file Excel compariranno nell'interfaccia dell'IOT SCADA Server una volta caricato il file .xmod al suo interno e configurato correttamente il PLC con il quale si intende comunicare.

The screenshot displays the 'Monitoraggio impianto' (Plant Monitoring) interface. The browser address bar shows '192.168.1.29/devices/devices.zul'. The interface includes a navigation menu on the left with options like 'Impianto', 'Allarmi', 'Storico allarmi', 'Report', 'Documenti', 'Preferiti', and 'Configurazione'. The main area shows the configuration for a 'PLC' device, specifically a 'Siemens - evaporatore 1'. A table lists various variables with columns for 'Misura', 'Valore', 'Min', 'Range', and 'Max'. The variables include 'VALVOLA SCARICO SFIATI', 'VALVOLA SELEZIONE SERBATOIO CONCENTRATO', 'VALVOLA SELEZIONE SERBATOIO PRODOTTO DA TRATTARE', and a series of 'WORD ALLARMI' and 'WORD USCITE DIGITALI'.

Misura	Valore	Min	Range	Max
VALVOLA SCARICO SFIATI	---	---	---	---
VALVOLA SELEZIONE SERBATOIO CONCENTRATO	---	---	---	---
VALVOLA SELEZIONE SERBATOIO PRODOTTO DA TRATTARE	---	---	---	---
WORD ALLARMI 01	---	---	---	---
WORD ALLARMI 02	---	---	---	---
WORD ALLARMI 03	---	---	---	---
WORD ALLARMI 04	---	---	---	---
WORD ALLARMI 05	---	---	---	---
WORD ALLARMI 06	---	---	---	---
WORD ALLARMI 07	---	---	---	---
WORD ALLARMI 08	---	---	---	---
WORD ALLARMI 09	---	---	---	---
WORD ALLARMI 10	---	---	---	---
WORD INGRESSI DIGITALI 01	---	---	---	---
WORD INGRESSI DIGITALI 02	---	---	---	---
WORD USCITE DIGITALI 01	---	---	---	---
WORD USCITE DIGITALI 02	---	---	---	---
WORD USCITE DIGITALI 03	---	---	---	---
WORD USCITE DIGITALI 04	---	---	---	---

Data e ora del server: 11/01/17 12.32

IoTSCADA Powered by Alleanzia

Figura 28 – Interfaccia dell'Iot SCADA Server: visualizzazione delle variabili di un PLC Siemens a configurazione avvenuta

3 L'XPANGO Driver Editor

Il processo di creazione del driver si conclude caricando il file Excel appena compilato sulla pagina <http://cloud.alleantia.com/xmod/convert.zul>.

Facendo click sul tasto *Upload File Excel* si aprirà una finestra di Windows attraverso la quale è possibile selezionare il file Excel del Template compilato. Quindi, una volta raggiunta la cartella che contiene il file Excel basta fare click su *Apri* per caricarlo.

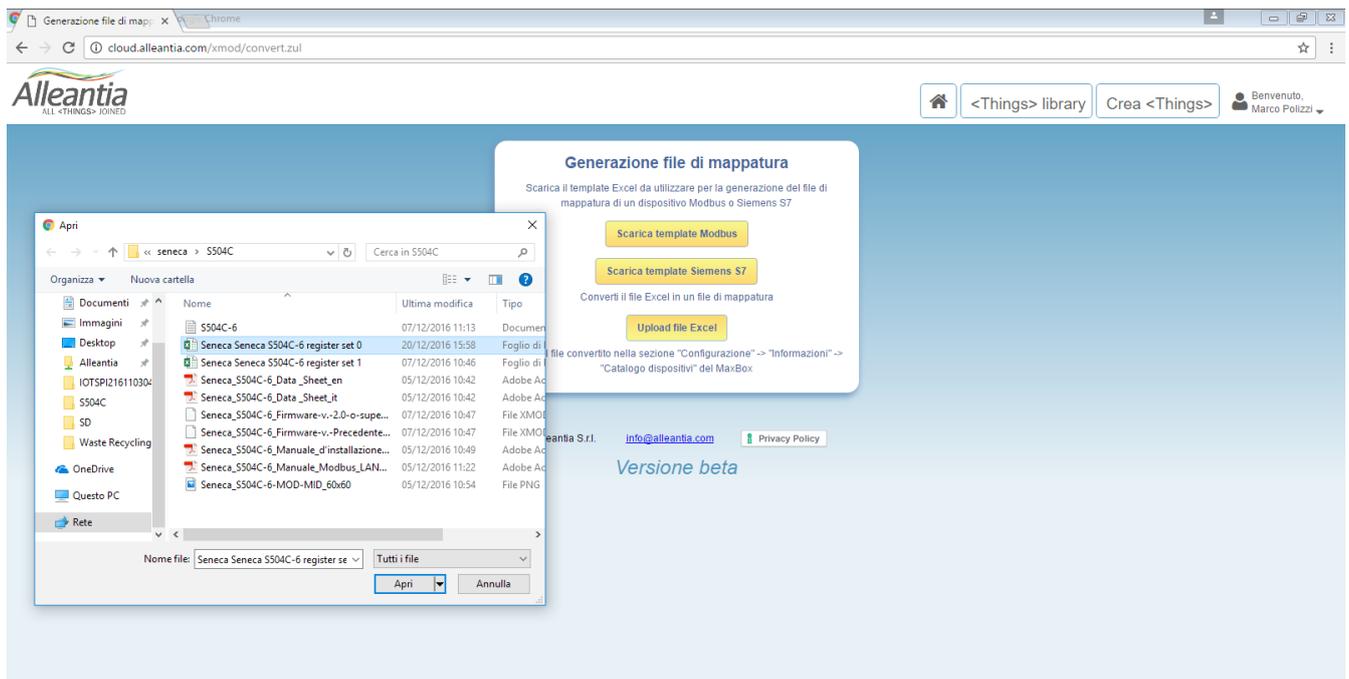


Figura 29 – XPANGO Driver Editor: caricamento del template compilato in formato Excel per convertirlo in un driver XPANGO

A questo punto l'XPANGO Driver Editor, il programma di mappatura appositamente sviluppato da Alleantia, convertirà il file .xlsx in un file .xmod. La procedura è identica sia per il template Siemens S7 che per il Template Modbus.

Se il processo di mappatura è andato a buon fine partirà in automatico il download del file .xmod del dispositivo.

3 L'XPANGO Driver Editor

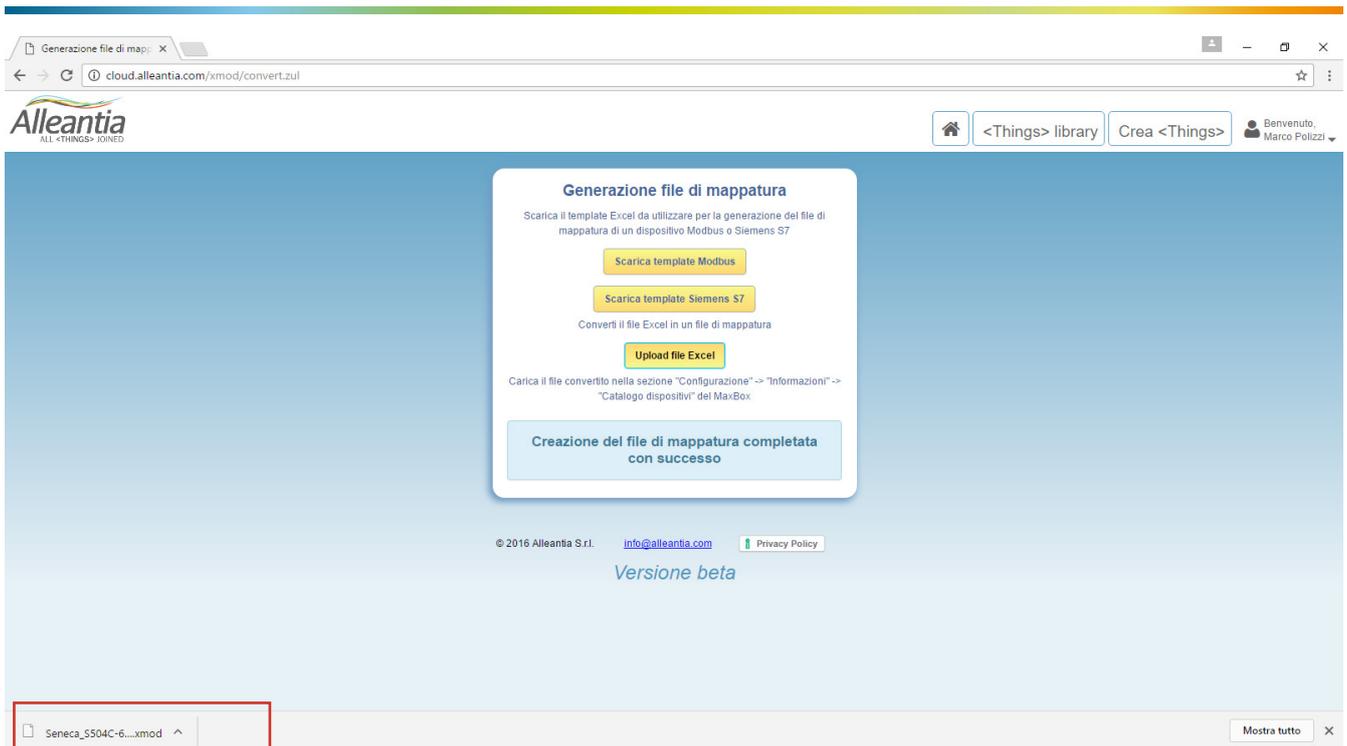


Figura 30 – XPANGO Driver Editor: completamento della mappatura avvenuta con successo e download automatico del driver XPANGO

Per riuscire a comunicare con il dispositivo è necessario che il suo driver XPANGO, non trovandosi nella *Library of Things* online di Alleantia, si trovi almeno su quella propria dell'IoT SCADA Server a disposizione.

Quindi bisogna innanzitutto effettuare l'upload del file .xmod all'interno dell'IoT SCADA Server (*Vedi Manuale d'installazione e uso IoT SCADA Server - par. 5.5.1*) e successivamente configurare correttamente il dispositivo impostando correttamente i parametri di comunicazione (*Vedi par. 5.2.2.1 del Manuale d'installazione e uso IoT SCADA Server*).

4 | Indicazioni e suggerimenti

Qualora la compilazione del template non sia stata eseguita in maniera corretta l'XPANGO Driver Editor non riuscirà a completare il processo di mappatura. Sulla pagina <http://cloud.alleantia.com/xmod/convert.zul> apparirà il messaggio "il file caricato contiene degli errori" e si avvierà in automatico il download del file errors.xlsx.

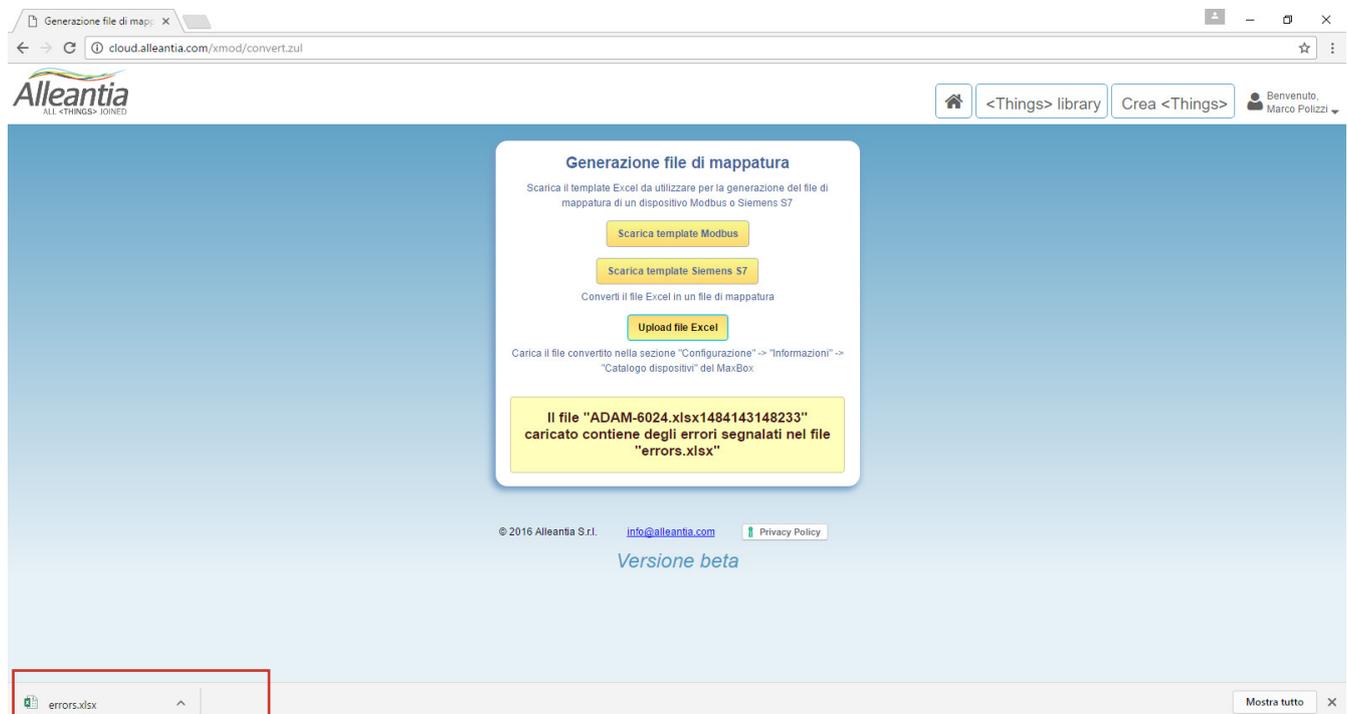


Figura 31 – XPANGO Driver Editor: mappatura del file non avvenuta con successo per la presenza di errori nel template Modbus e download automatico del file errors.xlsx

Aperto il foglio ci accorgeremo che le celle contenenti degli errori saranno evidenziate e conterranno delle note utili alla risoluzione del problema. Tale sistema da un lato favorisce la rapida individuazione degli errori (cosa tutt'altro che semplice se si pensa alla grande quantità di righe di cui è composta una mappa Modbus!) mentre dall'altro, "istruisce" l'utilizzatore fornendogli informazioni sulla sorgente degli errori stessi. Note le cause degli errori non sarà difficile mettere in atto gli accorgimenti giusti per eliminarli.

I dispositivi IoT SCADA Server sono in grado di leggere, e quindi di visualizzare nell'interfaccia, un certo numero di variabili e tale numero dipende dalla licenza caricata sul dispositivo stesso. Pertanto si consiglia di compilare il template inserendo solamente le variabili di stretto interesse oppure di disattivare quelle momentaneamente non necessarie settando la cella *Preselected* su FALSE.

È bene organizzare le variabili in sezioni così da rendere l'interfaccia dell'IoT SCADA Server più leggibile, specialmente con un numero molto elevato di variabili. Tale procedura va svolta inevitabilmente durante la compilazione del file Excel poiché per eseguirla successivamente è necessario ricompilarlo, quindi generare un nuovo driver XPANGO e ricaricarlo sul dispositivo.

Per la compilazione dei template non vanno usati strumenti office diversi da quelli prodotti da Microsoft. Ad esempio, delle incompatibilità che non permettevano di portare a termine correttamente la procedura, sono state riscontrate con l'utilizzo del software Apache OpenOffice.

I driver XPANGO per i PLC Siemens S7 creati dall'XPANGO Driver Editor possono risultare incompatibili, e quindi non configurabili, su dispositivi IoT SCADA server con versione software inferiore alla 3.4.1.

5 Contatti

Alleantia s.r.l.

www.alleantia.com

Sede legale: Via Tosco Romagnola, 136 56025 Pontedera (PI)

Sede operativa: Via Umberto Forti, 24/14 56121 Pisa

Partita IVA / Cod. fiscale: IT 02011550502

info@alleantia.com

