

Note generali sulla comunicazione (su rete Ethernet) tra un modulo logico **ZELIO Logic** (cui deve essere affiancabile l'estensione Ethernet **SR3NET01**) ed una piattaforma SCADA basata su **Vijeo Citect**

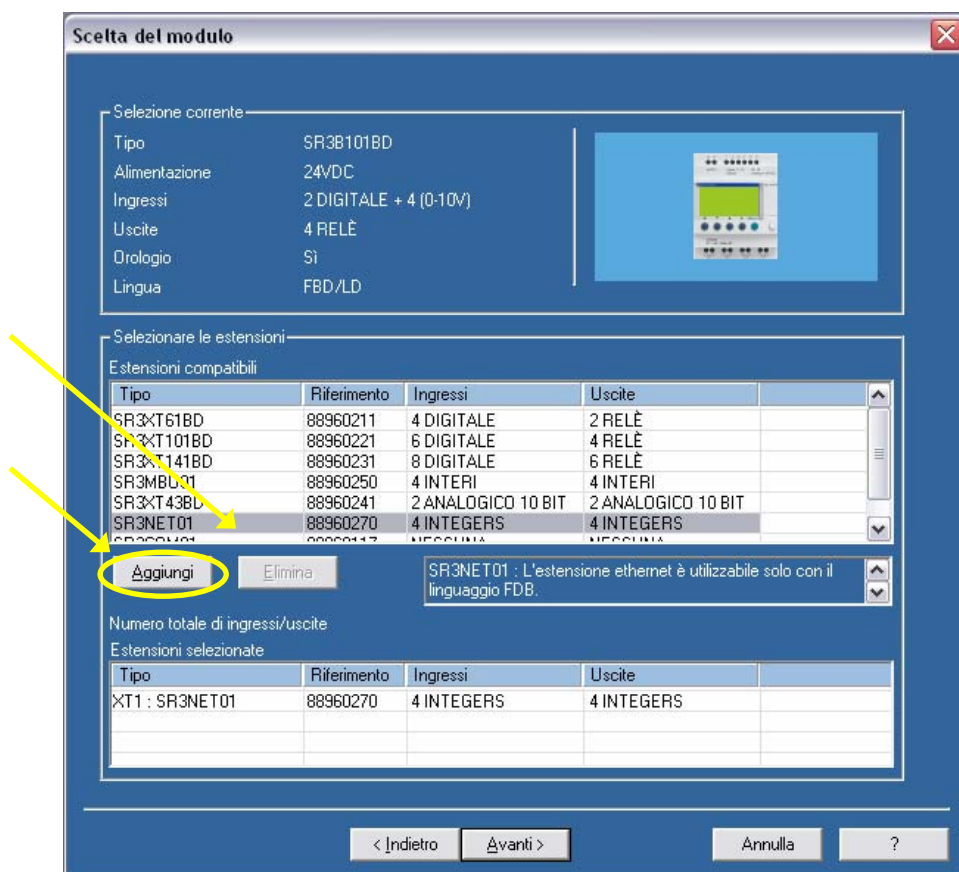


[una copia gratuita di Vijeo Citect può essere richiesta all'indirizzo:
<http://www.schneiderelectric.it/www/internet/question.nsf/q1?openform>]

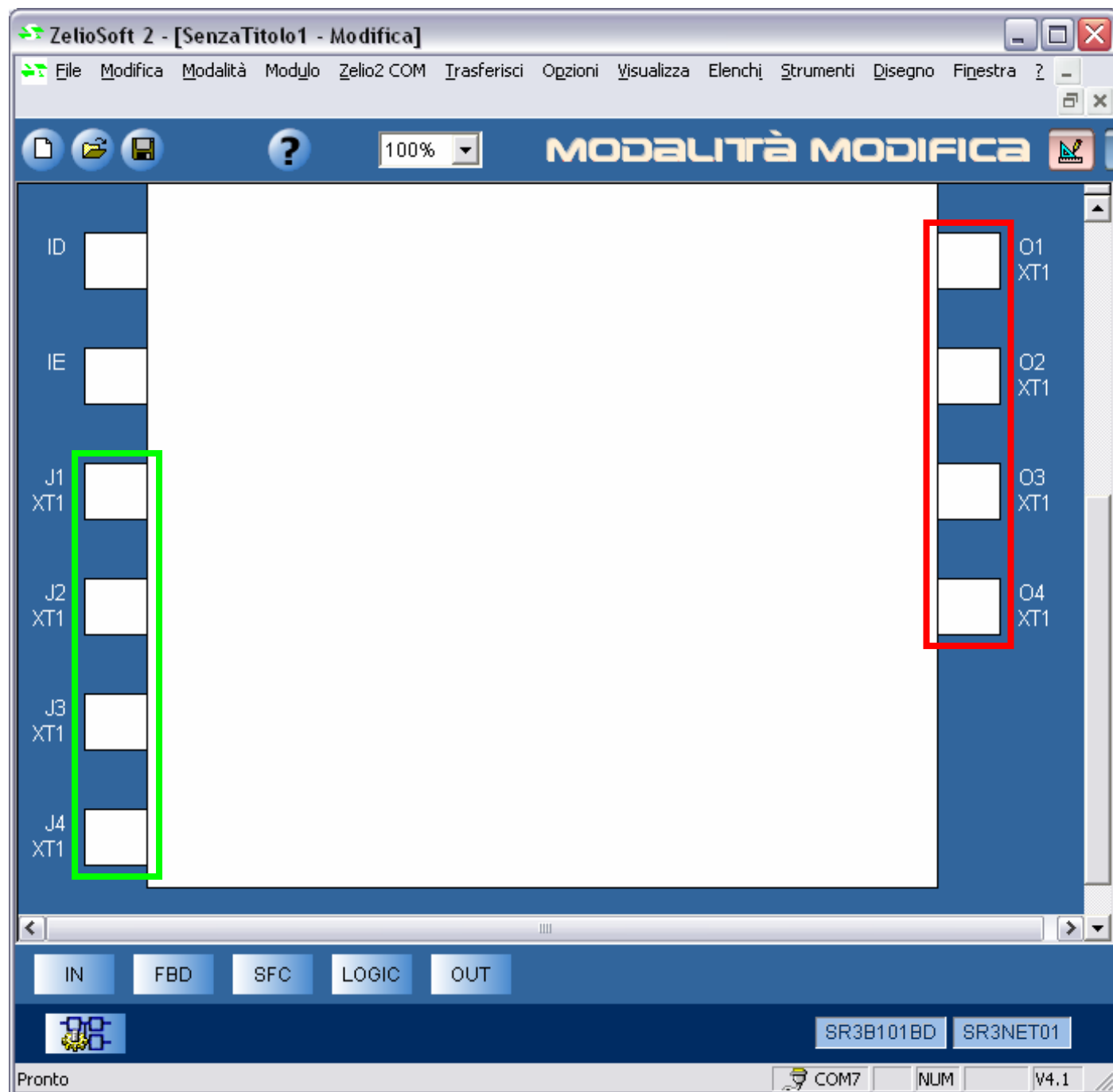
Lato ZELIO Logic:

Assunto che l'aggiunta del modulo SR3NET01 vincoli la programmazione del modulo logico cui viene affiancato al solo linguaggio FBD, e non più anche in LD, osserviamo come poter telegestire dallo SCADA **Vijeo Citect** lo **ZELIO Logic** attraverso gli ingressi/uscite Ethernet di cui disporremo in **Zelio Soft 2**.

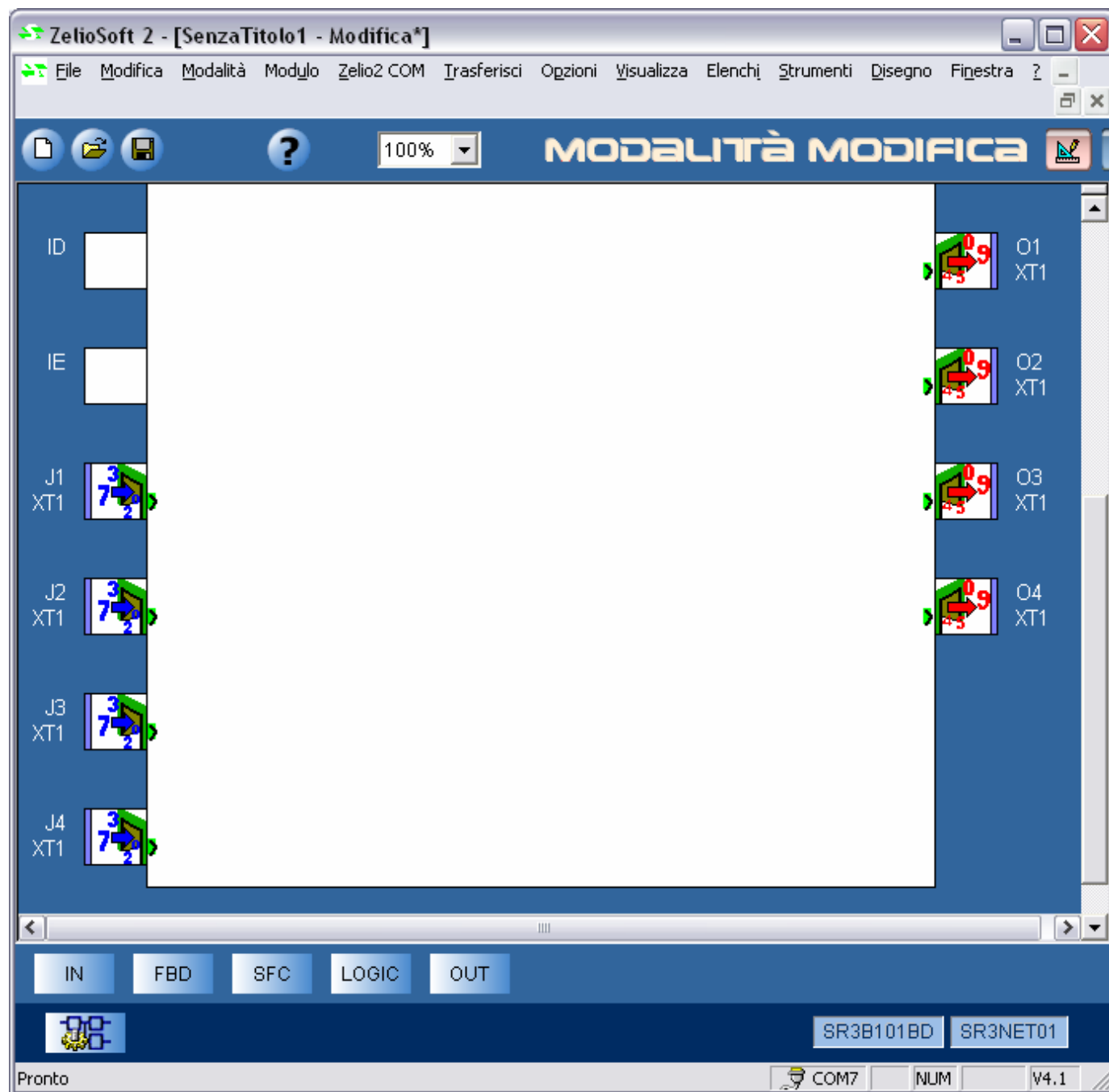
Per prima cosa va aggiunto il modulo SR3NET01 alla configurazione hardware scelta:



A questo punto, nell'interfaccia di programmazione, saranno disponibili **4 nuovi ingressi** e **4 nuove uscite**, oltre quelli corrispondenti agli I/O reali (fisici) del modulo logico:



Trattandosi di I/O da 16 bit ciascuno, per usufruirne siamo obbligati a scegliere gli appropriati blocchi di I/O corrispondenti agli INTERI (e codificati con 16 bit):



Sarà quindi possibile **scrivere** da remoto un massimo di 64 valori digitali sullo ZELIO (attraverso gli ingressi da **J1** a **J4**) e **leggere**, dalle 4 WORD di uscita **O1**, **O2**, **O3** ed **O4**, lo stato di altrettanti bit interni.

Nel caso servisse scrivere o leggere da remoto dei valori analogici interi, codificati con al più 16 bit, è evidente il limite relativo alla scrittura, ed alla lettura, di solo 4 valori in entrambi i casi.

Le WORD contrassegnate dalle etichette **J1**, **J2**, **J3** e **J4** ed **O1**, **O2**, **O3** ed **O4** sono accessibili (ad esempio dall'**IO Scanning** operato da un PLC TSX **master**, su protocollo **UNITE**) come normali WORD di memoria allocate ai seguenti indirizzi:

J1=%MW16	J2=%MW17	J3=%MW18	J4=%MW19
O1=%MW20	O2=%MW21	O3=%MW22	O4=%MW23

Per quanto riguarda il protocollo di comunicazione tra **Vijeo Citect** e lo **Zelio Logic**, l'unico che ci è possibile scegliere è il **Modnet** (Modbus/TCP) che prevede il seguente tipo di indirizzamento:

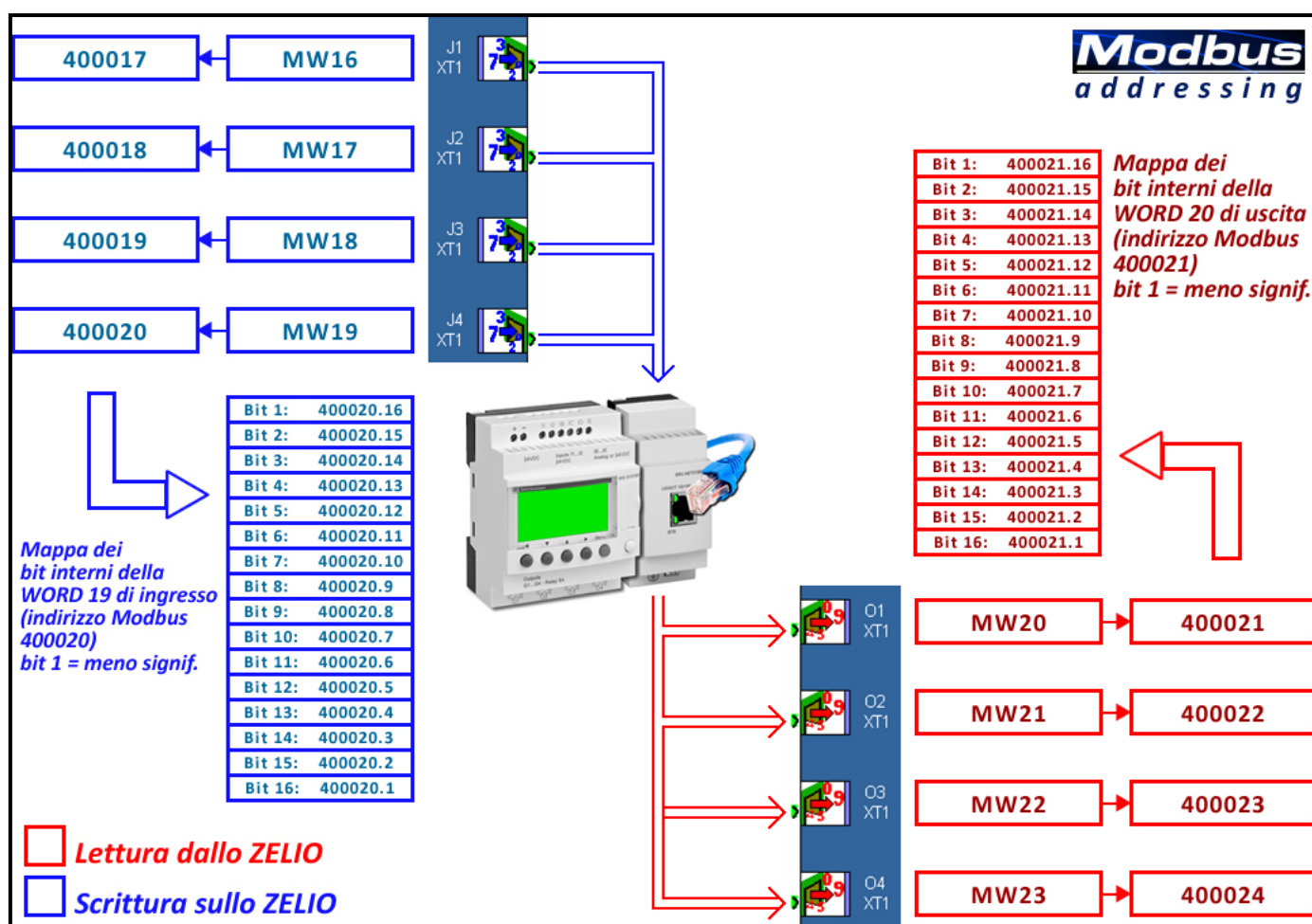
la locazione **%MW16** in **Modnet** sarà raggiungibile all'indirizzo **400017** - e non al **400016** come ci aspetteremmo - (in quanto **40001** è il primo indirizzo **Modnet** disponibile come indirizzo della prima tra le WORD allocabili, ovvero della **%MW0**).

Ci si aspetta di conseguenza che il bit interno meno significativo della WORD in questione (il bit 0) sia indirizzabile nel seguente modo:

%MW16:0 ⇔ 400017.1

in realtà, a beneficio della confusione, l'ordine di indirizzamento dei bit interni secondo questo protocollo è invertito, per cui, volendo indirizzare il bit meno significativo della WORD 16 dello Zelio si utilizzerà l'indirizzo **400017.16**, e al precedente indirizzo, **400017.1**, corrisponderà invece il bit più significativo della WORD 16 (ovvero il sedicesimo: **%MW0:15**).

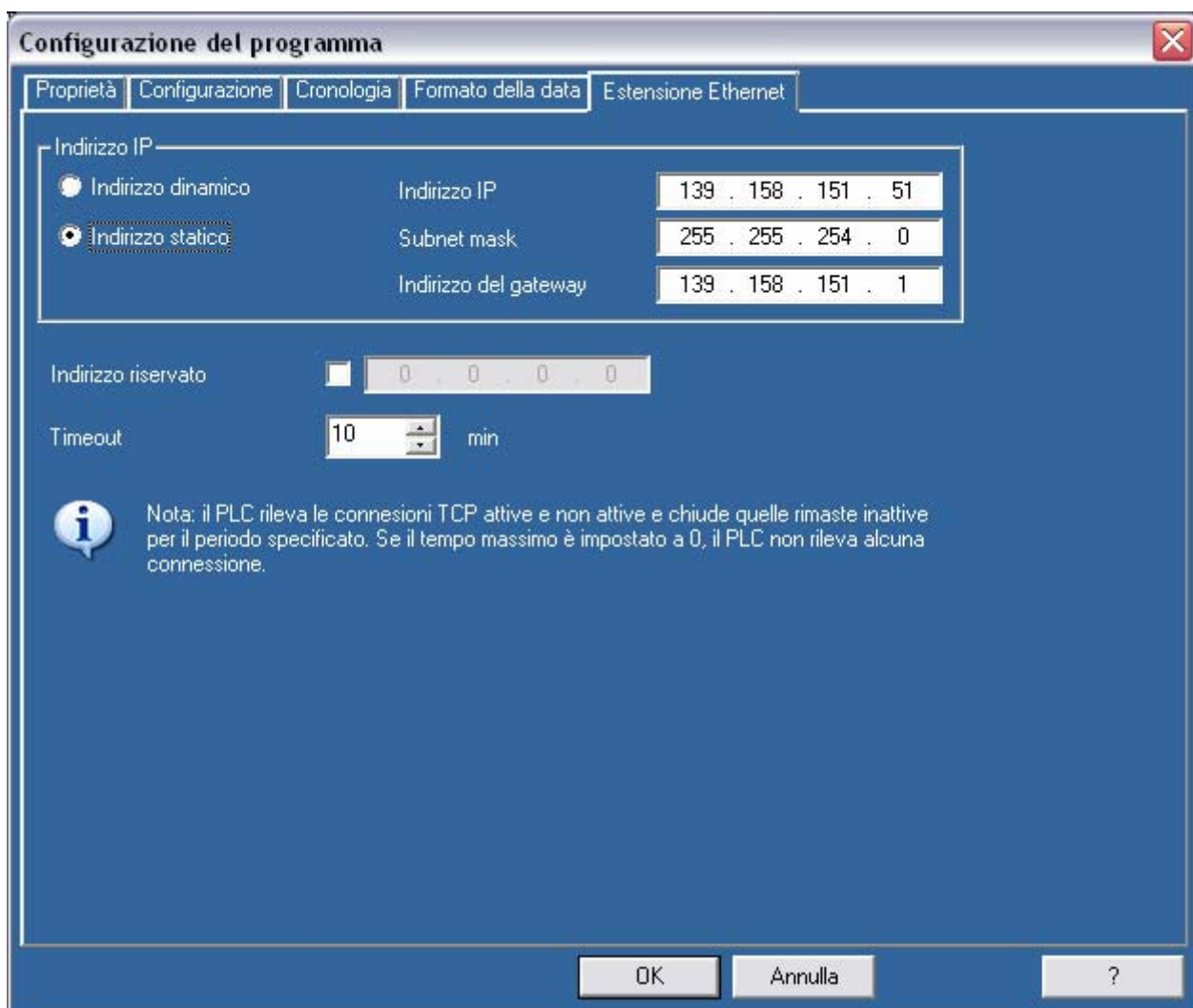
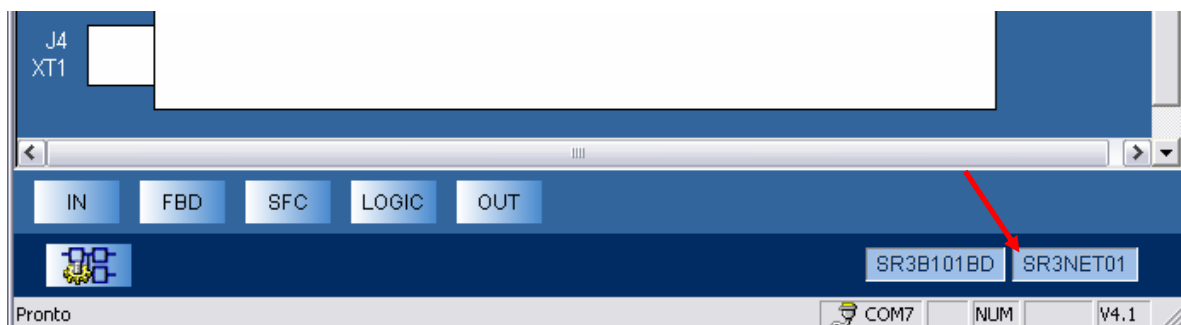
Quanto detto è riassunto nel seguente schema, dove è rappresentata, a scopo esemplificativo, la traduzione degli indirizzi delle due WORD 19 e 20:



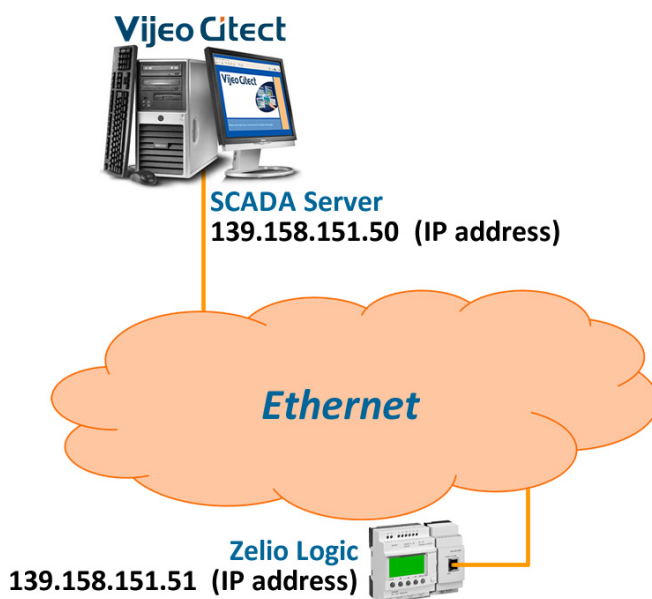
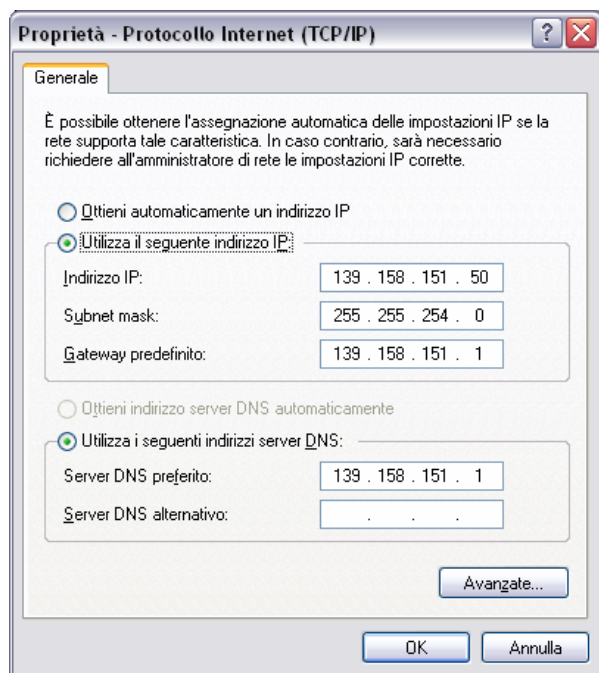
C'è da dire però che la versione 7 di **Vijeo Citect** ci mette a disposizione diverse varianti di **Modnet**, una delle quali, il **Modnet30** (impostato di default se configureremo come *device* un PLC Premium), ci consente di specificare gli indirizzi dei registri secondo la sintassi cui siamo (o dovremmo essere) più abituati: **%MW16**, **%MW20**, **%MW20.1** ecc. Ma vedremo tutto più in dettaglio nelle prossime pagine...

A questo punto vanno opportunamente settate le impostazioni di rete del modulo Ethernet SR3NET01 e del dispositivo di rete del PC cui si intende connetterlo. Qualora non vi siano particolari esigenze è preferibile impostare staticamente gli indirizzi, come raffigurato, ad esempio, nelle seguenti immagini:

In **Zelio Soft 2** si accede velocemente alla scheda di configurazione del modulo attraverso un doppio click sul nome dell'estensione Ethernet (come indicato dalla freccia):



Compatibilmente con i valori impostati per il modulo Ethernet nella schermata precedente di **Zelio Soft 2**, per l'interfaccia di rete del PC potrebbero venire settati i seguenti valori:



Di conseguenza la configurazione della rete sarebbe, banalmente, quella rappresentata nella vignetta qui sopra.

Verrà di seguito proposto un programma base per lo **Zelio Logic**, senza alcuna particolare logica di funzionamento interna, in grado però di:

- inviare in remoto lo stato di tutti gli I/O fisici del dispositivo
- consentire, sempre da remoto, il totale comando del modulo logico (in maniera del tutto equivalente all'attivazione degli ingressi fisici del modulo)

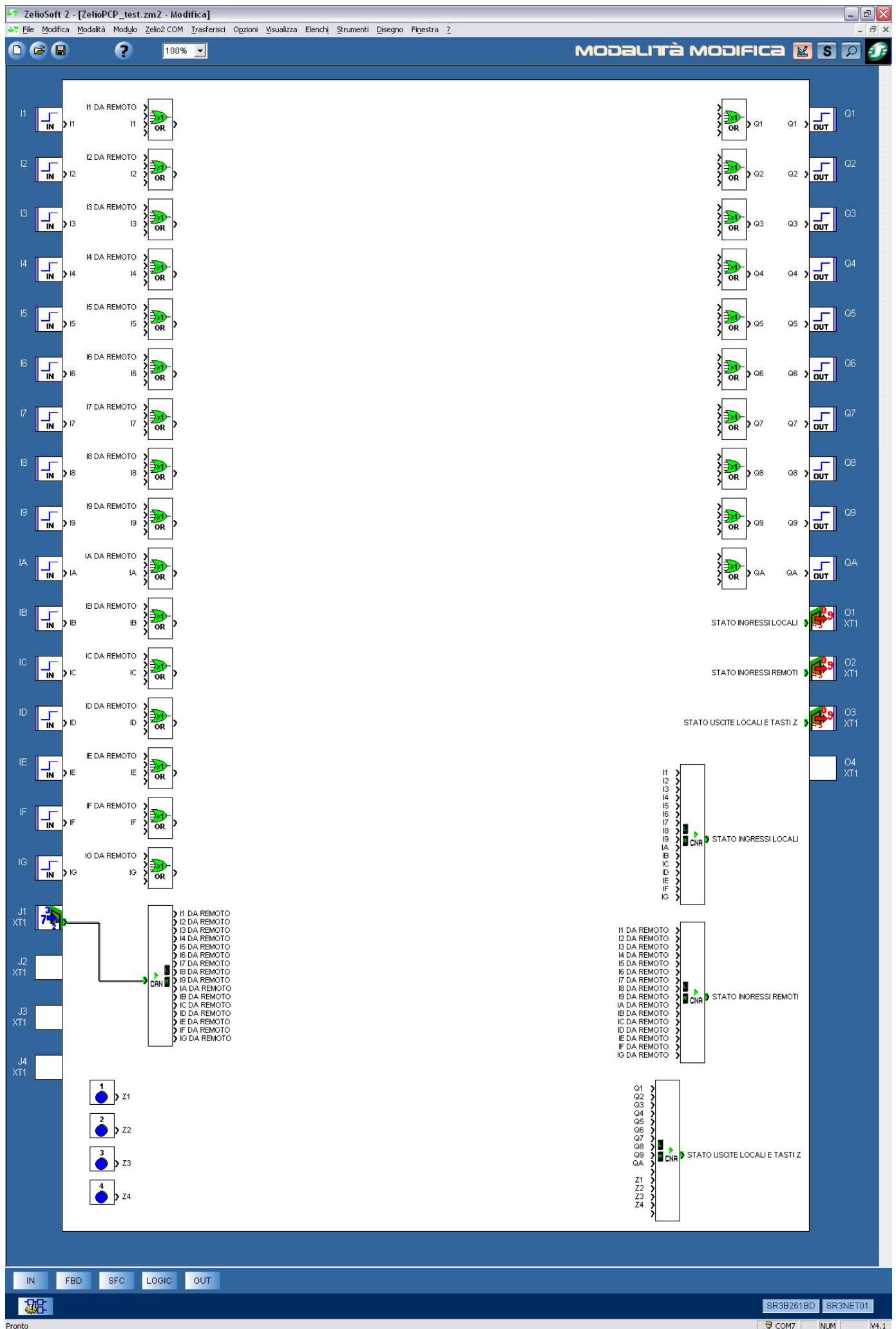
Per il programma, il cui schema è presente nella prossima pagina, è stato considerato il seguente hardware:

Zelio Logic SR3B261BD (16 Ingressi e 10 Uscite) + espansione Ethernet SR3NET01BD

La gran parte dei collegamenti è stata effettuata avvalendosi della modalità "a testo" piuttosto che di quella "a filo" per evitare di rendere lo schema troppo confuso a causa dei collegamenti. In questa modalità l'uscita di un blocco contraddistinta, ad esempio, dall'etichetta **> I1 DA REMOTO** è connessa all'ingresso del blocco (o dei blocchi) che riportano la stessa etichetta in uno dei propri ingressi (**I1 DA REMOTO >**).

Le porte OR che affiancano gli ingressi fisici hanno il seguente scopo: rendere equivalente l'effetto del comando relativo all'ingresso fisico locale al comando proveniente dall'omonimo ingresso remoto.

In termini di sicurezza ciò potrebbe non essere corretto, ma tale funzionalità è stata semplicemente implementata a titolo esemplificativo..



Esempio pratico di comunicazione tra uno **ZELIO Logic SR3B101BD** (con estensione Ethernet **SR3NET01**) ed una pagina di supervisione SCADA progettata con **Vijeo Citect v7.0**

Adesso verrà invece interamente proposto un esempio applicativo riguardante un banalissimo programma creato per uno **Zelio Logic SR3B101BD** allo scopo di renderlo controllabile da remoto da un'altrettanto semplice pagina di supervisione SCADA (ricordiamo che SCADA sta per *Supervisory Control And Data Acquisition*) creata con **Vijeo Citect 7.0**.

Per il programma che verrà proposto è stato stavolta considerato il seguente hardware: **Zelio Logic SR3B101BD** (6 Ingressi e 4 Uscite) + espansione Ethernet **SR3NET01BD**

Il principio di funzionamento del programma è molto semplice:

I1 dà l'impulso ad un *TIMER* che ha un ritardo allo spegnimento pre-impostato (a 60 secondi). E' quindi sufficiente che **I1** sia **alto** un solo istante perché lo sia, per i 60 secondi successivi, anche l'uscita del *TIMER*, ovvero l'uscita fisica **Q1** dello **Zelio Logic**.

Q1 è quindi **alta** fintanto che lo è anche **I1**, e permarrà tale per ancora i 60 secondi successivi al ritorno di **I1** a **0**.

La pressione di **I2** resetta il *TIMER* e ed azzerava immediatamente anche l'uscita **Q1**.

Osservando la logica del programma (pagina successiva) si nota perfettamente come i **bit 01** e **bit 02** della **WORD J1** in ingresso al modulo Ethernet svolgano esattamente (per via delle porte OR) le funzioni degli ingressi fisici locali **I1** ed **I2**.

Il valore intero in ingresso alla **WORD J4** viene invece confrontato con la costante numerica "60" settata nel modulo, e tale valore viene successivamente impostato come nuovo valore di ritardo allo spegnimento del *TIMER* se superiore alla costante di default.

Ciò significa che se da remoto viene inviato il valore 3600, il *TIMER*, quando attivato, disattiverà la propria uscita 60 minuti dopo la disattivazione del proprio ingresso (OR degli ingressi **I1** e **bit1**).

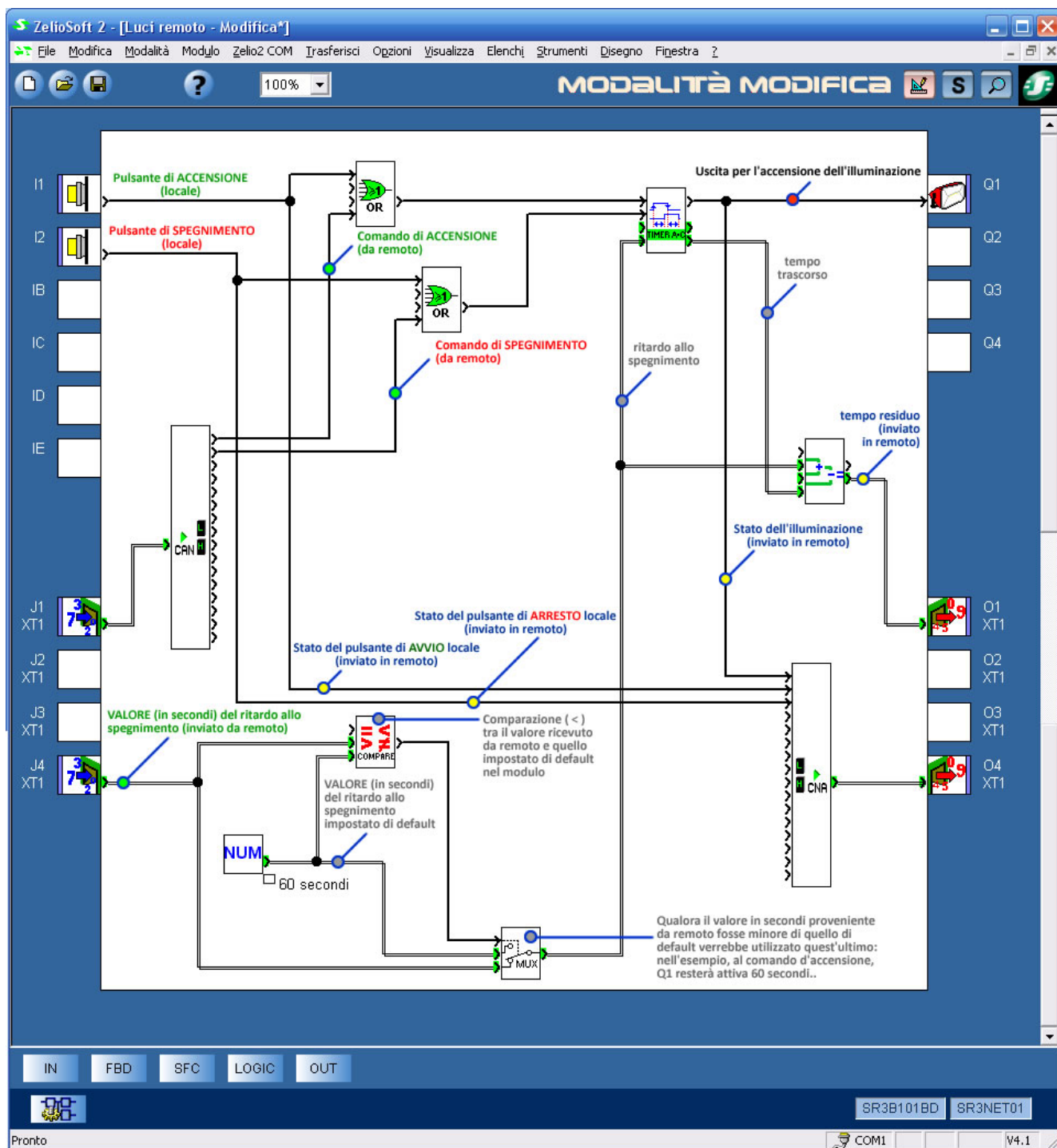
In uscita al modulo Ethernet viene reso disponibile, sulla **WORD O1**, il tempo residuo allo spegnimento dell'uscita **Q1**, calcolato come differenza tra quello impostato e quello trascorso.

La **WORD O4** conterrà invece, rispettivamente, nei suoi primi 3 bit:

1. lo stato attuale dell'uscita **Q1**
2. lo stato attuale dell'ingresso fisico **I1**
3. lo stato attuale dell'ingresso fisico **I2**.

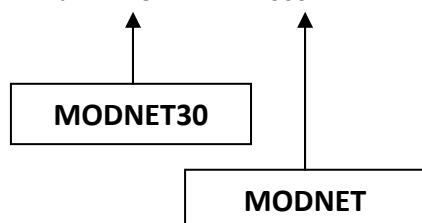
Naturalmente tutte le uscite del modulo logico (reali, da **Q1** a **Q4**, e virtuali, da **O1** ad **O4**) possono solo essere lette e MAI scritte.

Lo schema, commentato, del programma appena descritto è il seguente:



Da quanto detto prima sull'indirizzamento **Modnet** ci aspettiamo, volendoci interfacciare con lo **Zelio Logic** da remoto, le seguenti equivalenze in funzione del protocollo scelto:

J1 bit1	= %MW16.0	= 400017.16	= bit di accensione da remoto
J1 bit2	= %MW16.1	= 400017.15	= bit di spegnimento immediato da remoto
J4	= %MW19	= 400020	= valore impostato da remoto come ritardo di spegnimento del TIMER
O1 bit1	= %MW20.0	= 400021.16	= stato dell'uscita Q1 dello Zelio Logic
O1 bit2	= %MW20.1	= 400021.15	= stato dell'ingresso I1 dello Zelio Logic
O1 bit3	= %MW20.2	= 400021.14	= stato dell'ingresso I2 dello Zelio Logic
O4	= %MW23	= 400024	= valore indicante il numero di secondi residui allo spegnimento dell'uscita Q1

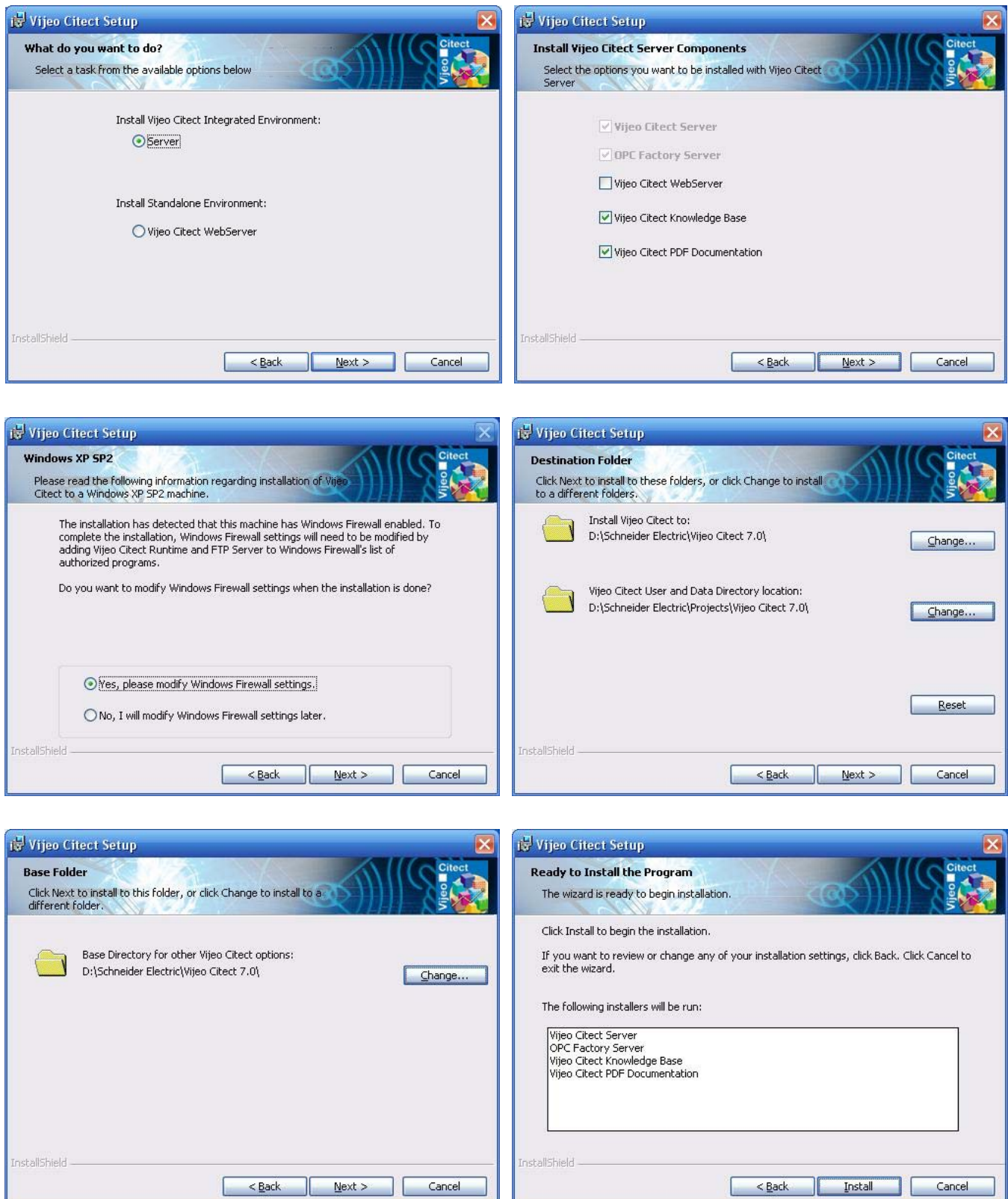


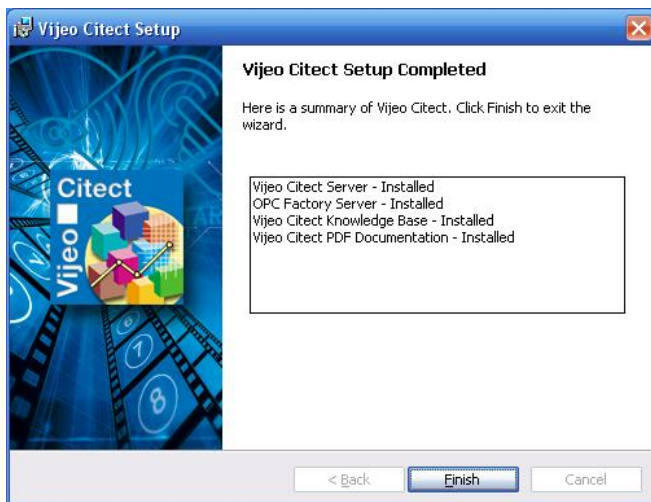
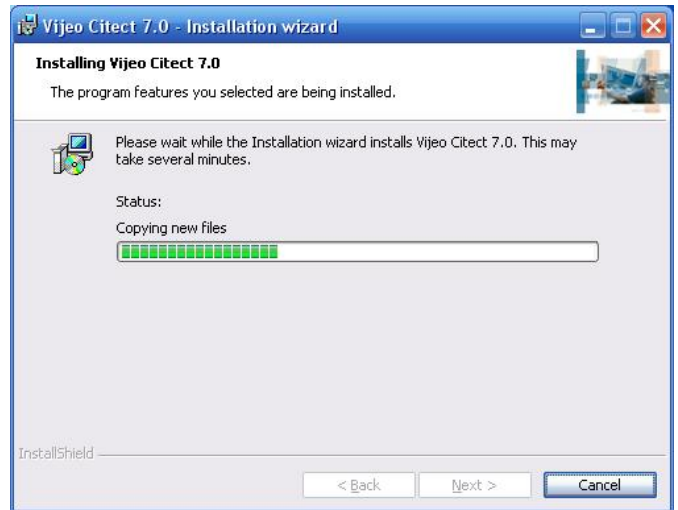
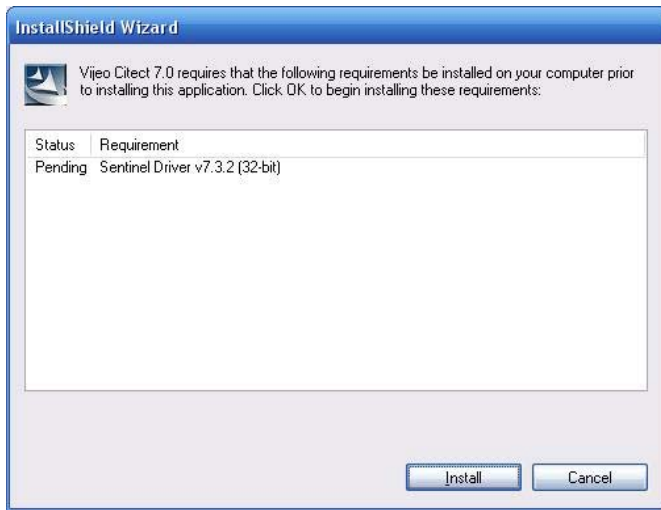
La configurazione del modulo Ethernet e dell'interfaccia di rete del PC può essere fatta esattamente come visto nelle pagine precedenti.

Passiamo adesso alla creazione della pagina di supervisione in Vijeo Citect:

Lato Vijeo Citect:

Le seguenti schermate riassumono brevemente i passi effettuati per l'installazione dell'ambiente di sviluppo **Vijeo Citect 7.0**:





A questo punto l'installazione del software è terminata.

L'ambiente di sviluppo e di supervisione **Vijeo Citect** consta essenzialmente di due parti:



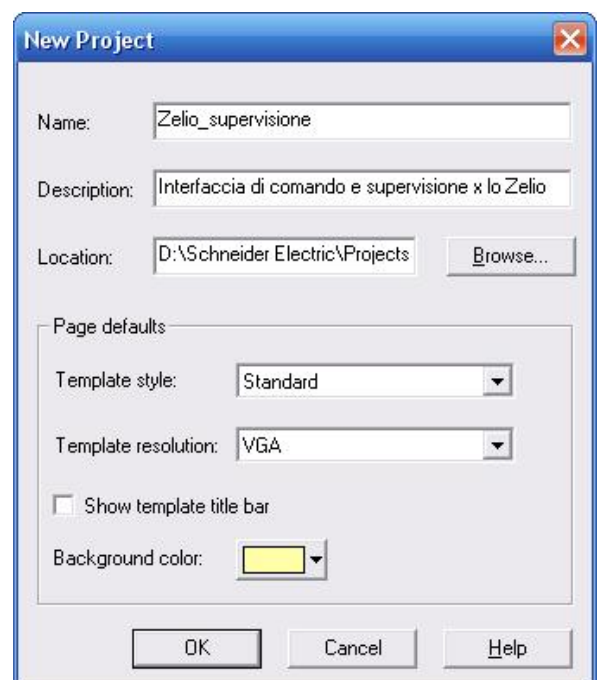
l'**EXPLORER**, col quale vengono costruite e modificate le pagine, creati i simboli grafici, scritti i propri script e le proprie funzioni ed impostato quant'altro necessario alla realizzazione ed al funzionamento dell'intero SCADA



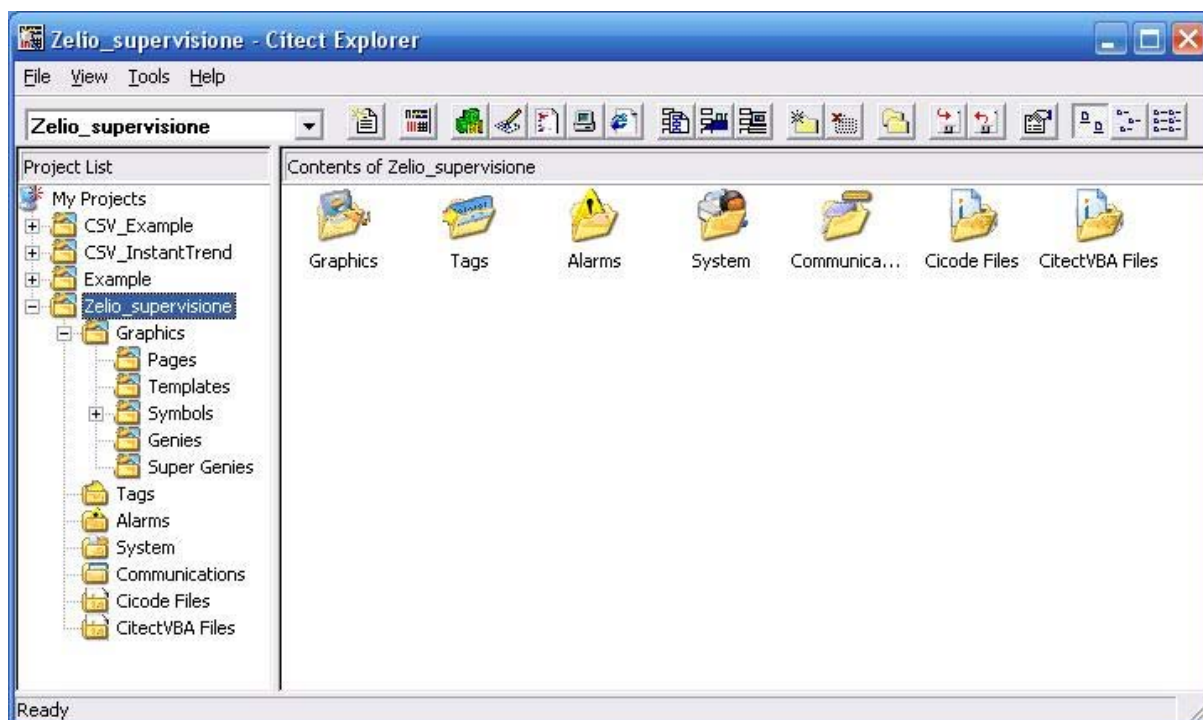
ed il **RUNTIME**, strumento col quale viene lanciata ed eseguita l'interfaccia SCADA realizzata; naturalmente si presume che il computer nel quale verrà lanciato il **RUNTIME** sia correttamente collegato a ciascun dispositivo (**DEVICE**) configurato durante la progettazione delle pagine SCADA

Lanciando quindi **Vijeo Citect Explorer** si potrà procedere alla creazione della nostra pagina di supervisione.

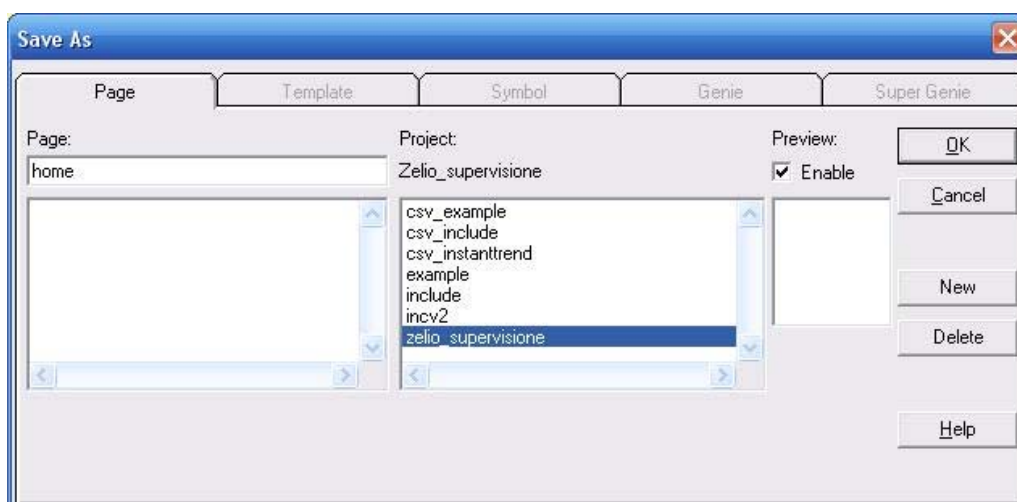
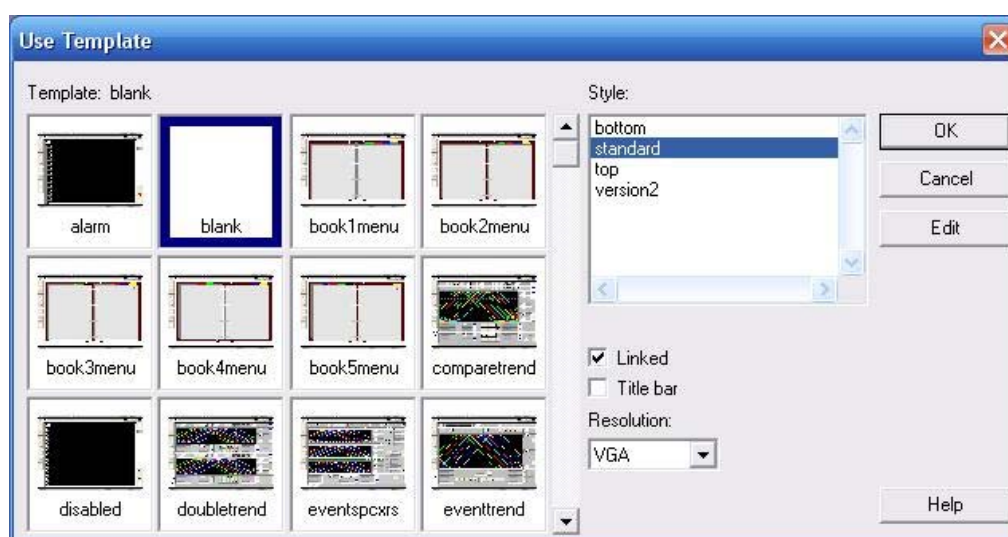
Nella parte sinistra della finestra "Citect Explorer" possiamo osservare l'insieme dei progetti d'esempio già a corredo del programma; a noi interessa crearne uno nuovo, quindi dobbiamo andare su *File* → *New Project...* ed inserire i dati del nostro progetto, come, ad esempio, nell'immagine a fianco:



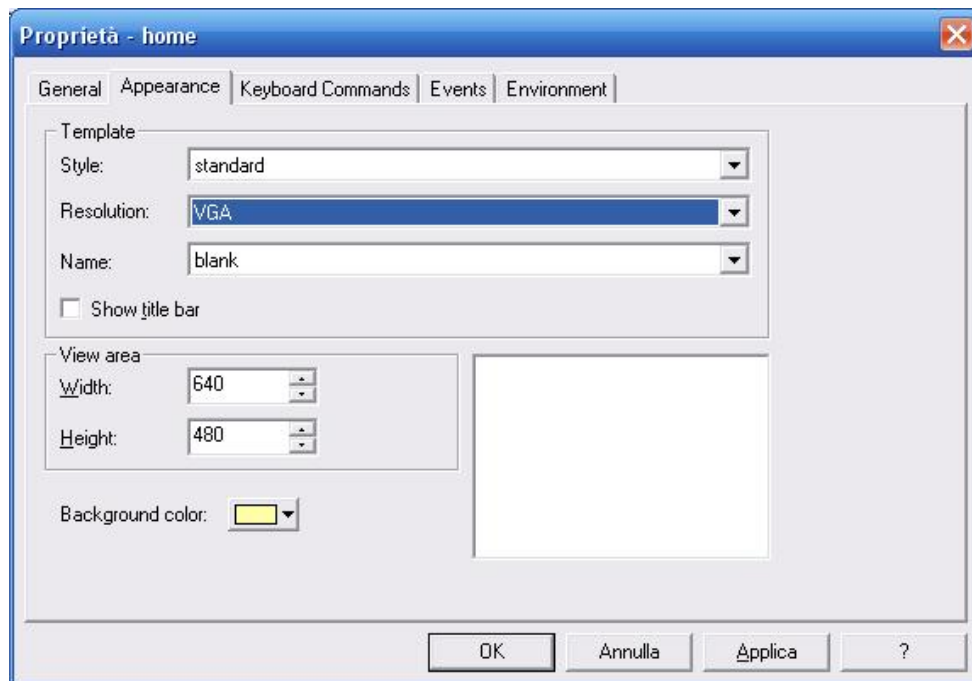
Dando quindi l'OK vedremo il nostro progetto ed al suo interno tutte le sezioni del quale è composto. In questa guida ne verranno considerate solo alcune.



Nella sezione "Pages", all'interno di "Graphics", andremo subito a creare la nostra pagina di supervisione scegliendone il layout e salvandola successivamente con un nome:



Andando quindi sulle *Proprietà* della pagina potremo sempre, in qualsiasi momento, variarne i parametri caratteristici (formato, colore di sfondo ecc.):



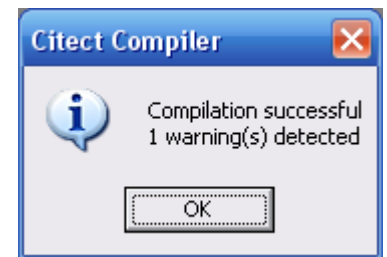
Creata quindi la pagina, prima di procedere, dobbiamo ancora definire con quale *device* dovrà successivamente essere in grado di colloquiare, configurando quindi i parametri del dispositivo remoto (nel nostro caso lo Zelio Logic) nelle sezioni appropriate del progetto.

Per cominciare andiamo sul "Citect Project Editor" e quindi da *Servers* → *Clusters* inseriremo un nome per il Cluster* cui apparterrà il nostro server SCADA aggiungendo (*Add*) il record appena creato



* [un cluster è, molto semplicemente, un gruppo di risorse – nel nostro caso di server – che agisce come un unico sistema garantendo quindi performance migliori. Il nostro cluster sarà costituito da un solo server, ma è bene sapere che Vijeo Citect prevede il funzionamento in rete con altre piattaforme analoghe al fine di realizzare configurazioni ridondanti e quindi di garantire sempre la continuità del servizio]

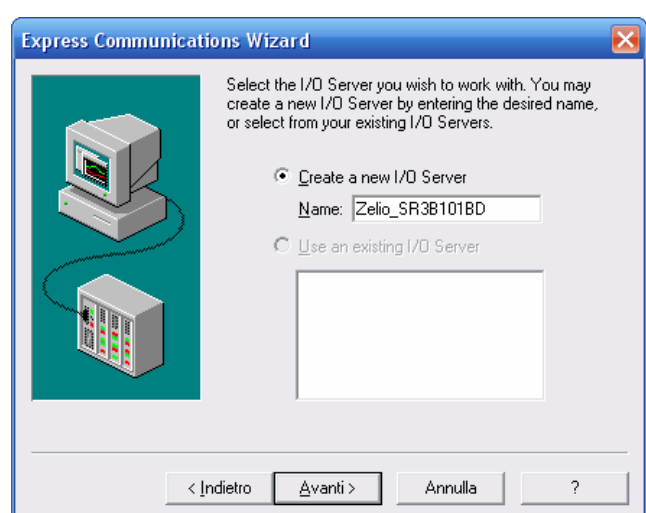
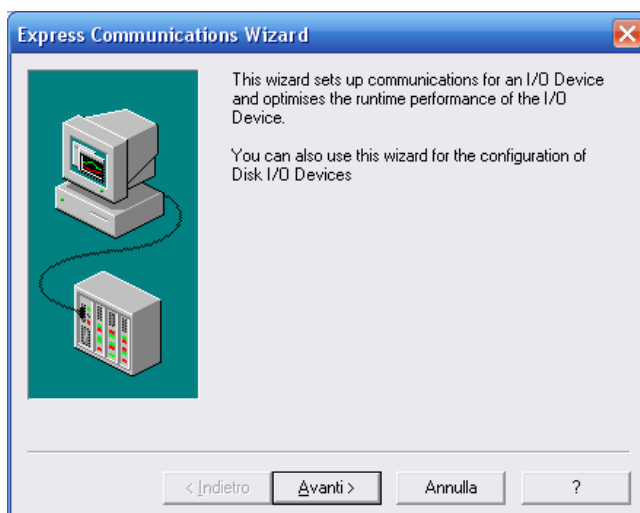
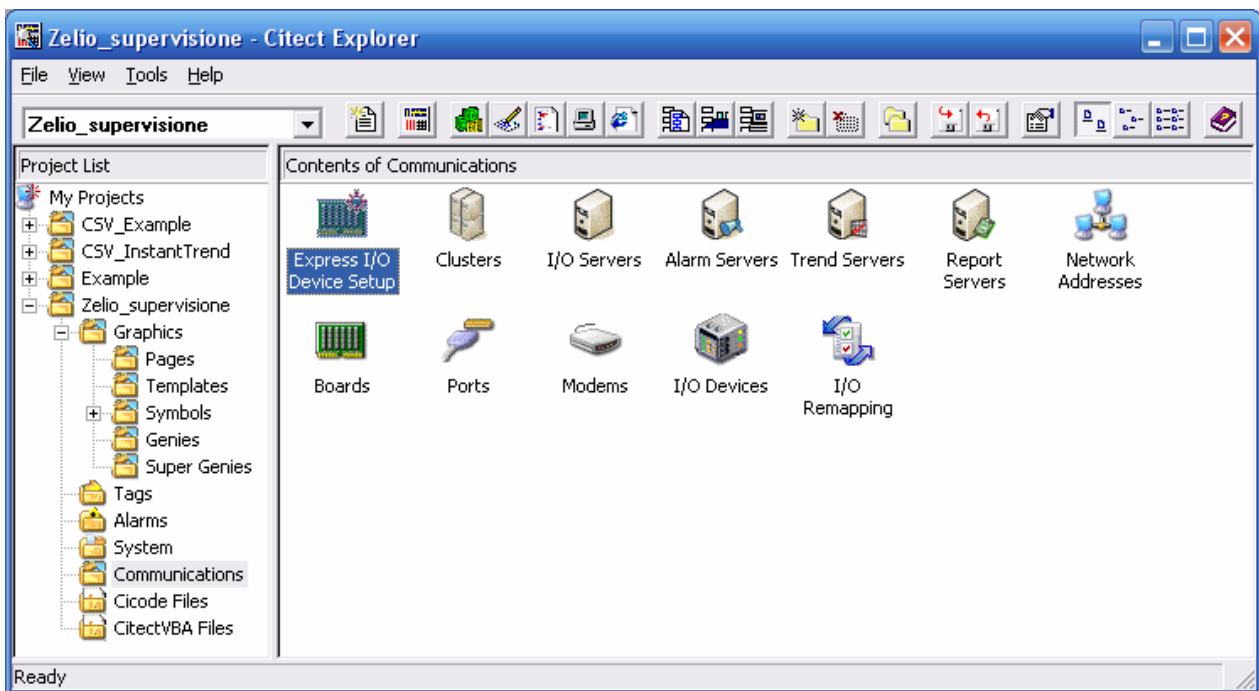
Possiamo adesso **COMPILARE** (Alt+F10, oppure *File* → *Compile*) per la prima volta il nostro progetto, anche se non abbiamo ancora definito alcuna informazione circa lo Zelio ed il protocollo da utilizzare per la comunicazione con esso (il *warning* segnalato è dovuto proprio all'assenza di *device* configurati).



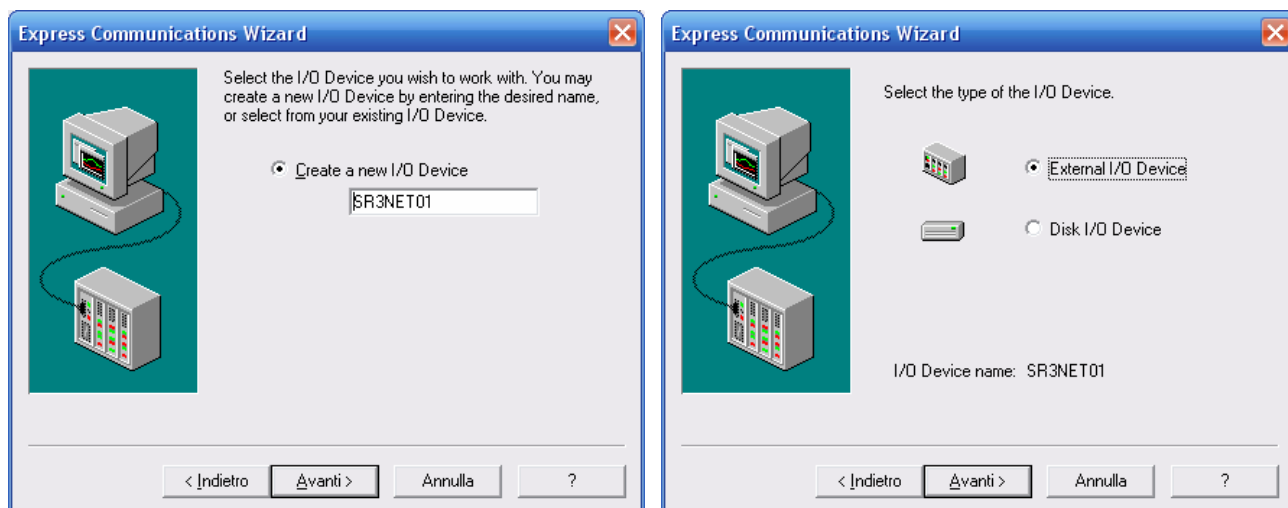
Passiamo quindi adesso a definire

1. le proprietà del dispositivo con cui dovrà comunicare (il nostro *device*)
2. il protocollo di comunicazione da utilizzare
3. il ruolo della nostra piattaforma SCADA

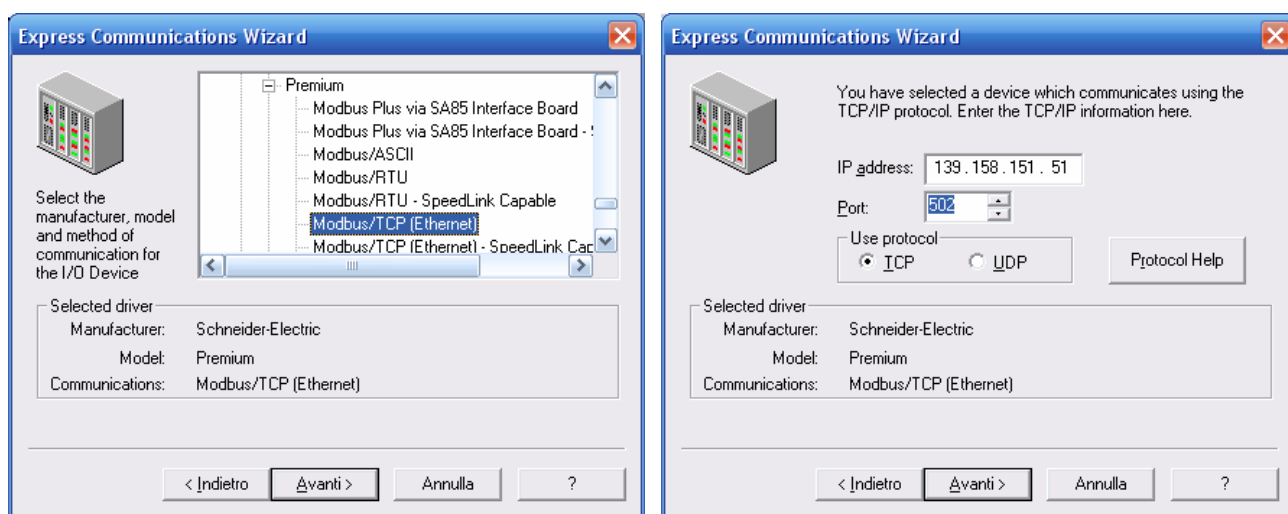
1. per la configurazione del nostro unico *device* passiamo per una *wizard* (configurazione guidata). Andiamo quindi dall'Editor su *Communication* → *Express Wizard* il che equivale a cliccare sull'icona evidenziata nella schermata dell'Explorer:



Il nostro I/O Server sarà, naturalmente, lo Zelio Logic, mentre come device potremo indicare più specificatamente il modulo Ethernet SR3NET01 (che è un External I/O Device):

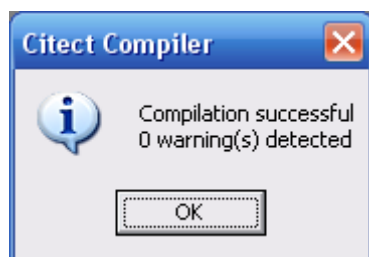
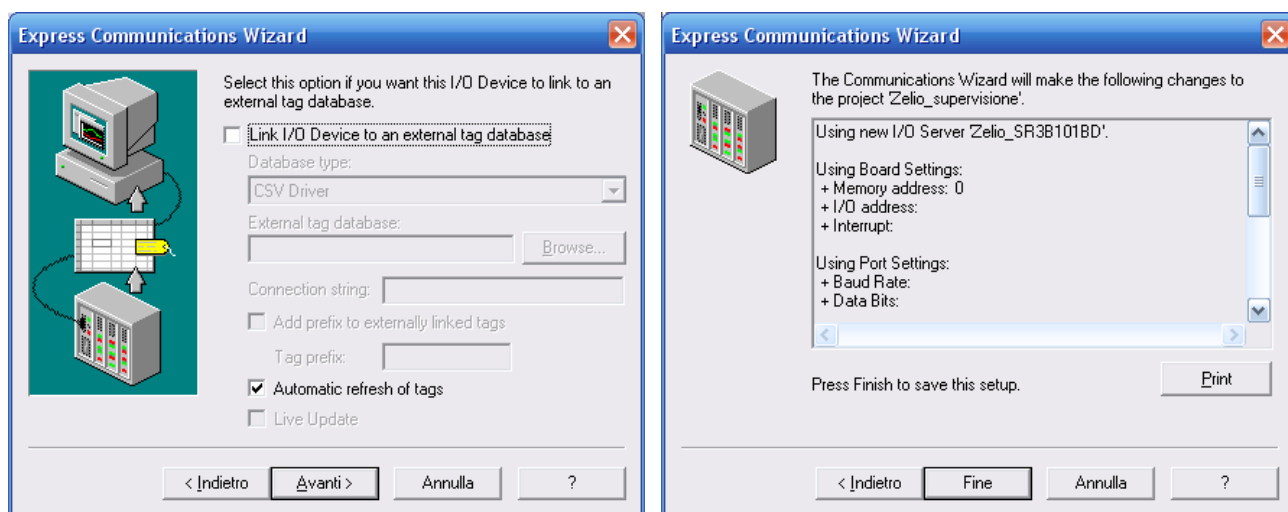


A questo punto, non trovando tra le risorse hardware il nostro modulo logico, andrà benissimo scegliere un **Premium** (in *Schneider-Electric*, *Telemecanique* o *Modicon*) e come protocollo di comunicazione il **Modbus/TCP (Ethernet)**:



E' importante impostare lo stesso indirizzo IP configurato per il modulo in Zelio Soft 2 e scegliere come porta la 502 (quella sulla quale lo Zelio Logic "ascolta" le richieste indirizzate al proprio modulo di espansione Ethernet).

Non disponendo di un file che contenga la lista delle poche variabili che ci interessa leggere allo Zelio lasciamo deselezionata la voce “*Link I/O Device...*” e concludiamo quindi la configurazione del nostro *device*:

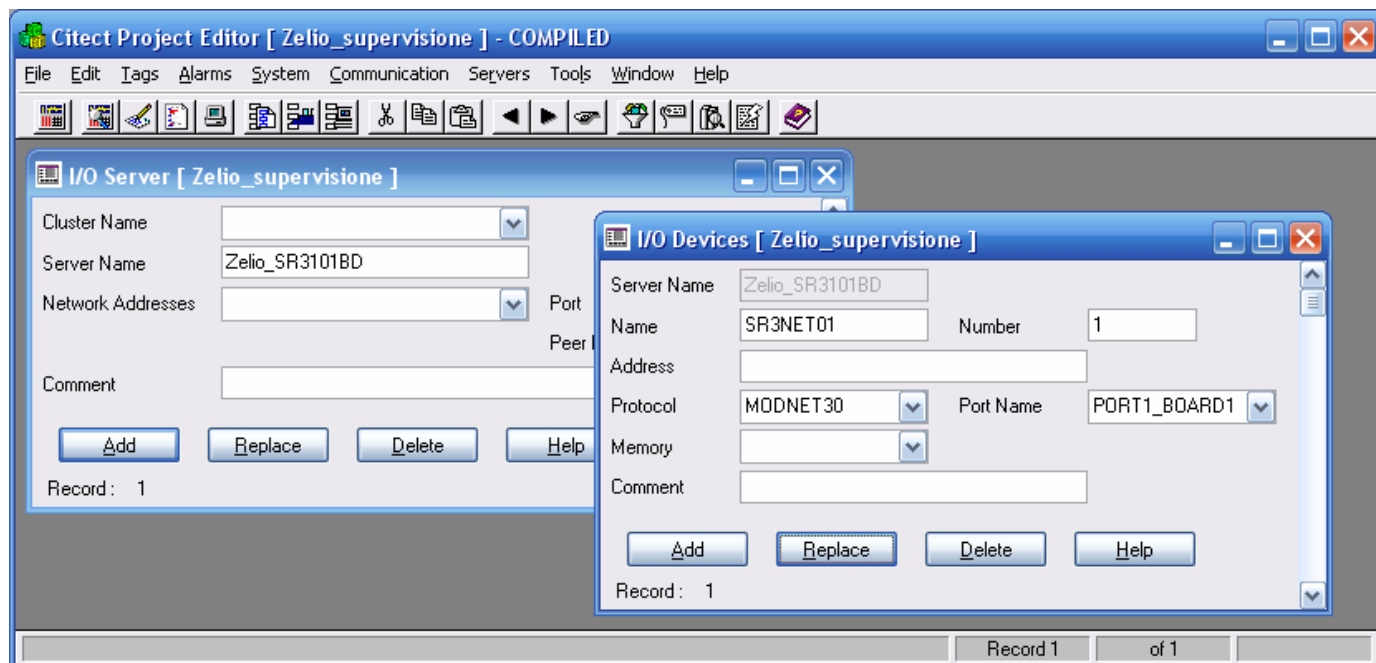


Avendo configurato almeno un *I/O device*, se andremo a ricompilare il nostro progetto non ci verrà più dato alcun *warning*!

Ricordiamoci però che abbiamo scelto un **Premium** come *device*, quando in realtà intendiamo comunicare con uno **Zelio Logic**.

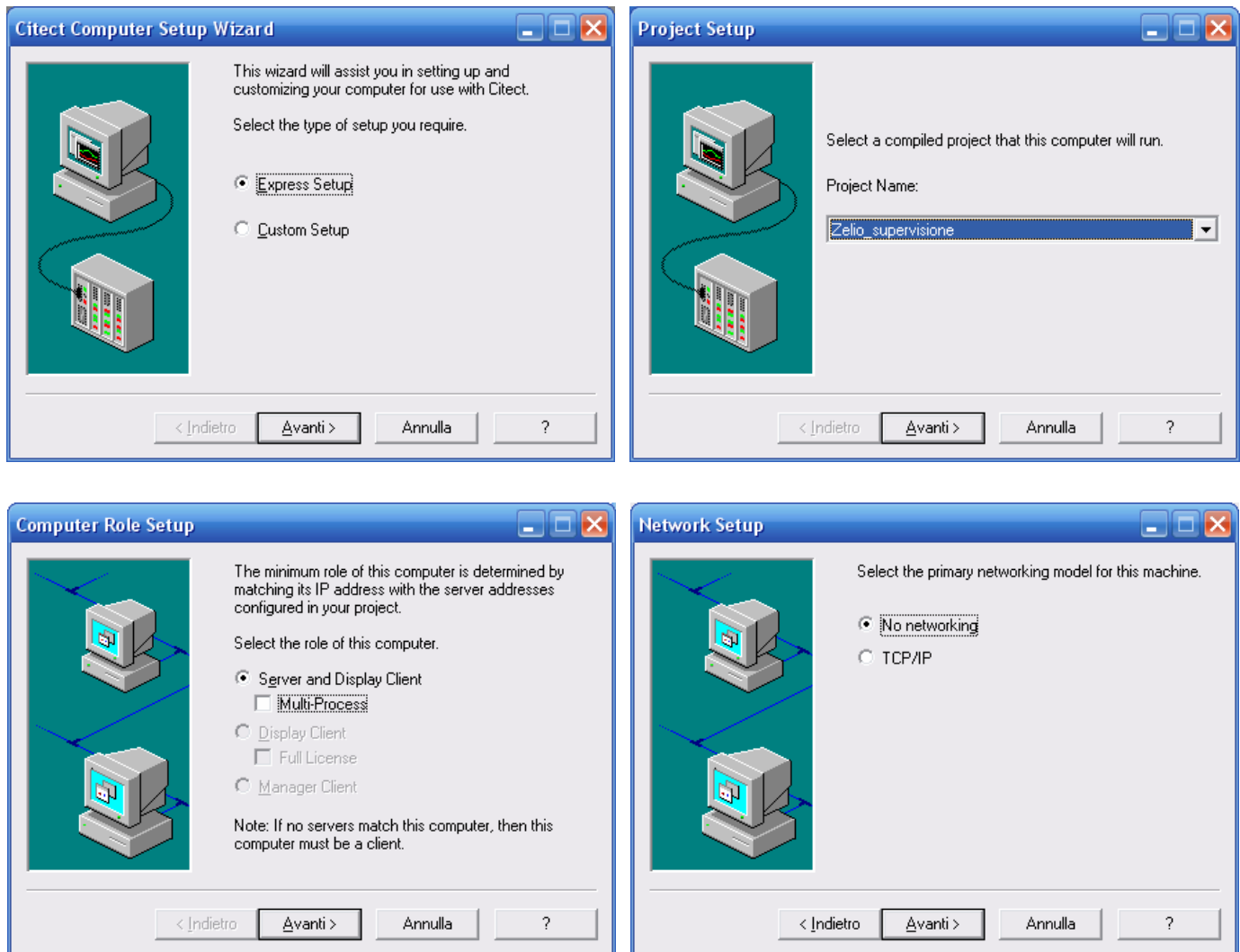
Se vogliamo cambiare il protocollo impostato automaticamente durante la *wizard* (ovvero il *MODNET30*) dobbiamo fare quanto descritto di seguito:

2. andiamo dall’Editor su *Communication* → *I/O Devices* il che equivale a cliccare sull'icona “*I/O Devices*” all’interno della sezione “*Communication*” del nostro progetto. Noteremo quindi che per il nostro modulo Ethernet è stato impostato come protocollo di comunicazione il *MODNET30*:



Come detto in precedenza, il *MODNET30* ci consente di referenziare le 8 WORD del nostro Zelio Logic (o i singoli bit di cui sono composte) utilizzando la sintassi %MWXX o %MWXX.xx. Qualora volessimo utilizzare l'indirizzamento Modbus potremmo scegliere il protocollo *MODNET* dal menù a tendina e quindi salvare (*Replace*, e non *Add*!) la modifica apportata al nostro *I/O Device*.

3. adesso non ci resta quindi che definire il ruolo del nostro sistema di supervisione. In uno qualsiasi dei tre moduli dell'ambiente di sviluppo (l'Editor, l'Explorer o il Graphics Builder) scegliamo *Tools* → *Computer Setup Wizard* e proseguiamo come raffigurato:



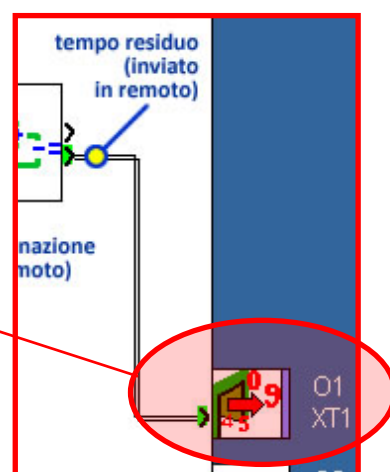
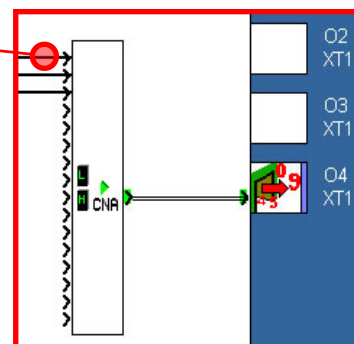
Nell'ultima schermata sceglieremo "No networking" perché intendiamo utilizzare la nostra piattaforma SCADA senza la presenza di alcuna altra piattaforma SCADA in rete. Andiamo quindi avanti ed avremo così concluso la configurazione del nostro sistema di supervisione.

Il passo successivo è quello di definire tutte le variabili che intendiamo leggere e/o scrivere dal nostro modulo logico remoto.

Per la definizione delle variabili occorre andare nella sezione “Tags” del nostro progetto e cliccare su “Variable Tags”, o, in maniera del tutto equivalente, andare dall’Editor su *Tags* → *Variable Tags* e riempire i campi di nostro interesse stando soprattutto attenti all’indirizzo e al tipo di ciascuna Tag.

Ricordiamoci che *Add* comporta sempre l’aggiunta di un nuovo record e che *Replace* ne salva le eventuali modifiche.

Ecco di seguito le definizioni delle nostre 7 variabili:

Citect Project Editor [Zelio_supervisione] - UNCOMPILED

File Edit Tags Alarms System Communication Servers Tools Window Help

Variable Tags [Zelio_supervisione]

Variable Tag Name:

Cluster Name: I/O Device Name:

Address: Data Type:

Raw Zero Scale: Raw Full Scale:

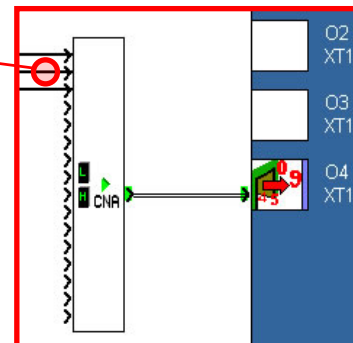
Eng Zero Scale: Eng Full Scale:

Eng Units: Format:

Deadband:

Comment:

Record: 3 Linked: No



Citect Project Editor [Zelio_supervisione] - UNCOMPILED

File Edit Tags Alarms System Communication Servers Tools Window Help

Variable Tags [Zelio_supervisione]

Variable Tag Name:

Cluster Name: I/O Device Name:

Address: Data Type:

Raw Zero Scale: Raw Full Scale:

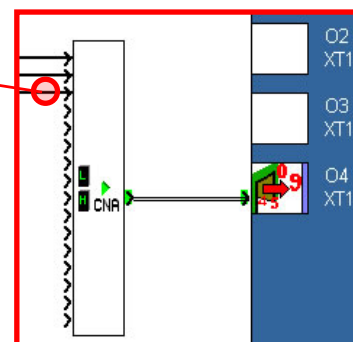
Eng Zero Scale: Eng Full Scale:

Eng Units: Format:

Deadband:

Comment:

Record: 4 Linked: No



Citect Project Editor [Zelio_supervisione] - UNCOMPILED

File Edit Tags Alarms System Communication Servers Tools Window Help

Variable Tags [Zelio_supervisione]

Variable Tag Name:

Cluster Name: I/O Device Name:

Address: Data Type:

Raw Zero Scale: Raw Full Scale:

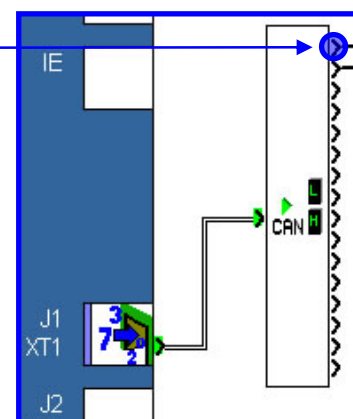
Eng Zero Scale: Eng Full Scale:

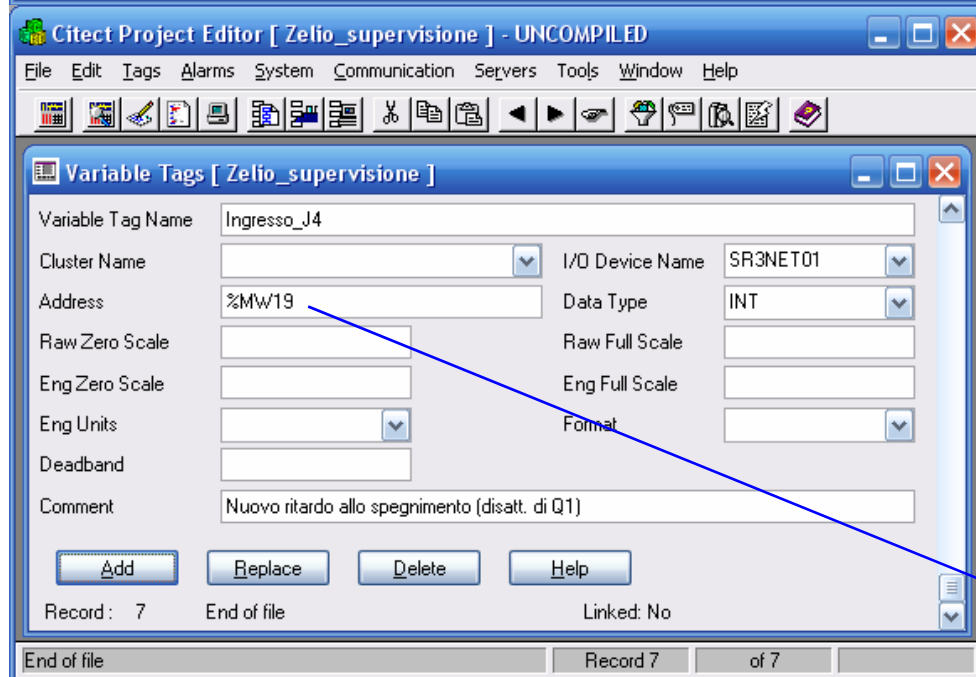
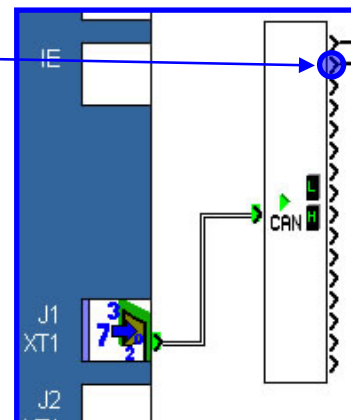
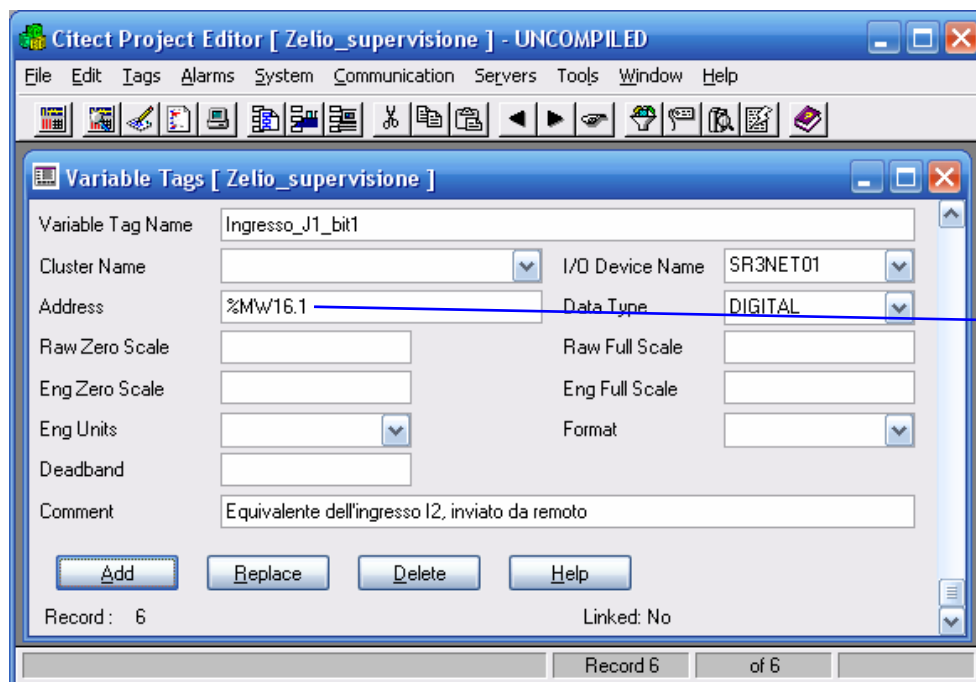
Eng Units: Format:

Deadband:

Comment:

Record: 5 Linked: No









Definite quindi le variabili non ci resta che disegnare gli oggetti del nostro sinottico virtuale e quindi legarne il comportamento a ciascuna di esse.

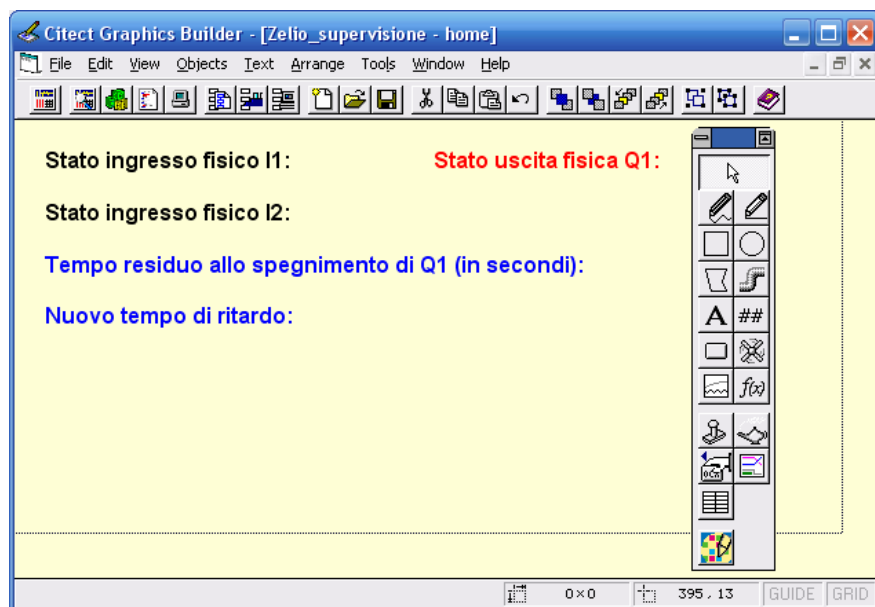
Torniamo quindi al "Graphics Builder" ed inseriamo finalmente, nella pagina grafica creata precedentemente, quanto necessario alla visualizzazione ed al comando del nostro modulo logico remoto.


La barra di comandi a fianco è quanto ci serve per il rapido inserimento dei simboli grafici e dei pulsanti di comando.

In particolare ne utilizzeremo gli strumenti:

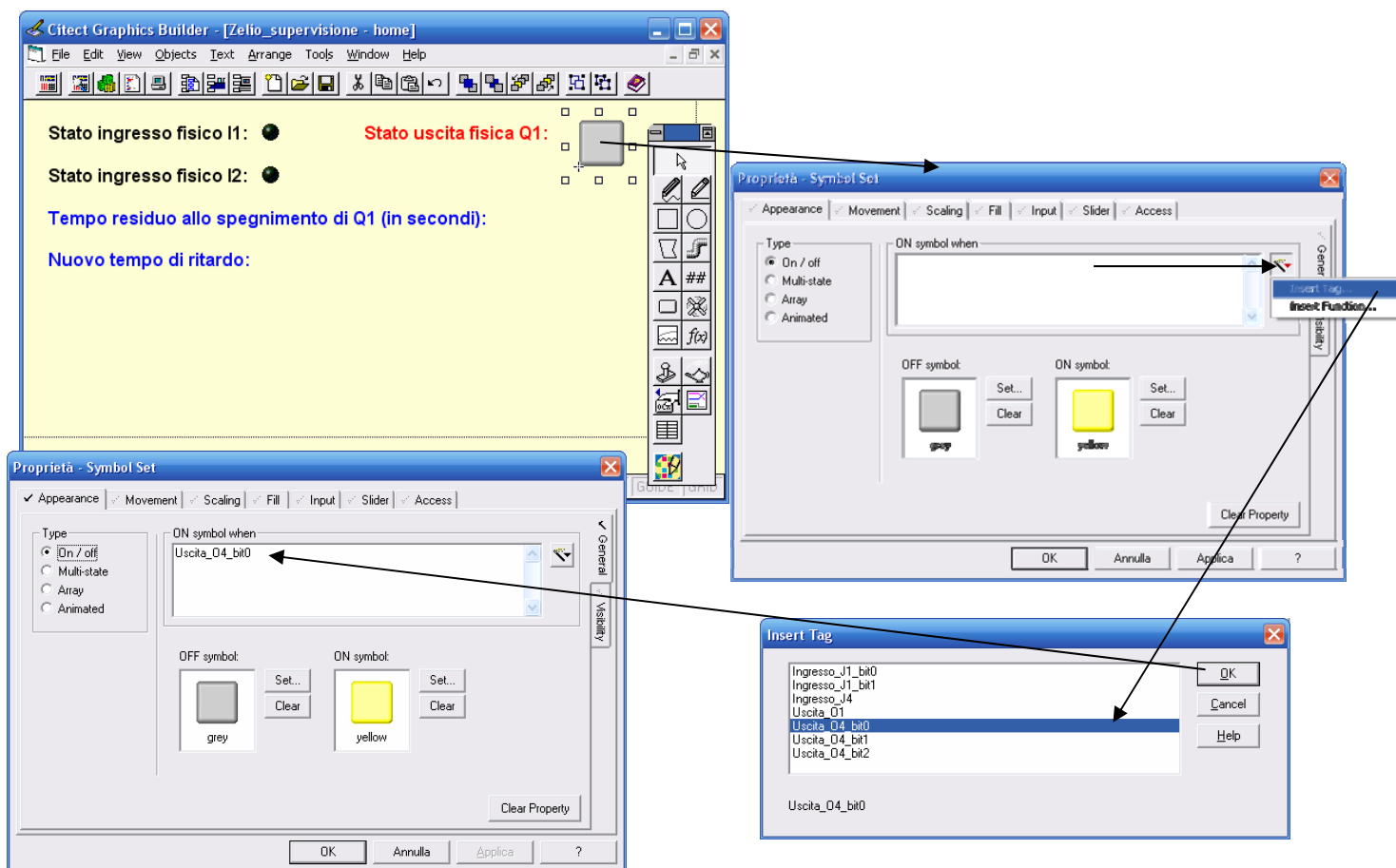
-  per l'inserimento di etichette e campi di testo
-  per l'inserimento dei campi numerici
-  per l'inserimento dei simboli grafici che ci interessano
-  per l'inserimento di pulsanti di comando

Assunto che per la costruzione della pagina si possa procedere come si vuole, si è iniziato, in quest'esempio, con l'inserimento dapprima di tutte le etichette descrittive:



...successivamente quindi, utilizzando il "Symbol Set" () si è passati all'iserimento delle spie di segnalazione, associando a ciascuna di esse il relativo Tag.

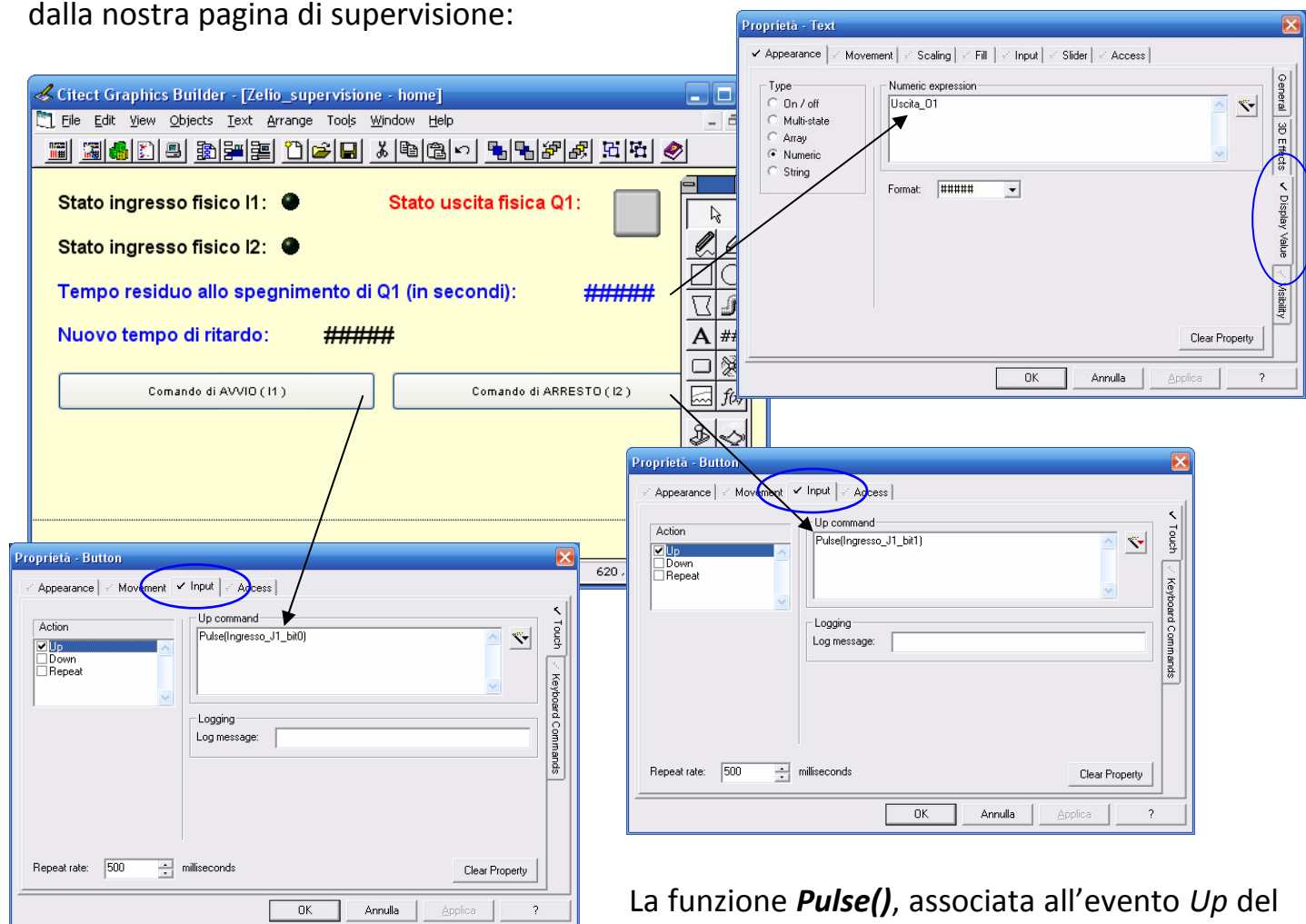
Ad esempio per l'uscita Q1 è stata scelta una spia luminosa quadrata cui è stato associato, secondo i passaggi logici raffigurati di seguito, il Tag "Uscita_O4_bit0":



Tutto ciò che si trova all'interno del box "ON symbol when" viene ciclicamente **valutato**, e, finchè semanticamente **VERO**, comporta la visualizzazione del simbolo scelto per il box "ON symbol:", altrimenti quella del simbolo presente nel box "OFF symbol:".

Il tag *Uscita_O4_bit0* assume il significato di *VERO* quando il bit letto è pari ad 1, e quindi se e solo se l'uscita Q1 dello Zelio è attiva la spia quadrata della pagina grafica è visualizzata come gialla.

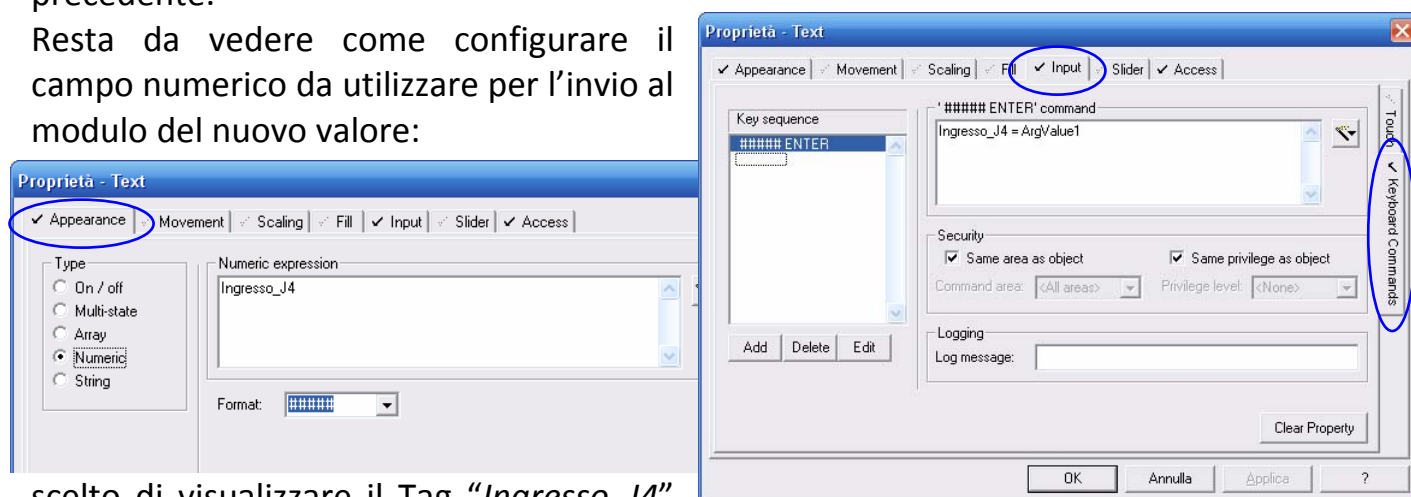
Aggiungiamo adesso il campo numerico (##) per la visualizzazione del valore dei secondi residui allo spegnimento ed i comandi (□) per l'attivazione e lo spegnimento del *TIMER* dalla nostra pagina di supervisione:



La funzione ***Pulse()***, associata all'evento *Up* del pulsante di comando, accetta come

argomento il nome di un Tag digitale. L'effetto è quello che al rilascio del pulsante viene scritto un "1" sulla variabile in questione e successivamente riportatone il valore allo stato precedente.

Resta da vedere come configurare il campo numerico da utilizzare per l'invio al modulo del nuovo valore:

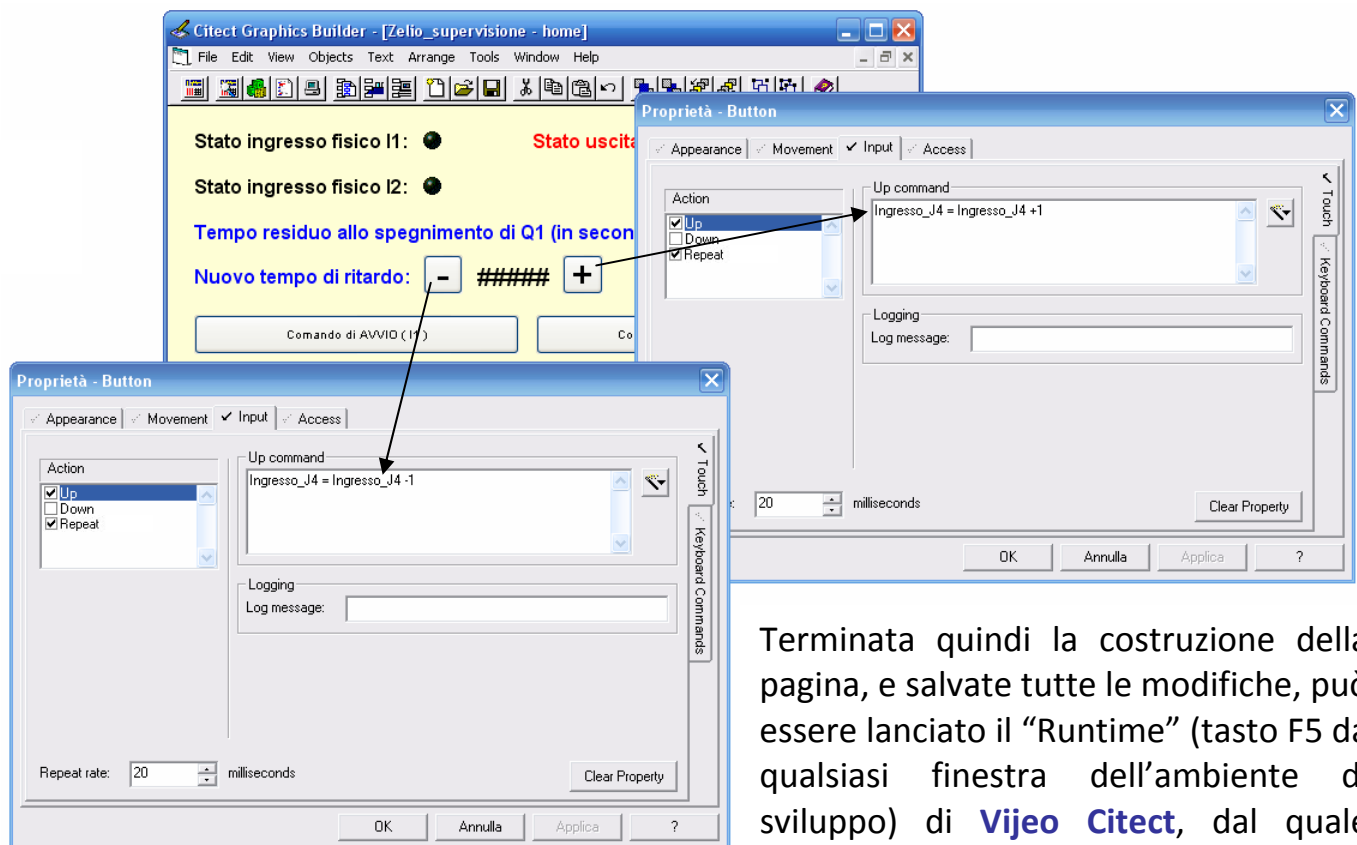


scelto di visualizzare il Tag "*Ingresso_J4*"

nel campo numerico creato, affinché possa venire accettato un nuovo valore come input vanno indicate, nella scheda *Input*, la *Key sequence* accettata (##### ENTER corrisponde

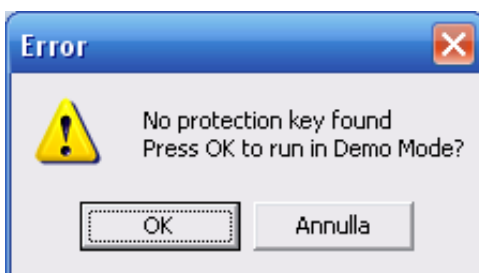
ad un input di massimo 5 cifre terminate comunque dalla pressione del tasto *INVIO*) e l'azione da eseguire al termine della sequenza stessa (*ArgValue1* è il valore numerico digitato; alla pressione del tasto *INVIO*, e quindi al termine dell'intera sequenza, il valore precedentemente assegnato alla WORD J4 viene sovrascritto con quello appena digitato. I valori ammissibili dallo Zelio, come nuovo tempo di ritardo allo spegnimento, vanno da 61 a 32767, valori più piccoli o più grandi verranno ignorati).

Come visibile dalla seguente schermata sono anche stati aggiunti 2 pulsanti per l'incremento ed il decremento unitario del valore inviato:



Terminata quindi la costruzione della pagina, e salvate tutte le modifiche, può essere lanciato il "Runtime" (tasto F5 da qualsiasi finestra dell'ambiente di sviluppo) di **Vijeo Citect**, dal quale potremo comandare il nostro modulo

logico e visualizzare le informazioni relative ai suoi ingressi ed alle sue uscite.



Al lancio del "Runtime", in assenza della chiave hardware USB, il programma ci avviserà di runnare in "Demo Mode", modalità che ne comporterà il funzionamento, senza tuttavia alcuna limitazione delle funzionalità, per soli 15 minuti. Trascorsi i 15 minuti occorrerà rilanciare il Runtime per avviare nuovamente la supervisione.

In assenza di comunicazione col dispositivo remoto i valori visualizzati (che andrebbero letti o scritti su di esso) vengono tutti sostituiti dalla stringa **#COM**, mentre gli oggetti grafici legati allo stato di particolari bit, o al risultato di espressioni che comunque coinvolgono dati relativi ad un dispositivo non raggiungibile, vengono ombreggiati.

