

UNIVERSITA' DEGLI STUDI DI PADOVA

FACOLTA' DI INGEGNERIA

Corso di laurea in Ingegneria dell'Automazione



Tesi di Laurea

REALIZZAZIONE DI SISTEMI DI
CONTROLLO E SUPERVISIONE PER
IMPIANTI TECNICI DI UFFICIO

Relatore:

Prof. STEFANO VITTURI

Laureando:

FILIPPO GENNARO

ANNO ACCADEMICO 2009 / 10

INDICE

Prefazione

Descrizione progetto tesi
Descrizione azienda

Capitolo 1 – Descrizione e programmazione PLC

- 1.1 PLC – Programmable Logic Control
- 1.2 Configurazione Stazione Simatic
- 1.3 Software Simatic Step 7
 - 1.3.1 Descrizione generale dei blocchi
 - 1.3.2 Strutture ammesse per i dati Step 7
 - 1.3.3 Blocco Organizzativo e blocchi Errore
 - 1.3.4 Funzioni: FC
 - 1.3.5 Blocchi Funzionali: FB
 - 1.3.6 Blocchi: Globali DB e Istanza DBI
 - 1.3.7 S7 - PLCSIM - Ambiente di simulazione
- 1.4 Struttura dell'algoritmo. Gerarchia blocchi, ingressi ed uscite
- 1.5 Software WinLC

Capitolo 2 – System Platform (ArchestrA)

- 2.1 Descrizione
- 2.2 Componenti utilizzati nel progetto
- 2.3 Struttura della Piattaforma
 - 2.3.1 Configurazione del PLC
 - 2.3.2 Configurazione Meter
 - 2.3.3 Configurazione Eventi
 - 2.3.4 Configurazione Allarmi
- 2.4 Editor Grafico. Figure create

Capitolo 3 – SQL Server 2005

- 3.1 Descrizione
- 3.2 Query create
- 3.3 Store Procedure

Capitolo 4 – Corporate Energy Management

- 4.1 Utilizzo del CEM

Capitolo 5 – Software di Supervisione

- 5.1 Descrizione
- 5.2 Windows Maker per la creazione dell'ambiente di Supervisione
- 5.3 Windows Viewer. Funzionalità di ogni singola "Videata"
 - 5.3.1 Windows – Globale
 - Pulsante – Gestione Scenari
 - Pulsante – Autoapprendimento ON/OFF
 - Pulsante – Vista rapida Scenario
 - Pulsante – Vie di fuga
 - Pulsante – Spegni tutto
 - Pop Up Comandi

- Pulsantiera
- 5.3.2 Windows – DushBoard
- 5.3.3 Windows – Impostazione Clima
 - Impianto Climatizzazione
 - Impostazione Manuale Orari
 - Personalizzazione Fascie Orarie
 - Personalizzazione Valori di Soglia
- 5.3.4 Windows – Impostazione Neon ad Aspirazione
- 5.3.5 Windows – Allarme
- 5.3.6 Windows – Mappa Consumi
- 5.3.7 Windows – Eventi
- 5.3.8 Windows – Trend
- 5.3.9 Windows – Riepilogo Utenze

Capitolo 6 – Sistema di supervisione, funzionalità in dettaglio

- 6.1 Gestione IN ed Out del PLC
- 6.2 Descrizione del concetto di scenario e relativa gestione
- 6.3 Gestione dei pulsanti e loro assegnazione a determinati scenari
- 6.4 Gestione delle varie utenze
 - 6.4.1 FC Luce
 - Gestione Luci
 - Gestione Neon
 - 6.4.2 FC Forze
 - Gestione Clima
 - Gestione Aspirazione
 - Gestione Prese
- 6.5 Allarmi
- 6.6 LDC – Lista Dati di Comunicazione

Capitolo 7 – Conclusioni

- 7.1 Principali problematiche riscontrate
- 7.2 Possibile implementazione per soluzioni industriale
- 7.3 Conclusioni ed impressioni

PREFAZIONE

Descrizione del progetto

Andremo ora ad illustrare in maniera dettagliata il progetto di Laurea Specialistica tenuto presso la ditta Autoware S.r.l.

Per iniziare possiamo dire che l'obiettivo da raggiungere in questa tesi è la creazione di un sistema di supervisione. Per capire a pieno di che si tratta dobbiamo brevemente descrivere e definire i principali componenti necessari a portare a termine l'obiettivo prefissato. Nella seguente tesi è riportata passo per passo la procedura seguita per creare il sistema di supervisione che consentirà di controllare l'ufficio Autoware.

Con il concetto di *supervisione* vogliamo intendere la possibilità di *controllare* un determinato impianto. Questo controllo può essere di tipo *passivo* oppure *attivo*. Si parla di controllo passivo nel caso sia richiesto all'utente di monitorare delle determinate variabili dell'impianto in questione per poterne elaborare successivamente i risultati ottenuti. Il controllo attivo invece consiste nel compiere azioni che andranno ad agire direttamente sulle variabili fisicamente presenti.

Nel nostro caso dobbiamo aver la possibilità di compiere entrambi i tipi di controllo, il sistema in questo caso dovrà quindi consentirci di monitorare ma anche modificare i valori delle grandezze prese in considerazione.

I principali strumenti o componenti necessari a definire un *sistema di supervisione* sono:

- Programmazione di un PLC
- Costruzione di un Software di supervisione
- Connessione tra PLC e suddetto Software
- Disponibilità di strumenti di rilevazione dati (Meter)

L'insieme e il reciproco legame tra questi componenti illustrati ci consente di raggiungere lo scopo prefissato.

Programmazione di un PLC

Nel caso in questione il Programmable Logic Control installato in ufficio è un Siemens S7 quindi la sua programmazione è stata effettuata tramite il software dedicato Step 7 della Simatic.

Il linguaggio di programmazione utilizzato è AWL.

Durante questa fase è stato creato l'algoritmo necessario per far sì che le operazioni desiderate vengano correttamente eseguite fisicamente sull'impianto. Questo tipo di programmazione è considerato a basso livello, vengono dunque settati direttamente i bit di ingresso, memoria oppure uscita del PLC tramite un linguaggio simile ad Assembly.

Costruzione di un Software di Supervisione

Una volta creato l'algoritmo che opererà le azioni a livello logico è necessario creare una interfaccia Utente – PLC in modo che siano visibili e facilmente modificabili le variabili interessate.

Connessione tra PLC e Software di Supervisione

Per una più facile interpretazione e una maggiore flessibilità del progetto è stato usato un software (non indispensabile) detto System Platform montato in un dispositivo Server. Al capitolo 4 verrà illustrata la sua configurazione.

Strumenti di rilevazione dati (Meter)

Sono strumenti necessari ad acquisire, a livello di impianto, i valori delle grandezze interessate. In questo ufficio sono presenti 3 Meter della Socomec che consentono di rilevare una gran quantità di valori di tensione, corrente, potenza ed energia.

Questi 3 Meter sono associati rispettivamente a:

- Impianto Generale
- Impianto Luce
- Impianto Clima

Il primo Meter ovviamente rileverà i dati dell'intero ufficio, risulterà essere quindi la somma del consumo di tutti gli utilizzatori presenti. Per quanto riguarda gli altri due, il loro utilizzo è importante per riuscire a capire il consumo specifico di solo impianto di illuminazione e impianto di climatizzazione.

In realtà, a livello di controllo PLC e supervisione, è stato utilizzato un ulteriore Meter. Esso non è presente fisicamente in ufficio risulta quindi dall'unione dei dati ricavati dai tre Meter presenti. Una volta ottenuti i valori, più interessanti sono quelli relativi al consumo di energia, è stato effettuato il calcolo:

$$FEM = \text{Impianto Generale} + \text{Impianto Clima} - \text{Impianto Luce}$$

il tutto denominato *FEM*, che sta per forza elettro motrice.

Riportiamo una figura rappresentante il Meter utilizzato per il progetto in questione.



Figura 1. Meter Socomec Diris A40

Forniamo di seguito la piantina in scala dell'ufficio per capire al meglio le dimensioni e la disposizione delle utenze che devono essere comandate.

In questo caso l'impianto risulterà essere un ufficio di circa 400 metri quadri. Al suo interno sono presenti numerosi interruttori e prese, postazioni per PC, strumenti di rilevazione dati, neon fluorescenti, generici utilizzatori, un impianto di climatizzazione ed uno di aspirazione. Il tutto sarà gestito tramite un PLC Siemens installato e collegato all'impianto elettrico dell'edificio.

Il primo passo risulta quindi essere una adeguata programmazione del PLC, dovremo creare un algoritmo in grado di soddisfare le più disparate richieste in fatto di accensioni, spegnimenti, temporizzazioni, etc.

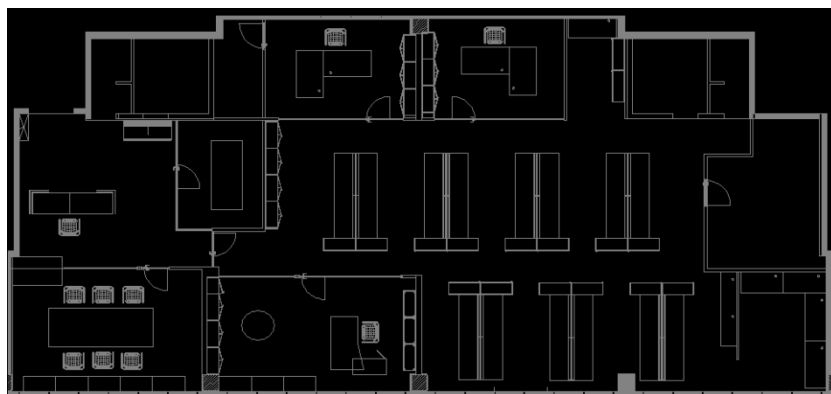


Figura 2. Piantina ufficio Autoware

Nei prossimi capitoli illustreremo in dettaglio le singole utenze da comandare.

Grazie all'utilizzo di strumenti di rilevazione (Meter) sarà possibile monitorare una serie di dati e compiere un'analisi dei consumi all'interno dell'ufficio. Di notevole interesse sarà il monitoraggio di consumo di Energia, Potenza, Tensione e Corrente.

Riportiamo di seguito uno schema che riassume la struttura dei componenti necessari a creare il nostro sistema di supervisione.

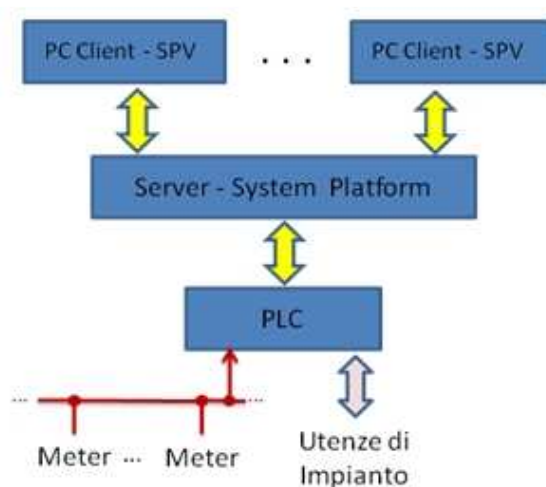


Figura 3. Struttura sistema di supervisione

Nei prossimi capitoli verranno descritti in dettaglio tutti i blocchi a partire dal PLC. Questo in particolare garantisce il collegamento tra la parte hardware e quella software dell'impianto. I Meter e le Utenze di Impianto infatti rappresentano i componenti fisici che saranno controllati e configurati a livello software tramite il Supervisore (SPV).

Il ruolo della System Platform è sostanzialmente quello di consentire e gestire il collegamento di svariati software di supervisione. Infatti ogni PC Client presenterà uno specifico Supervisore. Nel nostro caso comunque sarà presente un solo SPV e verrà descritto al capitolo 5.

Ora che sono stati definiti gli strumenti e descritto l'impianto in questione siamo in grado di capire a che ci si riferisce quando si parla di *Sistema di supervisione*.

Descrizione dell'azienda ospitante

Il progetto in questione è stato tenuto presso l'Azienda Autoware S.r.l. di Ing. Luigi De Bernardini.

Autoware fornisce soluzioni all'avanguardia per la gestione e l'automazione della produzione, per il controllo e la supervisione di impianti e processi, per la verifica della qualità attraverso innovative applicazioni di visione artificiale.

Attenta alle continue evoluzioni ed alle esigenze del mercato, Autoware è specializzata nella fornitura di soluzioni per la raccolta e la distribuzione dei dati di produzione sfruttando le più moderne tecnologie così da riempire il vuoto di comunicazione che esiste tra "shop floor" ed "ERP".

Affianca i clienti garantendo consulenza qualificata sin dalle prime fasi del progetto per definire assieme a loro le soluzioni più idonee alle loro necessità e li accompagna attraverso la realizzazione fino al raggiungimento degli obiettivi individuati.

Elenchiamo di seguito i principali settori di sviluppo dell'azienda in questione.

Industrial Automation

Profonda conoscenza delle piattaforme hardware e software offerte dal mercato, competenza di tecnici altamente specializzati, esperienza sviluppata nel corso degli anni, si traducono in fornitura di soluzioni di controllo di impianti e processi moderne ed efficaci, caratterizzate da un rapido ritorno sull'investimento.

Servizi di progettazione:

- Consulenza
- Definizione specifiche
- Progettazione sistemi controllo e architetture
- Schemi e disegni

Programmazione e configurazione:

- PLC e controllori
- HMI e SCADA
- Progettazione di data base, configurazione e raccolta dati
- I/O e driver seriali
- Configurazione sistemi operativi, server e server ridondanti
- Networking

Servizi sul campo e documentazione operativa:

- Costruzione quadri, installazione e cablaggio
- Avviamento sistemi
- Manuali operatore, istruzione personale
- Assistenza 24/7

MES - Manufacturing Executing System

Autoware è un partner di fiducia in grado di colmare efficacemente il gap tra Shop Floor ed ERP attraverso la fornitura di soluzioni che permettono di:

- Migliorare la conoscenza dell'azienda attraverso informazioni basate su dati reali, sintetizzate in cruscotti, report o grafici;
- Ottimizzare l'utilizzo delle risorse;
- Aumentare l'efficienza produttiva;
- Ridurre il magazzino;
- Ridurre gli scarti;
- Ridurre i tempi di risposta agli eventi;
- Aumentare la visibilità sui dati di produzione.

Il valore aggiunto dell'Azienda:

- Specifica conoscenza dei diversi processi produttivi;
- Sviluppo di un approccio indipendente basato sull'esperienza accumulata;
- Progetti personalizzati in funzione delle reali esigenze, sviluppati con attenzione al futuro;
- Elevata conoscenza delle tecnologie e dei prodotti disponibili sul mercato.

La Gestione e l'ottimizzazione del processo produttivo, interfacciamento di sistemi gestionali e linee di produzione è un compito impegnativo che richiede professionalità e vasta conoscenza delle diverse tecnologie.

CEM - Corporate Energy Management

Si tratta di una nuova applicazione per raggiungere in modo semplice e immediato gli obiettivi di efficienza energetica, letteralmente gestione dell'energia all'interno dell'azienda.

Invensys Operations Management, attiva nella ricerca e sviluppo di soluzioni industriali volte al miglioramento delle performance delle aziende manifatturiere e dei servizi, ha annunciato il lancio di Wonderware® Corporate Energy Management Application. La nuova applicazione consente alle aziende di implementare in modo semplice e veloce un programma di Energy Management per raggiungere gli obiettivi di sostenibilità energetica. Wonderware Corporate Energy Management Application, parte del portfolio aziendale di soluzioni per la gestione dell'energia, aiuta i clienti ad affrontare al meglio le problematiche legate al consumo energetico.

L'applicazione sfruttata in questo progetto, e descritta dettagliatamente in seguito, consente agli utenti di monitorare i consumi energetici, identificare e segnalare al personale addetto le relative inefficienze. E' in grado di rilevare le informazioni direttamente dai Meters presenti in rete, attraverso i più disparati protocolli di comunicazione (anche proprietari) e di ricevere informazioni con inserimenti manuali tramite le applicazioni Wonderware InTouch® HMI e Mobile Solution IntelTrac™ presenti in azienda. L'applicazione viene installata sull'infrastruttura software Wonderware System Platform, consentendo così l'integrazione con una vasta gamma di applicazioni, reti e fonti dati I/O. Le funzionalità dell'applicazione includono la registrazione dei consumi rilevati da contatori principali e di reparto e del fabbisogno energetico per diverse tipologie di energia, inclusa elettricità, acqua, gas, sistemi di raffreddamento, aria e vapore. L'applicazione, inoltre, associa i volumi di produzione all'energia utilizzata, fornendo i Key Performance Indicator utilizzati in diversi programmi di Energy Management. Può essere installata nel giro di pochi giorni e i report preconfigurati forniscono a manager, supervisori e operatori maggiore visibilità e informazioni sulle modalità di utilizzo dell'energia all'interno dell'impianto.

“Le attività di Energy Management a livello aziendale non sono solo competenza di direttori di stabilimento e personale tecnico, ma sono entrate a far parte delle responsabilità di tutto il personale attivo all’interno del ciclo produttivo” ha affermato Bill Schiel, Manager of Energy Management Solutions, Invensys Operations Management. “Qualsiasi reparto in azienda consuma energia, incrementando così i costi di prodotti e servizi. Per questo motivo, tutti devono essere coinvolti nello sviluppo di una strategia di gestione dell’energia che possa essere implementata in breve tempo per fornire benefici immediati all’intera azienda.”

La soluzione di Energy Management deve quindi essere in grado di fornire dati, analisi, allarmi, tendenze, scostamenti in modo diverso a tutti gli attori coinvolti nella filiera produttiva: dai manager coinvolti nei temi della contabilità industriali agli operatori che seguono il funzionamento di singole macchine o impianti.

Grazie all’applicazione Wonderware, le aziende possono creare una vasta gamma di web report relativi ai consumi energetici sulla base di periodi temporali, reparti, centri di costo ed eventi operativi. InTouch HMI fornisce visualizzazioni in tempo reale dei consumi energetici, offrendo così un quadro complessivo del livello di utilizzo dell’energia. Con Wonderware Corporate Energy Management Application, le aziende possono rapidamente raggiungere gli obiettivi aziendali di riduzione dei consumi energetici e gestione dell’energia, ottenendo risultati sostenibili a lungo termine.

Riportiamo di seguito una rappresentazione schematica del suddetto metodo di ottimizzazione di impianto.

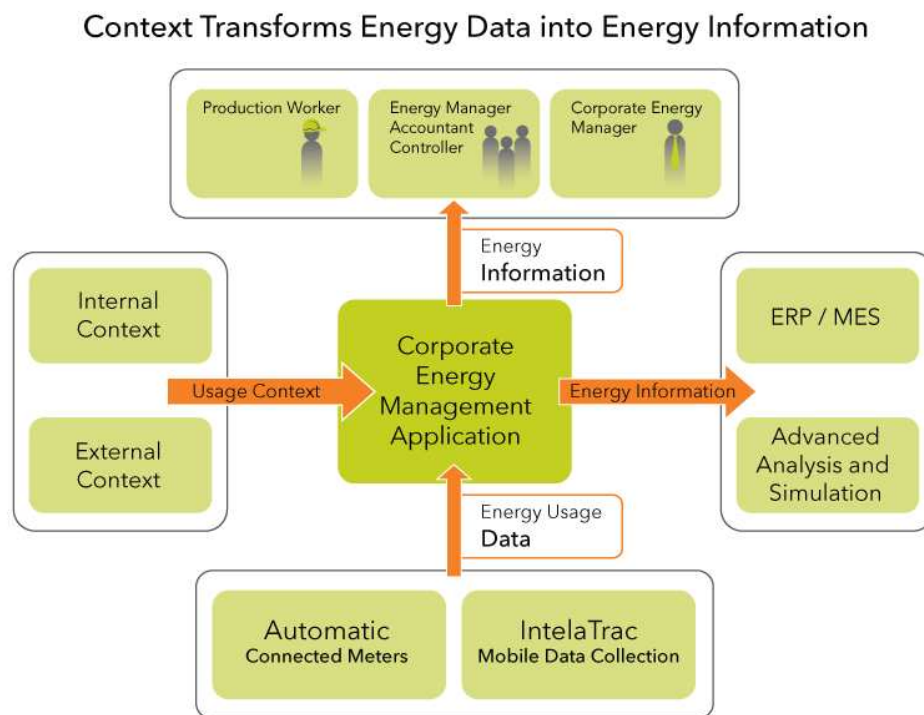


Figura 4. Il contesto di utilizzo trasforma semplici dati di consumo in informazioni utili per la gestione dell’energia

Se raccogliere dati dai contatori è un’operazione piuttosto semplice, le cose si complicano quando si vogliono utilizzare questi dati per ottimizzare i consumi energetici.

Wonderware Corporate Energy Management Application trasforma i dati sui consumi in informazioni utili agli energy manager per gestire l'utilizzo dell'energia in tempo reale. Queste informazioni diventano essenziali per avere una visione chiara dei consumi energetici e ridurre i costi.

L'Energy Management è un processo di miglioramento continuo. Tuttavia, per ottenere e mantenere nel tempo risultati di efficienza energetica, è necessario avere accesso ad informazioni attendibili. Questo prodotto costituisce una fonte d'informazioni complete e affidabili che l'intera organizzazione può utilizzare per verificare che nuovi asset e procedure abbiano un impatto positivo sulla gestione e sui consumi energetici.

In caso di variazioni rispetto a quanto pianificato, le informazioni, altamente contestualizzate, consentono di determinarne rapidamente le cause.

Una volta implementati cambiamenti di procedure o asset, l'applicazione consente di verificare che tali variazioni siano efficaci e sostenibili.

CAPITOLO 1

Descrizione e programmazione PLC

1.1 PLC – Programmable Logic Control

I costruttori di macchine ed impianti fanno fronte sempre più spesso alla richiesta di una maggiore flessibilità e produttività a prezzi decrescenti. Ciò comporta nuove prestazioni dei sistemi di automazione, ad esempio riguardo a set di funzioni, velocità di elaborazione e grandezza costruttiva ma anche in riferimento a engineering e possibilità di collegamento in rete. Sono richiesti controllori a logica programmabile (PLC) di piccole dimensioni, con un ampio set di funzioni ed elevate velocità di elaborazione. Contemporaneamente devono migliorare le possibilità di interconnessione in rete mentre i costi di engineering devono scendere. Il SIMATIC® S7-300® è il controllore più venduto nell'ambito di Totally Integrated Automation, il cui successo è attestato da molte applicazioni di riferimento in tutto il mondo nei più svariati settori industriali. Gli utenti dell'S7-300 traggono quindi vantaggio dall'esperienza e dal service mondiale del leader di mercato nonché dalla qualità del SIMATIC. L'S7-300 è concepito per soluzioni di sistema orientate in particolare alla tecnica manifatturiera e rappresenta, quale piattaforma di automazione universale, una soluzione ottimale per applicazioni in strutture centralizzate e decentralizzate. Grazie a continue innovazioni, questa piattaforma di automazione è sempre più potenziata. Esempi ne sono i sistematici sviluppi progressivi delle potenti unità centrali.

1.2 Configurazione Stazione Simatic

Prima di tutto è stato necessario impostare il tipo di CPU che si andrà ad utilizzare, cioè in linguaggio Siemens il tipo di processore montato sul nostro PLC.

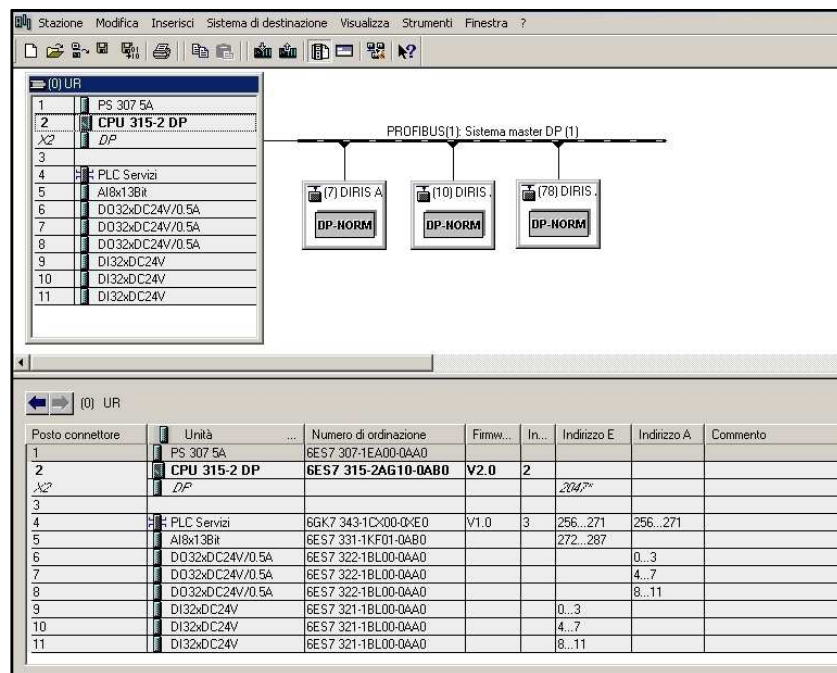


Figura 5. Hardware della stazione

Precisiamo che per tutti gli applicativi Simatic la denominazione PC è riferita direttamente al PLC (ed al suo processore interno, CPU) mentre PG riferita al Personal Computer in uso.

Per il progetto in questione è stato scelto di utilizzare un PLC dotato di processore CPU 315 – 2 DP.

Il controllore è provvisto di una scheda di acquisizione dati analogici di tipo AI8x13bit. Gli indirizzi di ingresso ad essa assegnati sono compresi tra 272 e 287. Quindi i famosi 16 byte di trasferimento dati prima descritti. Con temporizzazione programmata i Meter forniscono alla scheda acquisizione dati del PLC i 16 byte.

1.3 Software Simatic Step 7

STEP 7 è il pacchetto di base utilizzato per configurare e programmare i sistemi di automazione Simatic. Esso rappresenta parte integrante del software industriale ed è utilizzato per applicazioni di dimensioni o complessità superiori per le quali è necessaria ad esempio una programmazione con linguaggi evoluti o linguaggi di tipo grafico oppure è richiesto l'impiego di unità funzionali o di unità di comunicazione. Sotto STEP 7 possono essere utilizzati pacchetti software Simatic aggiuntivi come ad. es. gli Engineering Tools, i quali non sono stati utilizzati in questa esperienza.

Grazie al software STEP 7 è stato possibile creare il programma con il quale intendiamo controllare il nostro sistema. Una volta avviato il Simatic Manager questo programma verrà creato all'interno di un progetto denominato Ufficio_AW.

Forniamo prima di tutto una descrizione ed una vista dei blocchi creati, all'interno di quest'ultimo progetto citato, necessari a compiere le operazioni desiderate. Li descriveremo dettagliatamente ed illustreremo le loro funzionalità.

Il linguaggio con il quale sono state sviluppate le funzioni è AWL.

1.3.1 Descrizione generale dei blocchi

Innanzitutto, come si potrà notare, per una più semplice comprensione del programma tutti i blocchi presenti in macchina sono stati rigorosamente commentati (sfruttando l'editor e tabella dei simboli).

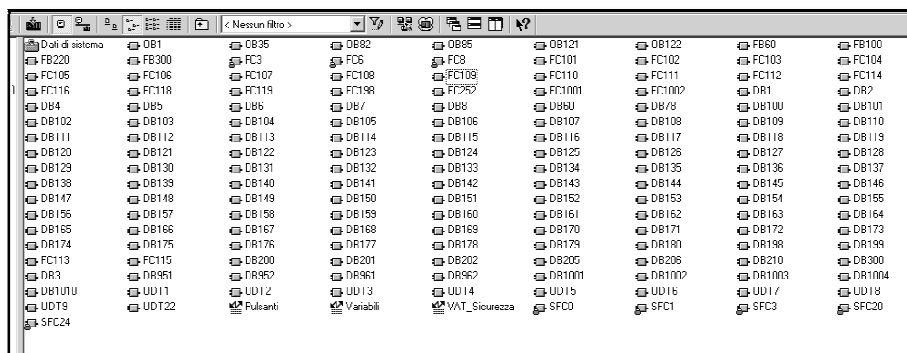


Figura 6. Descrizione blocchi

Oltre al blocco OB1, responsabile all'elaborazione dell'intero algoritmo, sono presenti altri blocchi organizzativi (OB35, OB82, OB85, OB121, OB 122) i quali hanno il compito di “by pass” in caso di errori in modo da garantire comunque la chiusura del ciclo programma.

Eventuali errori rilevati saranno visibili tramite i Led presenti sul PLC o in caso di simulazione sul pannellino rappresentante la CPU del software WinLC.

L'accensione del Led Rosso INTF sta ad indicare un errore riguardante il ciclo programma, il quale non è stato portato a termine correttamente a causa di errori ad esempio di lettura di DB.

Accensione Led EXTF invece riguarda errori in lettura di dispositivi esterni.

Di seguito è riportate due figure, la prima relativa a WinLC mentre la seconda ad un PLC reale (Siemens S300).



Figura 7. A destra PLC reale Siemens S7-300, a sinistra pannello di WinLC (PLC Software)

1.3.2 Strutture ammesse per i dati in Step 7

I dati memorizzati in DB, definiti su delle UDT oppure come variabili di Blocchi Funzionali o Funzioni possono essere del tipo:

- | | |
|---|---|
| <p>Dati Semplici:</p> <ul style="list-style-type: none"> - BOOL - BYTE - WORD - DWORD - INT - DINT - REAL - S5TIME - TIME - DATE - TIME_OF_DAY - CHAR | <p>Dati Composti:</p> <ul style="list-style-type: none"> - DATE_AND_TIME - STRING - ARRAY - STRUCT - UDT |
|---|---|

In seguito, in fase di specificazione delle variabili utilizzate per creare il legame tra PLC e software di Supervisione, andremo a definire più in dettaglio le caratteristiche per la scelta del tipo di dato.

1.3.3 Blocco organizzativo e blocchi errore

I blocchi organizzativi (OB) rappresentano l'interfaccia tra il sistema operativo e il programma utente. Essi vengono richiamati dal sistema operativo stesso e comandano l'elaborazione ciclica del programma su interrupt, il comportamento di avvio del sistema di automazione e la gestione degli errori. Programmato i blocchi organizzativi è possibile determinare il comportamento della CPU.

L'elaborazione ciclica del programma può essere interrotta mediante determinati eventi di avvio (interrupt). Se sopraggiunge uno di tali eventi, il blocco appena elaborato viene interrotto tra un comando e l'altro, e viene elaborato un altro blocco organizzativo assegnato all'evento di avvio. In seguito, l'elaborazione ciclica del programma riprende dal punto in cui è avvenuta l'interruzione. L'elaborazione ciclica del programma è l'elaborazione "normale" nei controllori programmabili. Il sistema operativo richiama ciclicamente l'OB1 e con esso avvia l'elaborazione ciclica del programma utente. La programmazione di questa elaborazione avviene durante la scrittura del programma utente con STEP 7 nell'OB1 e nei blocchi richiamati. L'elaborazione ciclica inizia non appena il programma utente si è concluso senza errori. Possibili cause di interruzione possono essere:

- interrupt o allarme
- comando di STOP
- la mancanza di tensione di rete
- un errore di un dispositivo o del programma (gestita tramite ulteriori OB)

E' possibile ottenere il tempo di ciclo del programma premendo la combinazione di tasti Ctrl + D dalla schermata del Simatic Manager.

Il tempo di ciclo è il tempo richiesto dal sistema operativo per l'elaborazione del programma ciclico nonché di tutte le parti di programma che interrompono tale ciclo (p. es. elaborazione di altri blocchi organizzativi) e per le attività di sistema (p. es. Aggiornamento dell'immagine di processo). Nel nostro caso notiamo dalla seguente figura che tale valore massimo si attesta circa a 50 ms.

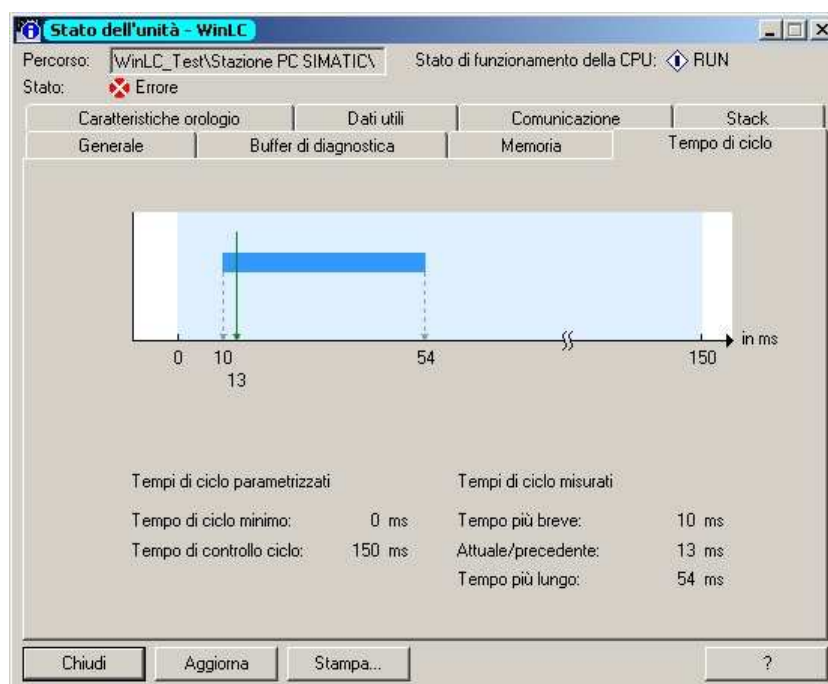


Figura 8. Tempo di ciclo dell'algorithm

1.3.4 Funzioni: FC

Le funzioni fanno parte dei blocchi programmati dall'utente. Consistono in blocchi di codice "privi di memoria". Vengono quindi sfruttate delle variabili temporanee memorizzate nello stack dei dati locali. Dopo l'elaborazione dell'FC, questi dati vanno perduti. Per la memorizzazione dei dati, le funzioni possono utilizzare blocchi dati globali.

Poiché a una FC non è abbinata alcuna memoria, è necessario indicarne sempre i parametri attualmente in uso. Inoltre ai dati locali di una FC non può essere assegnato alcun valore iniziale.

Questi dati locali sono definiti nell'interfaccia del Simatic illustrata in figura seguente.

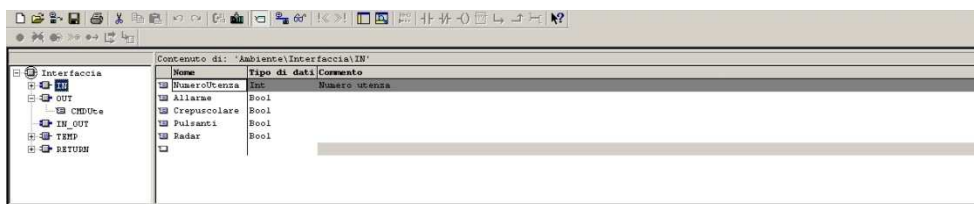


Figura 9. Dichiarazione di IN, Out e Variabili Temporanee per le varie FC

Il campo IN contiene i dati di ingresso della funzione in questione i quali verranno assegnati, con formato corretto, durante il richiamo della funzione stessa. Questo richiamo corrisponde al comando "CALL FC.." in linguaggio Siemens AWL.

Il campo OUT invece conterrà i dati che vengono modificati nella funzione ed avranno poi influenza sullo sviluppo dell'algoritmo.

Infine TEMP contiene i dati temporanei che interessano la FC, verranno sovrascritti da quelli di una successiva FC. La sua dimensione massima viene impostata sul Simatic Configurator e solitamente si attesta sui 256 byte.

Le FC caricate nella CPU in questione sono:

- FC101 (FC Luce)
- FC102 (FC Generale)
- FC103 (FC Forze)
- FC104 (FC HVAC – Asp)
- FC105 (FC IN_OUT_Ute_Scenario)
- FC106 (FC Orari_HVAC – Asp)
- FC107 (FC Gestione Scenari)
- FC108 (FC Aggiornamento)
- FC109 (FC Tempor Attiva Scenari)
- FC110 (FC Meter Socomec)
- FC111 (FC Gestione Pulsanti)
- FC112 (FC CheckOrologio_HVAC)
- FC113 (FC CheckOrologio_Neon)
- FC114 (FC Temporiz Allarme)
- FC115 (FC Orari_PLC_SPV)
- FC116 (FC Pulsanti)
- FC118 (FC Decode Pulsanti)
- FC119 (FC Assegnazione Pulsanti)

Al paragrafo 1.2.7 descriveremo in dettaglio il ruolo di ogni singola funzione.

1.3.5 Blocchi Funzionali: FB

I blocchi funzionali fanno parte, come le FC, di blocchi programmati dall'utente. In questo caso però un blocco funzionale è detto "con memoria". Esso dispone di un blocco dati correlato come memoria (blocco dati di istanza). Sia i parametri che vengono trasmessi all'FB, sia le variabili statiche vengono memorizzati nel blocco dati di istanza. Le variabili temporanee invece, anche in questo caso, vengono memorizzate nello stack dei dati locali.

Al termine dell'elaborazione dell'FB, i dati memorizzati nel DB di istanza non vanno perduti, come invece accade a quelli memorizzati nello stack dei dati locali.

Per quanto riguarda questi blocchi in caso di accesso ai parametri viene utilizzata la copia del parametro attuale nel DB di istanza. Se nel richiamare un FB non viene assegnato un parametro di ingresso o se non viene scritto un parametro di uscita nel blocco, vengono ancora utilizzati i valori meno recenti presenti nel DB di istanza (DB di istanza = memoria dell'FB).

Diversamente dai blocchi funzionali FC ora nell'interfaccia di definizione dei dati locali è presente anche il campo STAT necessario per definire la struttura da assegnare al DB di istanza relativo alla FB considerata.



Figura 10. Struttura di dichiarazione per FB, aggiunta di variabili STAT

Nel nostro programma sono presenti 4 diverse FB:

- FB60 (System_Clock)
- FB100 (Funzione Scenario)
- FB220 (FB_inp_ana_sim)
- FB300 (FB300 Utenze Analogiche)

Le maggiormente utilizzate sono la FB60 e la FB100. Nel paragrafo di seguito descriveremo i blocchi di istanza relativi a queste 2 funzioni.

1.3.6 Blocchi: Globali DB e Istanza DBI

A differenza dei blocchi di codice, i blocchi dati non contengono istruzioni STEP 7, ma servono alla registrazione dei dati utente. Nei blocchi dati sono quindi compresi i dati con cui opera il programma e servono alla registrazione di dati utente che possono essere usati da tutti gli altri blocchi.

La dimensione dei DB è variabile. La struttura dei blocchi dati globali può essere stabilita liberamente.

A ogni richiamo di un blocco funzionale che trasferisce parametri viene assegnato un blocco dati di istanza. Nel DB di istanza vengono memorizzati i parametri attuali e i dati statici dell' FB. Le variabili dichiarate nell' FB determinano la struttura del blocco dati di

istanza. Con il termine istanza si definisce il richiamo di un blocco funzionale. Per esempio, se un blocco funzionale viene richiamato cinque volte nel programma utente S7, di tale blocco esisteranno cinque istanze.

Per poter creare un blocco dati di istanza, deve già esistere l' FB a cui il blocco deve essere assegnato. Il numero della funzione viene stabilito al momento della creazione del blocco dati di istanza.

Struttura DB utilizzati

Illustriamo di seguito tutti i DataBlock utilizzati nel programma PLC e la loro funzione.

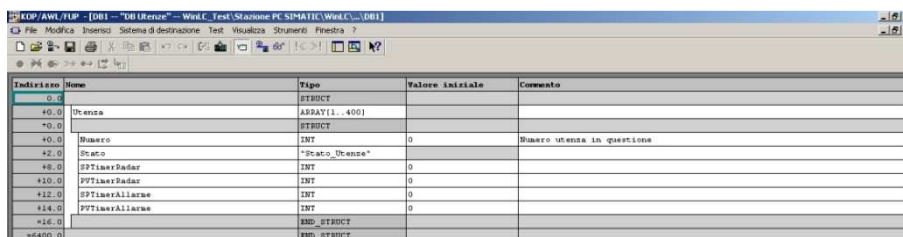
Come illustrato al paragrafo 1.2.2 la tipologia di variabili creabili è molteplice ed essa sarà visualizzata nella colonna *Type* del Simatic Manager.

Come noteremo in seguito, relativamente a certe variabili presenti nella colonna sopra citata, non saranno presenti solamente i tipi di variabile precedentemente descritti. Infatti avremo a che fare con variabili di tipo *UDT*. In questo caso dunque, prima di creare il DB, sarà necessario definire la *UDT* che si andrà ad utilizzare. L'acronimo inglese significa proprio tipo di dati definiti dall'utente.

DB1 – DB Utenze, contiene l'intera lista di utenze presenti nell'ufficio con le relative variabili necessarie a comandarle. La variabile *Utenza* è di tipo *Array* quindi tutto ciò che si troverà “al di sotto di Utenza” verrà ripetuto in numero di volte pari alla dimensione dell'Array.

Il “gruppo” *Utenza* conterrà:

- *Numero*, che rappresenta appunto il numero dell'utenza in questione
- *Stato* (di tipo *UDT*)
- ed alcune variabili utili per impostare valori di temporizzazione.



Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	Utenza	ARRAY[1..400]		
*0.0		STRUCT		
+0.0	Numero	INT	0	Numero utenza in questione
+2.0	Stato	"Stato_Utenze"		
+6.0	PTimerRadar	INT	0	
+10.0	PTimerRadar	INT	0	
+12.0	PTimerAllarme	INT	0	
+14.0	PTimerAllarme	INT	0	
+16.0		END_STRUCT		
+1600.0		END_STRUCT		

Figura 11. Struttura DB Utenze

Ruolo importante è assunto dalla variabile *Stato*, di tipo *UDT*, la quale contiene le variabili di tipo *BOOL*.

Il numero di queste variabili, definite in “*Stato_Utenze*”, sarà pari al numero di utenze totali utilizzate. Ogni utenza dunque possederà le sue variabili di stato che potranno essere settate indipendentemente dalle altre.

Ad esempio la variabile in posizione 0.0 della *UDT* viene utilizzata per fornire il comando manuale di accensione della prima utenza dell'ufficio tramite il Supervisore. Questa stessa variabile si troverà quindi all'interno del *DB Utenze* in posizione 2.0.

Il comando manuale di accensione tramite SPV relativo alla seconda utenza invece si troverà in posizione 2.0 + 16.0 del *DB Utenze*. E così via per tutte le successive.

Nella successiva figura possiamo notare la *UDT* sopra descritta.

Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	Cad_on_man_SPV	BOOL	FALSE	Pressione pulsante ON da SPV
+0.1	Cad_off_man_SPV	BOOL	FALSE	Pressione pulsante OFF da SPV
+0.2	CL_scenario_ON	BOOL	FALSE	Controllo logico ON (Ascensione) da scenario
+0.3	CL_scenario_OFF	BOOL	FALSE	Controllo logico OFF (Spegnimento) da scenario
+0.4	Uscita_FC_Luce	BOOL	FALSE	
+0.5	OH_OFF_scenario	BOOL	FALSE	
+0.6	StatoAutoapprend_ON	BOOL	FALSE	Bit relativo al lampeggio in modalità autoapprendimento, inserita come ON
+0.7	FFOn	BOOL	FALSE	Fronte ON Pulsanti, usato comando FF
+1.0	FFOff	BOOL	FALSE	Fronte OFF Pulsanti, usato comando FH
+1.1	FrontePulsanteON	BOOL	FALSE	
+1.2	FrontePulsanteOFF	BOOL	FALSE	
+1.3	FAOn	BOOL	FALSE	Fronte ON Allarme, usato comando FF
+1.4	FAOff	BOOL	FALSE	Fronte OFF Allarme, usato comando FH
+1.5	FronteAllarmeON	BOOL	FALSE	
+1.6	FronteAllarmeOFF	BOOL	FALSE	
+1.7	FFOn	BOOL	FALSE	Fronte ON Radar, usato comando FF
+2.0	FFOff	BOOL	FALSE	Fronte OFF Radar, usato comando FH
+2.1	FronteRadarON	BOOL	FALSE	
+2.2	FronteRadarOFF	BOOL	FALSE	
+2.3	F_salita_telo	BOOL	FALSE	
+2.4	Fronte_salita_telo	BOOL	FALSE	
+2.5	F_discesa_telo	BOOL	FALSE	
+2.6	Fronte_discesa_telo	BOOL	FALSE	
+2.7	StatoAutoapprend_OFF	BOOL	FALSE	Bit relativo al lampeggio in modalità autoapprendimento, inserita come OFF
+3.0	ForzaON	BOOL	FALSE	FORZA ON Clima, Asp e Neon per periodo selezionato
+3.1	ForzaOFF	BOOL	FALSE	FORZA OFF Clima, Asp e Neon per periodo selezionato
+3.2	Ritardoadallarme	BOOL	FALSE	
+3.3	NO_Crepuscolare	BOOL	FALSE	Togli funzione crepuscolare per l'utenza in questione
+3.4	bit10	BOOL	FALSE	
+3.5	bit110	BOOL	FALSE	
+3.6	bit111	BOOL	FALSE	
+3.7	bit112	BOOL	FALSE	
+4.0	Numero_Scen_Associato	INT	0	Usato per Utenze da 201 a 254, Pulsanti
+6.0		END_STRUCT		

Figura 12. Struttura UDT – Stato_Utenze

DB2 – DB Comandi, contiene i comandi forniti da Supervisore. I primi 3 dati contenuti nella DB sono di tipo INT, occupano dunque 2 Byte ciascuno. La prima variabile, denominata *Numero*, contiene un dato compreso 1 ÷ 54. La terza, denominata *Tipo_Scenario*, un numero compreso tra 1 ÷ 80 (avendo predisposto al software la gestione di 80 scenari).

Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	Numero	INT	0	Numero Utenza in questione
+2.0	Codice_comando	INT	0	Numero compreso: 0 - 32768
+4.0	Tipo_scenario	INT	0	Numero compreso: 0 - 100
+6.0	Comandi_Vari	STRUCT		
+0.0	Scenario_Activo	BOOL	FALSE	
+0.1	FORZAMENTO_ON_Clima	BOOL	FALSE	
+0.2	FORZAMENTO_OFF_Clima	BOOL	FALSE	
+0.3	All_Utenza_ON	BOOL	FALSE	
+0.4	All_Utenza_OFF	BOOL	FALSE	
+0.5	Crepuscolare	BOOL	FALSE	
+0.6	All_Press	BOOL	FALSE	
+0.7	FORZAMENTO_ON_Neon	BOOL	FALSE	
+1.0	FORZAMENTO_OFF_Neon	BOOL	FALSE	
+1.1	FORZAMENTO_ON_Asp	BOOL	FALSE	
+1.2	FORZAMENTO_OFF_Asp	BOOL	FALSE	
+1.3	All_Press_ON	BOOL	FALSE	
+1.4	All_Press_OFF	BOOL	FALSE	
+1.5	Clear Richiesta_Orari	BOOL	FALSE	
+1.6	Allarme_OFF	BOOL	FALSE	
+1.7	Tempo_Disact_Allarme	BOOL	FALSE	
+2.0		END_STRUCT		
+6.0		END_STRUCT		

Figura 13. Struttura DB Comandi

Ruolo importante è assunto dalla seconda variabile, denominata *Codice_comando*, la quale è indispensabile per consentire di inserire o disinserire una utenza da un determinato scenario.

Essendo la variabile di tipo INT e composta da 16 bit è stato scelto di assegnare a quest'ultima 3 diversi valori in modo da compiere 3 diverse operazioni. Mostriamo di seguito le operazioni consentite e i valori ad esse assegnati.

- *Codice_comandi* = 8 ⇒ Utenza inserita come ON

0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

- *Codice_comandi* = 16 \Rightarrow Utenza inserita come OFF

0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

- *Codice_comandi* = 16384 \Rightarrow Utenza disinserita da scenario

0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

I comandi successivi, quelli cioè inclusi in *Comandi_vari*, sono tutti di tipo BOOL quindi semplici bit. Grazie ad essi è possibile compiere forzamenti di varie utenze a stati definiti oppure impostare particolari motodi di funzionamento settando una variabile.

DB4 – DB Orari, contiene gli orari di sincronizzazione tra Supervisione e PLC. Grazie alla funzione FC 115 (*FC Orari_PLC_SPV*) è possibile sincronizzare correttamente i valori attuali di data ed ora, sfruttando le chiamate alle funzioni interne di Siemens “READ_CLK”, “DT_TOD” e “DT to DATE”.

Indirizzo	Nome	Tipo	Valore iniziale	Commento
*0.0	DT_Accuale	DATE_AND_TIME	DT#8-10-7-0:20:0.000	DT da orologio interno
*8.0	DATA_SET	DATE	D#1990-1-1	Data per sincronizzazione da SPV
*10.0	space	INT	0	
*12.0	ORA_SET	TIME_OF_DAY	TOD#0:0:0.0	Ora per sincronizzazione da SPV
*16.0	Data	DATE	D#1990-1-1	data attuale
*18.0	Ora	TIME_OF_DAY	TOD#0:0:0.0	ora attuale
*22.0	GiornoSettimana	INT	0	giorno della settimana
*24.0	Giorno_INIZIO_Richiesta	DATE	D#1990-1-1	
*26.0	Giorno_FINE_Richiesta	DATE	D#1990-1-1	
*28.0	libero1	DATE	D#1990-1-1	
*30.0	libero1	DATE	D#1990-1-1	
*32.0	DT_Disp2	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*40.0	DT_Disp3	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*48.0	DT_Disp4	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*56.0	DT_Disp5	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*64.0	DT_Disp6	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*72.0	DT_Disp7	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*80.0	DT_Disp8	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*88.0	DT_Disp9	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*96.0	DT_Disp10	DATE_AND_TIME	DT#90-1-1-0:0:0.000	DT a disposizione
*104.0	Orario1	"Intervallo"		
*112.0	Orario2	"Intervallo"		
*120.0	Orario3	"Intervallo"		
*128.0	Orario4	"Intervallo"		
*136.0	Orario5	"Intervallo"		
*144.0	Orario6	"Intervallo"		
*152.0	Orario7	"Intervallo"		
*160.0	Orario8	"Intervallo"		
*168.0	Orario9	"Intervallo"		
*176.0	Orario10	"Intervallo"		
*184.0	Orario11	"Intervallo"		
*192.0	Orario12	"Intervallo"		
*200.0	Orario13	"Intervallo"		
*208.0	Orario14	"Intervallo"		
*216.0	Orario15	"Intervallo"		
*224.0	Orario16	"Intervallo"		
*232.0	Orario17	"Intervallo"		
*240.0	Orario18	"Intervallo"		
*248.0	END_STRUCT			

Figura 14. Struttura DB Orari

La prima fornisce in uscita un dato che corrisponde a data ed ora attuali, mentre la seconda e la terza decodificano quello stesso dato e restituiscono separatamente il valore di Ora e Data attuali.

Le variabili sfruttate in questo caso sono di tipo DATE, TIME_OF_DAY e DATE_AND_TIME. Il loro utilizzo è più complesso rispetto a delle semplici INT dato che la fase di “passaggio” valori tramite Supervisore non è banale. Bisogna porre particolare attenzione alla formattazione del valore stesso per non incorrere in errori di decodifica.

Il software di creazione del supervisore, cioè InTouch, riconosce correttamente data ed ora direttamente dall’orologio di sistema e inserisce tali valori nelle variabili \$Time e \$Date entrambe di tipo Strings.

Sarà quindi cura del programmatore trasformare queste due variabili in dati accettabili dalla System Platform. Utilizzeremo la funzione StringToInt.

DB5 – DB HVAC – Asp, è il data block indispensabile per la gestione delle utenze di tipo clima ed aspirazione.

Nel progetto in questione le utenze 34 e 35 sono riferite rispettivamente ad Aspirazione Lato Direzione e Lato Operativo, mentre le 36 e 37 a Clima Lato Direzione e Lato Operativo.

Nel caso di utenze di questo tipo i comandi manuali da Supervisore o gestione scenari rimangono comunque impostati sulla *DB Utenze*, la differenza sta nel fatto che in questo nuovo DB è possibile impostare orari e forzamenti.

Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	HVAC	APPV(1..10)		Previste 10 diverse utenze tipo HVAC
+0.0		STRUCT		
+0.0	N_Utenza_Impianto	INT	0	Numero Utenza alla quale associo HVAC numero 1,2,...
+2.0	Priority	INT	0	
+4.0	ValoreSogliaL_Energia	INT	0	Valore soglia inferiore, superabile in caso di attivazione utenza
+6.0	ValoreSogliaH_Energia	INT	0	Valore soglia superiore, se superata spegnimento utenza
+8.0	Stato	STRUCT		
+0.0	CL_ON	BOOL	FALSE	Accendi
+0.1	CL_OFF	BOOL	FALSE	Spegni
+0.2	Attivo_InImpianto	BOOL	FALSE	Set se l'utenza è attiva nell'impianto in questione
+0.3	Disattivato_Scenari	BOOL	FALSE	Set se se inibita la possibilità di comandare l'utenza tramite scenario
+0.4	Fascia_Orario	BOOL	FALSE	Se true ci troviamo all'interno della fascia oraria di accensione
+0.5	FF_ON_Soglia	BOOL	FALSE	
+0.6	FF_ON_Soglia	BOOL	FALSE	
+0.7	FF_IN_Fascia	BOOL	FALSE	
+1.0	FF_OFF_Fascia	BOOL	FALSE	
+1.1	bit7	BOOL	FALSE	
+1.2	bit	BOOL	FALSE	
+1.3	bit21	BOOL	FALSE	
+1.4	FORZAMENTO_ON	BOOL	FALSE	
+1.5	FORZAMENTO_OFF	BOOL	FALSE	
+1.6	Soglia_Low	BOOL	FALSE	
+1.7	Soglia_High	BOOL	FALSE	
+2.0		END_STRUCT		
+10.0	OrarioMattina	"Intervallo"		
+24.0	OrarioPomeriggio	"Intervallo"		
+30.0	Forzamento_Orario_SPV	"Intervallo"		
+82.0	Giorno_INIZIO_Richiesta	DATE	D#1990-1-1	
+84.0	Giorno_FINE_Richiesta	DATE	D#1990-1-1	
+86.0	Int3	INT	0	
+88.0	Dir34	INWORD	D#16#0	
+82.0	Dir5	INWORD	D#16#0	
+66.0		END_STRUCT		
+660.0	Energy_Consumption	INT	0	Valore proveniente dal Meter Socomec siglato come FEM
+662.0		END_STRUCT		

Figura 15. Struttura DB HVAC – Asp

Ruolo importante è assunto dalle variabili di tipo *UDT* presenti, contengono infatti i bit che saranno settati in caso di corrispondenza dell'orario alle fasce impostate da Supervisore.

Nel capitolo 3 al paragrafo riferito al programma WindowViewer verranno descritti i concetti di forzamento e fasce orarie sopra citati.

Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	Start	TIME_OF_DAY	TOD#0:0:0.0	inizio intervallo ON
+4.0	Stop	TIME_OF_DAY	TOD#0:0:0.0	fine intervallo ON
+0.0	ON	BOOL	FALSE	uscita timer ON
+0.1	abilitata	BOOL	FALSE	abilitazione da SPV
+0.2	ForzaON	BOOL	FALSE	Forza ON da SPV
+0.3	ForzaOFF	BOOL	FALSE	Forza OFF da SPV
+0.4	Reset_Cad_Man	BOOL	FALSE	digitale a disposizione
+0.5	B_Dirp5	BOOL	FALSE	digitale a disposizione
+0.6	B_Dirp6	BOOL	FALSE	digitale a disposizione
+0.7	B_Dirp7	BOOL	FALSE	digitale a disposizione
+0.8	B_Dirp8	BOOL	FALSE	digitale a disposizione
+10.0	DataIN_Richiesta	DATE	D#1990-1-1	DATA DI RICHIESTA INIZIO FORZAMENTO
+12.0	DataFINE_Richiesta	DATE	D#1990-1-1	DATA DI RICHIESTA FINE FORZAMENTO
+14.0		END_STRUCT		

Figura 16. Struttura UDT – Intervallo

La variabile **BOOL ON** verrà settata in caso di entrata in fascia oraria e l'utenza in questione sarà attivata, grazie alle funzioni FC 112 ed FC 113. Verranno così resettati eventuali comandi manuali forniti da supervisore.

DB6 – DB Timer, è il DB che contiene i valori dei temporizzatori che permetteranno l'accensione del Scenario selezionato con ritardo stabilito.

Indirizzo	Nome	Tipo	Valore iniziale	Commento
0.0		STRUCT		
+0.0	Valori_Temporiz_Scenari	ARRAY[1..100]		Previsi 100 Scenari
+0.0		STRUCT		
+0.0	Numero_Scenario	INT	0	Numero Scenario in fase di attivazione
+2.0	SPTimer	INT	0	Set Point temporizzatore
+4.0	PPTimer	INT	0	Processor Value temporizzatore
+6.0		END_STRUCT		
+600.0	TempoAccessa_ON_Scenario	INT	300	Ritardo accensione Scenario alla pressione prolungata del relativo pulsante
+602.0	Accessa_OFF_Radar	INT	75	Ritardo spegnimento Utense alla disattivazione del Radar
+604.0	TempoAccessa_ON_Allarme	INT	300	Ritardo spegnimento Utense all'inserimento dell'allarme generale
+606.0	SPTimer_Allarme	INT	0	
+608.0	Ritardo_Disatt_Allarme	INT	0	
+610.0	Accessa_OFF_Radar_Magaz	INT	1500	Ritardo spegnimento Utense di Magazzino Grande alla disattivazione del Radar
+612.0		END_STRUCT		

Figura 17. Struttura DB Timer

1.3.7 S7 - PLCSIM - Ambiente di simulazione

Il software S7-PLCSIM consente di eseguire e testare i programmi utente su un controllore programmabile (PLC) simulato nel proprio PC o PG.

La simulazione viene effettuata interamente a livello software (quindi internamente a STEP 7) oppure tramite un software che simula un PLC reale. Nel primo caso quindi non è necessario essere collegati ad alcun componente hardware (CPU od unità I/O).

Utilizzando il PLC S7 simulato l'utente può controllare e testare i programmi per le CPU S7-300 e S7-400.

S7-PLCSIM mette a disposizione un'interfaccia molto semplice in cui controllare e modificare vari parametri del programma (ad esempio attivare e disattivare gli ingressi). Inoltre consente di utilizzare le diverse applicazioni di STEP 7 mentre si esegue il programma utente sul PLC simulato.

È quindi possibile usare tool quali esempio la tabella delle variabili (VAT) per controllare e modificare le variabili.

Questo simulatore presenta inoltre le seguenti caratteristiche:

- simulazione viene attivata e disattivata da un pulsante collocato sulla barra degli strumenti del SIMATIC Manager. Attivando il pulsante di simulazione si apre il software S7-PLCSIM con la CPU simulata.
- Possono essere create "sottofinestre" che permettono di accedere alle aree di memoria di ingressi e uscite, agli accumulatori e ai registri del PLC simulato. Si può accedere alla memoria anche mediante indirizzamento simbolico
- Come per le CPU reali, il modo di funzionamento CPU (RUN, RUN-P, STOP) può essere commutato. Inoltre S7-PLCSIM prevede il modo Interrompi che permette di arrestare temporaneamente la CPU senza modificare lo stato del programma.
- Si possono utilizzare OB di allarme nel PLC simulato per verificare le reazioni del programma

1.4 Struttura dell'algoritmo. Gerarchia blocchi, ingressi ed uscite

Come precedentemente illustrato il blocco principale sul quale vengono avviate le chiamate alle funzioni che compiono le operazioni desiderate è l'OB1.

Nel diagramma che segue possiamo notare ciò, 5 FC ed una FB sono richiamate direttamente nell'OB1 mentre le successive sono considerate chiamate annidate. Possiamo vedere come per la FC102 sono presenti altre 2 chiamate in cascata, quindi in questo caso sarà importante tener in considerazione il fatto che le variabili temporanee possono essere accidentalmente sovrascritte. Infatti la dimensione assegnata a queste temporanee (variabili locali, che in seguito illustreremo più dettagliatamente) è di 256 Byte. Questo valore dipende dal tipo di CPU, dalla sua capacità di memoria e da come viene impostata la configurazione della stazione.

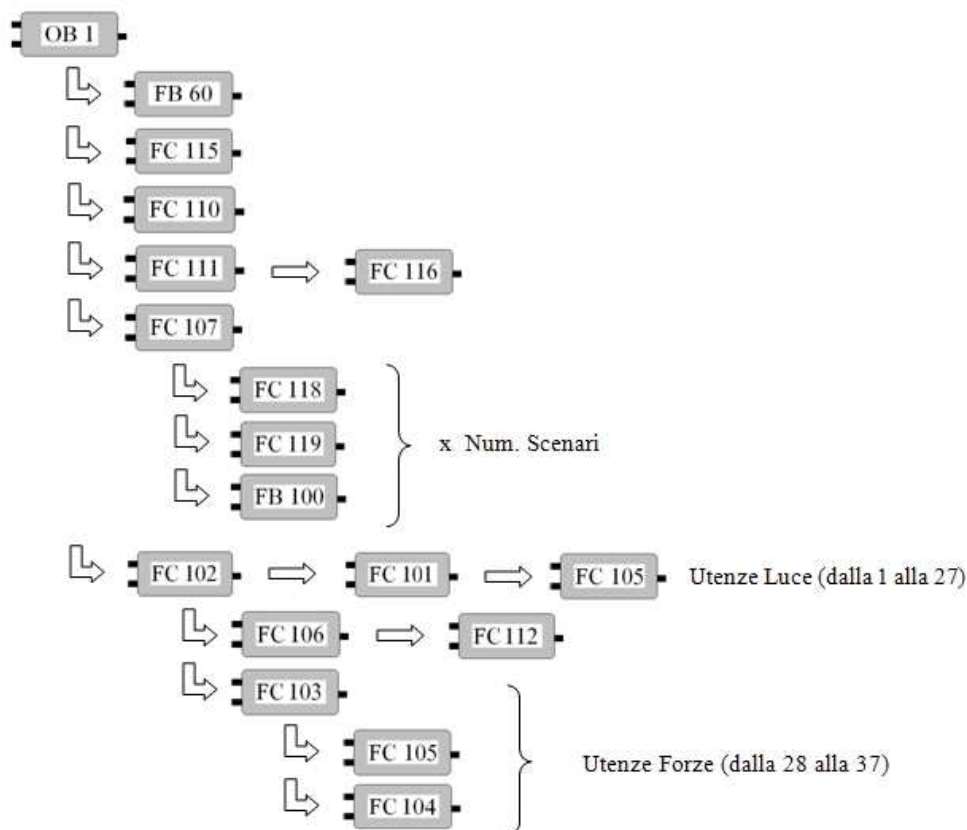


Figura 18. Gerarchia Blocchi FC

Il Blocco FC 102 è stato denominato *FC Generale* dato che al suo interno si trovano le chiamate della funzione FC 101 ed FC 103 relative alle utenze Luci e Forze (cioè prese, aspirazione, clima, motore telo) presenti in ufficio.

L'ufficio è composto da un totale di 50 utenze. Sono state suddivise per tipo in modo da agevolare il procedimento di gestione e di relativo comando. Elenchiamo queste utenze di seguito:

- 27 denominate di tipo Luce (4 sono Neon e in seguito vedremo la diversa gestione)
- 6 di tipo Prese
- 2 di tipo Clima
- 2 di tipo Aspirazione

- 2 di tipo Motore
- 8 di tipo Radar e Allarme
- 3 analogiche per la rilevazione dei dati dai Meter

Le prime 39 utenze sono di tipo digitale quindi gestite in maniera diversa dalle ultime. Il loro segnale di comando può giungere da supervisore oppure da pulsanti presenti in ufficio. nel primo caso il comando è impostato nel Pop Up comandi e può essere accensione manuale o inserimento in scenario e successiva attivazione dello stesso.

CAPITOLO 2

System Platform (ArchestrA)

2.1 Descrizione

Si tratta di un'architettura software installata su un dispositivo Server che consente di monitorare un numero elevatissimo di variabili di un impianto, gestire la connessione di PLC e PC Client.

ArchestrA utilizza in ambiente industriale la tecnologia Microsoft .NET e altre tecnologie Microsoft per fornire strumenti efficaci per la costruzione di soluzioni software di gestione delle operazioni in ambito manifatturiero, industria di processo e infrastrutture. Il risultato è un'architettura Service-Oriented (iSOA) che presenta tutte le funzionalità necessarie nelle infrastrutture manifatturiere e industriali, tra cui namespace comune, gestione ad oggetti, sicurezza industriale, elevata disponibilità e ridondanza, connessione ai sistemi d'impianto e di business, interfaccia con i client, portale web e gestione sistemi. Questa architettura è denominata anche IAS, acronimo riferito ad Industrial Application Server.

Grazie alla tecnologia ArchestrA, è possibile implementare rapidamente le applicazioni utilizzando la tecnologia ad oggetti senza la necessità di programmare. Si possono creare template adatti a qualsiasi utilizzo, che possono essere replicati per creare nuove applicazioni, semplicemente ri assemblandoli e apportandovi piccole modifiche, risparmiando tempo e diminuendo i costi di sviluppo. Le soluzioni basate sui toolkit ArchestrA consentono ai decision-maker di raggiungere gli obiettivi di business, salvaguardando gli investimenti nei sistemi esistenti e i diritti di proprietà intellettuale.

Vantaggi apportati dall'utilizzo di questa piattaforma software:

- Riduzione dei costi di implementazione e dei costi legati al ciclo di vita dell'applicazione
- Configurazione e manutenzione semplificate
- Ampia scalabilità ed elevata sicurezza

Funzionalità:

- Complessità ridotta grazie ad un modello d'impianto comune
- Ambiente di sviluppo unico e altamente espandibile
- Manutenzione semplificata grazie a strutture ad oggetti, basate su template
- Sviluppo dell'applicazione e gestione delle modifiche da remoto
- Sicurezza e comunicazione dati integrati nel sistema

Come vedremo in seguito, grazie a questa piattaforma, potremo facilmente creare una connessione tra software di supervisione (nel nostro caso InTouch della Wonderware) e "il motore dell'impianto", cioè il PLC che consente di compiere fisicamente le operazioni.

Wonderware System Platform fornisce un'unica piattaforma software scalabile per tutte le applicazioni che costituiscono i sistemi di automazione industriale quali SCADA, HMI di supervisione e MES (Manufacturing Execution Systems) / EMI (Enterprise Manufacturing Intelligence).

Al centro di System Platform vi è il “modello d’impianto”, cioè la rappresentazione logica dei processi che vengono realmente controllati e supervisionati. Grazie alla tecnologia ad oggetti ArchestrA, informazioni storiche e in tempo reale sono configurate, rese accessibili, distribuite e conservate con un semplice point-and-click.

System Platform comprende un database storico di processo ad elevate prestazioni con archiviazione dello storico di produzione, compressione dati efficiente, auto-configurazione dell’archivio storico e un information server industriale che semplifica drasticamente l’organizzazione e condivisione delle informazioni nell’intero impianto.

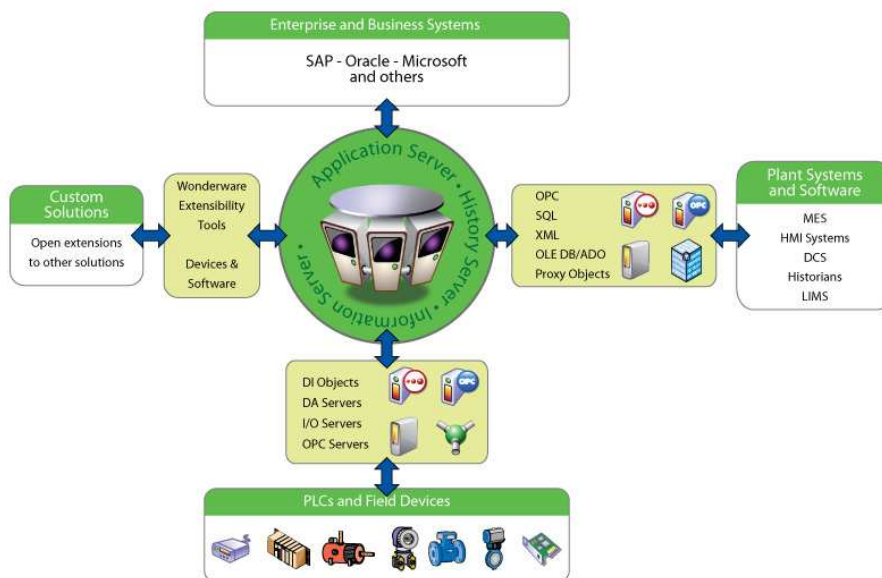


Figura 19. Illustrazione Piattaforma software

A livello di impianto l’utilizzo di una System Platform è sinonimo di incremento di produttività e di gestione della produzione stessa.

Nella figura seguente è riportata una schematizzazione delle diverse fasi di una possibile linea di produzione. In verde è rappresentata la produzione in senso lato, cioè le fasi di lavorazione del materiale (lavorazione materia prima) per consentire di ottenere il prodotto finito. In arancione invece la System Platform stessa con tutte le sue funzionalità e le configurazioni necessarie.

Infine tutte le operazioni che avvengono per compiere tali configurazioni sono rappresentate in viola nella figura ... in questa sezione sono inclusi anche i dispositivi utilizzati per il compimento del risultato cioè PLC, PC, dispositivi portatili, ecc.

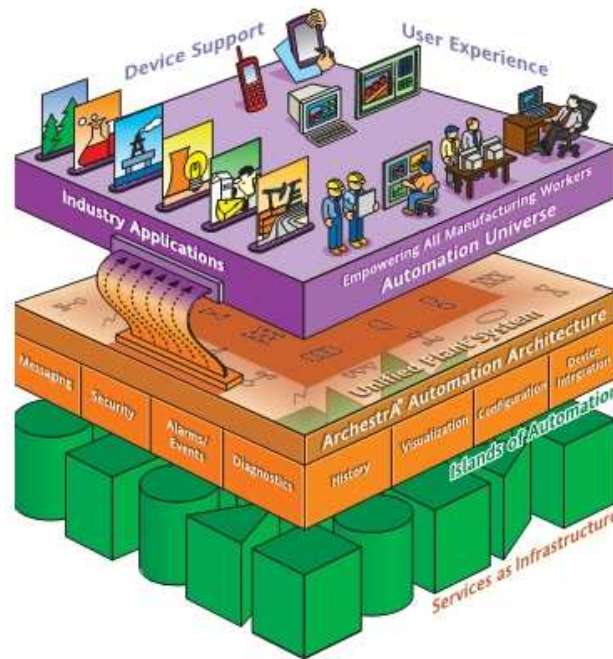


Figura 20. Piattaforma software

2.2 Componenti utilizzati nel progetto

Passiamo ora a descrivere in dettaglio la procedura seguita per compiere la configurazione relativa al nostro progetto.

Il software consente di creare degli “oggetti” che rappresenteranno i dispositivi presenti nel nostro impianto. Grazie ad una serie di Template preconfigurati è possibile creare delle Istanze relative ad ognuno di essi. In questo modo ogni oggetto fisico che ci interessa comandare e controllare disporrà di un suo oggetto astratto che lo rappresenta nel migliore dei modi.

Definiamo il concetto di oggetto illustrando quelli presenti nel nostro impianto.

1. Meter (nel nostro caso 3 strumenti di rilevazione dati della Socomec)
2. PLC
3. Sensori di presenza

Illustriamo in figura... la sezione Template. Necessario è innanzitutto descrivere i singoli componenti in evidenza, per meglio comprendere la loro funzionalità.



Galaxy

Con il termine Galaxy si considera l'intera applicazione, il sistema di Archestra completo, costituito da un unico spazio, nel nostro caso denominato *UfficiAW_CEM*, ed una raccolta di *Winplatforms (AppEngines e Objects)*.



Area

Con il termine Area intendiamo l'oggetto della piattaforma che rappresenta una sezione fisica di un impianto o di un processo di automazione. Può essere intesa anche come una parte definita di una applicazione industriale.

Lo scopo principale di una zona è quello di organizzare tutti gli altri tipi di Automation Objects in modo logico per imitare così il più possibile la struttura fisica. Le Aree sopportano una composizione di tipo gerarchica, consentono cioè le sottostrutture (sub-area). E' classificato come oggetto di sistema.

Forniscono inoltre un ruolo chiave nel raggruppamento di allarmi ed eventi, permettendo agli operatori che ne fanno uso di monitorare le zone interessate.



AppEngine

L'oggetto AppEngine rappresenta il motore, cioè il mezzo che ospita ed esegue oggetti di Area, oggetti DeviceIntegration ed ApplicationObjects.



WinPlatform

L'oggetto WinPlatform fornisce le seguenti funzioni:

- monitoraggio di vari dispositivi collegati al computer su cui è montato. Molti dati monitorati possono essere storicizzati e successivamente è possibile creare delle statistiche
- avviare e fermare oggetti quali AppEngine
- controllare lo stato corrente degli oggetti che vengono impiegati nella piattaforma. Possibilità di configurare il WinPlatform in modo da riavviare l'oggetto AppEngine in caso di rilevazione di anomaliedi funzionamento del motore



ViewEngine

Utilizzato per ospitare un ulteriore oggetto necessario ad avviare il WindowViewer e monitorare le variabili gestite dalla piattaforma. Questo oggetto è descritto al punto successivo e si tratta del InTouchViewApp.

ViewEngine supporta caratteristiche comuni al motore (AppEngine) quali: undeployment, avvio e spegnimento. Il ViewEngine non supporta ridondanza.



InTouchViewApp

Quest'oggetto rappresenta un'applicazione in ambiente Wonderware Application Server. È necessario a gestire il check-in, check-out, e il dispiegamento di una domanda di InTouch, cioè una modifica compiuta dall'operatore a livello di supervisore (WindowViewer).

L'oggetto InTouchViewApp viene quindi utilizzato per integrare le applicazioni InTouch in un sistema globale definito in una Galaxy dell'Application Server.



S7C

L'oggetto S7C appartiene ad una famiglia di oggetti chiamati Device Integration (DIObjects). E' un oggetto che viene utilizzato per rappresentare la connessione TCP / IP ad una porta di una serie di controllori S7-200/300/400 Siemens, ognuno dei quali è rappresentato da un AutomationObject.

Utilizza il protocollo di comunicazione Ethernet TCP / IP per comunicare con la famiglia di controllori S7-200/300/400 Siemens. Non supporta MPI, Profibus, e

altri protocolli non-Ethernet. Può essere utilizzato con schede di interfaccia di rete standard.

Questo oggetto è utilizzato per configurare il bus di comunicazione e per monitorare o controllare le sue prestazioni in fase di esecuzione. Il DINetwork controlla le comunicazioni con il server di accesso ai dati e lo stato del percorso di comunicazione.



PLC

Rappresenta il PLC fisico. Nel nostro caso Siemens S7-300

La piattaforma IAS possiede un sistema di monitoraggio variabili e restituzione di commenti in caso di errori nel procedimento di test in WindowViewer. Il Software è fornito sempre dalla Wonderware e denominato System Management Console (SMC).

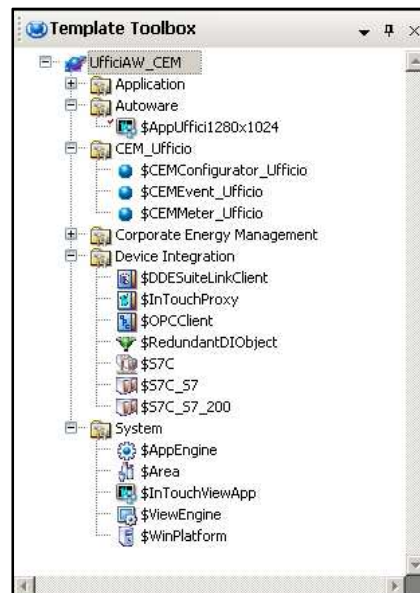


Figura 21. Template preconfigurati

2.3 Struttura della piattaforma

Una volta compreso come utilizzare la sezione ToolBox è stato sufficiente creare le istanze dei blocchi che ci interessano e assegnarli correttamente all'interno della WinPlatform. Ogni oggetto è stato inserito nell'Area corretta e successivamente è stato compiuto un Deploy.

Con questo termine intendiamo l'operazione di codifica delle operazioni da compiere per rendere il tutto comprensibile al "motore" della piattaforma che dovrà compiere le dovute azioni.

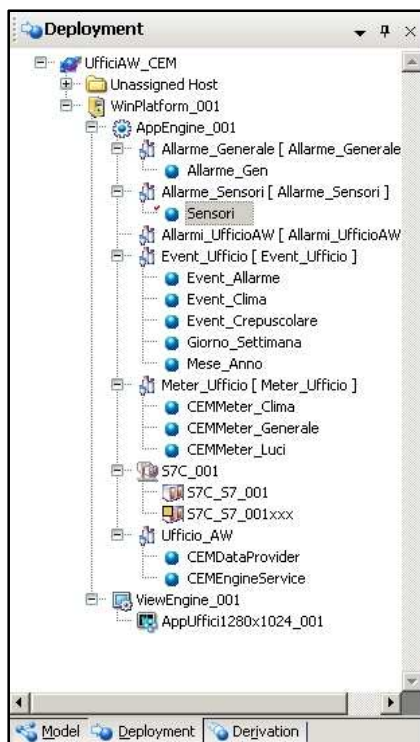


Figura 22. Istanze totali dell'impianto

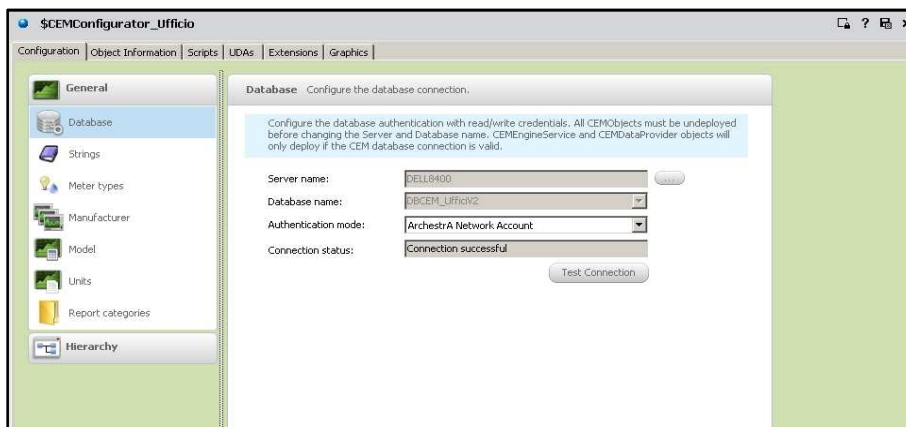


Figura 23. CEMConfigurator_Ufficio

Innanzitutto, non appena sono state create le istanze, passo fondamentale da compiere è stato quello di configurare correttamente il *CEMConfigurator_Ufficio*, che possiamo trovare in figura.. al paragrafo precedente. In figura... invece vediamo la schermata che appare in IAS compiendo un doppio click sull'oggetto in questione.

Nella sezione Database sarà necessario impostare il corretto Server Name, cioè quello nel quale è inserito il DB, il nome del DB stesso e l'autenticazione per effettuare l'accesso ai dati.

Il software di storicizzazione dei dati è SQL Server 2005 che descriveremo più in dettaglio al prossimo capitolo.

Ulteriore impostazione da compiere è la definizione delle grandezze, unità di misura e variabili che interessano il nostro impianto. Tutto ciò nella sezione Strings.

2.3.1 Configurazione del PLC

Indispensabile punto è la configurazione dell'oggetto PLC. La comunicazione tra esso ed il PC è di tipo Ethernet, dovremo quindi impostare il corretto indirizzo IP grazie al quale verranno scambiati i dati.

Nella figura seguente è riportata la schermata di configurazione Generale, si nota dunque il valore di indirizzo IP impostato.

Da tenere in considerazione è il fatto che l'algoritmo di controllo creato in Step 7 non è stato caricato subito nel PLC montato in ufficio. Una fase di test e collaudo del supervisore era necessaria. Come vedremo nei prossimi capitoli è stato utilizzato un PLC virtuale di tipo Software (WinLC) che ci ha consentito di simulare il comportamento del controllore reale e compiere le nostre modifiche senza incorrere in problemi in caso di errata stesura dell'algoritmo.

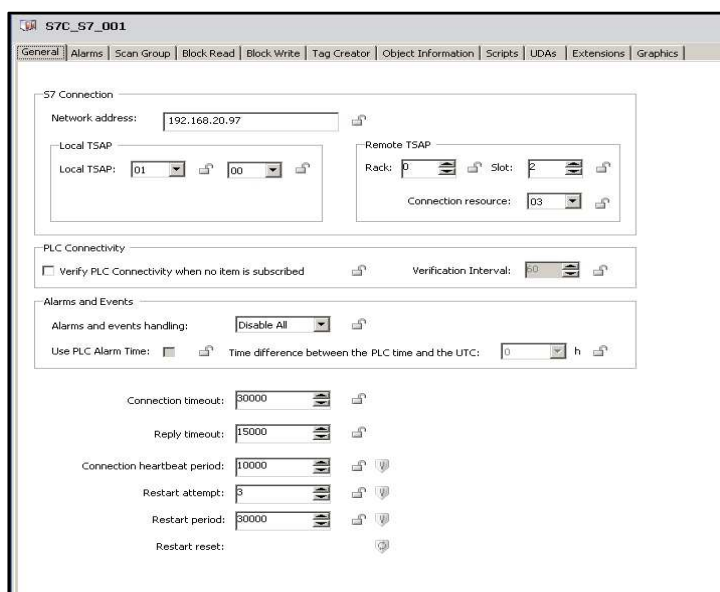


Figura 24. PLC – Generale

Siamo giunti ad un punto cruciale ed ora, una volta capito come il PLC comunica con il PC (e quindi con il supervisore), possiamo porci una domanda: come riconosce il PLC le variabili create e modificate nel supervisore?

Per comprendere al meglio la questione dobbiamo illustrare i diversi tipi di variabili che interessano il nostro impianto e che devono essere gestite dal controllore.

- IN – Variabili di ingresso. Di tipo BOOL, sono fisicamente connesse al PLC e tipicamente possono essere segnali provenienti da sensori. Nel nostro caso gli ingressi sono i segnali provenienti dai sensori di presenza (denominato Radar nell'algoritmo Step7) e il segnale dell'allarme.
- Out – Variabili di uscita. Sempre di tipo BOOL, ed anch'esse fisicamente connesse al PLC. Riguardano tutte le utenze d'ufficio quali luci, neon, aspirazione e clima.
- Variabili interne contenute in DB

Come abbiamo già visto in precedenza la struttura di Step7 riguardo a variabili di tipo BOOL è di 8 bit. Quindi che si parli di ingressi, uscite oppure variabili interne contenute in DB la forma non cambia. Riportiamo 3 esempi per una diretta comprensione.

- Prima variabile di IN del PLC: **E 0.0**
- Ottava variabile di Out del PLC: **A 0.7**
- Dodicesima variabile posta in *DB Utenze*: **DB1.DBX 1.3**

Una volta creato il Supervisore e tutte le variabili interne indispensabili al suo corretto funzionamento è stato necessario creare i “collegamenti” delle stesse con il PLC. Nella figura... è ripotata appunto la pagina relativa alla configurazione delle variabili presenti nel controllore. Nel nostro caso il PLC sarà unico ma è possibile impostarne svariate decine tutti con le loro variabili interne.

Nella colonna Attribute impostiamo i nomi delle variabili in ambiente IAS mentre in Item Reference la loro assegnazione al PLC. Una variabile presente in questa lista verrà richiamata da Supervisore come *Galaxy:S7C_S7_001.S7C_001.nome variabile*.

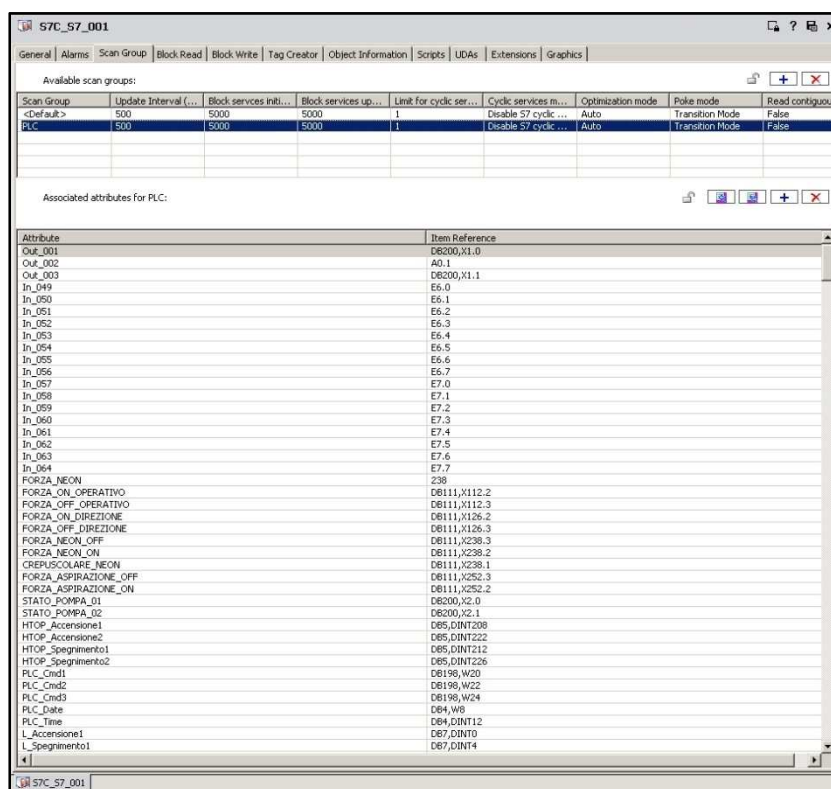


Figura 25. PLC – Scan Group

In totale le variabili create sono oltre 500 ma non sono le totali che vengono utilizzate nel progetto. Infatti il Supervisore, creato con WindowMaker, utilizza variabili in ambiente InTouch (cioè interne al programma) e le oltre 500 variabili create in ambiente IAS che saranno viste come *Galaxy:...* come accennato sopra.

Particolare importanza è posta riguardo alla rilevazione dei consumi di Energia rilevati dai 3 Meter. Dato che il fondoscala del dispositivo è 9999 Watt è stato necessario creare un semplice algoritmo per poter rilevare valori superiori a tale numero. Sono state quindi create delle variabili di tipo UDA, User Define Attribute, e create le righe di codice necessarie a compiere le operazioni desiderate.

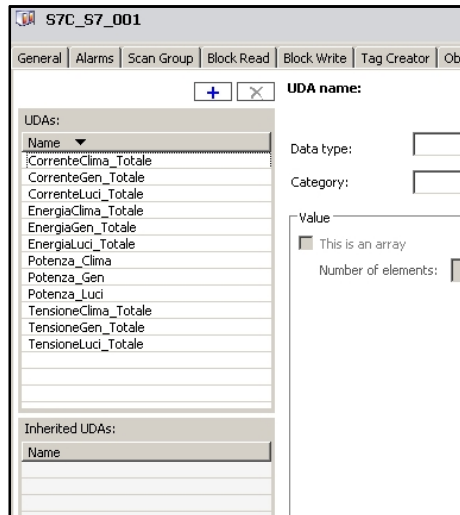


Figura 26. PLC – UDAs

Di seguito è riportato un esempio di Script associato alle variabili UDA sopra citate.

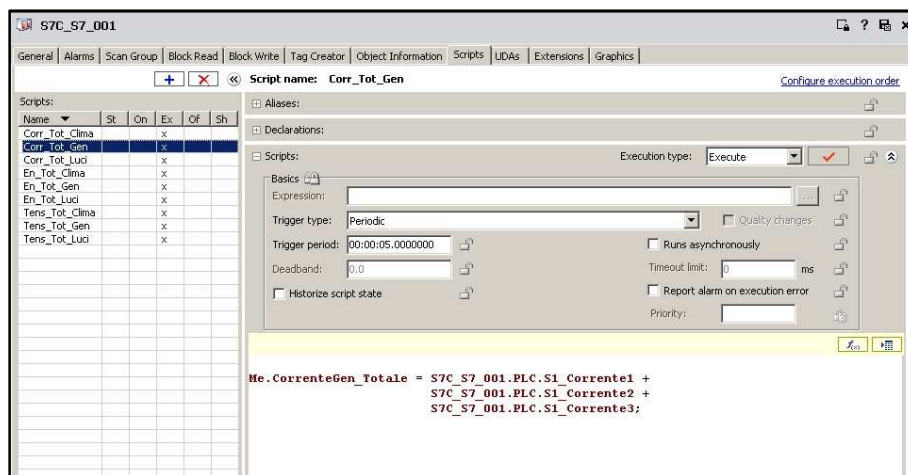


Figura 27. PLC – Scripts

2.3.2 Configurazione Meter

Per quanto riguarda questo oggetto le configurazioni da compiere sono riportate nella pagina denominata appunto *Configuration*. Riportiamo nella figura seguente la configurazione relativa al *Meter Generale*.

Sarà necessario impostare inizialmente il tipo di misurazione che deve essere compiuta, cioè la grandezza di interesse. Nel nostro caso i Meter sono 3 ed ognuno di essi dovrà rilevare valori di Potenza, Tensione, Corrente ed Energia dell'impianto in questione.

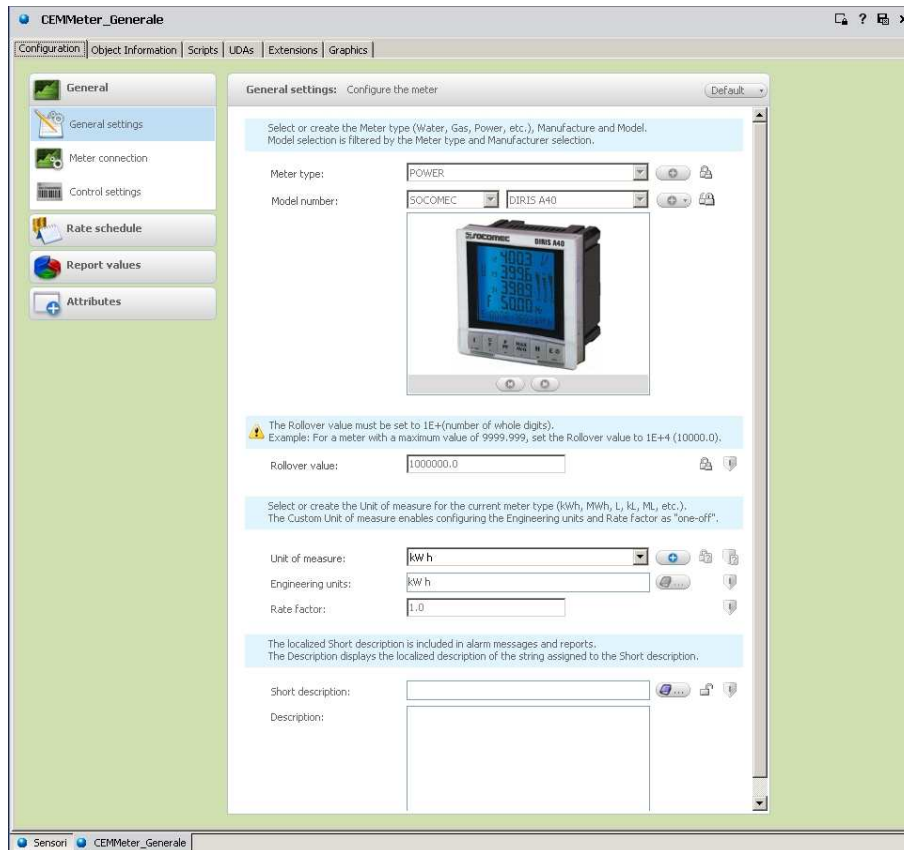


Figura 28. Configurazione Meter

Per ognuno dei tre Meter le accortezze da tenere in considerazione saranno diverse. Come specificato in precedenza una delle funzioni della System Platform è quella di monitorare e storicizzare. Con l'aggiunta del pacchetto CEM questo diventa ancora più importante dato che include la possibilità di creare dei Report in seguito alla rilevazione di dati. In questo caso sarà necessario associare ad ogni Oggetto Meter un determinato Evento in funzione del quale verranno storicizzati i dati per essere in seguito analizzati.

2.3.3 Configurazione Eventi

Gli Eventi vengono considerati come degli oggetti configurabili, lo stesso che accade per gli altri componenti che rappresentano parti fisiche della piattaforma.

Questo tipo di oggetto viene utilizzato per simulare l'avvenimento di una condizione in seguito alla quale si desidera monitorare la variazione di determinate variabili. Questa condizione può essere ad esempio l'attivazione di qualche dispositivo ed una possibile variabile che può interessare al monitoraggio è l'Energia consumata, o meglio l'incremento di consumo causato.

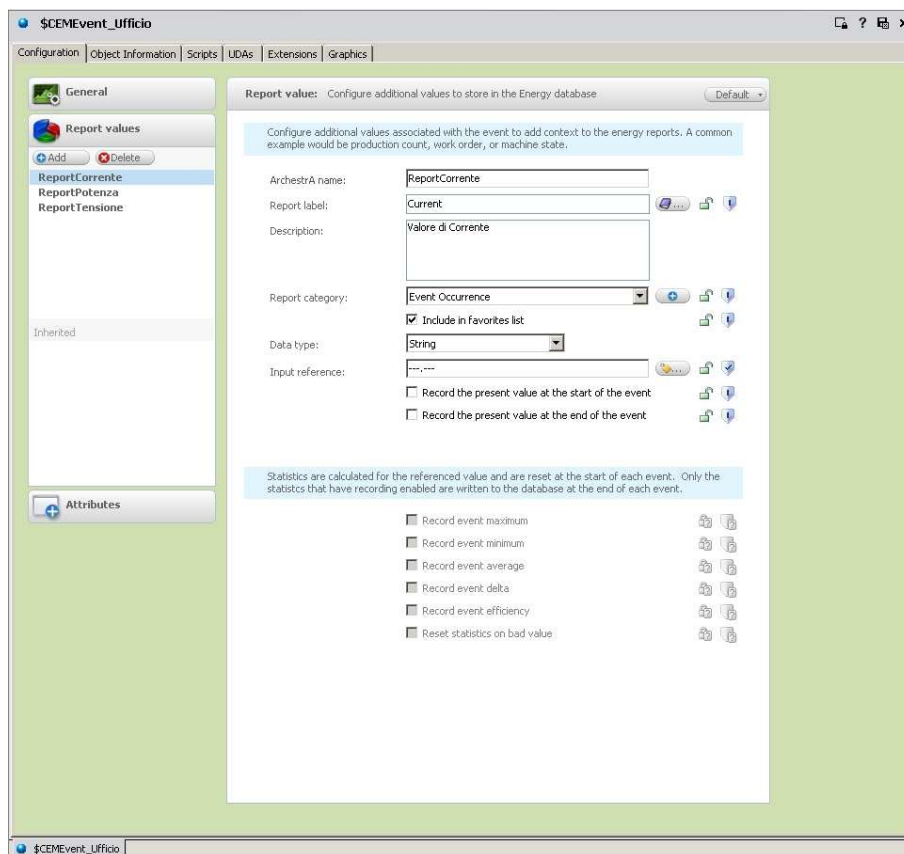


Figura 29. Configurazione Eventi

Gli eventi configurati in questa esperienza sono 5. Li elenchiamo di seguito:

- Attivazione Allarme
- Attivazione impianto di climatizzazione
- Presenza o meno di crepuscolare
- Consumo per giorno settimana
- Consumo per mese all'anno

Grazie alla funzionalità del software della Wonderware utilizzato per il progetto è stato possibile scegliere le grandezze che si desidera maggiormente monitorare. È possibile vederlo alla sezione *Report Values*, in figura 28, dove sono state configurate 3 voci relative alle suddette grandezze.

Alla voce *Report Category* è stato assegnato *Event Occurrence*, quindi nel caso ad esempio di attivazione allarme al passaggio del bit **E 7.7** ad 1 verrà avviata la storicizzazione dei dati.

Da precisare che ad ogni Evento verrà associato uno o più Meter al quale fare riferimento. La sola presenza di oggetto Meter comunque garantisce che avvenga una storicizzazione costante dei dati misurati in una determinata posizione del DataBase.

Quindi in caso di occorrenza Evento i dati verranno posti in posizione diverse che andremo ad illustrare al capitolo relativo a SQL Server 2005.

alla sezione *General – Meter Connection* saranno associate le 3 grandezze, Corrente, Tensione o Potenza, ad uno specifico Meter. Quindi relativamente all'Evento Crepuscolare verrà associato il solo Meter relativo all'impianto di illuminazione.

2.3.4 Configurazione Allarmi

All'interno dell'oggetto Area *Ufficio_AW* è stata creata una sub – Area chiamata *Allarmi_Ufficio_AW*. Infine all'interno di quest'ultima altre due Aree una relativa all'allarme generale mentre una ai sensori presenti in ufficio.

In entrambi i casi sono state create delle variabili UDA che che andranno a puntare alla variabile corretta direttamente nel PLC.

Riportiamo in figura ... la configurazione dell'oggetto *Sensori*.

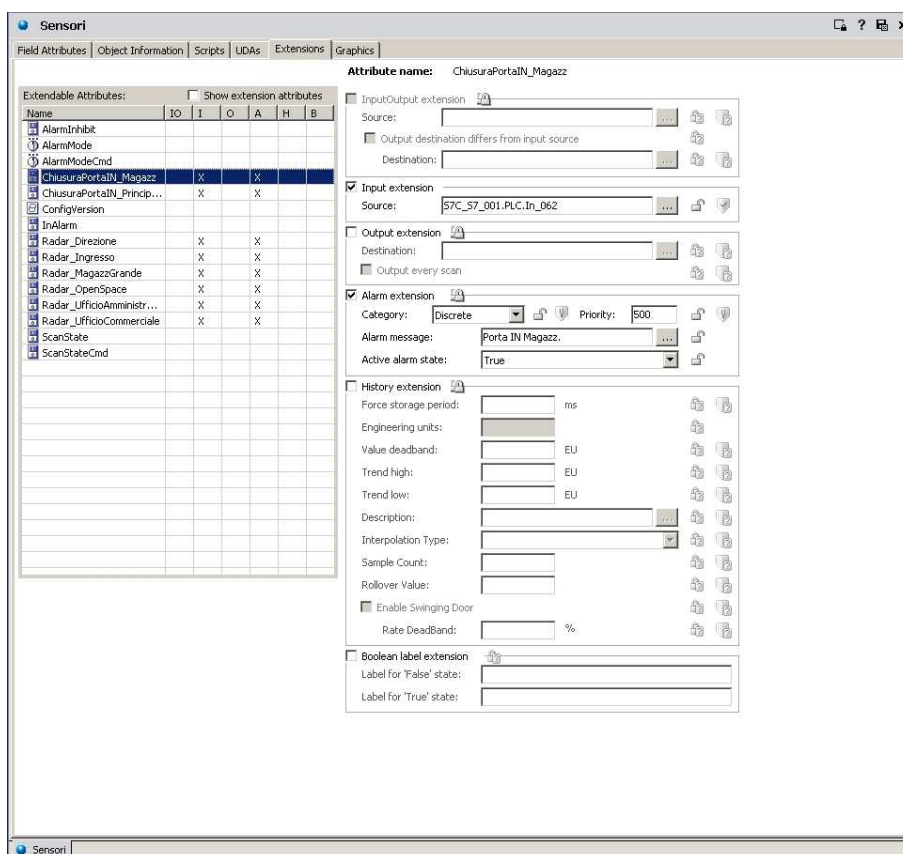


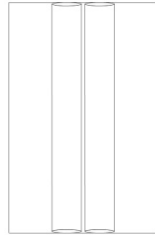
Figura 30. UDAs Sensori e loro estensione

2.4 Editor grafico. Figure create

Grazie all'editor grafico di ArcestraA è stato possibile creare facilmente delle figure che successivamente andranno importate in InTouch e consentiranno di visualizzare la variazione di parametri stabiliti.

Ogni figura conterrà appositi Scripts che comporteranno visualizzazione di immagini, variazioni di colore o fenomeni di lampeggio.

Plafoniera:



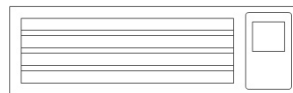
Faretti:



Ventola Aspirazione:



Climatizzatore:



Prese:



Neon:



Tutte queste figure sono illustrate nelle figure seguenti in fase di descrizione del software di supervisione.

CAPITOLO 3

SQL Server 2005

3.1 Descrizione

Microsoft SQL Server è un DBMS relazionale, meglio noto come Relational Database Management System (RDBMS), prodotto da Microsoft. Nelle prime versioni era utilizzato per basi dati medio-piccole, ma a partire dalla versione 2000 è stato utilizzato anche per la gestione di basi dati di grandi dimensioni.

Un database di SQL Server 2005 è costituito da un insieme di oggetti che contengono dati, ad esempio viste e tabelle, e da altri oggetti che legano tra loro questi dati, quali indici, stored procedure, funzioni definite dall'utente e trigger.

Alle tabelle che contengono i dati sono associati numerosi tipi di controlli, ad esempio vincoli, trigger, valori predefiniti e tipi di dati utente personalizzati, che garantiscono la validità dei dati. È possibile aggiungere vincoli di integrità referenziale dichiarativa (DRI, Declarative Referential Integrity) alle tabelle per garantire che i dati correlati delle diverse tabelle rimangano consistenti. Nelle tabelle è possibile includere indici, con caratteristiche molto simili a quelli dei libri, che consentono di trovare rapidamente le righe. Un database può, inoltre, contenere procedure che utilizzano codice di programmazione Transact-SQL o .NET Framework per eseguire operazioni con i dati del database. Tali operazioni includono la creazione di viste che offrono accesso personalizzato ai dati della tabella o di una funzione definita dall'utente per l'esecuzione di un calcolo complesso in un subset di righe.

Un'istanza di SQL Server può supportare più database. Ogni database consente di archiviare dati correlati o non correlati di altri database. In un'istanza di SQL Server possono, ad esempio, essere presenti un database in cui sono archiviati i dati sul personale e un altro in cui sono archiviati i dati relativi ai prodotti. In alternativa, è possibile archiviare in un database i dati aggiornati sugli ordini dei clienti e in un altro database correlato la cronologia degli ordini da utilizzare per la creazione di report annuali.

Per compiere la connessione tra Supervisore, visualizzatore di dati, e DataBase, serbatoio di dati, è necessario conoscere il concetto di Query e di Store Procedure.

3.2 Query create

Tramite il software di supervisione InTouch è possibile richiamare dei segmenti di codice i quali consentiranno di compiere la connessione con un DataBase, leggere una serie di dati che interessano ed elaborarne dei risultati.

Questi segmenti di codice sono stati inseriti nella sezione *Action* presente nel pannello di WindowMaker riguardante gli *Animation Link*. Dette Query, queste sequenze di istruzioni non fanno altro che fornire al supervisore i parametri corretti per compiere la connessione con il DataBase interessato ed avviare successivamente delle possibili Stored Procedure per elaborare dei dati.

Ecco quindi che possiamo introdurre una nuova funzione consentita dagli Scripts del software InTouch.

Andremo ora ad analizzare in dettaglio la formulazione di queste Query. Inizialmente riportiamo la più utilizzata a livello di supervisore. È posta nell'Action dell'Animation Link relativo al calcolo dei valori (massimi, minimi e medi) ricavati da uno dei tre Meter presenti in ufficio. Quindi sarà avviata alla pressione del pulsante **Calcola** situato nella Windows – Mappa Consumi, figura ... capitolo 3.

```

SQL_MsgError=SQLErrorMsg(SQLConnect(ConnectionId_01, "DSN=ENEA;UID=sa;PWD=autoware" ));

LogMessage(SQL_MsgError);

SQL_MsgError=SQLErrorMsg(SQLSetStatement(ConnectionId_01, "EXEC sp_MeterGEN " + Var_app +", " +
Consumi_Y + Consumi_M + Consumi_D +", " + Consumi_Y1 + Consumi_M1 + Consumi_D1 +"" ));

LogMessage("EXEC sp_MeterGEN " + Consumi_Y + Consumi_M + Consumi_D +", " + Consumi_Y1 + Consumi_M1 +
Consumi_D1 +""));

LogMessage(SQL_MsgError);
SQLExecute( ConnectionId_01, "BindList_MeterGEN", 0 );
SQLFirst(ConnectionId_01);
SQLEnd(ConnectionId_01);
SQLDisconnect( ConnectionId_01 );

```

3.3 Store Procedure

Per sfruttare le potenzialità di SQL Server, viene messo a disposizione un linguaggio di programmazione chiamato T-SQL.

Durante lo sviluppo di una applicazione Client le richieste per l'interrogazione di dati possono essere inviate direttamente al database, dal programma, tramite istruzioni di INSERT, SELECT, UPDATE o altro ancora, oppure è possibile creare piccoli programmi o funzioni all'interno del database e poi farle eseguire dal programma esterno. Queste procedure si chiamano Stored Procedure.

Si tratta di moduli o routine che incapsulano codice in modo da poterlo riutilizzare. Una stored procedure può accettare parametri di input, restituire al client risultati tabulari o scalari, richiamare istruzioni DDL (Data Definition Language) e DML (Data Manipulation Language) e restituire parametri di output. Le stored procedure di SQL Server 2005 possono essere di due tipi: Transact-SQL o CLR.

Come per un programma, anche una Stored Procedure è in grado di accettare o ricevere parametri, restituire valori calcolati, effettuare scelte. Questo oggetto è utilissimo per diversi motivi tra cui la sicurezza e le performance. Per questi motivi, specialmente nello sviluppo di applicazioni client/server, le Stored Procedure dovrebbero essere preferite alle semplici istruzioni SQL.

Illustriamo le Store Procedure richiamate maggiormente nel nostro progetto. Di seguito è riportata quella denominata *sp_MeterGEN*

```

USE [ENEA]
GO
/* Object: StoredProcedure [dbo].[sp_MeterGEN]  Script Date: 09/21/2010
...
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:                <Author,,Name>
-- Create date: <Create Date,,>
-- Description: <Description,,>
-- =====

```



```

ALTER PROCEDURE [dbo].[sp_MeterGEN]
@varIntouch varchar(50),
@varDateStart varchar(50),
@varDateEnd varchar(50)

AS

/* Dichiarazione variabili */

DECLARE @StartDate DATETIME
DECLARE @EndDate DATETIME
DECLARE @value as integer

BEGIN
SET @StartDate = @varDateStart
SET @EndDate = @varDateEnd
END

SELECT max(value) as massimo,min(value) as minimo,avg(value) as media
FROM Runtime.dbo.History
WHERE TagName IN ( '' + @varIntouch + '' )
and DateTime >= @StartDate
AND DateTime <= @EndDate
AND value is not null
AND wwResolution = 6000

```

La prima parte di codice, pagina precedente, è necessaria all'assegnazione del nome, del DataBase in cui è inserita la Store Procedure e alle variabili passate dal supervisore. In questo caso infatti tramite la Query, che mostreremo in seguito, l'operatore potrà scegliere la variabile da analizzare e l'intervallo di tempo secondo il quale prelevare i valori.

Queste variabili corrispondono rispettivamente a:

- @vaIntouch
- @StartDate
- @EndDate

La seconda parte di codice invece effettua il vero e proprio calcolo di valore estrapolando i valori presenti nel DataBase *Runtime.dbo.History* in base a quanto assegnato come variabili passate dal supervisore.

Il risultato di tale Store Procedure è una tabella composta da una singola riga e da 3 colonne denominate *massimo*, *minimo* e *media*.

```

USE [ENEA]
GO
/* Object: StoredProcedure [dbo].[sp_Eventi] Script Date: 09/21/2010 ...
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- =====
-- Author:          <Author,,Name>
-- Create date:    <Create Date,,>
-- Description:    <Description,,>
-- =====

ALTER PROCEDURE [dbo].[sp_Eventi]
@Grandezza varchar(50),
@TypeEvent varchar(50),
@varDateStart varchar(50),
@varDateEnd varchar(50)

AS

/* Dichiarazione variabili */

```

```

DECLARE @StartDate DATETIME
DECLARE @EndDate DATETIME
DECLARE @value as integer

BEGIN
SET @StartDate = @varDateStart
SET @EndDate = @varDateEnd

select t1.id,t1.value,t3.time,t4.cost,t4.start_time,t4.end_time,
t5.event_name,t6.name
FROM DBCEM_UfficiV2.dbo.event_data_value as t1
inner join DBCEM_UfficiV2.dbo.event_data as t2
on t1.event_data_id=t2.id
inner join DBCEM_UfficiV2.dbo.events_sub as t3
on t2.events_sub_id=t3.id
inner join DBCEM_UfficiV2.dbo.events as t4
on t3.events_id=t4.id
inner join DBCEM_UfficiV2.dbo.event as t5
on t4.event_id=t5.id
inner join DBCEM_UfficiV2.dbo.strings as t6
on t2.name_str_id=t6.id

where t4.event_id IN ( '' + @TypeEvent + '' )
and t6.name IN ( '' + @Grandezza + '' )
and time >= @StartDate
and time <= @EndDate

END

```

In questa seconda Store Procedure le variabili passate dal supervisore sono 4. Le ultime due come prima sono data di inizio e data di fine mentre la prima contiene il tipo di evento compiuto e la seconda la grandezza di interesse.

Come si nota, nel segmento di codice sopra riportato, per ottenere il valore finale relativo al consumo in Watt, Volt oppure Ampere è necessario compiere una sovrapposizione di indici di diverse tabelle. L'utilizzo dei DataBase quindi non è di immediata comprensione.

Un'altro Database di rilevante importanza è il WWALMDB che contiene tutti i dati relativi agli allarmi che verranno descritti in seguito.

Inseriamo una figura che illustra la struttura del software SQL Server.

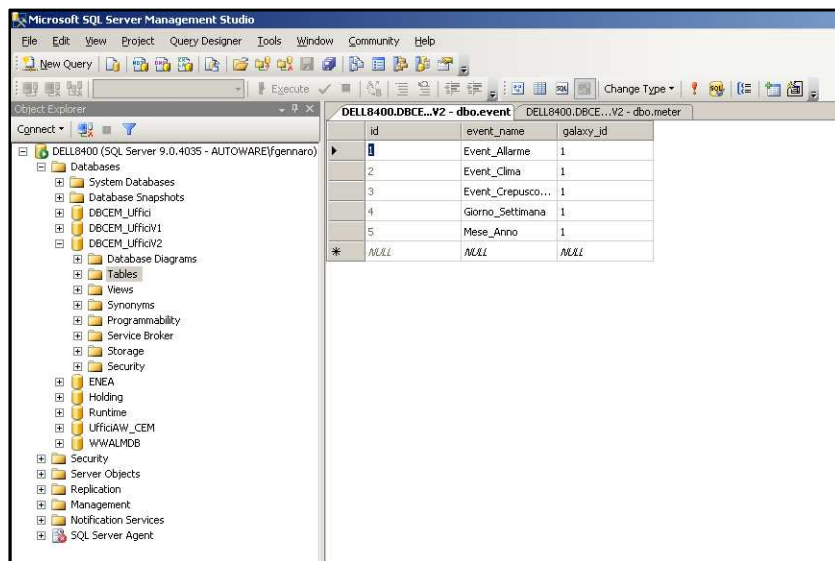


Figura 31. SQL Server 2005

CAPITOLO 4

Corporate Energy Management

4.1 Utilizzo del CEM

La Wonderware ha fornito un pacchetto software che funziona in ambiente IAS. Installando tale pacchetto verranno aggiunti dei Toolbox e dei Symbol non presenti di default in IAS stesso.

Grazie a queste aggiunte è stato possibile creare le istanze degli oggetti che abbiamo visto in figura 20 e la loro successiva configurazione.

I blocchi fondamentali sono il Configurator, grazie al quale è stato creato il legame con il Database nel quale verranno immagazzinati i dati. Specifichiamo che alla base del CEM sta il concetto di ottimizzazione della produzione una volta conosciuti i valori di consumo e il prezzo unitario dello stesso. Inserendo, nelle posizioni che in seguito mostreremo, i valori di Moneta corrente sarà possibile calcolare i costi di gestione di un impianto.

La programmazione della piattaforma fa sì che una volta creato il Database non sia più necessario compiere altre operazioni per memorizzare i dati. Basterà infatti creare delle specifiche Query per ricavare i valori di interesse.

Nel nostro caso è stato impostato un valore di 15 minuti relativo alla storicizzazione dei dati ricevuti dai Meter a livello di CEM. La piattaforma invece compie costantemente, ogni millisecondo, un monitoraggio di tutte le variabili create.

Il nome assegnato al DataBase è CEM_UfficiV2. Inizialmente esso è vuoto ma strutturato in maniera standard come impostato in fase di programmazione del CEM.

All'interno della sottocartella Table troveremo molte voci e quelle di maggior interesse nel nostro progetto sono:

- *dbo.event*
- *dbo.event_data*
- *dbo.event_data_value*
- *dbo.events*
- *dbo.events_sub*
- *dbo.meter*
- *dbo.meter_data*
- *dbo.meter_data_value*
- *dbo.meter_value*
- *dbo.strings*

Come abbiamo visto al Capitolo 4, paragrafo riferito alle Store Procedure, la procedura per ricavare i dati all'interno del DataBase non è immediata. Bisogna infatti capire come mettere in relazione i vari campi delle diverse Table presenti.

Innanzitutto è necessario sapere che i dati rilevati dai Meter che interessano in fase di statistiche sono contenuti nelle sottocartelle con nome che termina in *..value*. Passo successivo è associare....

Per ricavare, ad esempio, il valore di Potenza fornito da un Meter è necessario associare al campo *dbo.meter_data_value.id* l'id stesso riferito a *dbo.meter*.

Precedentemente è stato illustrato il CEM e tutte le sue funzionalità. Riportiamo ora quelle che più interessano nel nostro progetto.

Meter ed Eventi

Al capitolo precedente sono state descritte le impostazioni compiute a livello di Meter e di Eventi. Queste sono frutto della presenza del pacchetto CEM il quale ha consentito l’inserimento dei suddetti oggetti in ambiente IAS e la loro successiva configurazione.

Costi e Fascie Orarie

Alla sezione *Rate Schedule* nell’oggetto Meter è possibile impostare a piacimento delle diverse fasce orarie con costi diversi. Questa differenziazione dipende dal fornitore di energia, esso dispone delle proprio proposte e dei propri prezzi. In questo caso la differenziazione dei costi è così suddivisa:

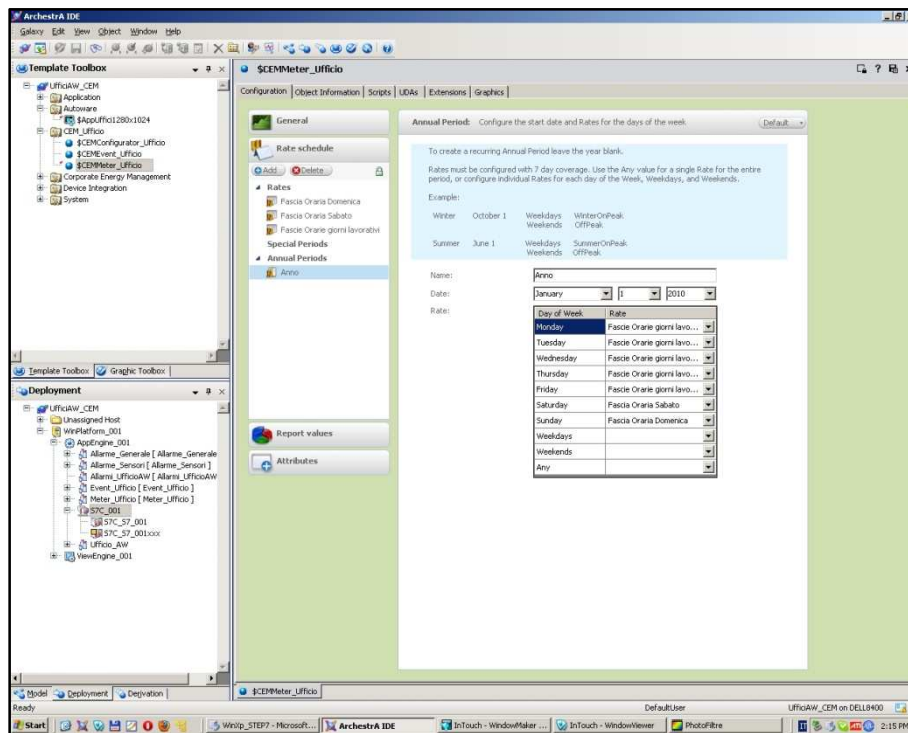


Figura 32. Impostazione Fascie di costo

Tramite la lettura sul DataBase è possibile ricavare i valori di costo e confrontarli successivamente con quelli riportati sulle bollette.

CAPITOLO 5

Software di supervisione

5.1 Descrizione del Software

Il software InTouch offre funzioni di visualizzazione grafica vettoriale che portano ad una gestione, controllo e ottimizzazione dei sinottici sempre più vicino alla visione reale di impianto, aumentando l'efficienza operativa.

InTouch HMI vanta una reputazione eccellente e consolidata. Quelle che oggi sono conosciute come interfacce uomo-macchina (HMI) hanno come pioniere InTouch di Wonderware, da oltre vent'anni presente sul mercato. Nessuna soluzione HMI è paragonabile al software InTouch in termini di innovazione, robustezza, integrazione e connettività dei dispositivi, percorso di migrazione dalle versioni software precedenti e facilità d'uso.

Tutto questo si traduce in sistemi accuratamente progettati e basati su standard che massimizzano la produttività, ottimizzano l'efficacia dell'utente, migliorano la qualità e riducono i costi di sviluppo, manutenzione e utilizzo, aiutando a ottenere il meglio dalla propria azienda.

Il software InTouch è una soluzione HMI aperta e scalabile, dotata di funzionalità grafiche che consentono notevoli potenzialità e flessibilità nella progettazione di applicazioni, grazie alla connettività con una vastissima gamma di sistemi e dispositivi industriali.

Wonderware InTouch HMI offre le funzionalità di cui ogni utente ha bisogno per ottenere significativi miglioramenti nella produttività delle attività operative o di engineering, sia che si tratti di personale di produzione o progettazione, utenti finali, system integrator o costruttori di macchine.

Questo software Wonderware è considerato il più diffuso HMI al mondo utilizzato in più di un

terzo degli impianti su scala globale, in ogni settore e area geografica. Si tratta in definitiva di un software che viene installato sui PC a bordo macchina di un qualsiasi impianto. Consente di monitorare, storicizzare e modificare variabili interessate.

Vantaggi apportati nell'utilizzo del software in questione:

- La straordinaria facilità d'uso consente a sviluppatori e operatori di diventare produttivi in minor tempo e più facilmente
- Integrazione dei dispositivi e connettività senza paragoni praticamente per ogni tipo di sistema e dispositivo
- La rappresentazione grafica vettoriale, estremamente sofisticata e l'interazione avanzata con l'impianto mettono a disposizione le informazioni giuste alle persone che ne hanno necessità al momento appropriato
- La storicizzazione dei percorsi di migrazione dalle versioni software precedenti assicura la protezione degli investimenti nelle applicazioni HMI

Funzionalità:

- Indipendente dalla risoluzione grafica, grazie agli "ArchestrA Symbol" (grafica vettoriale) che "danno vita" all'impianto

- Script sofisticati per espandere e personalizzare le applicazioni in base alle proprie esigenze
- Allarmi distribuiti in tempo reale con viste storiche
- Trend integrati, in tempo reale e storici
- Integrazione con controlli Microsoft ActiveX e .NET
- Ricca libreria con oltre 500 simboli grafici vettoriali e personalizzabili

Per quanto riguarda il progetto in questione il software di supervisione InTouch è stato installato in un PC Server per controllare l'ufficio.

Grazie ad un ulteriore software della Google, Sketchup, è stato possibile creare l'ambientazione il più verosimile possibile alla realtà.

5.2 WindowMaker per la creazione dell'ambiente di Supervisione

Riportiamo nella figura seguente una vista relativa alla pagina iniziale del software InTouch utilizzato per la creazione dell'ambiente di supervisione, quello cioè indispensabile per poter correttamente effettuare il comando e monitoraggio delle variabili di interesse.

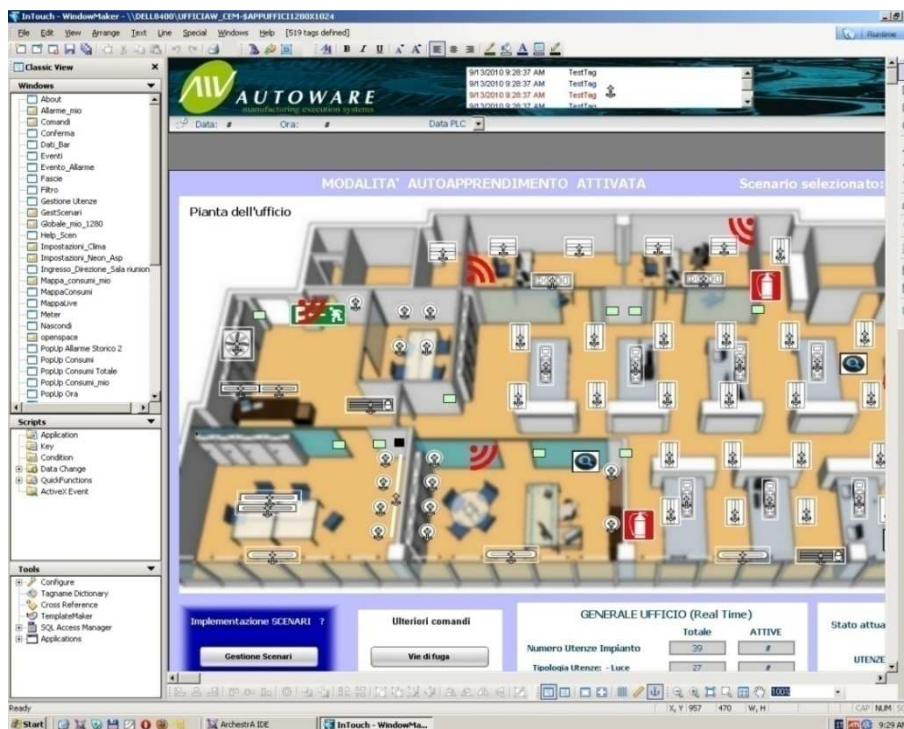


Figura 33. WindowMaker

Grazie a WindowMaker è possibile importare immagini e successivamente aggiungere *Animation Link*, *Script*, *Pop Up*, figure e connessioni con DataBase. Successivamente con l'ausilio del WindowViewer sarà possibile visualizzare e monitorare in maniera interattiva il tutto.

Illustriamo di seguito, suddivisa in punti, la procedura compiuta per la creazione del supervisore.

1. Creazione delle finestre che consentono di inserire gli elementi che verranno visualizzati nel Viewer. Importante è stata la creazione della finestra *ToolBar* necessaria per passare da una pagina all'altra.
2. Importazione delle immagini nei classici formati *bmp* oppure *jpg* (le immagini di sfondo sono state create con il software Google SketchUp).
3. Aggiunta di pulsanti ed *Animation Link* con i loro *Script* interni i quali consentono di compiere azioni determinate.
4. Creazione degli *Script* che consentono di effettuare il collegamento con il DataBase.
5. Creazione di un *EngineService* (in ambiente ArchestrA, descritto nel prossimo capitolo) per consentire il collegamento tra Viewer del Supervisore ed il PLC.

Specifichiamo che ogni operazione compiuta nel WindowViewer può comportare elaborazione di dati, settaggio di bit, asserimenti o deasserimenti di pin che si rispecchieranno direttamente sul PLC. Quest'ultimo quindi sarà la macchina che compie le operazioni decise e impostate dall'operatore.

Molte figure presenti nel Supervisore sono state create e successivamente importate tramite System Platform, che verrà descritta nel prossimo capitolo.

5.3 WindowViewer. Funzionalità di ogni singola "Videata"

Come precedentemente accennato, grazie al pacchetto WindowViewer di InTouch, è possibile visualizzare le pagine create tramite WindowMaker.

In totale il supervisore contiene una trentina di pagine e di seguito andremo a descrivere in dettaglio le più importanti.

Specifichiamo che la risoluzione utilizzata è 1280×1024 .

Tutte le videate che riportiamo nei prossimi paragrafi garantiscono delle specifiche funzionalità ma alcune sezioni risulteranno presenti in ognuna di esse. Quest'ultime sono tre e vengono illustrate di seguito.

TopBar



Figura 34. TopBar

Finestra sempre presente in primo piano, posta nella posizione più alta dello schermo che ha la funzione di:

- visualizzare l'ora attuale e tramite il *PopUp_Ora* (apertura alla pressione del pulsante contrassegnato con DataPLC) il tempo di vita del PLC
- fornire il logo dell'azienda ed eventuali contatti (pulsante About)
- fornire una visione in tempo reale di alcuni allarmi in funzione all'interno dell'ufficio. Specificheremo in seguito il concetto di allarme e il legame che è stato scelto di utilizzare con i sensori di presenza.

La tabella posizionata a fianco del logo riporta in maniera sintetica l'attivazione dei sensori di posizione presenti in ufficio. Alla pressione del pulsante sinistro del mouse

verrà aperta la pagina contrassegnata come *Allarme* e sarà così possibile visualizzare il tutto più dettagliatamente.

DatiBar



Figura 35. DatiBar

Importante finestra, sempre in primo piano, che consente di visualizzare in tempo reale il Consumo Generale dei tre diversi Meter dell'ufficio, il Consumo di Energia Totale ed ultimo Evento compiuto.

I valori visualizzati sono visibili anche nella tabella dettagliata presente in sezione eventi, le loro unità di misura sono relativamente Volt, A, kWatt, kWatt / h.

ToolBar

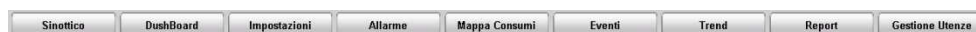


Figura 36. ToolBar

Finestra posta nella parte inferiore dello schermo necessaria a compiere la transizione da una pagina all'altra del Supervisore.

5.3.1 Windows - Globale



Figura 37. Windows – Sinottico

La figura precedente riporta la piantina in scala dell'ufficio Autoware. Più avanti andremo ad illustrare in dettaglio le varie zone, ci concentriamo ora nel descrivere le funzionalità possibili. In background è stata importata la piantina come immagine *jpg* e

superiormente ad essa sono state aggiunte ulteriori figure rappresentanti le varie utenze presenti in ufficio. Quest'ultime risultano essere immagini create con l'editor di ArchestrA ed importate successivamente in InTouch, rimandiamo quindi al capitolo 2. Alla pressione delle lenti in figura 36 è possibile visualizzare i pop up delle zone in dettaglio.

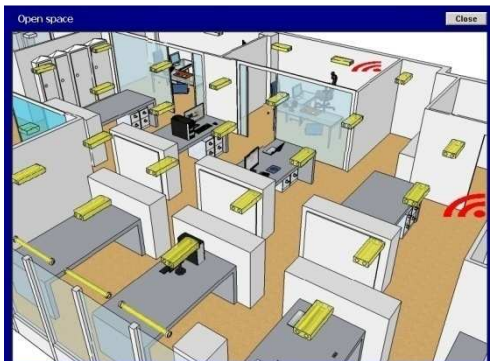


Figura 38. Zona in dettaglio – Open Space

Descriviamo di seguito la funzionalità dei principali pulsanti presenti in figura 36.

Pulsante – Gestione Scenari

Alla pressione del pulsante è associato uno Script che consente l'apparizione del PopUp riportato in Figura ... Questa schermata permette di creare, selezionare, rinominare oppure cancellare lo scenario desiderato. I dati a fianco sono associati allo scenario selezionato.

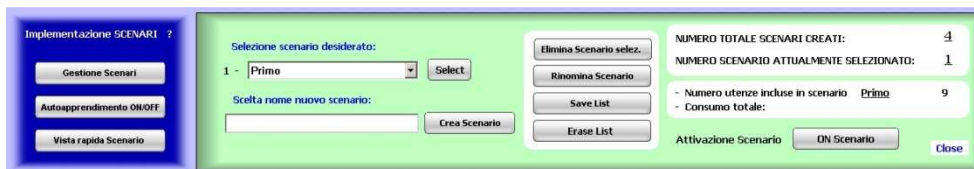


Figura 39. Windows – Gest_Scenari

Una volta compiuta la selezione dello scenario e l'assegnazione delle utenze desiderate sarà possibile rendere attive le modifiche effettuate premendo il tasto **ON Scenario**. Questo, in base alla selezione compiuta, imporrà ad 1 il merker che consentirà l'attivazione dello scenario.

Pulsante – Autoapprendimento ON/OFF

Passo successivo alla creazione dello scenario. Una volta premuto questo pulsante la piantina dell'ufficio verrà sostituita con la sua stessa immagine sfuocata e sarà possibile selezionare le utenze che si desidera includere nel determinato scenario (con il semplice click dei tasti del mouse).

Precisiamo che tutte le utenze presenti in ufficio sono associabili ad uno scenario e le operazioni che possono essere compiute sono:

- inserimento in scenario come ON (pressione del tasto sinistro del mouse)
- inserimento in scenario come OFF (pressione del tasto destro del mouse)
- disinserimento da scenario

Successivamente alla selezione, attorno alla figura relativa all'utenza scelta, sarà visibile un rettangolo lampeggiante. Il colore verde è associato ad inserimento come ON mentre il rosso inserimento come OFF. In caso di disinserimento ovviamente il lampeggio non sarà più visibile.

Alla pressione di Autoapprendimento ON/OFF apparirà inoltre un nuovo pulsante sulla parte superiore destra della finestra. La sua funzione è quella di consentire l'apertura del PopUp denominato *Select_Rapida* che consentirà di selezionare o deselegionare più rapidamente un vasto gruppo di utenze.



Figura 40. Autoapprendimento ON/OFF

Pulsante – Vista rapida Scenario

Grazie a questo pulsante abbiamo la possibilità di visualizzare rapidamente lo stato delle variabili in base allo scenario selezionato, se sono quindi incluse in esso.

Pulsante – Spegni tutto

È un pulsante che pone ad 1 una variabile. A livello di PLC, appena questa variabile diventa 1 tutte le utenze dell'ufficio vengono spente. Questa funzione può essere utilizzata in caso di malfunzionamenti oppure di manutenzione impianto.

Pulsante – Vie di fuga

Crea una sovrapposizione di una immagine rappresentante le vie di fuga dell'ufficio e la posizione degli estintori. Nel nostro caso possono essere informazioni superflue essendo l'ambiente di ridotte dimensioni e popolato da un numero limitato di persone ma nel caso di un grosso impianto risulta sicuramente informazione necessaria e vitale.

Pop Up Comandi

Riportiamo di seguito la finestra che appare in caso di click su una qualsiasi utenza, in caso di autoapprendimento non attivato, presente nel *Sinottico* di impianto.



Figura 41. Pop Up Comandi

Possiamo notare che in essa sono racchiusi tutti i comandi che si possono assegnare alle varie utenze quindi:

- Comando manuale (da SPV) allo stato ON oppure OFF
- Inserimento in Scenario selezionato come ON oppure OFF
- Disinserimento da Scenario selezionato
- Funzione crepuscolare attivata/disattivata per tale utenza

La configurazione di questa finestra è stato frutto del legame tra ArchestrA Editor e WindowMaker di InTouch. La parte grafica e parte degli script sono stati creati tramite l'Editor mentre su InTouch sono state create le variabili che comandano il funzionamento.

Pulsantiera

Nel prossimo capitolo descriveremo in dettaglio il principio che sta alla base del funzionamento dei pulsanti. Intanto ci limitiamo a dire che se la modalità autoapprendimento è attiva al click della figura relativa alla pulsantiera apparirà il secondo Pop Up riportato in figura ...

In questo caso sarà possibile impostare un valore di temporizzazione alla pressione del pulsante, di default è stato impostato 5 secondi. Quindi, come illustrato in figura da esempio, alla pressione del pulsante 2 per un tempo di 5 secondi si avrà la successiva attivazione dello scenario numero 2.



Figura 42. Pulsantiera ad autoapprendimento rispettivamente OFF ed ON

5.3.2 Windows – DushBoard

Questa seconda pagina è stata inserita per permettere di monitorare in tempo reale tutte le variabili ed i Meter stessi implementati tramite CEM.

È stato necessario impostare l'oggetto CEMDataProvider in ambiente IAS per consentire il collegamento con in DataBase. Questa videata rappresenta il cuore del pacchetto Corporate Energy Management, o meglio la visualizzazione finale di tutte le impostazioni compiute durante la strutturazione del progetto.



Figura 43. Windows – DushBoard

5.3.3 Windows – Impostazioni Clima

Le pagine considerate di impostazione sono due. Una relativa al solo impianto di climatizzazione mentre una che raggruppa impianto di aspirazione e neon.

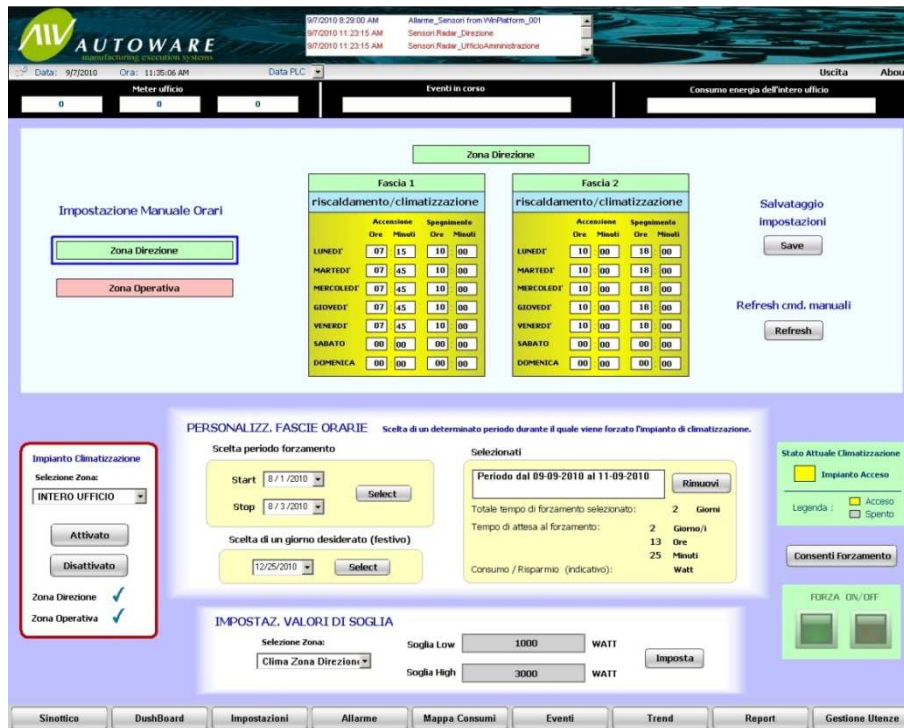


Figura 44. Windows – Impostazioni Clima

In figura ... possiamo notare che per quanto riguarda l'impianto di climatizzazione l'ufficio è suddiviso in due zone:

- zona Direzionale
- zona Operativa

La gestione ed il controllo di queste è indipendente l'una dall'altra.

Impianto Climatizzazione

Partiamo a descrivere la funzionalità del riquadro bianco con bordo rosso in basso a sinistra dello schermo (figura ...). Denominato Impianto Climatizzazione, consente di attivare o disattivare l'utenza clima relativa ad una specifica zona dell'ufficio. Una volta compiuta la selezione della zona desiderata, alla pressione del pulsante **Attivato** il supervisore setterà ad 1 il bit corretto posto in *DB HVAC – Asp*. Grazie alla struttura dell'algoritmo caricato nel PLC tutte le modifiche effettuate all'interno di questa pagina, es. modifica orari, forzamenti, avranno effetto solamente se il bit sopra citato è posto ad 1.

Il concetto di base sfruttato in questo caso è quello di considerare il nostro impianto di climatizzazione (che può comunque essere benissimo allargato a tutti i dispositivi presenti) come quello presente in una grande azienda. Se una sezione di questa azienda venisse per qualche ragione chiusa o resa inutilizzabile sarebbe molto importante poter disattivare utenze che nn risulterebbero necessarie in un simile scenario.

Impostazione Manuale Orari

In questo riquadro, posto nella parte superiore dello schermo, è possibile inserire orari di accensione e spegnimento preferiti in base alle esigenze dell'azienda. Inoltre ogni giorno della settimana può presentare diversi orari e le due fasce orarie possono essere impostate a piacere.

Alla pressione del pulsante **Save** le modifiche compiute saranno rese permanenti e verranno create le assegnazioni al *DB Orari* del PLC.

Riportiamo di seguito una parte dello script incorporato nel suddetto pulsante per capire come vengono passati i valori al PLC tramite il SPV.

```

DIM GS AS INTEGER;
GS = 86400 * $DateTime;

{ OPERATIVA }

IF StringFromTime(GS,4) == "Mon" THEN
  GiornoSettimana = "LUNEDI";
  Galaxy:S7C_S7_001.PLC.HTOP_Accensione1 = (HT_Lu_Ac1_HH * 3600000) + (HT_Lu_Ac1_MM * 60000);
  Galaxy:S7C_S7_001.PLC.HTOP_Accensione2 = (HT_Lu_Ac2_HH * 3600000) + (HT_Lu_Ac2_MM * 60000);

  Galaxy:S7C_S7_001.PLC.HTOP_Spegnimento1 = (HT_Lu_Sp1_HH * 3600000) + (HT_Lu_Sp1_MM * 60000);
  Galaxy:S7C_S7_001.PLC.HTOP_Spegnimento2 = (HT_Lu_Sp2_HH * 3600000) + (HT_Lu_Sp2_MM * 60000);

ENDIF;

IF StringFromTime(GS,4) == "Tue" THEN
  GiornoSettimana = "MARTEDI";
  Galaxy:S7C_S7_001.PLC.HTOP_Accensione1 = (HT_Ma_Ac1_HH * 3600000) + (HT_Ma_Ac1_MM * 60000);
  Galaxy:S7C_S7_001.PLC.HTOP_Accensione2 = (HT_Ma_Ac2_HH * 3600000) + (HT_Ma_Ac2_MM * 60000);

  Galaxy:S7C_S7_001.PLC.HTOP_Spegnimento1 = (HT_Ma_Sp1_HH * 3600000) + (HT_Ma_Sp1_MM * 60000);
  Galaxy:S7C_S7_001.PLC.HTOP_Spegnimento2 = (HT_Ma_Sp2_HH * 3600000) + (HT_Ma_Sp2_MM * 60000);

ENDIF;

...

```

Lo Script riportato illustra le condizioni impostate in caso il giorno della settimana risulti Lunedì o Martedì per la Zona Operativa. Gli altri giorni e soprattutto la Zona Direzionale sanno gestiti allo stesso modo con un semplice cambiamento di variabili.

Personalizzazione Fascie Orarie

Grazie a questo pannello è stata introdotta la possibilità di compiere forzamenti dell'impianto ad uno stato di ON oppure OFF. Sono personalizzabili le date di inizio e di fine forzamento ed inoltre è consentito aggiungere singole giornate, come Natale, Pasqua, ecc.

Il tipo di forzamento desiderato, sempre acceso o sempre spento, viene scelto nel pannellino posto nella zona inferiore destra dello schermo mentre in quello esattamente superiore è possibile monitorare lo stato delle utenze senza dover passare alla pagina *Sinottico*.

In questo caso una complicazione non banale è stata causata dal fatto di dover "passare" correttamente le variabili di ora e data al PLC.

Nel capitolo precedente, al paragrafo Configurazione PLC, è stato descritto il tipo di variabile TDO,,,

Impostazione Valori di Soglia

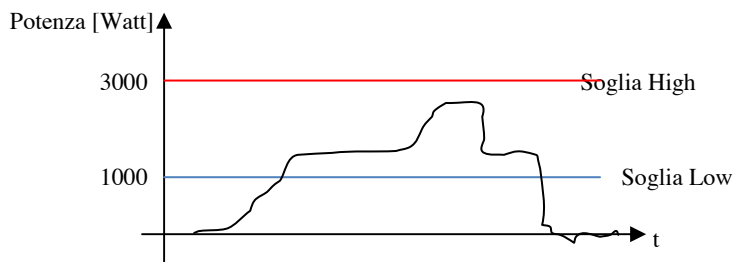


Figura 45. Soglie

Se il Meter relativo all’impianto di climatizzazione fornisce valore nullo in consumo di Energia e viene attivata almeno una delle utenze non si verifica alcun problema. Supponiamo quindi che dopo tale accensione il valore rilevato dal Meter si attesti sui 2000 Watt. È dunque consentita l’accensione di una ulteriore utenza, essendo il valore di Energia inferiore al massimo di soglia consentito. Abbiamo così creato una “fascia”, se il valore misurato rimane al suo interno è possibile attivare una ulteriore utenza.

Questi valori di soglia vengono letti dal PLC tramite delle PEW che andranno a puntare esattamente alle connessioni fisiche create con i Meter. Quest’ultimi restituiscono una serie di 16 valori, sarà poi compito della funzione FC 110 decodificare tali dati e scrivere nel DB HVAC – Asp, alla posizione corretta, il valore misurato di Energia.

5.3.4 Windows – Impostazioni Neon ed Aspirazione

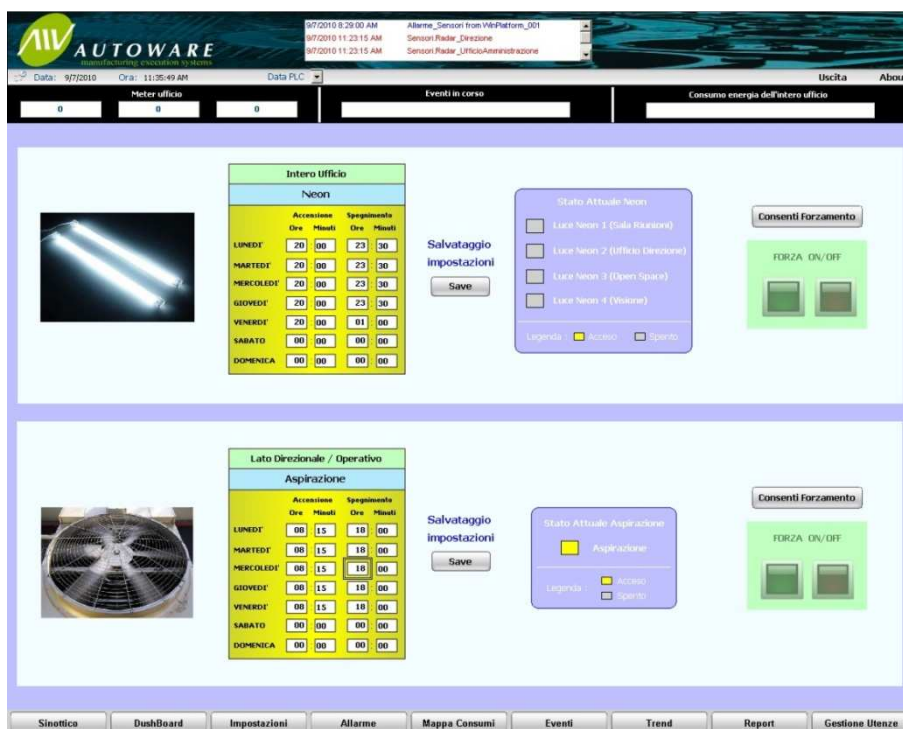


Figura 46. Windows – Impostazioni Neon - Aspirazione

Rispetto al paragrafo precedente, ora per quanto riguarda Neon ed Aspirazione le impostazioni che possono essere compiute sono più ridotte. Rimangono l'assegnazione manuale degli orari e la possibilità di compiere un forzamento ad ON oppure OFF. Il forzamento in questo caso, una volta attivato, ha effetto da subito.

5.3.5 Windows – Allarme

Particolare attenzione va posta a riguardo. Il concetto di allarme considerato in questo progetto è relativo al sistema di antifurto installato in ufficio. A livello di impianto industriale per allarme si intende invece un avviso dovuto a malfunzionamenti o arresti improvvisi non programmati.

In ufficio quindi non saranno mai presenti degli allarmi, tranne nel caso particolare di visualizzazione di Sensori attivati in caso di asserimento di antifurto.

Allari a livello di set bit per accensione di luci o utenza clima non esistono. Piuttosto un concetto di allarme è quello di superamento soglia e disattivazione del clima per evitare picchi di corrente che possono causare danni all'impianto.

The screenshot displays the AUTOWARE software interface for alarm monitoring. The interface is divided into several sections:

- Top Status Bar:** Shows the date (9/7/2010) and time (11:36:31 AM). It also displays the PLC data and a status bar with '0'.
- Left Sidebar (Monitoraggio Sensori Allarme):** Lists 8 sensors with corresponding LED indicators:
 - A - Radar Ingresso
 - B - Radar Direzione
 - C - Radar Ufficio Amministrazione
 - D - Radar Ufficio Commerciale
 - E - Radar Open Space
 - F - Radar Magazzino Grande
 - Porta Principale
 - Porta Magazzino
 - Allarme Generale
 - Allarme On
- Central Area:**
 - Storico Allarme:** A table showing a list of alarm events. The table has columns for Time, Name, AlarmComment, and AlarmDuration. The data shows several 'ATTIVO SPENTO ALLARME' events from 8/31/2010 to 9/2/2010.
 - Real Time Sensors:** A table showing current sensor status. The table has columns for Time, Name, and AlarmComment. The data shows 'Allarme_Sensori from WinPlatform_001', 'Sensori Radar_Direzione', 'Sensori Radar_Ufficio Amministrazione', and 'Sensori Radar_Ufficio Commerciale'.
- Bottom Section:**
 - Sensori attivati ad Allarme ON:** A section with a 'Clear' button and a list of active sensors.
 - Navigation Bar:** Includes buttons for 'Sinotica', 'Dashboard', 'Impostazioni', 'Allarme', 'Mappa Consumi', 'Eventi', 'Trend', 'Report', and 'Gestione Utente'.

Figura 47. Windows – Allarme

Nel pannello posto a sinistra, in figura ... , sono stati riportati gli 8 sensori presenti in ufficio e i Led che forniscono informazione riguardo ad attivazione allarme o ad un suo asserimento. I p 6 vengono attivati in caso di rilevazione di movimento umano mentre i 2 riferiti a Porte solamente in caso di apertura.

I due grandi pannelli a destra invece indicano uno lo storico di attivazione dell'allarme, cioè data ed ora di attivazione per un intervallo di tempo di 3 settimane, mentre il secondo i sensori attivati in tempo reale.

L'ultimo pannello posto in basso a sinistra è attivo solamente in caso di allarme acceso. Esso infatti fornisce il nome del sensore, o dei sensori, che ha causato l'asserimento dell'allarme stesso.

5.3.6 Windows – Mappa Consumi

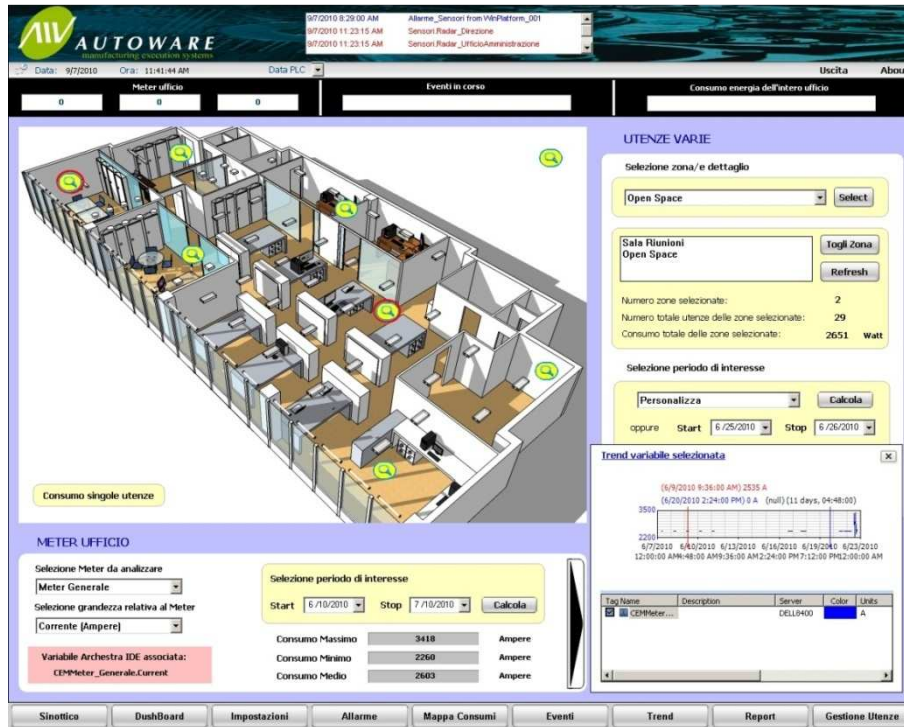


Figura 48. Windows – Mappa Consumi

Questa è la prima videata in cui è stato creato il collegamento con il Database. Alla pressione del pulsante **Calcola** una volta effettuate le scelte del Meter, della grandezza da analizzare e dell'intervallo di tempo di interesse verrà avviato uno Script che consentirà di rilevare dati dal DataBase.

Successivamente sarà anche possibile visualizzare graficamente l'andamento delle variabili considerate aprendo il Pop Up della parte inferiore sinistra nello schermo denominato *Trend variabile selezionata*.

Nella parte destra invece è possibile vedere in tempo reale il consumo di determinate zone dell'ufficio e capire il numero di utenze attivate.

Cliccando invece sopra ai simboli "gialli" sarà possibile visualizzare i dati relativi ad una specifica zona. Funzione molto utile in caso di grandi e complessi impianti.

5.3.7 Windows – Eventi

In questa pagina gli aspetti che ci interessano maggiormente sono la visualizzazione dello storico Eventi occorsi e il consumo in Tensione, Corrente e Potenza delle utenze presenti in ufficio.

Precisiamo ora il concetto di Evento. Per valutare al meglio il consumo di determinate utenze sono stati creati dei “punti di partenza” e di “arrivo” in base ai quali vengono memorizzati i dati di interesse. Nell’arco di tempo considerato verranno compiute delle statistiche in funzione di determinati avvenimenti. Questa è una delle funzionalità offerte dal pacchetto CEM installato in ambiente IAS.

Il software, come illustrato in precedenza, compie un salvataggio dei dati richiesti con un intervallo di tempo di 15 minuti. Avviando una Store Procedure, in ambiente SQL Server, sarà possibile filtrare i dati memorizzati in base al tipo di Meter e alle grandezze che interessano nelle statistiche.

5.3.8 Windows – Trend

Terminiamo il capitolo riferito alla descrizione del supervisore illustrando due videate di pura visualizzazione. In questo caso infatti non sarà possibile compiere nessuna modifica a livello di impianto ma solamente monitorare l’andamento di determinate variabili.

Il Trend è uno strumento importante in fase di valutazione dell’operato di un impianto. Nel nostro caso infatti grazie a questa videata possiamo controllare tutte le variabili che ci interessano.

5.3.9 Windows – Riepilogo Utenze

Evitiamo di riportare la figura relativa alla videata dato che risulta essere semplicemente una tabella contenente tutte le utenze controllabili dell’ufficio in questione.

Risulta essere molto utile e flessibile perchè in caso di installazine del sistema di supervisine in un diverso impianto in cui i dati relativi a variabili presenti fossero memorizzati in uno specifico DataBase, besterebbe solamente creare il collegamento a quel determinato DB e visualizzare il tutto.

CAPITOLO 6

Sistema di Supervisione, funzionalità in dettaglio

6.1 Gestione IN ed Out del PLC

Come accennato nei precedenti capitoli le variabili di ingresso del PLC sono di tipo BOOL. Esse non sono necessariamente degli ingressi fisici, come vale per i sensori di presenza e l'allarme. Infatti per quanto riguarda i pulsanti fisici presenti in ufficio il collegamento con il PLC è ottenuto tramite impostazione di variabili temporanee che vengono resettate ogni volta a fine ciclo del programma. Si tratta di Merker, detti anche variabili di appoggio. Sono state usate queste variabili perchè l'assegnazione dei pulsanti è voluta come fronte di salita di un segnale, dunque quest'ultimo va monitorato e viene successivamente compiuto il necessario asserimento.

I pulsanti in realtà sono assegnati a degli ingressi precisamente da *E 0.0* ad *E 5.6*. i Merker settati invece saranno :

- *M 120.3* fino a *M 120.7* Pulsanti Porta Sala Riunioni (5)
- *M 121.1* e *M 121.2* Pulsanti Interni Sala Riunioni (2)
- *M 121.3* e *M 121.4* Salita e discesa schermo (2)
- *M 121.5* fino a *M 122.1* Open Space Lato Saletta (5)
- *M 122.3* e *M 122.4* Open Space Lato Direzione (2)
- *M 122.6* e *M 122.7* Interno Direzione (2)
- *M 123.1* fino a *M 122.3* Open Space Retro Direzione (3)
- *M 123.4* fino a *M 124.1* Open Space Lato Magazzino (6)
- *M 124.2* fino a *M 124.7* Open Space Lato Zona Break (6)
- *M 125.0* e *M 125.1* Ufficio Amministrazione (2)
- *M 125.4* e *M 125.5* Ufficio Commerciale (2)
- *M 125.6* Magazzino Piccolo (1)
- *M 126.1* fino a *M 126.3* Interno Magazzino Grande (3)

Ai sensori di presenza ed all'allarme sono stati assegnati invece gli ingressi da *E 6.5* ad *E 7.6*. Infine il segnale del sensore crepuscolare è associato all'ingresso *E 7.7*.

6.2 Descrizione del concetto di Scenario e relativa gestione

In un ufficio come in un qualsiasi ambiente in cui siano presenti numerose utenze è importante poter controllare le stesse in maniera autonoma. Un locale di vaste dimensioni può essere occupato da molteplici persone con esigenze a volte molto diverse.

Nel progetto in questione è stato implementato un concetto di gestione delle utenze tramite inserimento delle stesse in una lista preferenziale che verrà letta e utilizzata in caso di necessità.

Propriamente con la parola "Scenario" intendiamo una serie di utenze, anche di diversa natura, che devono essere gestite a piacimento dall'operatore.

Il tutto è stato sviluppato sfruttando la possibilità di legare un ambiente di Supervisione come InTouch ad un PLC che compia le operazioni.

Inizialmente specifichiamo che il progetto creato è predisposto per gestire un numero massimo di scenari pari ad 80.

A livello di PLC è stata creata un'unica FB che ha il compito di “popolare” il Data Block di Istanza, DBI. È stata compiuta questa scelta grazie al fatto che il DB di istanza relativo ad una FB viene creato in base alla struttura che la stessa FB richiede, si può così evitare sprechi di memoria. Basti pensare al risultato che sarebbe stato ottenuto se fosse stato popolato un DB di tipo Globale. La sua dimensione necessariamente doveva essere sempre la massima possibile (data dal numero massimo delle utenze gestibili) dato che a priori non si è a conoscenza del numero di utenze che si desidera controllare e quindi inserire nel DB.

Nel caso di attivazione dello stato di autoapprendimento nel Supervisore il primo passo da compiere è scegliere lo Scenario da “popolare”. Verrà scelto un nome ed associato ad esso un numero posizionato in una Word di memoria del PLC, precisamente in **DB 2.DBW 4**.

Questa Word avrà un'importanza cruciale in fase di riconoscimento Scenario.

Passo successivo alla scelta sopra citata è il passaggio dei comandi. Come precedentemente illustrato anch'essi sono collocati nel DB comandi, in questo caso però in **DB 2.DBW 2**. Quindi nel sinottico d'ufficio al click del mouse sopra ad una utenza essa verrà inserita nello scenario selezionato.

Le funzioni sviluppate in Step 7 e successivamente caricate nel PLC che riguardano la gestione degli scenari sono molteplici. Di seguito riportiamo i segmenti di codice più significativi e illustriamo la procedura seguita.

```
TITLE = Inserimento Utenze in Scenari

iniz: O #Comandi.IN_Ute_scen_ON
      O #Comandi.IN_Ute_scen_OFF
      SPBN fin

/// scenario N ////

L DB2.DBW 4
L 100 /// "puntatore" del DBistanza, in base al tipo scenario
+I
T #NDBScenario

L #NumeroUtenza
L 1
-I
L 16
*I /// set bit presenza (autoapprendimento) utenza in scenario
SLD 3
LAR2

AUF DB 1
U #Comandi.IN_Ute_scen_ON
= DBX [AR2,P#2.6]
U #Comandi.IN_Ute_scen_OFF
= DBX [AR2,P#4.7]

AUF DB [#NDBScenario]
L 10
SLD 3
LAR1
L 100
I1: T #indiceIN
L DBW [AR1,P#0.0]
L 0
==I /// se la word è 0 salta e carica NumeroUtenza
SPB ei1

L DBW [AR1,P#0.0]
L #NumeroUtenza /// se l'utenza è già presente salta cmq
==I /// e incrementa accumulatore
SPB ei1
```

```

+AR1 P#4.0
L #indiceIN /// incrementa AR1 e continua ciclo
LOOP li1

ei1: L #NumeroUtenza
T DBW [AR1,P#0.0]
U #Comandi.IN_Ute_scen_ON
= DBX [AR1,P#2.0] // set ON in DBscenario
U #Comandi.IN_Ute_scen_OFF
= DBX [AR1,P#2.1] // set OFF in DBscenario

fin: NOP 0

```

Il segmento di codice sopra riportato è riferito alla funzione FC 105 e questo in particolare è responsabile all’inserimento come ON o come OFF delle utenze nello scenario selezionato. La stessa funzione contiene la possibilità di disinserimento utenze. Una volta “popolato” il DB di istanza sarà necessario settare le corrette variabili del *DB Utenze* in caso di attivazione dello Scenario.

```

TITLE =Gestione scenario

// Tramite chiamata di FB100 setto il bit CL_scenario relativo ad ogni utenza
// contenuta in DB1 (Stato Utenza)

///// SCENARIO 1 ///// ----- /////

CALL FC 118 /// decodifica pulsanti da attivare in base allo scenario scelto
NumeroScenario := 1

CALL FC 119
PULSANTI := #App_Pulsanti

U(
O E [#App_Pulsanti.NumPulsante1]
O E [#App_Pulsanti.NumPulsante2]
O E [#App_Pulsanti.NumPulsante3]
O E [#App_Pulsanti.NumPulsante4]
)
= #Pulsante ///

CALL FC 109 // Alla pressione prolungata del pulsante (4 sec)
Pulsanti := #Pulsante // attiva Scen_OK
Numero_Scenario := 1
FineCont := #Scen_OK

U M 802.0 // comando da SPV
O #Scen_OK // pressione prolungata del pulsante
= #AttivaScenario

CALL FB 100 ,DB 101 // set di CL_On oppure CL_Off
AttivaScenario := #AttivaScenario

R M 802.0
L 0
T #App_Pulsanti.NumPulsante1
T #App_Pulsanti.NumPulsante2
T #App_Pulsanti.NumPulsante3
T #App_Pulsanti.NumPulsante4

```

Questo secondo segmento invece è riferito ad una sezione della funzione *FC 107, FC Gestione Scenari*.

6.3 Gestione dei pulsanti e loro assegnazione a determinati Scenari

Nei capitoli precedenti abbiamo descritto il concetto di scenario e la sua attivazione. Essa può avvenire tramite comando fornito da Supervisore (set di uno specifico Merker) oppure alla pressione prolungata (periodo di tempo impostabile) di un determinato pulsante.

Il metodo di assegnazione di tale periodo di pressione e del relativo pulsante può essere visualizzata in Figura 40 al capitolo 5.

Per rendere questa scelta di attivazione il più flessibile possibile è stata creata una routine che compie la decodifica dello Scenario selezionato e del pulsante desiderato. Vengono sfruttati i blocchi funzione di STEP 7 e di seguito vedremo i segmenti di codice più rappresentativi.

```
TITLE =Scrivi numero Scenario

U #Utenza.Stato.ON_OFF_scenario;
SPBN out;
L DB2.DBW 4;
T #Utenza.Stato.Numero_Scen_Associato;
L 0;
T DB2.DBW 4;
out: R #Utenza.Stato.ON_OFF_scenario;
```

La routine sopra riportata compie l'assegnazione del numero dello scenario selezionato al pulsante scelto. Precisiamo che i pulsanti in ufficio sono circa 50 ed ognuno di essi è stato associato ad una utenza presente in *DB Utenze*, dalla utenza 201 in poi.

```
TITLE =Pulsante Corretto

...
...

//// pulsanti da 16 a 8 ////

due: L #NumeroPulsante
L 8
<=I
SPB uno
L #NumeroPulsante
L 9
-I
T #appoggio
AUF DB 8
T DBD [AR2,P#0.0]
L 1
SLD 3
OD
T DBD [AR2,P#0.0]
+AR2 P#4.0
SPA fine

//// pulsanti da 8 a 0 ////

uno: L #NumeroPulsante
L 0
<=I
SPB fine
L 0
SLD 3
AUF DB 8
T DBD [AR2,P#0.0]
L #NumeroPulsante
L 1
-I
OD
```

```
T DBD [AR2,P#0.0]
+AR2 P#4.0
```

```
////////// FINE E LOOP CICLO //////////
```

```
fine: NOP 0
out: +AR1 P#16.0
L #indice
LOOP Loo1
```

Questa seconda routine invece è necessaria a settare il pulsante scelto. Come detto gli ingressi del PLC sono molteplici ed ogni pulsante presente in ufficio è fisicamente connesso a quest'ultimo tramite i bit di ingresso E, i quali vanno da **E 0.0** ad **E 5.6**. Quindi ad esempio se l'utenza selezionata è la 203 il pulsante corretto da settare sarà il terzo. Corrisponde dunque all'ingresso **E 0.2**.

6.4 Gestione delle varie utenze

Come già definito sono presenti utenze di tipologia diversa. Possiamo distiguerle in due grandi categorie: tipo **Luce** e tipo **Forze**.

Il primo tipo comprende utenze quali plafoniere e neon, mentre il secondo le rimanenti cioè Prese, Clima, Aspirazione e telo motorizzato.

Le utenze di tipo Luce sono gestite tramite la funzione FC 101 mentre le tipo Forze tramite la FC 103. Ognuna di queste tipologie di utenze sono gestite in maniera diversa e riporteremo in seguito parte del codice significativa per definirle in dettaglio.

6.4.1 FC Luce

Gestione Luci

Per quanto riguarda la gestione delle sole utenze luci, cioè le plafoniere la routine utilizzata è quella che si può notare nel prossimo segmento di codice.

```
TITLE =Accendi luce

// accensione tramite scenario, pulsanti o radar

U(
O #Utenza.Stato.CL_scenario_ON
O #Utenza.Stato.Cmd_on_man_SPV
O(
U #Utenza.Stato.FrontePulsanteON
X #CMDUte
)
)
O(
U #Crepuscolare
UN #Utenza.Stato.NO_Crepuscolare
U #Radar
)
)
UN #Allarme
UN #Stato_Crepusc
= #CMDUte;
R #Utenza.Stato.CL_scenario_ON;
R #Utenza.Stato.Cmd_on_man_SPV;

TITLE =Spegni luce

// spegnimento da scenario, disatt radar, disatt allarme
```

```

U(
O #Utenza.Stato.CL_scenario_OFF
O #Utenza.Stato.Cmd_off_man_SPV
O #Fine_conteggio
O #SpegniTutto
)
UN #Allarme

R #CMDUte
R #Utenza.Stato.CL_scenario_OFF
R #Utenza.Stato.Cmd_off_man_SPV
R #SpegniTutto

```

Gestione Neon

TITLE =GESTIONE NEON

```

L #NumeroUtenza
L 23
<I
SPB none
L #NumeroUtenza
L 26
>I
SPB none

```

/// FORZAMENTO ///

```

U DB2.DBX 6.7
= #Utenza.Stato.ForzaON
R #Utenza.Stato.ForzaOFF

```

```

U DB2.DBX 7.0
= #Utenza.Stato.ForzaOFF
R #Utenza.Stato.ForzaON

```

////////////////////////////////////

```

CALL FC 113
Orario := DB7.OrarioNeon

```

////////////////////////////////////

```

U(
O #Utenza.Stato.Cmd_off_man_SPV
O #Utenza.Stato.CL_scenario_OFF
)
U DB7.DBX 8.0
SPB speg

```

```

U(
O #Utenza.Stato.Cmd_on_man_SPV
O #Utenza.Stato.CL_scenario_ON
)
UN DB7.DBX 8.0
SPB acc

```

/// Funzionamento Auto ///

```

U DB7.DBX 8.0
= #CMDUte
R #Utenza.Stato.Cmd_off_man_SPV
R #Utenza.Stato.Cmd_on_man_SPV

```

```

UN DB7.DBX 8.0
R #CMDUte
R #Utenza.Stato.Cmd_off_man_SPV
R #Utenza.Stato.Cmd_on_man_SPV
SPA forz

```

/// Funzionamento Manuale ///

```

speg: SET
R #CMDUte

```



```

R #Utenza.Stato.CL_scenario_OFF
SPA forz

acc: SET
= #CMDUte
R #Utenza.Stato.CL_scenario_ON
SPA forz

forz: U #Utenza.Stato.ForzaON
SPBN off
= #CMDUte

off: U #Utenza.Stato.ForzaOFF
SPBN end
R #CMDUte
SPA end

```

6.4.2 FC Forze

Per quanto riguarda la gestione delle utenze forze invece la funzione principale non è più la FC 101 ma bensì FC 103. All'interno di quest'ultima sono contenute le chiamate alle ulteriori funzioni che andranno a gestire separatamente Clima, Aspirazione, Motore e Prese. Avendo suddiviso le 5 diverse utenze per tipo sono stati creati dei salti a sezione di codice interessato.

Gestione Clima

In questo caso, in FC 103, prima di compiere le azioni di Set avverrà la chiamata alla FC 104 – *FC HVAC – Asp* che in breve si può riassumere come la funzione che monitora le soglie e setta l'entrata o l'uscita dalle fasce orarie grazie alla successiva chiamata ad FC 106 – *FC Orari HVAC – Asp*.

TITLE =Comando Climatizzatore Lato Direzionale oppure Operativo

```

clim: CALL FC 104
NumeroUtenza := #NumeroUtenza
CL_ON_HVAC := #ON_Utenza
CL_OFF_HVAC := #OFF_Utenza
Ok_Soglia1 := #Possibile_ON
Ok_Soglia2 := #Spegnimento
ForzaON := #Utenza.Stato.ForzaON
ForzaOFF := #Utenza.Stato.ForzaOFF
Reset_Cmd_Man := #RESET_CMD

////////////////////////////////////
U #RESET_CMD
R #Utenza.Stato.Cmd_on_man_SPV
R #Utenza.Stato.CL_scenario_ON
R #Utenza.Stato.Cmd_off_man_SPV
R #Utenza.Stato.CL_scenario_OFF
////////////////////////////////////

// RESET UTENZA

U(
O #Spegnimento // perchè soglia high superata
O #Allarme
O #OFF_Utenza // dato dagli orari
O #Utenza.Stato.CL_scenario_OFF
O #Utenza.Stato.Cmd_off_man_SPV
O #SpegniTutto
)
UN(
O #ON_Utenza // dato dagli orari
O #Utenza.Stato.CL_scenario_ON
O #Utenza.Stato.Cmd_on_man_SPV
)

```

```

SPB res

// SET UTENZA

U #Possibile_ON // ancora al di sotto di soglia low
U(
O #ON_Utenza // dato dagli orari
O #Utenza.Stato.CL_scenario_ON
O #Utenza.Stato.Cmd_on_man_SPV
)
UN #Allarme

= #CmdUte

U(
O #Utenza.Stato.CL_scenario_OFF
O #Utenza.Stato.Cmd_off_man_SPV
)
SPB res
SPA for

res: SET
R #CmdUte
R #SpegniTutto

// FORZATURA ON OPPURE OFF

for: U #Utenza.Stato.ForzaOFF
R #CmdUte
/// se forzamento resetta tutti cmd manuali ///
R #Utenza.Stato.CL_scenario_OFF
R #Utenza.Stato.Cmd_off_man_SPV
R #Utenza.Stato.CL_scenario_ON
R #Utenza.Stato.Cmd_on_man_SPV

U #Utenza.Stato.ForzaON
S #CmdUte
/// se forzamento resetta tutti cmd manuali ///
R #Utenza.Stato.CL_scenario_OFF
R #Utenza.Stato.Cmd_off_man_SPV
R #Utenza.Stato.CL_scenario_ON
R #Utenza.Stato.Cmd_on_man_SPV

SPA fine

```

Gestione Aspirazione

Molto simile al caso precedente è la gestione dell'impianto di aspirazione. L'unica differenza sta nel fatto che le fasce orarie non sono più due, cioè mattiniera e pomeridiana ma bensì una sola e non è necessaria la gestione delle soglie di consumo.

Gestione Prese

```

TITLE =Posto Lavoro Scrivanie

pres: U(
O #Utenza.Stato.CL_scenario_ON
O #Utenza.Stato.Cmd_on_man_SPV
)
X #CmdUte
UN #Allarme
= #CmdUte
R #Utenza.Stato.CL_scenario_ON
R #Utenza.Stato.Cmd_on_man_SPV

U(
O #Utenza.Stato.CL_scenario_OFF
O #Utenza.Stato.Cmd_off_man_SPV
O #SpegniTutto

```

```

)
UN #Allarme
R #CmdUte
R #Utenza.Stato.CL_scenario_OFF
R #Utenza.Stato.Cmd_off_man_SPV
R #SpegniTutto

////////////////////////////////////
// Allo spegnimento Allarme accensione di Appliquet Reception
U #Utenza.Stato.FronteAllarmeOFF
= #All_OFF
CALL FB 100 , DB 179
    AttivaScenario := #All_OFF
////////////////////////////////////

SPA fine;

```

6.5 Allarmi

In ufficio Autoware è presente un sistema di antifurto fornito di 6 sensori di movimento e 2 sensori a pressione. Gli ingressi all'ufficio sono 2 e i sensori a pressione sono posti sulle due porte di ingresso. Di seguito riportiamo l'assegnazione dei sensori agli ingressi del PLC, videata del supervisore illustrata in Figura 45.

	Nome Sensore	Ingresso PLC
1	Radar Ingresso	<i>E 6.5</i>
2	Radar Direzione	<i>E 6.8</i>
3	Radar Ufficio Amministrazione	<i>E 6.7</i>
4	Radar Ufficio Commerciale	<i>E 6.6</i>
5	Radar Open Space	<i>E 6.9</i>
6	Radar Magazzino Grande	<i>E 7.0</i>

7	Porta Ingresso Principale	<i>E 6.5</i>
8	Porta Magazzino	<i>E 6.8</i>

Una caratteristica importante visibile sempre in Figura 45 è il fatto che in caso di Allarme attivata vengono visualizzati i nomi dei sensori che hanno causato l'asserimento dell'antifurto.

6.6 LDC – Lista Dati di Comunicazione

Viene utilizzato un foglio Excel per riportare tutti i dati più rilevanti dell'intero programma PLC. Dai bit settati in caso di comandi forniti da supervisore a posizioni e strutture dei Data Block presenti in memoria.

Tabelle di scambio dati PLC/SPV

Lista abbreviazioni:

- Tag = Nome utenza
- A = Uscita di comando dal PLC (Uscita)
- E = Risposta al comando, fine corsa o termico e teleruttore inseriti (Ingresso)
- CL = Bit di comando utenza dal programma automatico (Consenso Logico)
- SPV = Registro di stato e di comando dal SuPerVisore

LEGGENDA COMANDO UTENZE

VALORE	COMANDO	VALORE
Decimale		Binario
0	Nessun comando	00000000_00000000
1	Automatico	00000000_00000001
2	ON Manuale	00000000_00000010
4	OFF Manuale	00000000_00000100
8	Inserimento UtENZA in scenario ON	00000000_00010000
16	Inserimento UtENZA in scenario OFF	00000000_00100000
32	Tacita allarmi	00000000_01000000
64	Simulazione ON	00000000_01000000
128	Simulazione OFF	00000000_10000000
256	Manutenzione ON	00000001_00000000
512	Manutenzione OFF	00000010_00000000
1024	Set a "I" ing.digitale simulato	00001000_00000000
2048	Set a "M" ing.digitale simulato	00010000_00000000
4096	Abilita timeout simulazione	00010000_00000000
8192	Disabilita timeout simulazione	00100000_00000000
16384	Disinserimento UtENZA da scenario	01000000_00000000
32768	Disinserimento tutte UtENZE da scenario	10000000_00000000

LEGGENDA TIPO SCENARIO

VALORE	TIPO
Decimale	
1	Scenario 1
2	Scenario 2
3	Scenario 3
..	...

Figura 49. LDC

CAPITOLO 7

Conclusioni

7.1 Principali problematiche riscontrate

Come illustrato nei capitoli precedenti, il progetto in questione può essere ritenuto completo. Con questo termine intendiamo il fatto che per portare a termine gli obiettivi dell'esperienza sono state necessarie basi di competenza per quanto riguarda l'automazione allo stato dell'arte.

Gli argomenti trattati in questa esperienza possono essere riassunti in:

- conoscenza e stesura di un algoritmo di controllo in ambiente STEP 7 da caricare in un PLC Siemens
- creazione di un software di supervisione
- rilevazione dati dal campo tramite Meter e Sensori e successivo condizionamento del segnale
- comprensione e implementazione del concetto di prelievo dati da DataBase e loro rielaborazione
- programmazione in ambiente System Platform e CEM

Particolari difficoltà stanno nel fatto di saper coniugare nel migliore dei modi diversi ambienti di programmazione. Nel caso di Step 7 infatti la programmazione può essere definita a basso livello, avendo a che fare con il linguaggio AWL simile ad Assembly cioè a livello macchina. Uno svantaggio in questo caso è dato dal fatto che la comprensione e la logica di programmazione stessa può essere faticosa. Positivo invece è la presenza di precisi ed adeguati metodi di debug, possono essere monitorate tutte le variabili in uso ed il ciclo programma.

Molto diverso invece è il caso di InTouch il quale risulta essere un ambiente, che consente di creare e monitorare un supervisore, definito ad alto livello. In questo caso quindi le operazioni consentite al programmatore sono più intuitive e di più semplice compimento. Non si tratta più di svolgere una programmazione a “contatti” come può essere definita quella di Step 7 ma bensì molto più grafica e intuitiva. L'ambiente in questione è prettamente di tipo grafico, all'interno del quale possono essere aggiunte semplici routine di chiamate a funzioni preconfigurate.

In questo caso però il grosso svantaggio è il fatto che non è presente nessun sistema per seguire il ciclo di svolgimento delle operazioni. Non essendoci un metodo di debug può risultare molto complesso a volte identificare possibili errori.

Rilevazione dati dai Meter

Assolutamente non banale è stato configurare correttamente la stazione Simatic tenendo conto della presenza dei 3 Meter presenti in ufficio. Come precedentemente illustrato i Meter forniscono in uscita una serie di 16 byte di dati con frequenza prefissata. Questa lettura verrà compiuta tramite una scheda di acquisizione dati in protocollo PROFIBUS.

Nella figura seguente invece si può notare la configurazione della linea di rete, nel nostro caso Profibus (standard di Siemens), alla quale sono stati collegati i Meter stessi.

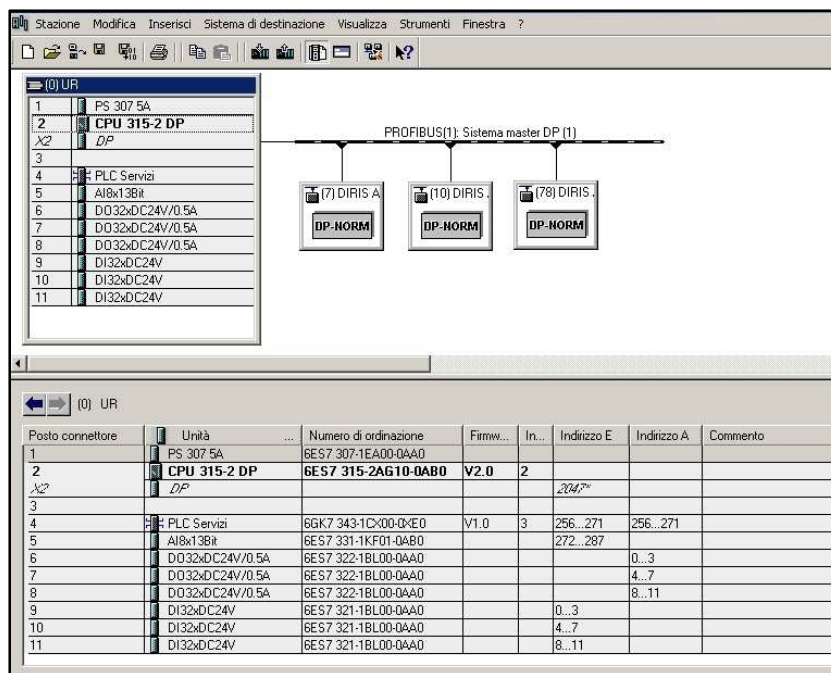


Figura 50. Configurazione Stazione con aggiunta dei Meter

7.2 Possibile implementazione per soluzioni industriali

Il progetto sviluppato viene utilizzato per controllare un ufficio di medie dimensioni. Tutte le apparecchiature presenti sono gestite da un solo PLC che svolge senza problemi i compiti assegnati.

Il sistema di supervisione creato può essere definito flessibile quindi il suo utilizzo ampliato non solamente a livello di gestione dell'ufficio considerato ma bensì a qualsiasi altro ambiente in cui ci siano esigenze di controllo utenze. Sarà necessario ovviamente compiere le dovute modifiche al supervisore e agli oggetti presenti in IAS.

Impianto di Climatizzazione ed Aspirazione

La gestione di queste utenze, che nel caso di questo ufficio sono le 34, 35, 36 e 37, dipenda dalla struttura del DB 5. Esso è stato costruito pensando alla possibilità di un semplice ampliamento fornendo maggiori utenze di questo tipo. Ognuna di esse può essere gestita in maniera indipendente dalle altre.

Utenze Luce

Tutte le utenze presenti in ufficio sono state inseriti in *DB Utenze* preimpostato con un valore massimo di 200 utenze.

Anche il concetto di scenario è semplicemente riconfigurabile dato che al comando fornito da Supervisore vengono settati dei bit presenti nella DB sopra citata.

Rilevazione dati da sensori analogici

Tramite funzione FB 300 ed FB 220 è possibile filtrare il segnale proveniente da sensori analogici. Non è il caso dell'ufficio in questione ma in un impianto di qualsiasi genere possono essere presenti anche molteplici sensori di temperatura, pressione, luminosità, ecc.

In questo caso quindi, come per la lettura dei dati dai Meter, si tratterà di utilizzare delle PEW e leggere i dati provenienti dai sensori. Queste PEW sono delle periferiche esterne, cioè ingressi provenienti da una scheda del tipo citato precedentemente.

7.3 Conclusioni ed impressioni

Il progetto in questione è stato portato a termine in maniera soddisfacente ed il sistema di supervisione risulta adeguato all'impianto che si desidera controllare.

In questo ufficio, Autoware, era già presente un PLC Siemens ed un Supervisore software assegnato ad esso. Il programma Step 7 caricato in esso era molto più scarno e privo di funzionalità di quello descritto in questa relazione. Non erano presenti le funzioni che consentono di creare e gestire scenari, pulsanti, crepuscolare e forzamenti vari.

Anche lo stesso supervisore, montato su un server denominato ENEA, era molto meno funzionale e consentiva minore flessibilità. Come detto infatti in questo caso, compiendo un numero ridotto di modifiche, è possibile utilizzare il supervisore creato anche per impianti di dimensioni ben maggiori.

Altra differenza importante in questa esperienza è la presenza di una System Platform che ha consentito di creare una diversa e più complessa connessione tra PLC e supervisore senza dover appesantire o rendere più complicata la programmazione del tutto. È stato quindi possibile inserire il pacchetto CEM, impossibile senza presenza di IAS, e rendere il sistema di supervisione più funzionale ed utile.

BIBLIOGRAFIA

- Siemens, Simatic Software, *AWL per S7 – 300/400 Programmazione di blocchi*
- Siemens, Simatic Software, *Software di sistema per S7 – 300/400 Funzioni standard e di sistema*
- Wonderware, Wonderware Italia, www.wonderware.it, 10 Ottobre 2010
- Wonderware, *Traninig Manual – InTouch 8.0 , Basic Course*
- Wonderware, *Industrial SQL Server 9.0*
- Wonderware, *Traninig Manual - System Platform – Part 1, Wonderware Application Server 3.1 and Device Integration Products*

Ringraziamenti particolari

Professor Stefano Vitturi

Ing. Luigi De Bernardini

Ing. Riccardo Zorzato, Ing. Massimo Scattolaro

Ai familiari, a tutti gli amici che mi hanno sostenuto