

UNISIM: Sviluppo, validazione e prototipizzazione rapida per sistemi di automazione

G. De Tommasi, P. Di Sanzo, A. Pironti

Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II

Napoli, 23-24 Novembre 2005



Sommario

- 1 Introduzione
- 2 UNISIM: il tool
- 3 Considerazioni conclusive



Sommario

- 1 Introduzione
- 2 UNISIM: il tool
- 3 Considerazioni conclusive



Sommario

- 1 Introduzione
- 2 UNISIM: il tool
- 3 Considerazioni conclusive



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Il contesto

- Strumenti per lo sviluppo e la validazione di software di automazione
- Norma IEC 61131-3
- Strumenti proprietari:
 - recepiscono solo in parte la norma IEC
 - non consentono la migrazione del software tra piattaforme diverse
 - sono dotati di *emulatori* che consentono una prima validazione del software
 - la validazione *hardware-in-the-loop* del software può essere fatta solo con il controllore target



Gli obiettivi

Lo strumento di sviluppo *ideale* dovrebbe consentire di:

- Scrivere software di automazione potendo prescindere dalla piattaforma implementativa
- Emulare una piattaforma target virtuale
- Prototipizzare sistemi di controllo di piccole/medie dimensioni
- Migrare il software di automazione tra piattaforme diverse



Gli obiettivi

Lo strumento di sviluppo *ideale* dovrebbe consentire di:

- Scrivere software di automazione potendo prescindere dalla piattaforma implementativa
- Emulare una piattaforma target virtuale
- Prototipizzare sistemi di controllo di piccole/medie dimensioni
- Migrare il software di automazione tra piattaforme diverse



Gli obiettivi

Lo strumento di sviluppo *ideale* dovrebbe consentire di:

- Scrivere software di automazione potendo prescindere dalla piattaforma implementativa
- Emulare una piattaforma target virtuale
- Prototipizzare sistemi di controllo di piccole/medie dimensioni
- Migrare il software di automazione tra piattaforme diverse



Gli obiettivi

Lo strumento di sviluppo *ideale* dovrebbe consentire di:

- Scrivere software di automazione potendo prescindere dalla piattaforma implementativa
- Emulare una piattaforma target virtuale
- Prototipizzare sistemi di controllo di piccole/medie dimensioni
- Migrare il software di automazione tra piattaforme diverse



La norma IEC 61131-3

Programmable controllers - Part 3: Programming languages

Definisce:

- i linguaggi di programmazione per controllori a logica programmabile
- i tipi di dato e le strutture dati
- la struttura di un *progetto di automazione*

Progetto di automazione

Insieme dei dati, dei programmi e delle informazioni di configurazione di una soluzione d'automazione



La norma IEC 61131-3

Programmable controllers - Part 3: Programming languages

Definisce:

- i linguaggi di programmazione per controllori a logica programmabile
- i tipi di dato e le strutture dati
- la struttura di un *progetto di automazione*

Progetto di automazione

Insieme dei dati, dei programmi e delle informazioni di configurazione di una soluzione d'automazione



La norma IEC 61131-3

Programmable controllers - Part 3: Programming languages

Definisce:

- i linguaggi di programmazione per controllori a logica programmabile
- i tipi di dato e le strutture dati
- la struttura di un *progetto di automazione*

Progetto di automazione

Insieme dei dati, dei programmi e delle informazioni di configurazione di una soluzione d'automazione



La norma IEC 61131-3

Programmable controllers - Part 3: Programming languages

Definisce:

- i linguaggi di programmazione per controllori a logica programmabile
- i tipi di dato e le strutture dati
- la struttura di un *progetto di automazione*

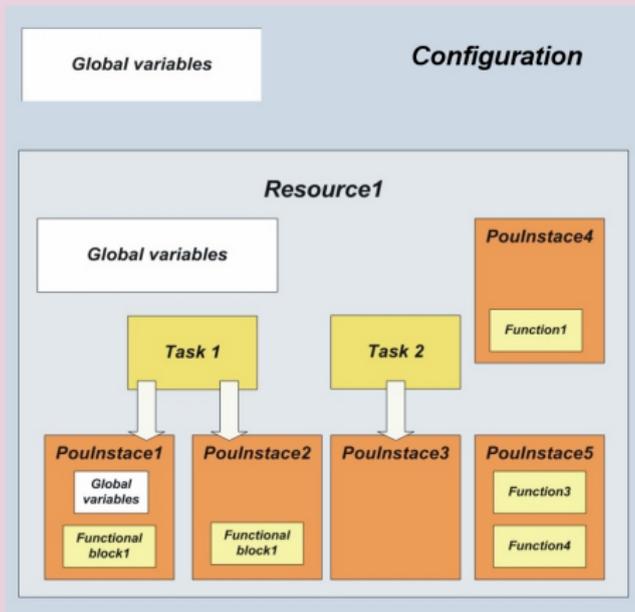
Progetto di automazione

Insieme dei dati, dei programmi e delle informazioni di configurazione di una soluzione d'automazione



La norma IEC 61131-3

Risorse, task, POUs, . . .



- Organizzazione fondata nel 1992
- Riunisce i maggiori produttori nel settore dell'automazione industriale
- Incoraggia l'utilizzo di standard per lo sviluppo di software di automazione

Membri

- ABB
- Beckhoff
- Omron
- Schneider Electric
- Siemens
- Rockwell Automation
- ...

Website

<http://www.plcopen.org>

- Organizzazione fondata nel 1992
- Riunisce i maggiori produttori nel settore dell'automazione industriale
- Incoraggia l'utilizzo di standard per lo sviluppo di software di automazione

Membri

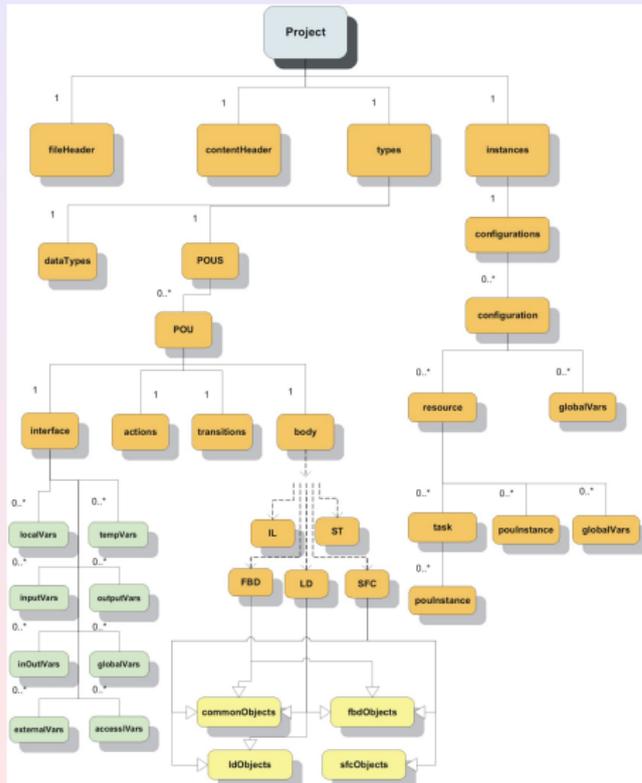
- ABB
- Beckhoff
- Omron
- Schneider Electric
- Siemens
- Rockwell Automation
- ...

website

<http://www.plcopen.org>



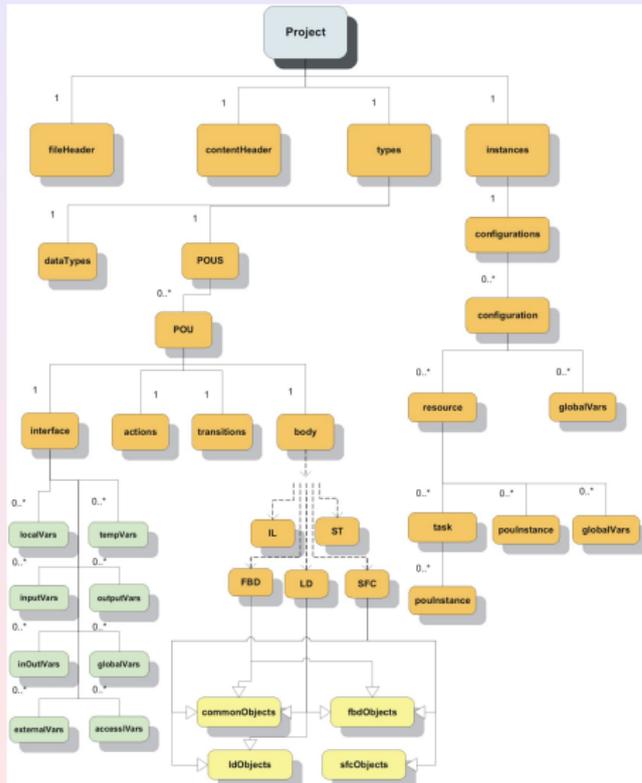
XML Formats for IEC 61131-3



- Prima versione ufficiale rilasciata il 28 aprile 2005
- Definisce i documenti mediante *schemi XML*
- Consente lo scambio di progetti, programmi e librerie



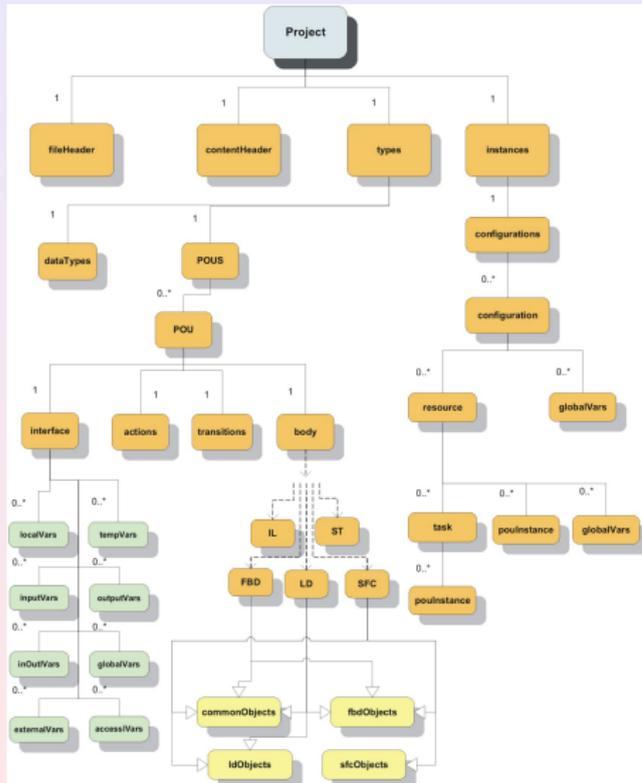
XML Formats for IEC 61131-3



- Prima versione ufficiale rilasciata il 28 aprile 2005
- Definisce i documenti mediante *schemi XML*
- Consente lo scambio di progetti, programmi e librerie



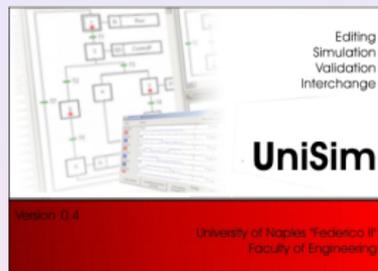
XML Formats for IEC 61131-3



- Prima versione ufficiale rilasciata il 28 aprile 2005
- Definisce i documenti mediante *schemi XML*
- Consente lo scambio di progetti, programmi e librerie



UniSim



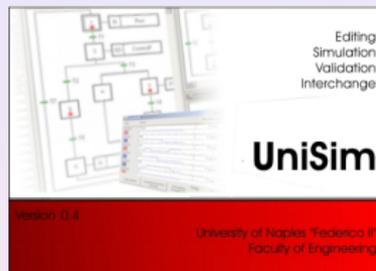
Funzionalità

UniSim consente di completare il ciclo di sviluppo di un progetto:

- editing
- validazione mediante simulazione
- prototipizzazione rapida e test *hardware-in-the-loop*
- controllo di processi soft real-time
- export/import di progetti utilizzando *XML Formats for IEC 61131-3*



UniSim

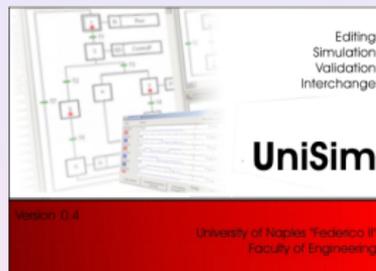


Funzionalità

UniSim consente di completare il ciclo di sviluppo di un progetto:

- editing
- validazione mediante simulazione
- prototipizzazione rapida e test *hardware-in-the-loop*
- controllo di processi soft real-time
- export/import di progetti utilizzando *XML Formats for IEC 61131-3*

UniSim

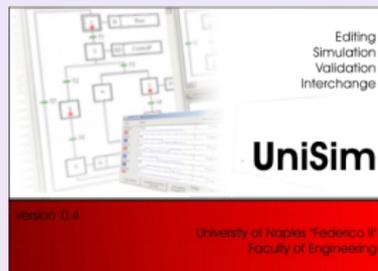


Funzionalità

UniSim consente di completare il ciclo di sviluppo di un progetto:

- editing
- validazione mediante simulazione
- prototipizzazione rapida e test *hardware-in-the-loop*
- controllo di processi soft real-time
- export/import di progetti utilizzando *XML Formats for IEC 61131-3*

UniSim



Funzionalità

UniSim consente di completare il ciclo di sviluppo di un progetto:

- editing
- validazione mediante simulazione
- prototipizzazione rapida e test *hardware-in-the-loop*
- controllo di processi soft real-time
- export/import di progetti utilizzando *XML Formats for IEC 61131-3*

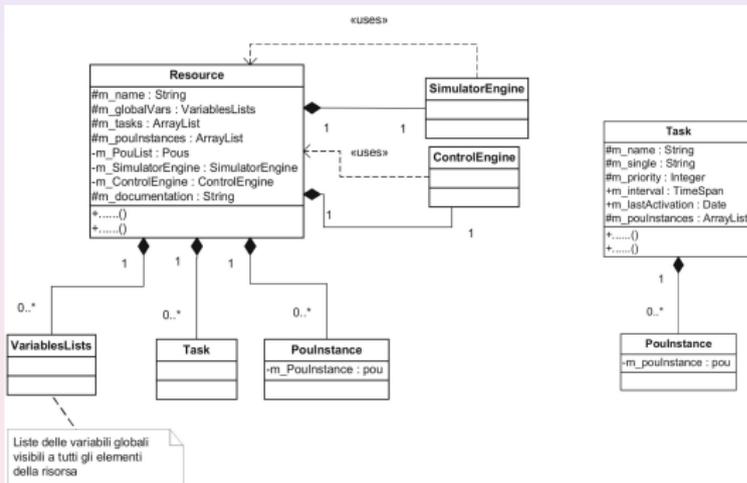


Funzionalità

UniSim consente di completare il ciclo di sviluppo di un progetto:

- editing
- validazione mediante simulazione
- prototipizzazione rapida e test *hardware-in-the-loop*
- controllo di processi soft real-time
- export/import di progetti utilizzando *XML Formats for IEC 61131-3*

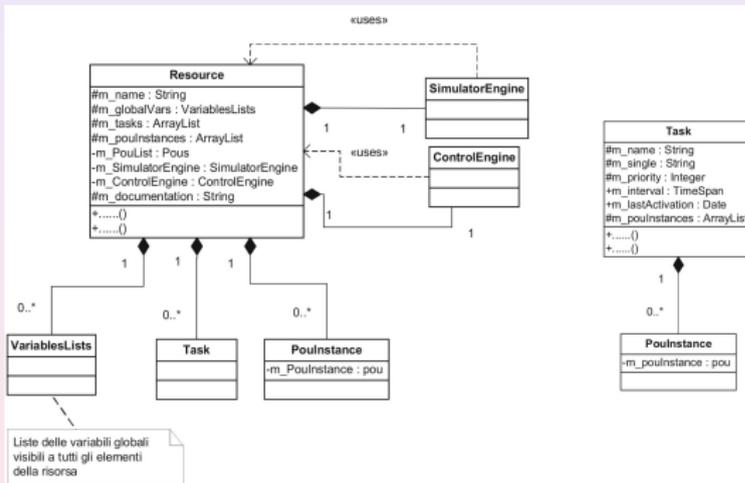
UniSim - Implementazione



- Progettato seguendo l'approccio *object-oriented*
- L'architettura del software rispecchia il modello software definito dalla norma
- Sviluppato con la piattaforma *.Net Framework*



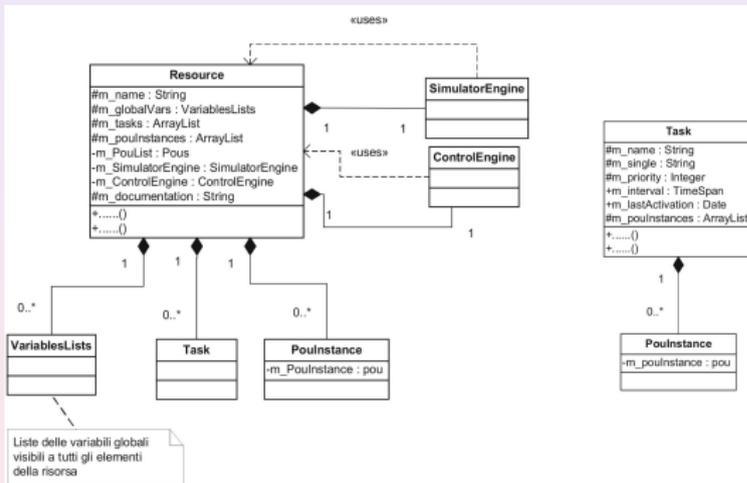
UniSim - Implementazione



- Progettato seguendo l'approccio *object-oriented*
- L'architettura del software rispecchia il modello software definito dalla norma
- Sviluppato con la piattaforma *.Net Framework*



UniSim - Implementazione

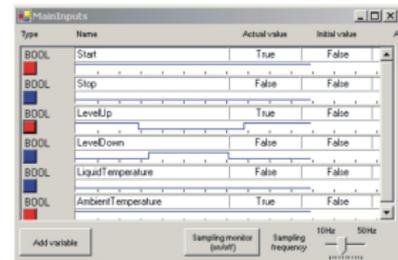
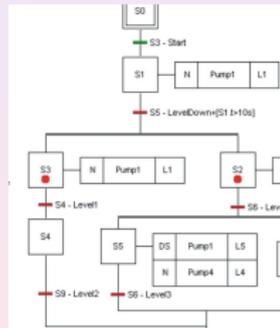


- Progettato seguendo l'approccio *object-oriented*
- L'architettura del software rispecchia il modello software definito dalla norma
- Sviluppato con la piattaforma *.Net Framework*



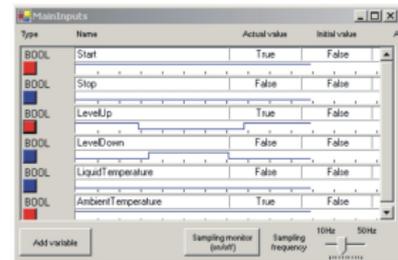
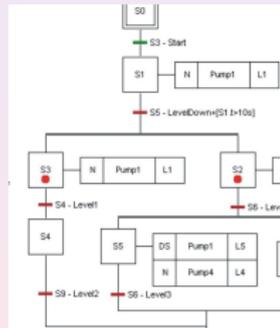
UniSim - Editor e Simulatore

- Editor grafico
- Simulatore che consente di monitorare graficamente
 - i programmi
 - l'andamento temporale delle variabili
- Durante la simulazione è possibile modificare programmi, task, liste delle variabili



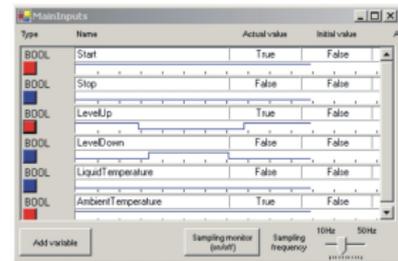
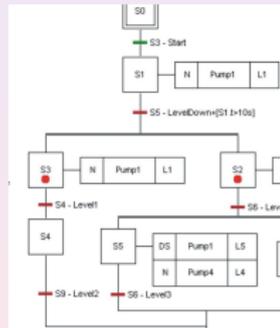
UniSim - Editor e Simulatore

- Editor grafico
- Simulatore che consente di monitorare graficamente
 - i programmi
 - l'andamento temporale delle variabili
- Durante la simulazione è possibile modificare programmi, task, liste delle variabili



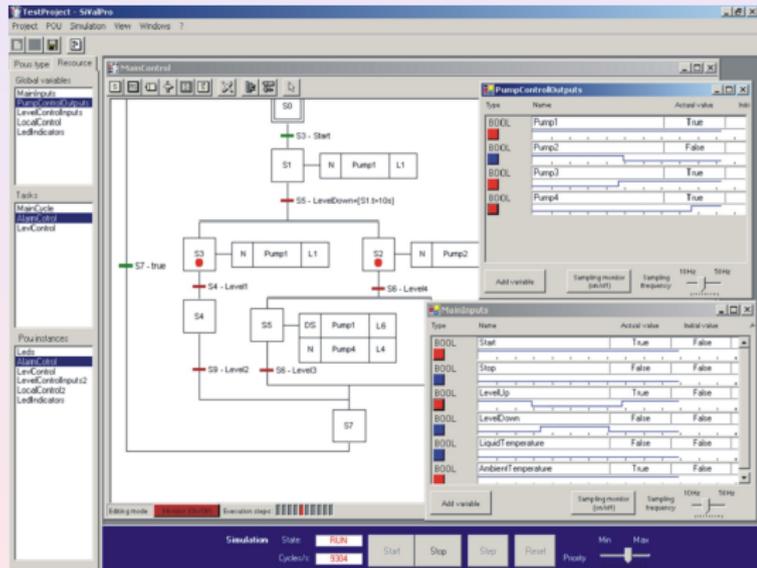
UniSim - Editor e Simulatore

- Editor grafico
- Simulatore che consente di monitorare graficamente
 - i programmi
 - l'andamento temporale delle variabili
- Durante la simulazione è possibile modificare programmi, task, liste delle variabili



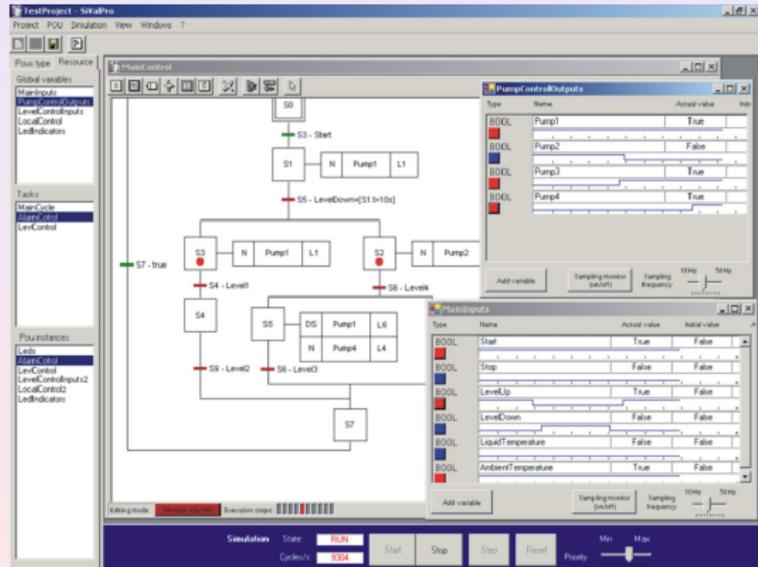
UniSim - Modulo per la validazione *hardware-in-the-loop*

- Eseguire il programma in tempo reale
- Si interfaccia con periferiche di I/O commerciali
 - definita un'interfaccia astratta per i driver
- Consente il monitoraggio grafico (da utilizzare preferibilmente in caso di fault)



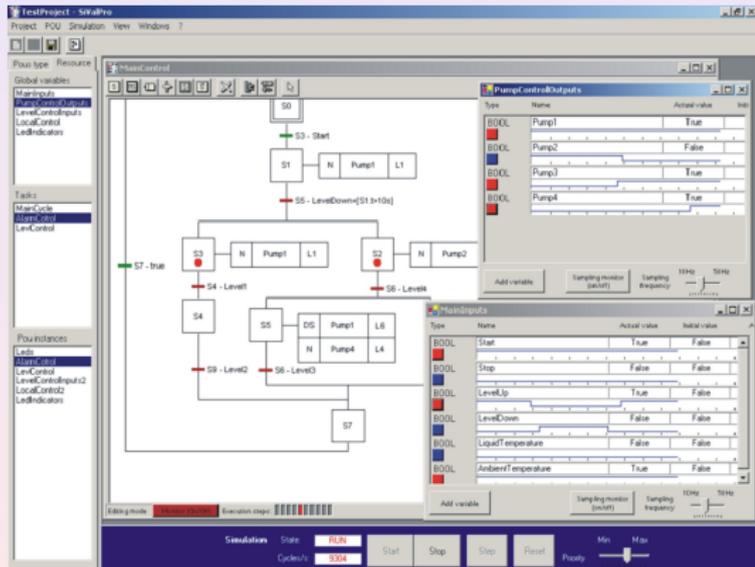
UniSim - Modulo per la validazione *hardware-in-the-loop*

- Eseguire il programma in tempo reale
- Si interfaccia con periferiche di I/O commerciali
 - definita un'interfaccia astratta per i driver
- Consente il monitoraggio grafico (da utilizzare preferibilmente in caso di fault)

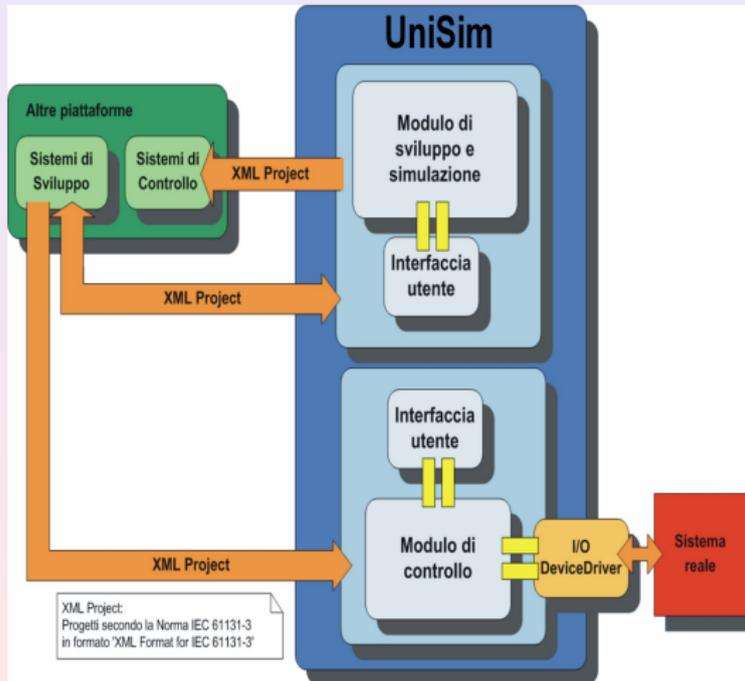


UniSim - Modulo per la validazione *hardware-in-the-loop*

- Eseguire il programma in tempo reale
- Si interfaccia con periferiche di I/O commerciali
 - definita un'interfaccia astratta per i driver
- Consente il monitoraggio grafico (da utilizzare preferibilmente in caso di fault)



UniSim



UniSim - Limitazioni

Work to do...

- Consente di effettuare la configurazione completa di una **sola** risorsa
- Il **solo** linguaggio di programmazione disponibile è l'SFC
- Consente l'utilizzo delle **sole** POU di tipo *program*
- È possibile utilizzare **solo** il tipo di dati booleano
- È stata definita l'interfaccia **solo** per i driver di schede di I/O digitali



UniSim - Limitazioni

Work to do...

- Consente di effettuare la configurazione completa di una **sola** risorsa
- Il **solo** linguaggio di programmazione disponibile è l'SFC
- Consente l'utilizzo delle **sole** POU di tipo *program*
- È possibile utilizzare **solo** il tipo di dati booleano
- È stata definita l'interfaccia **solo** per i driver di schede di I/O digitali



UniSim - Limitazioni

Work to do...

- Consente di effettuare la configurazione completa di una **sola** risorsa
- Il **solo** linguaggio di programmazione disponibile è l'SFC
- Consente l'utilizzo delle **sole** POU di tipo *program*
- È possibile utilizzare **solo** il tipo di dati booleano
- È stata definita l'interfaccia **solo** per i driver di schede di I/O digitali



UniSim - Limitazioni

Work to do...

- Consente di effettuare la configurazione completa di una **sola** risorsa
- Il **solo** linguaggio di programmazione disponibile è l'SFC
- Consente l'utilizzo delle **sole** POU di tipo *program*
- È possibile utilizzare **solo** il tipo di dati booleano
- È stata definita l'interfaccia **solo** per i driver di schede di I/O digitali



UniSim - Limitazioni

Work to do...

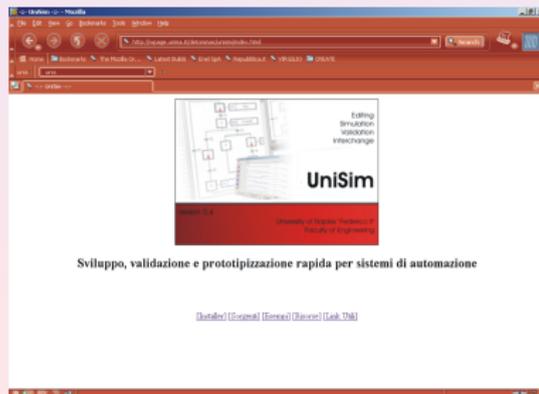
- Consente di effettuare la configurazione completa di una **sola** risorsa
- Il **solo** linguaggio di programmazione disponibile è l'SFC
- Consente l'utilizzo delle **sole** POU di tipo *program*
- È possibile utilizzare **solo** il tipo di dati booleano
- È stata definita l'interfaccia **solo** per i driver di schede di I/O digitali



Risorse Web

<http://wpage.unina.it/detommas/unisim/>

- file d'installazione
- sorgenti
- progetto d'esempio



Prossimamente. . .

Pubblicazione sul sito
SourceForge



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL, ...), delle rimanenti POU (*function, function block, ...*),...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL, ...), delle rimanenti POU (*function, function block, ...*),...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL,...), delle rimanenti POU (*function, function block,...*),...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL,...), delle rimanenti POU (*function, function block,...*),...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL,...), delle rimanenti POU (*function, function block,...*),...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL, ...), delle rimanenti POU (*function*, *function block*, ...), ...
- Grazie



Conclusioni

- Primo tool che utilizza *XML Formats for IEC 61131-3* per lo scambio dati
- Rispetta la norma IEC
 - Portabilità del software
- Prototipizzazione rapida e la validazione *hardware-in-the-loop* di semplici sistemi di automazione (*SoftPLC*)
- È libero (licenza GPL) e facile da utilizzare (help in linea + tutorial)
- Implementazione degli altri linguaggi previsti dallo standard (Ladder, IL, ...), delle rimanenti POU (*function*, *function block*, ...), ...
- Grazie

