

L'architettura di sviluppo per PLC presente in Laboratorio

Introduzione

Le postazioni presenti in laboratorio sono configurate per consentire lo sviluppo e la validazione di software per PLC. Ogni postazione è costituita da:

- un PC standard con scheda di I/O Smartlab della Decision Computer;
- un PLC Allen Bradley SLC5/03.

Sul PC sono stati installati i seguenti prodotti:

- *RS Logix 500* [4]; è il sistema di sviluppo della Rockwell Automation che consente di realizzare progetti d'automazione utilizzando i controllori programmabili Allen Bradley.
- *RS Linx* [5]; è il pacchetto software della Rockwell Automation che permette di gestire le comunicazioni tra PC e i controllori Allen Bradley.
- *Decision Industrial Interface* [6], una DLL contenente l'API per pilotare la scheda di I/O Smartlab.
- i simulatori degli impianti relativi alle esercitazioni; questi simulatori non sono altro che programmi che gestiscono la scheda di I/O Smartlab in maniera opportuna utilizzando la Decision Industrial Interface.

I passi che si dovranno effettuare per poter sviluppare un nuovo progetto d'automazione utilizzando queste postazioni sono:

1. configurazione del collegamento tra PC e PLC tramite RS Linx;
2. creazione di un nuovo progetto di automazione con RS Logix 500;
3. validazione del progetto software mediante l'utilizzo di un simulatore d'impianto che utilizzi la scheda di I/O Smartlab.

Nei paragrafi successivi verrà fatta una breve descrizione del PLC SLC5/03, della scheda di I/O Smartlab, dopodiché verranno presentati i collegamenti tra i vari componenti che compongono la postazione di lavoro.

Il Controllore programmabile Allen Bradley SLC5/03

Il controllore presente in laboratorio è un PLC Allen Bradley serie SLC 500, montato su uno chassis modulare insieme al modulo alimentatore e a due moduli di I/O digitali [9-10].

Il modulo processore SLC5/03 (Fig. 1) può essere considerato un PLC di classe media ed, in particolare:

- è dotata di 8 KB di memoria per il programma utente;
- può pilotare fino a 4096 d'ingresso e 4096;
- possiede due canali di comunicazione integrati (DH-485 e RS-232);

- possiede un set d'istruzioni esteso che comprende istruzioni matematiche avanzate (come funzioni trigonometriche ed esponenziale)¹ e funzioni speciali (per esempio la funzione PID);
- consente la modifica del programma online, vale a dire mentre il PLC è in RUN.

La scheda processore è montata nello slot 0 dello chassis; al suo fianco sono alloggiati i due moduli di I/O, uno con 16 ingressi e l'altro con 16 uscite digitali (rispettivamente 1746-IB16 e 1746-OW16). Un esempio di modulo di I/O è riportato in Fig. 1.

Sul pannello frontale del modulo processore è presente un insieme di led: alcuni di essi indicano lo stato del processore (Run, Fault, ecc.), altri indicano l'attività sulle porte di comunicazione. Sempre sul pannello frontale, poi, ci sono la chiave che permette di selezionare la modalità di funzionamento e le due porte di comunicazione (DH-485 e RS-323). La porta di comunicazione seriale (RS-232) supporta diversi protocolli di comunicazioni (DH-485, DF1 Full-Duplex e Half-Duplex), mentre la porta DH-485 può essere utilizzata solo con il protocollo omonimo.

In caso di fault del processore è possibile risalire all'errore che lo ha determinato osservando la combinazione di led accesi sul pannello frontale.

Anche sul pannello frontale dei moduli di I/O sono presenti dei led che, in questo caso, indicano lo stato di ognuno degli ingressi e delle uscite. La tensione d'ingresso e d'uscita è di 24 V dc.

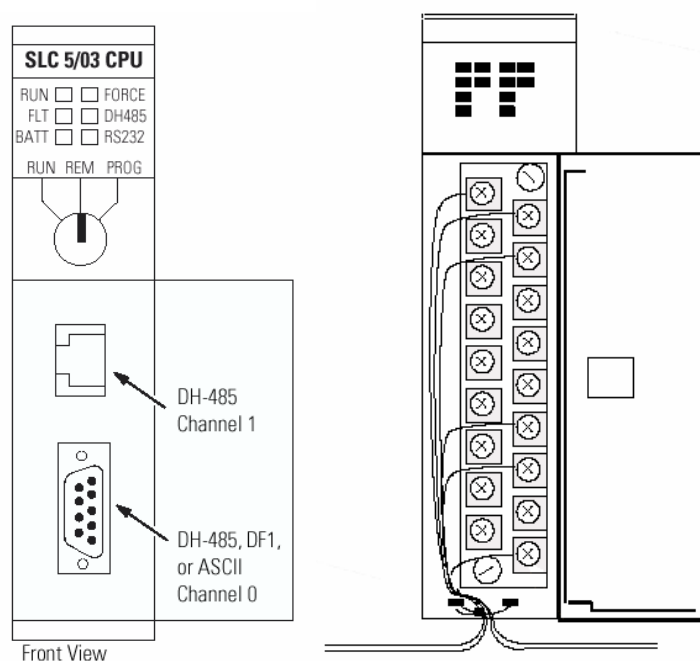


Figura 1 -SLC5/03 e modulo I/O 1746

La scheda di I/O Smartlab

La scheda Smartlab è una scheda di I/O digitale su bus ISA con 16 ingressi e 16 uscite, e consente di collegare un normale PC direttamente ai moduli di I/O digitali del PLC SLC5/03.

¹ Si noti che istruzioni che al giorno d'oggi vengono considerate standard per un normale personal computer, vengono presentate come "avanzate" per i controllori a logica programmabile.

Insieme al driver, il produttore fornisce un'API (*Decision Industrial Interface*) che consente di pilotare la scheda attraverso diversi linguaggi di programmazione (Visual Basic, C/C++, Delphi, Java) [6].

Questo componente permette di accedere alla scheda mediante un nome simbolico e sviluppare dei programmi che simulino il funzionamento dell'impianto, permettendo, quindi, di validare il software di controllo del PLC.

L'interfaccia è comune a tutti i prodotti della casa produttrice e non è particolare per la scheda Smartlab. Questa caratteristica garantisce un buon livello di astrazione e permette minimizzare le modifiche al software quando si vuole cambiare solo l'hardware di acquisizione.

Decision Industrial Interface, inoltre, supporta il multi-threading, cioè consente a thread diversi appartenenti alla stessa applicazione di accedere alla scheda di I/O in maniera concorrente.

L'insieme di istruzioni messe a disposizione può essere suddiviso nelle seguenti categorie:

- *Connessione alla scheda* – queste funzioni permettono di aprire un collegamento con la scheda, in maniera analoga a quanto viene fatto quando si vuole accedere ad un file.
- *Caratteristiche della scheda* – queste funzioni permettono di recuperare informazioni che riguardano la particolare scheda, ad esempio è possibile recuperare informazioni circa il numero di ingressi ed uscite digitali.
- *Gestione dell'I/O digitale* – queste funzioni permettono di gestire un eventuale I/O digitale presente sulla scheda. È possibile, per esempio, interrogare lo stato di un ingresso oppure assegnare il valore logico di un'uscita.
- *Gestione dell'I/O analogico* – queste funzioni permettono di gestire l'I/O analogico qualora la scheda ne possieda uno.
- *Gestione dei timer* – queste funzioni consentono di gestire eventuali timer hardware presenti sulla scheda.
- *Gestione dei contatori* – queste funzioni consentono di utilizzare gli I/O digitali come contatori.
- *Gestione delle notifiche* – queste funzioni permettono di agganciare delle callback a particolari eventi (per esempio variazione di un valore d'ingresso).
- *Acquisizione continua in background* – queste funzioni permettono di acquisire gli ingressi in maniera continua senza la necessità di generare comandi espliciti per ogni singola acquisizione.

Ovviamente non tutte le funzioni sono disponibili per ogni tipo di scheda.

In appendice A è riportato un semplice esempio di utilizzo della *Decision Industrial Interface* sviluppato in C++.

La postazione di lavoro

In Fig. 2 è riportato uno schema delle postazioni presenti in laboratorio nella quale sono messi in evidenza i collegamenti tra i vari componenti.

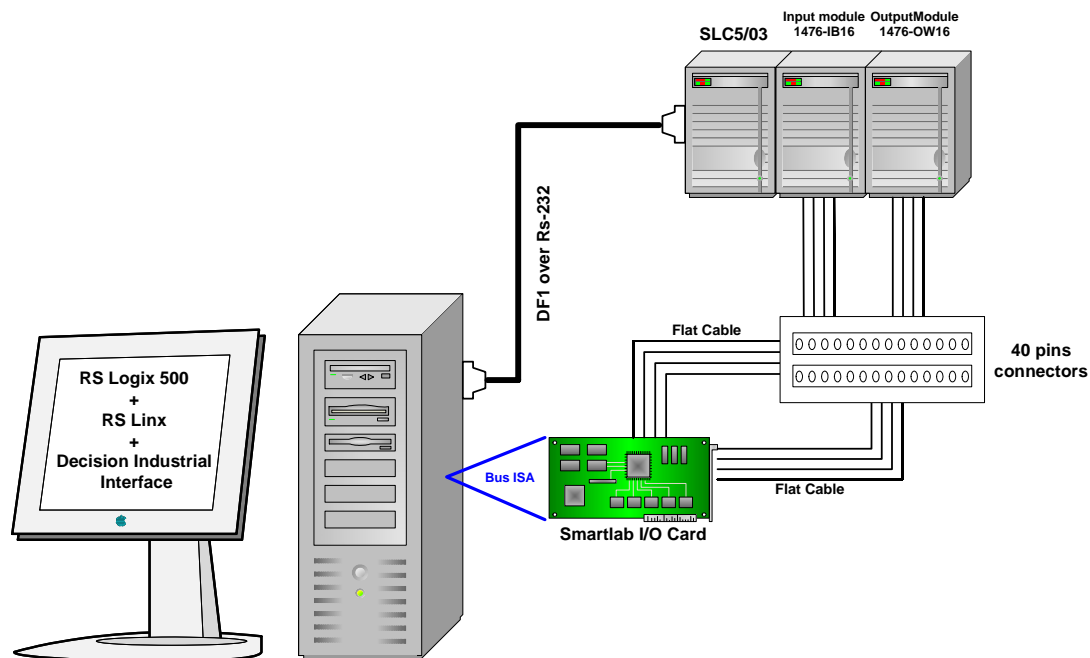


Figura 2 - La postazione di lavoro in laboratorio

Il PC svolge la doppia funzione di *terminale di programmazione* per il PLC e di *simulatore d'impianto*, attraverso la scheda ISA Smartlab.

Affinché sia possibile sviluppare un *progetto di automazione* per l'SLC5/03, come già anticipato, sul PC sono stati installati i sistemi di sviluppo RS Logix 500 e Rs Linx.

Per *progetto di automazione* si intende l'insieme:

- dei dati di configurazione del controllore programmabile (ad esempio la configurazione dell'hardware all'interno dello chassis);
- della configurazione delle reti di comunicazione;
- dei programmi e delle strutture dati.

Considerato come *terminale di comunicazione* il PC comunica con il modulo processore attraverso la porta seriale utilizzando il protocollo proprietario DF1.

Attraverso questo collegamento è possibile inviare al modulo processore i dati di configurazione e il programma utente ed è possibile effettuare il debug on-line dell'applicazione sviluppata.

Si osservi che il collegamento seriale non permette di collegare lo stesso PC a più moduli processore contemporaneamente.

Per ovviare a questo inconveniente è possibile utilizzare la porta DH-485 presente sul modulo processore; in questo caso, però, sarebbe necessario installare sul personal computer una scheda che supporti questo protocollo di comunicazione. Inoltre anche il protocollo DH-485 ha delle limitazioni:

- consente di collegare tra loro al più 32 dispositivi;

- la lunghezza massima dei collegamenti è di 1200 m, estendibili fino a 2400 attraverso l'utilizzo di dispositivi particolari.

La rete DH-485 è una rete di campo con accesso a token, quindi mantiene un elevato grado di affidabilità.

I protocolli di comunicazione DH-485 e quelli che utilizzano il collegamento seriale RS-232 non rappresentano le uniche possibilità di collegamento tra un PC ed il modulo processore. In alcuni casi potrebbe presentarsi la necessità di utilizzare una rete Ethernet e il protocollo TCP/IP per la comunicazione (si pensi ad architetture distribuite in cui il modulo processore sia molto distante dalla stazione di sviluppo). Se si presentasse una necessità di questo tipo, è necessario l'acquisto di un modulo di comunicazione aggiuntivo che permetta di collegare un PLC della serie SLC ad una rete di questo tipo. Ovviamente l'utilizzo di questa soluzione comporta un decadimento delle prestazioni in termini di determinismo della comunicazione ed è più indicata nel caso di collegamenti tra modulo processore e terminale di programmazione oppure sistema di supervisione (SCADA).

Il collegamento tra l'I/O della scheda Smartlab (che viene gestita dal *simulatore d'impianto* attraverso la Decisione Industrial Interface) e i moduli d'ingresso e uscita del PLC, infine, avviene attraverso due morsettiere a 40 pin e dei cavi flat che escono direttamente dalla scheda.

Riferimenti

[1] <http://www.ab.com/>, sito ufficiale Allen Bradley.

[2] <http://www.rockwellautomation.com/>, sito ufficiale Rockwell Automation.

[3] <http://www.decision.com.tw/>, sito ufficiale.

[4] *RS Logix 500 – Getting Results Guide*, LG500-GR001A-EN-P, 2002 Rockwell Automation.

[5] *RS Linx – Getting Results Guide*, 2002 Rockwell Automation.

[6] *Decision Industrial Interface V1.8 – A Universal Application Programmer's Interface To Data Acquisition Products – User Manual*, NT-Ware Systemprogrammierungs.

[7] *Smartlab - Operation Manual*, Decision Computer International.

[8] *SLC 500 Modular Hardware Style – User Manual*, 2003 Rockwell Automation.

[9] *Digital I/O Modules - Installation Instructions*, 2003 Rockwell Automation.

[10] *Power Supplies – Installation Instructions*, 2003 Rockwell Automation.

Appendice A

Di seguito è riportato un semplice esempio in C++² che illustra l'utilizzo delle funzioni della Decision Industrial Interface per la gestione della scheda Smartlab.

```
#include <Dii.h>
#include <stdio.h>
#include <stdlib.h>

void main(){
    // Controllo delle schede installate
    int deviceIndex = 0;
    const int deviceNameLength = 128;
    char deviceName[deviceNameLength];
    printf("Seleziona la scheda desiderata:\n");
    while(DiiGetInstalledDevice(deviceIndex,deviceName,deviceNameLength) == TRUE){
        printf("%d) %s\n",deviceIndex,deviceName);
        deviceIndex++;
    }
    if(deviceIndex == 0){
        printf("Non ci sono schede installate su questo PC\n");
        return;
    }
    // Seleziona una scheda
    printf("\nSeleziona:");
    int selectedDevice;
    scanf("%d",&selectedDevice);
    while(selectedDevice >= deviceIndex){
        printf("\nScelta errata, riprova:");
        scanf("%d",&selectedDevice);
    }
    if(DiiGetInstalledDevice(selectedDevice,deviceName,deviceNameLength) != TRUE){
        printf("\nErrore");
        return;
    }
    printf("\nDevice selezionata: %s\n",deviceName);
    // Apre un collegamento con la scheda
    HANDLE h = DiiOpenNamedDevice(deviceName,FALSE);
    if(h == INVALID_HANDLE_VALUE){
        printf("Errore durante il collegamento alla scheda %s\n",deviceName);
        return;
    }
    // Caratteristiche della scheda
    printf("\n\n %s - CARATTERISTICHE\n\n",deviceName);
    // I/O digitale
    DWORD nOfDigitalIn = 0;
    DWORD nOfDigitalOut = 0;
    if(DiiGetNumberOfDigitalInOutChannels(h,&nOfDigitalIn,&nOfDigitalOut) == FALSE){
        printf("La scheda %s non possiede canali di I/O digitale\n",deviceName);
    }else{
        printf("%d Canali d'ingresso digitali\n",nOfDigitalIn);
        printf("%d Canali d'uscita digitali\n",nOfDigitalOut);
    }
    // I/O analogico
    DWORD nOfAnalogIn = 0;
    DWORD nOfAnalogOut = 0;
    if(DiiGetNumberOfAnalogInOutChannels(h,&nOfAnalogIn,&nOfAnalogOut) == FALSE){
        printf("La scheda %s non possiede canali di I/O analogico\n",deviceName);
    }else{
        printf("%d Canali d'ingresso analogici\n",nOfAnalogIn);
        printf("%d Canali d'uscita analogici\n",nOfAnalogOut);
        int resolution = DiiGetResolution(h);
        printf("La risoluzione dei canali analogici è %d bit\n",resolution);
    }
    // Se la scheda possiede output digitali effettua un test
    if(nOfDigitalOut != 0){
        int value = 1;
        printf("\n");
    }
}
```

² Per effettuare la compilazione di questo codice utilizzando il compilatore Microsoft è sufficiente utilizzare il comando:
cl nomefile.cpp [path/]Dii.lib

chiaramente è necessario che la *Decision Industrial Interface* sia installata sul PC sul quale si sta lavorando.

```
    for(int i = 0;i<100;i++){
        printf("Test dell'output digitale %c\015",47+45*(value % 2));
        BOOL state = (value++ % 2) == 1;
        if(DiiSetDigitalBit(h,0,state) != TRUE){
            printf("Errore durante il test dell'output della scheda%s\n",deviceName);
            break;
        }
        Sleep(500);
    }
}
printf("\n");
// Chiude il collegamento con la scheda
if(DiiCloseDevice(h) == FALSE){
    printf("Errore durante la chiusura del collegamento con la scheda %s\n",deviceName);
}
}
```