

---

ALMA MATER STUDIORUM  
UNIVERSITA DEGLI STUDI DI BOLOGNA  
SEDE DI CESENA

---

**Seconda Facoltà di Ingegneria con sede a Cesena**  
Corso di Laurea Specialistica in  
INGEGNERIA ELETTRONICA E DELLE TELECOMUNICAZIONI

**STUDIO ED IMPLEMENTAZIONE DI UN  
POSIZIONATORE INTERFACCIATO SU  
RETE PROFINET**

Tesi in:  
SISTEMI E TECNOLOGIE PER L'AUTOMAZIONE LS

*Presentata da:*  
**FILIPPO RIVALTA**

*Relatore:*  
**Prof. Ing. CARLO ROSSI**

*Correlatore:*  
**Ing. MAURIZIO MORETTI**

**- Terza Sessione -**  
*Anno Accademico 2010 / 2011*







## Indice

1 INTRODUZIONE.....	5
2 ANALISI DEL SISTEMA BOSS.....	9
2.1 Principi di funzionamento.....	11
2.1.1 Fasi di lavoro.....	12
2.2 Aspetti critici e possibili soluzioni.....	15
3 STANDARD DI COMUNICAZIONE NELL'AUTOMAZIONE.....	17
3.1 Bus di Campo (Fieldbus).....	17
3.1.1 Safety.....	19
3.2 IEC 61158 .....	20
3.2.1 Il Livello Physical: IEC 61158.2.....	22
3.2.2 Il Livello Data Link: IEC 61158.3-4.....	23
3.2.3 Il Livello Application: IEC 61158.5-6.....	25
3.2.4 Il Livello di System & Network Management: IEC 61158.7 .....	27
3.3 Industrial Ethernet.....	29
3.3.1 Sicurezza di rete.....	33
3.3.2 Affidabilità della rete.....	33
3.3.3 Traffico Real-Time con IGMP Snooping.....	35
3.3.4 Quality of Service.....	35
4 BUS DI CAMPO IMPLEMENTABILI SU CONTROLLORI SIEMENS.....	37
4.1 Multi Point Interface .....	37
4.2 AS-Interface.....	37
4.3 ProfiBUS.....	40
4.3.1 Tecnologie di trasmissione e topologia di rete.....	41
4.3.2 Comunicazione.....	44
4.3.3 Profili applicativi.....	46
4.3.4 Integrazione dei dispositivi e certificazione.....	46
4.4 ProfiNET.....	47
4.4.1 ProfiNET CC-A (Conformance Class A).....	51
4.4.2 ProfiNET CC-B (Conformance Class B).....	54
4.4.3 ProfiNET CC-C (Conformance Class C).....	55
4.4.4 Funzioni opzionali aggiuntive.....	57
4.4.5 Profili Applicativi.....	59
4.4.6 Installazione di una Rete ProfiNET.....	59
4.4.7 Sicurezza.....	60
4.4.8 Certificazione dei prodotti.....	61
4.4.9 Limiti del ProfiNET.....	61
5 RICERCA COMPONENTISTICA PER IL NUOVO SISTEMA.....	63
5.1 Controllori.....	63

5.1.1 SIMATIC S7-1200.....	64
5.1.2 SIMATIC S7-300.....	65
5.1.3 SIMATIC S7-400.....	66
5.2 Azionamenti.....	67
5.2.1 Serie G.....	68
5.2.2 Serie S.....	68
6 IMPLEMENTAZIONE DEL NUOVO SISTEMA.....	71
6.1 Componenti scelti.....	72
6.2 Schema del sistema.....	74
6.3 Comunicazione tra PLC e S110.....	75
6.4 Stati di funzionamento realizzati e standardizzati.....	77
6.4.1 Descrizioni degli stati di funzionamento.....	79
6.5 Struttura del software PLC.....	81
6.6 Migrazione del software.....	83
6.6.1 Gestione della memoria.....	83
6.6.2 Utilizzo della CPU.....	87
6.7 Interfaccia uomo macchina - HMI.....	89
6.7.1 Struttura delle pagine.....	90
6.8 Web-Server.....	91
6.8.1 Comandi AWP.....	92
6.8.2 Web Server realizzato.....	95
6.8.3 Valutazione del Web-Server presente sul S7-1200.....	96
7 CONCLUSIONI.....	99
8 BIBLIOGRAFIA.....	101
9 Ringraziamenti.....	103
Appendice A: LISTA COMPONENTI ACQUISTATI.....	104
Appendice B: MOVIMENTI STANDARDIZZATI.....	105







# 1 INTRODUZIONE

Lo scopo di questa tesi è quello modificare un macchinario per il caricamento automatico di barre su un tornio monomandrino sostituendo l'azionamento analogico presente con un azionamento digitale.

Il macchinario in questione è il BOSS realizzato dalla IEMCA di Faenza. Il BOSS attualmente è comandato da un PLC Siemens tipo: S7-300 ed è abbinato ad un azionamento realizzato dalla “Elmo”. Questo tipo di configurazione richiede un PLC con delle buone prestazioni in quanto tutto il carico di lavoro grava su di esso. Infatti, il PLC si trova sia a dover comandare il motore tramite il posizionatore analogico che a controllare l'intero processo macchina. Questo comporta che il PLC dovrà necessariamente avere un modulo per le uscite analogiche ed uno di Motion Control al quale è collegato l'encoder che verifica se il pezzo ha raggiunto la corretta posizione.

Si è scelto quindi di sostituire l'azionamento di tipo analogico con uno digitale, questa soluzione permette di ridurre il carico di lavoro del PLC in quando tutta la parte di verifica del corretto posizionamento viene gestita dall'azionamento, quindi non sono più richiesti moduli di Motion Control e uscite analogiche, inoltre questa soluzione riduce il cablaggio del sistema e permette una maggior flessibilità.

In base alla scelta dell'azionamento verrà decisa anche la sostituzione del PLC S7-300 con un nuovo PLC di fascia più bassa in quanto il suo carico di lavoro non richiede più quelle performance che erano necessarie nel vecchio sistema BOSS.

Di fatto nel nuovo sistema i costi e la complessità migreranno dal PLC al azionamento digitale, questo permette una riduzione della spesa sulla componentistica e sul cablaggio. Ulteriori vantaggi di questa nuova soluzione risiedono nella migliore flessibilità del sistema. Si vedrà in seguito che i principali standard per la comunicazione digitale nel settore dell'automazione sono pensati per lavorare su Bus dati. In questo modo è possibile espandere un sistema aggiungendo semplicemente i nuovi moduli sul bus. Inoltre, i più recenti standard di comunicazione nell'automazione, sono compatibili con l'Industrial Ethernet e alcuni di essi (Ad esempio PROFINET), grazie a stringenti vincoli sulle tempistiche possono operare al livello di comando anche in quei settori dove è necessaria una comunicazione di tipo Hard Real Time. Questo permette di collegare tutti i livelli della piramide dell'automazione con un unica rete.

Quindi sarà possibile monitorare e gestire da remoto (anche tramite web-server)



Figura 1.1: Piramide dell'automazione

tutta la catena di comando. Infine, questi protocolli essendo compatibili con lo standard Ethernet sono facilmente implementabili anche su WLAN consentendo una rapida messa in rete di tutti i macchinari di un impianto.

I macchinari per il caricamento barre dei torni devono poter essere flessibili e facilmente configurabili in quanto lo stesso macchinario dovrà poter essere utilizzabile con diversi tipi di tornio di marche e modelli differenti. Il nuovo sistema dovrà quindi tenere come punto di riferimento per il suo sviluppo la massima flessibilità e la facilità di configurazione. Attualmente il sistema BOSS ha i principali parametri di configurazione gestibili dal pannello operatore, altri invece per poter essere modificati dall'utente richiedono una password di sicurezza. L'elevata flessibilità e la velocità di messa in opera sono essenziali per poter essere competitivi nel mercato.

L'obiettivo di questa tesi è quello di realizzare un prototipo simulatore del macchinario BOSS composto da un nuovo PLC, un azionamento e un motore sincrono, gli ingressi saranno simulati tramite degli interruttori. Questo passaggio è necessario per poter valutare i reali vantaggi di questa soluzione senza gravare troppo sui costi della ricerca. Il progetto inizialmente verrà sviluppato sul PLC S7-300 (già presente nel BOSS) perché è un PLC stabile e ben conosciuto, successivamente il codice verrà migrato su un nuovo PLC meno costoso.

Si vedrà in seguito che il PLC adottato per questo progetto e il protocollo di comunicazione utilizzato consentono di realizzare un web-server sul macchinario che può accedere alle variabili del PLC. Si è scelto di sviluppare alcune pagine HTML per testare la funzionalità di questo web-server in

previsioni di poter effettuare in futuro un'assistenza clienti da remoto.

La IEMCA attualmente lavora principalmente con controllori (PLC, PC industriali) prodotti da Siemens. Quindi la ricerca della componentistica del nuovo sistema è stata limitata all'interno della gamma di prodotti Siemens o compatibili con essi in maniera tale da poter utilizzare il know-how dell'ufficio tecnico e andare incontro alle esigenze dei clienti IEMCA.

Questa scelta sembrerebbe contrastare con la necessità dell'azienda di realizzare prodotti al minor prezzo possibile a parità di qualità. Questo è vero in parte, sarà fondamentale infatti che i componenti utilizzati nel progetto soddisfino i principali standard di comunicazione nel campo dell'automazione. In maniera tale che in futuro sia possibile sostituire alcuni moduli Siemens del sistema con altri prodotti da aziende concorrenti con costi minori, a parità di prestazioni. Si fa notare che già adesso il sistema BOSS della IEMCA utilizza un azionamento “*Elmo*” gestito da un PLC Siemens.

*“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”*

## 2 ANALISI DEL SISTEMA BOSS

Il sistema BOSS è un macchinario per il caricamento automatico di barre su torni monomandrino. Questi macchinari prelevano una barra grezza dal magazzino e la spingono fino nel tornio che effettua la lavorazione. Le barre grezze possono superare i 4 metri di lunghezza mentre tipicamente un tornio effettua delle lavorazioni più corte. Quindi, finita la tornitura del pezzo, il tornio tronca la parte di barra lavorata che cade in un magazzino ed a questo punto il caricatore può far avanzare la restante parte di barra grezza nel tornio per una nuova lavorazione. Si ripete questa procedura fino a quando la barra grezza è troppo corta per essere lavorata (*spezzone*). A questo punto il macchinario esegue la fase di cambio barra e ricomincia la lavorazione.



Figura 2.1: BOSS modello 552-HD

Si intuisce che per un corretto funzionamento della macchina di caricamento barre è necessario che esista un interfacciamento tra la macchina e il tornio per poter svolgere correttamente tutte le fasi della lavorazione. Questo rende necessario che il BOSS sia estremamente flessibile per poterlo interfacciare facilmente con i diversi tipi di tornio monomandrino in commercio e per poterlo adattare alle esigenze del cliente.

Il BOSS è composto da un pannello operatore di tipo touch screen o di tipo testuale con il quale l'operatore può interfacciarsi con la macchina. Inoltre, tramite il pannello, l'installatore può modificare i principali parametri di programmazione della macchina per poterla adattare alle esigenze del cliente.

Il motore da 3Nm è controllato da un azionamento realizzato da “Elmo”. L'azionamento è gestito dal PLC tramite due linee analogiche che definiscono la velocità di rotazione del motore e la coppia massima che deve mantenere. Ed un



attuatori.

Il carrello spingi barra è posizionato su una catena mossa dal motore che è collegato alla prima ruota dentata. (Vedi Fig. 2.3). La catena, a causa di un assestamento meccanico non quantificabile a priori, negli anni tende ad allentarsi; quindi il carrello eseguirà un percorso più lungo. A causa di questo assestamento meccanico non è più possibile stabilire una corretta associazione tra i giri del motore e il posizionamento del pezzo in uscita. Per individuare quanto il pezzo avanza nel tornio, occorre monitorare con un altro encoder, la seconda ruota dentata. I valori di rotazione della seconda ruota dentata e la “bandierina” (Vedi fase di lavoro numero 6 Cap. 2.1.1 Fasi di lavoro a pag. 12) permettono una buona precisione di movimento priva di degradazioni.

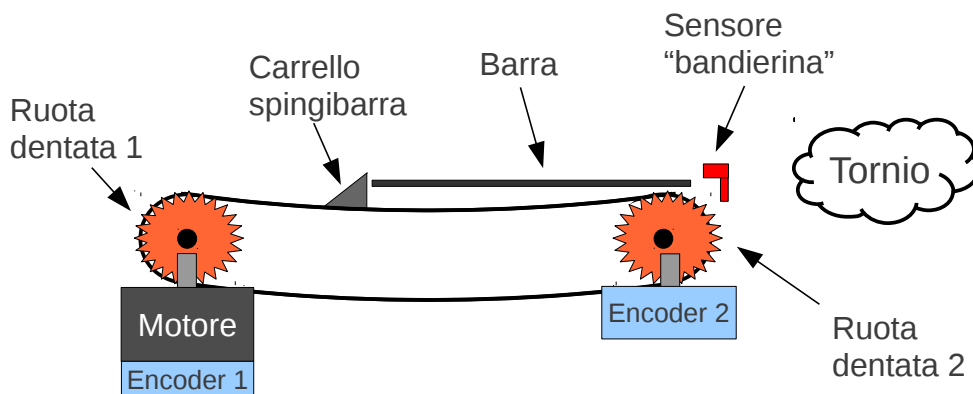


Figura 2.3: Schema doppio encoder BOSS, con allentamento accentuato della catena

## 2.1 Principi di funzionamento

Nel BOSS sono individuabili due macro funzioni: il cambio barra e l'avanzamento del pezzo nel tornio durante la lavorazione della barra.

Il ciclo di lavoro di questa macchina va dalla fine lavorazione di una barra, all'inserimento della successiva barra nel tornio (chiamata anche intestatura) e quindi all'inizio di lavorazione della stessa.

Questo ciclo di lavoro per il BOSS è suddiviso in nove fasi per il cambio barra e una fase per l'avanzamento del pezzo (fase 0), alcune di queste fasi prevedono un movimento del carrello altre fasi gestiscono solo gli ingressi I/O. Alle fasi in cui è previsto un movimento del carrello è associata una “ricetta”, salvata in un database, nella quale sono descritti i parametri di movimento che il motore dovrà attuare (velocità, coppia, accelerazione, etc etc). Questa soluzione permette di modificare rapidamente i parametri delle fasi di lavoro con

movimento del pezzo senza doverli andare a ricercare all'interno del programma.

### 2.1.1 Fasi di lavoro

All'avvio il sistema ha bisogno di sapere in quale posizione si trova il carrello quindi tipicamente il BOSS richiede un azzeramento dell'asse prima di potere eseguire le fasi di lavoro in modalità automatica. Esistono casi però in cui il cliente ha necessità di spegnere il BOSS prima che esso abbia finito il proprio ciclo.

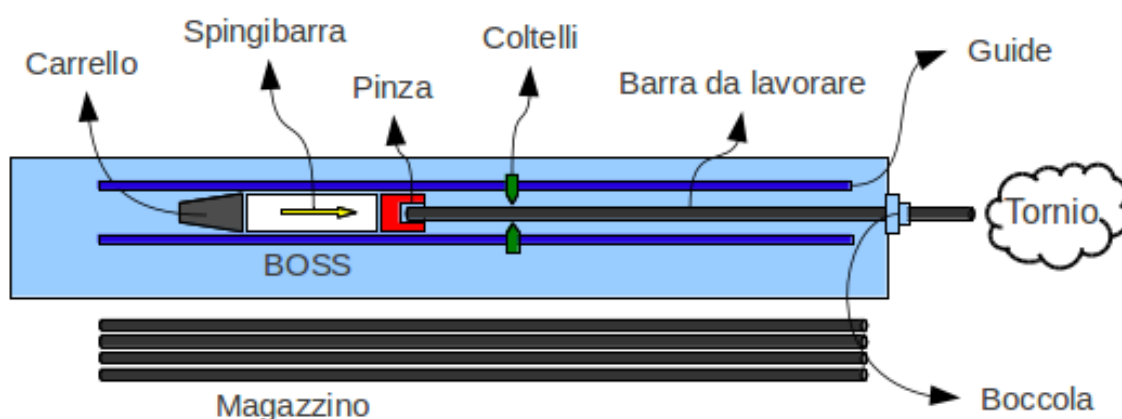


Figura 2.4: Schema BOSS durante lavorazione barra

In questi casi, all'avvio il BOSS non può effettuare lo zero asse, in quanto potrebbe avere la barra inserita nel madrino del tornio, quindi imposta il riferimento del carrello al valore di posizione salvato prima di essere spento. Questa procedura è chiamata preset dell'encoder.

Dopo aver azzerato l'asse o presetato l'encoder il BOSS può eseguire le fasi di lavoro; che sono:

- **Fase 0: Lavorazione della barra.** In questa fase il tornio esegue la tornitura del pezzo. Terminata la lavorazione richiede al BOSS di fare avanzare la parte grezza della barra per eseguirne una nuova. Questa fase termina quando la barra grezza è troppo corta per poter essere lavorata.
- **Fase 1: Ritorno spingi barra:** Finita la lavorazione della barra, lo spingibarra ritorna indietro fino a posizionare lo spezzone di fronte ai coltelli.
- **Fase 2: Azzeramento del asse:** Il carrello continua la sua corsa indietro per effettuare lo zero asse. Lo zero asse viene eseguito ogni ciclo di



lavorazione per assicurarsi che non ci siano degradazioni nella precisione dei movimenti.

- **Fase 3: Estrazione spezzone:** I coltelli si chiudono e lo spezzone viene estratto dalla pinza. Il carrello avanza e spinge fuori lo spezzone, senza afferrarlo, poi ritorna indietro. La barra può essere spinta fuori attraverso una finestra presente sotto le guide del macchinario BOSS o in uscita al BOSS.
- **Fase 4: Test spezzone:** Verifica che lo spezzone sia caduto fuori dal macchinario.
- **Fase 5: Apertura guide e asse libero:** La guida si solleva portando con sé lo spingi barra e la pinza. In questo modo è possibile inserire un'altra barra. La scelta di sollevare oltre alle guide anche lo spingi barra e la pinza consente di realizzare un macchinario più corto. Infatti lo spingi barra più la pinza sono lunghi circa un metro. (Fig. 2.5). Durante questa fase il carrello deve essere libero di muoversi e quindi non deve essere frenato dal motore. Questo è necessario perché l'apertura delle guide, che è meccanica, deve poter spostare di pochi centimetri il carrello.

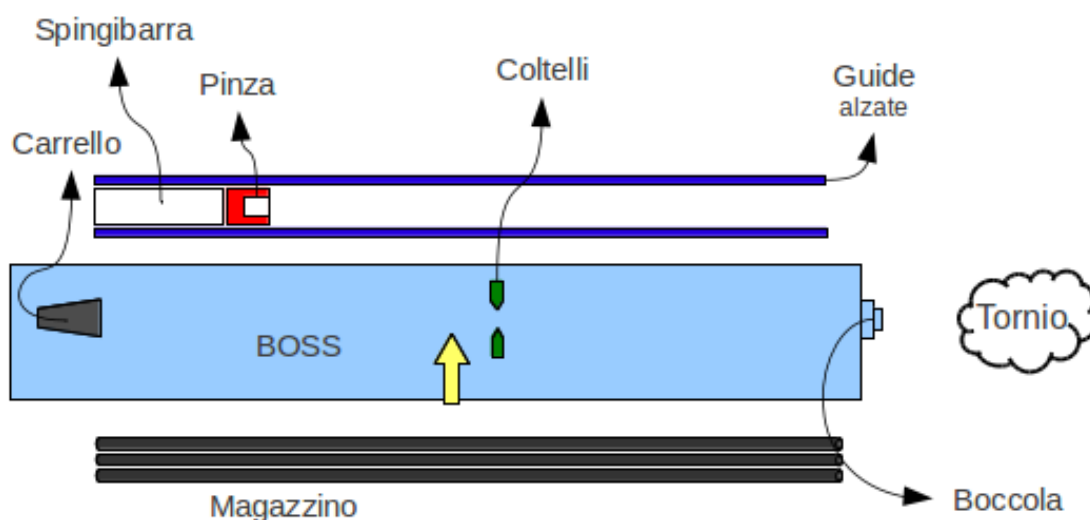


Figura 2.5: Schema BOOS durante un cambio barra

- **Fase 6: Preavanzamento barra:** A questo punto cade una nuova barra dal magazzino all'interno del macchinario. Il pezzo deve essere fatto avanzare (fino a un punto chiamato B) per fare in modo che quando le guide verranno richiuse, lo spingi barra non urti contro di esso.

Durante questa fase generalmente viene abbattuta la bandierina di intestatura. La bandierina di intestatura è un sensore posto ad una distanza nota dal tornio (Parametro B1C) e serve per misurare quanto è lunga la barra e per calcolare il punto C, che è il filo pinza del tornio (Fig. 2.6). Se la barra è corta, non colpisce la bandierina in questa fase ma la colpisce all'inizio della fase 9.

- **Fase 7: Ritorno carrello:** Il carrello ritorna indietro.
- **Fase 8: Chiusura guide e introduzione barra nella pinza:** Le guide e i coltelli si chiudono, (anche qui come per la fase 5 occorre che l'asse sia libero di muoversi) il carrello si unisce allo spingi barra e avanza fino a inserire la barra nella pinza. A questo punto i coltelli si aprono e liberano il pezzo.

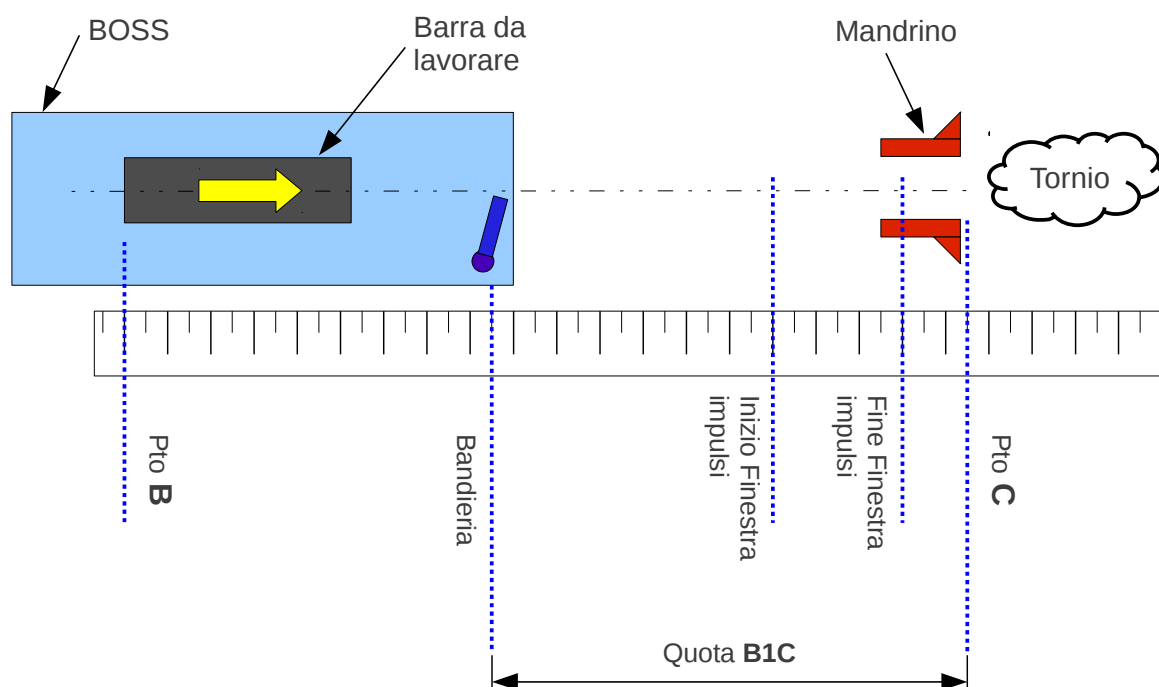


Figura 2.6: Quote importanti per il sistema BOSS. Nello schema si nota che la barra è posizionata sul punto B; quindi si è conclusa la fase di lavoro 6 e siamo nel caso particolare in cui la bandierina non è stata ancora abbattuta.

- **Fase 9: JOG intestatura e intestatura:** Nel caso in cui la bandierina non è stata abbattuta nella fase 6, il macchinario fa avanzare la barra piano piano fino ad colpire il sensore, calcola il punto C e poi può procedere con la fase 9 vera e propria che riguarda l'inserimento della barra nel mandrino del tornio.

Può capitare che la barra presenti delle bave che compromettono l'inserimento del pezzo nel mandrino oppure che la barra abbia una

forma volutamente non cilindrica (es: esagonale). Il BOSS se durante un movimento incontra un impedimento va in allarme e blocca il sistema, se però questo impedimento avviene all'interno della finestra degli impulsi (Fig. 2.6) il problema può essere riconducibile alla mancata inserzione della barra nel mandrino. In questo caso la barra viene fatta indietreggiare e avanzare fino a quando non entra nel mandrino del tornio, il quale ruota leggermente ad ogni passaggio. Quando la barra è correttamente inserita nel mandrino del tornio si riparte dalla **Fase 0**.

Ad ogni fase sono associati dei parametri che servono per adattare il caricatore al tornio del cliente e al tipo di lavorazione che deve effettuare. IEMCA negli anni ha sviluppato svariate funzioni per poter affinare sempre meglio il prodotto con le esigenze del cliente. Nel simulatore sviluppato in questa tesi non verranno implementate tutte le funzioni in quanto si tratterebbe di un banale copia e incolla. A titolo d'esempio, solo per capire il grado di flessibilità raggiunto nel BOSS, se ne elencano le principali funzioni.

Esistono due tipi di avanzamento del pezzo nel tornio, a *posizione*: il pezzo avanza di una distanza prefissata; o a *battuta*: il pezzo avanza fino a un fermo meccanico. Quindi è necessario che esistano dei parametri per la definizione della coppia massima e della velocità onde evitare di rovinare il tornio. Per ridurre le vibrazioni è stata sviluppata una funzione che mantiene sempre il motore in coppia contro il mandrino del tornio. Un'altra funzione sviluppata dalle richieste di alcuni clienti permette di cambiare il tipo di lavorazioni in corsa, ad esempio, quando la barra è troppo corta, piuttosto che eseguire un cambio barra, eseguono un altro tipo di lavorazione più piccola per poter sfruttare meglio il materiale. In altri casi può essere richiesto che il caricatore effettui un anticipo di cambio barra che permette di ridurre i tempi morti del tornio. Questo anticipo può fermarsi prima della fase di preavanzamento onde evitare che la nuova barra urti la vecchia ancora in lavorazione nel tornio oppure se le barre sono corte il preavanzamento finisce alla fase di intestatura, riducendo quasi a zero i tempi morti del tornio. Esistono anche funzioni che descrivono il tipo di espulsione dello spezzone.

In totale ci sono 45 parametri (ognuno con uno o più campi) modificabili dall'utente tramite il pannello operatore.

## **2.2 Aspetti critici e possibili soluzioni**

Come già accennato il BOSS presenta alcuni difetti in flessibilità e modularità del sistema. Il BOSS svolge a pieno le sue funzioni di caricatore di barre nel tornio ma la sua architettura non lo rende facilmente riutilizzabile per altri

progetti. Inoltre questa architettura di sistema richiede un PLC performante che deve gestire sia la parte di motion che di controllo del sistema.

I nuovi protocolli di comunicazione basati su ethernet migliorano ulteriormente la flessibilità (vedi capitoli successivi) e rendono possibile l'esecuzione da remoto (tramite internet) di diverse operazioni che prima invece rendevano necessario un operatore sul posto. Come ad esempio la diagnostica, l'assistenza o l'aggiornamento del software.

Questi nuovi protocolli aprono nuovi orizzonti che fino ad ora erano poco utilizzati nell'automazione di piccoli macchinari come i caricatori di barre. Occorre sempre tener presente che il caricatore è di fatto un accessorio del tornio, il quale potrebbe lavorare anche senza. Quindi per poter essere competitivi nel mercato occorre mantenere prezzi bassi in virtù del fatto che il caricatore è un prodotto non sempre strettamente necessario per il cliente.

Concludendo, si intuisce che più che di criticità è corretto parlare di miglioramenti che si possono implementare sul sistema. Miglioramenti che oggi sono possibili grazie alla maturazione di nuove tecnologie che consentono un aggiornamento del sistema. L'Industrial Ethernet, ad esempio, è presente sul mercato da diverso tempo ma solo adesso sono stati implementati standard che soddisfano le esigenze dei bus di campo su componenti poco costosi.

Il BOSS non richiede urgentemente un aggiornamento, in quanto ad oggi svolge ancora in pieno le funzionalità di un buon caricatore ma è importante per una azienda mantenersi in linea con le nuove tecnologie per non rimanere indietro ed essere superata dai competitor. Soprattutto quando le nuove tecnologie come l'industrial ethernet rivoluzioneranno il modo di lavorare dell'assistenza ai clienti.

## **3 STANDARD DI COMUNICAZIONE NELL'AUTOMAZIONE**

Come già accennato in precedenza, l'azienda IEMCA lavora principalmente con prodotti Siemens, quindi è importante che il prototipo sul quale verrà sviluppato il nuovo macchinario per il caricamento automatico di barre su un tornio sia basato su prodotti Siemens al fine di poter utilizzare il know-how dell'ufficio tecnico e di andare incontro alle esigenze dei clienti IEMCA.

La scelta di limitare la ricerca della componentistica in campo Siemens sembrerebbe contrastare con la necessità dell'azienda di realizzare prodotti al minor prezzo possibile a parità di qualità.

Questo è vero in parte, sarà fondamentale infatti che i componenti scelti soddisfino i principali standard di comunicazione nel campo dell'automazione. In maniera tale che in futuro sia possibile sostituire alcuni moduli Siemens del sistema con altri prodotti da aziende concorrenti con costi minori, a parità di prestazioni. Già adesso il sistema BOSS della IEMCA utilizza un azionamento “Elmo” gestito da un PLC Siemens.

Purtroppo come spesso accade nel campo dell'automazione gli standard proposti sono vaghi, di conseguenza si possono creare diverse soluzioni che rispettano lo standard ma che non sono compatibili tra loro. Vedremo successivamente come ad esempio lo standard IEC 61159 che regola la comunicazione a livello di campi supporta diversi protocolli che non sono compatibili tra loro.

Quindi, il protocollo scelto per la comunicazione tra PLC e il posizionatore, non dovrà solo rispettare gli standard internazionali per la comunicazione a livello di campo, ma all'interno dello standard dovrà utilizzare un protocollo che abbia sviluppo nei prossimi anni. Solo in questo modo sarà possibile sostituire uno o più moduli del sistema senza modificarne l'essenza.

Tutta la scelta dei componenti del sistema deve essere lungimirante al fine di fornire pezzi di ricambio e garantire così al cliente una corretta assistenza negli anni a venire. Ma il protocollo di comunicazione tra il livello di comando e il livello di direzione operativa può fare la differenza tra un macchinario vendibile e un macchinario che diventa obsoleto o difficile da integrare negli impianti della azienda cliente.

### **3.1 Bus di Campo (Fieldbus)**

Per fieldbus si intendono quei protocolli industriali usati in sistemi di controllo che richiedono vincoli temporali stringenti nella comunicazione. Un complesso

sistema produttivo automatizzato, come una linea di montaggio, è sempre costituito da un insieme di controllori organizzato gerarchicamente. Il livello di campo è il livello più basso della piramide. In questo livello troviamo i componenti che realizzano e monitorano effettivamente i singoli movimenti (sensori e attuatori).(Fig. 3.1) I Bus di campo si occupano anche della comunicazione in questo livello e della comunicazione con il livello superiore.

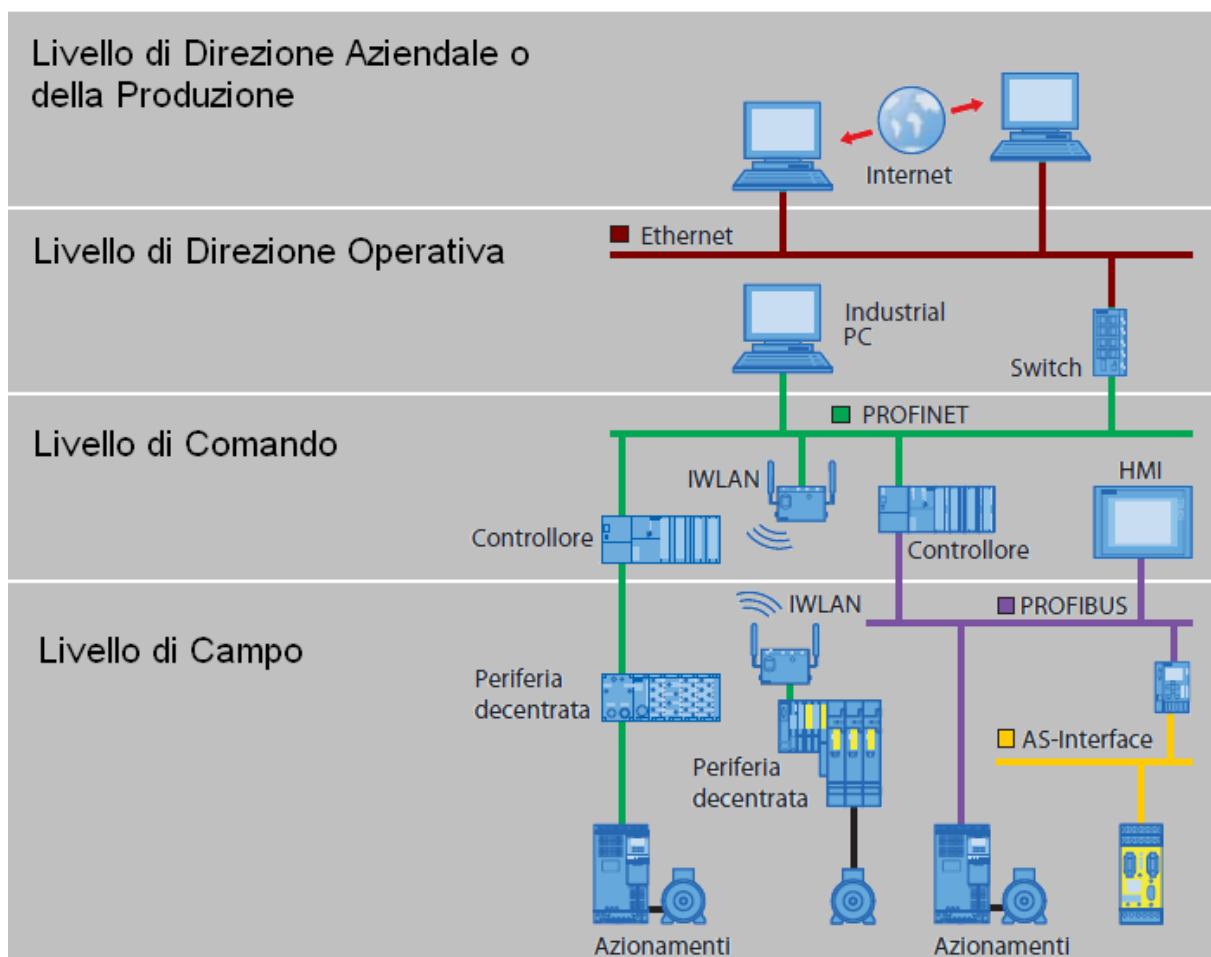


Figura 3.1: Sviluppo di una rete aziendale in tutti i suoi livelli. In questo schema si possono notare tre FieldBus differenti: AS-Interface, ProfiBUS e ProfiNET.

I pacchetti dati che vengono trasmessi su questi bus sono piccoli rispetto alle comunicazioni effettuate a livello di direzione operativa o di direzione aziendale. Nei fieldbus diventano molto più importanti i tempi di ricezione/invio dell'informazione e ogni quanto possono essere aggiornati rispetto alla velocità di trasmissione intesa come capacità di trasferire grosse quantità di dati; in questi casi, si parla di sistemi Real-Time. I sistemi real-time sono quelli in cui la correttezza del sistema non dipende solamente dal risultato logico dell'attività di calcolo ma anche dall'istante in cui il risultato logico è disponibile. Le informazioni devono pervenire entro limiti temporali precisi. In questi sistemi i dati vengono aggiornati con tempi attorno alle decimi di millisecondi, i sistemi

che richiedono aggiornamenti più frequenti (sotto al millisecondo) sono detti IRT (Isochronous Real-time).

Il passaggio della comunicazione da analogico a digitale ha dato enorme sviluppo ai fieldbus, aprendo la possibilità di scambiare dettagliatamente e velocemente informazioni sullo stato della processo produttivo e del ambiente circostante. Inoltre con la comunicazione digitale sia è resa possibile una più facile decentralizzazione dei componenti e una semplificazione del cablaggio.

Alcuni sistemi di comunicazione a Bus di Campo sono relativi a standard proprietari, mentre altri fanno riferimento a standard definiti a livello Europeo (ad esempio gli standard WorldFIP, Profibus e P-Net definiti dal CENELEC con normativa EN50170). In ogni caso, gli standard di comunicazione per Bus di Campo attualmente disponibili sono molti e presentano differenze, a volte, notevoli. I servizi offerti dallo standard WorldFIP sono differenti da quelli offerti, ad esempio, dagli standard P-Net e Profibus. Vi sono alcuni standard, poi, come il Profibus, che offrono più profili di comunicazione, ciascuno caratterizzato da particolari caratteristiche di comunicazione.

### **3.1.1 Safety**

Alcuni protocolli di fieldbus posso essere usati anche in quei sistema che richiedono una procedura di fail-safe. I componenti di failsafe si attivano in caso di guasto ed eseguono una routine per la messa in sicurezza del sistema per evitare danni a cose o persone. Fino a pochi anni fa, i sistemi per rispondere a questi requisiti di sicurezza dovevano essere ridondanti, cioè una parte del sistema era dedicata al funzionamento standard e una era dedicata al fail-safe.

Adesso, grazie al fatto che alcuni controllori hanno CPU apposite che rispondono a i requisiti di messa in sicurezza (fail-safe) delle principali norme internazionali e grazie alla affidabilità di alcuni protocolli di bus di campo è possibile ridurre la ridondanza. Questo permette di ridurre di molto i costi di progettazione e di realizzazione dell'impianto. (Fig. 3.2)

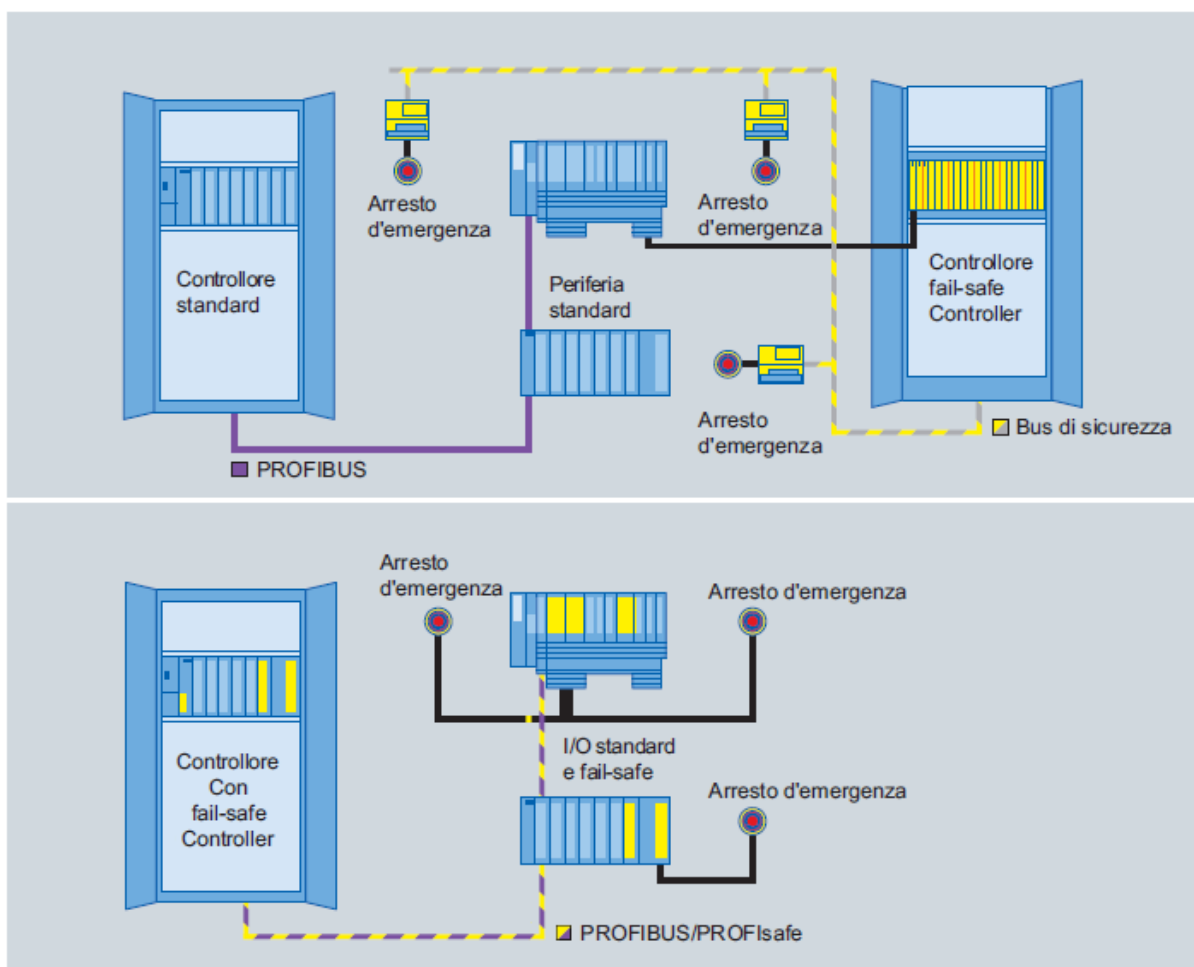


Figura 3.2: **Alto** schema con Bus e Controllore per Fail-Safe ridondante  
**Basso** schema con Bus e Controllore per Fail-Safe integrato

### 3.2 IEC 61158

L'attività di standardizzazione internazionale ha avuto inizio a metà degli anni '80, con il tentativo di regolamentare i protocolli di comunicazione dei bus di campo al fine di limitare la proliferazione di standard proprietari che obbligano i produttori di macchine automatiche ad acquistare tutta la componentistica dal fornitore proprietario del protocollo, con conseguente aumento dei costi. Dopo quasi 15 anni di lavoro, la commissione IEC non ha raggiunto lo scopo di unificare i protocolli di comunicazione a livello di campo in uno unico, ma ha delineato solo una schema di riferimento.

Infatti, la norma IEC 61158 originale identifica otto differenti protocolli per la comunicazione su bus di campo chiamati “Type” che sono:

- Type 1 Foundation Fieldbus H1
- Type 2 ControlNet
- Type 3 PROFIBUS



- Type 4 P-Net
- Type 5 FOUNDATION fieldbus HSE (High Speed Ethernet)
- Type 6 SwiftNet
- Type 7 WorldFIP
- Type 8 Interbus

Ai quali sono stati aggiunti altri nuovi protocolli per le comunicazioni fail-safe (comunicazioni di messa in sicurezza del sistema in caso di guasto) e per le comunicazioni Real Time basate su ethernet, come ad esempio ProfiNET.

Il mancato raggiungimento del obiettivo di unificare i protocolli di comunicazione a livello di campo è imputabile alla pressione esterna esercitata da parte dei grandi produttori che non volevano essere troppo vincolati ad una norma e dal fatto che il livello di campo di un processo produttivo può essere molto variegato e di conseguenza richiede protocolli di comunicazione con vincoli diversi ed eterogenei.

I servizi e i protocolli di comunicazione previsti per il Bus di Campo IEC 61158 sono raggruppati in tre livelli principali, ciascuno dei quali è definito da uno o più documenti. (Fig. 3.3)

- IEC 61158.1 Overview and guidance for the IEC 61158 series
- IEC 61158.2 Physical Layer specification and service definition
- IEC 61158.3 Data Link Service definition
- IEC 61158.4 Data Link Protocol specification
- IEC 61158.5 Application Layer Service definition
- IEC 61158.6 Application Layer Protocol specification
- IEC 61158.7 System & Network Management definition

Il livello più alto è quello Application definito dal documento IEC 61158.5 per quanto riguarda i servizi e dal documento IEC 61158.6 per il protocollo. Sotto di esso vi è il livello Data Link definito dai documenti IEC 61158.3 e IEC 61158.4 per quanto riguarda i servizi e il protocollo rispettivamente. Sotto ancora vi è il livello Physical, definito dal documento IEC 61158.2. Nel documento IEC 61158.1 c'è la parte introduttiva e di guida alla norma. Un livello di System & Network Management supervisiona l'intera pila di comunicazione (sigla IEC 61158.7)

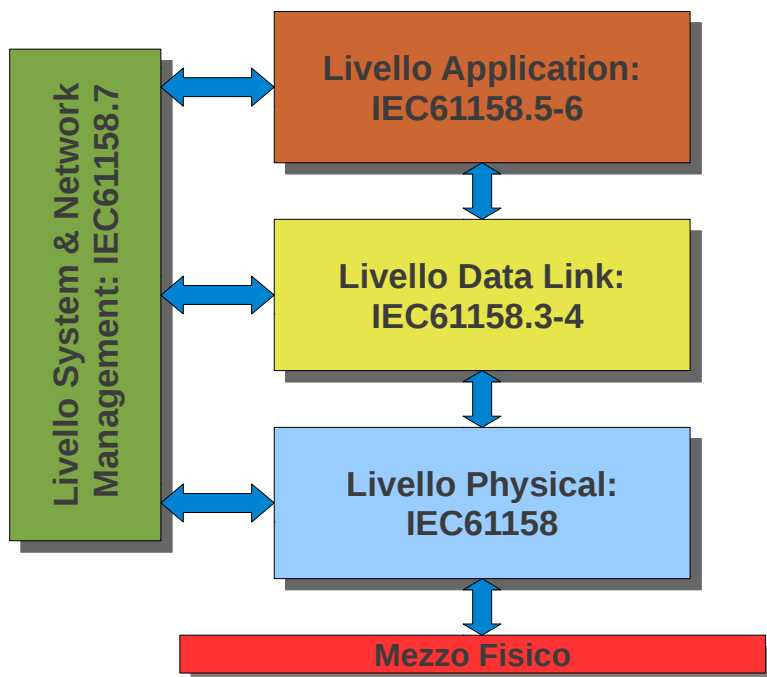


Figura 3.3: EC 61158: Architettura Generale

### 3.2.1 Il Livello Physical: IEC 61158.2

Una qualunque entità di comunicazione si compone di una interfaccia, denominata Data Terminal Equipment (DTE), e da un modulo chiamato Data Communication Equipment (DCE) nel quale sono definite le modalità di trasmissione. Il livello Physical ha una struttura modulare, dove i blocchi funzionali che lo costituiscono appartengono sia al DTE che al DCE.

Il blocco funzionale nel livello Physical relativo al DTE è il DCE Independent Sublayer (DIS). I blocchi che costituiscono il DCE sono, invece, il Medium Dependent Sublayer (MDS) e il Medium Attachment Unit (MAU). (Fig. 3.4)

Il compito del DIS è quello di tradurre i dati ricevuti dal livello Data Link chiamate Physical Interface Data Units (PhIDU) in Physical Service Data Units (PhSDU) utili al MDS e viceversa.

Il MDS, si occupa del preambolo e dei delimitatori alle PhSDU, dello svolgimento delle funzioni di temporizzazione e di sincronizzazione e della codifica Manchester Biphase L del messaggio. Vista la dipendenza del DTE dalle scelte trasmissive, tre tipi di MDS sono attualmente previsti relativamente all'uso di cavo di rame, di radio e di fibra ottica.

Il livello di MAU permette differenti velocità di trasmissioni nell'ambito dello stesso MDS, ossia nello stesso tipo di mezzo di comunicazione. Questo implica che i collegamenti possono avere diverse lunghezze a seconda della velocità di trasmissione (da poche decine di metri a 1900m con cavo i rame e fino a svariati chilometri con la fibra ottica)

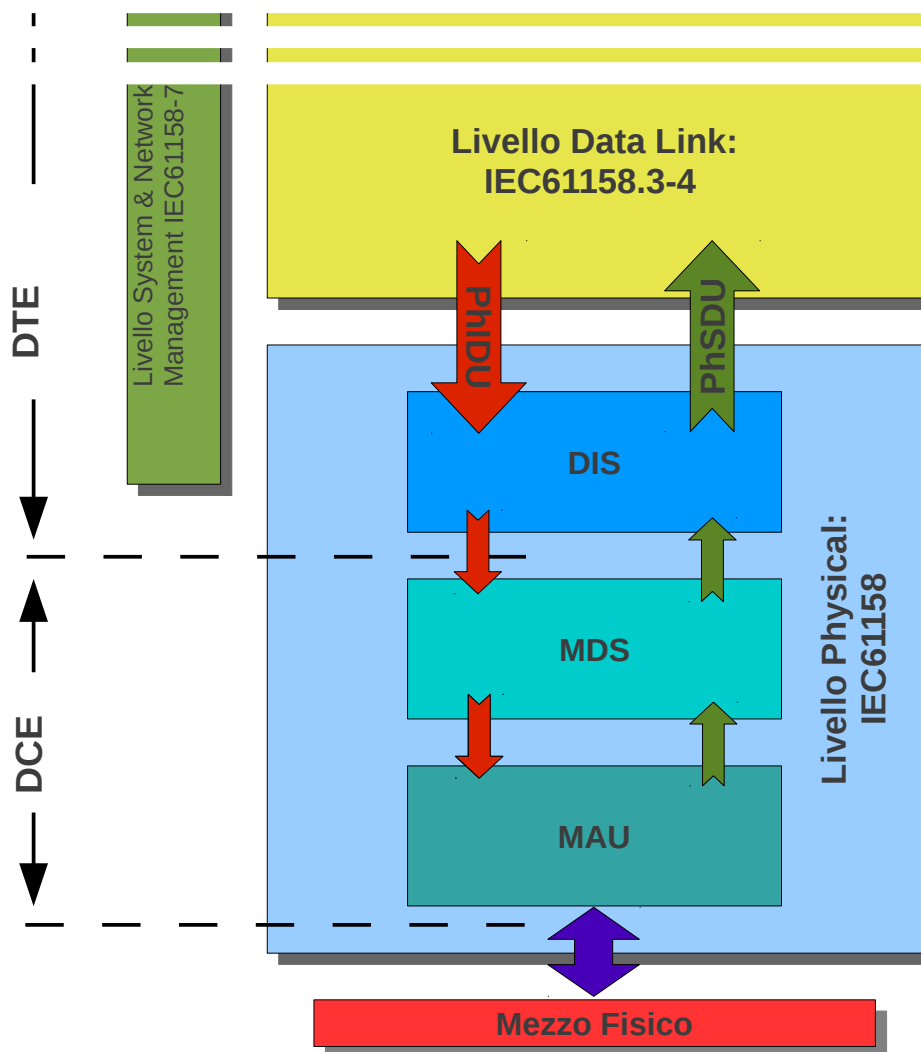


Figura 3.4: Struttura del Livello Physical del Bus di Campo IEC 61158

### 3.2.2 Il Livello Data Link: IEC 61158.3-4

Nell'ambito dell'architettura IEC 61158, il Data Link rappresenta il livello più critico per quanto riguarda la definizione dei servizi e delle procedure. Il suo compito è, infatti, relativo alla gestione real-time delle esigenze trasmissive dei processi del sistema di comunicazione a Bus di Campo.

I processi a livello di campo sono spesso caratterizzati da esigenze temporali assai stringenti e molto spesso differenti tra loro. In generale è possibile distinguere due tipologie di processi tempo-critici: processi periodici (ciclici) e processi asincroni (aciclici).

Un processo periodico è caratterizzato essenzialmente dalla frequenza fissa con cui produce informazioni. I processi periodici sono caratterizzati dall'istante in cui iniziano a generare dati e ogni quanto vengono aggiornati. Tali parametri

sono sempre noti a priori. L'esigenza di comunicazione tipica di un processo periodico è la garanzia che ciascuna informazione prodotta venga trasmessa prima della generazione dell'informazione successiva.

Un processo asincrono è, invece, caratterizzato dall'aleatorietà con cui produce informazioni. A volte, tali informazioni sono vitali per l'intero sistema di controllo (si pensi ad esempio agli allarmi). Analogamente ai processi periodici, i processi asincroni sono caratterizzati dall'istante in cui di generano informazione e e dagli intervalli tra una generazione e l'altra. In tal caso, però, tali parametri non sono mai noti a priori e non sono prevedibili. L'esigenza di comunicazione tipica di un processo asincrono è che ciascuna informazione prodotta venga consegnata al controllore in tempi quanto più bassi possibile, evitando, dunque, eccessivi tempi di attesa prima della trasmissione.

Il livello di Data Link è stato definito dal comitato IEC avendo presenti le due tipologie di processi appena descritti e cercando, dunque, di identificare le giuste strategie di comunicazione in grado di soddisfare le loro esigenze di comunicazione. La soluzione che è stata individuata è composta da due differenti strategie: gestione dell'accesso al mezzo fisico di tipo centralizzata tramite pre-schedulazione e gestione dell'accesso di tipo distribuita tramite uso di token.

La gestione centralizzata è in grado di soddisfare tutte le esigenze trasmissive dei processi periodici. Tale gestione si basa sull'uso di una tabella di schedulazione. In tale tabella vengono determinati a priori tutti gli istanti di trasmissione di ciascun processo periodico, in modo da soddisfarne le esigenze di comunicazione. La gestione centralizzata della comunicazione prevede che istante per istante venga data l'autorizzazione alla trasmissione solo al processo che ne ha diritto, sulla base delle informazioni contenute nella tabella di schedulazione. In tal modo vi è la garanzia che tutte le esigenze trasmissive dei processi schedulati (ossia inclusi nella tabella di schedulazione) vengano sempre rispettate.

La gestione dell'accesso di tipo distribuita tramite l'uso del token è un meccanismo assai diffuso e ben noto. Ad esempio esso viene adottato nello standard IEEE 802.4 (noto come Token Bus). In accordo a tale meccanismo, un particolare messaggio (token) circola tra i nodi del sistema di comunicazione. Esiste un timer che limita il tempo di possesso del token. La gestione dell'accesso al mezzo di comunicazione di tipo distribuito tramite l'uso del token, permette di soddisfare le esigenze trasmissive dei processi asincroni. Per tali processi l'uso della pre-schedulazione comporterebbe uno spreco di larghezza di banda, in quanto potrebbe accadere di cedere il token ad un nodo che non abbia nulla da trasmettere.

D'altra parte il meccanismo del token è in grado di soddisfare le esigenze

temporali di tutti i processi periodici caratterizzati da periodi non troppo stringenti (alcune decine di msec.). In tal caso la circolazione del token in un tempo inferiore al più piccolo periodo di produzione può essere in grado di garantire che tutte le informazioni prodotte vengano trasmesse. È importante notare, però, che l'accesso distribuito basato sulla circolazione del token, non appare affatto soddisfacente nel caso di processi periodici caratterizzati da periodi assai stringenti (alcuni msec). In tal caso l'incertezza che contraddistingue l'istante di arrivo del token, può essere causa della perdita di una o più informazioni.

Il protocollo di livello Data Link prevede la presenza di un particolare nodo di comunicazione, chiamato Link Active Scheduler (LAS), che ha il completo controllo della comunicazione, grazie al quale esso è in grado di realizzare sia l'accesso centralizzato che quello totalmente distribuito.

Il LAS consulta continuamente la sua tabella di schedulazione. Nel caso in cui essa preveda, ad un determinato istante, la trasmissione da parte di un processo periodico pre-schedulato, il LAS ne autorizza la trasmissione, realizzando dunque l'accesso di tipo centralizzato. Negli intervalli della tabella di schedulazione non sono occupati da alcuna trasmissione pre-schedulata, il LAS attiva la circolazione del token, inviandolo al particolare nodo cui esso spetta. Allo scadere del tempo associato al token, o quando il nodo non ha più nulla da trasmettere, il token viene ripreso dal LAS, che lo invia al successivo nodo, in accordo ad una lista di circolazione di token. In questo modo il LAS realizza l'accesso di tipo distribuito. Sebbene tale accesso sia effettuato in modo piuttosto originale, esso realizza a tutti gli effetti il meccanismo di circolazione del token.

### **3.2.3 Il Livello Application: IEC 61158.5-6**

Il livello Application del Bus di Campo IEC definisce un Application Process (AP) come un insieme di informazioni e di risorse di processamento (sia hardware che software), utilizzate per supportare un'attività di processamento di informazioni di tipo distribuita e tempo critica.

In generale, un AP può essere rappresentato in termini di Application Process Objects (APOs), che costituiscono una rappresentazione astratta di uno specifico aspetto di un AP, ossia di un insieme specifico di attività di processamento o di informazioni. Poiché gli APO non sono dotati di capacità di comunicazione, essi vengono supportati dalle Application Entities (AEs), che forniscono un determinato insieme di servizi di comunicazione. I servizi di comunicazione forniti dall'AE sono raggruppati in Application Service Elements (ASEs). Un ASE è dunque un insieme di servizi atti a convogliare richieste e risposte verso e da APO remoti.

Vista la varietà di risorse presenti in un AP, esistono differenti classi di APO, e

per ciascuna di esse viene definito un particolare ASE. Alcuni degli ASE esistenti nel Bus di Campo IEC sono:

- *Variable ASE*: relativo alla lettura/scrittura di variabili di processo
- *Event ASE*: relativo alla gestione della comunicazione di eventi (es: Allarmi)
- *Load Region ASE*: relativo al download/upload di programmi o file
- *Function Invocation ASE*: relativo all'esecuzione o alla terminazione da remoto di funzioni o procedure

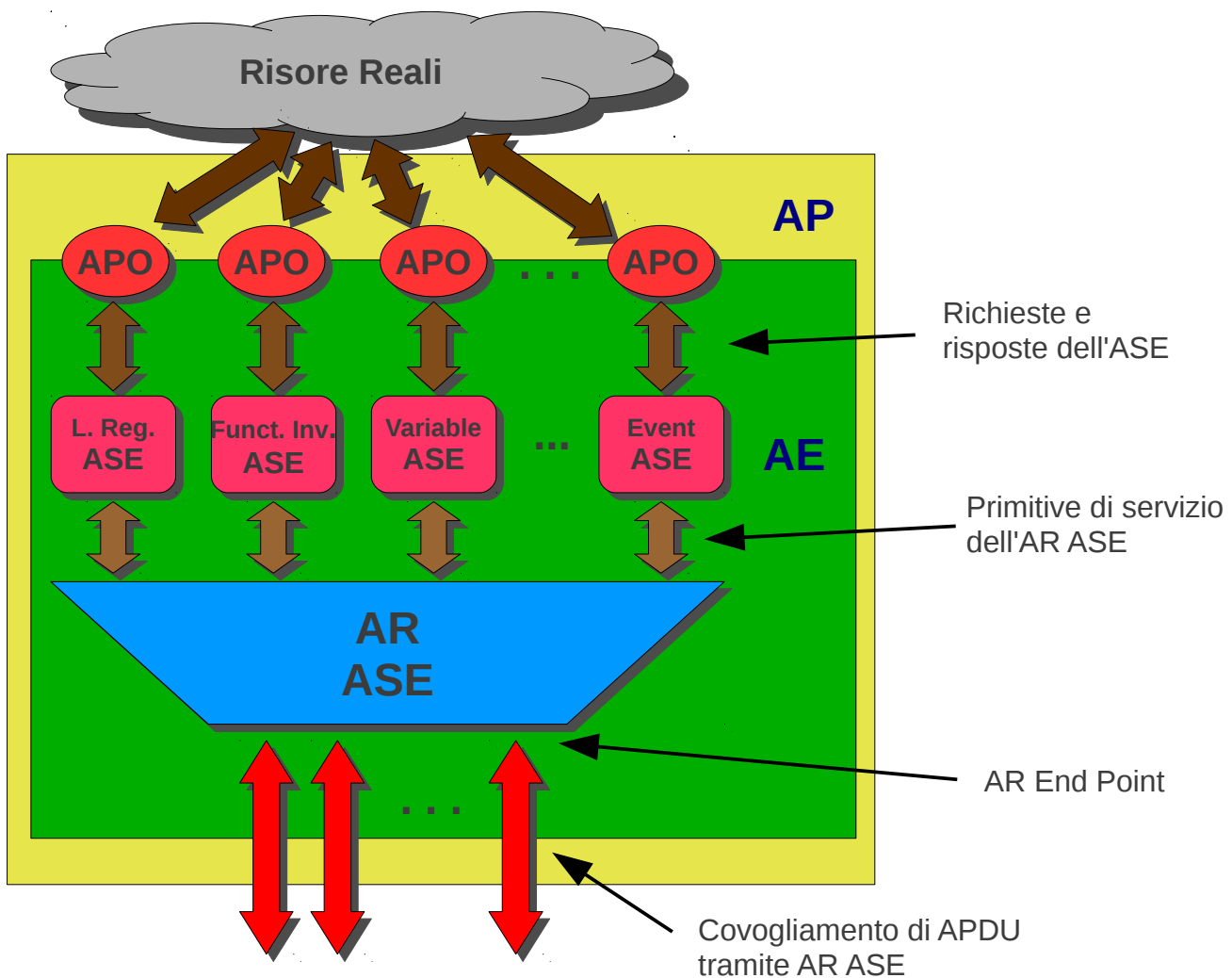


Figura 3.5: Interazione tra i vari tipo di ASE e l' AR-ASE

Le informazioni vengono scambiate tra ASE appartenenti allo stesso tipo, attraverso messaggi particolari detti Application Protocol Data Units (APDUs). Tali messaggi vengono convogliati attraverso connessioni logiche tra due o più ASE chiamate Application Relationship (AR). I servizi necessari alla gestione delle connessioni logiche AR vengono forniti da un particolare ASE, detto

appunto AR ASE. Questo ASE, dunque, comprende servizi per stabilire e mantenere una Application Relationship, e per convogliare APDU tra differenti AP. (Fig. 3.5)

Un punto chiave nella definizione di uno standard di livello Application sono i modelli di comunicazione tra gli AP. L'obiettivo che si è voluto perseguire nell'ambito del comitato IEC è stato quello di offrire una gamma di modelli di comunicazione, che risultasse quanto più ampia possibile, in modo da soddisfare le più comuni esigenze di comunicazione presenti nelle applicazioni reali nell'area del controllo di processo e dell'industria manifatturiera. Su tale punto risiede la differenza sostanziale con altri standard di comunicazione per Bus di Campo, che, in generale, offrono a livello Application solo un limitato set di modelli di comunicazione. Le relazioni tra AP previste al livello Application dal comitato congiunto IEC sono essenzialmente di due tipi: *publisher/subscriber* e *client/server*.

Il modello *publisher/subscriber* permette lo scambio informativo unidirezionale in cui una entità (il *publisher*) invia una o più informazioni a una o più entità (*subscriber*). Nell'ambito della comunicazione *publisher/subscriber*, sono state previste due differenti modalità trasmissive, quelle di tipo push (*publisher* invia in modo autonomo informazioni ai *subscriber*) e pull (il *publisher* non ha la capacità di iniziare una transazione trasmette solo quando obbligato da una terza entità, chiamata)

Il modello di comunicazione *client/server* è piuttosto conosciuto e permette uno scambio di informazioni bidirezionale tra due entità, il *client* e il *server*. I servizi di comunicazione supportati per il modello *client/server* possono essere sia confermati che non confermati.

### **3.2.4 Il Livello di System & Network Management: IEC 61158.7**

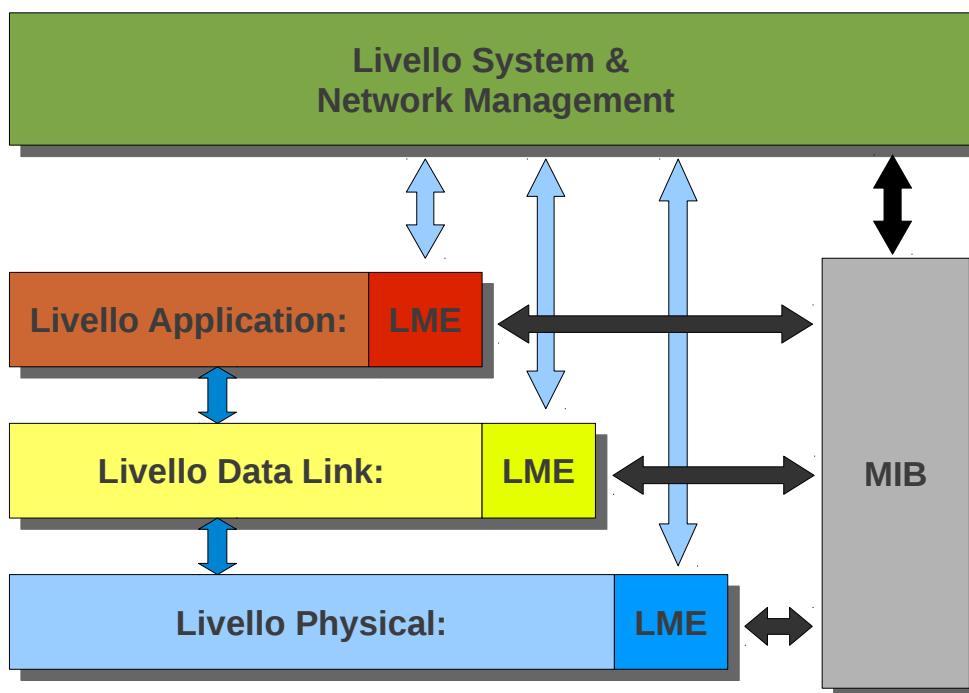
Visto che le prestazioni delle applicazioni presenti in un Bus di Campo sono strettamente correlate alle prestazioni della rete di comunicazione, nel documento IEC 61158.7, che definisce il livello di System & Network Management, le funzioni tipiche di System Management sono più strettamente legate a quelle di Network Management rispetto ad una comune rete OSI.

Gli obiettivi che hanno animato la definizione del livello di System & Network Management sono:

- La gestione e il monitoraggio dell'intero sistema di comunicazione allo scopo di individuare l'aggiunta o la rimozione di un dispositivo e per rendere possibile l'identificazione di tali dispositivi.
- La gestione delle specifiche capacità di un dispositivo, cioè ad esempio la creazione di un'istanza di un oggetto applicativo, la creazione o l'eliminazione

di una connessione logica, il caricamento o lo scaricamento della configurazione di un dispositivo.

- La gestione dei guasti, ed in particolare la gestione del riconoscimento di un guasto ed il suo recupero.
- La gestione delle prestazioni della comunicazione, allo scopo di soddisfare tutte le richieste tempo-critiche.



*Figura 3.6: Architettura adottata per il System & Network Management*

In tale architettura sono distinguibili le seguenti entità (Fig 3.6):

- LME. Ciascuno dei tre livelli è dotato di una entità di supervisione (Link Management Entity-LME). Tale entità ha lo scopo di notificare al livello System & Network Management qualunque anomalia occorsa durante le normali procedure di comunicazione. Inoltre, in alcuni casi, ciascun LME può intraprendere azioni di recupero di anomalie, come ad esempio recupero di guasti. Ciascuna azione di recupero viene notificata al System & Network Management.
- MIB. La Management-Information-Base (MIB) è una particolare banca dati, in cui vengono memorizzate informazioni relative al nodo di comunicazione (performance, presenza di guasti, etc.) che vengono passate dai LME di ciascun livello. Su tali informazioni si basano le azioni che il System & Network Management intraprende.



### 3.3 Industrial Ethernet

Per rendere sempre più efficiente e flessibile la loro organizzazione interna, sempre più aziende produttrici stanno rapidamente migrando verso la tecnologia industrial ethernet che connette le reti di livello di campo e dell'impianto con con il sistema aziendale. L'industrial ethernet è molto flessibile e riesce a unire in un unico standard requisiti di Real Time che i requisiti tipici richiesti dalla classica rete presente negli uffici di produzione. Riuscire a connettere l'impianto con il sistema aziendale migliora notevolmente la visibilità delle informazioni, la qualità e le prestazioni dell'intero processo produttivo, riducendo i costi di progettazione e sviluppo.

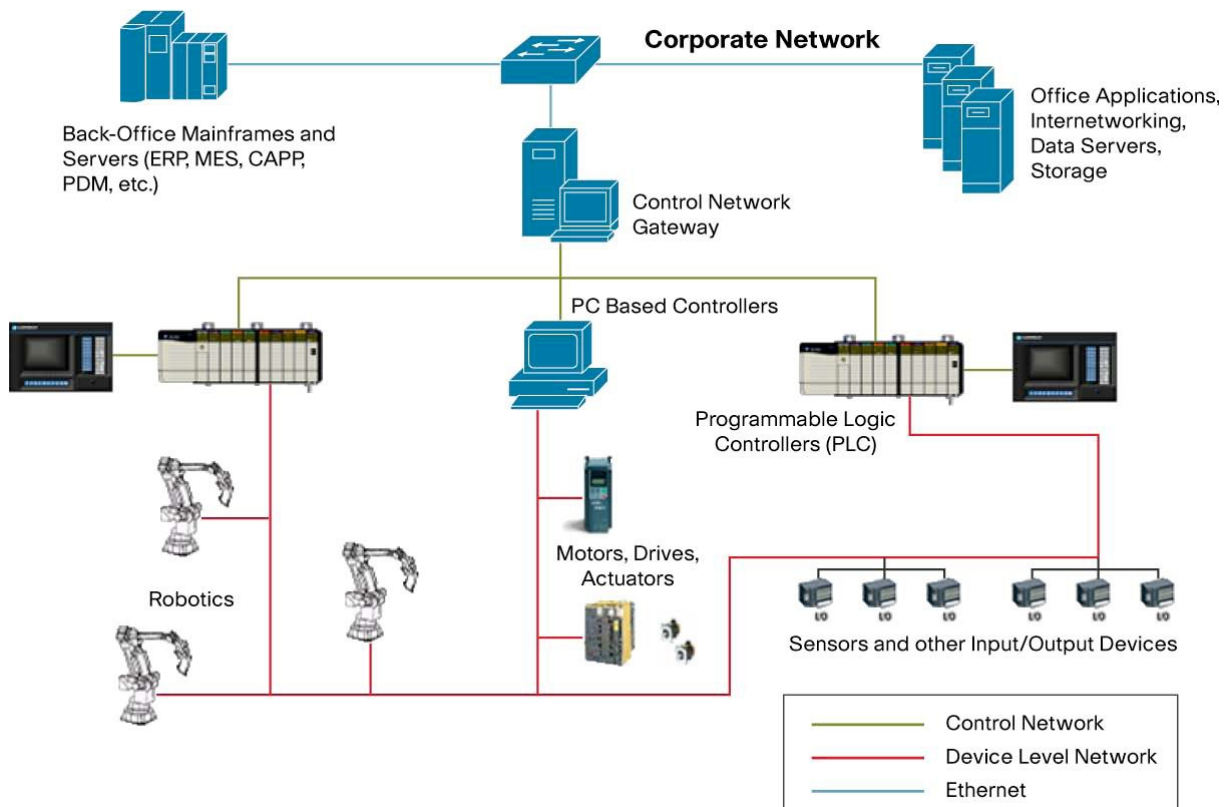


Figura 3.7: Schema di una rete classica nell'automazione

Ulteriori vantaggi risiedono nel fatto che l'industrial ethernet è uno standard aperto e i prodotti compatibili con esso sono sviluppati da diverse aziende. Questo consente un notevole risparmio rispetto a i più costosi e chiusi standard proprietari ottimizzati per la comunicazione su bus di campo. Oltretutto è importante ricordare come lo standard ethernet ha tuttora un continuo sviluppo (l'ethernet a 1GB è ormai una realtà consolidata in molte aziende) e ha sempre mantenuto un compatibilità i protocolli ethernet più vecchi.

Oggi la maggior parte delle aziende mantengono separata la rete del livello di comando con la rete del livello di gestione operativa, negli anni queste reti si sono sviluppate separatamente focalizzandosi ognuna su i differenti flussi informativi e di controllo richiesti.



Figura 3.8: Pila ISO/OSI

In alto troviamo la rete aziendale che deve supportare le classiche le applicazioni aziendali. Questa rete è quasi sempre connessa al mondo esterno tramite internet e ad altri siti di produzione dell'azienda. Tipicamente è basata su ethernet e protocolli IP. Sotto di essa è presente la rete di controllo che unisce i controllori (PLC, PC-embedded ecc.) dell'automazione e gli HMI (Human Machine Interface). Più in basso troviamo la rete di livello di campo che connette i

sensori e gli attuatori. Questa rete è spesso realizzata con diversi standard bus di campo, come: deviceNET, ProfiBUS, AS-Interface ecc. Ognuno con diverse caratteristiche che lo rendono più o meno adatto a diversi ambiti di lavoro. Come conseguenza sia ha un moltiplicarsi di diverse reti anche nello stesso macchinario. Nell'industrial ethernet le informazioni specifiche del bus di campo usate dai I/O device sono incapsulate nei classici frame ethernet.

Le specifiche dell'ethernet si riferiscono solo ai primi due layer della pila ISO/OSI. Mentre le soluzioni che implementano l'industrial ethernet comprendono anche i layer 3 e 4 usando l'indirizzamento IP e i protocolli TCP (Transmission Control Protocol) o UDP (User Datagram Protocol).

Le applicazioni TCP stabiliscono la connessione tra dispositivi sulla quale vengono spediti i pacchetti. La connessione è mantenuta aperta anche dopo la fine di trasmissione del pacchetto. A causa del elevato overhead il TCP non è l'ideale per le applicazioni real time.

UDP è considerato un protocollo “stateless”, non richiede infatti una connessione tra i dispositivi per spedire pacchetti, in questo modo si ha un bassissimo overhead. Quindi è un protocollo spesso utilizzato per comunicazioni real time, es: trasmissione voce, e per comunicazioni broadcast e multicast. Di contro però, non si ha garanzia sulla corretta ricezione dei pacchetti e sul loro ordine di arrivo.

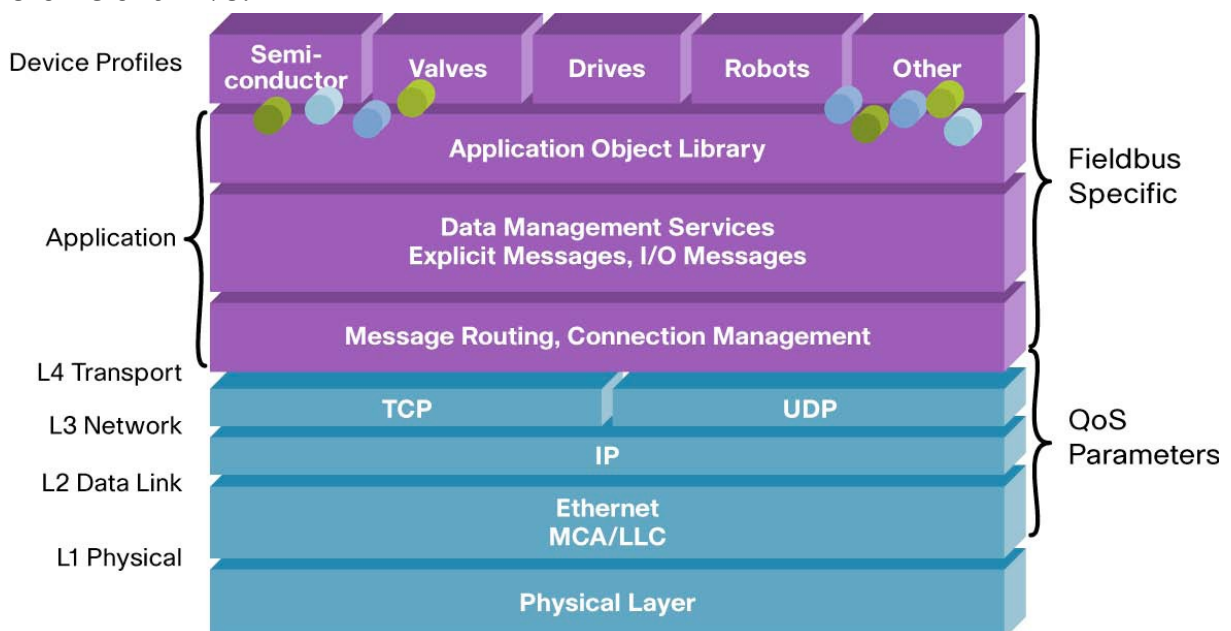


Figura 3.9: Industrial ethernet applicata all'automazione

Uno dei vantaggi dell'industrial ethernet è che, in teoria, l'organizzazione della rete e i dispositivi rimangono quelli usati nel ethernet tradizionale. Nella pratica l'industrial ethernet e l'ethernet tradizionale devono rispondere a requisiti molto diversi tra loro. Nell'ambito industriale sono richiesti: tempi di comunicazione

corti e predicibili (per applicazioni real-time), flessibilità sul numero di nodi connessi e una segmentazione più elevata rispetto al ethernet tradizionale. Inoltre i dispositivi per l'industrial ethernet dovranno lavorare in condizioni ambientali più critiche, dovranno reggere ampi range di temperature e ha possibili urti o a vibrazioni continue.

Nelle comunicazione nell'ambito dell'automazione, l'80% del traffico sulla rete è locale e tipicamente si basa su multicast. Nelle comunicazioni nel mondo dell'information technology avviene il contrario. Il traffico è tipicamente di tipo unicast e avviene tra host remoti. Quindi per implementare una buona rete su industrial ethernet è necessario tenere conto di queste differenze.

Nella rete tradizionale ad ogni segmento ethernet è connesso solo con un nodo e i vari segmenti sono collegati tra loro tramite switch. Gli switch lavorano sul layer 2 del modello OSI utilizzando il MAC address, rispetto ad altri componenti di routing hanno il vantaggi di avere:

- *Prestazione deterministiche e prevedibili.* Questo assicura che i pacchetti spediti vengano ricevuti in tempi prestabiliti assicurando ottime prestazioni per gli ambiti real-time.
- *Bassi tempi di latenza.* Il tempo di latenza riferito a quando il pacchetto viene preso dalla rete a quando viene reindirizzato sul segmento corretto è molto basso. Nelle applicazioni industriali real-time sono tollerati tempi di latenza attorno a i 10ms. Se consideriamo un pacchetto tipico, in quest'ambito, di 500 Byte, uno switch ha un tempo di latenza medio attorno ai 30 microsecondi, che è 100 volte più veloce di quanto richiesto.
- *Standardizzazione.* Essendo l'ethernet uno standard affermato e ampiamente utilizzato, i componenti necessari al suo funzionamento sono prodotti da più aziende e facile da reperire.

In una rete industriale molto ampia è meglio utilizzare dei managed switch. I managed switch sono switch programmabili che possono implementare varie funzionalità al fine di migliorare la qualità del servizio (QoS, Quality of Service). Alcune delle funzionalità principali possono essere:

- *Priorizzare il traffico.* Implementando un semplice parametro di QoS è possibile organizzare il traffico specificando quali pacchetti sono prioritari rispetto ad altri. In questo modo, in caso di congestione dovuto ad un alto volume di traffico i pacchetti prioritari (es. Allarmi) non vengono persi e raggiungono nei tempi prestabiliti il nodo ricevitore.
- *Broadcast e multicast.* Nelle applicazioni industriali il broadcast e il multicast è ampiamente utilizzato. Uno switch intelligente può configurare

dinamicamente le sue interfacce in maniera tale da effettuare il forward solo alle porte interessate da questo tipo di pacchetto.

- *Network analyzer.* Molti managed switch hanno al loro interno un analizzatore del traffico.
- *Security.* Un managed switch può essere configurato per ridurre o evitare i più comuni tipi di attacco (intenzionale o non intenzionale) come il broadcast o multicast storm. Questo tipo malfunzionamento si verifica quando un device produce un elevato numero di messaggi broadcast che invadono la rete, il managed switch in questo caso può isolare il device evitando che questo blocchi l'intera rete.
- *Diagnostica.* Uno degli aspetti critici quando si deve risolvere un problema su un grande impianto è riuscire a reperire le giuste informazioni. Un managed switch può venire incontro a queste esigenze tenendo memorizzate al suo interno le informazioni inerenti al traffico transitato fino al momento del guasto.

### 3.3.1 Sicurezza di rete

L'ethernet tradizionale, offre buone prestazioni nel campo dell'automazione ma inferiori rispetto a protocolli fieldbus nati apposta per questi ambiti di lavoro. D'altra parte l'ethernet rispetto a molti fieldbus offre un ottimo livello di sicurezza intrinseca nello standard adatto per mantenere la disponibilità (sulle operazioni di automazione e di controllo), l'integrità (dei dati e del sistema da attacchi accidentali e intenzionali) e la riservatezza del sistema (bloccando l'accesso a utenti non autorizzati). Tutto questo è facilmente realizzabile tramite VPN, firewall, controllo di accesso e autenticazione ecc.

Questi aspetti sono molto importanti, il fatto che si possano unire tutti i livelli della piramide dell'automazione con un'unica rete vuole dire che ci sono un ampio numero di utenti e quindi sono fondamentali politiche di accesso per evitare che persone non competenti possano creare danni all'impianto.

### 3.3.2 Affidabilità della rete

Su una rete per applicazioni real-time l'affidabilità è fondamentale. I dati devono raggiungere la destinazione con tempi rapidi e deterministici. L'affidabilità deve essere implementata su tutti i livelli della pila ISO/OSI, anche se solo i primi tre livelli riguardano fisicamente la rete. L'affidabilità dal quarto livello in su è realizzata da applicazione e protocolli software.

- **Physical layer.** Si possono applicare diverse tecniche per migliorare l'affidabilità del primo livello della pila ISO/OSI. Normalmente si progetta la rete usando ridondanza sui devices che la compongono. Si possono

implementare più moduli di alimentazione eventualmente affiancati da un UPS, in aggiunta, si possono utilizzare componenti che supportano l'upgrade e la manutenzione senza che debbano essere scollegati dalla rete.

- **Data link layer.** Una tecnica può essere quella di aggiungere più segmenti alla rete di quanti ne siano realmente necessari. Le reti tipicamente possono essere realizzate con tipologia ad anello o una a stella. Le due topologie presentano prestazioni, complessità, costi di manutenzione e realizzazione diversi.

Nella topologia a stella i componenti sono organizzati gerarchicamente e ogni segmento che li collega è ridondante. La topologia ad anello ogni componente è collegato con altri due in questo maniera in caso di guasto su un segmento la comunicazione può comunque avvenire girando in senso opposto lungo l'anello. Questa topologia è più semplice da cablare ma non è priva di compromessi sulla prestazioni della rete.

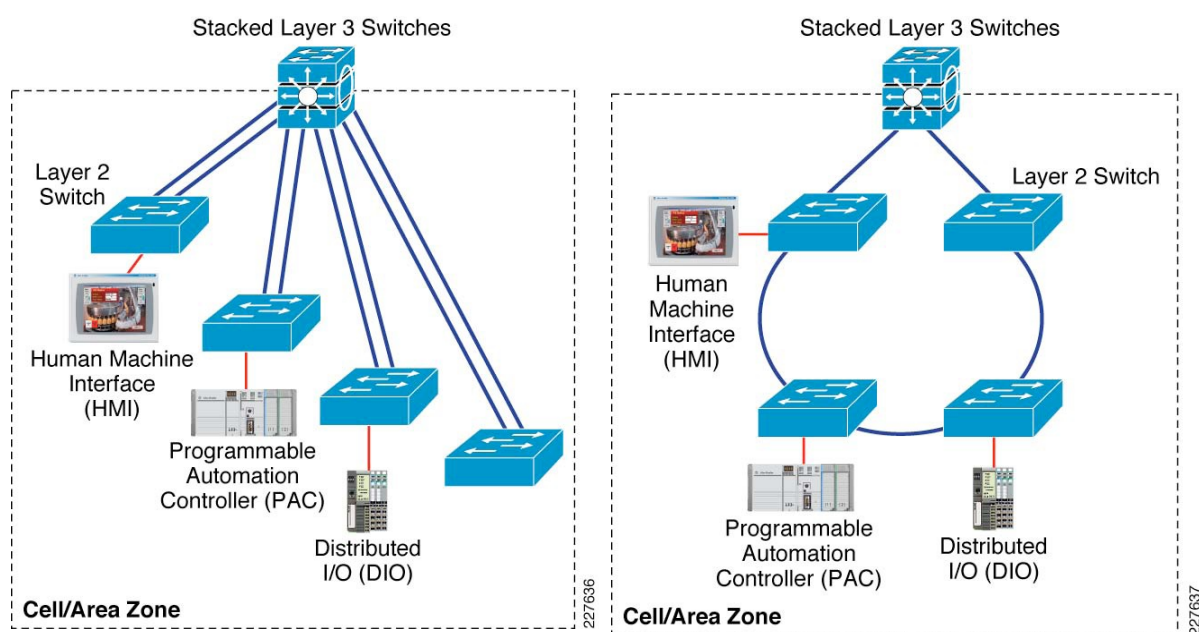


Figura 3.10: Esempio di rete a Stella e ad Anello

La topologia a Stella, d'altro canto è più complessa e costosa ma consente di resistere a multipli collegamenti guasti, minimizza il tempo di interruzione del servizio in caso di guasto, il basso numero di switch rende minore e più deterministico il tempo di percorrenza dei dati.

Se si usano reti con ridondanza di collegamenti occorrono politiche di instradamento che evitano i loop nella rete. Altrimenti si creano casi di broadcast storm.

- **Network layer** La maggior parte del traffico in una rete di comunicazione nell'ambito dell'automazione è locale. Quindi non raggiunge il terzo livello

della pila ISO/OSI. Questo livello è interessato in quei casi in cui ci sono ad esempio delle VPN dove è richiesto un instradamento a livello di indirizzo IP. Esistono diversi standard proprietari che occupano della affidabilità di routing.

### 3.3.3 Traffico Real-Time con IGMP Snooping

Le applicazioni real-time richiedono una precisa sincronizzazione tra i processi, la rete deve essere ottimizzata per supportare un traffico consistente per soddisfare i requisiti delle applicazioni real time. Per poter raggiungere queste prestazioni è necessario implementare sulla rete tecniche di prioritizzazione del traffico e di segmentazione della rete. In maniera tale che i pacchetti con informazioni critiche vengano ricevuti senza troppi ritardi. In ambiente industriale il traffico multicast è il prevalente e deve essere maneggiato l'IGMP (Internet Group Management Protocol), con questo protocollo il traffico multicast viene gestito con tecniche di “publish and subscribed”. Gli switch che implementano l'IGMP processano (“snoop”) il traffico multicast in transito quindi lo reindirizzano solo a chi ne ha fatto richiesta (“subscribed”).

Questa tecnica è molto importante in una industrial ethernet in quanto limita il transito del traffico multicast in aree della rete che non è richiesto. Inoltre, in questo modo i devices collegati alla rete non devono processare frame in arrivo che in realtà non gli interessano. Una industrial ethernet che non implementa tecniche di IGMP rischia di essere congestionata rapidamente dal traffico di broadcast e multicast.

Il controllo e la gestione del IGMP è affidata agli switch della rete, i quali memorizzano al loro interno gli host “*subscribed*” che desiderano ricevere determinate informazioni multicast. Un host può entrare o lasciare la tabella IGMP informando lo switch con un “*IGMP join*” o “*IGMP leave*”.

### 3.3.4 Quality of Service

Una rete industrial ethernet deve poter supportare tipi di informazioni molto diversi tra loro, che vanno dall'invio di dati di routine a dati di controllo critici (traffico I/O, allarmi ecc) o anche traffico audio-video. E' necessario che la rete sappia distinguere la priorità dei diversi tipi di traffico. Implementando una prioritizzazione delle informazioni è possibile realizzare una buona rete real-time con bassi tempi di latenza e minima perdita di pacchetti in caso di congestionamento. La capacità di condividere la stessa rete con applicazioni molto diverse tra loro è il fattore chiave di differenza tra l'industrial ethernet e gli altri protocolli per la comunicazione a livello di comando.

Esistono diverse tecniche per implementare il QoS, tutte però si svolgono su tre passi:

- Individuare tutte le differenti tipologia di traffico esistenti sulla rete, classificarle e inserire dei tag per poterle identificare successivamente. La classificazione deve tenere conto della capacità dei componenti collegati alla rete, dalla sua infrastruttura.
- Implementare avanzate tecniche di bufferizzazione per evitare, in caso di congestione della rete, la perdita di pacchetti con alta priorità.
- Utilizzare tecniche di scheduling per trasmettere velocemente i pacchetti prioritari presenti in coda.

Un altro aspetto, molto importante, da tenere in considerazione è che la stessa informazione ha priorità diversa a seconda del segmento di rete in cui si trova. E' facile intuire che in un ufficio il traffico voce avrà una priorità maggiore rispetto al traffico generato da un devices di I/O dell'impianto. Mentre nell'impianto per mantenere i requisiti sulle tempistiche il traffico deve avvenire il contrario. Per implementare una buona QoS è necessario che ci sia una segmentazione della rete, solo in questo modo è possibile differenziare le priorità del traffico.

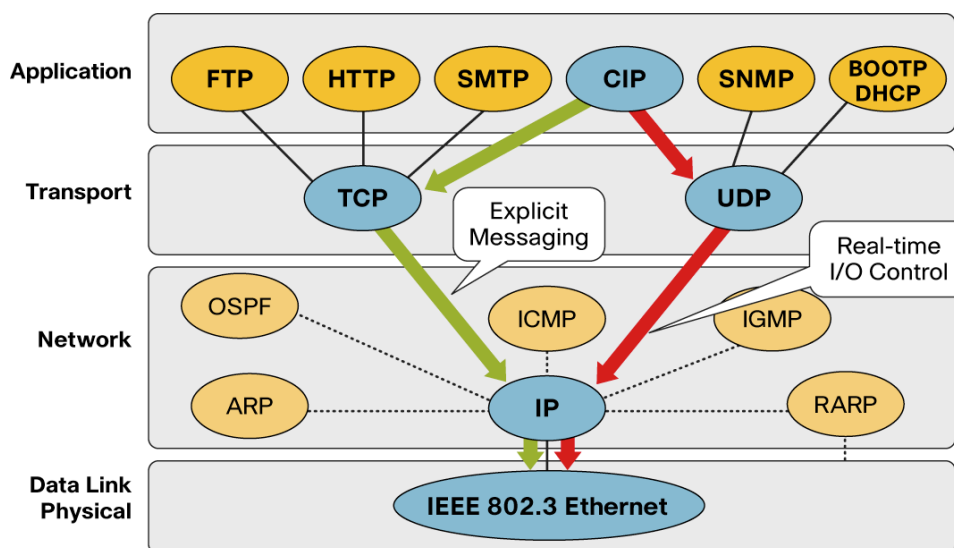


Figura 3.11: QoS per applicazioni su Industrial Ethernet

Le applicazioni su industrial ethernet, traffico realtime degli I/O devono condividere le stesse risorse con il traffico dati della rete tradizionale (FTP, HTTP, SMTP...). Un semplice metodo per realizzare una efficiente QoS è quello di prioritizzare il traffico UDP che è il più usato per traffico realtime di I/O e per il traffico voce. In questo modo si possono realizzare i benefici di condivisione delle risorse mantenendo un buon QoS per le applicazioni Real-Time.



## **4 BUS DI CAMPO IMPLEMENTABILI SU CONTROLLORI SIEMENS**

In Siemens i tre bus di campo principalmente supportati per il collegamento tra i controllori e le apparecchiature di campo sono: As-Interface, ProfiBUS e ProfiNET. Il controllore è collegabile a questi bus tramite interfaccia integrata nella CPU o attraverso speciali unità di comunicazione, inoltre esistono appositi moduli esterni per collegare protocolli bus diversi tra loro. I protocolli MPI, ProfiBUS e ProfiNET sono utilizzati anche per poter comunicare tra più controllori dell'impianto e tra controllori e HMI (come ad esempio i pannelli operatore).

### **4.1 Multi Point Interface**

Multi Point Interface (MPI) è una interfaccia di comunicazione proprietaria dei PLC (Programmable Logic Controller) SIMATIC S7 della Siemens. Viene usata per mettere in comunicazione elementi della famiglia SIMATIC, tra cui postazioni di programmazione (PG o PC), interfacce operatore e dispositivi di ingresso e uscita di segnali. Questa tecnologia è stata il precursore del protocollo Profibus, è basata sullo standard EIA-485 (ex RS-485) e supporta una velocità di trasmissione fino a 187.5 K Baud. Essendo uno standard proprietario è difficile reperire altre informazioni più specifiche, in quanto restano nascoste dietro le API fornite da Siemens.

Attualmente lo standard non viene quasi più usato se non per eseguire semplici collegamenti, ad esempio tra PLC e HMI.

### **4.2 AS-Interface**

AS-Interface (Actuator Sensor Interface, AS-i) è un protocollo di comunicazione aperto progettato per connettere semplice device di I/O (attuatori, sensori, encoder...). Utilizza un conduttore a due cavi usato sia per la comunicazione che per l'alimentazione. Tipicamente nella piramide dell'automazione i bus AS-Interface occupano solo il livello più basso (Livello di Campo) e sono interfacciati ai livelli superiori tramite ProfiBUS o DeviceNET o Industrial Ethernet. In quanto offre buoni vantaggi nella comunicazione tra I/O device ma non è adatto per la comunicazione a livello di controllo.

Questo standard risponde ai requisiti di sicurezza citati nel paragrafo 3.1.1 a pagina 19. La tecnologia chiamata “Safety as Work” implementata su AS-Interface fa sì che la componentistica per la sicurezza e i I/O device possano essere connessi assieme sullo stesso bus supportando un grado di sicurezza fino a

SIL 3 (Safety Integrity Level 3).

Essendo uno standard aperto, esistono molti produttori di componentistica che comunica su AS-Interface. Tutti i prodotti che utilizzano questo protocollo devono essere certificati da enti esterni per assicurare che il dispositivo rispetti le specifiche del protocollo.

Le rete realizzata con AS-interface richiedere un Master (chiamato *Scanners* o *Gateways*) da dove parte il bus sul quale si possono collegare gli Slave (chiamati anche *Modules*). Il Master può funzionare anche come gateways con i bus di comunicazione del livello di controllo. La comunicazione segue la procedura classica del metodo Master-Slave, nella quale il master inizia lo scambio di dati con lo slave inviando un frame di 14 bit (di cui 5bit sono di indirizzamento) la risposta dello slave è di 7 bit (di cui solo 4 bit contengono l'informazione). Il master resta in attesa della risposta solo per un tempo massimo prestabilito al fine di soddisfare i requisiti di rete deterministica per applicazioni Real-Time.

Oltre ai componenti master e slave la rete AS-Interface richiede moduli per l'alimentazione.

AS-Interface nasce alla fine degli anni 80 e le sue specifiche vengono pubblicate solo nel 1994. Ha subito diversi aggiornamenti nel corso degli anni. L'ultima versione (la ver 3.0 sviluppata nel 2007) supporta:

- Il doppio indirizzamento (indirizzo A e indirizzo B)
- Slave con indirizzamento a 8 bit di ingresso e 8 bit di uscita
- Slave con il vecchio indirizzamento a 4bit di ingresso e 4 bit di uscita.
- Canali analogici a 8, 12 o 16bit.
- Comunicazione seriale full-duplex

Per bus superiori a 100 metri di lunghezza sono necessari ripetitori di segnale che isolino galvanicamente il segmento ogni 100 metri e ripetano il segnale.; ovviamente è necessario un modulo di alimentazione per ogni segmento dato che sono isolati uno dall'altro.

Per rispettare i tempi di latenza richiesti dalla maggior parte delle applicazioni real-time la rete in ogni caso non può superare i 300 metri (Fig. 4.1). Per reti più ampie si può implementare una topologia a stella.

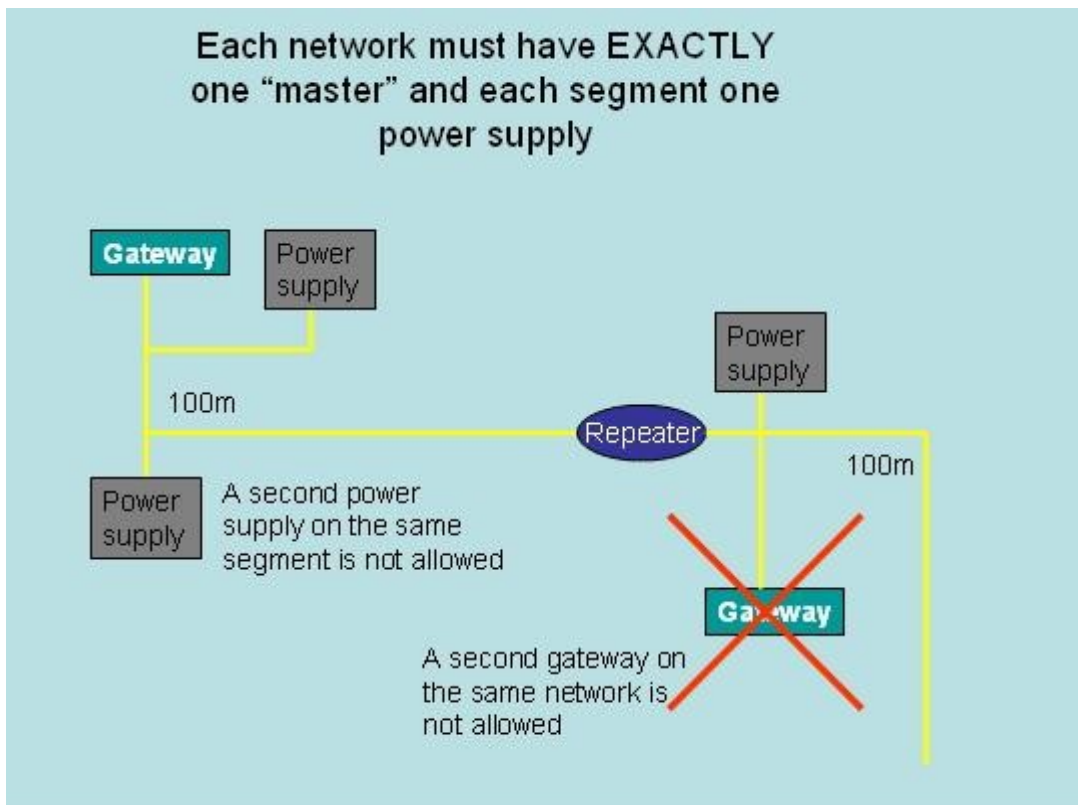


Figura 4.1: Schema di un BUS AS-Interface

Da Notare che i segmenti non superano i 100metri di lunghezza

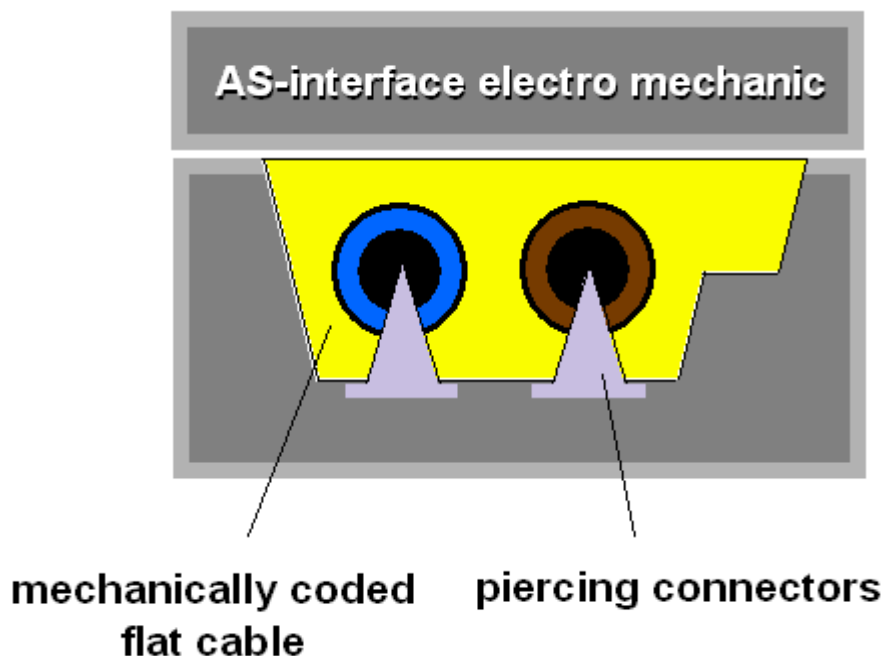


Figura 4.2: Connettore a Vampiro per BUS AS-Interface

Il cavo come è già stato detto è un due poli (tipicamente di colore giallo) e gli slave si agganciano a lui con dei connettori a vampiro. In questo modo è estremamente facile aggiungere o togliere slave alla rete.

### 4.3 ProfiBUS

ProfiBUS è uno standard sviluppato nel 1987, le sue specifiche e gli sviluppi sono gestiti dall'organizzazione ProfiBUS & ProfiNET International (PI). E' uno standard suddiviso in moduli (Fig. 4.4). Il fulcro del sistema è il modulo “ProfiBUS DP” che implementa il protocollo di comunicazione sul quale viaggiano le informazioni delle diverse applicazioni.

Sono implementabili trasmissioni via cavo, via fibra ottica o wireless. Le comunicazioni via cavo su RS-485-IS e MBP-IS (IS: Intrinsecal Safe) sono utilizzabili in quei sistemi nei quali ci possono essere rischio di esplosioni.

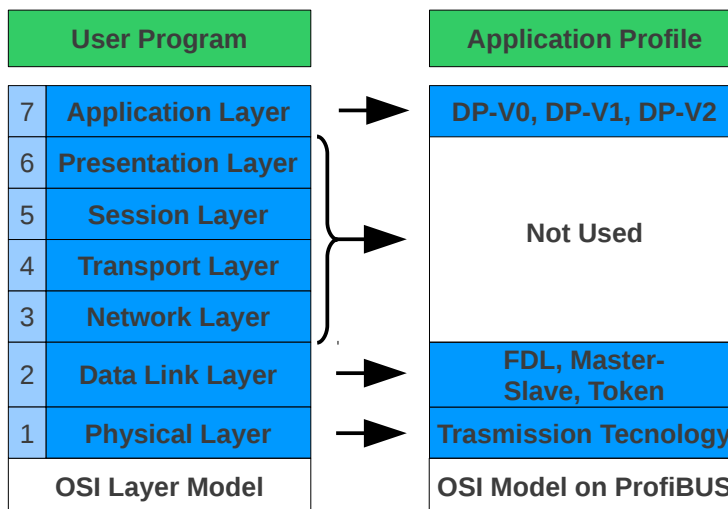


Figura 4.3: Layer di ProfiBUS rapportati al modello OSI

Lo standard ProfiBUS mette a disposizione diversi profili per lo scambio di dati di specifiche applicazioni. L'utilizzo di profili preimpostati permette una miglior interoperabilità tra componenti realizzati da produttori diversi.

Se rapportato al modello di riferimento ISO/OSI per la comunicazione, lo standard profibus si sviluppa su layer 1,2 e 7. (Fig. 4.3)

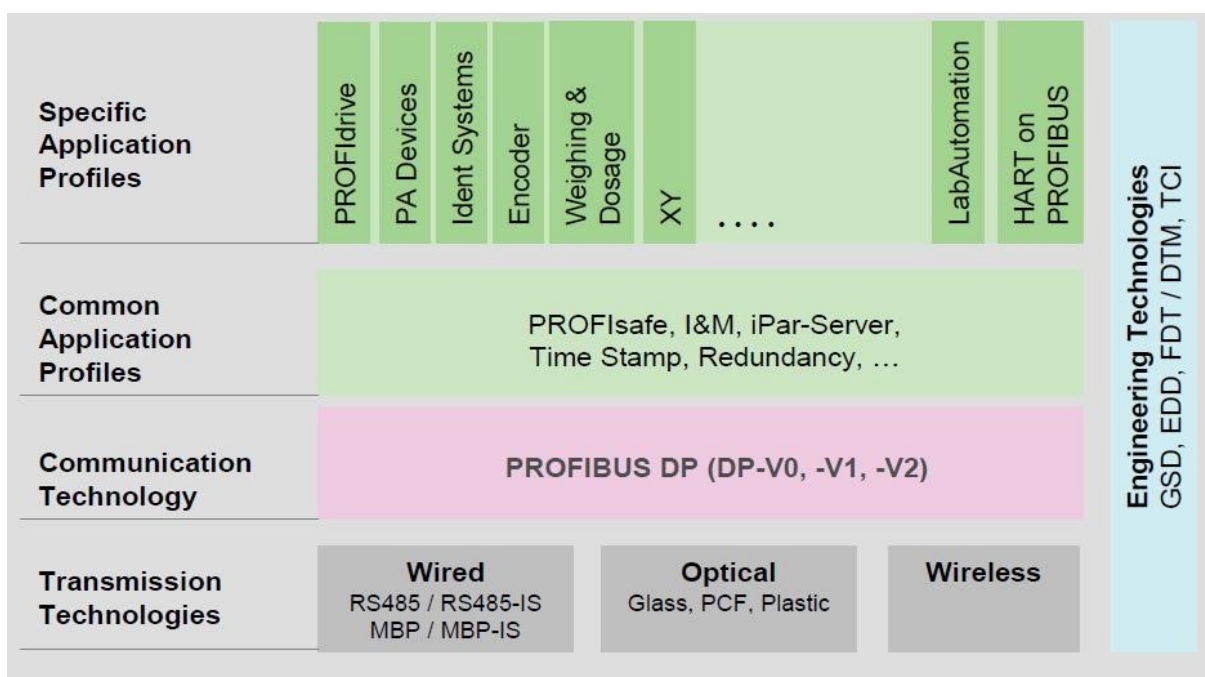


Figura 4.4: Sistema a Blocchi di Profibus

### 4.3.1 Tecnologie di trasmissione e topologia di rete

	RS485	RS485-IS	MBP	MBP-IS	Fibra ottica
<i>Tecnologia di trasmissione</i>	Digitale con segnale differenziale, NRZ	Digitale con segnale differenziale, NRZ	Digitale con bit di sincronizzazione e codifica di Manchester	Digitale con bit di sincronizzazione e codifica di Manchester	Ottico-Digitale, NRZ
<i>Velocità di trasmissione</i>	Da 9.6 a 12000 Kbit/s	Da 9.6 a 15000 Kbit/s	31,25Kbit/s	31,25Kbit/s	Da 9.6 a 12000 Kbit/s
<i>Sicurezza dei dati</i>	HD=4; Parity bit; delimitatore Start/end	HD=4; Parity bit; delimitatore Start/end	Preambolo; delimitatore Start/end	Preambolo; delimitatore Start/end	HD=4; Parity bit; delimitatore Start/end
<i>Cavo</i>	Doppino intrecciato, cavo schermato a 2 poli	Doppino intrecciato, cavo schermato a 4 poli	Doppino intrecciato, cavo schermato a 2 poli	Doppino intrecciato, cavo schermato a 2 poli	Multi o Single mode; Fibra di vetro o plastica.
<i>Alimentazione sul Bus</i>	Possibile solo con cablaggio aggiuntivo	Possibile solo con cablaggio aggiuntivo	Opzionale	Opzionale	Possibile solo con cavi ibridi
<i>Protezione Ex</i>	No	EX ib	No	EX ia/ib	No
<i>Topologia della rete</i>	Bus con terminale	Bus con terminale	Bus con terminale	Bus o ad albero con terminale	A stella, ad anello o a Bus.
<i>Numero di Nodi</i>	32 Nodi per segmento, Max 126 nodi nella rete	32 Nodi per segmento, Max 126 nodi nella rete	32 Nodi per segmento, Max 126 nodi nella rete	32 Nodi per segmento, Max 126 nodi nella rete	Max 126 nodi nella rete
<i>Numero di ripetitori</i>	Max 9 con refresh del segnale	Max 9 con refresh del segnale	Max 4 con refresh del segnale	Max 4 con refresh del segnale	Limitato solo dai ritardi di propagazione

La tecnica di trasmissione più usate è la RS485. La sua variante RS 485-IS, con sicurezza intrinseca per gli impianti con possibili rischi di esplosione, impone dei limiti sulle correnti e sulle tensioni del bus. Ogni qualvolta che si aggiunge un nodo al bus occorre verificare che la corrente totale richiesta sul quel segmento non superi il limite massimo imposto.

La trasmissione MBP (Manchester code Bus Powerd) implementa l'alimentazione e la comunicazione dei dispositivi allacciati al bus su un unico cavo. Questo permette una considerevole riduzione dell'overhead nella trasmissione. MBP è stato specificatamente sviluppato per venire incontro alle caratteristiche richieste dallo standard IEC 61158.2. MBP-IS può essere utilizzato in aree con possibili rischi di esplosione. A differenza del RS 485-IS questo metodo trasmissivo limita la corrente e la tensione di alimentazione dei dispositivi ad esso collegati.

In ambienti in cui sono presenti pesanti interferenze elettromagnetiche o dove sono richieste lunghe distanze è conveniente utilizzare la trasmissione su fibra ottica. Questo tipo di trasmissione è stata aggiunta dopo la nascita dello standard profiBUS e per poter assicurare una corretta compatibilità si è pensato semplicemente di interfacciare il metodo trasmissivo RS485 con un convertitore opto-elettronico (LED o Laser). Questo spiega perché le velocità di trasmissione su fibra ottica sono le stesse del RS485.

Con trasmissione RS485, si possono realizzare segmenti lunghi da 100 metri (con velocità sopra i 3000Kbit/s) a 1200 metri (con velocità sotto i 93Kbit/s), con MBP si possono realizzare collegamenti fino a 1900 metri. Per la fibra ottica i collegamenti vanno da 100 metri con fibra plastica (diametro del core 1000um) a più di 15Km con la fibra a singolo modo.

PROFIBUS è utilizzato anche nella comunicazione senza fili; Anche se, PI (ProfiBUS & ProfiNET International) non ha ancora messo a disposizione specifiche o linee guida, l'interoperabilità con sistemi cablati è ancora garantita. Le linee guida che specificano l'integrazione di WirelessHART (utilizzata nei processi di automazione) e Wireless Sensor/Actuator Network (WSAN) sono ancora in via di sviluppo.

Con ProfiBUS si possono implementare diverse topologie di rete anche se per ogni tecnologia di trasmissione usata ne esiste una più implementata. La trasmissione con RS 485 tipicamente avviene su rete di tipo BUS con due terminali ai capi del segmento. Su ogni segmento si possono installare fino a 32 nodi. Nel caso ne fossero richiesti più slave è necessario utilizzare ripetitori di segnale. Per la trasmissione MBP in pratica la topologia più usata è la *“trunk & spur”* (Fig. 4.5)

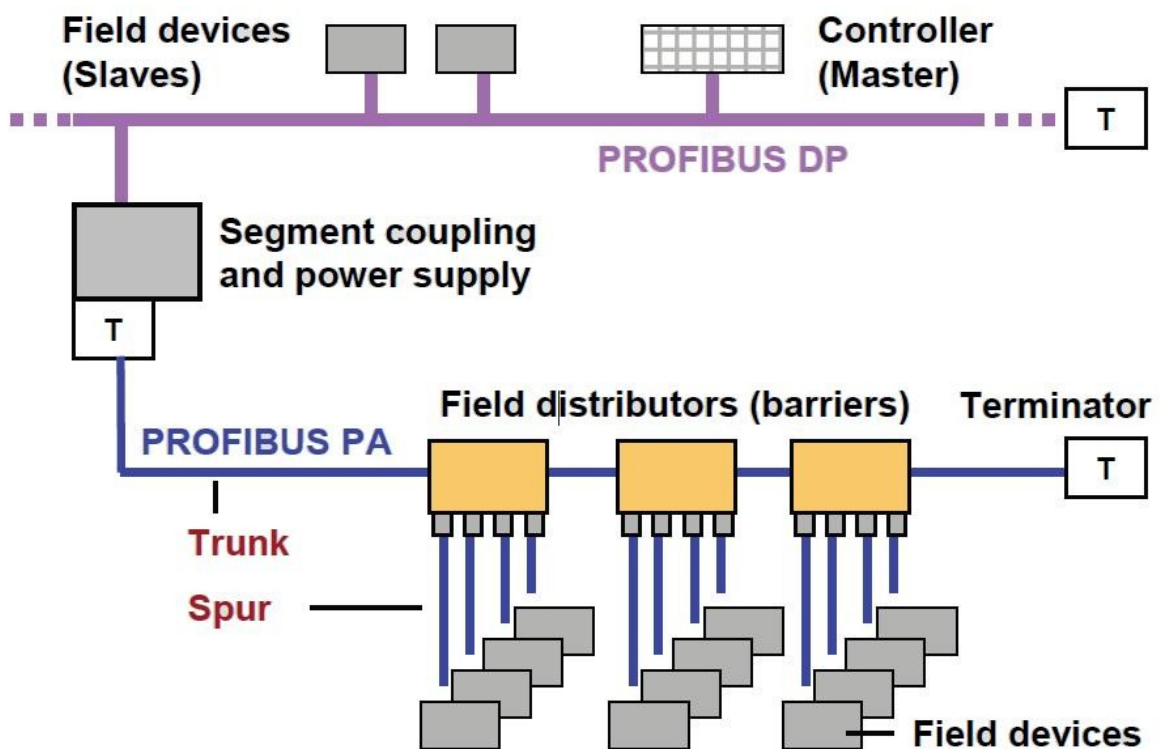


Figura 4.5: Esempio rete MBP e RS485 Interfacciati

In alcuni campi applicativi è essenziale che un guasto non interrompa l'interno processo produttivo e che si possa eseguire la manutenzione e la sostituzione dei componenti guasti senza che il ciclo di lavorazione venga interrotto. La soluzione migliore in questi casi è assicurarsi una ridondanza dei componenti del sistema.

In Profibus è possibile implementare sistemi con:

- Master in parallelo, in caso di guasto la comunicazione viene gestita dal secondo Master.
- Ridondanza dei collegamenti, realizzabile anche tramite topologia di rete da anello
- Segmenti accoppiatori in parallelo, come nel master, in caso di guasto la comunicazione viene gestita dal secondo accoppiatore.
- Slave in parallelo. I due (o più slave) devono essere identici. Solo lo slave primario risponde alle comunicazioni, in caso di guasto la comunicazione passa ad uno Slave secondario.

(Vedi Fig. 4.6)

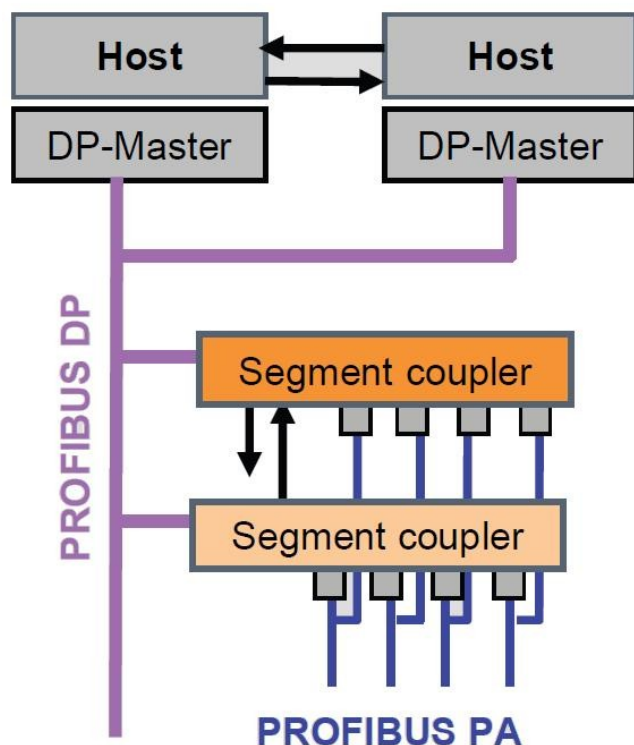


Figura 4.6: Esempio di ridondanza in un sistema Profibus

Non solo, in Profibus esistono particolari tool di sviluppo, integrabili nel sistema, che effettuano la diagnostica del Bus. In questo modo è possibile effettuare periodicamente la diagnostica del bus per monitorare un eventuale degrado nelle prestazioni che possono causare errori sulla comunicazione.

### 4.3.2 Comunicazione

In Profibus tutti i componenti utilizzano il protocollo di comunicazione Profibus DP (Decentralized Periphery), il quale specifica le regole per le comunicazioni cicliche e asincrone. Tutta la gestione dello scambio dati è eseguita dal master, il quale interroga ciclicamente gli slave, in caso di necessità può sospendere il ciclo e interrogare un altro slave in modo aciclico. In un sistema multi-master la comunicazione viene gestita dal master in possesso del token. (Fig. 4.7)



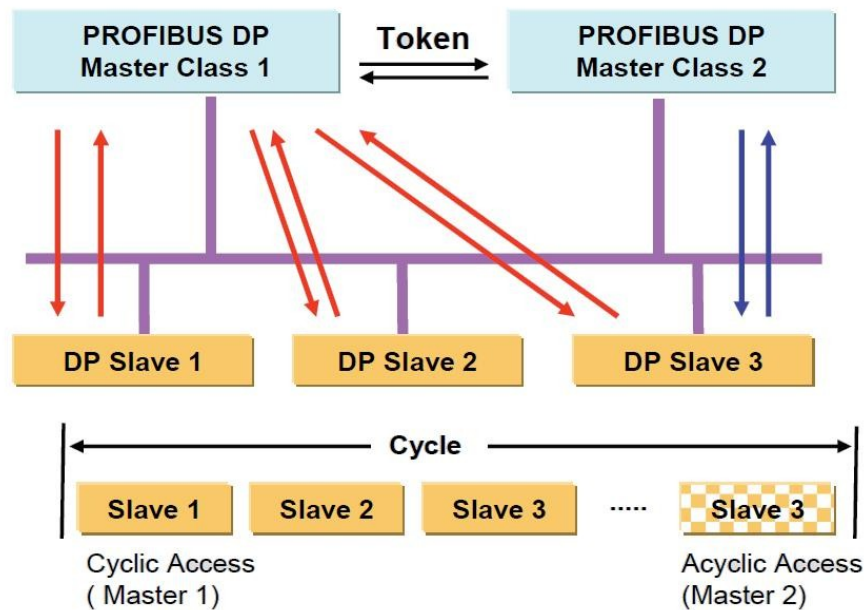


Figura 4.7: Comunicazione Ciclica a Aciclica con MultiMaster

Il protocollo Profibus DP si distribuisce su tre livelli con prestazioni diverse:

- **DP-V0** implementa le funzioni base per le comunicazioni cicliche e un sistema di diagnostica per individuare slave guasti.
- **DP-V1** oltre alle funzioni già presenti in DP-V0, implementa le comunicazioni acicliche.
- **DP-V2** oltre alle funzioni già presenti in DP-V1, implementa comunicazioni Broadcast, Isocrone e altre particolari funzioni utilizzate per il controllo degli azionamenti.

Prima di iniziare la comunicazione ciclica occorre settare alcuni parametri, come ad esempio: indirizzo del master, timeout, ID(è univoco, viene definito dalla organizzazione PI). Questi vengono trasferiti a gli slave e verificati. Gli slave rispondono con un report indicante il proprio set di parametri, se coincidono con quelli inviati dal master la comunicazione può avere inizio. Questa diagnostica consente di evitare di iniziare una comunicazione qualora ci fosse uno slave guasto. Uno slave può avere una comunicazione ciclica attiva solo con un Master alla volta.

Esistono invece due metodi di comunicazione aciclica:

- **MS1.** Può avvenire solo tra un master e uno slave che hanno già in corso una comunicazione tra di loro. In questo modo il master è già a conoscenza dei parametri di comunicazione dello slave.
- **MS2** può avvenire tra più master contemporaneamente che non abbiano in corso una comunicazione ciclica. Ovviamente i master che iniziano

questo tipo di comunicazione devono essere a conoscenza dei parametri di comunicazione degli slave. Se non avviene uno scambio di dati la comunicazione MS2 cade in automatico dopo un tempo di Timeout.

### 4.3.3 Profili applicativi

Essendo ProfiBUS uno standard aperto, i componenti del sistema possono essere realizzati da diversi produttori. Per semplificare la comunicazione e per assicurare la massima compatibilità tra i componenti realizzati dai diversi produttori vengono forniti dei profili applicativi standardizzati. Ogni componente inserito nel sistema comunicherà i suoi dati tramite un'applicazione implementata nello standard. I principali profili applicativi sono:

- **PROFIdrive** è il profilo utilizzato dagli azionamenti elettrici. Definisce il comportamento del dispositivo, il metodo di accesso e i formati dei dati per l'unità.
- **ProfiBUS PA (PA Device, PA Profile)** Questo profilo è abbastanza generico ed è utilizzabile dalla maggior parte dei componenti presenti in un impianto. Definisce le funzioni e i parametri di componenti come ad esempio sensori, attuatori, valvole per adattarli ai rispettivi compiti. Inoltre questo profilo specifica come devono essere usati diversi servizi presenti in ProfiBUS, come lo scambio dati ciclico.
- **HART on ProfiBUS** è un profilo messo a disposizione per poter collegare dispositivi HART su ProfiBUS
- **PROFIsafe** In caso di guasto nell'impianto occorre adottare misure di sicurezza per evitare che possano verificarsi lesioni a gli operatori e per limitare i danni al sistema. Questo profilo serve proprio per implementare le procedure di Fail-Safe sul bus di comunicazione.
- **Identification & Maintenance.** Memorizza le informazioni e le specifiche di ogni dispositivo collegato al bus.

### 4.3.4 Integrazione dei dispositivi e certificazione

Essendo ProfiBUS uno standard aperto, esistono una moltitudine di dispositivi che possono comunicare su di esso. Prima di installare un nuovo componente occorre conoscere la sua versione, le sue specifiche e le sue operazioni. Per ridurre i tempi di installazione ogni dispositivo ha memorizzato al suo interno dei campi che servono per descriverlo in maniera standard (Fig. 4.8).

- **GSD *General Station Description*** E' un data sheet elettronico che le informazioni necessarie per poter instaurare una comunicazione ciclica con il dispositivo.
- **EDD *Electronic Device Description*** Definisce i parametri per la comunicazione aciclica.

- **DTM Device Type Manager** è un programma in cui sono specificate le funzionalità del dispositivo e i suoi parametri di comunicazione. A differenza del GSD e EDD questo è scritto sotto forma di programma software, quindi richiede anche un'interfaccia (**FDT Field Device Tool**)
- **TCI Tool Calling Interface** E' un'interfaccia che si inserisce tra il dispositivo e il sistema. E' usato per quei dispositivi con funzionalità complesse; permette di caricare dinamicamente parametri sul componente e di effettuare la diagnostica.

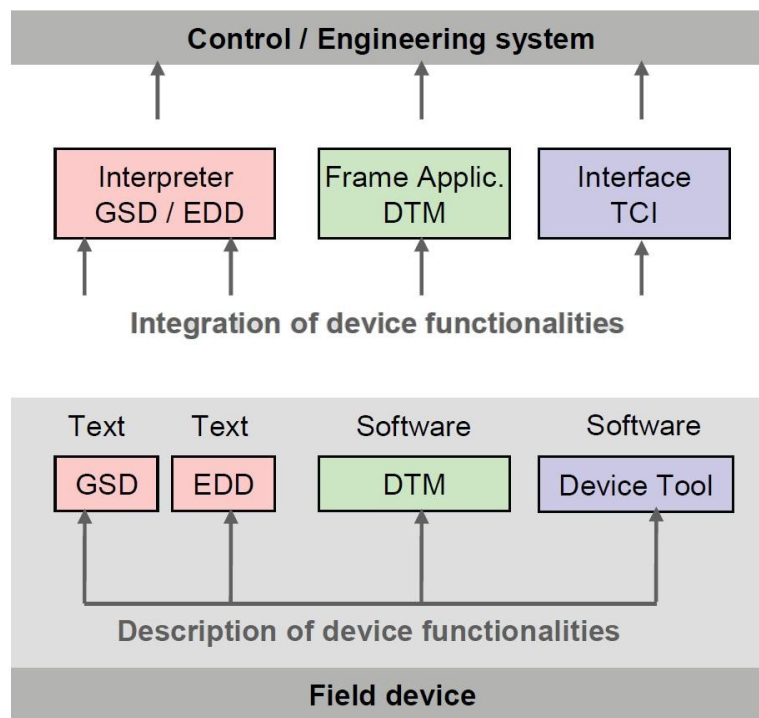


Figura 4.8: Tecnologie per l'integrazione dei dispositivi

Oltre a questi campi, per assicurare che un dispositivo risponda in maniera corretta ai requisiti del protocollo devono essere rilasciate delle certificazioni, eseguita da enti esterni per conto dell'organizzazione ProfiBUS & ProfiNET International, le quali certificano il corretto funzionamento su ProfiBUS del componente.

#### 4.4 ProfiNET

Profinet è uno standard che risponde ai requisiti descritti dalla norma IEC 61158; le sue specifiche e gli sviluppi sono gestiti dall'organizzazione ProfiBUS & ProfiNET International (PI). E' uno standard basato su industrial ethernet, quindi con un'unica rete si riesce ad cablare l'intera azienda, dall'impianto a gli uffici, anche tramite WLAN.

E' scindibile in due concetti: ProfiNET CBA e ProfiNET IO. Il primo riguarda le comunicazioni tra controllori e per comunicazioni soft real-time; il secondo riguarda una comunicazione più specifica verso i moduli I/O, per le comunicazioni cicliche real-time (RT) e isocrone real-time (IRT).

Come per ProfiBUS, ProfiNET ha tre livelli di implementazione (chiamati CC - “Conformance Class”). ProfiNET IO CC-A implementa le funzioni base per comunicazioni Real-Time ed è l'unica classe nella quale possono avvenire comunicazioni wireless, il CC-B oltre alle precedenti funzioni implementa la diagnostica della rete e le informazioni sulla topologia della rete, il CC-C oltre alle precedenti funzioni implementa le comunicazioni per applicazioni IRT, con banda di comunicazione riservata e isocrone.

ProfiNET IO segue un modello di comunicazione del tipo *Provider/Consumer*

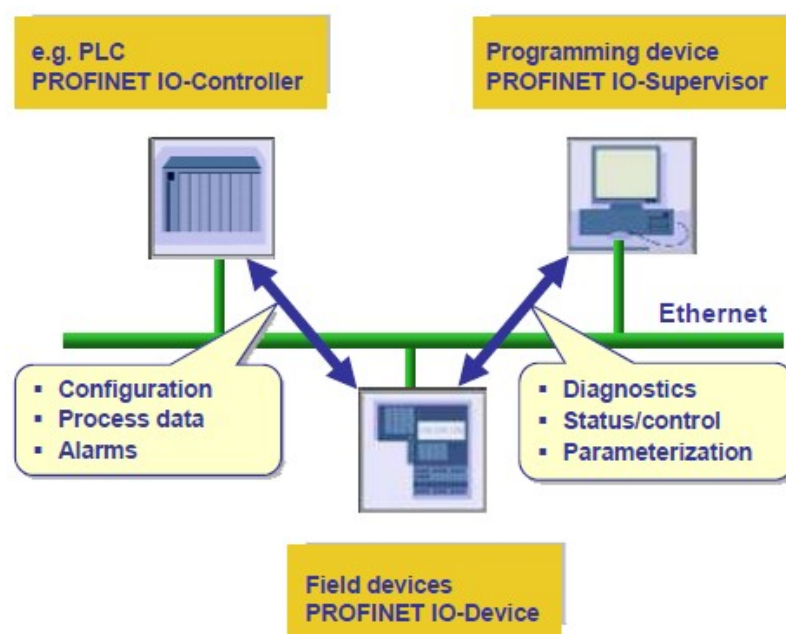


Figura 4.9: Rete su ProfiNET IO

Sono individuabili tre tipi di dispositivi collegabili alla rete: **IO-Controller** è tipicamente un PLC o comunque un componente sul quale viene lanciato il programma di gestione e controllo del processo produttivo, **IO-Device** è il dispositivo collegato a gli I/O del livello di campo (è comparabile con lo Slave in ProfiBUS), **IO Supervision** è un componenti programmabile, che controlla l'andamento dell'impianto; tipicamente è un HMI o un PC:

Come per ProfiBUS, anche in ProfiNET esistono alcuni campi per la descrizione del *IO-Device* come il GSD nel quale è descritto (in linguaggio XML) il funzionamento del dispositivo e il DAP (*Device Access Point*) che individua lo slot e il subslot per ogni I/O collegato all'*IO-Device* (Fig. 4.10). In una comunicazione ciclica per dati di input/output occorrono l'indirizzo dell'*IO-*

Device, lo slot e il subslot in cui il dispositivo è collegato. Per l'indirizzamento di una comunicazione aciclica oltre ai dati appena descritti per la comunicazione ciclica occorre l'*index* che identifica il dato che si vuole leggere/scrivere all'interno della coppia slot/subslot.

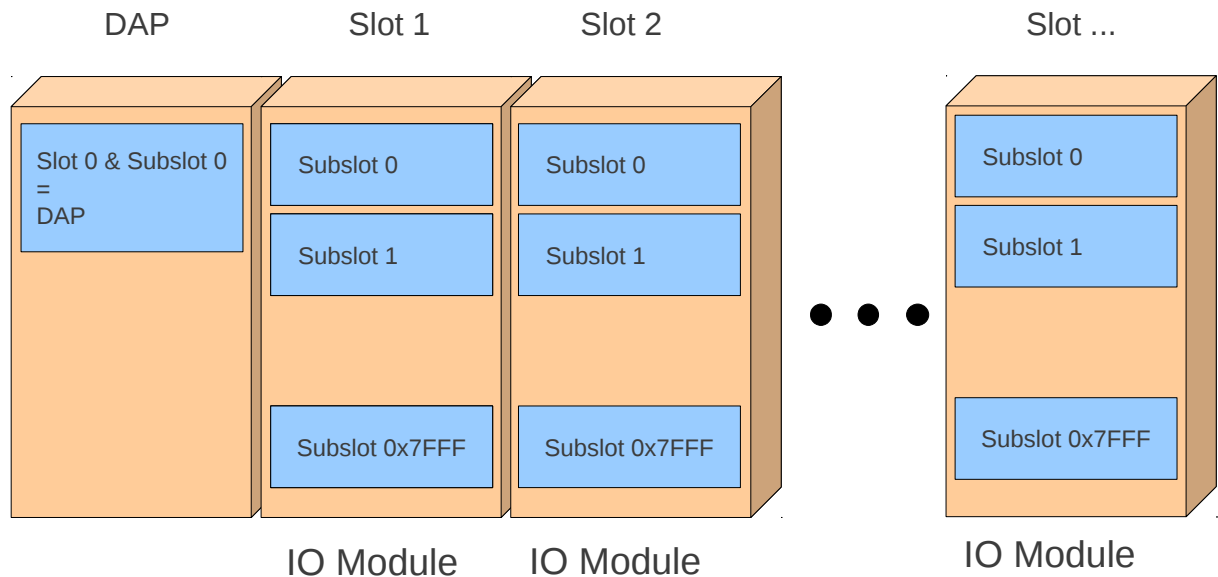


Figura 4.10: Indirizzamento dei dati di I/O basato su Slot e Subslot

Qualunque scambio di dati è incapsulato in un AR (Application Relation), al suo interno sono presenti vari CR (Communication Relation) nel quale sono specificati i dati. I CR possono essere:

- IO data CR usati nelle comunicazioni cicliche
- Record data CR usati nelle comunicazioni acicliche
- Alarm CR usati dagli allarmi

In un sistema ProfiNET possono coesistere più *IO-Controller* che accedono a gli stessi dati di un *IO-Device*. Un controllore quindi può avere un AR aperto con ogni dispositivo di I/O (Fig. 4.11)

Ogni dispositivo ProfiNET IO ha un ID univoco, al quale è associato poi un indirizzo MAC e uno IP. L'indirizzo IP può essere assegnato dinamicamente tramite il protocollo DCP (*Dynamic Configuration Protocol*), il DHCP (*Dynamic Host Configuration Protocol*) che è più usato a livello globale.

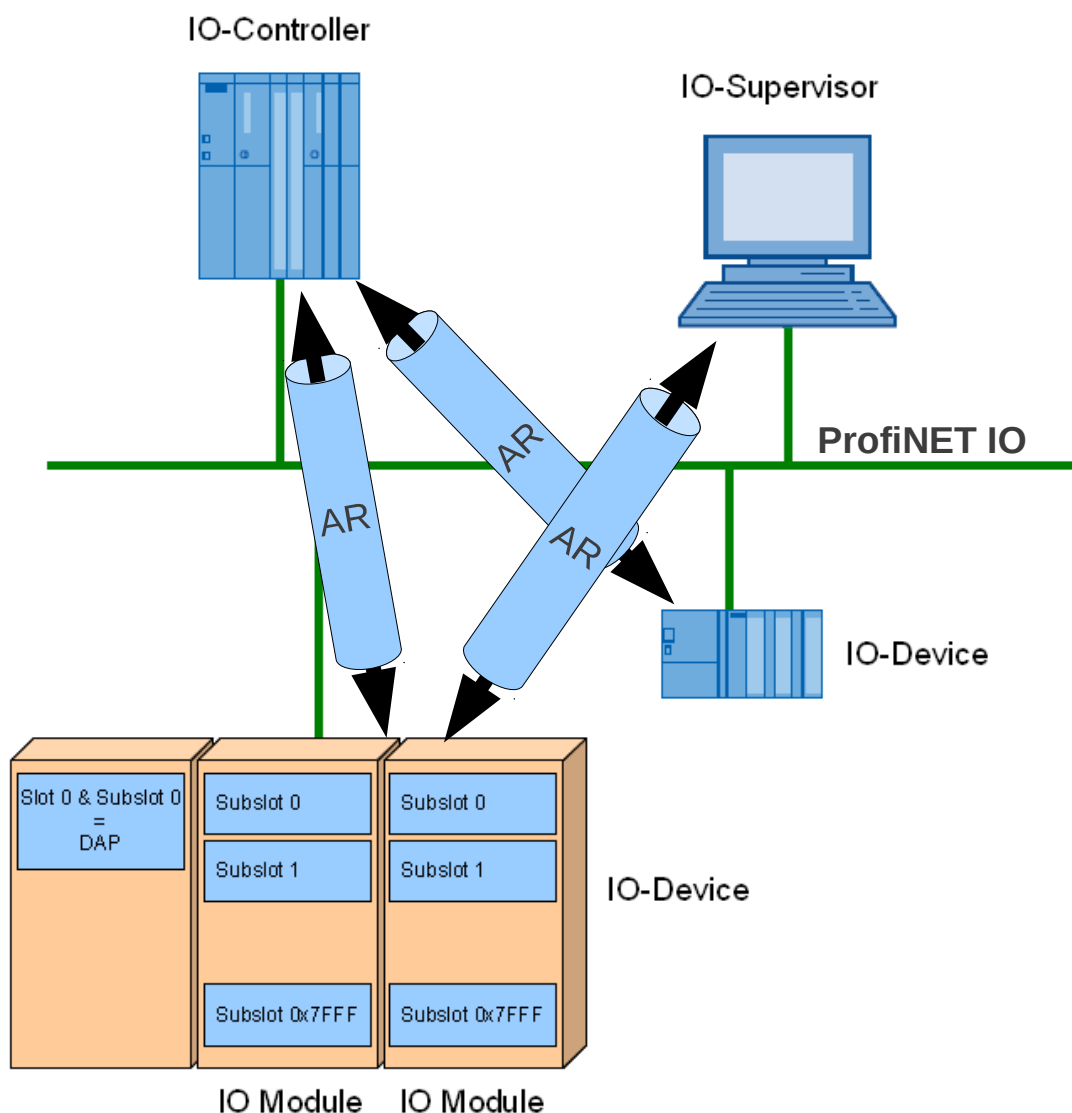


Figura 4.11: Comunicazione AR con device ad accesso multiplo su ProfiNET IO

Durante lo sviluppo della rete occorre indicare cosa è collegato a tutti gli slot/subslot dei moduli di IO. Per far questo si programma il file GSD di tutti i moduli tramite i tool di sviluppo e di programmazione dei controllori. In questo modo i controllori riescono a conoscere perfettamente la rete.

La comunicazione in ProfiNET si appoggia sull'industrial ethernet che descrive solo i layer 1 e 2 della pila ISO/OSI (Vedi Fig. 3.8 a pag. 30). Su questi layer implementa le comunicazioni RealTime (RT) e Isochronous RealTime (IRT) tramite particolari accorgimenti descritti successivamente (Vedi capitolo ProfiNET CC-A (Conformance Class A) a pag 51 e ProfiNET CC-C (Conformance Class C) a pag 55). Invece, per i dati che non necessitano di

particolari requisiti temporale la comunicazione viene eseguita tramite TCP o UDP. (Fig. 4.12)

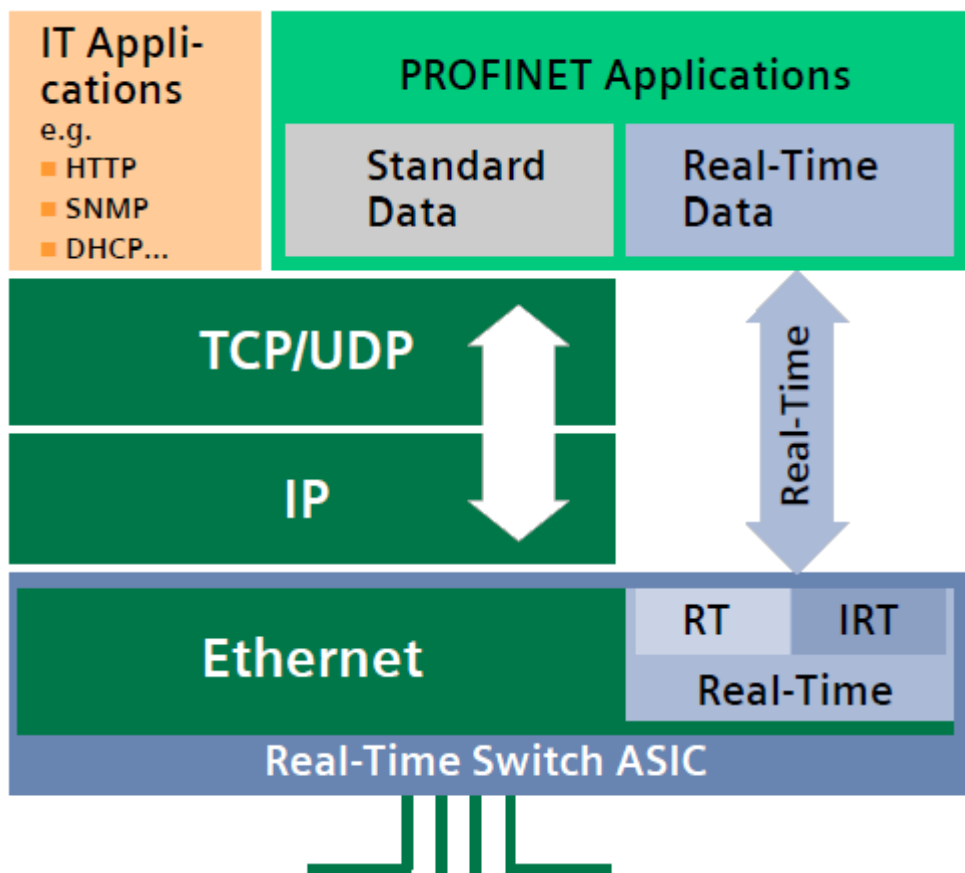


Figura 4.12: Schema a blocchi di ProfiNET

Questo consente a ProfiNET di poter supportare tutte le funzione classiche del web. In questo modo è possibile monitorare l'impianto e eseguire test di diagnostica da qualunque parte del mondo senza dover aggiungere particolari interfacce.

#### 4.4.1 ProfiNET CC-A (Conformance Class A)

ProfiNET di Classe A permette la comunicazione ciclica per lo scambio dati RealTime di I/O e Aciclica per la lettura e scrittura di dati. Inoltre implementa le funzioni di *Identification & Maintenance* per la lettura delle informazioni sul dispositivo e un sistema con tre livelli di allarmi (richiesta manutenzione, urgente richiesta manutenzione, diagnostica).

##### Scambio dati Ciclico

I dati per comunicazioni realtime cicliche vengono scambiati tramite “IO data CR” senza conferma di ricezione (ACK). I tempi di ciclo possono essere settati

individualmente per ogni dispositivo con un range che va da 250 us a 512ms. Il message frame contenente i dati è seguito da un insieme di informazione sulla validità dei dati, la ridondanza a la diagnostica. Se i dati non arrivano entro un tempo limite il *Consumer* invia un errore alla applicazione.

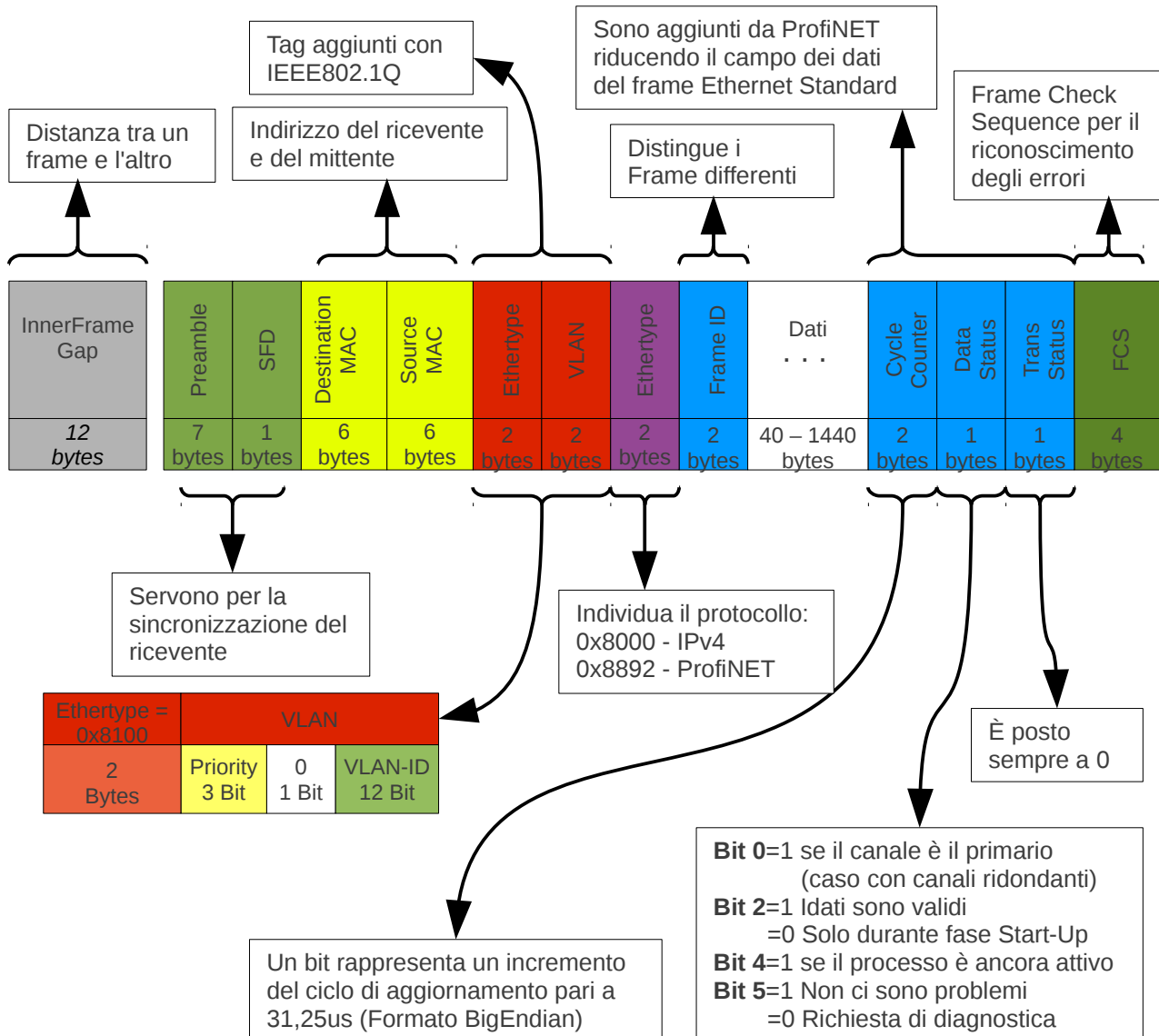


Figura 4.13: Frame Ethernet adattato al protocollo ProfiNET con comunicazione ciclica

La comunicazione ciclica avviene su rete ethernet direttamente sul layer 2 con EtherType 0x8892 (PROFINET real-time telegram), quindi senza l'utilizzo di TCP, UDP, IP. Per ottimizzare lo scambio ciclico dei dati, in accordo con lo standard IEEE802.1Q che specifica l'implementazione di più reti VLAN su lo stesso collegamento fisico, è possibile utilizzare dei tag per specificare la priorità del pacchetto. (Fig. 4.13) Rispetto al frame ethernet con IEEE802.1Q, sono stati aggiunti 6 byte (Frame ID, CycleCounter, DataStatus, TransStatus) sottraendoli allo spazio dedicato ai dati.



## Scambio dati Aciclico

Lo scambio dati aciclico è utilizzato per l'assegnazione/configurazione di parametri o la lettura dello stato dei dispositivi di I/O, sfruttando il “Record Data CR”. Sono disponibili altri record data per:

- **Diagnostica** l'utente in qualunque momento può leggere le informazioni di stato della rete o di qualunque dispositivo.
- **Registro errori** per avere informazioni dettagliate su quando è avvenuto un errore o è stato lanciato un allarme.
- **Identification & Maintenance (I&M)** per avere informazioni sul dispositivo. I&M è diviso in 5 blocchi (IM0 ... IM4). Solo IM0 è obbligatorio.

La comunicazione aciclica avviene su UDP con richiesta di ACK che conferma la corretta ricezione del messaggio. Se non si riceve l'acknowledgement il dato viene rinviato.

Il frame ethernet della comunicazione aciclica in ProfiNET differisce leggermente dal frame per la comunicazione ciclica, infatti non sono presenti i dati CycleCounter, DataStatus, TransStatus. In quanto non servono per la

InnerFrame Gap	Preamble	SFD	Destination MAC	Source MAC	Ethertype	VLAN	Ethertype	Frame ID	Dati ...	FCS
12 bytes	7 bytes	1 bytes	6 bytes	6 bytes	2 bytes	2 bytes	2 bytes	2 bytes	36 – 1490 bytes	4 bytes

Figura 4.14: Frame Ethernet adattato al protocollo ProfiNET con comunicazione aciclica (per la descrizione dei campi si rimanda alla Fig. 4.13)

comunicazione aciclica.

## Diagnostica dei dispositivi e di rete

Si basa sulla capacità dei dispositivi di determinare il proprio stato e di comunicarlo con i giusti meccanismi. Questo concetto è molto utile per poter eseguire rapidamente operazioni di manutenzione e ricerca del guasto.

Gli allarmi devono coprire eventi previsti dal sistema (come inserzione o rimozione di un modulo) o guasti del sistema. In ProfiNET sono previsti due livelli di preallarme:

- **Richiesta di Manutenzione (Maintenance Required)** il guasto non comporta l'interruzione dei processi e il sistema è ancora pienamente funzionante. (es. nel caso in cui il dispositivo guasto è ridondante).
- **Pretesa di Manutenzione (Maintenance Demanded)** il guasto comporta il rischio di interruzione dei processi o il sistema subisce una degradazione

sulla qualità delle sue risposte (es. un sensore che non fornisce più il livello di accuratezza richiesto o un dispositivo che è a fine vita).

Gli allarmi della diagnostica devono essere utilizzati se l'errore o evento si verifica all'interno di un IO-Device o in combinazione con i componenti collegati. Possono essere definiti dal programmatore degli allarmi di processo (*Process Alarm*), che indicano che l'impianto non sta funzionando in maniera corretta. (Es. un sensore di indica una temperatura eccessiva). Questi allarmi possono implementare diverse priorità.

#### 4.4.2 ProfiNET CC-B (Conformance Class B)

Conformance Class B espande la classe precedenti aggiungendo altre funzioni utili per l'implementazione della rete: come la diagnostica della rete e la individuazione della topologia. Per quest'ultimo scopo, ProfiNET utilizza il protocollo SNMP, parte del MIB2 e il LLDP-EXT. Questi protocolli devono essere integrati nei dispositivi certificati ProfiNET CC-B.

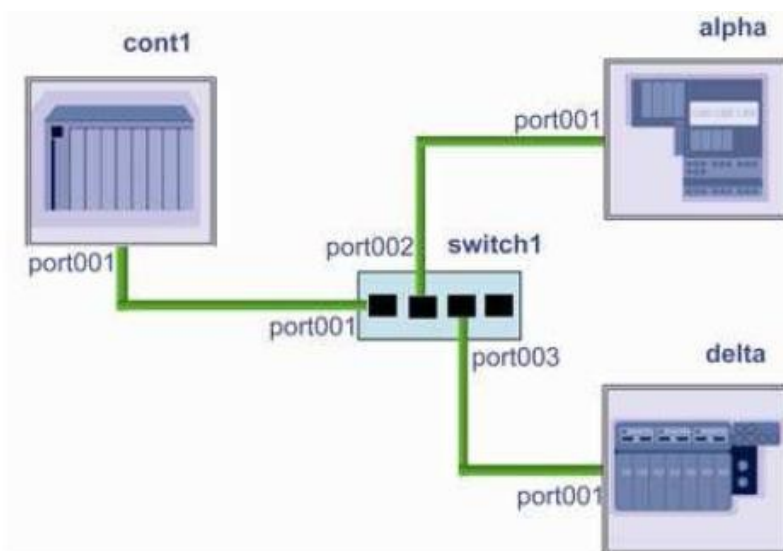


Figura 4.15: I dispositivi individuano il vicinato

Il Protocollo SNMP è uno standard molto utilizzato per la manutenzione e il controllo dei componenti di una rete e del loro funzionamento. SNMP ha accesso in lettura ai componenti, per poter conoscere alcuni dati pertinenti alla rete come le porte utilizzate e l'individuazione del vicinato.

I dispositivi di campo che comunicano in ProfiNET scambiano informazioni sui loro indirizzi con i dispositivi collegati allo switch. In questo modo tutti i componenti sono identificabili e localizzabili all'interno della rete ed è possibile averne una rappresentazione grafica.

Nel caso in cui un dispositivo si guasta ed è necessario provvedere alla sua

sostituzione si può riconfigurare la rete oppure il componente sostitutivo riceverà il nome e i parametri del suo predecessore.

Gli switch della rete possono essere configurati come *IO-Device* quindi posso individuare e segnalare guasti nella rete all'*IO-Controller* tramite gli allarmi “*Alarm CR*”. In questo modo la diagnostica della rete viene integrata nel sistema.

#### **4.4.3 ProfiNET CC-C (Conformance Class C)**

Conformance Class C espande la classe precedenti aggiungendo altre funzioni necessarie per la sincronizzazione utilizzata in applicazioni che richiedono tempistiche sulla comunicazione molto stringenti. Conformance Class C consente di avere jitter inferiori al 1us e i tempi di ciclo passano dai 250us fino a 31,25us con particolari accorgimenti. Tutto questo senza precludere il funzionamento dei dati trasmessi sulla rete ethernet standard.

Per poter assicurare un jitter inferiore al 1us, i dispositivi devono avere un clock sincronizzato. E' possibile avere più componenti con sincronizzazioni differenti ma è fondamentale che i dispositivi con lo stesso allineamento di clock risiedano tutti sulla stessa linea bus che non può contenere dispositivi non sincronizzati o sincronizzati con clock differente.

Per poter eseguire una corretta sincronizzazione è necessario conoscere la lunghezza del cavo di collegamento, è possibile calcolarne una stima misurando il tempo che intercorre tra l'invio di un pacchetto e la ricezione del suo “echo”. Questa misura è precisa solo se realizzata con dispositivi che possano effettuare un eco veloce e che abbiano timer efficienti allo scopo.

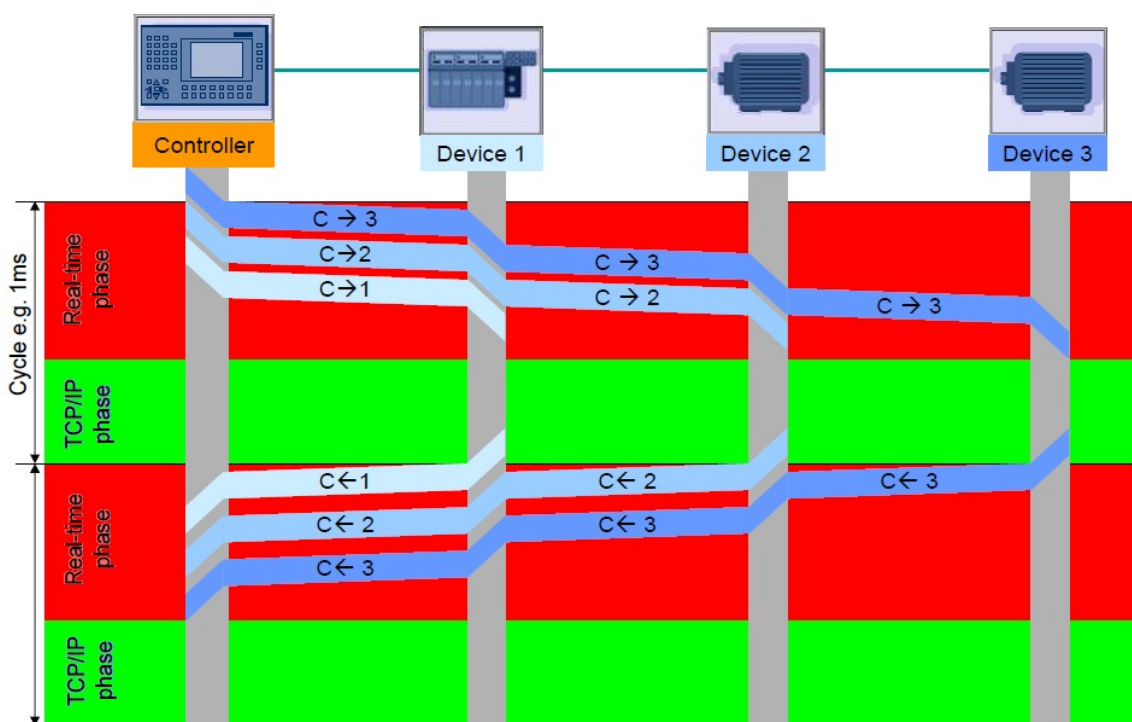


Figura 4.16: Comunicazione sincronizzata (senza DFP)

La sincronizzazione è essenziale per poter eseguire una comunicazione real-time su un Data Link Layer, come ethernet, condiviso da altre comunicazioni a bassa priorità. Infatti, la comunicazione in ProfiNET CC-C è divisa in due fasi: la fase “rossa” nella quale avvengono solo le comunicazioni real-time (ad alta priorità) e una fase “verde” dove possono avvenire le normali comunicazioni TCP/IP o a bassa priorità. (Fig. 4.16) Si intuisce quindi che il ruolo della sincronizzazione è quello di evitare la trasmissioni di pacchetti TCP/IP (o a bassa priorità) durante la comunicazione real-time.

Nel caso in cui siano richiesti requisiti ancora più stringenti sulle tempistiche di comunicazione si può migliorare l'efficienza della comunicazione sincrona utilizzando il Dynamic Frame Packing (DFP). L'idea è quello di eseguire la comunicazione tra dispositivi tramite un unico frame ethernet; ogni dispositivo “apre” il pacchetto, estrae il contenuto che lo riguarda, riassume il frame e lo rinvia a gli altri componenti sul bus. (Fig.4.17).

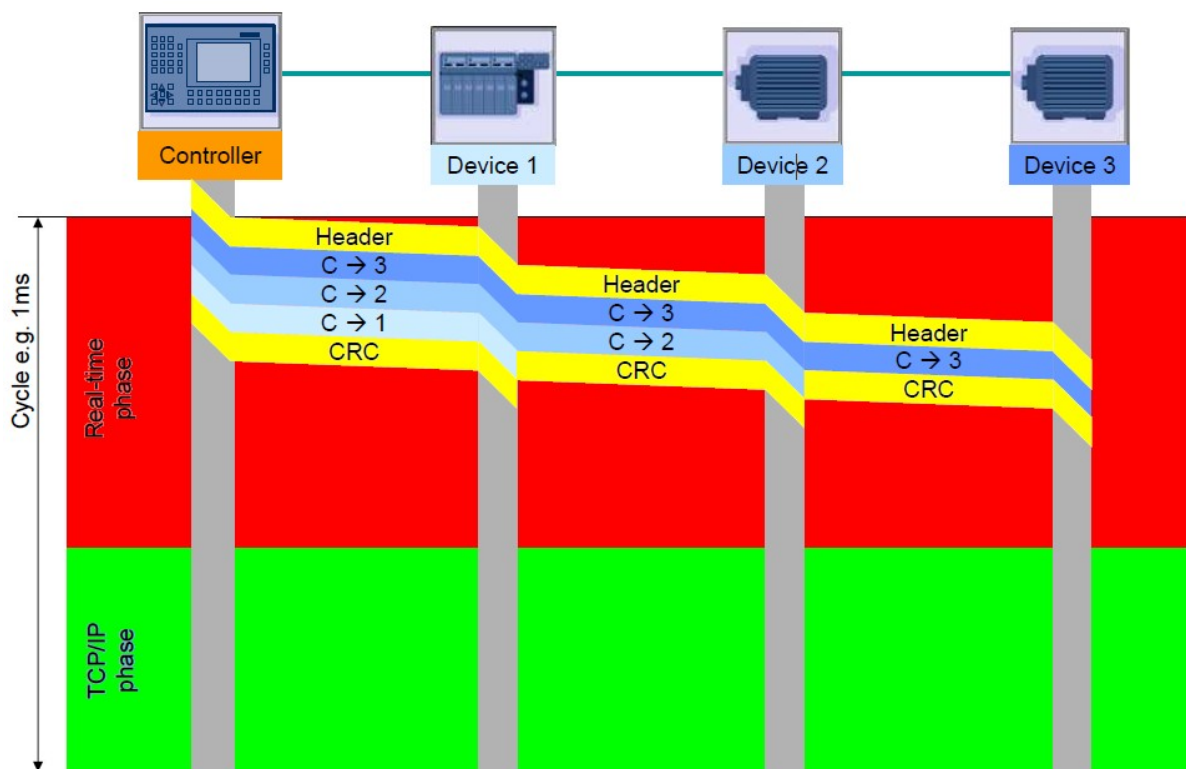


Figura 4.17: Comunicazione sincrona con DFP

Idealmente, nella fase real-time, viene spedito quindi un solo frame per tutti i dispositivi. Questo permette di raggiungere l'obiettivo di un cycle rate pari a 31,25us ma comunque, l'area “verde” per le comunicazioni standard ethernet dovrà subire una riduzione. Questo comporta che i pacchetti ethernet dovranno essere piccoli (eventualmente eseguendo una frammentazione del frame) per assicurarsi che la comunicazione sia finita prima del inizio della fase real-time.

#### 4.4.4 Funzioni opzionali aggiuntive

Esistono ulteriori funzioni non descritte nelle varie “Conformance Class” ma che possono essere implementate individualmente dai vari dispositivi.

- **Accesso Multiplo ai device di campo** Lo stesso dispositivo può essere letto e scritto da due controllori diversi. Questo può avvenire con una assegnazione fissata a priori durante la programmazione del sistema. Ci possono essere diversi motivi per i quali è necessario eseguire l'accesso multiplo. Un esempio può essere il caso in cui si debba eseguire una procedura di fail-safe nella quale la CPU di sicurezza prende il controllo del sistema e deve accedere a moduli di I/O condivisi con la CPU utilizzata nel funzionamento standard.
- **Estensione del Identification & Maintenance (I&M).** I&M contiene 5 campi ma solo IM0 è obbligatorio. I campi IM1 ... IM4 sono opzionali,

vengono creati e programmati durante la configurazione, contengono informazioni: sull'impianto in cui andranno ad operare (IM1), sull'installazione (Data e locazione)(IM2), un area per i commenti (IM3) e una firma digitale (IM4).

- **Individual Parameter Server.** I parametri base del dispositivo sono definiti nel GSD e sono caricati tutte le volte all'accensione. Altri parametri non possono essere caricati con il file GSD, a causa delle loro dimensioni o per motivi di sicurezza. Questi parametri specifici al dispositivo o alla tecnologia sono contenuti nel iPar (Individual Parameters). Sono configurati dal tool di programmazione specifico del componente e devono essere riconfigurati qual'ora il componente deve essere sostituito.
- **Configuration in Run (CiR)** ProfiNET consente una riprogrammazione dei IO-Controller senza che si debba interrompere il processo operativo dell'impianto. Questo consente di eseguire riparazioni, aggiornamenti o espansioni all'impianto durante il suo funzionamento, senza interruzioni. (Fig. 4.18)

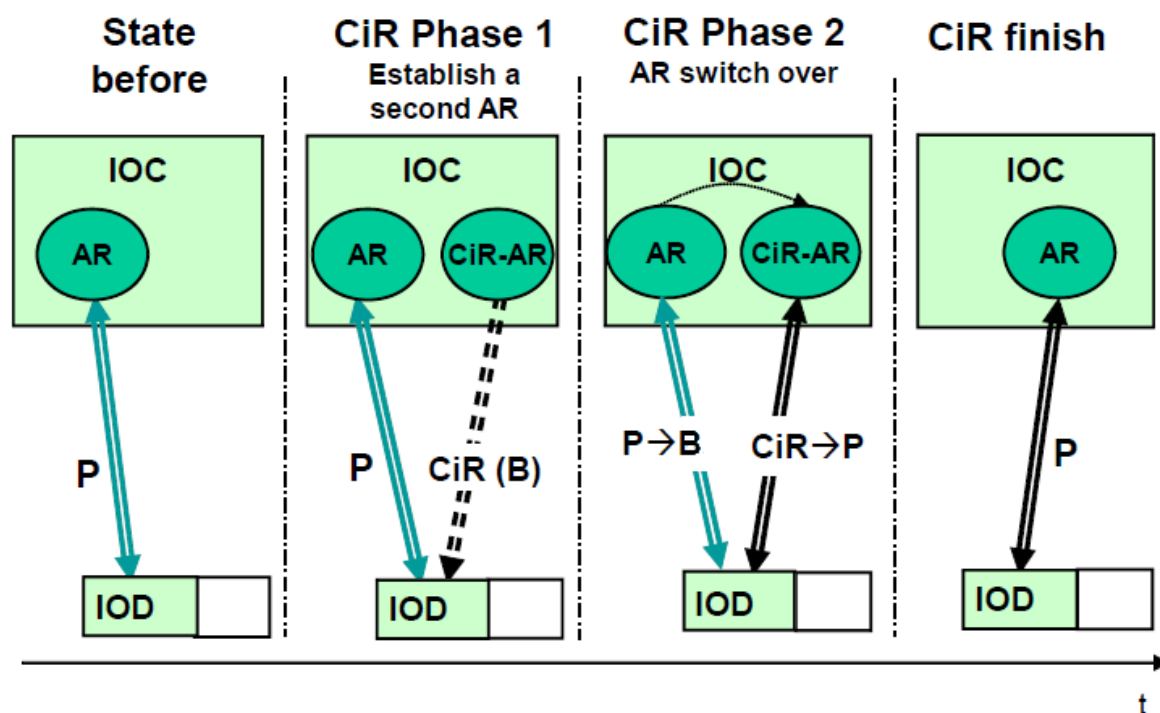


Figura 4.18: Cambio di AR tramite CiR (P: Primary, B: BackUp)

- **Time Stamping.** E' possibile associare a tutti i frame in transito la data e l'ora dell'invio. Questo serve per poter implementare un corretto controllo degli allarmi o dei messaggi di stato.
- **Ripartenza veloce.** All'accensione del sistema c'è uno scambio dati per la configurazione tra i controllori e i dispositivi i I/O. Molti di questi parametri

non cambiano fino a quando il sistema non subisce variazioni. Fast Start Up (FSU) definisce uno scambio dati ottimizzato per evitare la trasmissione di parametri che non sono stati modificati, mantenendoli memorizzati all'interno dei dispositivi.

- **Alta reperibilità** Avere una ridondanza nei collegamenti permette di evitare che un guasto su un segmento di rete comporti l'interruzione del processo di lavorazione. E' però necessario implementare protocolli che individuino il percorso alternativo per raggiungere il dispositivo. ProfiNET implementa MRP (Media Redundancy Protocol) MRPD (Media Redundancy for Planned Duplication). Il primo è un protocollo che nasce al di fuori dell'automazione e consente tempi di recupero inferiori ai 200ms, il secondo è descritto nella IEC61158 e consente il recupero della comunicazione anche per reti IRT (è implementabile solo su ProfiNET CC-C).
- **Tool Calling Interface** Alcuni dispositivi complessi (come i drivers o azionamenti) hanno programmi apposta per la loro configurazione (TCI). La configurazione dei parametri di questi dispositivi avviene in programmazione tramite il TCI e i dati vengono poi scritti sul dispositivo tramite ProfiNET

#### **4.4.5 Profili Applicativi**

Come in ProfiBUS anche in ProfiNET troviamo dei profili applicativi utilizzati per semplificare la comunicazione e per assicurare la massima compatibilità tra i componenti realizzati dai diversi produttori. Ogni componente inserito nel sistema comunicherà i suoi dati tramite un'applicazione implementata nello standard. Molti profili applicativi di ProfiBUS sono implementati anche in ProfiNET. Tra i principali nuovi profili applicativi c'è PROFIenergy

PROFIenergy permette di ridurre i consumi dei dispositivi, spegnendoli quando non sono utilizzati o riducendo i loro consumi abbassando la velocità del clock (quando possibile).

#### **4.4.6 Installazione di una Rete ProfiNET**

ProfiNET è basata su rete ethernet a 100Mbps, full duplex. Anche se è possibile realizzare collegamenti con velocità maggiori. Lo standard definisce non solo le procedure sulla comunicazione ma anche le caratteristiche che devono avere i componenti attivi e passivi connessi alla rete.

Il mezzo di comunicazione può essere il cavo ethernet classico (in rame) o la fibra ottica, solo i dispositivi CC-A possono utilizzare comunicazioni wireless (Bluetooth, WLAN). Come è già stato detto, per realizzare una rete conforme a CC-A sono necessari switch che implementano i tag VLAN descritti dal IEEE802.1Q. Mentre per realizzare reti conformi CC-B o CC-C servono

managed switch e managed switch con Bandwith Reservation (Switch sincronizzati). Tipicamente tutti i dispositivi ProfiNET hanno uno switch a 2 porte integrato che supportano l'”*autonegotiation*” (Negoziazione sulla trasmissione dei parametri) e l'”*autocrossover*” (che permette il forward automatico di alcuni dati). Su ProfiNET si possono realizzare reti in linea, a stella, ad anello o ad albero (ibride). Come per l'ethernet via cavo i segmenti non possono superare i 100m.

Lo standard indica anche alcune specifiche sul tipo di cavi utilizzabili con ProfiNET in funzione del tipo di shock meccanico che devono subire:

- **ProfiNET Type A:** Non sono previsti movimenti dopo l'installazione.
- **ProfiNET Type B:** Cavo flessibile, movimenti occasionali o vibrazioni.
- **ProfiNET Type C:** Cavo molto flessibile, movimenti continui (torsione e o trazione)

In alternativa al cavo in rame è possibile utilizzare il cavo in fibra ottica. La Fibra è particolarmente utilizzata in casi in cui sono richieste segmenti lunghi oltre le centinaia di metri o ambienti in cui ad alto inquinamento EMI (ElectroMagnetic Interference) o con esigenti requisiti EMC (ElectroMagnetic Compatibility), in quanto la fibra ottica non è soggetta e non genera campi elettromagnetici esterni alla cavo. La fibra più usata è la POF (Plastic Optical Fiber) con cladding di 1mm.

#### **4.4.7 Sicurezza**

Come già accennato per l'industrial ethernet (Capitolo 3.3.1 Sicurezza di rete a pag 33) il fatto che adesso tutto l'impianto e l'azienda vengano cablati su un unica rete basata su ethernet impone di implementare alcuni accorgimenti tecnici sulla sicurezza informatica per salvaguardare l'integrità dei dati e per evitare manomissioni, volontarie e non, sul corretto funzionamento dell'impianto.

Lo standard ProfiNET si raccomanda di suddividere la rete in aree con un modulo di sicurezza a monte. Questa tecnica da un lato riduce il traffico rete nelle aree locali, dove tipicamente avviene il traffico Real-Time che è il più oneroso in termini di banda, dall'altro consente una facile implementazione delle norme sulla sicurezza informatica. Infatti è possibile per ogni area definire quali utenti e quali pacchetti possono accedervi e quali modifiche possono eseguire. (Fig. 4.19)



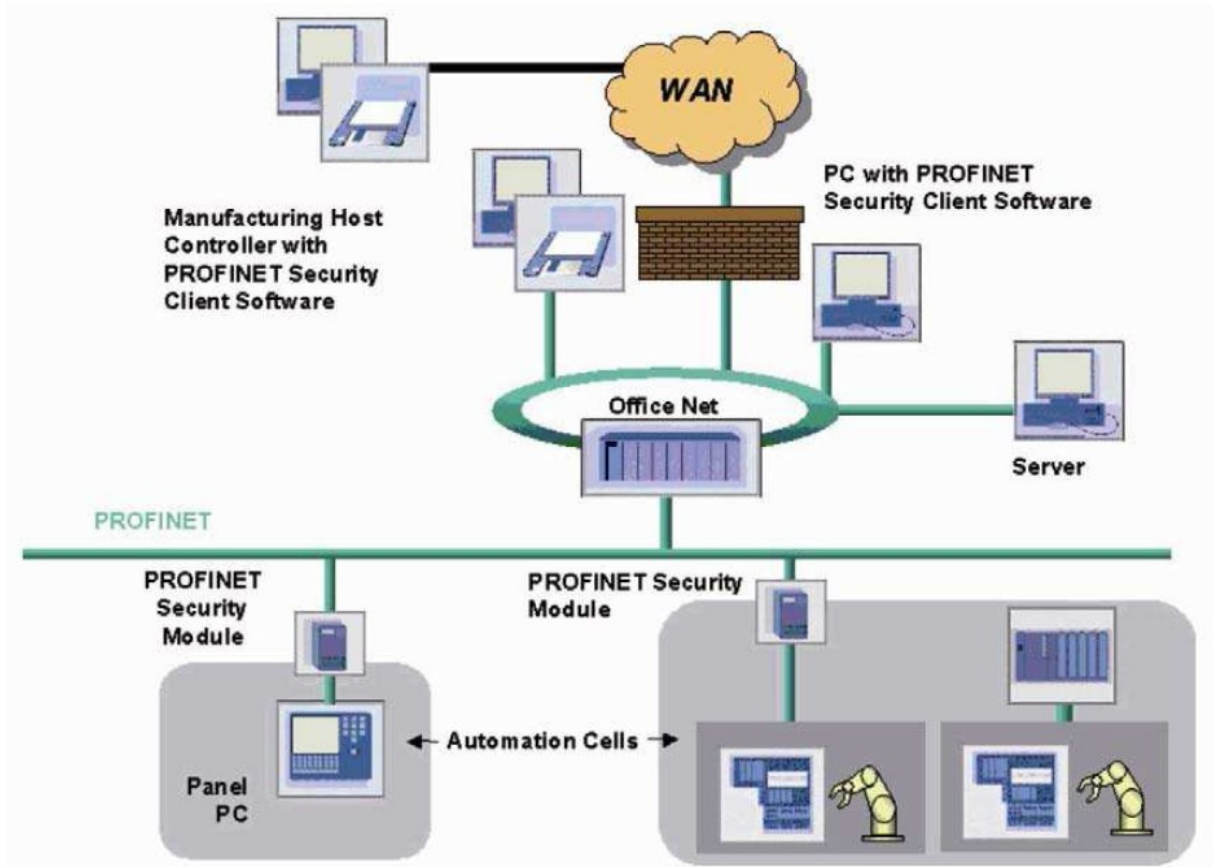


Figura 4.19: Schema di una rete ProfiNET con implementati di moduli per la sicurezza

#### 4.4.8 Certificazione dei prodotti

Come per ProfiBUS anche i produttori di dispositivi che comunicano in ProfiNET devono eseguire una certificazione che assicuri che il loro dispositivo risponda in maniera corretta ai requisiti del protocollo. La certificazione viene eseguita da enti esterni per conto dell'organizzazione ProfiBUS & ProfiNET International (PI), i quali certificano il corretto funzionamento su ProfiNET del componente.

#### 4.4.9 Limiti del ProfiNET

ProfiNET e ProfiBUS sono gestiti dall'organizzazione PI, la quale propone ProfiNET come lo standard che soppianderà il più vecchio ProfiBUS. Anche se, ad oggi, considerando i tre livelli di prestazioni di ProfiNET (CC-A, CC-B, CC-C in ordine crescente di performance), il ProfiBUS è comparabile con il CC-B con la differenza che i dispositivi ProfiBUS costano meno ed è uno standard ampiamente affermato.

Quindi se si deve realizzare un impianto con prestazioni superiori allo stesso realizzato in ProfiBUS è necessario utilizzare ProfiNET CC-C che richiede una componentistica più complessa (deve implementare la sincronizzazione) e

di conseguenza anche più costosa.

Non solo, implementare una rete IRT in ProfiNET richiede una buona pianificazione del sistema, sarà, infatti, possibile inserire devices RT su un segmento IRT solo isolandoli con uno switch sincronizzato (che funziona come un device IRT) ma non è sempre consigliabile. Occorrerà, infatti, prestare molta attenzione perché il traffico RT sarà limitato solo nei periodi “green” (Vedi ProfiNET CC-C (Conformance Class C) a pag. 55) e quindi se questo traffico è alto si rischia una rapida congestione del segmento. Si intuisce che esiste una forte dipendenza tra le prestazioni del ProfiNET e la rete in cui viene implementata. Inoltre, queste prestazioni possono variare notevolmente se vengono aggiunti nuovi moduli (nei casi più gravi può essere richiesta una nuova pianificazione della rete).

In caso di guasti sulla rete lo standard ProfiNET suggerisce diverse soluzioni che comportano la ridondanza dei collegamenti e implementa due standard per individuare un nuovo routing con i collegamenti aggiuntivi: MRP e MRPD (Vedi Funzioni opzionali aggiuntive a pag. 57). Il primo è un protocollo che non nasce per ripristinare i collegamenti in un fieldbus e quindi ha delle tempistiche non sempre adatte ai requisiti RealTime, mentre MRPD offre prestazioni nettamente migliori. MRPD però è implementabile solo su ProfiNET CC-C.

Come si è appena visto, i punti più critici del ProfiNET riguardano proprio la rete. D'altra parte l'ethernet non nasce per le comunicazioni a livello di campo e far condividere lo stesso mezzo a dati molto diversi tra loro, come il traffico RT o IRT e quello TCP/UDP richiede dei compromessi e sono richiesti componenti hardware ad alte prestazioni. Non per niente, nonostante ProfiNET sia uno standard aperto è ancora difficile trovare componenti non realizzati da Siemens soprattutto per quanto riguarda controllori IRT. Questo è dovuto al fatto che sono molto difficili da realizzare e che il software per implementare la comunicazione è complesso e le API sono costose.

## **5 RICERCA COMPONENTISTICA PER IL NUOVO SISTEMA**

Come già accennato in precedenza, l'azienda IEMCA lavora principalmente con prodotti Siemens, quindi è importante che il prototipo, sul quale verrà sviluppato il nuovo macchinario per il caricamento automatico di barre su un tornio, sia basato su prodotti Siemens al fine di poter utilizzare il know-how dell'ufficio tecnico e non richiedere nuovi tool di sviluppo. D'altro canto, è necessario che i componenti scelti siano intercambiabili con altri prodotti di pari prestazioni non Siemens, questo consente all'azienda di non dipendere troppo da un solo fornitore e gli permette di evitare che un ritardo o un aumento di prezzo di forniture comporti problemi alla produzione.

La ricerca sulla componentistica verrà eseguita tenendo conto che nel nuovo sistema il carico computazionale migrerà dal PLC all'azionamento. In questo sistema, il controllore deve gestire solo le fasi di lavoro mentre la parte di Motion Control (movimento dell'asse e il corretto posizionamento del pezzo) viene totalmente gestita dall'azionamento digitale. E' fondamentale che durante la ricerca della componentistica si abbia sempre una visione chiara del sistema finale da realizzare. Infatti, tutti i dispositivi all'interno del sistema andranno ad interagire tra loro e il più lento di essi determinerà la velocità di esecuzione del macchinario quindi tutti i dispositivi più performanti non verranno sfruttati al massimo e renderanno solo più complesso e costoso il sistema.

D'altra parte, il macchinario dovrà essere leggermente sovradimensionato, in virtù del fatto che è importante mantenere una buona flessibilità del sistema per poterlo facilmente integrare con vari tipi di tornio. La flessibilità è molto importante, anche perché non è detto che in futuro non si provi ad integrare questo sistema con macchinari diversi dal tornio. Realizzare un sistema flessibile aumenta i tempi e i costi di progettazione iniziale perché si devono tenere in considerazione molti più requisiti ma si riducono i costi per gli sviluppi futuri.

### **5.1 Controllori**

La gamma di controllori Siemens è molto vasta e spazia dai PLC a gli industrial PC o PC embedded. I controllori basati su PC sono molto flessibili, nuove tecnologie gli consentono di essere utilizzati nei sistemi IRT, anche se il loro principale vantaggio risiede nella facilità di integrazione con le rete aziendali e la possibilità di eseguire linguaggi di programmazione di alto livello. Queste funzionalità sono però aspetti che non sono richiesti dal sistema. Quindi nella

ricerca dei componenti escluderemo a priori il controllo basato su PC e si presterà attenzione sui PLC.

### **5.1.1 SIMATIC S7-1200**

È l'ultimo PLC sviluppato da Siemens, sostituisce S7-200, ed è il più economico attualmente in listino. Sono disponibili 3 modelli di CPU la più veloce ha un tempo istruzione per operazione a bit di 0,1 us. Le tre CPU sono: 1211C, 1212C e la 1214C. Ad ogni CPU può essere applicata una Signal Board per aumentare il numero di I/O (digitali e o analogici) senza richiedere moduli aggiuntivi. In ogni caso, qualora il numero di ingressi e uscite presenti sulla Signal Board non siano sufficienti possono essere aggiunti moduli esterni (Signal Module) sul lato destro della CPU. Alla CPU 1212C possono essere collegati due Signal Module mentre alla CPU 1214C ne possono essere collegati otto. Inoltre la 1214C ha una capacità di memoria (memoria di lavoro 50k, memoria di caricamento 2MB espandibile con memory card) superiore alla prima CPU di ingresso del S7-300.



*Figura 5.1: S7-1200*

Pur mantenendo un prezzo inferiore.

Il S7-1200 dispone di un'interfaccia industrial ethernet (ProfiNET 10/100Mbit/s) integrata per la comunicazione con i diversi moduli del sistema (HMI, posizionatori etc etc) e per la programmazione del PLC. Sono supportati anche i principali protocolli di comunicazione che si basano su ethernet come il TCP e l'IP. Questo consente di implementare anche web-server sul PLC.

S7-1200 possiede 3 contatori a 100KHz e 3 a 30KHz utilizzabili per il monitoraggio preciso di trasduttori incrementali, conteggi di frequenza e veloce rilevamento di eventi processo. Sono disponibili anche due uscite utilizzabili per PTO (Pulse Train Output) o per PWM (Pulse Wave Modulation) con frequenza di lavoro massima a 100KHz.

### 5.1.2 SIMATIC S7-300

E' il PLC Siemens più venduto e opera nei più svariati settori industriali, ha subito costanti revisioni software e hardware per mantenerlo aggiornato ai nuovi protocolli e alle nuove tecnologie.

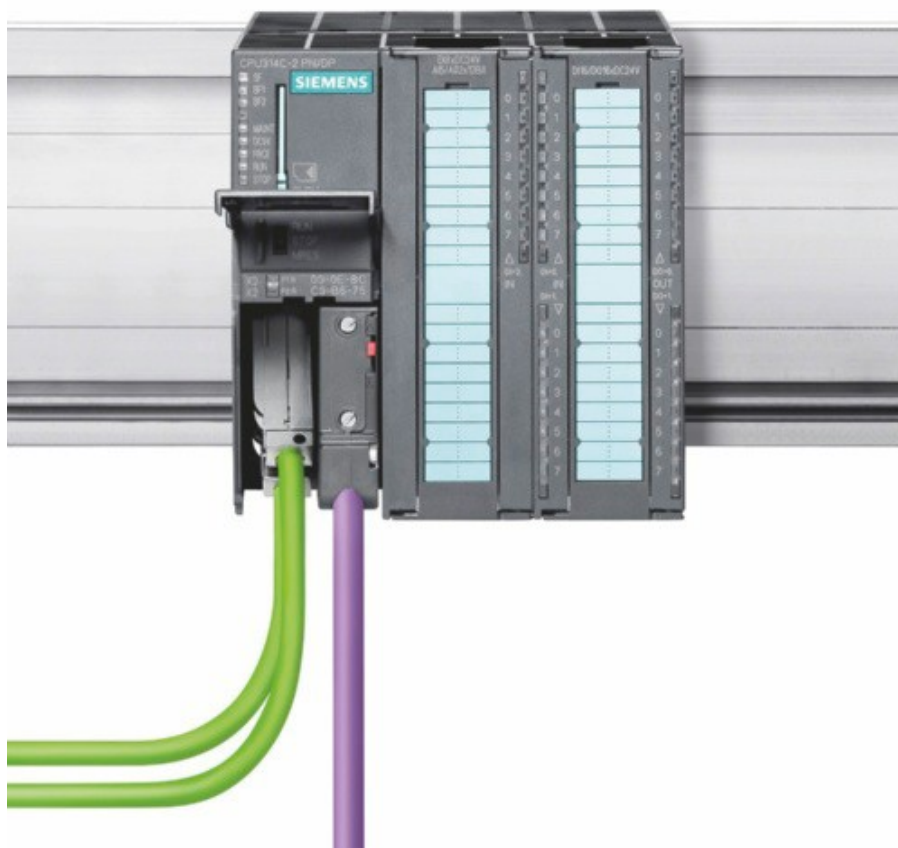


Figura 5.2: S7-300 con una CPU compatta 2 porte ProfiNET e 1 porta ProfiBUS e Signal Board (lato sinistro)

Sono disponibili 22 diverse CPU di cui 6 CPU sono di fail-safe, 2 CPU tecnologiche (e 1 tecnologica con Fail-Safe) sviluppate specialmente per le funzioni di Motion Control. La CPU più veloce (319) ha un tempo istruzione per operazione a bit di 0,004 us, di 0,01us per operazione su Word o a virgola fissa e di 0,04 per operazioni a virgola. Ha una memoria di lavoro di 2 Mbyte espandibile con memory card.

Anche per l'S7-300 ad ogni CPU sono collegabili segnali di I/O tramite la Signal Board, qualora il numero di ingressi e uscite presenti sulla Signal Board non siano sufficienti posso essere aggiunti moduli esterni (Signal Module). In questo modo si possono avere fino a 1024 segnali di I/O direttamente collegati al PLC che possono essere ulteriormente espansi fino a 65536 segnali di I/O tramite moduli di I/O decentrati collegati tramite ProfiBUS o ProfiNET al PLC.

Tutte le CPU utilizzabili sul S7-300 implementano i protocolli di comunicazione: MPI, AS-Interface, ProfiBUS e ProfiNET e le sue funzioni aggiuntive come: MRP (Media Redundancy Protocol) e IRT (Isochronous Real-Time). Inoltre è possibile realizzare web-server con comunicazione HTTP, funzioni di E-Mail client e comunicazione dati FTP.

### **5.1.3 SIMATIC S7-400**

Il S7-400 è il PLC Siemens più performante, è spesso usato per il controllo e il coordinamento di interi processi produttivi e in quei processi che richiedono l'elaborazione di molti dati e tempi di reazione deterministici. Gli ultimi aggiornamenti software e hardware implementano le principali funzioni ProfiNET (MRP Media Redundant Protocoll, comunicazioni dati IRT...) Inoltre implementa la Configuration in Run (CiR) che permette di eseguire modifiche e ampliamenti all'impianto senza interrompere il processo lavorativo.

Anche sul S7-400 sono implementabili Web-Server che consentono di accedere ai dati del PLC da remoto tramite HTTP o FTP o di inviarli tramite eMail.



*Figura 5.3: S7-400*

Sono disponibili diverse CPU in funzione della velocità di calcolo, della memoria di lavoro e della capacità di indirizzamento richiesta. Sono presenti anche CPU fault-tollerant e CPU fail-safe. Il PLC S7-400 implementa un sistema di Multicomping che permette il funzionamento contemporaneo di più

CPU sullo stesso PLC Questo sistema consente di suddividere la potenza di calcolo tra le CPU o di assegnare a ciascuna una determinata funzione. Le CPU effettuano anche il controllo della diagnostica del sistema.

Un'altra particolarità dei processori utilizzati nel S7-400 riguarda l'accesso alla memoria. Grazie ad una particolare architettura a doppio bus queste CPU eseguono l'accesso alla memoria dati e alla memoria del codice in un unico clock.

Esiste una particolare versione del S7-400 chiamata S7-400H particolarmente sviluppata per processi fault-tollerant: quindi per quei sistemi che devono poter continuare a funzionare anche in caso di guasto. L'S7-400H implementa funzionalità per l'utilizzo di comunicazioni, periferie ridondanti. E' possibile, inoltre, avere due S7-400H in parallelo; in caso di guasto del PLC primario, il PLC di riserva acquisisce tutte le funzionalità del primario in meno di 100ms).

## 5.2 Azionamenti

Come per i controllori, la gamma di azionamenti Siemens è molto vasta e spazia da piccoli azionamenti (inverter) a bassa potenza utilizzati per pompe o ventole fino ad azionamenti ad alta tensione con potenze da 120MW. Escludendo dalla ricerca gli azionamenti ad alta tensione e quelli DC, riduciamo la scelta ai prodotti in tabella.

<i>For Basic Application</i>	<i>For demanding application</i>		<i>For basic servo application</i>	<i>For sophisticated application</i>	
<b>G110</b>	<b>G120</b>	<b>G130/G150</b>	<b>S110</b>	<b>S120</b>	<b>S150</b>
0,12-3kW	0,37-250kW	75kW-2,7MW	0,12-90kW	0,12kW-4,5MW	75kW-1,2MW
Pumps, fan, Conveyor belt	Pumps, fan, Conveyor belt, compressors	Pumps, fan, Conveyor belt, compressors	Single-axis positioning	Production machines, process lines	Test stands, centrifuges

Gli azionamenti possono implementare un ampio raggio di protocolli di comunicazione, ProfiNET, ProfiBUS, AS-Interface, USS, CANOpen, Modbus, BacNet. Ogni azionamento implementa uno o più di questi standard a seconda del campo applicativo per il quale è stato progettato.

Negli azionamenti SINAMICS sono implementate anche diverse funzioni di sicurezza integrata, che si possono ricondurre in due classi:

- Funzioni di messa in stop in sicurezza dell'azionamento (senza

necessariamente disconnettere l'alimentazione). Assicurano che il motore sia fermo e frenato.

- Funzioni di controllo in modo sicuro della velocità di un'azionamento. Evitano che il motore raggiunga velocità troppo elevate.

Si ricorda, inoltre, che alcuni fieldbus di nuova generazione possono essere utilizzati per le procedure di fail-safe senza necessitare di bus ridondanti. (Vedi Cap: 3.1.1 Safety a pag. 19).

### **5.2.1 Serie G**

Analizzando le funzionalità degli azionamenti SINAMICS si nota che la serie G sono azionamenti poco sofisticati e sono utilizzati in ambiti di lavoro con movimenti continui e dove non sono richieste particolari precisioni. Come pompe, ventole, compressori o nastri trasportatori. Possono pilotare motori asincroni e solo per i modelli G130/G150 è possibile pilotare dei motori sincroni senza encoder. I modelli G110 utilizzano una comunicazione seriale, mentre il G120, il G130 e il G150 implementano diversi standard di comunicazione tra cui ProfiNET, e ProfiBUS. Tutti i modelli della serie G sono singolo asse tranne il G120D che può pilotare più assi. Le alimentazioni vanno da 220V fino a 960V per i modelli di taglia maggiore.



*Figura 5.4: SINAMICS G120*

### **5.2.2 Serie S**

I convertitori SINAMICS serie S sono progettati per campi applicativi ingegneristico meccanici complessi, nonché per una vasta gamma di compiti di motion control. Alcuni esempi applicativi possono essere: movimento delle linee di produzione, posizionamento pezzi, centrifughe o banco di prova. La serie S,



rispetto alla serie G, è sviluppata per pilotare dei motori in ambiti dove sono richieste precisioni maggiori. Le potenze coprono un ampio range e vanno da 0,12kW a 4500kW. Possono pilotare motori asincroni, sincroni; l'S120 può pilotare anche motori lineari. Implementano diversi protocolli di comunicazione compreso ProfiNET e ProfiBUS. I modelli della serie S (tranne l'S110) supportano anche il profilo applicativo ProfiSafe. (Vedi il Cap. 4.3.3 Profili applicativi a pag 46). L'S120 è l'unico azionamento della serie S multi asse. Le alimentazioni vanno da 220V fino a 690V per i modelli di taglia maggiore.

*“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”*

## 6 IMPLEMENTAZIONE DEL NUOVO SISTEMA

L'obiettivo dell'azienda è quello di realizzare un nuovo sistema BOSS più flessibile del precedente, sostituendo l'azionamento analogico con un azionamento digitale e il PLC con uno meno performante. Si è scelto di collegare i dispositivi presenti nel BOSS con il protocollo di comunicazione industrial ethernet ProfiNET e di sviluppare un web-server per poter interagire con la macchina da remoto. Questa tesi ha lo scopo di individuare un azionamento ed un PLC che rispondano ai requisiti dell'azienda e di verificare il corretto funzionamento del nuovo sistema realizzando un prototipo che simuli le principali funzioni del caricatore BOSS.

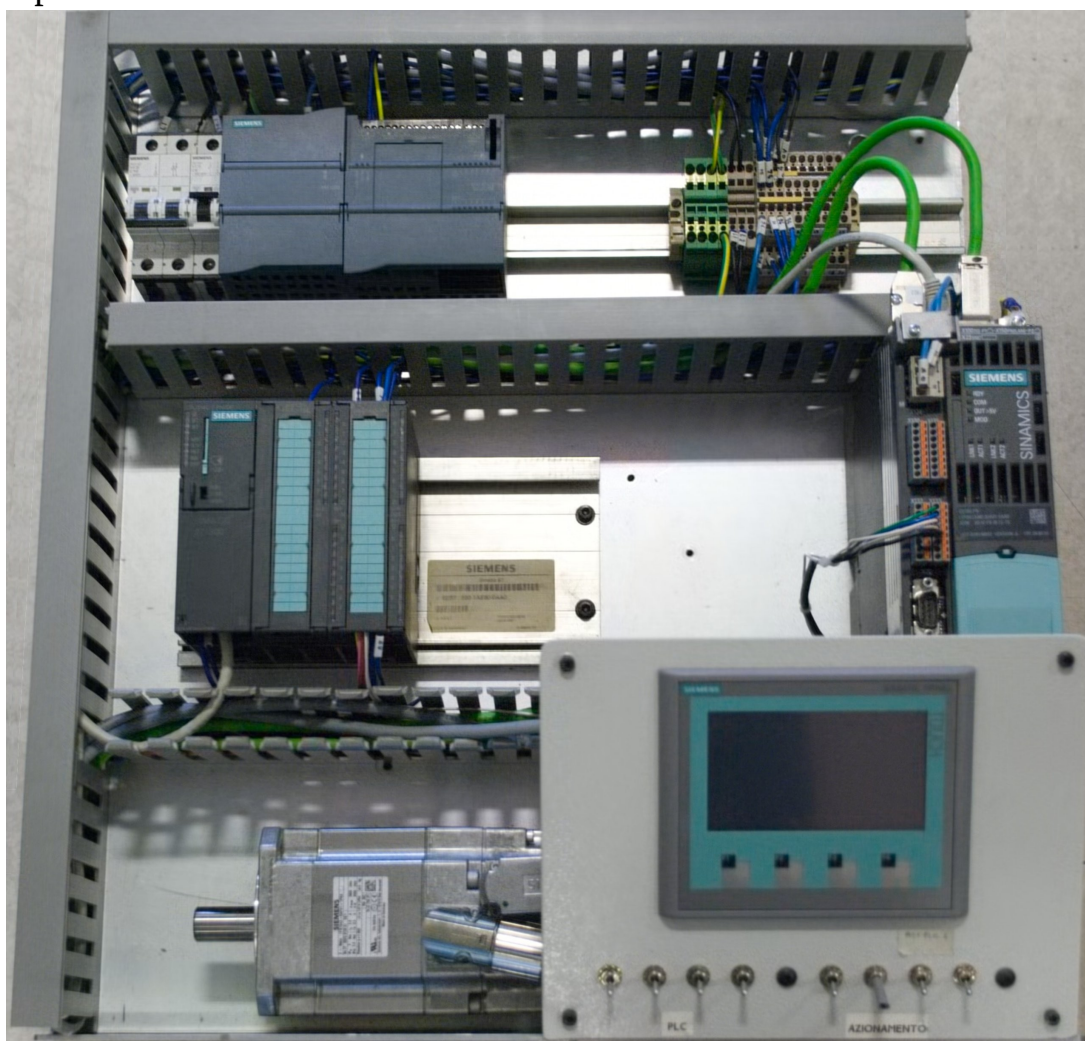


Figura 6.1: Prototipo simulatore realizzato

## 6.1 Componenti scelti

**Azionamento:** la gamma SINAMICS di Siemens è molto vasta ma gli unici azionamenti che possono pilotare motori sincroni con una buona precisione sono gli azionamenti della serie S. All'interno di questa serie il modello con la taglia di potenze più vicino alle esigenze IEMCA è il modello S110. Anche il modello S120 presenta versioni di piccola taglia ma essendo un multi-asse è stato scartato.



Figura 6.2: SINAMICS S110 e motore sincrono modello 1FK7

- 1) Control Unit CU305
- 2) X22, interfaccia seriale RS232
- 3) Pannello Operatore BOP20, non montato nel simulatore
- 4) Power Module 340, parte di potenza che comanda il motore
- 5) Interfacce X100 (DRIVE-CLiQ), X126 due porte (ProfiNET), X23 (encoder)
- 6) Motore con resolver integrato digitale a 14bit e interfaccia (DRIVE-CLiQ)
- 7) Collegamento di alimentazione del motore
- 8) Collegamento DRIVE-CLiQ del motore

Non è escludibile a priori che il BOSS possa funzionare correttamente anche con un azionamento SINAMICS serie G accoppiato ad un motore asincrono. Ma per verificare se questo “downgrade” del sistema non peggiora le prestazioni finali

occorre un'ulteriore rivoluzione del sistema BOSS. IEMCA si è imposta di effettuare uno step intermedio, fermandosi alla sostituzione del azionamento analogico con uno digitale senza andare a modificare il motore del BOSS. Riservandosi di valutare solo in un secondo tempo anche questo ulteriore sviluppo.

L'azionamento S110 ha come interfaccia tra la parte di potenze e i comandi esterni una controll unit modello CU305, che presenta due porte ProfiNET e 4 morsettiere da 8 poli per segnali esterni (I/O digitali e analogici). (Fig. 6.2)

**Controllore:** il sistema BOSS attuale utilizza un PLC S7-300. Questo controllore offre una elevata capacità di calcolo che era necessaria per il tipo di schema implementato nel BOSS. Nel nuovo sistema l'azionamento digitale esegue da solo la parte di motion control e riduce il carico di lavoro del PLC. L'S7-1200 ha una buona memoria e implementa ProfiNET ma ha una capacità di calcolo inferiore del S7-300. Inizialmente si è scelto di realizzare il prototipo simulatore con l'azionamento S110 comandato dal PLC S7-314C che è un controllore stabile e ben conosciuto all'interno dell'ufficio tecnico. Successivamente, dopo che si sarà ottenuto un software che implementa le principali funzionalità del BOSS il progetto verrà migrato sul nuovo PLC S7-1212C. Questa scelta consentirà all'azienda di poter produrre in futuro il BOSS sia con il PLC S7-1200 che con il più potente S7-300 a seconda delle esigenze dei clienti.

## 6.2 Schema del sistema

Il PLC S7-300 ha due porte ProfiNET che possono funzionare come switch di rete, mentre il PLC S7-1200 ha una sola porta ProfiNET; quindi per realizzare rete di comunicazione tra i dispositivi univoca, indipendentemente dal PLC adottato, si è scelto di usare lo switch presente nell'azionamento S110.

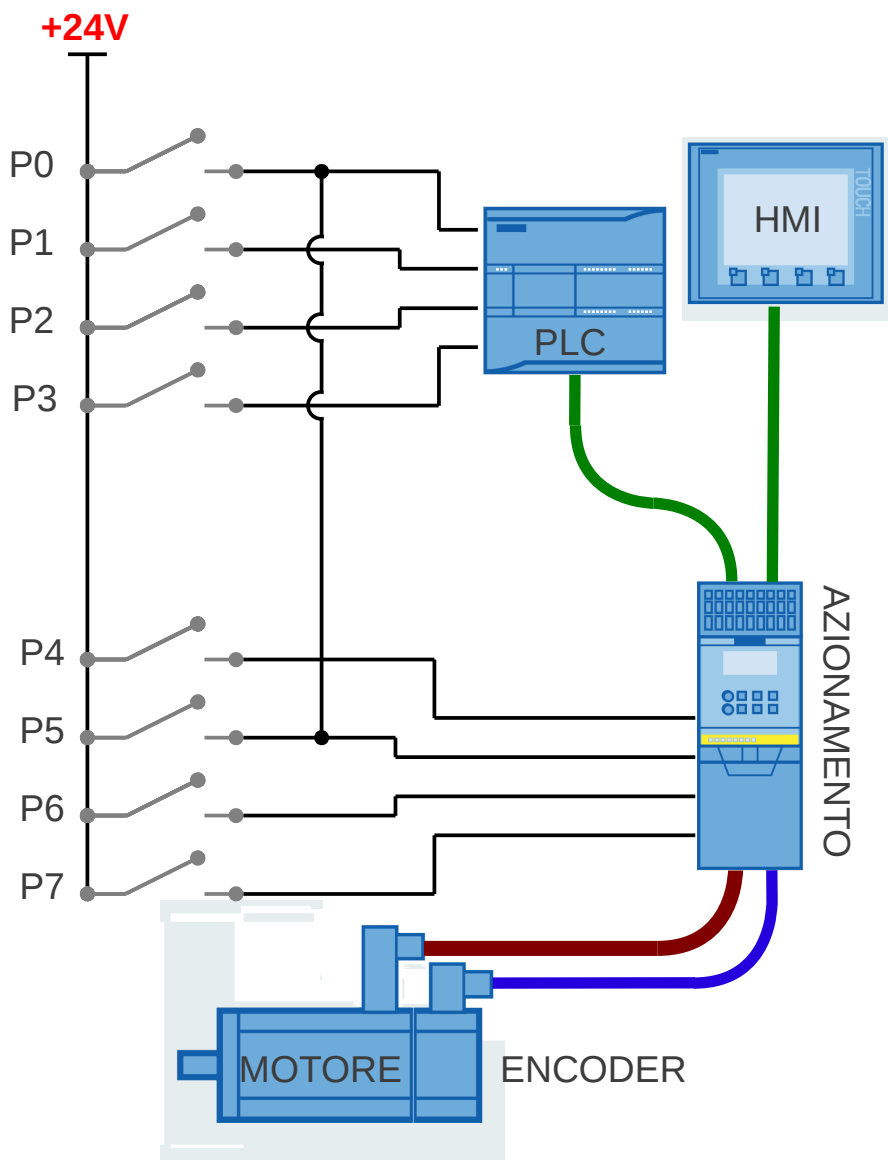


Figura 6.3: Schema del simulatore

NOTE sullo Schema in figura 6.3:

**Componenti:**

*PLC: SIMATIC S7-1212C o S7-314C.*

*HMI: TP 177B 4” Touch Screen.*

*Azionamento: S110 con Control Unit CU305.*

*Motore: Sincrono - 1FK7 Compat – 0,82kW // 3,0Nm.*

*Encoder: Resolver 2-poli analogico convertito in digitale 14 bit [ DRIVE-CLiQ ]*

**Collegamenti:**

*In verde il collegamento ProfiNET (usa lo switch dell'azionamento).*

*In rosso il collegamento di potenza con il motore sincrono.*

*In blu il collegamento con l'encoder integrato nel motore.*

*P0 - Collegato a P5 e all'ingresso PLC – I0.0.*

*P1 - Collegato all'ingresso PLC – I0.1.*

*P2 - Collegato all'ingresso PLC – I0.2.*

*P3 - Collegato all'ingresso PLC – I0.3.*

*P4 – Collegato alla morsettiera X133 dell'azionamento – Non utilizzato*

*P5 – Collegato alla morsettiera X133 dell'azionamento – Reset dell'azionamento.*

*P6 – Collegato alla morsettiera X133 dell'azionamento – Sensore zero asse.*

*P7 – Collegato alla morsettiera X133 dell'azionamento – Non utilizzato*

*Per ulteriori informazioni sui componenti utilizzati si faccia riferimento all' Appendice A:  
LISTA COMPONENTI ACQUISTATI a pag 104*

Per interagire con il sistema e per simulare alcuni dei sensori del BOSS sono stati inseriti degli interruttori (P0 ... P7). I primi quattro sono collegati a gli ingressi del PLC, i restanti sono collegati all'azionamento.

### **6.3 Comunicazione tra PLC e S110**

La comunicazione tra l'azionamento S110 e il PLC avviene tramite il profilo applicativo PROFIdrive messo a disposizione da ProfiNET (è lo stesso profilo disponibile anche in ProfiBUS) (vedi cap: 4.3.3 e 4.4.5 alle pagg. 46 e 59). Il PLC ha il ruolo di IO-Controller mentre l'azionamento prende il ruolo di IO-Device (o Drive-unit). La comunicazione in PROFIdrive può essere ciclica o no e avviene attraverso dei telegrammi.

L'S110 ha a disposizione 12 telegrammi standard definiti da PROFIdrive, 18 telegrammi specifici del costruttore e 1 telegramma libero. Ogni telegramma è composto da delle *parole di comando* (inviate dal PLC all'azionamento) e da delle *parole di stato* (risposte dell'azionamento inviate al PLC), chiamate indistintamente da Siemens PZD (Prozessdaten – Dati di processo).

Nel simulatore realizzato si utilizza una comunicazione ciclica con il telegramma 110 modificato. Infatti il telegramma 110 è l'unico telegramma disponibile che consente di pilotare un azionamento in MDI (Manual Data Input) e quindi permette di impostare i parametri di velocità e accelerazione del movimento.

Il telegramma 110 originale è composto da 12 parole di comando e 7 parole di stato. E' possibile ampliare il numero delle parole (trasformando il telegramma 110 in un telegramma libero) aggiungendo, pacchetti composti da 2 parole di comando e 2 di stato. Con il limite di non superare 32 PZD in un telegramma. A questo telegramma sono state aggiunte due parole di comando *CPRif* e *MOMRED*. Il PLC utilizza queste parole aggiuntive per la funzione di Preset e per limitare la coppia del motore durante quei movimenti contro un fermo meccanico (vedi cap. 2.1.1 Fasi di lavoro a pag. 12). Per poter inserire queste due PZD sono stati aggiunti due pacchetti da due parole ciascuno. Il telegramma ottenuto è riportato in tabella.



Off set [Bit]	<b>Parola di stato</b>	Off set [Bit]	<b>Parola di comando</b>
0	<u>ZSW1</u> : Parola di stato 1	0	<u>STW1</u> : Parola di comando 1
16	<u>AKTSATZ</u> : Blocco di movimento attivo	16	<u>SATZANW</u> : Scelta blocco di movimento
32	<u>Pos_ZSW</u> : Stato del funzionamento	32	<u>Pos_STW</u> : Comando del funzionamento
48	<u>ZSW2</u> : Parola di stato 2	48	<u>STW2</u> : Parola di comando 2
64	<u>MELDW</u> : Parola di segnalazione	64	<u>Over</u> : Override della velocità
80	<u>XistP</u> : indica la posizione (32 bit)	80	<u>MDIPos</u> : Posizione MDI (32 bit)
96		96	
		112	<u>MDIVel</u> : Velocità MDI (32 bit)
		128	
		144	<u>MDIAcc</u> : Accelerazione MDI
		160	<u>MDIDec</u> : Decelerazione MDI
		176	<u>MDIMode</u> : Mode MDI
112	<b>Non utilizzati</b>	192	<u>MOMRED</u> : Coppia MDI
128	<b>Non utilizzati</b>	208	<u>CPRif</u> : Preset Encoder (32 bit)
144	<b>Non utilizzati</b>	224	
160	<b>Non utilizzati</b>	240	<b>Non utilizzati</b>

Telegramma utilizzato nella comunicazione tra l'azionamento e il PLC

In grigio le PZD aggiunte al telegramma originale.

Per una descrizione accurata delle singole parole e del significato dei loro bit si rimanda al manuale del SINAMICS S110.

## 6.4 Stati di funzionamento realizzati e standardizzati

Nel cap. 2.1.1 (pag. 12) sono descritte le fasi di lavoro di un caricatore barre BOSS e alcune di queste prevedono uno o più movimenti dell'asse. In questa tesi sono stati eseguiti tutti questi singoli movimenti con l'S110 per assicurarsi che poi questo azionamento possa rispondere ai requisiti richiesti di un caricatore barre per torni monomandrini.

In un caricatore monomandrino tipo BOSS, l'azionamento è chiamato in causa

durante 14 stati di funzionamento.

STATI del BOSS		
Accensione		1
Startup		2
MANUALE	Preset	3
	Zero Asse	4
	Asse Fermo	5
	Movimento JOG avanti	6
	Movimento JOG indietro	7
AUTOMATICO	Zero Asse	8
	Asse Fermo in coppia	9
	Asse Fermo non in coppia	10
	Movimento Assoluto a posizione	11
	Movimento Relativo a posizione	12
	Movimento Avanti a battuta JOG	13
	Movimento Indietro a battuta JOG	14

Per ognuna di queste condizioni di lavoro sono stati trascritti i valori delle parole di stato e di comando che il PLC riceve e invia al S110. Grazie a queste trascrizioni è stato poi redatto un documento nel quale sono descritti tutti i singoli bit di tutte le parole di comando e di stato in funzione della azione svolta dal S110 (Vedi Appendice B: MOVIMENTI STANDARDIZZATI a pag 105). Questo consente di standardizzare in maniera chiara e univoca come viene gestito un azionamento digitale in un BOSS IMECA.

Questa “carta d'identità” realizzata ha una duplice funzione. In caso di guasto di un BOSS, l'assistenza clienti, confrontando la comunicazione ProfiNET nel macchinario guasto e il documento potrà individuare rapidamente se il guasto riguarda il PLC o l'azionamento. Realizzando riparazioni e sostituzioni più veloci e mirate.

In secondo luogo, IEMCA non è intenzionata a legarsi ad un solo fornitore di azionamenti, quindi in futuro verranno costruiti BOSS con azionamenti realizzati da diversi produttori. Questo documento consentirà ai produttori di azionamenti di verificare se il loro prodotto è pienamente compatibile con il tipo di comunicazione che si è instaurata tra il PLC e l'S110.

In teoria non sarebbe necessario documentare la comunicazione tra il PLC e l'azionamento in quanto avviene secondo il profilo applicativo PROFIdrive di ProfiNET certificato della Profibus & ProfiNET International (PI) (vedi cap.

4.4.8 Certificazione dei prodotti a pag 61). Capita troppo spesso, però, che i protocolli nel campo dell'automazione siano blandi e che due prodotti standard non siano pienamente compatibili tra loro. Quindi, per prevenire questi problemi che si è scelto comunque di redarre questo documento.

### 6.4.1 Descrizioni degli stati di funzionamento

Per le informazioni sui bit che compongono le PZD si faccia riferimento all'Appendice B: MOVIMENTI STANDARDIZZATI.

1. **Accensione:** questo stato ha inizio dopo l'alimentazione del sistema. L'azionamento risponde con alcuni bit delle parole di stato attivi, ma non gli vengono inviati comandi.
2. **Startup:** ha inizio quando viene premuta la marcia nel BOSS. L'azionamento è fermo e non è stato impostato alcun riferimento dell'asse. In questa fase vengono attivati i bit dal [32.5] al [32.1] della parola di comando **STW1** che servono per abilitare l'azionamento.

Dalla premuta del pulsante marcia alla messa in automatico (o manuale) del sistema devono passare 2 secondi. Dopo lo stato di startup, l'azionamento ha bisogno di essere riferito (Preset o zero asse), il BOSS si riferisce in modalità manuale dove all'avvio effettua il preset del encoder.

3. **Manuale – Preset:** si informa il sistema che il carrello è all'ultima posizione rilevata prima dello spegnimento della macchina. Attivando il bit **Imp\_punto\_riferimento** della parola di comando **Pos\_STW** l'azionamento imposta il valore dell'encoder al valore inserito nella parola di stato **CPRif**. All'accensione **CPRif** contiene il valore del ultima posizione registrata prima dello spegnimento.
4. **Manuale – Zero asse:** la macchina effettua l'azzeramento dell'asse. Quando viene settato il bit **Avvio\_ric\_riferimento** della parola di comando **STW1** l'azionamento esegue la procedura di zero asse per il quale è stato programmato. La procedura di azzeramento dell'asse utilizzata prevede di fare tornare indietro velocemente il carrello fino al fronte di salita del sensore di Zero Asse. A questo punto il carrello avanza piano fino al fronte di discesa del sensore, poi si ferma e l'azionamento utilizza questa posizione come zero. (Vedi Fig. 6.4) Più precisamente imposta il valore dell'encoder al valore inserito nella parola di stato **CPRif** che è posta a “0”.

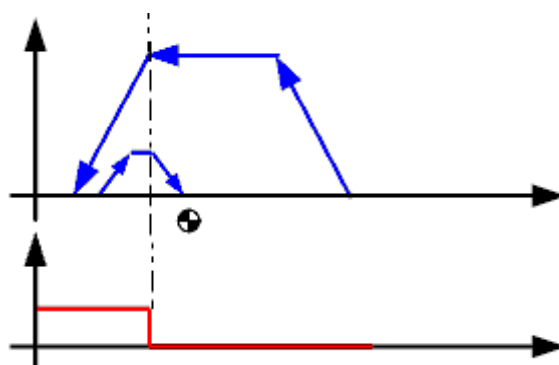


Figura 6.4: Movimento eseguito per l'azzeramento dell'asse.  
In **blu**: i trapezi del movimento  
In **rosso**: il grafico del sensore di zero

5. **Manuale – asse fermo**: L'azionamento è fermo e dato che è attivo il bit **ON\_OFF1** della parola di comando **STW1**, il motore è in coppia e non può muoversi.

6. **Manuale – Movimento JOG avanti**: Il carrello si muove in avanti fino al raggiungimento di una battuta meccanica.

Il movimento avanti termina quando il bit [1.6] **Lim\_coppia\_pos\_ragg** della parola di stato **ZSW1** è attivo. Oppure il movimento termina al rilascio manuale del bit [33.1] **JOG2** della parola di comando **STW1**.

7. **Manuale – Movimento JOG indietro**: Il carrello si muove in avanti fino al raggiungimento di una battuta meccanica.

Il movimento indietro termina quando il bit [1.7] **Lim\_coppia\_neg\_ragg** della parola di stato **ZSW1** è attivo. Oppure il movimento termina al rilascio manuale del bit [33.0] **JOG1** della parola di comando **STW1**.

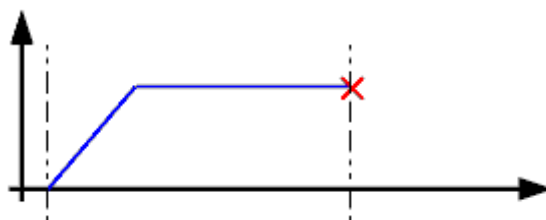


Figura 6.5: Movimento di JOG in avanti

Si passa a questo punto ai possibili stati di funzionamento in modalità automatica.

8. **Automatico – Zero asse:** è identico all'azzeramento dell'asse in manuale. Infatti, l'azionamento non distingue se il BOSS è in modalità manuale o no.
9. **Automatico – asse fermo in coppia:** è uguale allo stato Asse fermo in manuale, con la differenza che qui è attivo anche l'allarme **Monitoraggio\_di\_fermo** della parola di comando **STW1**.
10. **Automatico – asse fermo e libero:** il motore non è in coppia, quindi il motore è libero di muoversi. Occorre disabilitare il bit **ON\_OFF1** (così il motore non è più in coppia) e **Monitoraggio\_di\_fermo** di **STW1** inoltre è necessario che l'azionamento continui a monitorare l'encoder per memorizzare sempre la posizione del carrello, per fare questo si abilita **Funzionamento\_a\_seguire** di **Pos\_STW**.

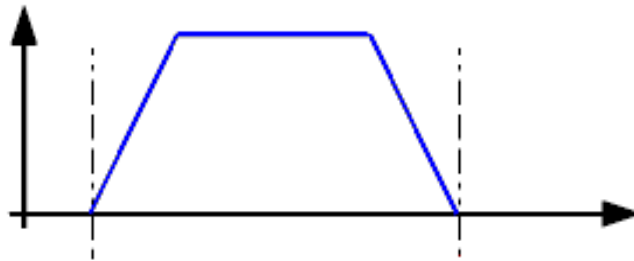


Figura 6.6: Movimento a posizione

11. **Automatico – Movimento Assoluto a posizioni:** l'azionamento si muove fino a raggiungere la posizione indicata nella parola di comando **MDIPos**. **MDIMode** è posta a “0”.
12. **Automatico – Movimento Relativo a posizioni:** l'azionamento si muove di una lunghezza pari a **MDIPos**. **MDIMode** è posto a “10”.
13. **Automatico – Movimento avanti a battuta (JOG):** E' identico al movimento manuale di jog in avanti.
14. **Automatico – Movimento indietro a battuta (JOG):** E' identico al movimento manuale di jog in indietro.

## 6.5 Struttura del software PLC

Il programma è sviluppato in linguaggio Ladder Diagram. Questo permette di istruire velocemente gli installatori IEMCA che devono interfacciare la macchina con il tornio, in quanto ladder diagram è un linguaggio semplice e

facilmente intuibile anche da non esperti di linguaggi di programmazione.

Il programma è strutturato in blocchi (Fig. 6.7), il blocco organizzativo **OB1** è il “main” del programma e lancia le principali funzioni (FC). Ci sono altri OB (es: OB100,...) che vengono lanciati in particolari momenti, come in caso di interrupt hardware, di guasto, all'avvio ecc. ecc. Ogni blocco è stato realizzato per eseguire determinate funzioni e interagisce con gli altri blocchi attraverso i DB o tramite celle di memoria. Nessun blocco esegue delle funzioni che sono implementate anche da un altro blocco. Queste regole rigide servono per assicurare uniformità tra i progetti IEMCA e per massimizzare la flessibilità del progetto.

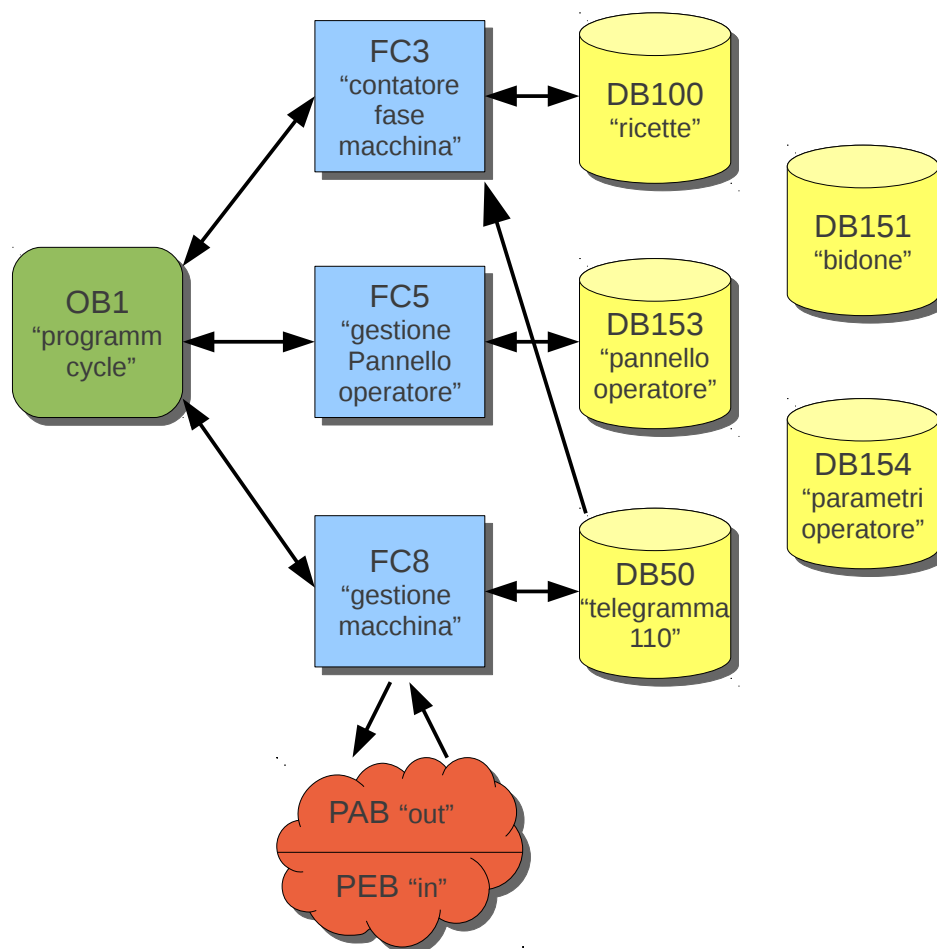


Figura 6.7: Struttura del software

La funzione **FC3** scandisce le fasi della macchina, si appoggia al Database **DB100** che contiene le “ricette” indicanti i parametri di movimento di ogni fase. **FC3** durante le fasi che prevedono un movimento asse, preleva la relativa ricetta dal **DB100** e la mette a disposizione per **FC8** che dovrà inviare i dati

all'azionamento. Mentre per le fasi che non richiedono un movimento dell'asse, gestisce gli I/O della relativa fase (eventualmente richiamando una sotto-funzione). Ovviamente il numero delle fasi e le loro funzioni sono rimaste invariate rispetto al BOSS con l'azionamento analogico (Vedi cap. 2.1.1 Fasi di lavoro a pag. 12).

La funzione **FC5** gestisce i segnali che arrivano dal pannello operatore (marcia, arresto, etc etc). Il passaggio di dati dal pannello operatore e FC5 avviene tramite la scrittura/lettura di dati presso il DB153. FC5 si occupa anche che l'operatore avvii la macchina con la giusta sequenza. Ad esempio non è possibile mettere la macchina in automatico prima di aver effettuato lo zero asse.

La funzione **FC8** gestisce le operazioni della macchina. Legge i dati in arrivo dall'azionamento (area di memoria PEB) e li salva nel DB50 mettendoli a disposizione degli altri FC che ne fanno richiesta. A seconda del lavoro che la macchina deve svolgere scrive i dati (incluso le ricette del movimento) da inviare all'azionamento nel DB50 e poi li spedisce (area di memoria PAB).

Oltre ai DB già descritti sono presenti il DB151 “bidone” utilizzato per memorizzare generiche variabili del sistema e il DB154 “parametri utente” che contiene i parametri principali che servono per implementare le funzioni specifiche richieste dall'utente (Esempio: quota B1C, il tipo di avanzamento del pezzo, il tipo di espulsione etc etc). Questi DB sono utilizzati dai diversi blocchi funzione.

E' importante notare che le funzioni FC3, FC5, FC8 ed i relativi DB svolgono gli stessi compiti che avevano nel sistema BOSS con azionamento analogico (ovviamente in quel caso non è presente il DB50). Stesso discorso anche per i parametri settabili dagli utenti e i nomi di alcune variabili. Questa scelta assicura che ci sia la massima uniformità con i progetti passati ed evita incoerenze che possono portare ad errori in fase di interfacciamento del BOSS con i torni.

## **6.6 Migrazione del software**

Il progetto inizialmente sviluppato sul PLC S7-300 con CPU 314C viene successivamente migrato sul PLC S7-1200 con CPU 1212C.

Questi due PLC, oltre alle differenze in termini di prestazioni tra le CPU S7-314C e l'S7-1212C presentano anche differenze sulla gestione del programma e sulla quantità della memoria.

### **6.6.1 Gestione della memoria**

I PLC hanno tre tipi di memoria:

- **Memoria di caricamento:** è non volatile e viene utilizzata per salvare il programma utente, i dati e la configurazione. Per entrambe le CPU scelte è

espandibile con una Memory Card Siemens MMC.

- **Memoria di lavoro:** è volatile e viene utilizzata per salvare alcuni elementi del progetto durante l'esecuzione del programma utente. La CPU copia alcuni elementi del progetto dalla memoria di caricamento in quella di lavoro.
- **Memoria a ritenzione:** è non volatile e viene utilizzata per salvare i valori di locazioni di memoria utente selezionate in caso di mancanza di alimentazione. In caso di spegnimento o interruzione dell'alimentazione la CPU ripristina i valori a ritenzione quando viene accesa.

PLC	Mem di caricamento	Mem di lavoro	Mem a ritenzione
S7-1212C	1 MByte (Espandibile MMC, Max 24MB)	25 KByte	2 KByte
S7-314C	Dipende dalla MMC utilizzata, Max 8MB	192 KByte	64 KByte

*Capacità della memoria dei due PLC utilizzati*

Il programma realizzato sul S7-314C è memorizzato su una memory card da 64KByte quindi la memoria di caricamento integrata nel S7-1212C è di molto superiore ai requisiti. Il PLC serie 300 ha una memoria di lavoro molto più capiente del 1200 perché è un PLC con una elevata potenza di calcolo e quindi richiede più memoria di lavoro per poter essere veloce nelle operazioni. Nel nostro sistema, il PLC non deve svolgere operazioni complesse, quindi anche la bassa capienza della memoria di lavoro del S7-1212C non rappresenta un limite nella migrazione del software. Un discorso a parte deve essere fatto sulla memoria a ritenzione.

Il PLC S7-300 memorizza i DB e i valori salvati in essi nella memoria di caricamento, il S7-1200 memorizza i DB nella memoria di caricamento ma non i valori salvati in essi. Questo implica che per memorizzare un valore in un database è necessario che tutto il DB venga salvato anche nella memoria a ritenzione.

I database utilizzati in questo programma sono stati copiati da progetti precedenti per mantenere la massima compatibilità; in passato però non essendoci il problema del limite della memoria a ritenzione i database venivano sovradimensionati per non dover essere ampliati in futuro.

Nel progetto realizzato vengono salvati nella memoria a ritenzione il DB151 e il DB154 che essendo sovradimensionati portano ad un'occupazione del 49% della memoria. Quindi, attualmente la memoria a ritenzione non rappresenta un limite, qualora lo fosse per sviluppi futuri si potrebbe utilizzare una CPU 1214 che ha il doppio della memoria a ritenzione o si potrebbe strutturare in maniera



più ordinata i DB.

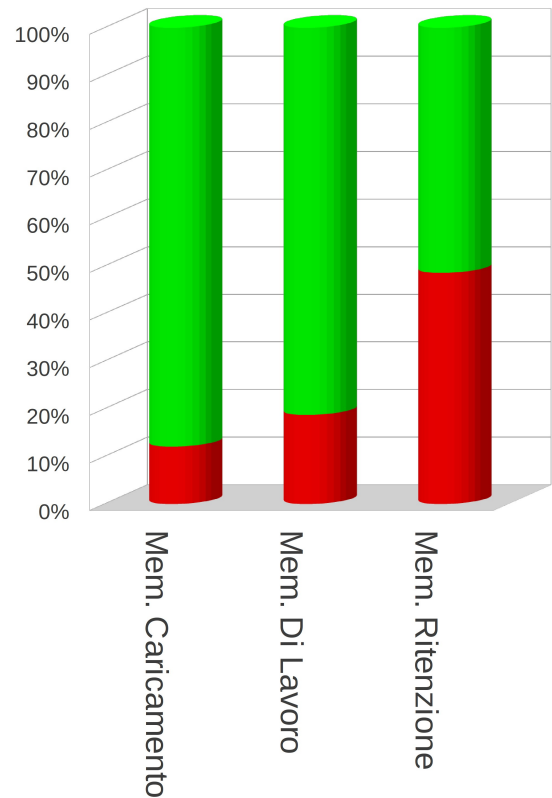
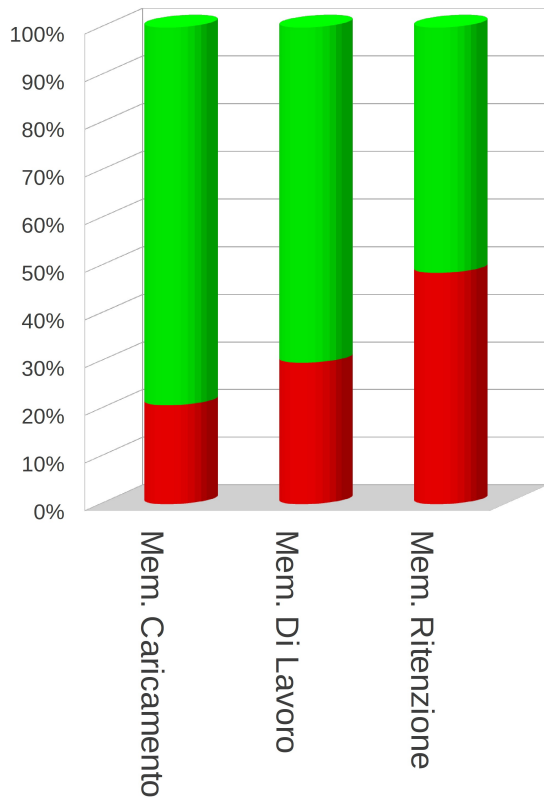
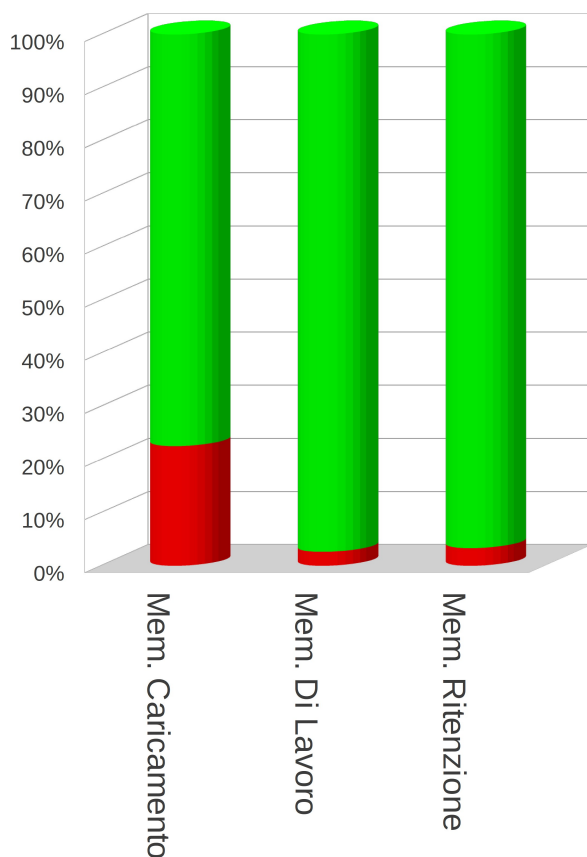


Figura 6.8: Utilizzo delle memoria nel PLC S7-1212C con Web-Server  
In **rosso** la percentuale di memoria occupata

Figura 6.9: Utilizzo delle memoria nel PLC S7-1212C Senza Web-Server  
In **rosso** la percentuale di memoria occupata

In figura 6.8, 6.9 e 6.10 si possono notare l'elevata differenza nell'utilizzo della memoria a ritenzione tra il PLC S7-300 e il S7-1200.



*Figura 6.10: Utilizzo delle memoria nel PLC S7-314C Senza Web-Server  
In rosso la percentuale di memoria occupata*

Nel PLC S7-1200 c'è stata anche una semplificazione nell'uso dei OB lanciati in caso di guasto hardware. Per l'S7-300 avevamo: OB83 - I/O point fault, OB86 - Loss of rack fault e OB122 - Module access errore. Tutti questi blocchi operativi nel S7-1200 sono stati integrati in un unico blocco chiamato: OB82 - Diagnostic error input.

Per quanto riguarda la gestione del software ci sono alcune differenze tra la serie 300 e la serie 1200.

S7-1200 associa ad ogni contattore e ad ogni temporizzatore un DB contenente i dati necessari per gestire la funzione, mentre nel S7-300 questi dati erano normali variabili (eventualmente salvabili nella memoria di ritenzione).

Durante la migrazione del software dal S7-314C al S7-1212C è stato necessario creare un DB per ogni contattore e temporizzatore. Questa conversione è avvenuta con successo e a parte richiedere del tempo non ha presentato particolari problemi se non quelli legati alla compatibilità dei nomi. Infatti,

come già accennato nel capitolo 6.5, tutti i blocchi funzionali nei progetti IEMCA devono mantenere una uniformità. Quindi si è scelto di dedicare ai DB riguardanti i temporizzatori e i contatori i nomi dal DB500 in su. Onde evitare di creare nuovi DB con lo stesso nome di un DB presente in un progetto più vecchio.

Esiste un'ulteriore differenza riguardante la gestione dei DB tra il PLC S7-300 e il S7-1200. Le variabili nei database, prima dei PLC serie 1200, erano accessibili tramite l'indirizzamento simbolico (con il nome della variabile) e o assoluto (con l'indirizzo fisico della variabile). Per ottimizzare gli spazi nella memoria, l'S7-1200, può frammentare il DB e quindi le variabili in esso contenute non sono più accessibili tramite l'indirizzo assoluto perché questo non è più noto al programmatore. Nei progetti finora utilizzati in IEMCA si è sempre fatto largo uso dell'indirizzamento assoluto e questa nuova gestione dei DB da parte dei nuovi PLC Siemens può rappresentare un inconveniente nella realizzazione di nuovi programmi. La possibilità di frammentare il DB è comunque opzionale e quindi si può escludere per mantenere la compatibilità con i DB precedenti.

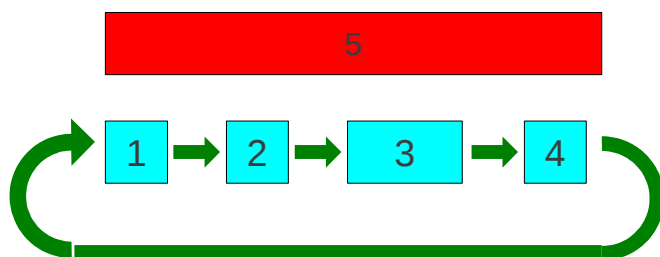
### 6.6.2 Utilizzo della CPU

Il PLC S7-300 è un PLC con una potenza di calcolo nettamente superiore al PLC S7-1200.

<b>Tempi di elaborazione della CPU</b>		
	S7-1212C	S7-314C
<i>Tipo di operazione</i>	<i>Tempo per operazione</i>	<i>Tempo per operazione</i>
Su bit (min)	0,1 us	0,06 us
Su Word (min)	12 us	0,12 us
A virgola mobile (min)	18 us	0,59 us

Questa differenza nelle prestazioni si traduce in una notevole differenza sui tempi di ciclo della PLC. Questi tempi sono visibili nella pagina di diagnostica del PLC con il collegamento attivato tra il PLC e il PC. Il progetto realizzato con S7-1212C ha un tempo di ciclo medio attorno ai 9 ms (tempo minimo 8ms, massimo 13ms) contro un tempo di ciclo medio di 1 ms per S7-314C (tempo minimo 0ms, massimo 3ms).

Nel progetto con S7-1212C sono stati misurati anche i tempi di ciclo mentre il PC interrogava il Web-Server. Ma non sono state visualizzate particolari differenze rispetto al tempo di ciclo medio senza il web-server attivo.



*Figura 6.11: Gestione di un ciclo di un PLC Siemens*

- 1 – Scrittura delle uscite*
- 2 – Lettura degli ingressi*
- 3 – Esecuzione degli OB di ciclo*
- 4 – Esecuzione dell'autotest*
- 5 – Elaborazione degli allarmi in qualunque punto del ciclo di scansione*

Occorre precisare che queste differenze nei tempi di ciclo non comportano una particolare riduzione delle prestazioni nel BOSS. Nel nuovo sistema il motion del motore è gestito totalmente dall'azionamento. I movimenti eseguiti, sia a posizione che a battuta, sono controllati dall'azionamento, la loro velocità di esecuzione e la loro precisione nel posizionamento non dipende dai tempi di ciclo del PLC.

Mentre la durata di un cambio barra con un BOSS dotato di S110 e S7-1200 subirà un ritardo rispetto all'analogo macchinario dotato di S7-300. Per calcolare questi ritardi occorre ipotizzare che le fasi abbiano la stessa durata sia nel BOSS con S7-300 che nel BOSS con S7-1200. Questa è un'ipotesi realistica in quanto è normale che il carrello impieghi lo stesso tempo a muoversi e che le valvole o i relè abbiano gli stessi tempi di commutazione indipendentemente dal PLC che li aziona. A questo punto le uniche differenze temporali risiedono nei tempi per avviare la fase e per accorgersi che la fase è terminata. Un PLC necessita di un ciclo per impostare le variabili che avviano una nuova fase e nel caso peggiore impiegherà un altro tempo di ciclo per accorgersi che la fase è terminata (Vedi Fig. 6.12).

Quindi, se si considera un ciclo di scansione di durata pari a 1ms per il PLC S7-300 in totale con nove fasi di lavoro si avrà un tempo di elaborazione dati di 18ms ogni cambio barra. Invece, per il PLC S7-1200 considerando un ciclo di durata pari a 9ms avremo un tempo totale di 162ms. La differenza di un cambio barra tra un BOSS dotato di PLC S7-314C e uno con S7-1212C è di 144ms, che è un tempo molto piccolo se rapportato ai 30 secondi circa di un cambio barra.

Quindi si può affermare che il PLC S7-1212C produce un degrado nelle prestazioni del sistema BOSS quasi insignificante.

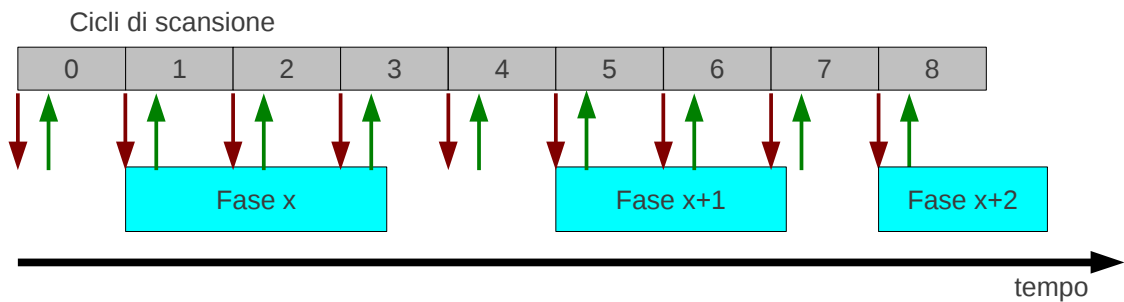


Figura 6.12: Tempi morti tra un ciclo e l'altro prima della avvio di una nuova fase

In **rosso** istante di scrittura delle uscite.

In **verde** istante di lettura degli ingressi.

Si nota che nel ciclo 3 la lettura degli ingressi avviene poco prima della fine della fase x quindi il PLC si accorge che la fase è finita solo un ciclo dopo, ciclo 4. Quindi c'è un tempo morto, nel caso peggiore di un ciclo di scansione. Questo non avviene nel ciclo 7 dove non c'è nessun ciclo “inutile” tra la fine di una fase e l'inizio di un'altra.

## 6.7 Interfaccia uomo macchina - HMI



Figura 6.13: TP 177B 4" PN/DP

Il simulatore è stato dotato di un pannello operatore touch screen modello: TP177B 4" PN/DP. Questo HMI ha un display TFT wide screen ad 4,3 pollici e 256 colori, ha una risoluzione di 480 x 272 pixels, supporta i protocolli di comunicazione ProfiNET, ProfiBUS e dispone di una porta USB e di un lettore di memory card MMC con la quale è possibile esportare e importare dati.

### 6.7.1 Struttura delle pagine

Gli HMI servono per poter gestire la macchina, ma in questo caso essendo il prodotto ancora in fase pre-prototipale le pagine sono state strutturate per fornire informazioni utili all'ingegnere che deve sviluppare il progetto. Quindi in primo piano sono presenti le PZD (divise tra le parole di comando e quelle di stato) e i relativi valori.

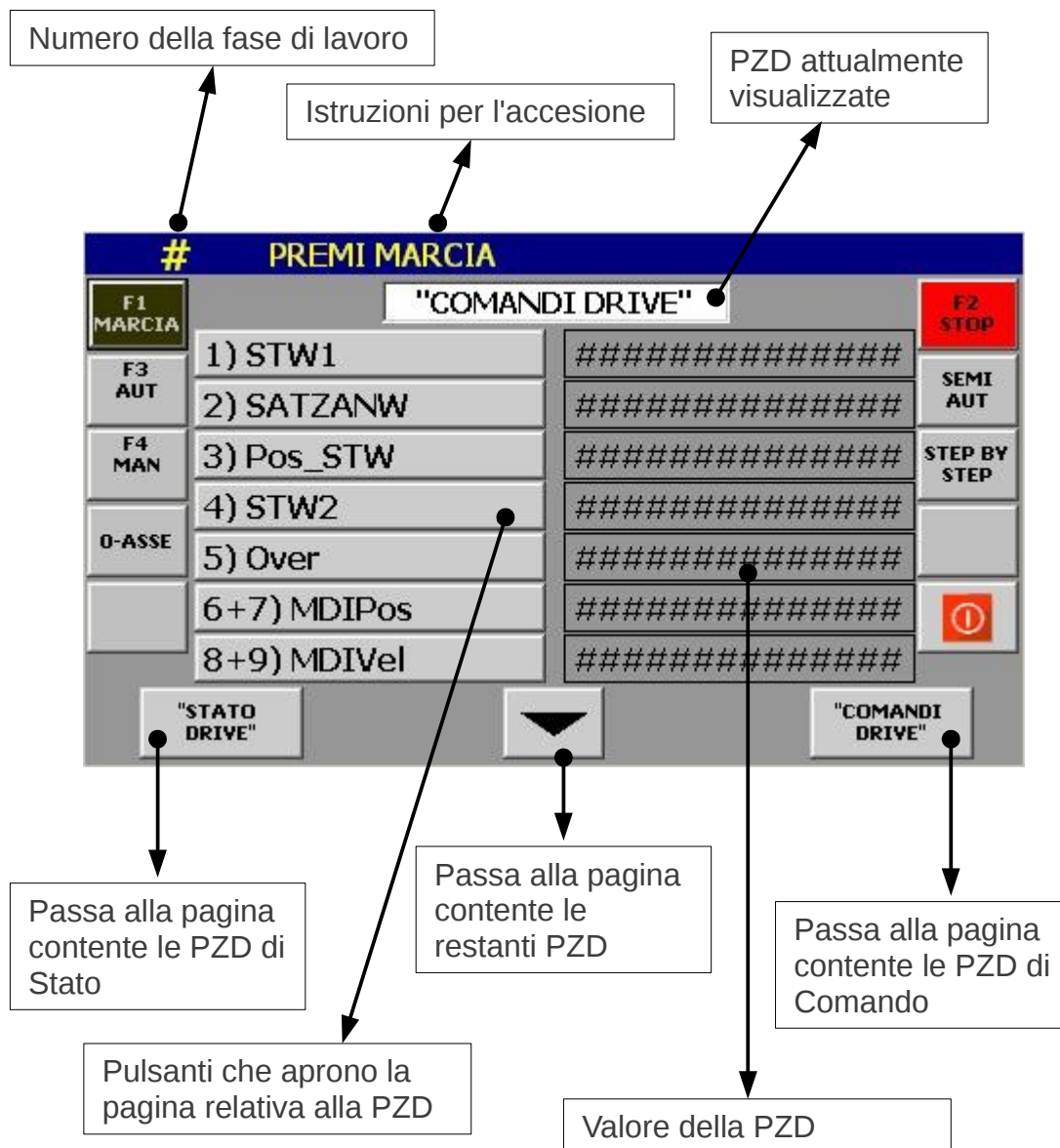


Figura 6.14: Pagina relativa alle prime 7 PZD delle parole di comando. I pulsanti nelle colonne a sinistra e destra servono per gestire la macchina.

Premendo su una parole di codice si apre la pagina della PZD nella quale sono visualizzati i valori e le descrizioni dei singoli bit che la compongono.

Grazie a queste pagine è stato poi possibile redarre il documento che descrive la comunicazione tra il PLC e l'azionamento. (Vedi: Appendice B: MOVIMENTI STANDARDIZZATI a pag 105).

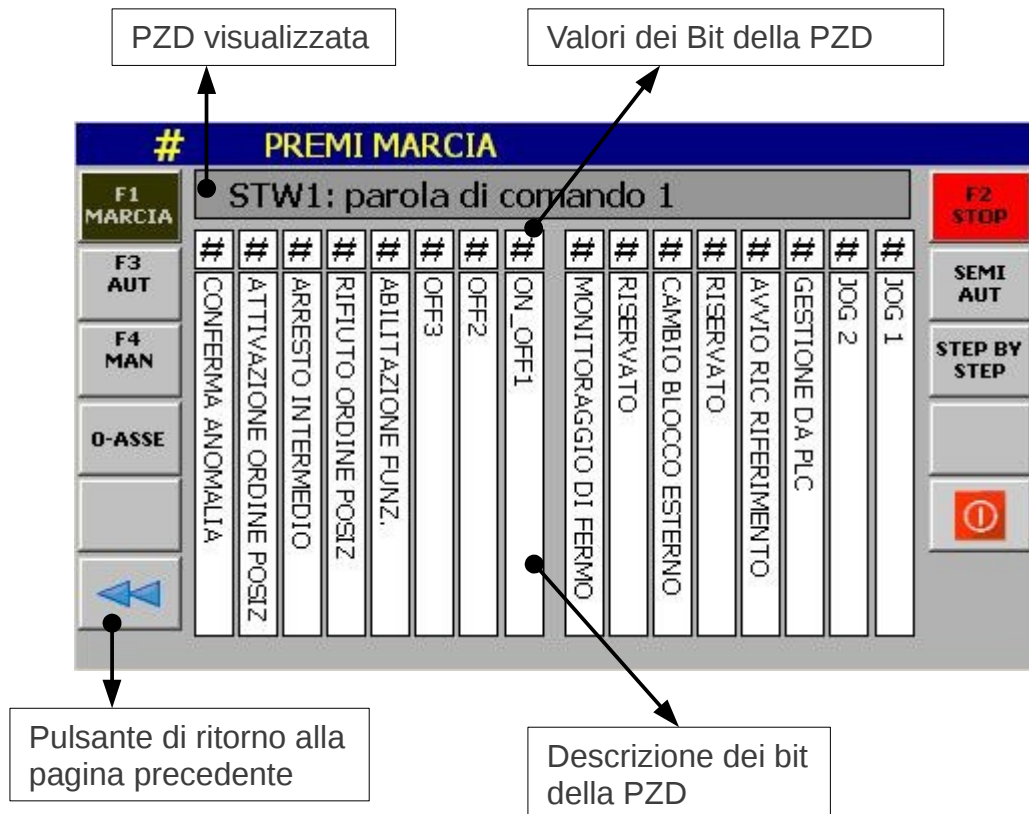


Figura 6.15: Pagina descrittiva di una PZD.

## 6.8 Web-Server

Sul simulatore finale (con il PLC S7-1212C) è stato implementato anche un web-server con il quale è possibile visualizzare i parametri di funzionamento del BOSS.

I PLC Siemens che possono implementare un web-server hanno già al loro interno delle pagine web standard con le quale l'operatore può eseguire diverse operazioni, come visualizzare la diagnostica del PLC, eseguire il login, leggere e scrivere le variabili del PLC (la scrittura delle variabili è possibile solo se è stato effettuato il login come amministratore) e visualizzare il numero di serie della CPU, la versione etc etc.

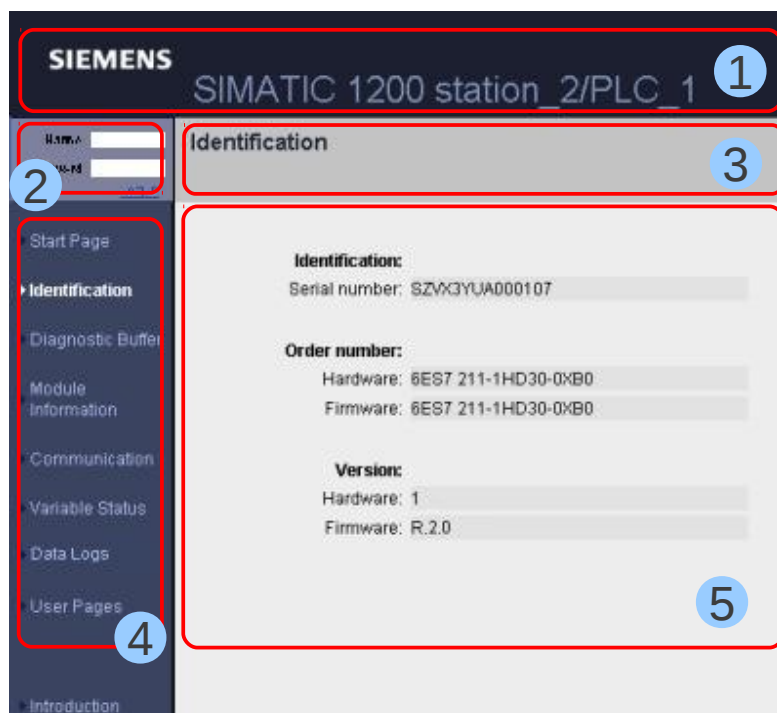


Figura 6.16: Layout delle pagine Web Standard

1 – Intestazione del Web-Server

2 – Pannello per il login / logout

3 – Intestazione della pagina Web standard con il nome della pagina visualizzata.

In questo esempio è la pagina di identificazione della CPU

4 – Menu per passare ad un'altra pagina web standard

5 – Area di contenuti per la specifica pagina Web standard visualizzata

Molti dei parametri messi a disposizione dalle pagine web-standard non sono interessanti per l'operatore del BOSS o sono complessi da eseguire, ad esempio: per poter visualizzare una o più variabili del PLC è necessario conoscerne il nome o l'indirizzo, entrambi sono parametri tipicamente non noti all'utente.

Per questo motivo, in aggiunta alle pagine web standard si possono affiancare delle pagine web personalizzate realizzate in HTML e JavaScript. Le pagine web personalizzate, tramite alcuni comandi (AWP) inseriti nel codice HTML permettono di far visualizzare o scrivere le variabili del PLC. In questo modo l'operatore del BOSS può facilmente vedere le variabili del PLC che gli sono state messe a disposizione tramite il codice HTML senza necessariamente conoscerne il nome o l'indirizzo.

### 6.8.1 Comandi AWP

I comandi AWP permettono di far visualizzare o scrivere i valori delle variabili PLC e hanno la seguente sintassi: `<!-- AWP_<command name and parameters> -->`. Il comando AWP per la lettura di una variabile ha una



forma anche compatta del tipo: **:=nome\_variabibile:**

Essendo inseriti tra “<!--” e “-->” per un browser, i comandi AWP sono semplici commenti HTML. Quando un PC fa richiesta al PLC di accedere ad una pagina web personalizzata, il PLC invia il codice HTML al browser dove i comandi AWP di lettura sono sostituiti con il valore della variabile interessata e i restanti comandi non sono visibili.

Il fatto che nel codice HTML che arriva al browser i comandi AWP di lettura siano sostituiti con il valore della variabile permette di utilizzare i comandi AWP anche all'interno di codici Javascript senza particolari difficoltà.

```
<script type="text/javascript">
function color_tab(){
  var x= :=SW0: ;
  if(x==1){
    document.getElementById("td1").style.backgroundColor="#00BB
00";
  }
  else{
    document.getElementById("td1").style.backgroundColor="#BB00
00";
  }
}
window.onload = color_tab;
</script>
```

*Esempio di un sorgente Javascript contenente un comando AWP per la lettura della variabile SW0. Dal browser vedremo al posto di :=SW0: o “0” o “1”. E quindi l'elemento “td1” avrà lo sfondo rosso o verde.*

Tramite i comandi AWP è possibile anche associare un testo al valore di una variabile grazie a gli ENUM e scrivere una variabile. Per scrivere i dati nella CPU si utilizzano il comando POST messo a disposizione dal protocollo HTTP.

```

<html>
<head>
...
<!-- AWP_In_Variable Name='SW0' -->
...
</head>
<body>
  <form method="post">
    <p> Valore di SW0:
      <input name='SW0' size="3" type="text" />
    </p>
  </form>
</body>
</html>


```

Esempio di un sorgente HTML con scrittura della variabile SW0 tramite il metodo POST implementato dal HTTP.

Anche nelle pagine web personalizzate la scrittura di una variabile può avvenire solo se è stato eseguito il login presso le pagine web standard.

Le pagine web personalizzate nel PLC S7-1200 sono memorizzate all'interno di uno o più database. Come impostazione predefinita i database partono dal numero 333. Quindi per mantenere una uniformità nei progetti IMECA futuri è necessario che il database DB333 e limitrofi siano liberi per il web-server. Visto che il web-server creato per questo progetto è composto da 5 pagine HTML e occupa solo due database non sarà necessario impegnare molti DB dopo il 333 anche perché la memoria nel PLC è limitata e quindi è impossibile creare molte pagine web che occupino diversi DB.

Dopo aver creato i DB contenenti le pagine web personalizzate è necessario chiamare la funzione WWW all'interno della programma del PLC affinché le pagine siano accessibili da un browser.

KOP / FUP	SCL	Descrizione
	<pre>ret_val:=WWW( ctrl_db:=_uint_in_ );</pre>	<p>Consente l'accesso alle pagine Web personalizzate</p>

Funzione WWW in linguaggio KOP e SCL (rispettivamente Ladder Diagram e Structured Text ST di Siemens)

## 6.8.2 Web Server realizzato

Inizialmente sono state eseguite diverse prove per testare le funzionalità del web-server, come la scrittura e la lettura di variabili, l'utilizzo di enum e integrare i comandi AWP all'interno di Javascript.

Infine, si è optato per realizzare un web-server orientato verso l'assistenza per i clienti che permette di visualizzare i parametri impostati dall'operatore e i parametri impostati dagli installatori IEMCA.

**IEMCA**  
Via Granarolo, 167  
48018 Faenza - RA -  
Italy  
Phone: +39 0546 698000  
Fax: +39 0546 46338  
E-mail: [iemca@iemi.it](mailto:iemca@iemi.it)

**Web-Server  
Caricatore**

**MENU**

[PARAMETRI UTENTE](#)  
Parametro:

[PARAMETRI IEMCA](#)  
Parametro:

Ver. sw: .xxx.xx  
Ver. Web-Server: 0.1v

[Aggiorna](#)

**1) Regolazione Fine Barra**

Fine barra 2 (Ultimo)  mm

Fine barra 1  mm

[TOP PAGE](#)

**2) Lunghezza intestatura**

Lunghezza intestatura  mm

[TOP PAGE](#)

**3) Modo di intestatura**

Tipo di intestatura  mm

Altro campo ammesso dopo "!" (al click sul GUI con 0mm)  mm

Figura 6.17: Layout della pagina web contenente i parametri utente

Il layout della pagina è composto 4 frame (Vedi Fig. 6.17). In alto, a sinistra troviamo i dati dell'azienda, nel centro troviamo l'intestazione del web-server. In basso, a sinistra è presente il menù, realizzato in JavaScript, per selezionare il parametro da visualizzare e la versione software del caricatore.

Nel centro è presente il frame principale dove sono elencati i parametri con i

relativi valori e sono descritte le loro funzioni. Attualmente si è scelto di disabilitare la scrittura della variabili.

### **6.8.3 Valutazione del Web-Server presente sul S7-1200**

Il web-server, messo a disposizione da Siemens, presenta diverse lacune strutturali che non sempre le pagine web personalizzate possono colmare, dimostrando di essere un progetto ancora acerbo ma con molte potenzialità.

Come è già stato detto in precedenza la pagine web standard mettono a disposizione molti dati tipicamente non interessanti per il cliente finale e per poter visualizzare una variabile è necessario conoscerne il nome o l'indirizzo. Questi, però, sono aspetti secondari, che possono essere aggirati con le pagine web personalizzate. Il problema principale riguarda la sicurezza informatica.

Quando qualunque strumento viene messo in una rete più o meno grande occorre prestare attenzione a rischi di manomissione accidentale o voluta che lo strumento può subire, ma nel web-server implementato da Siemens molti accorgimenti sulla sicurezza informatica non sono stati minimamente presi in considerazione.

In primo luogo le pagine web standard sono sempre presenti anche se si abilitano le pagine web personalizzate. Questo comporta che tutte le variabili presenti nel PLC siano leggibili dalle pagine web standard, è sufficiente conoscerne il nome o l'indirizzo della variabile. Questi parametri non sono noti all'operatore ma sono comunque facilmente indovinabili, soprattutto gli indirizzi che tipicamente sono sequenziali.

Oltre a questo, non sono implementate politiche di accesso di tipo RBAC (Role-based access control). Le quali prevedono che ogni utente possa accedere ad un numero limitato delle risorse in funzione del ruolo che riveste. Il web-server presentato da Siemens non ha una divisione in ruoli e addirittura sul PLC S7-1200 non è possibile creare più utenti. Quindi esiste solo un super utente (l'amministratore) che ha accesso illimitato a tutte le risorse del sistema.

Questo aspetto limita molto le funzionalità del web-server, non si può permettere a un cliente di leggere e scrivere tutte le variabili di un sistema quindi non si possono lasciare le credenziali di accesso dell'amministratore ai clienti, ma in questo modo impediamo ai clienti di poter scrivere qualunque variabile. Anche le variabili che riguardano la configurazione della macchina.

Paradossalmente in molti pannelli touch-screen della Siemens (come ad esempio in quello utilizzato nel simulatore) che non hanno l'accessibilità da remoto è possibile avere più utenti e definire per ognuno a quali variabili del sistema possono accedere. Questo dimostra che il progetto del web-server è

incompleto non perché Siemens non sia sensibile sulle tematiche relative alla sicurezza informatica ma perché probabilmente è entrato in commercio prima del suo completo sviluppo.

D'altra parte, la possibilità di implementare dei semplici web-server anche su PLC di fascia bassa rivoluziona il mondo dell'assistenza ai clienti riducendo i costi e migliorando i servizi. Quindi sicuramente Siemens continuerà a sviluppare il web-server fino a colmare queste lacune.

*“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”*

## **7 CONCLUSIONI**

Questa tesi ha portato alla realizzazione di un simulatore di un macchinario BOSS, nel quale sono stati sostituiti l'azionamento analogico prodotto da “Elmo” con un azionamento digitale Siemens S110 e il PLC S7-300 con il S7-1200. Il prototipo così realizzato permette di simulare le principali funzionalità di un caricatore di barre per torni mono-mandrino. La scelta di utilizzare un azionamento digitale di alto livello come l'S110 ha consentito di spostare il carico di lavoro del motion-controll dal PLC all'azionamento. In questo modo è stato possibile utilizzare un PLC di fascia più bassa e quindi anche meno costoso.

Protocolli di comunicazione per BUS di campo esistono da diversi anni, ma è solo con l'ingresso sul mercato di PLC poco costosi con l'industrial ethernet integrato che è stato reso possibile e interessante questo progetto.

Il PLC S7-1200 è un PLC economico con una capacità di calcolo inferiore al S7-300 ma ha comunque una memoria di caricamento ampia con la quale può implementare programmi complessi. Inoltre ha una porta ethernet integrata che gli consente di comunicare con il protocollo HTTP e il protocollo industrial ethernet ProfiNET. Durante i test, è emerso che la memoria a ritenzione, di questo PLC, viene impiegata per il 49%. Questo non è un grosso limite per gli sviluppi futuri, in quanto il problema è aggirabile o utilizzando una CPU 1214 che ha il doppio della memoria a ritenzione o strutturando diversamente i DB sovradimensionati che sono salvati nella memoria a ritenzione.

Il sistema simulato con l'S7-1200, presenta tempi di ciclo più lunghi rispetto al sistema simulato con S7-300. La differenza dei tempi di un cambio barra tra i due sistemi è pari a 144ms che è insignificante rispetto al tempo totale di cambio barra del BOSS che è circa 30 secondi. Quindi questo nuovo PLC dimostra che pur avendo una capacità di calcolo limitata permette comunque di produrre un BOSS con delle prestazioni in linea a quelle del suo predecessore.

Sul S7-1200 è stato sviluppato anche un web-server che permette di visualizzare i parametri della macchina impostati dall'operatore. Il web-server messo a disposizione da Siemens presenta alcune lacune per quanto riguarda la sicurezza informatica, però apre nuovi orizzonti nel mondo dell'assistenza clienti e nella gestione della macchina che meritano di essere approfonditi.

Nel complesso quindi si può affermare che l'aggiornamento del sistema BOSS con il PLC S7-1212C e l'azionamento SINAMICS S110 porta diversi vantaggi: semplificazione del cablaggio della macchina, una maggiore flessibilità del prodotto e un risparmio economico. Inizialmente, basandosi sui listini dei

fornitori, si prevedeva un discreto risparmio economico solo utilizzando il PLC S7-1200 e un azionamento non Siemens ma compatibile con l'S110. Negli ultimi tempi però c'è stato un forte sviluppo del mercato degli azionamenti digitali di piccola taglia e quindi ad un aumento della concorrenza. Questo ha abbassato i prezzi ed ha permesso un risparmio superiore a quello inizialmente preventivato.



## 8 BIBLIOGRAFIA

1. “*Industrial Ethernet: A Control Engineer's Guide*” - 2010 - CISCO System Inc. ([www.cisco.com](http://www.cisco.com))
2. “*Bus di Campo IEC 61158: Caratteristiche Principali*” - 1998 - Prof. Salvatore Cavalieri Università di Catania, Facoltà di Ingegneria Istituto di Informatica e Telecomunicazioni
3. “*PROFINET System Description; Technology and Application*” - 2011 - PI - PROFIBUS & PROFINET International 3. ([www.profibus.com](http://www.profibus.com))
4. “*PROFIBUS System Description; Technology and Application*” - 2011 - PI - PROFIBUS & PROFINET International 4. ([www.profibus.com](http://www.profibus.com))
5. “*Media Redundancy (MRP) for ProfiNET IO*” - 2008 - Max Felser - Bern University of Applied Sciences - Engineering and Information Technology
6. [as-interface.net](http://as-interface.net)
7. “*A Performance analysis of EtherCAT and ProfiNET*” - 2008 - Gunnar Prytz - ABB AS Corporate Research Center - Billingstad, Norway
8. “*Industrial Redundant Network*” 2010 - Marc Lee - Technical Manager, Industrial Networking - Belden Automation (Asia Pacific) Pted Ltd
9. “*IEC 61158 Industrial communication networks - Fieldbus specifications*” - 2010 – IEC: International Electrotechnical Commission 9. ([www.iec.com](http://www.iec.com))
10. [en.wikipedia.org/wiki/fieldbus](http://en.wikipedia.org/wiki/fieldbus)
11. “*Controllori SIMATIC*” - 2011- Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
12. “*Sistema di automazione S7-1200 – Manuale di sistema*” - 2011 - Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
13. “*Totally Integrated Automation – Product Guide*” - 2011 - Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
14. “*SINAMICS - the seamless and integrated converter family for every*

*“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”*

*application”* - 2011- Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))

15. *“Manuale di guida alle funzioni – SINAMICS S120”* - 2009 - Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
16. *“Manuale del prodotto – SINAMICS S110”* - 2009 - Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
17. *“Sensorless vector control (SVC)”* - 1999 - Martin Brown - A & D SD D
18. *“Motori sincroni 1FK7 – 2° generazione – Manuale di progettazione”* - 2011 - Siemens AG 11. ([www.automation.siemens.com](http://www.automation.siemens.com))
19. *“Sensori e sistemi AS-Interface”* - 2005 - Andrea Fenzi - ed. Tecniche Nuove

## **9 Ringraziamenti**

Con questa tesi si conclude finalmente un lungo percorso di studi che mi ha portato a conoscere meglio vecchi amici ed a incontrare nuovi compagni di avventura.

In primis occorre ringraziare tutta la mia famiglia che mi ha spronato per raggiungere questa agognata meta. Carissimi genitori, con con questa siamo a due, e pu basta!

Un ringraziamento sentito va all'azienda IEMCA che mi ha ospitato per questo progetto ed in particolar modo a tutti i colleghi dell'uffici tecnico elettrico: Maurizio, Andrea, Roberto, Massimo e Gabriele.

Un ringraziamento ironico va all'ufficio tirocini che ci ha messo un mese e mezzo per sbrigare la mia pratica... Mentre un ringraziamento sincero va alla segreteria didattica della facoltà di ingegneria (in particolare a Benedetta) ed al Prof. Aldo Romani che ci hanno messo mezza giornata per completare le mie pratiche nonostante il professore fosse a Parigi e la segretaria e Cesena...

A questo punto occorre salutare e ringraziare i compagni di viaggio, partiamo dai Cocinghiati: JJ, Tutor, Murò, Mundrò, Vale, Lasi, Vito, Ivnk, Ronca sr. Fuori da cocinghia ma comunque legati ad essa ringrazio:Ronca jr, mamma Pi ed Elena. Un ringraziamento va anche ai compagni di studio/lavoro: Tasso, Vimone, il mitico Dini, Sergio (Enrico), gli ex colleghi di Sortron, Albert ed infine Elia. A parte voglio ringraziare Boris perché tutte le volte che ho studiato con lui io ho preso 30 e lui 28 nonostante lui ne sapesse sempre più di me. Quindi capite che si meritava almeno un ringraziamento tutto per lui. Fuori dal mondo universitario dovrei ringraziare molta gente che mi ha comunque sostenuto; in particolar modo ringrazio Ivan per serate passate davanti ad una buona birra e l'Alessia per avermi raccontato tutta la storia italiana dal 1300 ad oggi e parte della storia di Roma. Sono sicuro che mi sarà utilissima nella vita, non ho dubbi... Quasi per ultime, ma non meno importanti, ringrazio Vale e Marti perché sono semplicemente uniche e insostituibili.

Infine:

A Sara che continua a supportarmi/sopportarmi; grazie.

## Appendice A: LISTA COMPONENTI ACQUISTATI

Elenco dei componenti ordinati per la realizzazione del simulatore con i relativi codici prodotto (catalogo Siemens)

Qta	nr. Catalogo	Descrizione
<b>DRIVE</b>		
1	6SL3040-0JA01-0AA0	SINAMICS S110 control unit CU305 PN con profibus interfaccia senza scheda di memoria
1	6SL3210-1SB14-0AA0	Power module pm340 ingresso: 1ac 200-240v, 50/60hz uscita: 3ac 3,9a (0,75kw) f. costr.: blocksize grandezze costruttive fsa con filtro d rete montato raffreddamento interno ad aria
1	6FX5002-2DC10-1AC0	Cavo segnali confezionati (SINAMICS DRIVE-CLiQ) connettore IP20/IP67, con 24V motion-connect 500 lunghezza (m) 0+0+2+0
1	6SL3262-1AA00-0BA0	SINAMICS piastra coll. Schermi PM240FSA
1	6FX5002-5CG01-1AC0	Cavo di potenza confezionato 4x1.5 C, connettore GR.1 (SINAMICS AC DRIVE) UL/CSA, desina motion-connect 500 DMAX=8,4 mm, 0+0+2+0
1	1FK7042-5AF21-1PA0	Servomotore sincro 1FK7 compact, ZK300 V 3,0 NM, 100k, 3000g/min 0.82kW raffreddamento naturale, IM B5 connettori segnale/potenza ruotabili di 270 gradi. Resolver 2-poli con interfaccia DRIVE-CLiQ tolleranza N senza frenotipo di protezione IP64
<b>PLC</b>		
1	6ES7212-1AD30-0XB0	SIMATIC S7-1200, cpu 1212c, cpu compatta, dc/dc/dc, i/o onboard: 8 di 24v dc; 6 do 24 v dc; 2 ai 0 - 10v dc or 0 - 20ma, alim. dc 20.4 - 28.8 v dc, memoria programma/dati: 25 kb
2	6ES7223-1BL30-0XB0	SIMATIC S7-1200, I/O digitali SM 1223, 16di / 16do, 16di dc 24 v, sink/source, 16do, transistor 0.5a
1	6ES7222-1BL30-0XB0	SIMATIC S7-1200, I/O digitali SB 1222, 4do, dc 24 v, 200KHz
<b>OP</b>		
1	6AV6642-0BD01-3AX0	SIMATIC TP177B 4" color PN/DP display 16:9 TFT 256 colori, MPI PROFIBUS DP - protocollo RS485 RS422 USB, interf.ethernet, 10/100 interface stamp. interface slot per mm card configurabile con wincc flexible 2008
<b>ACCESSORI COLLEGAMENTO</b>		
2	6GK1901-1BB10-2AA0	IE FC RJ45 plug 180 2x2, RJ45 plug connector (10/100Mbit/s) w. rugged metal housing and fc connecting method, for ie fc cable 2x2 180 dgr cable outlet 1 pack = 1 pcs
1	6XV1840-2AH10	SIMATIC net, ie fc tp standard cable, gp 2x2 (profinet typ a) tp installation cable for connection to fc outlet rj45, for universal application, 4-wire, shielded, cat. 5, sold by the meter max. consignment: 2000 m, min. ordering quantity: 20 m

## Appendice B: MOVIMENTI STANDARDIZZATI

Di seguito sono riportati i valori delle parole di comando e di stato in funzione dell'azione eseguita dal S110.

Per una descrizione accurata delle singole parole e del significato dei loro bit si rimanda al manuale del SINAMICS S110.

Parole di stato	Indirizzo	Tipo	Accensione	Startup (Asse fermo non riferito)	Atesa "Zs"	Manuale				Automatico					
						Zero Asse	Asse fermo	Momento Jog Avanti	Momento Jog Indietro	Zero Asse	Asse fermo in coppia	Asse fermo libero	Mov Assoluto a posizione	Mov Relativo a posizione	Mov avanti a battuta Jog
<b>Parole di stato</b>															
<b>ZSW1</b>	Struct	0													
Avviso_attivo	Bool	0.7	0	0	0	0	0	0	0	0	0	1	0	0	0
Blocco_inserzione	Bool	0.6	1	0	0	0	0	0	0	0	0	0	0	0	0
Arresto_per_OFF3	Bool	0.5	0	1	1	1	1	1	1	1	1	1	1	1	1
Arresto_per_OFF2	Bool	0.4	0	1	1	1	1	1	1	1	1	1	1	1	1
Anomalia_attiva	Bool	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0
Funzionamento_abilitato	Bool	0.2	0	0	1	1	1	1	1	1	1	0	1	1	1
Pronto_al_funzionamento	Bool	0.1	0	0	1	1	1	1	1	1	1	0	1	1	1
Pronto_alla_inserzione	Bool	0.0	0	1	1	1	1	1	1	1	1	1	1	1	1
Lim_coppia_neg_ragg	Bool	1.7	0	0	0	0	0	0	0	0	0	0	0	0	0
Lim_coppia_pos_ragg	Bool	1.6	0	0	0	0	0	0	0	0	0	0	0	0	0
Asse_fermo	Bool	1.5	1	1	1	0	1	0	0	0	1	1	0	0	0
Tacit_blocco_movimento	Bool	1.4	0	0	0	0	0	0	0	0	0	0	0	0	0
Asse_azzerato	Bool	1.3	0	0	1	0	1	1	1	0	1	1	1	1	1
Posizione_raggiunta	Bool	1.2	0	0	0	0	0	0	0	0	1	0	0	0	0
Gestione_rich_al_PLC	Bool	1.1	1	1	1	1	1	1	1	1	1	1	1	1	1
No_err_inseguimento	Bool	1.0	1	1	1	1	1	1	1	1	1	1	1	1	1
<b>AKTSATZ</b>															
	Struct	2													
Riservato_08	Bool	2.7	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_07	Bool	2.6	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_5	Bool	2.5	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_4	Bool	2.4	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_3	Bool	2.3	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_2	Bool	2.2	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_1	Bool	2.1	0	0	0	0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_0	Bool	2.0	0	0	0	0	0	0	0	0	0	0	0	0	0
MDI_attivo	Bool	3.7	0	0	1	0	1	0	0	0	1	0	1	1	0
Riservato_015	Bool	3.6	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_014	Bool	3.5	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_013	Bool	3.4	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_012	Bool	3.3	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_011	Bool	3.2	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_010	Bool	3.1	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_09	Bool	3.0	0	0	0	0	0	0	0	0	0	0	0	0	0

"Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET"

Pos.	Nome	Tipo	Indirizzo	Accessione	Startup (Asse fermo non ritenuto)	Manuale					Automatico						
						Preset	Zero Asse	Asse fermo	Movimento Jog Avanti	Movimento Jog Indietro	Zero Asse	Asse fermo in coppia	Asse fermo libero	Mov Assoluto a posizione	Mov Relativo a posizione	Mov avanti a battuta Jog	Mov indietro a battuta Jog
<b>Pos_ZSW</b>													<b>Struct</b>		<b>4</b>		
	Fc_sw_av_raggiunto	Bool	4.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Fc_sw_ind_raggiunto	Bool	4.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Movimento_indietro	Bool	4.5	0	0	0	1	0	0	1	1	0	0	1	1	0	1
	Movimento_avanti	Bool	4.4	0	0	0	0	0	1	0	0	0	0	0	0	1	0
	Riservato_04	Bool	4.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riferimento_presente	Bool	4.2	0	0	1	0	1	0	0	0	1	0	0	0	0	0
	Limitazione_vel_attiva	Bool	4.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Funzionamento_a_seguire	Bool	4.0	1	1	0	0	0	0	0	0	0	1	0	0	0	0
	Riservato_016	Bool	5.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Posizionamento_a_battuta	Bool	5.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Coppia_a_battuta_ragg	Bool	5.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Battuta_ragg	Bool	5.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Usc_2_blocco_spos	Bool	5.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Usc_1_blocco_spos	Bool	5.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Segn_camma_2	Bool	5.1	1	1	0	1	0	0	0	1	0	0	0	0	0	0
	Segn_camma_1	Bool	5.0	1	1	0	1	0	0	0	1	0	0	0	0	0	0
<b>ZSW2</b>													<b>Struct</b>		<b>6</b>		
	Asse_in_parcheggio	Bool	6.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_07	Bool	6.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_06	Bool	6.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Record_DDS_sel_bit_4	Bool	6.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Record_DDS_sel_bit_3	Bool	6.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Record_DDS_sel_bit_2	Bool	6.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Record_DDS_sel_bit_1	Bool	6.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Record_DDS_sel_bit_0	Bool	6.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Funz_vitale_bit_3	Bool	7.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Funz_vitale_bit_2	Bool	7.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Funz_vitale_bit_1	Bool	7.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Funz_vitale_bit_0	Bool	7.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Commutazione_record_dati	Bool	7.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_11	Bool	7.2	0	0	1	1	1	1	1	1	1	0	1	1	1	1
	Riservato_10	Bool	7.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Pos_su_riscontro_fisso	Bool	7.0	0	0	1	1	1	1	1	0	0	0	0	1	1	1
<b>MELDW</b>													<b>Struct</b>		<b>8</b>		
	Sovraccarico_termico	Bool	8.7	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Surriscaldamento_motore	Bool	8.6	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Riservato_05	Bool	8.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_04	Bool	8.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Giri_inf_magg_p2155	Bool	8.3	1	1	1	1	1	0	0	1	1	1	0	0	0	0
	Giri_inf_magg_p2161	Bool	8.2	1	1	1	0	1	0	0	0	1	1	0	0	0	0
	Coppia_inf_magg_p2194	Bool	8.1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Acc_dec_conclusa	Bool	8.0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Riservato_15	Bool	9.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_14	Bool	9.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Impulsi_abilitati	Bool	9.5	0	0	1	1	1	1	1	1	1	1	1	1	1	1
	Riservato_12	Bool	9.4	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	Riservato_11	Bool	9.3	0	0	1	1	1	1	1	1	1	1	1	1	1	1
	Riservato_10	Bool	9.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Riservato_09	Bool	9.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Scostamento_Vrif_Vreal	Bool	9.0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
<b>XistP</b>													<b>DInt</b>		<b>10</b>		
						Old Pos	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist	Pos ist

“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”

Parole di Comando	Tipo	Offset	Accensione	Startup (Asse fermo non riferito)	Attesa "2s"	Manuale				Automatico					
						Preset	Zero Asse	Asse fermo	Movimento Jog Avanti	Movimento Jog Indietro	Zero Asse	Asse fermo in coppia	Asse fermo libero	Mov Assoluto a posizione	Mov Relativo a posizione
<b>STW1</b>															
Conferma_anomalia	Bool	32.7	0	0		0	0	0	0	0	0	0	0	0	0
Attivazione_ordine_posiz	Bool	32.6	0	0		0	0	0	0	0	0	0	0	0	0
Arresto_intermedio	Bool	32.5	0	1		1	1	1	1	1	1	1	1	1	1
Rifiuto_ordine_posiz	Bool	32.4	0	1		1	1	1	1	1	1	1	1	1	1
Ablit_funzionam	Bool	32.3	0	1		1	1	1	1	1	1	1	1	1	1
OFF3	Bool	32.2	0	1		1	1	1	1	1	1	1	1	1	1
OFF2	Bool	32.1	0	1		1	1	1	1	1	1	1	1	1	1
ON_OFF1	Bool	32.0	0	0		1	1	1	1	1	1	0	1	1	1
Monitoraggio_di_fermo	Bool	33.7	0	0		0	0	0	0	0	1	0	0	0	0
Riservato_15	Bool	33.6	0	0		0	0	0	0	0	0	0	0	0	0
Cambio_blocco_esterno	Bool	33.5	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_13	Bool	33.4	0	0		0	0	0	0	0	0	0	0	0	0
Avvio_ric_riferimento	Bool	33.3	0	0		0	1	0	0	1	0	0	0	0	0
Gestione_da_PLC	Bool	33.2	0	1		1	1	1	1	1	1	1	1	1	1
Jog_2	Bool	33.1	0	0		0	0	0	1	0	0	0	0	1	0
Jog_1	Bool	33.0	0	0		0	0	0	0	1	0	0	0	0	1
<b>SATZANW</b>															
Riservato_08	Bool	34.7	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_07	Bool	34.6	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_5	Bool	34.5	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_4	Bool	34.4	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_3	Bool	34.3	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_2	Bool	34.2	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_1	Bool	34.1	0	0		0	0	0	0	0	0	0	0	0	0
Blocco_movimento_bit_0	Bool	34.0	0	0		0	0	0	0	0	0	0	0	0	0
Attivazione_MDI *	Bool	35.7	0	0		1	1	1	1	1	1	1	1	1	1
Riservato_15	Bool	35.6	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_14	Bool	35.5	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_13	Bool	35.4	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_12	Bool	35.3	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_11	Bool	35.2	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_10	Bool	35.1	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_09	Bool	35.0	0	0		0	0	0	0	0	0	0	0	0	0
<b>Pos_STW</b>															
Riservato_08	Bool	36.7	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_07	Bool	36.6	0	0		0	0	0	0	0	0	0	0	0	0
Funz_incrementale_a_imp	Bool	36.5	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_05	Bool	36.4	0	0		0	0	0	0	0	0	0	0	0	0
Sensore_riscontro_fisso	Bool	36.3	0	0		0	0	0	0	0	0	0	0	0	0
Camma_di_riferimento	Bool	36.2	0	0		0	0	0	0	0	0	0	0	0	0
Imp_punto_riferimento	Bool	36.1	0	0		**0/1/0	0	0	0	0	0	0	0	0	0
Funzionamento_a_seguire	Bool	36.0	0	0		0	0	0	0	0	0	1	0	0	0
Riservato_016	Bool	37.7	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_015	Bool	37.6	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_014	Bool	37.5	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_013	Bool	37.4	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_012	Bool	37.3	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_011	Bool	37.2	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_010	Bool	37.1	0	0		0	0	0	0	0	0	0	0	0	0
Riservato_09	Bool	37.0	0	0		0	0	0	0	0	0	0	0	0	0

\*Arrivazione\_MDI potrebbe porla fissa a 1 nella programmazione dell'azionamento

\*\* Il Bit ha un fronte di salita e poi di discesa. Nell'istante che è a "1" l'azionamento imposta il riferimento a OldPos

“Studio ed Implementazione di un Posizionatore Interfacciato su Rete ProfiNET”

	Tipo	Offset	Accensione	Startup (Asse fermo non riferito)	Attesa "Zs"	Manuale				Automatico						
						Preset	Zero Asse	Asse fermo	Movimento Jog Avanti	Movimento Jog Indietro	Zero Asse	Asse fermo in coppia	Asse fermo libero	Mov Assoluto a posizione	Mov Relativo a posizione	Mov avanti a battuta Jog
<b>STW2</b>	<b>Struct</b>	<b>38</b>														
Asse_in_parcheggio	Bool	38.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_07	Bool	38.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_06	Bool	38.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Record_DDS_sel_bit_4	Bool	38.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Record_DDS_sel_bit_3	Bool	38.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Record_DDS_sel_bit_2	Bool	38.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Record_DDS_sel_bit_1	Bool	38.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Record_DDS_sel_bit_0	Bool	38.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Funz_vitale_bit_3	Bool	39.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Funz_vitale_bit_2	Bool	39.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Funz_vitale_bit_1	Bool	39.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Funz_vitale_bit_0	Bool	39.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Commutazione_record_dati	Bool	39.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_11	Bool	39.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Riservato_10	Bool	39.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Pos_su_riscontro_fisso	Bool	39.0	0	0	1	1	1	1	1	0	0	0	0	0	1	1
<b>Over</b>	<b>Int</b>	<b>40</b>														
			0	0	1000	500	1000	500	500	500	500	500	50	50	500	500
<b>MDIPos</b>	<b>DInt</b>	<b>42</b>														
			0	0	0	0	0	0	0	0	0	0	Pos	Pos	0	0
<b>MDIVel</b>	<b>DInt</b>	<b>46</b>														
			0	0	0	vel	0	vel	vel	vel	vel	vel	vel	vel	vel	vel
<b>MDIAcc</b>	<b>Int</b>	<b>50</b>														
			0	0	0	100	0	100	100	100	100	100	100	100	100	100
<b>MDIDec</b>	<b>Int</b>	<b>52</b>														
			0	0	0	100	0	100	100	100	100	100	100	100	100	100
<b>MDIMode</b>	<b>Int</b>	<b>54</b>														
			0	0	0	0	0	0	0	0	0	0	10	0	0	0
<b>MOMRED***</b>	<b>Int</b>	<b>56</b>														
			0	0	0	25	0	25	25	25	25	25	25	25	25	25
<b>CPRif***</b>	<b>DInt</b>	<b>58</b>														
			0	0	Old Pos*	**0	**0	**0	**0	**0	**0	**0	**0	**0	**0	**0

\*Old Pos - CPRif contiene l'ultimo valore di XistP prima che la macchina venisse spenta

\*\*0 - Dopo il preset CPRif contiene sempre " 0 "

\*\*\* MOMRED e CPRif non sono presenti nel telegramma 110, sono state aggiunte nel nostro telegramma libero