

UNIVERSITÀ DEGLI STUDI DI PADOVA  
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

**Tesi di Laurea**

*Collegamento alla rete PROFINET IO di una scheda  
per l'acquisizione di dati provenienti da un sistema  
di pesatura per macchine imbottigliatrici*

Relatore: Prof. Stefano Vitturi  
Laureando: Giuseppe Zoppellaro

ANNO ACCADEMICO 2009-2010



*ai miei genitori  
a mio fratello*



# Indice

<b>Indice</b>	<b>IV</b>
<b>Elenco degli acronimi</b>	<b>V</b>
<b>Introduzione</b>	<b>1</b>
<b>Presentazione Weightpack</b>	<b>5</b>
<b>1 La comunicazione industriale</b>	<b>9</b>
1.1 I bus di campo . . . . .	9
1.2 Sistemi di comunicazione . . . . .	11
1.3 Sistemi distribuiti . . . . .	12
1.4 Il fenomeno Ethernet . . . . .	13
1.4.1 Ethernet tra Real time e determinismo . . . . .	13
1.4.2 Ethernet industriale . . . . .	15
1.4.3 Real Time Ethernet . . . . .	16
1.4.4 Protocolli Industrial Ethernet . . . . .	17
<b>2 Il protocollo di comunicazione industriale Profinet IO</b>	<b>19</b>
2.1 Scalabilità e prestazioni . . . . .	19
2.2 Classi di comunicazione . . . . .	21
2.3 Classi dei dispositivi . . . . .	22
2.4 Il periodo di trasmissione Profinet . . . . .	24
2.5 Lo stack di comunicazione . . . . .	25
2.6 I canali di comunicazione . . . . .	26
2.7 Scambio di dati ciclico . . . . .	27
2.7.1 Il ciclo RT . . . . .	27
2.8 Integrazione dei bus di campo . . . . .	29
<b>3 Strumentazione da laboratorio e dispositivi Profinet IO</b>	<b>31</b>
3.1 Il PLC “Siemens S7-300” . . . . .	32
3.2 Lo switch “Scalance” . . . . .	34

3.3	L'IO device "ET200M" . . . . .	34
3.4	L'IO device "Anybus" . . . . .	35
3.4.1	Anybus-IC . . . . .	36
3.4.2	Interfaccia di comunicazione seriale asincrona - SCI . . . . .	38
3.4.3	Interfaccia di comunicazione seriale sincrona - SSC . . . . .	38
3.4.4	La scheda "Evaluation Board" . . . . .	39
3.5	La stazione PC . . . . .	39
3.5.1	Il tool progettuale Step7 . . . . .	39
3.5.2	Lo SCADA "Zenon" . . . . .	42
3.6	L'oscilloscopio "HP" . . . . .	43
<b>4</b>	<b>Misure e risultati sperimentali</b>	<b>47</b>
4.1	Prima fase: allestimento . . . . .	47
4.2	Seconda fase: considerazioni preliminari alle misure . . . . .	50
4.3	Terza fase: primo set di misure . . . . .	52
4.3.1	Misura 1 . . . . .	52
4.3.2	Misura 2 . . . . .	54
4.3.3	Misura 3 . . . . .	56
4.3.4	Misura 4 . . . . .	58
4.3.5	Misura 5 . . . . .	60
4.3.6	Misura 6 . . . . .	62
4.4	Quarta fase: secondo set di misure . . . . .	65
4.4.1	Misura 7 . . . . .	65
4.4.2	Misura 8 . . . . .	66
4.4.3	Misura 9 . . . . .	68
4.4.4	Misura 10 . . . . .	70
4.5	Valutazioni . . . . .	74
<b>5</b>	<b>Conclusioni</b>	<b>77</b>
	<b>Bibliografia</b>	<b>79</b>
	<b>Ringraziamenti</b>	<b>83</b>

# Elenco degli acronimi

<b>ACK</b>	Acknowledgment
<b>ABIC</b>	AnyBus Integrated Circuit
<b>ADC</b>	Analog to Digital Converter
<b>AR</b>	Application Relationship
<b>ASCII</b>	American Standard Code for Information Interchange
<b>ASE</b>	Application Service Element
<b>ASIC</b>	Application Specific Integrated Circuit
<b>CIM</b>	Computer Integrated Manufacturing
<b>CP</b>	Communication Profile
<b>CPU</b>	Central Processing Unit
<b>CR</b>	Communication Relationship
<b>CSMA/CA</b>	Carrier Sense Multiple Access with Collision Avoidance
<b>CSMA/CD</b>	Carrier Sense Multiple Access with Collision Detection
<b>DAC</b>	Digital to Analog Converter
<b>DCS</b>	Distributed control system
<b>DHCP</b>	Dynamic Host Configuration Protocol
<b>DI</b>	Digital Input
<b>DO</b>	Digital Output
<b>ERP</b>	Enterprise Resource Planning

<b>FIFO</b>	Firs Input First Output
<b>FTP</b>	File Transfer Protocol
<b>GSD</b>	General Station Description
<b>HMI</b>	Human Machine Interface
<b>HRT</b>	Hard Real-Time
<b>HTTP</b>	Hyper Text Transfer Protocol
<b>ICT</b>	Information and Communication Technology
<b>ID</b>	Device Ident
<b>IEEE</b>	Institute of Electrical and Electronic Engineers
<b>IP</b>	Internet Protocol
<b>IRT</b>	Isochronous Real Time
<b>ISO</b>	International Organization for Standardization
<b>ISO/OSI</b>	International Organization for Standardization (ISO)/Open Systems Interconnection (OSI)
<b>LAN</b>	Local Area Network
<b>LLC</b>	Logical Link Control
<b>MAC</b>	Medium Access Control
<b>MES</b>	Manufacturing Execution System
<b>MIF</b>	Monitor Interface
<b>NMT</b>	Network Management
<b>OPC</b>	OLE for Process Control
<b>OSI</b>	Open Systems Interconnection
<b>PC</b>	Personal Computer
<b>PI</b>	Profibus International
<b>PLC</b>	Programmable Logic Controller
<b>PNI</b>	Profibus Network Italia
<b>PNIO</b>	Profinet IO
<b>RS232</b>	Recommended Standard 232



---

<b>RT</b>	Real Time
<b>RTE</b>	Real Time Ethernet
<b>RTU</b>	Remote Terminal Unit
<b>SCADA</b>	Supervisory Control And Data Acquisition
<b>SCI</b>	Serial Communications Interface
<b>SDU</b>	service data unit
<b>SNMP</b>	Simple Network Management Protocol
<b>SRT</b>	Soft Real-Time
<b>SSC</b>	Synchronous Serial Channel
<b>TCP</b>	Transmission Control Protocol
<b>TCP/IP</b>	Transmission Control Protocol/Internet Protocol
<b>TDMA</b>	Time Division Multiple Access
<b>UDP</b>	User Datagram Protocol



# Introduzione

I dispositivi impiegati per l'automazione nei sistemi di gestione della produzione o di controllo di processo lavorano all'interno di un sistema informativo che, secondo il modello di riferimento Computer Integrated Manufacturing (CIM), può essere suddiviso in quattro livelli gerarchici: di pianificazione, di gestione del processo, di cella e di campo.

Le reti di comunicazione dei livelli più alti, ovvero quelli di pianificazione e di gestione del processo, devono soddisfare esigenze diverse da quelle presenti nei più bassi livelli di cella e di campo. Questi ultimi infatti, contrariamente ai primi, sono caratterizzati dallo scambio di modeste quantità di dati ma spesso richiedono rigidi vincoli temporali ed un'elevata immunità ai disturbi elettromagnetici che possono essere presenti in ambiente industriale. La risposta a tale esigenza è arrivata e viene già impiegata da una ventina d'anni ed ha preso il nome di "bus di campo".

In genere la comunicazione con mezzo bus di campo è di tipo seriale e quindi due fili sono sufficienti a tale collegamento: ne scaturisce una drastica riduzione dei cavi presenti aumentando la semplicità del cablaggio, specie in sistemi complessi di grosse dimensioni. I bus di campo sono reti di comunicazione basate su vari protocolli ottimizzati per il collegamento di dispositivi di automazione di processo quali Programmable Logic Controller (PLC), microcontrollori, sensori, attuatori ed altre apparecchiature coinvolte nel processo produttivo. Come la maggior parte delle reti di comunicazione, anche i bus di campo si rifanno al modello di riferimento OSI del quale però, tipicamente implementano solo il livello applicazione ed i due livelli dello strato fisico, allo scopo di realizzare un'architettura di comunicazione sufficientemente snella ma nel contempo robusta (Stack OSI riportato in figura 1). Lo sviluppo e l'impiego di questi sistemi di comunicazione industriale prende il via a partire dagli anni '80 e oggi, tra i bus di campo più diffusi e conosciuti, si annoverano *Profibus*, *CANbus*, *DeviceNet* e *ControlNet*.

Ulteriore passo in avanti è stato fatto negli ultimi dieci anni con la comparsa dei "bus di campo di seconda generazione": l'evoluzione tecnologica richiede di poter gestire in modo sempre più integrato tutto ciò che riguarda l'automazione, quindi da un lato grosse quantità di dati e dall'altro criticità temporali sempre più stringenti. La tecnologia *Ethernet*, permettendo la realizzazione di reti più estese, comprensive di un

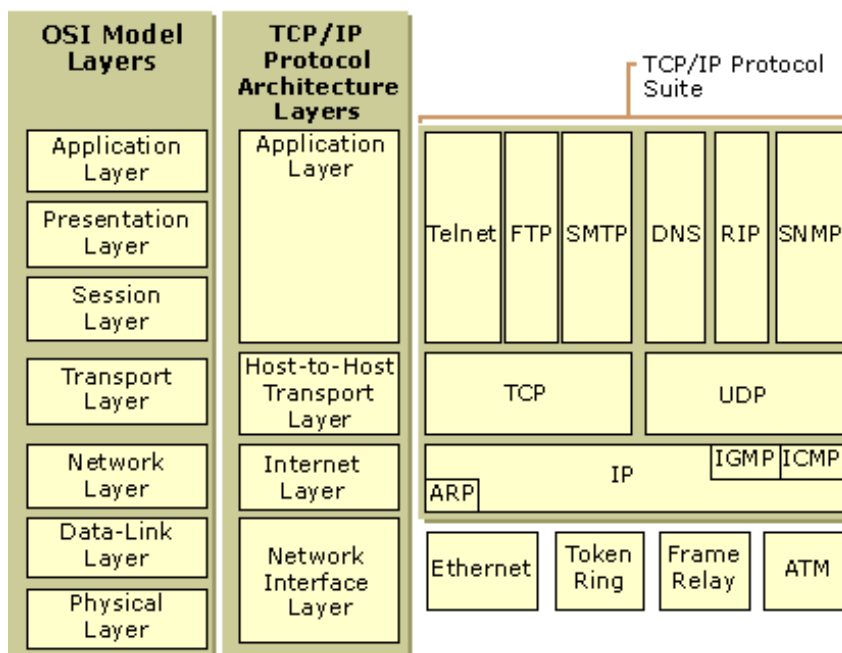
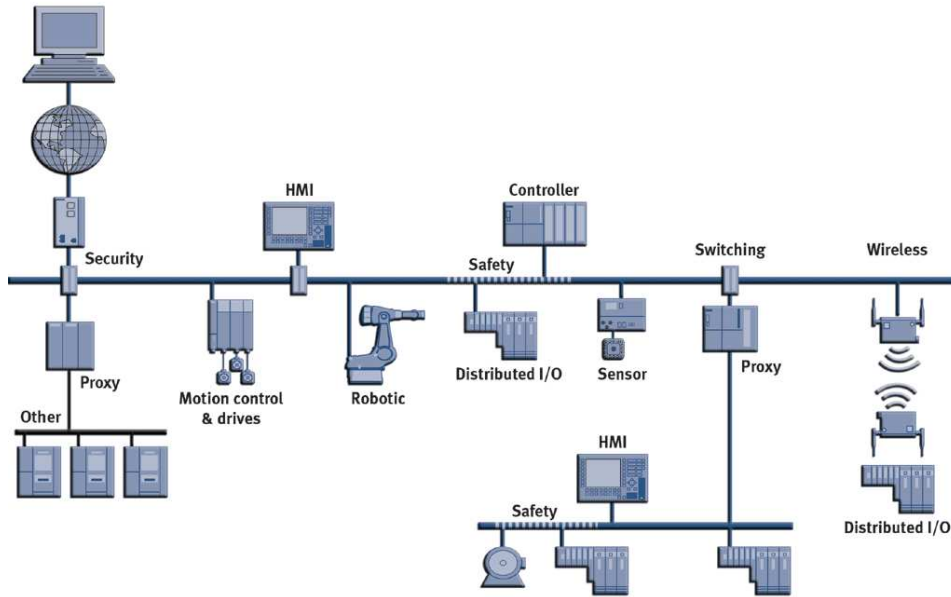


Figura 1: Rappresentazione dello stack OSI affiancato allo stack TCP/IP.

maggior numero di nodi e caratterizzate da una velocità di trasmissione più elevata, è stata prescelta per superare i limiti di prestazione caratteristici dei bus di campo tradizionali. A questi benefici di carattere puramente tecnico si affiancano vantaggi economici di contenimento dei costi legati all'impiego di dispositivi standardizzati, alla semplificazione della manutenzione ed alla riduzione del rischio di obsolescenza che, invece, caratterizzano tutte le soluzioni proprietarie. Per contro, la tecnologia Ethernet non è nata per la gestione del traffico real-time; di conseguenza, la sua applicazione a livello di campo ha richiesto l'introduzione di adattamenti specifici che, sviluppati da diversi operatori del settore, hanno dato origine ad un certo numero di protocolli Ethernet industriali, tra i quali, ad esempio, *Profinet*, *Ethernet Powerlink* ed *EtherCAT*. Questo tipo di approccio consente appunto lo sfruttamento sinergico sia dei vantaggi offerti da una classica rete Ethernet (elevate quantità di dati trasmissibili, utilizzo di standard di comunicazione già affermati nel mondo office, ecc.) sia di quelli offerti dai Bus di campo ormai già consolidati (determinismo, flessibilità di struttura, diagnostica, cablaggio seriale, etc.). La flessibilità d'impiego e la naturale predisposizione all'integrazione dei vari livelli gerarchici di comunicazione, costituiscono il punto di forza delle soluzioni Ethernet industriali che sembrano destinate sempre più ad assumere un ruolo dominante nel panorama delle reti di comunicazione industriale.

Tra le varie soluzioni che si stanno imponendo è emersa la tecnologia sviluppata da PI per far interagire segmenti separati di bus di campo attraverso Ethernet e collegarli ai sistemi di gestione dell'azienda secondo una struttura rigorosamente gerarchica. Una delle caratteristiche di *Profinet* che risulta fondamentale per il raggiungimen-

to dell'integrazione verticale tra livello di campo ed elementi MES superiori, è la possibilità di utilizzare un unico supporto fisico per la trasmissione di diverse classi di segnale. Mediante l'uso di *Profinet* è, infatti, possibile utilizzare contemporaneamente comunicazione basata su *TCP/IP* e comunicazione Real-Time scalabile per i controllori su un unico cavo, periferia decentrata e Motion Control. Un altro aspetto



**Figura 2:** Esempio grafico di una rete basata su protocollo *Profinet*.

di fondamentale importanza per il successo di *Profinet* sul mercato è la salvaguardia di precedenti investimenti. In questo caso ci si riferisce alla possibile integrazione dei Bus di campo tradizionali (*Interbus* e *Profibus*) magari già presenti in campo, con una struttura *Profinet IO*. Questa possibilità si rivela come estremamente efficace per ampliamenti o modifiche di impianti esistenti e gestiti mediante Bus di campo, laddove si voglia con queste modifiche sfruttare i vantaggi di una tecnologia a base Ethernet Industriale senza agire sul progresso, ma anche per la realizzazione di sistemi globali di automazione industriale a tecnologia mista, in cui si vadano a sfruttare al meglio ed in modo sinergico i vantaggi di ogni singola tecnologia. Dopo una breve panoramica sulle caratteristiche dell'Ethernet industriale sarà introdotta la soluzione presentata dal consorzio PI. Su questo tipo di tecnologia si è basato il lavoro di questa tesi: è stato testato ed utilizzato un sistema *Profinet IO* nel laboratorio di Navigazione autonoma del Dipartimento di Ingegneria dell'Informazione dell'Università di Padova. Questo lavoro di tesi è stato possibile grazie a **Weightpack S.p.a.**, azienda promotrice di questo progetto, che ha fornito tutto l'occorrente per la creazione della rete e per i successivi test operativi. Weightpack è specializzata nella progettazione di macchine per l'imbottigliamento ed è detentrica di alcuni brevetti: merito del centro aziendale di R&S in continua ricerca di soluzioni migliori e tecnologicamente avanzate messe a disposizione da un mercato sempre più dinamico

---

in materia di automazione industriale. L'azienda ha in programma un piano di migrazione da *Profibus* a *Profinet IO*, e per questo motivo vuole verificare le effettive prestazioni di alcune soluzioni proposte dal mercato e capire se tali prodotti siano o meno adatti al loro tipo di esigenze. In particolare in questa tesi sono stati presi in considerazione nei test il chip *Anybus-IC* di HMS ed un sistema Supervisory Control And Data Acquisition (SCADA) *ZENON* di Copa-Data.

# Presentazione Weightpack

WeightPack S.p.A. è un'azienda italiana che fornisce un'ampia gamma di macchine e servizi per linee da imballaggio a valore aggiunto, con la leadership tecnica in settori come lattiero-caseari, alimenti, bevande, detergenti liquidi, prodotti chimici, per la cura personale, olio da motore, etc.



**Figura 3:** Foto della sede di Weightpack.

WeightPack è stata fondata da Carlo Corniani, conosciuto come l'inventore della prima riempitrice a peso netto a controllo completamente elettronico nel 1979. Le macchine di ultima generazione sono state progettate sfruttando al massimo la profonda conoscenza nel riempimento a peso netto maturata in molti anni di esperienza. Avendo a che fare con il riempimento di una grande varietà di liquidi non gasati, l'azienda ha messo a punto un gran numero di macchine imbottigliatrici, tutte basate sulla tecnologia di riempimento a peso netto. Inoltre, soprattutto per quanto riguarda i prodotti alimentari, viene data molta importanza all'ambiente stesso di riempimento; si può passare da un ambiente di riempimento classico per prodotti liquidi non sensibili che sono distribuiti a temperatura ambiente fino a prodotti che richiedono un ambiente ad una determinata temperatura, il più possibile sterile usan-

---

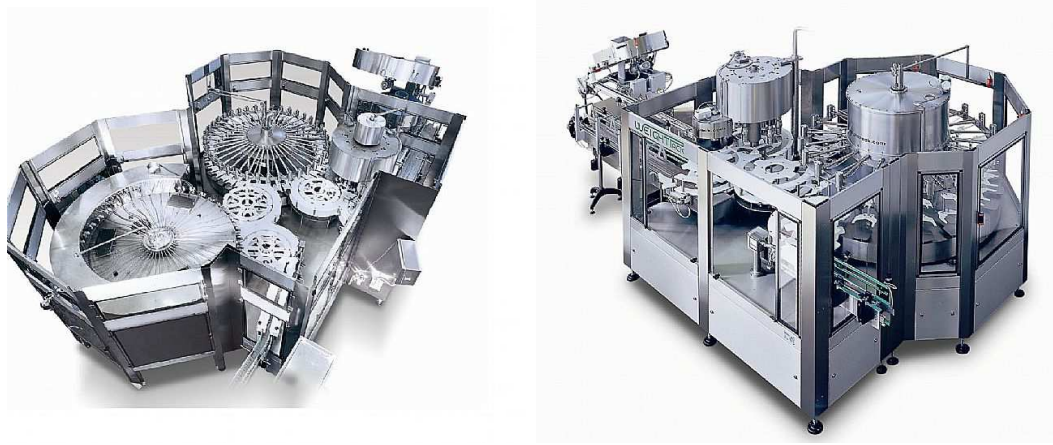
do tecniche di filtraggio del tutto innovative. Parametri critici di processo e al fine di garantire la qualità del prodotto sono:

- garanzia di riempimento accurato sia per il consumatore sia per il fornitore;
- capacità di calcolare la tara, il peso netto di riempimento di ogni bottiglia e la deviazione standard di entrambi;
- garanzia di un riempimento accurato grazie al continuo monitoraggio del peso netto di ogni bottiglia durante la fase di riempimento;
- capacità di autocorrezione su ogni rotazione della riempitrice per ottenere un preciso peso finale;
- capacità di controllare e registrare le statistiche su ogni contenitore;
- rifiuto automatico di contenitori riempiti in modo sbagliato;
- capacità di interrompere il riempimento nei seguenti casi:
  - peso dosato errato;
  - tara fuori tolleranze;
  - incremento peso;
  - tappo mal posizionato/danneggiato.

Dovendo competere a livello internazionale, i loro prodotti sono soggetti a continue evoluzioni tecnologiche per garantire elevate prestazioni nel tempo e soddisfare appieno le richieste del cliente finale. Per questo motivo WeightPack si avvale, oltre che a team di ingegneri interni, di una stretta collaborazione con centri di ricerca universitari, mettendo a disposizione degli studenti risorse e competenze maturate in anni di attività. WeightPack coniuga elevata professionalità e competenze con metodologie di lavoro che prevedono la massima cura dei particolari di progettazione e costruttivi. L'obiettivo è quello di riempire un contenitore nel minor tempo possibile prestando attenzione a parametri rigorosi come può essere la quantità di prodotto.

Ogni stazione di riempimento è costituita da una piastra nella quale è installato un sensore di peso. Il contenitore si posiziona sulla piastra e da questo momento ha inizio la fase di riempimento. Per prima cosa viene rilevato il peso del contenitore (tara); se il valore rientra nel range stabilito si apre la valvola attraverso la quale fluisce il prodotto da imbottigliare. Tutta questa fase viene monitorata campionando il peso finché non si raggiunge il valore desiderato. Ogni singola stazione è provvista di un sistema che controlla costantemente il peso del prodotto; le stazioni, tramite una scheda con microcontrollore e chip di interfaccia, comunicano al PLC tutti i dati che identificano univocamente ogni singolo contenitore riempito. Successivamente si tappa il contenitore sempre attraverso operazioni meccanizzate. Alla fine del processo avviene la selezione per lo stoccaggio solo se il prodotto risulta conforme ai parametri della ricetta. Si è così descritto in breve il principio di funzionamento di una macchina riempitrice Weightpack. Nota non secondaria è che ogni macchina





**Figura 4:** Foto di due modelli diversi di riempitrici prodotte da Weightpack.

viene progettata per soddisfare le esigenze del cliente e quindi ogni macchina è soggetta a modifiche di progettazione più o meno sostanziali a seconda della richiesta: molte di queste modifiche sono diventate brevetti industriali Weightpack. A riprova di quanto l'azienda si stia impegnando nella continua innovazione del prodotto c'è da annoverare il recente inserimento in *Kilometro Rosso*, parco scientifico tecnologico situato nel bergamasco e sede di reparti di ricerca di importanti aziende. Weightpack crede nello sharing di informazioni tecnologiche con le aziende che fanno parte di *Kilometro Rosso* e che sono sempre state all'avanguardia nella R&S. Come si può leggere nel sito aziendale:

“Essere insediati in *Kilometro Rosso* significa essere inseriti in un contesto ambientale e architettonico di prestigio, dotato di strutture e impianti d'avanguardia, a stretto contatto con le principali realtà di ricerca e industriali di una delle regioni più ricche d'Europa. Significa vivere in una realtà che raccoglie le migliori competenze in ambito tecnico-scientifico nazionali e internazionali, che favorisce lo scambio di know-how e conoscenze, creando un terreno fertile per la produzione e la promozione di nuove idee. Ogni centro di ricerca, società e laboratorio operante nel Parco è messo nelle condizioni di sviluppare al meglio tutte le sinergie, collaborazioni e relazioni di proprio interesse e può usufruire di una vasta gamma di servizi comuni, materiali e immateriali ad alto valore aggiunto. Tali servizi vengono predisposti, organizzati ed offerti dalla società di gestione e sviluppo del Parco o da altri soggetti da essa delegati, in base a specifiche esigenze e secondo accordi che vengono stipulati con le realtà insediate nel Parco.”



# Capitolo 1

## La comunicazione industriale

La comunicazione industriale è un argomento particolarmente complesso da trattare e per questo si è pensato di dedicare in questa tesi un breve capitolo, non con lo scopo di farne un'approfondita trattazione bensì di dare al lettore una discreta visione d'insieme sulla comunicazione industriale allo stato attuale. In un sistema automatizzato infatti, il bus di campo indica il tipo di comunicazione adottato per lo scambio di dati tra i diversi dispositivi costituenti il sistema sia di campo, sia di controllo. In questo settore si colloca il protocollo *Profinet IO*, che resta il bus di campo utilizzato in questo lavoro di tesi.

### 1.1 I bus di campo

La centralizzazione delle funzioni di controllo è stato il sistema più usato fin dagli anni '80: tutte le funzioni di controllo vengono svolte da un'unica unità, alla quale si collegano i diversi componenti del sistema d'automazione (trasduttori, attuatori, elementi di regolazione, sensori), dislocati nelle diverse aree dell'impianto. Il collegamento tra gli elementi periferici e l'unità centrale avviene per mezzo di cablaggi dedicati con tecnologia analogica utilizzando linee di comunicazione punto-punto (4/20 mA). I costi dei cablaggi, i problemi di disturbi elettromagnetici e di schermatura, la convivenza a bordo quadro di elementi fortemente disomogenei, hanno spinto la ricerca verso soluzioni di controllo alternative.

Con l'espressione bus di campo si vuole descrivere una nuova forma di comunicazione digitale dedicata ai sistemi a basso livello, quali sensori, attuatori, trasduttori, etc. con la quale si va a sostituire nell'industria la vecchia comunicazione analogica punto-punto. La modalità di comunicazione numerica presenta vantaggi sia in termini di possibilità di trasmettere un maggior quantitativo di informazioni, sia per la migliore potenzialità di interfacciamento con altri apparati. Fin dai primi anni '80 si sono manifestati nel mercato i primi sistemi di comunicazione proprietari ad interfaccia numerica, seriale o parallela, che facilitavano la comunicazione tra sistemi periferici complessi (come telecamere e attuatori) e l'unità centrale (come PLC e PC). Tale

modalità di connessione, tipicamente RS232 con protocollo ASCII, risulta adatta per la fase di configurazione e predisposizione del dispositivo del trasduttore; in modalità operativa (con basso flusso di informazioni) ci si ritrova però ad avere un quantitativo di cavi paragonabile a quello con tecnologia analogica tradizionale. La tecnologia a bus, in cui un unico conduttore connette più sistemi, consente una notevole riduzione dei cablaggi: il bus rappresenta il mezzo fisico migliore per trasportare dati. I modi per collegare insieme più apparati sono svariati e verrà scelta la topologia migliore a seconda dell'uso; le più affermate in tal senso sono quella ad anello, a stella e a bus lineare, come da figura 1.1.

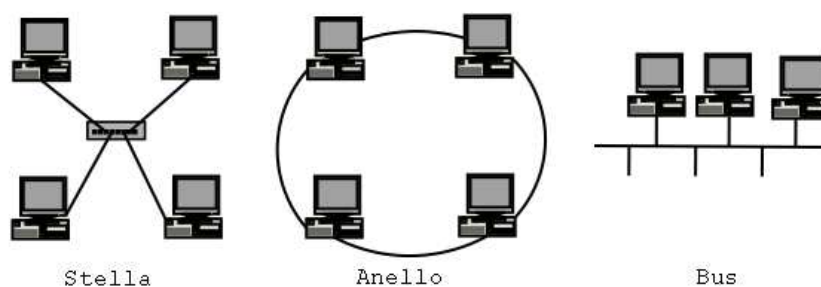


Figura 1.1: Topologie di rete.

Di contro si ha che in un sistema più popolato servono delle regole più stringenti per gestire lo scambio dei dati: bisogna prevedere l'accesso contemporaneo al mezzo e regolamentare i dati in modo che vengano trasmessi in maniera rapida ed affidabile. Tutto questo insieme di regole si chiama **protocollo**. I più recenti protocolli industriali sono stati sviluppati per poter gestire, oltre la trasmissione dei dati, anche una serie di informazioni atte al controllo e alla supervisione della rete stessa: queste informazioni devono essere disponibili in tempi brevissimi ed in grado di viaggiare in entrambe le direzioni; la tecnologia analogica risulta insufficiente a tali scopi già per impianti di medie dimensioni. Riassumendo si possono elencare i vantaggi introdotti con l'utilizzo dei bus di campo:

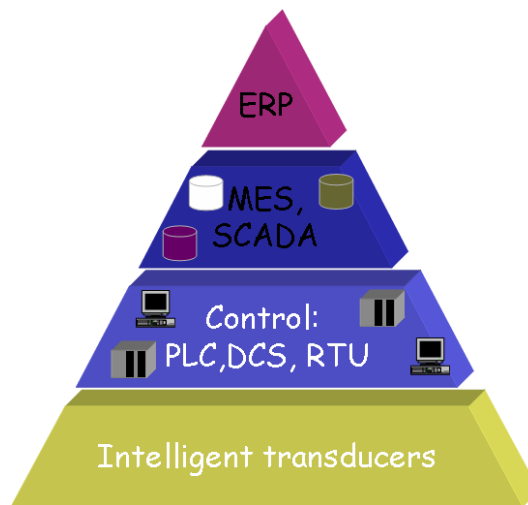
- riduzione massiccia dei cavi e relativo costo di cablaggio;
- semplificazione della messa in servizio;
- possibilità di collegamento di prodotti di costruttori diversi;
- raccordo di moduli diversi su una stessa linea;
- flessibilità di estensione dell'impianto;
- distanze di copertura tramite bus maggiori di quelle coperte con cablaggio standard;
- estensione dei campi di applicazione;
- riduzione dei costi globali.

Se da un lato c'è una semplificazione in molti passaggi, dall'altro lato si evidenziano degli aspetti ostici derivanti dalla crescita tecnologica:

- una necessità di conoscenze superiori per l'utilizzo di queste nuove tecnologie;
- un investimento in strumentazione ed accessori (tools di sviluppo, monitoraggio e diagnosi) che possono diventare abbastanza onerosi.

## 1.2 Sistemi di comunicazione

Un sistema di comunicazione nell'automazione industriale contiene al suo interno più settori di applicazione e di interesse. Per descrivere al meglio questa struttura si è fatto uso di una rappresentazione grafica piramidale chiamata CIM: "produzione integrata di fabbrica" è l'integrazione automatizzata tra i vari settori di un sistema di produzione (progettazione, ingegnerizzazione, produzione, controllo della qualità, pianificazione della produzione e marketing) al fine di minimizzare i tempi di sviluppo di un prodotto, ottimizzare la gestione delle risorse ed eventualmente essere flessibili per coprire quanto più possibile il mercato. Come si può vedere in figura 1.2 la



**Figura 1.2:** CIM - schema della produzione integrata di fabbrica.

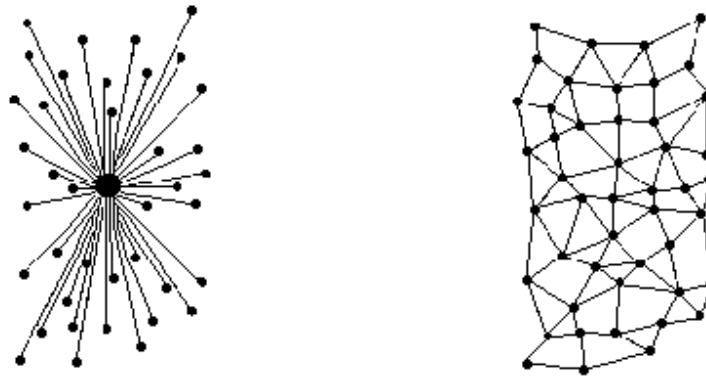
struttura contiene al suo interno quattro livelli:

- LIVELLO 1: costituito di trasduttori (fine corsa, sensori, encoder, sonde, etc.) ed attuatori (relè, sistemi di avviamento, motori, etc.) posti nell'impianto di produzione;
- LIVELLO 2: vi sono i controllori di cella (PLC, Distributed control system (DCS), Remote Terminal Unit (RTU)) che controllano trasduttori ed attuatori quali sono collegati o direttamente o mediante bus di campo;

- LIVELLO 3: chiamato anche livello di sistema, è composto da SCADA, che controllano e supervisionano i sistemi distribuiti del livello 2 e da sistemi tipo CAD/CAM e MES il cui compito è elaborare i dati di produzione e presentarli in forma adeguata al management;
- LIVELLO 4: Enterprise Resource Planning (ERP), “pianificazione delle risorse d’impresa”, si tratta di un sistema di gestione che integra tutti i processi di business rilevanti di un’azienda (vendite, acquisti, gestione magazzino, contabilità, etc.); con la riduzione dei costi per l’Information and Communication Technology (ICT), si sono sviluppate applicazioni che aiutano i business manager ad implementare questa metodologia nelle attività di business come controllo di inventari, tracciamento degli ordini, servizi per i clienti, finanza e risorse umane.

### 1.3 Sistemi distribuiti

I sistemi distribuiti sono sistemi composti da un gran numero di CPU distribuite nello stesso impianto e connesse tra loro da una rete ad alta velocità; il nome è in contrapposizione ai sistemi centralizzati formati da un’unica unità operativa, dalla sua memoria e da un insieme di periferiche e terminali ad essa collegati (schema in figura 1.3). Al giorno d’oggi l’automazione industriale e di processo impone l’utilizzo



**Figura 1.3:** *Rappresentazione di una rete centralizzata (a sinistra) e di una rete distribuita (a destra).*

di dispositivi intelligenti, atti a svolgere funzioni di automazione autonoma ausiliarie. L’evoluzione dell’uso di questi sistemi distribuiti in un impianto, ha dimostrato da subito alcuni vantaggi: la distribuzione di applicazioni che richiedono lavorazioni su macchine distanti e separate, l’affidabilità dell’impianto nel suo complesso, la scalabilità tra le varie tipologie di componenti impiegati, la modularità dell’impianto stesso. Non sono di certo mancati i difetti, reputabili principalmente ad una maggiore complessità del software di gestione rispetto a sistemi centralizzati e la necessità di un efficiente sistema di comunicazione deterministico.

Tutto ciò deriva dai progressi tecnologici nel campo delle comunicazioni digitali (inclusi i protocolli usati in internet) e nello sviluppo dei microprocessori. Il vantaggio sfruttato nell'automazione e nella gestione computerizzata dei reparti è strettamente legato al concetto di *integrazione*: una struttura aziendale diviene più snella e più efficiente se i terminali dedicati a compiere una determinata funzione possono comunicare in Real-Time con altri terminali riservati ad altre funzioni, magari dislocati in aree diverse dell'impianto.

## 1.4 Il fenomeno Ethernet

Nei primi tempi Ethernet non veniva preso in considerazione nell'applicazione di cella poichè i bus impiegati erano in gran parte proprietari e di consolidato impiego; l'utilizzo di Ethernet nell'ambito dell'industria era legato quasi esclusivamente a scambio di dati dal quadro elettrico verso i livelli di fabbrica e/o impianto legati alla produzione fino alla parte finanziaria dell'azienda. Nel corso degli ultimi anni, si è però assistito ad un'evoluzione del fenomeno: attraverso l'utilizzo di particolari protocolli studiati appositamente per sopperire alle mancanze dei requisiti strettamente necessari per la comunicazione di campo, si è arrivati alla risoluzione di problemi di determinismo e real time delle reti su Ethernet.

### 1.4.1 Ethernet tra Real time e determinismo

La caratteristica principale del mezzo Ethernet è quella di operare a basso livello (2- DataLink) con il protocollo Carrier Sense Multiple Access with Collision Detection (CSMA/CD); con tale protocollo non può essere garantito alcun determinismo temporale.

Una delle parole chiave principali è funzionamento real-time: il sistema deve reagire entro un intervallo di tempo prevedibile e non superiore ad un certo valore prefissato. Un'altra definizione importante è quella di determinismo: esecuzione di operazioni in istanti predefiniti ed eventualmente periodici. Una rete deterministica e real time si chiama *rete isocrona*, [12].

Il tempo impiegato dal telegramma per arrivare a destinazione dipende dal traffico sulla rete, da eventuali errori che possono richiedere ritrasmissioni dello stesso etc. Prima di trasmettere su un mezzo dotato di accessi multipli (MA, Multiple Access), ogni nodo verifica, rilevando la presenza della portante, che nessun altro stia utilizzando il canale (CS, Carrier Sense). Una volta iniziata la trasmissione i nodi ascoltano ciò che si trova sul canale per verificare che non vi siano collisioni (CD, Collision Detect), ossia che altri nodi non abbiano iniziato a trasmettere prima di accorgersi che era già in atto una comunicazione.

Una rete basata sul protocollo CSMA/CD è intrinsecamente non deterministica e pertanto impedisce di fatto di poter calcolare entro quanto tempo il dato trasmesso arrivi alla stazione di destinazione. Oggi, con una accurata progettazione della rete e grazie alla sempre maggiore velocità di trasmissione, è possibile fare in modo che le collisioni spariscono completamente e che i ritardi siano del tutto limitati e preve-

dibili.

L'impiego di Ethernet come bus di campo è risultato penalizzato nell'impiego di cella anche per il fatto che il numero di byte trasmessi è troppo elevato rispetto al numero di byte che contengono l'informazione. Questi due limiti sono stati affrontati e superati con l'introduzione delle seguenti innovazioni tecnologiche.

**Fast Ethernet:** termine che indica un numero di standard Ethernet che trasportano il traffico alla velocità di 100 Mbps rispetto alla velocità originale Ethernet di 10 Mbps. Molte applicazioni industriali non risentono un sostanziale giovamento nel semplice incremento di velocità di scambio: le performance di device di campo sono limitate più dalla bassa velocità del loro processore o del loro firmware piuttosto che dalla velocità della rete in sé. Migliora la gestione delle collisioni ma non risolve il problema del determinismo.

**Switch:** dispositivo hardware che opera al livello Data-Link (livello 2) del modello ISO/OSI. Inoltrando il messaggio solo al destinatario permette di risolvere il problema delle collisioni. In un hub i dati inviati da un nodo vengono inviati in broadcast a tutti i nodi della rete; lo switch è una sorta di hub intelligente che può leggere e processare l'indirizzo di destinazione dei dati entranti e mandarli unicamente alle porte richieste: più nodi possono trasmettere o ricevere contemporaneamente [4]. Vengono affidate allo switch gli oneri della gestione delle congestioni; il Flow Control, meccanismo atto a limitare il carico della rete che elimina la possibilità di perdere pacchetti in porte congestionate; la VLAN, che permette allo switch di raggruppare in modo logico più device e di isolare il traffico tra i gruppi; la autonegotiation, cioè il riconoscimento automatico e negoziazione sia della velocità di trasferimento (10 – 100 Mbit/s), sia del tipo di funzionamento (full-duplex/half-duplex). A queste si aggiungono funzionalità di diagnostica, sicurezza, analisi di rete, mirroring delle porte, ridondanza, etc.

La classificazione tradizionale degli switch prevede la divisione in tre grosse classi (Store & Forward, Cut-Through e Fragment-Free). Quello più affidabile è denominato Store & Forward, esegue un controllo completo di ogni pacchetto ricevuto prima di inoltrarlo alla destinazione; il prezzo da pagare è un ritardo variabile. Gli switch Cut-Through ricevono e memorizzano solo i byte iniziali del pacchetto necessari a capire la destinazione dello stesso; subito dopo inizia la trasmissione minimizzando così l'introduzione di ritardi. Infine, i Fragment-Free sono la versione più sicura dei Cut-Through poiché controllano che il pacchetto sia composto di almeno 64 byte impedendo l'inoltro di frame spuri.

**Full duplex:** introduce percorsi separati per la trasmissione e la ricezione dei dati e si applica a segmenti che contengano non più di due device: così trasmissione e ricezione possono avvenire simultaneamente senza il rischio di collisione.

**Priorità:** compreso in un livello tra 0 e 7 viene utilizzato dagli switch per stabilire l'importanza nello smistare i messaggi memorizzati nel loro buffer e riducendo



ulteriormente l'incidenza del traffico non critico sui dati di maggior importanza. La definizione dei livelli di priorità viene realizzata utilizzando tre bit di una estensione di 4 byte prevista per gli standard 802.1p e 802.1q che modificano la trama originaria di Ethernet.

Anche in caso di utilizzo di switch aderenti allo standard IEEE 802.1q/1p, che introduce il concetto di priorità tra i frame, non risolve il problema. Infatti, l'uso dei priority tag per rendere un pacchetto più prioritario rispetto ad un altro garantisce che un frame ad alta priorità sorpassi un frame a bassa priorità nello switch, ma non permette di interrompere un frame già in fase di trasmissione per trasmetterne uno prioritario. In definitiva il problema dei ritardi random non è risolto.

Concludendo, si può affermare che il solo utilizzo di Ethernet standard non garantisce il determinismo in caso di reti trafficate [5]. Per garantire applicazioni di Hard real-time, richieste specialmente nel motion control, sono necessarie delle migliorie da apportare all'hardware per ridurre questo jitter; il controllore di rete offerto da Siemens, ad esempio, per le reti Profinet si basa su un Application Specific Integrated Circuit (ASIC) con elevate prestazioni denominato ERTEC che permette un'accurata sincronizzazione hardware tra i dispositivi [16].

Un corretto sistema RT dovrà garantire sia il funzionamento corretto del sistema sia la sua puntuale e deterministica esecuzione. I sistemi RT generalmente sono suddivisi in due principali sottocategorie [4]:

- **SOFT REAL-TIME (SRT)**: sistema che se non rispetta la sua scadenza (sfonda la deadline) provoca un danno non irreparabile al funzionamento del sistema;
- **HARD REAL-TIME (HRT)**: sistema che nel caso superi temporalmente la sua deadline provoca un danno irreparabile al sistema bloccando l'esecuzione dello stesso.

### 1.4.2 Ethernet industriale

Ethernet viene già da tempo impiegato nel settore industriale, ma solo a livello amministrativo: era la rete che collegava i vari uffici oppure collegava gli impianti di produzione a livello di area, non era la rete su cui viaggiavano i dati critici dell'automazione. Con il termine **Industrial Ethernet** ci si riferisce in genere a quelle applicazioni che utilizzano Ethernet per le comunicazioni a livello di campo, ossia di PLC e periferia.

Non essendoci però uno standard imposto da direttive, molti costruttori hanno implementato una soluzione consona alle proprie esigenze con la conseguente proliferazione di svariati protocolli applicativi. L'introduzione di Ethernet a livello di cella e di campo nel panorama industriale garantirebbe più flessibilità, nuovi mezzi trasmissivi (come la tecnologia wireless), la possibilità di usare tutti i protocolli ICT e quindi anche il supporto diretto di Internet che consentirebbe di creare interfacce remote per i dispositivi usando un normalissimo Web browser. Ma sarebbe inaccettabile se l'adozione della tecnologia Ethernet portasse alla perdita di tutte quelle

caratteristiche essenziali nel mondo della comunicazione industriale quali l'affidabilità, la robustezza EMC e meccanica, la comunicazione tempo-deterministica, le azioni tempo-sincronizzate tra le periferiche di campo e la frequenza ed efficienza nello scambio dei dati.

Nei processi a livello di campo la comunicazione deve soddisfare le condizioni appena citate, richiedendo esigenze temporali assai stringenti e spesso molto differenti tra loro. Ci sono due tipologie di casi temporalmente critici: processi periodici e processi asincroni. Nel primo caso l'evento deve avvenire con particolare regolarità temporale e quindi con una determinata frequenza per tutto il periodo di produzione; la comunicazione deve garantire che l'informazione prodotta giunga a destinazione prima della generazione dell'informazione successiva. Nei processi asincroni l'istante temporale in cui si verificherà l'evento non è noto, si considera solo la possibilità che esso si verifichi; il sistema di comunicazione ne deve garantire la trasmissione nei tempi più brevi possibili, dal momento della generazione fino alla ricezione da parte dei processi consumatori.

### 1.4.3 Real Time Ethernet

Tutti i sistemi Real-Time hanno un certo livello di jitter (variazione sul tempo ideale); questo deve essere misurabile in modo tale da poter garantire le prestazioni del sistema, cioè tale sistema deve garantire che un'elaborazione termini entro un dato vincolo temporale o entro una data scadenza (imposta dalla *deadline*). Bisogna quindi garantire la fattibilità della schedulazione: questo concetto è alla base della teoria dei sistemi deterministici ed oltre a garantire la validità dei dati scambiati ne devono assicurare la tempestività.

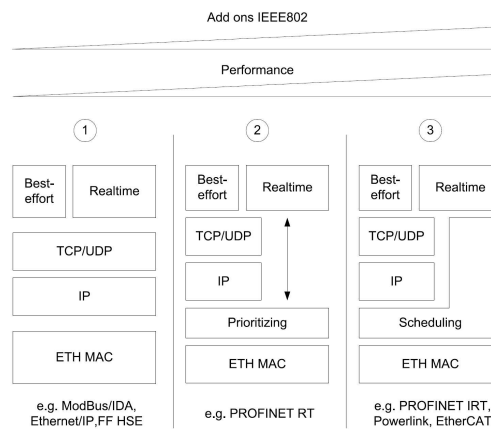
Per definire una comunicazione Real Time Ethernet (RTE) si sono introdotti alcuni parametri chiamati Performance Indicators, indicati nella normativa IEC 61784 [3], [15] e riportati di seguito:

- **DELIVERY TIME:** tempo necessario per trasmettere una service data unit (SDU) da un nodo sorgente ad un nodo destinatario. Viene misurato a livello applicazione tenendo conto del tempo di trasmissione e di tutti i tempi d'attesa.
- **NUMERO DI NODI:** ogni Communication Profile (CP) impone un numero massimo di stazioni che possono essere collegate alla rete affinché sia garantita la comunicazione Real-time.
- **TOPOLOGIA DI RETE:** disposizione fisica della rete (vedi figura 1.1 a pagina 10).
- **NUMERO DI SWITCH TRA DUE NODI:** ogni CP impone un numero massimo di switch tra due stazioni.
- **THROUGHPUT RTE:** rappresenta l'insieme di dati scambiati tramite un collegamento nel tempo di un secondo.

- **BANDA NON RTE:** indica la percentuale di banda che può essere destinata alla comunicazione non Real-Time in un collegamento.
- **ACCURATEZZA DI SINCRONIZZAZIONE:** indica la deviazione temporale massima dei clock appartenenti a due diversi nodi della rete.
- **TEMPO DI RECUPERO RIDONDANTE:** tempo massimo tra il fallimento di un singolo evento ed il ripristino del sistema considerando un fallimento permanente (si stabilisce una sorta di *redundancy recovery time*).

#### 1.4.4 Protocolli Industrial Ethernet

Diversi approcci da parte di altrettante case produttrici hanno dato origine a più protocolli RTE, studiati e sviluppati a seconda delle prestazioni e delle esigenze degli utilizzatori. Si può comunque raggrupparli in tre categorie, riportate in figura 1.4; dalla categoria 3 alla 1 aumentano le prestazioni del protocollo ma anche le funzionalità che si devono aggiungere a livello data link. Per una massima riduzione dei tempi si è reso necessario un intervento nella procedura di schedulazione a livello Medium Access Control (MAC). Di seguito i diversi tipi di approccio.



**Figura 1.4:** Vari protocolli basati su Ethernet industriale.

**Incapsulamento:** si ha quando un pacchetto dati ad alto livello viene inglobato in un pacchetto standard Transmission Control Protocol (TCP) o User Datagram Protocol (UDP). In questo modo si lascia funzionare normalmente Ethernet e i livelli Transmission Control Protocol/Internet Protocol (TCP/IP) dello stack di comunicazione e le nuove funzionalità vengono inserite a livello applicazione (Livello 7). Esempi di RTE che usano questo approccio sono Modbus/TCP, Ethernet/IP, P-NET, Vnet/IP, Foundation Fieldbus HSE (vedi figura 1.4, colonna 1).

**Incapsulamento diretto a livello 2:** alcuni protocolli utilizzano la struttura Ethernet standard a Livello 1 e il MAC a livello 2 ma incapsulano i dati direttamente senza far uso del protocollo TCP/IP. Applicazioni SRT possono usare l'incapsulamento diretto, ma lo stack TCP/IP permette funzionalità aggiuntive per la gestione della rete usando tool standard. Ethernet fornisce uno schema nel quale il traffico SRT viene incapsulato con maggior priorità rispetto al TCP/IP. Queste realizzazioni RTE non alterano l'hardware di comunicazione ethernet in alcun modo, ma sono realizzate specificando uno speciale tipo di protocollo (Ethertype) nel frame ethernet (Per esempio l'Ethertype di Profinet è `0x8892`). Esempi di RTE che usano questo approccio sono TCnet, EPA, Profint CBA, Profinet RT (vedi figura 1.4, colonna 2).

**Ethernet modificata:** L'unico modo per assicurare la funzionalità IRT con risposte sotto i millisecondi è cambiare il modo in cui opera Ethernet. Viene mantenuta la connessione fisica di ethernet ma la funzionalità del protocollo è direttamente modificata al livello 2 per fornire capacità di comunicazione deterministica. Questo richiede lo sviluppo di interfacce hardware con speciali ASIC o FPGA. Quindi il determinismo è implementato attraverso una procedura che fornisce cicli di trasmissione fissi e con una sincronizzazione ciclica di tutti i nodi. Molto spesso la funzionalità di switch è già integrata all'interno del dispositivo di campo. Le modifiche sono obbligatorie per tutti i device all'interno del segmento di rete Real-time, permettendo comunque la trasmissione di traffico non real-time. Esempi di RTE che usano questo approccio sono: Ethernet Powerlink, Sercos, EtherCAT e Profinet IRT (vedi figura 1.4, colonna 3).

# Capitolo 2

## Il protocollo di comunicazione industriale Profinet IO

Profinet è un protocollo innovativo che è stato sviluppato dal consorzio Profibus International (PI) ed è diventato uno standard internazionale con la normativa IEC 61158. I suoi obiettivi principali sono la riduzione dei costi di progetto e di messa in servizio dell'impianto e l'integrazione tra la nuova automazione di dispositivi basati su Ethernet e quelli tradizionali basati sui bus di campo. Allo stato attuale, per quanto riguarda la sola parte di comunicazione dati, PROFINET raggruppa sotto il suo nome due famiglie di protocolli entrambi basate su Ethernet [6], Profinet CBA e Profinet IO:

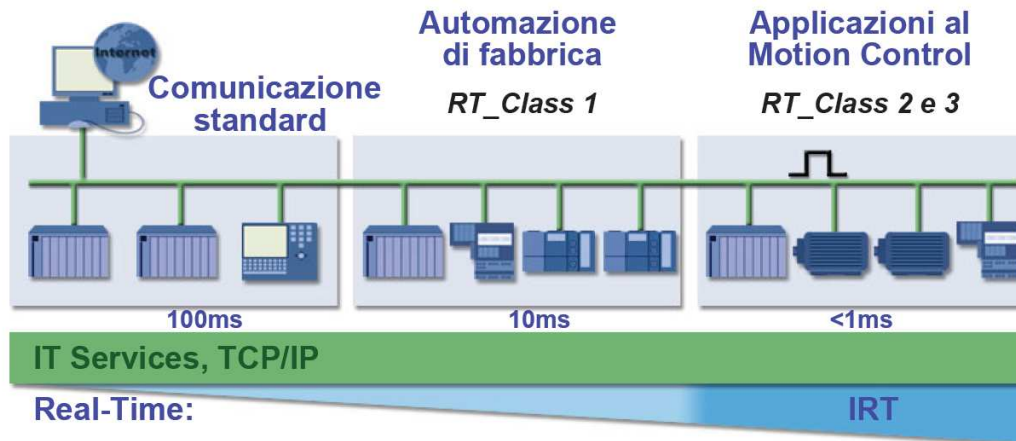
- PROFINET CBA (Component Based Automation) è dedicato all'integrazione ad alto livello, per esempio tra differenti linee di produzione, ed offre il supporto per l'integrazione con OLE for Process Control (OPC), Internet ed il WEB, nonché con PROFIBUS e con tutti gli altri bus di campo.
- PROFINET IO è stato progettato per dotare i dispositivi di campo di interfaccia Ethernet e garantire delle performance deterministiche e real-time.

In questo capitolo verrà descritto lo standard Profinet IO nel suo complesso, scendendo nel dettaglio solo dove necessario alla comprensione di fenomeni analizzati successivamente nel lavoro di laboratorio. Per una trattazione più approfondita dell'argomento si rimanda a [8], [13], [17], [18] nonché al sito web del consorzio PI <http://www.profibus.com>.

### 2.1 Scalabilità e prestazioni

Già si è discusso sul problema del determinismo nelle reti industriali tracciandone una panoramica abbastanza ampia; nello specifico andremo ad analizzare come Profinet IO risolve tale problema e quali altre soluzioni tecnologiche lo rendono un sistema

affidabile e prestante.



**Figura 2.1:** La comunicazione in Profinet.

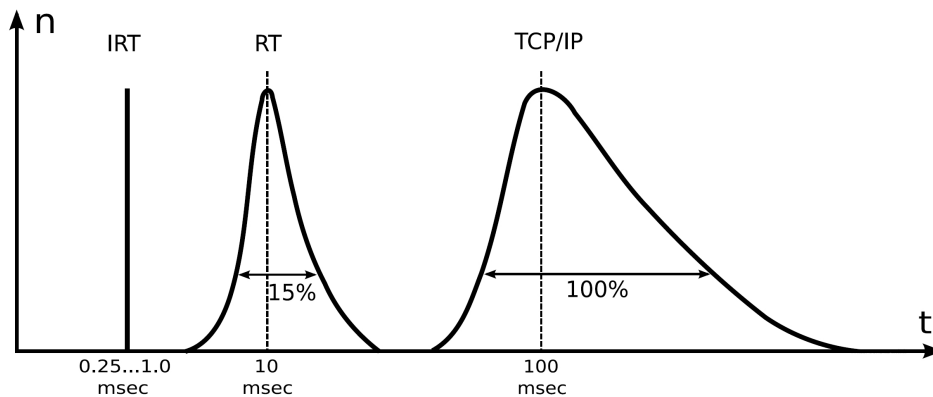
Profinet IO (PNIO) è stato progettato per provvedere al controllo remoto di dispositivi di Input/Output. È un protocollo pensato per il campo che ne soddisfa le caratteristiche critiche di periodicità e real-time. Utilizzando lo Switched Ethernet, tale protocollo raggiunge le più alte prestazioni in termini di tempistiche. I target primari sono la sostituzione dei tradizionali moduli distribuiti di Input-Output, che sono interconnessi attraverso i bus di campo, e l'estensione di applicazione all'ambito del motion control. Supporta tutti i tipi di comunicazione (Non-RT, Real Time (RT) e Isochronous Real Time (IRT)) ed è dedicata ad applicazioni con limiti critici per le condizioni di tempo reale (si fa riferimento al motion control). L'obiettivo di PNIO è quello di trasmettere dati, attraverso l'utilizzo di strutture e apparecchiature di comunicazioni semplici, rispettando performance prefissate. Con il duplice scopo di sfruttare la vasta esperienza accumulata nel settore dei bus di campo e di proteggere investimenti precedenti, PROFINET IO ha incorporato in sé concetti quali l'architettura modulare e la programmazione dei dispositivi di campo, ereditandoli direttamente da PROFIBUS. Per ottenere le alte prestazioni richieste nello scambio dati, sono stati necessari alcuni elementi innovativi:

- sono state previste priorità per lo scambio dati;
- accesso differenziato da parte di più sistemi di controllo per lo stesso dispositivo di campo;
- descrizione in un file GSD delle funzionalità del singolo dispositivo di campo;
- estensione del modello PROFIBUS DP: la metodologia di accesso Master-Slave (di PROFIBUS DP) è stata convertita al modello Provider-Consumer, ove per produttore si intende colui che genera i dati, mentre per consumatore colui che riceve e utilizza i dati quando stabilisce la connessione con il produttore.

In più la comunicazione avviene su un unico mezzo trasmissivo. In figura 2.1 sono evidenziate le tre classi con prestazioni RT di Profinet IO.

## 2.2 Classi di comunicazione

Le varie classi di comunicazioni gestite in Profinet IO sono state divise nelle seguenti quattro categorie.



**Figura 2.2:** Rappresentazione grafica delle prestazioni temporali delle quattro classi di comunicazione supportate da Profinet IO.

**Non-RT:** viene considerato per la trasmissione di dati di configurazione o per implementare cicli (con periodo maggiore ai 100ms) che tolleri fino al 100% di variabilità. Per la comunicazione si fa affidamento sui protocolli standard quali TCP e UDP. Vengono supportati anche servizi ICT come la gestione della rete Dynamic Host Configuration Protocol (DHCP), diagnosi di rete Simple Network Management Protocol (SNMP), accesso remoto tramite WEB Hyper Text Transfer Protocol (HTTP) e funzioni di posta elettronica e-MAIL.

**RT Classe 1:** può realizzare cicli di pochi millisecondi (tipicamente 1-64 ms) a variabilità contenuta (15%). Essa si basa sullo standard Ethernet 802.1Q in cui i dati vengono trasmessi tramite telegrammi Ethernet con priorità; per l'implementazione è sufficiente hardware ethernet standard.

**RT Classe 2:** chiamato anche IRT-Flex, i frame vengono trasmessi con o senza sincronizzazione (andando a comportarsi come la Classe 1). La sincronizzazione consente tempi di ciclo di 1 ms e tutti i telegrammi possono essere inviati e ricevuti in una finestra temporale definita in fase di progetto e riservata al solo traffico IRTflex. Il jitter massimo si attesta sui 10  $\mu$ s, l'indirizzamento dei messaggi è fatto in base all'indirizzo MAC e serve hardware ASIC per la sincronizzazione.

**RT Classe 3:** denominata IRT-Top è il protocollo di comunicazione più performante riservato ad applicazioni Motion Control. Rispetto alla RT Classe 2 si ha una più stringente pianificazione temporale del traffico ottenuta tramite la sincronizzazione tra il livello di rete e l'applicazione software che gestisce i dati. Il ciclo è selezionabile da un minimo di  $250 \mu s$  ad un massimo di  $4 ms$  con intervalli di  $125 \mu s$ . Il jitter è garantito ben al di sotto del  $\mu s$ , così facendo si conosce l'istante pressochè esatto di invio e ricezione di un telegramma IRT-Top. Il completo determinismo ottenuto prevede precisioni di sincronismo maggiori gestite direttamente dall'hardware. Una stazione (Master Sync) ha il compito di inviare il telegramma alle altre stazioni appartenenti al dominio di sincronizzazione. La conoscenza della topologia di rete è fondamentale per la determinazione dei ritardi di trasmissione ed in caso di modifiche sulla topologia o sui rapporti di comunicazione, è necessario riprogettare la configurazione.

Nel grafico a figura 2.2 sono riportate le quattro classi appena descritte, la classe RT 2 e RT 3 sono raggruppate sotto l'acronimo IRT, la classe 1 denominata semplicemente con RT e la classe non-RT è riportata come TCP/IP. Si può pertanto affermare che i livelli di prestazioni di PROFINET coprono l'intero campo delle applicazioni per l'automazione. In particolare la caratteristica di scalabilità e di standardizzazione della comunicazione sono uno dei punti di forza che permettono l'applicazione del protocollo a tutti i livelli della piramide CIM.

## 2.3 Classi dei dispositivi

La metodologia di accesso Master/Slave utilizzata in PROFIBUS DP è stata convertita con il modello Provider/Consumer. Questo modello sta prendendo piede grazie allo sviluppo di applicazioni distribuite come Profinet. Nei sistemi Master/Slave uno solo dei nodi svolge la funzione di gestore della rete mentre gli altri sono semplici utilizzatori del canale di comunicazione (cioè possono ricevere o trasmettere informazioni ma in funzione del controllo del bus fatto dal master). Una rete master/slave in cui ci sono più dispositivi che fungono da master si dice multi-master: in tal caso è necessario un sistema di arbitraggio per risolvere i conflitti che nascono quando più master richiedono il controllo del canale di comunicazione.

Una rete in cui non ci sono dispositivi che si dedicano al controllo degli accessi al bus è detta di tipo Producer/Consumer. Qualunque nodo può acquisire momentaneamente il controllo del bus ed iniziare una trasmissione (interfaccia Producer), gli altri nodi in questa fase si attiveranno in ricezione (interfacce Consumer). In questo tipo di protocollo ogni stazione deve essere provvista della lista di variabili che può produrre. Durante la normale trasmissione dati, ad ogni stazione può essere richiesto di produrre una variabile o può richiedere essa stessa la produzione di una variabile. PROFINET specifica quattro tipi di dispositivi:

- **IO-CONTROLLER:** dispositivo controllore sul quale gira il programma di automazione; ha il controllo del processo distribuito di uno o più dispositivi di



campo ed è responsabile della configurazione e della parametrizzazione dei dispositivi associati ad esso. Riceve dati e allarmi e li processa all'interno del programma utente. Esso rappresenta una stazione centrale intelligente, come un PLC.

- IO-DEVICE: dispositivo di bus di campo remoto, che viene assegnato ad un IO-Controller. È configurato da un IO-Controller o da un IO-Supervisor e trasmette ciclicamente i suoi dati di processo al controllore. Esso provvede anche ad informare un IO-Controller su condizioni di diagnostica o di allarme.
- IO-SUPERVISOR: dispositivo/PC di programmazione con funzioni di configurazione e di diagnostica, che può scambiare dati con IO-Controller e IO-Device; ha accesso temporaneo ai dispositivi di campo durante il processo di comunicazione.
- IO-PARAMETER-SERVER: dispositivi usati per lo scambio di dati di configurazione rilevanti per le applicazioni con IO-Device.

Il traffico generato da IO-Supervisor e da IO-Parameter-Server avviene tipicamente in fase di offline, quindi non sono necessarie prestazioni real time. Durante il funzionamento, i dispositivi periferici (IO-Device) trasferiscono le informazioni di input al controllore; esso le elabora e trasferisce le informazioni di output ai dispositivi periferici. Al contrario, lo scambio di dati tra IO-Controller e IO-Device è collegato al compito di automazione in via di svolgimento e perciò ha esigenze temporali critiche.

Dal punto di vista della comunicazione tutti i dispositivi su Ethernet sono trattati allo stesso modo. Il protocollo prevede che ad ogni dispositivo venga assegnato un numero univoco di identificazione, il Device Ident (ID) Number, composto di 32 bit. È stato specificato anche un modello uniforme per i dispositivi IO-Device PROFINET, il quale permette una configurazione dei singoli moduli del dispositivo. Tale modello è simile per caratteristiche a quello di PROFIBUS DP. Un dispositivo di campo può essere modulare, cioè includere degli slot fisici per l'inserzione di moduli, oppure non espandibile (slot virtuali): questa modellizzazione permette di indirizzare i canali di I/O presenti nel sistema con indice univoco. PROFINET IO inoltre distingue tra slot, inteso come spazio fisico ove inserire periferiche (modulo) in un IO-Device, e subslot, che rappresenta un ulteriore livello di indirizzamento. Generalmente uno slot può contenere più subslot, e la capacità del modulo viene definita da chi fornisce il dispositivo di campo, il quale è responsabile della mappatura della funzionalità del dispositivo. Ogni slot e subslot rappresenta un modulo IO e ha un numero fisso di bit di input e output.

Il protocollo PROFINET IO prevede di poter implementare il trasferimento di tre tipologie di dati:

- TRASFERIMENTO CICLICO DI DATI CRITICI DI IO: tali dati vengono immagazzinati nell'area di indirizzamento I/O del controllore; si tratta di dati ciclici di

input e output, che vengono inviati sul canale real-time. I dati di I/O sono trasmessi a cicli di tempo prefissati, la cui frequenza di aggiornamento può essere diversificata per ciascun dispositivo;

- **TRASFERIMENTO ACICLICO DI ALLARMI:** trasferimento che avviene sul canale real-time; è prevista anche la possibilità di assegnare una priorità agli allarmi;
- **TRASFERIMENTO ACICLICO DI DATI NON CRITICI:** si tratta di parametrizzazione, configurazione, lettura di informazioni diagnostiche che sono trasferiti attraverso canali standard sulla base di UDP/IP.

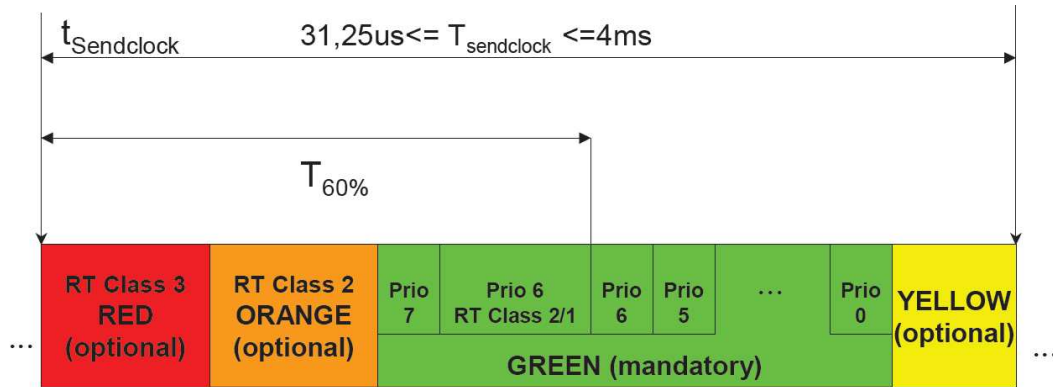
Le caratteristiche di un IO Device sono riportate nel file, in formato XML, chiamato General Station Description (GSD) [11]. All'interno del file si devono trovare proprietà dell'IO-Device, i moduli inseriti (quantità e tipo), i dati di configurazione individuali dei moduli (quantità e tipo), i parametri dei moduli (tipo 4-20 mA) ed il testo con errori per la diagnostica.

## 2.4 Il periodo di trasmissione Profinet

PROFINET IO effettua lo scambio di dati su un ciclo altamente ripetibile (descritto nello standard IEC 61158-5-10) e illustrato nella figura 2.3. Un messaggio di sincronizzazione, sync frame, individua l'inizio del ciclo. Le diverse fasi del ciclo sono:

- **RED phase:** In questa fase solo i messaggi RT Classe 3 possono essere inviati in un fissato momento attraverso un percorso definito con topologia rigida. Ciò comporta che tutti i dispositivi sappiano precisamente quando e verso quale porta è permesso comunicare o mettersi in ascolto.
- **ORANGE phase:** Solo frame di tipo RT Classe 2 sono trasmessi durante questa fase. Anche questi messaggi devono essere scambiati con precise scadenze temporali sebbene la topologia della rete non sia definita rigidamente.
- **GREEN phase:** Questa fase è composta dai messaggi che utilizzano le priorità Ethernet scandite dallo standard IEEE 802.1Q. La comunicazione durante la fase verde (GREEN) viene usata da dispositivi RT Classe 1, da dispositivi RT Classe 2 che non hanno terminato di spedire il frame nella ORANGE phase e dai protocolli TCP e UDP.
- **YELLOW phase:** Questo periodo di transizione è utilizzato dallo stesso tipo di traffico della fase GREEN. Durante questo periodo (YELLOW phase) solo i frame che possono essere trasferiti completamente entro la fine della fase stessa sono effettivamente trasmessi.

Una porzione della fase verde è lasciata alla comunicazione non-RT, essa ha bassa priorità e tempi di ritardo molto variabili, infatti il traffico di tipo IP è usato per grandi trasferimenti di dati e la sola cosa significativa è la banda. In PNIO la fase



**Figura 2.3:** *Ciclo di comunicazione Profinet IO.*

Non-RT occupa almeno il 40%. Nella RT classe 3, essendo la più performante, non devono esserci ritardi di trasmissione o ricezione. Il tool della rete di configurazione calcola il percorso per ogni frame del ciclo; di conseguenza il traffico isocrono deve essere gestito con una rigida topologia, che deve essere determinata durante la configurazione del sistema. Sui frame non è effettuato alcun controllo sull'indirizzo MAC e il riferimento fondamentale sono le tempistiche. Se un messaggio atteso non arriva a destinazione, un frame "dummy" è trasmesso con l'indicazione dell'errore; mentre se un frame non può essere trasmesso ritorna indietro. Proprio per queste motivazioni la durata della RED phase deve essere maggiore della più lunga propagazione di ritardo tra sorgente e destinazione di qualsiasi frame IRT. È da notare che la RT classe 3 presenta un jitter molto basso, dal momento che ogni causa di incertezza viene evitata.

Nella RT Classe 2 la topologia delle rete può essere cambiata e non è necessaria la pianificazione di ciascun messaggio, inoltre viene utilizzato l'indirizzo MAC, come di consueto. Nel caso di traffico ingente nella rete, i frame vengono bufferizzati e inviati appena possibile.

In RT Classe 1 è utilizzata la gestione della priorità e, se i pacchetti sono dotati di VLAN tag, trasportano un valore di priorità da 0 a 7: gli switch gestiscono questo traffico in base a questa scala di priorità.

## 2.5 Lo stack di comunicazione

I dispositivi con funzionalità tempo-critiche possono negoziare protocolli di comunicazione con capacità real-time che riducano il tempo di elaborazione dello stack: in questo modo Profinet IO soddisfa le elevate richieste real-time disposte sul sistema di comunicazione. L'eliminazione di alcuni livelli di protocollo riduce la lunghezza del messaggio, il tempo per la preparazione dei dati da trasmettere ed i tempi di procedura dell'applicazione. Un ulteriore beneficio si ha considerando che la potenza del processore del dispositivo per la comunicazione viene fortemente ridotta. Nel

caso di Profinet RT avviene un incapsulamento a livello 2 (si veda 1.4.4) mentre nel caso IRT si rientra nella categoria Ethernet modificata (sempre 1.4.4) dove viene però richiesta una speciale interfaccia hardware per poter fornire una comunicazione deterministica ed isocrona di alto livello.

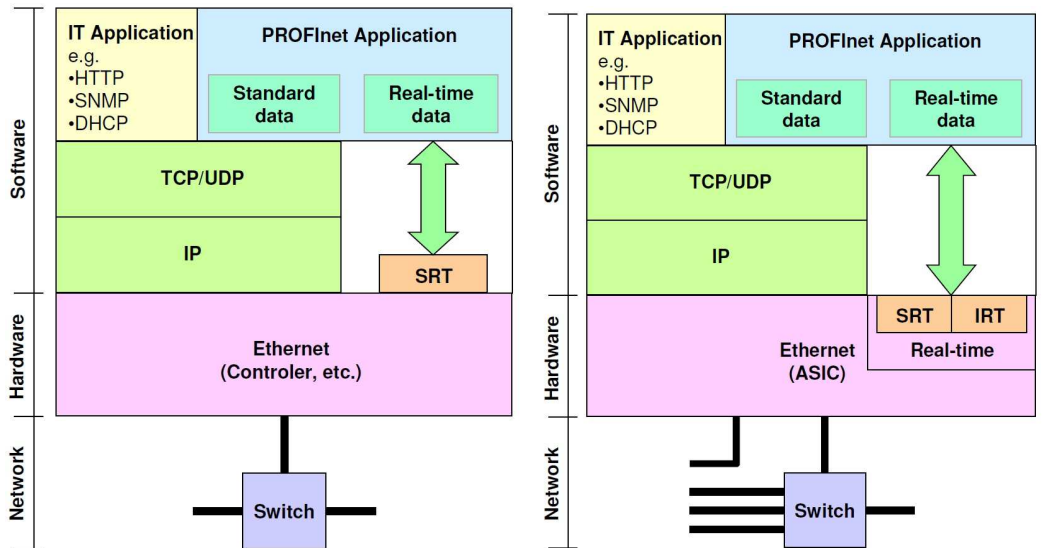
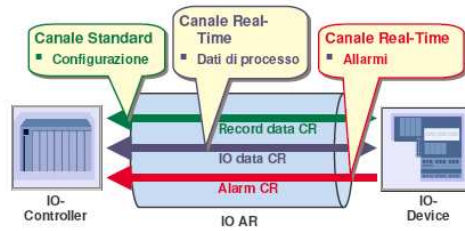


Figura 2.4: Stack di comunicazione RT (a sinistra) e IRT (a destra).

## 2.6 I canali di comunicazione

Per garantire l'interoperabilità tra diverse applicazioni non è sufficiente stabilire un canale di comunicazione solo ma è stato necessario implementare ulteriori specifiche. Lo standard PROFINET IO implementa routine automatiche attraverso la loro suddivisione in application process, eseguite da differenti stazioni distribuite sulla rete. Le applicazioni di processo comunicano con lo scambio di application object, strutture formali che contengono informazioni riguardo i processi reali della propria stazione. Il concetto di Application Service Element (ASE) è il cardine su cui si basa il livello Application del modello di riferimento ISO/OSI nel sistema Profinet. ASE provvede a fornire una serie di servizi per realizzare la comunicazione tra application process distribuiti e lo scambio di application object. In una tipica "IO Configuration", come in figura 2.5, vengono stabilite le istanze di relazione tra gli IO-Controller e gli IO-Device, Application Relationship (AR). L'istanza di relazione IO-AR contiene diverse relazioni di comunicazione Communication Relationship (CR) attraverso le quali sono trasferite la configurazione, i dati di processo e gli allarmi. Il dispositivo IO-Controller trasferisce i dati di parametrizzazione e di configurazione del dispositivo IO-Device assegnato su "Record data CR". La trasmissione ciclica dei dati di ingresso e di uscita è implementata su "IO data CR", ed il monitoraggio delle relazioni di comunicazione avviene attraverso la stessa verifica dei messaggi ciclici



**Figura 2.5:** *Canali di comunicazione a livello Application.*

ricevuti. Per esempio, se un frame atteso in ingresso non si manifesta entro 3 cicli, l'IO-Controller determina l'errore o il guasto del rispettivo IO-Device. Gli eventi aciclici sono trasmessi su "Alarm CR" agli IO-Controller; una trattazione completa ed esaustiva la si può trovare in [13].

## 2.7 Scambio di dati ciclico

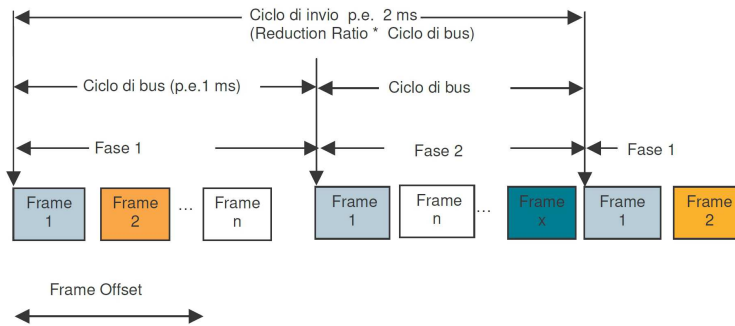
La IO CR implementa lo scambio dei dati ciclico. La frequenza di aggiornamento di ogni dispositivo è costante e prefissata in fase progettuale; tale sistema consente di evitare al controllore l'elaborazione dei dati in base all'effettiva urgenza delle informazioni trasferite dai vari IO Device. Per la descrizione di uno scambio dati ciclico, lo standard PNIO definisce alcuni termini:

- **CICLO DI BUS:** la base dei tempi è di  $31.25 \mu s$ ; può arrivare fino a  $4 ms$ , valore tipico è  $1 ms$ .
- **REDUCTION RATIO** indica ogni quanti cicli viene spedito il frame da un IO Device all'IO Controller; il ciclo di invio è calcolato moltiplicando il Reduction Ratio per il Ciclo di bus.
- **FRAME OFFSET** indica l'offset di un dato frame dall'inizio del ciclo bus al momento in cui viene spedito il frame stesso.
- **FASE** indica in quale ciclo di bus il frame corrispondente verrà spedito.

Nella figura 2.6 viene rappresentato quanto appena descritto.

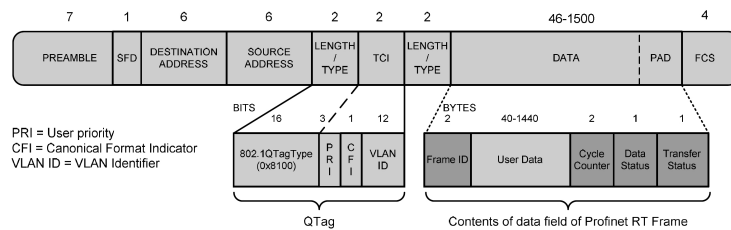
### 2.7.1 Il ciclo RT

Per soddisfare alle esigenze del Real-time PNIO ha ottimizzato il canale RT Classe 1 (vedi 2.2). Lo scopo della suddivisione del ciclo RT è quello di dividere la banda del sistema per evitare il sovraccarico della linea e la perdita di pacchetti. PNIO gestisce e limita l'invio di pacchetti e l'uso di priorità nei frame; per prevenire sovraccarichi ed aumentare l'uso della banda è possibile sincronizzare il ciclo di invio del provider con quello di ricezione del consumer: questa si basa sul contatore di ciclo con un'accuratezza di un ciclo di bus [16]. Attraverso il Reduction Ratio e la fase di trasmissione si possono distribuire i pacchetti RT nel tempo. Il traffico di rete viene



**Figura 2.6:** *Traffico ciclico dei dati in PNIO.*

così controllato e limitato attraverso questi procedimenti facendo attenzione che il processo che si occupa dell'invio di pacchetti non spedisca alcun pacchetto RT se all'inizio del nuovo ciclo esiste una situazione di sovraccarico dell'interfaccia locale; e tenendo presente che non è generalmente concesso utilizzare più di un certo valore di banda onde evitare il sovraccarico del sistema. Un pacchetto di PNIO tipo RT è riportato in figura 2.7 dove:



**Figura 2.7:** *Struttura del frame Profinet IO Real Time Classe 1.*

- PREAMBLE: sequenza di bit usata per la sincronizzazione con il ricevitore;
- START FRAME DELIMITER: sequenza di bit che specifica l'inizio del frame;
- DESTINATION ADDRESS: è l'indirizzo MAC del nodo destinatario;
- SOURCE ADDRESS: è l'indirizzo MAC del nodo sorgente;
- TYPE: 0x8100 esprime il Virtual LAN Header;
- VLAN: contiene le specifiche per la VLAN;
- ETHERTYPE: per PNIO è 0x8892;
- FRAME ID: indica il tipo di frame Profinet, ad esempio RT classe 1, 2 o 3, trasmissione aciclica, etc.;
- DATA: contiene i dati da spedire;

- CYCLE COUNTER: contatore di ciclo, ogni bit rappresenta un incremento temporale di  $31.25\mu s$  (valore che va a definire il tempo di bus descritto in 2.7);
- DATA STATUS: indica lo stato dei dati;
- TRANSFER STATUS: indica se la trasmissione è avvenuta regolarmente;
- FRAME CONTROL SEQUENCE: è la sequenza di controllo della correttezza del frame.

## 2.8 Integrazione dei bus di campo

PROFINET specifica un modello per l'integrazione dei bus di campo esistenti (PROFIBUS, INTERBUS, DeviceNet, etc.) permettendo la costruzione di un sistema misto, in cui coesistono segmenti basati e su bus di campo e su Ethernet: è una vera e propria tecnologia di transizione dai bus di campo a PROFINET.

Il grande numero di sistemi fieldbus esistenti rende necessaria l'integrazione in PROFINET per una ragione di protezione degli investimenti: l'operatore di sistema desidera integrare facilmente i sistemi già installati con quelli nuovi tipo PNIO; il costruttore di macchine vorrebbe utilizzare componenti testati e disponibili per la progettazione su PNIO; infine il produttore di dispositivi vorrebbe poter integrare i componenti di campo esistenti su reti PNIO senza costi per la sostituzione [8].





## Capitolo 3

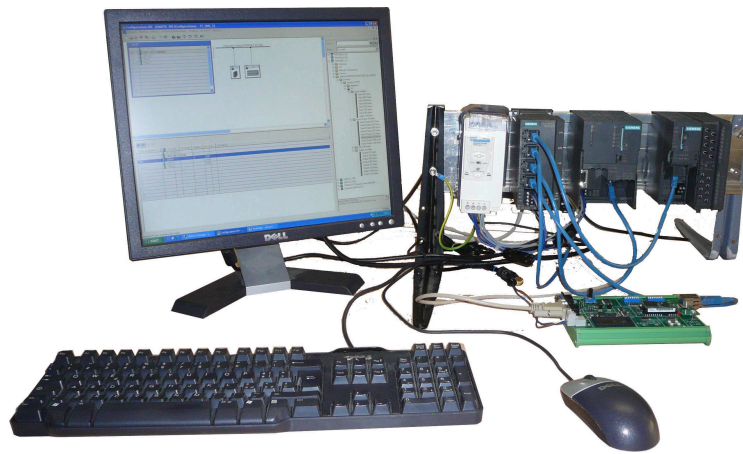
# Strumentazione da laboratorio e dispositivi Profinet IO

Weightpack, come già descritto a pagina 5, è azienda leader nella produzione e commercializzazione di macchine riempitrici a peso netto di sostanze liquide non gasate. Tali macchine, essendo interamente progettate e prodotte negli stabilimenti aziendali, richiedono un know-how non indifferente sia per il funzionamento meccanico, sia per il controllo elettronico del processo. Weightpack ha preferito sviluppare una struttura produttiva leggera facendo ampio uso di terze parti per la costruzione ed il pre-assemblaggio dei moduli, prediligendo in questo modo l'investimento economico nella ricerca e nello sviluppo aziendale. Da questo tipo di filosofia nasce questo lavoro di tesi, sviluppato da Weightpack in collaborazione del Dipartimento d'Ingegneria dell'Informazione dell'Università degli studi di Padova e del CNR.

Allo stato attuale le macchine imbottigliatrici sono controllate e gestite da un PLC, che funge da Master, hanno alcuni Slave decentrati, uno SCADA che ne effettua la supervisione, varie schede con trasduttori di acquisizione dati dal campo e svariati sistemi che controllano attuatori e motori che movimentano la macchina; essendo impiegato un PLC di marca Siemens, vien da sè che il fieldbus più utilizzato nelle macchine è il Profibus.

Come descritto nel capitolo 2, Profinet IO è un nuovo sistema per la comunicazione industriale e Weightpack, focalizzata sull'innovazione tecnologica dei suoi prodotti, non ha potuto di certo stare a guardare da distante l'evolversi dell'automazione industriale. Volendo migrare interamente da un sistema Profibus ad un sistema di comunicazione Profinet risulta necessario reperire dispositivi in commercio che siano in grado di sostituire, od eventualmente integrare, quelli utilizzati e consolidati da ormai molti anni in ambiente Profibus.

In quest'ottica si è deciso di provare e testare il funzionamento di un chip che riesca a collegare il sistema di acquisizione dati della pesatura alla rete Profinet IO che si intende costituire come impianto di trasmissione dati a bordo macchina. In questo capitolo verrà descritto l'ambiente in cui sono stati fatti i test e le misurazioni (Foto in figura 3.1).Viene quindi spontaneo dover per lo meno dare una descrizione dei



**Figura 3.1:** Foto dei dispositivi utilizzati nel laboratorio del Dipartimento di Ingegneria dell'Informazione.

dispositivi PNIO in possesso, i software di supporto utilizzati per questo lavoro e gli strumenti di misura con i quali si son potuti rilevare dati importanti ai fini della caratterizzazione del nostro dispositivo d'interesse.

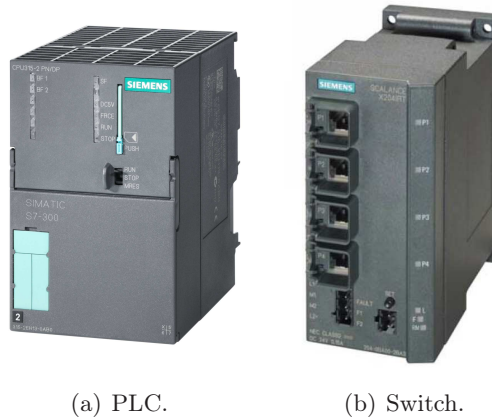
La linea di prodotti più completa presente sul mercato è senz'altro quella di Siemens, la quale propone una serie di controllori PLC e unità di periferia decentrata già muniti di interfaccia Profinet IO, nonché una linea di switch ad alte prestazioni denominati "Scalance" che supportano pienamente le funzionalità di comunicazione richieste.

### 3.1 Il PLC "Siemens S7-300"

Il dispositivo utilizzato come IO-Controller in una prima fase di misure è la CPU 315-2 PN/DP della famiglia Simatic S7-300 di Siemens riportato il figura 3.2(a), la cui documentazione tecnica è consultabile in [20], [21] e [23]. Trattasi di un PLC di media potenza di calcolo, adatto al comando di singole macchine o parti di linee di produzione, è dotato di una memoria interna di tipo RAM di 128 *KByte* in cui viene memorizzato il programma utente e di una memoria esterna di tipo MMC di capacità massima di 8 *MByte*, utilizzata anche per l'archiviazione di dati.

La CPU 315-2 PN/DP integra un'interfaccia di comunicazione MPI/DP per il collegamento Profibus: con questa è possibile effettuare fino a 16 collegamenti con altre stazioni S7-300, dispositivi di programmazione Siemens (PG) o PC ed è possibile scambiare dati con al massimo 32 moduli collegati al bus di campo.

La seconda interfaccia di comunicazione presente sul frontale della CPU è di tipo RJ45 e ne permette il collegamento alla rete Profinet. Questa supporta i seguenti modi di comunicazione:



**Figura 3.2:** Foto della CPU 315 PN/DP e dello Scalance X204IRT utilizzati in laboratorio.

- COMUNICAZIONE S7 riservata allo scambio dati tra stazioni di controllo Simatic.
- RT CLASSE 1 per lo scambio di dati ciclici con IO-Device.
- COMUNICAZIONE PG/PC per l'interfacciamento con dispositivi Human Machine Interface (HMI) e sistemi SCADA.
- COMUNICAZIONE OPEN TCP/IP per Simatic Net OPC Server.

La CPU integra un'interfaccia Web Server accessibile da qualsiasi PC della rete che permette un rapido accesso a dati di diagnostica in pagine in formato html. L'attivazione del Web Server è facoltativa ed avviene in fase di configurazione della CPU; le informazioni disponibili con tale supporto sono:

- informazioni identificative del prodotto;
- contenuto del buffer di diagnostica;
- messaggi (in modalità sola lettura);
- informazioni sul collegamento di rete Profinet;
- stato e tabelle delle variabili;
- pagina iniziale con informazioni generali e di stato della CPU.

Questa CPU non supporta le classi RT 2 e 3. Questa caratteristica non ha tuttavia limitato i test di misura effettuati in laboratorio poichè il device oggetto delle misure di prestazione non avrebbe neanche supportato i protocolli IRT. Per quanto riguarda la comunicazione Profinet IO tale CPU riesce a supportare

intervalli di trasmissione minimi di  $1ms$ <sup>1</sup>: tale valore dipende dalle prestazioni della CPU, dal numero di IO device e dal flusso di dati nella rete.

## 3.2 Lo switch “Scalance”

Siemens propone una serie di switch ad alte prestazioni adatti all'utilizzo in ambienti industriali. Quello a nostra disposizione nel lavoro di tesi è il modello Scalance X204IRT. Come si può vedere dalla foto a figura 3.2(b), ha 4 porte RJ45 ed una doppia alimentazione a  $24 Vdc$ . Le porte Ethernet hanno funzionalità di autonegotiation e di autocrossover: vengono così evitati errori causati dall'errato cablaggio dei cavi di rete Ethernet rendendo più semplice l'installazione all'utente finale. L'X204IRT assicura ottime prestazioni e massima compatibilità con i più comuni protocolli basati su Ethernet e supporta tutte le tre classi Real Time previste dallo standard Profinet, utilizzando il meccanismo switching “Cut Through”. Come per la CPU315, anche lo switch dispone di un Web Based Management accessibile tramite browser o utilizzando servizi TELNET, TFTP o SNMP. Oltre a numerose funzioni di diagnostica e di stato, è possibile:

- impostare le porte per la rindondanza nel caso di collegamento ad anello.
- gestire la funzionalità di e-mail agent per l'invio di e-mail contenenti dati sulla diagnostica.
- configurare e abilitare il Port Mirroring, vale a dire che il traffico di dati viene rispecchiato dalla Mirror Port alla Monitor Port.

## 3.3 L'IO device “ET200M”

È un'unità di periferia decentrata per la gestione di ingressi ed uscite analogiche e/o digitali ed anche questo modulo, come il PLC, è di marca Siemens. La gamma di prodotti proposta da Siemens per la branca dell'automazione industriale vanta di una modularità d'impiego che permette differenti configurazioni con l'utilizzo di pochi moduli. La soluzione utilizzata in fase di misura è un modello ET200M che prevede il collegamento di un modulo con ingressi ed uscite, come l'SM323 (8 Digital Input (DI) e 8 Digital Output (DO)) per il collegamento di interruttori normali e di prossimità a due fili (ad esempio elettrovalvole, relè, piccoli motori, spie luminose, etc.). Il costruttore specifica nel datasheet [21] i tempi di commutazione delle uscite che variano tra i  $100 \mu s$  e i  $500 \mu s$ , mentre i tempi di ritardo di commutazione degli ingressi variano tra  $1.2 ms$  e  $4.8 ms$ . Durante il lavoro di tesi non si disponeva di tale modulo, bensì del SM374, unità di simulazione che, con l'ausilio di un giravite, si poteva impostare come unità a 16 DI, a 16 DO oppure a 8 DI e 8 DO. Quest'ultimo settaggio è stato quello di nostro interesse nella fase di configurazione con Step7: nel

---

<sup>1</sup>Tempo che intercorre tra due comunicazioni successive. Sebbene nel software Step7 di Siemens non vi sia la possibilità di modificare tale valore, esso corrisponde al parametro presentato in 2.7 come Reduction Ratio e, nel caso della CPU 315-2 PN/DP, risulta preconfigurato al valore 32

database del programma infatti non è presente l'unità I/O SM374 in quanto si tratta di un'unità che serve a simularne altre (SM323 nel nostro caso). La connessione fisica avviene tramite *backplane*, standard di connessione Siemens, attraverso il quale vengono scambiati tutti i dati IO e di configurazione.

Il modulo utilizzato è l'IM 153-4 PN grazie al quale sono gestibili fino a 12 unità



**Figura 3.3:** Foto del modulo ET200M complessivo di unità modulare I/O SM374.

modulari di I/O; integra al suo interno uno switch a due porte che permette di inserirlo in una struttura con topologia di rete lineare o ad anello. Supporta servizi Ethernet quali ping, arp, diagnostica di rete SNMP e funzioni di allarme quali allarmi di diagnostica, estrazione/inserimento, manutenzione ed interrupt di processo; permette la comunicazione in tempo reale RT Classe 1 e Classe 2.

### 3.4 L'IO device "Anybus"

Anybus è una linea di prodotto di HMS, azienda che sviluppa, produce e commercializza tecnologie di comunicazione intelligenti progettate per l'applicazione in sistemi di automazione industriale. HMS fornisce soluzioni efficienti ed affidabili per la connessione di dispositivi alle maggiori reti di comunicazione, oltre a prodotti per l'interconnessione tra reti con differenti protocollo. La sede centrale dell'azienda, il centro di sviluppo prodotti ed il principale impianto di produzione si trovano ad Halmstad, in Svezia; HMS ha inoltre creato una rete globale di distribuzione presente in oltre 35 paesi del mondo, Italia compresa. HMS opera sul mercato delle interfacce di reti industriali offrendo prodotti di connettività destinati prevalentemente a sistemi di automazione. L'interfaccia di comunicazione è una combinazione di software ed hardware che permette ai dispositivi di automazione di comunicare con altri dispositivi e sistemi informativi avanzati di gestione dei processi aziendali, quali Manufacturing Execution System (MES) ed ERP. La famiglia Anybus di HMS è oggi una gamma di soluzioni per la connettività di rete industriale tra le più utilizzate: gateway, schede per PC e moduli di interfacce Anybus intercambiabili; sono sistemi di connessione completi e sicuri che consentono di collegare i dispositivi utilizzati con reti Profibus, Profinet, DeviceNet, Modbus-TCP, Modbus-RTU,



Figura 3.4: Gamma di prodotti Anybus di HMS e relativi protocolli di rete supportati.

Ethernet/IP, EtherCAT, CC-Link, ControlNet, CANOpen, Interbus, FIPIO, J1939, Lonworks, AS-Interface USB e Bluetooth.

### 3.4.1 Anybus-IC

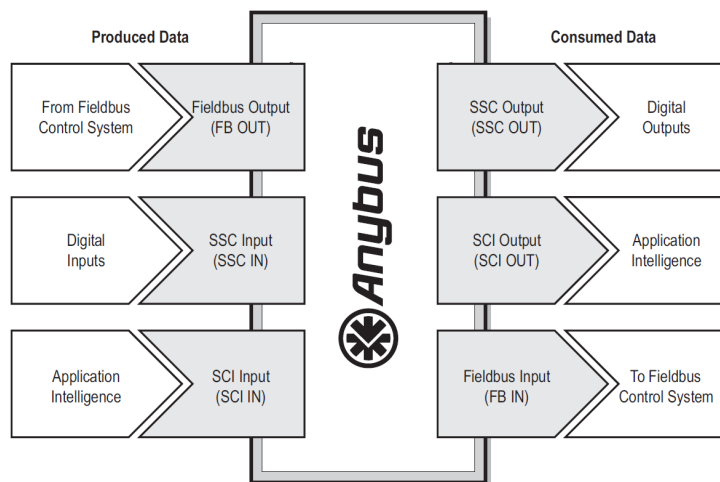
È un chip progettato e costruito da HMS come interfaccia di comunicazione ultra-compatta installabile su alloggiamento per chip DIL-32. AnyBus Integrated Circuit (ABIC), è una famiglia di interfacce complete a singolo chip per reti industriali, ottimizzata per i dispositivi di campo che richiedono una connettività di rete multipla in un ingombro ridotto; è basato sul microcontroller di comunicazione Anybus NP30 RISC di HMS, include tutti i componenti elettronici e il software necessario per implementare un'interfaccia di comunicazione industriale completa. Tutti i componenti sono integrati in un unico socket che occupa circa  $9 \text{ cm}^2$ . Questo chip richiede l'alimentazione di  $5 \text{ V}$  e fornisce un'interfaccia di rete completa con isolamento galvanico. Per ogni rete è disponibile una versione ABIC separata. La standardizzazione delle interfacce meccaniche, elettriche e software offre la possibilità di interscambio tra un chip ABIC e l'altro (vedi a figura 3.5) ed è proprio l'intercambiabilità di questi chip il motivo di interesse di Weightpack per questa famiglia di dispositivi. ABIC integra



Figura 3.5: ABIC per vari tipi di rete.

l'hardware necessario per collegare il dispositivo sul quale il chip andrà installato alla

rete selezionata, Profinet IO nel nostro caso. Se tale dispositivo è dotato di microcontrollore integrato l'ABIC viene interfacciato tramite la porta seriale asincrona a TTL, altrimenti, in apparati privi di intelligenza, esso controlla in autonomia una serie di shift register (per un massimo di 128 bits di traffico scambiato). In entrambi i casi l'ABIC gestisce l'intero protocollo Profinet IO. Il modulo è dotato di un sistema di mappatura flessibile dei dati: i dati ricevuti da un canale di comunicazione possono essere mappati in un altro canale, e viceversa. Per meglio spiegare il fenomeno



**Figura 3.6:** Gestione dei canali nell'ABIC.

si veda la figura 3.6. I dati prodotti da un canale possono venir mappati nel buffer d'uscita di un altro canale con inoltre da un'interfaccia all'altra. In questo modo il modulo supporta la connettività Fieldbus, oltre a fornire I/O interni all'applicazione. Mappando gli I/O Fieldbus all'interfaccia SSC, gli ingressi e le uscite digitali di tale interfaccia diventano immediatamente disponibili dal fieldbus. Il modo con cui viene effettuata la mappatura viene specificato dai parametri in fase di configurazione. Oltre all'interfaccia dati seriale, il modulo è dotato di una seconda interfaccia, Monitor Interface (MIF) a due fili utilizzata per la configurazione e il monitoraggio attraverso un programma di emulazione, tipo Hyperterminal, installato nel terminale PC. L'utilizzo di questa interfaccia è opzionale ma fornisce l'accesso ai parametri del modulo, nonché a funzioni per la configurazione ed il monitoraggio dei dati. In laboratorio si avevano a disposizione due ABIC, entrambi per Profinet IO Classe 1. Questa famiglia di chip infatti nasce per integrare dispositivi di processo all'interno delle reti aziendali di automazione industriale e non è stato progettato per applicazioni di motion control: vien da sè che non supporta i protocolli di comunicazione IRT di Profinet IO.

### 3.4.2 Interfaccia di comunicazione seriale asincrona - SCI

A diporre di un microcontrollore dedicato sono i cosiddetti dispositivi “intelligenti”; questo viene collegato all’ABIC attraverso una porta seriale asincrona di comunicazione, a tecnologia TTL. Il canale Serial Communications Interface (SCI) può essere utilizzato sia per lo scambio dati sia per la configurazione dell’ABIC. Il protocollo di comunicazione si basa sullo standard Modbus-RTU e vengono utilizzati solitamente tre segnali:

- SCI\_Tx Asynchronous serial output; trasporta i dati dall’ABIC all’applicazione;
- SCI\_Rx Asynchronous serial input; trasporta i dati dall’applicazione all’ABIC;
- SCI\_DE questo segnale abilita l’uscita di dati su reti in half-duplex. Connettendo questo segnale a terra si disabilita l’invio della configurazione via SCI e il chip viene inizializzato in automatico come dispositivo stand alone.

Attraverso questa interfaccia è possibile scambiare fino ad un massimo di 128 bytes in ingresso ed altrettanti in uscita con un baudrate compreso tra 4.8 kbps e 57.6 kbps.

### 3.4.3 Interfaccia di comunicazione seriale sincrona - SSC

Per i dispositivi sprovvisti di microcontrollore l’ABIC fornisce un’interfaccia di comunicazione a registri con variazione del clock, tale Synchronous Serial Channel (SSC), per l’accesso diretto ai dati ciclici di I/O senza l’ausilio di un controllore esterno aggiuntivo. Il canale SSC è destinato all’uso con un anello di shift register, esterni

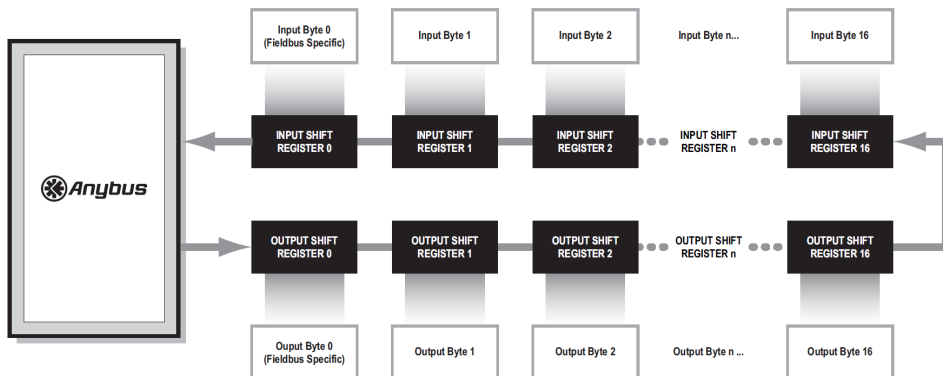


Figura 3.7: Anello di comunicazione del canale SSC nell’ABIC.

all’ABIC, per trasmettere input ed output discreti; supporta al massimo 16 bytes in input e altrettanti in output di traffico dati totale. Come illustrato in figura 3.7 i primi registri input e output sono riservati di default a specifiche per il funzionamento del fieldbus; questa opzione è deselezionabile dai parametri di configurazione



dell'ABIC (manualmente dall'interfaccia MIF su Hyperterminal come descritto in 3.4.1 oppure da parametrizzazione fornita attraverso canale SCI come descritto in 3.4.2).

L'aggiornamento sul canale SSC dei dati in ingresso ed in uscita avviene con un tempo di refresh di 4 *ms*. Tale tipo di funzionamento è stato oggetto di approfondimento durante questo lavoro di tesi poichè si era rilevata una discordanza notevole tra valori forniti nel datasheet [25] ed i risultati delle misure eseguite in laboratorio: test sul clock del dispositivo e l'analisi dei pacchetti dati scambiati hanno confermato la validità delle misure da noi effettuati e l'imprecisione è stata segnalata al costruttore.

#### 3.4.4 La scheda "Evaluation Board"

Per poter utilizzare e testare il chip ABIC si è avuta in dotazione la scheda di valutazione ABIC-EVB, sempre di HMS. Tale scheda consente di inizializzare il modulo ABIC e di monitorarne lo scambio dati, ha al suo interno degli shift register per poter utilizzare il canale SSC, una porta MIF per la configurazione via terminale, una RS323 e un uscita TTL per il canale SCI.

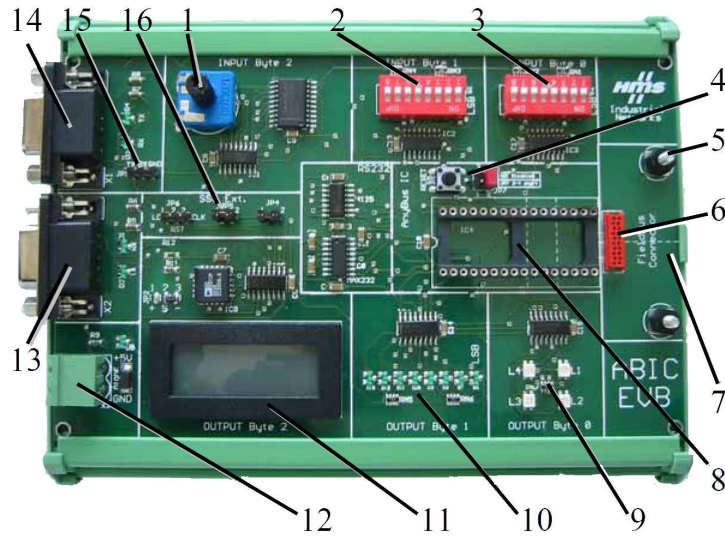
Queste interfacce funzionano in maniera indipendente una dall'altra e possono essere utilizzate in contemporanea. Per il canale sincrono l'EVB mette a disposizione 3 byte di ingresso (due dip switch da 8 bit ed un potenziometro con ADC) e 3 byte di uscita (due a led ed uno con schermo 0-5 *V* analogico preceduto da un DAC). Per quanto riguarda il canale asincrono invece l'EVB dispone di una interfaccia a TTL per collegamento al micro-controllore esterno ed una RS-232 per il collegamento seriale al PC. Per trasmettere dati nella linea seriale sono necessari dei drivers per la conversione dei livelli logici TTL in livelli di tensione RS-232; mentre per convertire i livelli di tensione RS-232 relativi ai dati ricevuti occorre impiegare dei ricevitori di linea in grado di fornire livelli logici TTL al micro-controllore: entrambi i dispositivi sono integrati nella EVB grazie al circuito MAXIM 232 a 16 pin. In figura 3.8 si riporta una foto dettagliata con spiegate in tabella 3.1 le varie componenti fisiche della scheda.

### 3.5 La stazione PC

La stazione PC è stata utilizzata in diversi modi nel corso del lavoro. In un primo tempo è servita per la configurazione del PLC e della rete Profinet, poi per la configurazione dell'ABIC; ha quindi fatto da SCADA al sistema gestito dal PLC ed infine ha fatto da soft-PLC. Tutte queste funzioni sono state possibili grazie a diversi software e tools di sviluppo forniti interamente da Weightpack: nel seguito daremo una descrizione degli stessi.

#### 3.5.1 Il tool progettuale Step7

Siemens offre un tool di sviluppo per i suoi prodotti chiamato STEP 7 che permette di gestire e configurare la maggior parte dei sistemi di automazione facenti capo alla



**Figura 3.8:** *Illustrazione con dettagli dell'Evaluation Board fornita da HMS in supporto all'ABIC.*

#	SPIEGAZIONE DELLA FIGURA 3.8
1	Potenzimetro per valori analogici (Input byte # 2)
2	Dip switch per valori di input digitali (Input byte # 1)
3	Dip switch per valori di input digitali (Input byte # 0)
4	Jumper per l'abilitazione del canale SCI/switch di reset
5	Viti per il fissaggio di connettori di campo
6	Interfaccia di campo
7	Socket per il connettore di campo
8	Alloggiamento DIL-32 per l'ABIC
9	Led per l'uscita digitale/Led di diagnostica (Output byte #0)
10	Led per l'uscita digitale (Output byte #1)
11	Display digitale per l'uscita analogica (Output byte #2)
12	Connettore di alimentazione (5 V dc, 800 mA)
13	Connettore di interfaccia MIF (RS-232)
14	Connettore di interfaccia SCI (RS-232)
15	Connettore di interfaccia SCI (5 V TTL)
16	Connettori per l'espansione degli shift register dell'interfaccia SSC

**Tabella 3.1:** *Spiegazione delle componenti della scheda ABIC-EVB.*

dicatura SIMATIC. Il software viene fornito per l'installazione su sistemi operativi Windows. Per ogni progetto esso prevede determinati passaggi (step):

- CREAZIONE DELLA STRUTTURA DEL PROGETTO E PREPARAZIONE DELLA STAZIONE

Un progetto è paragonabile ad una cartella, nella quale tutti i dati sono organizzati gerarchicamente. Preparando la stazione si definisce il controllore, che nel nostro caso sarà una stazione SIMATIC 300.

- CONFIGURAZIONE DELL'HARDWARE

Nella fase di configurazione si definisce in una tabella di configurazione quali unità vengono utilizzate per la soluzione di automazione, e mediante quali indirizzi le unità devono essere raggiungibili dal programma. È inoltre possibile impostare le proprietà delle unità per mezzo di parametri.

- PROGETTAZIONE DI RETI E COLLEGAMENTI DI COMUNICAZIONE

La base per la comunicazione è costituita da una rete precedentemente configurata. A tal fine, occorre creare le sotto-reti necessarie per le reti di automazione, definire le proprietà delle sotto-reti, le proprietà del collegamento di rete per le stazioni collegate, ed eventualmente i collegamenti occorrenti per la comunicazione.

- CREAZIONE DEL PROGRAMMA

Con uno dei linguaggi di programmazione disponibili, l'utente crea un programma assegnato a unità o indipendente dalle unità, e lo salva sotto forma di blocchi, sorgenti o schemi. Nel nostro caso i programmi di prova sono stati scritti in linguaggio AWL.

- CARICAMENTO DEI PROGRAMMI NEL SISTEMA DI DESTINAZIONE

Dopo aver terminato la configurazione, la parametrizzazione e la creazione del programma, è possibile caricare il programma utente completo o i suoi singoli blocchi nel sistema di destinazione (unità programmabile della soluzione hardware decisa dall'utente). La CPU contiene già il sistema operativo.

Per la configurazione di una rete Profinet è necessario inoltre parametrizzare l'interfaccia di accesso alla rete stessa; in pratica viene abilitata la scheda di rete del PC in cui è caricato lo STEP 7. Bisogna tenere conto dell'indirizzo IP e dell'indirizzo di sotto rete dell'interfaccia per evitare malfunzionamenti in fase di download delle impostazioni. Anche ad ogni dispositivo di rete deve essere assegnato un IP e un Subnet Mask oltre che un nome dispositivo univoco.

### Configurazione hardware

In "Configurazione HW" si possono settare i parametri per ogni unità inserita nel progetto. Per la CPU315-2 PN/DP è stato attivato il web server e associato un indirizzo IP (192.168.0.1) per poter accedere ai dati di diagnostica in modo rapido ed è stato impostato, talvolta, il campo relativo alla schedulazione dell'orologio (blocco OB35). Quest'ultima permette di interrompere la sequenza ciclica di programma, che viene elaborata nel blocco organizzativo OB1, per eseguire le istruzioni caricate nel blocco organizzativo OB35.

Il tempo di schedulazione è impostabile da un minimo di 1 *ms* a un massimo di 60

s. Per le misurazioni questo valore è stato impostato a 4 *ms*, 16 *ms*, 64 *ms* e 256 *ms*, a seconda della misura da fare. Si è usata la schedulazione dell'orologio del PLC in quanto era necessario cercare di seguire il ciclo di aggiornamento della comunicazione degli IO device per avere delle misure che fossero attendibili. Chiaramente, come ci si poteva aspettare, non si è potuto creare un perfetto sincronismo tra le due tempistiche; tuttavia l'accorgimento utilizzato è stato di grande aiuto ai fini della comprensione delle misure effettuate.

Il set di dati impostati per configurare l'IM153-4 (IP 192.168.0.5) prevedono la scelta della modalità di sincronizzazione e il tempo di aggiornamento del ciclo IO. Nella scheda "Sincronizzazione" è possibile impostare il tipo di comunicazione; come già detto l'IM153-4 prevede le classi di comunicazione RT e IRT. Visto che la CPU supporta solamente la RT classe 1 non si ha possibilità di scelta. Il ciclo di IO si riferisce alla ciclicità temporale con cui avviene lo scambio di dati Real-Time tra l'IO-device (ET200M) e l'IO-Controller (CPU315-2 PN/DP). Il valore impostabile parte da un minimo di 1ms ad un massimo di 512 *ms*.

Il chip ABIC (IP 192.168.0.4) supporta solamente la RT classe 1 ed anche in questo caso i tempi di aggiornamento utilizzati sono uguali ai tempi impostati nella schedulazione dell'orologio. Sebbene il dispositivo avesse un web server integrato si è preferito utilizzare l'interfaccia MIF per la sua configurazione.

Contrariamente nello Scalance (IP 192.168.0.2) è stato utilizzato parecchio la funzione di web server, in quanto dotato di una chiara e semplice diagnostica e di una buona scelta per il settaggio del dispositivo. Dal web management dello switch si è attivata la funzionalità di Mirroring quando è stato necessario monitorare il traffico scambiato su di una determinata porta dello switch; per fare ciò si è utilizzato il programma freeware "WireShark Network Protocol Analyzer": è un analizzatore di rete software che sfrutta l'accesso a risorse di basso livello del sistema operativo andando a catturare i pacchetti memorizzati nel buffer di ingresso della scheda di rete. Questo si è rilevato molto utile per l'analisi della tipologia dei pacchetti ma non di certo per una misurazione temporale, a causa del non determinismo del sistema operativo.

Parametro fondamentale in fase di configurazione è il tempo di ciclo di Profinet IO. Step7 non dà la possibilità all'utente di settarlo manualmente, bensì viene calcolato ed impostato in base ai dispositivi collegati e alle prestazioni selezionate su di essi. Avendo la nostra configurazione pochi dispositivi collegati tale tempo è risultato di 1ms (il minimo tempo supportato dalla CPU).

### 3.5.2 Lo SCADA "Zenon"

Il software utilizzato come sistema SCADA è ZENON 6.50 di Copadata. Il motivo per cui Weightpack si è interessata a questo software sta nel fatto che il sistema offerto da Copadata non funge solo da sistema di supervisione remota. Esso contiene al suo interno il programma Straton che funge da soft-PLC, questo potrebbe essere utile nel caso si volesse sgravare di qualche onere il PLC principale che gestisce la macchina. Altro fattore d'interesse per questo programma è la sua versatilità: la configurazione per il runtime di Zenon viene realizzata tramite l'editor. Tutti i tipi di progetto (stand-alone, in rete o ridondanti) sono creati per tutte le piattaforme

(palmari CE, computer e pagine web) con lo stesso editor e beneficiano di una integrazione completa e di una indipendenza da un tipo specifico di risoluzione; ciò significa che ogni progetto può essere utilizzato senza alcun ulteriore adattamento in ogni tipologia di rete e con ogni sistema operativo; anche la grandezza dell'immagine viene adattata automaticamente alla risoluzione del monitor del sistema destinatario. È un utile strumento che permette di strutturare la produzione con la modellazione ad impianti secondo lo standard ISA-S95 ed analizzare dati in tempo reale; esigenze differenti vengono gestite con profili individuali che si orientano all'utente loggato e permettono diritti di accesso differenziati.

Altro punto di forza di Zenon sta nella precisione che c'è alla base del suo sviluppo, specie per quanto riguarda la registrazione e l'analisi dei dati di processo supervisionati e/o elaborati dal sistema. Si può disporre in ogni momento di dati archiviati ed elaborati ed avere una solida base per poter prendere delle decisioni ponderate consultando. Zenon garantisce un salvataggio e un'archiviazione precisi e puntuali di tutti i dati che possono essere rilevanti e lancia autonomamente un allarme ogni volta in cui sia necessario l'intervento di un operatore addestrato.

Tutte queste caratteristiche permettono ad un'azienda di aumentare la produttività e di ridurre sia il Time to Market sia i costi. Una descrizione più approfondita del software e delle sue caratteristiche la si può trovare nel sito web di Copadata [28], in [29] e [30].

### Il soft-PLC “Straton”

La soluzione integrata di Zenon e Straton permette lo scambio di variabili strutturate. I blocchi di funzioni di straton possono essere supportati da variabili complesse. In tal modo viene eliminato il problema del cosiddetto “broad-band cabling”. Questa soluzione offre una visione d'insieme più chiara e, di conseguenza, una maggiore velocità nella fase di progettazione. Straton RTK (Real Time Kernel) offre un effettivo funzionamento in tempo reale per le applicazioni PLC su un PC industriale equipaggiato con Windows XP o Windows XP embedded. Straton RTK si distingue per la possibilità di ridurre i tempi di latenza del sistema e di gestire i “bluescreen”, cioè quegli errori di sistema che il sistema operativo non è in grado di risolvere. La selezione del fieldbus si effettua mediante un configuratore semplice da utilizzare, i cui driver sono già inclusi nel pacchetto software. Straton offre due possibili implementazioni del Real Time: la prima definita HRT, la seconda SRT (vedi sezione 1.4.1).

## 3.6 L'oscilloscopio “HP”

Per effettuare le misurazioni ci si è dotati di un oscilloscopio Hewlett Packard 54504A; è un oscilloscopio digitale a due canali con una banda passante di 400 *Mhz* e con profondità di memoria di 200 *MSa/s*; è provvisto di due sonde passive 10:1 da 1 *MΩ*. È dotato di funzione autoscale e di varie funzioni matematiche che riescono a calcolare direttamente i semiperiodi del segnale, il rise-time, il fall-time, il ritardo

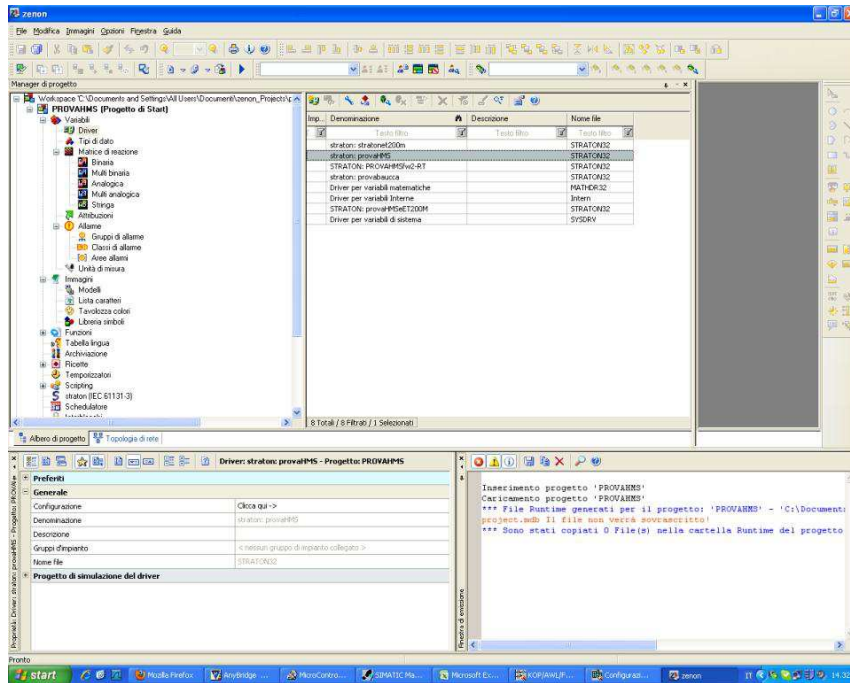


Figura 3.9: Istantanea dell'ambiente di sviluppo di Zenon.

tra due segnali, etc. L'oscilloscopio è stato utilizzato per le varie misure temporali e di tensione ma anche per comprendere l'andamento di alcuni segnali e carpirne le relative forme d'onda.

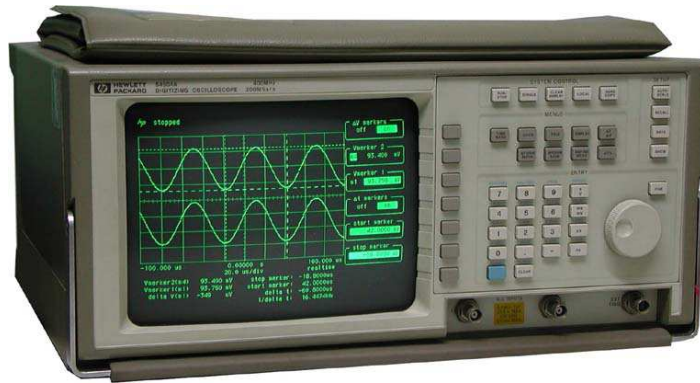
A causa della limitata disponibilità di memoria dello strumento la precisione delle misure temporali presenta un errore relativo sulla misura che varia in base alla scala di visualizzazione impostata. I valori calcolati sono accettabili e vengono riportati nella tabella 3.2. Le misurazioni, di cui si parlerà nel capitolo 4, sono state effettuate

$ms/DIV$	$Sa/s$	Errore assoluto
200	250	4 ms
100	500	2 ms
50	1k	1 ms
20	2.5k	0.4 ms
10	5k	0.2 ms
5	10k	0.1 ms
2	25k	0.04 ms
1	50k	0.02 ms

Tabella 3.2: Si riportano gli errori assoluti ricavati in base alle impostazioni dell'oscilloscopio nell'acquisizione delle misure.

tenendo conto della precisione dell'oscilloscopio. Nel caso di misura con tempo di

aggiornamento di  $256\text{ ms}$  si ha un errore sulla singola misura di  $1\text{ ms}$ ; con tempo di aggiornamento di  $64\text{ ms}$  si ha un errore di  $0.4\text{ ms}$  mentre nel caso di  $16\text{ ms}$  si ha un errore di  $0.1\text{ ms}$ . Dove siano presenti misure eseguite con tempi di aggiornamento di  $4\text{ ms}$ , l'errore scende ancora fino ad un minimo di  $20\text{ }\mu\text{s}$ .



**Figura 3.10:** *Illustrazione dell'oscilloscopio HP54504A in dotazione.*





# Capitolo 4

## Misure e risultati sperimentali

Il fulcro su cui si è centrato il lavoro di tesi e su cui si è strutturato tutto il lavoro è il chip ABIC, già descritto nella sezione 3.4. L'interesse di Weightpack nei confronti di tale chip è dovuto alla sua facilità d'integrazione in sistemi d'automazione con field-bus differenti, i cui vantaggi, come la protezione degli investimenti e la dinamicità della macchina, sono tutt'altro che scontati. Si è deciso di effettuare qualche test di misura per comprendere se l'ABIC fosse o meno preciso nel traffico dati, per vedere se le caratteristiche fornite dal costruttore fossero verificate e, più in generale, per stabilire se fosse un dispositivo adatto alle esigenze operative di Weightpack.

Il chip è stato configurato per comunicare attraverso il protocollo Profinet IO utilizzando l'interfaccia Ethernet; il bus di campo utilizzato è l'SSC (vedi descrizione in sezione 3.4.3) con configurati 3 byte di ingresso e 2 in uscita. Si è quindi inserito il chip di HMS in una rete Profinet IO con gli altri prodotti già elencati nel capitolo 3. Prima di descrivere la fase operativa delle misurazioni effettuate si fa una breve panoramica sul lavoro inizialmente eseguito (paragrafo 4.1) e sulle considerazioni preliminari inerenti il chip sotto test (paragrafo 4.2). Si è quindi diviso il lavoro svolto in più fasi.

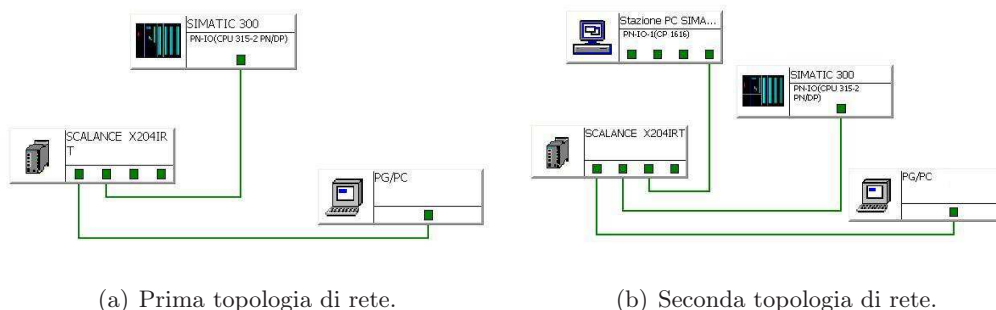
### 4.1 Prima fase: allestimento

Il primo step del lavoro di tesi prevedeva la configurazione di una rete di tipo Profinet IO, utilizzando il tool di sviluppo di Siemens Step7, ed avendo a disposizione i dispositivi:

- Alimentatore Telemecanique  $230V_{ac}/24V_{dc} - I_{out} 5A$
- CPU 315-2 PN/DP
- SCALANCE X204 IRT
- Stazione PC, con Windows XP PRO sp3, con installata la scheda CP-1616 e software Simatic Step 7

Come prima attività si è proceduto all’installazione fisica dei dispositivi sopra elencati. Su una barra profilata di tipo Simatic S7 di Siemens si sono disposti l’alimentatore, la CPU e lo switch. A protezione di CPU e switch sono stati inseriti dei fusibili di protezione e l’intera stazione è stata messa a terra come indicato nei manuali d’installazione [20]. La scheda CP1616 è stata installata nello slot PCI del computer. Tutte le versioni forniteci dai vari supporti tecnici Siemens contattati hanno espresso la loro perplessità nell’utilizzo di tale scheda al fine di una supervisione, in quanto questo non è il compito per cui la CP1616 è stata sviluppata. La CP1616 infatti non funziona come una normale scheda ethernet e per interfacciarsi con il pc in cui è installata ha bisogno di un programma ad hoc che lo sviluppatore/programmatore deve creare per poter comunicare con la scheda. La CP1616 può essere usata come interfaccia verso reti Profinet IO in associazione con il pacchetto software WinAC. Per la sua configurazione si dovrebbe usare il software Simatic NET. Non potendo dunque utilizzare lo Step 7 dalla postazione fissa con la CP1616 abbiamo provato a configurare la rete per mezzo di un computer portatile collegato direttamente allo switch tramite scheda di rete ethernet. Da questa postazione si riesce a “vedere” la CP1616 come un nodo di rete; si riesce infatti ad impostarle un indirizzo IP ed un nome, ad accedere al suo Web server e a caricarne il relativo firmware. Per prendere familiarità con il SIMATIC Manager si è inizialmente progettato una rete solo con CPU, Switch e computer portatile riuscendo quindi ad instaurare un collegamento PROFINET IO solo tra i due dispositivi (CPU e Switch) che supportano tale funzionalità ma senza potessero scambiare dati (figura 4.1(a)). Successivamente, sempre dal computer portatile, si è provato ad aggiungere al progetto la scheda CP1616:

- si è aggiunta una “stazione PC SIMATIC” nel progetto;
- si è inserita la CP1616 nella stazione;
- non si è potuto inserire il software di controllo di tale scheda (in quanto non previsto nel pacchetto Simatic Step 7);
- si è creata la nuova topologia di rete (figura 4.1(b));.



**Figura 4.1:** Le due topologie di rete inizialmente utilizzate.

Da quanto rilevato con Topologia Profinet di Step7 si è riusciti a configurare la rete in maniera corretta ed a caricarne la configurazione sia nella CP, sia nella CPU, senza alcun “System Fault” delle stesse. Tale assenza di errori non è condizione sufficiente a permettere di dire di aver instaurato un dialogo Profinet tra i vari componenti della rete: nella configurazione hardware non era stato impostato nessun tipo di comunicazione tra PLC e CP poiché non c’era modo di parametrizzare correttamente quest’ultima.

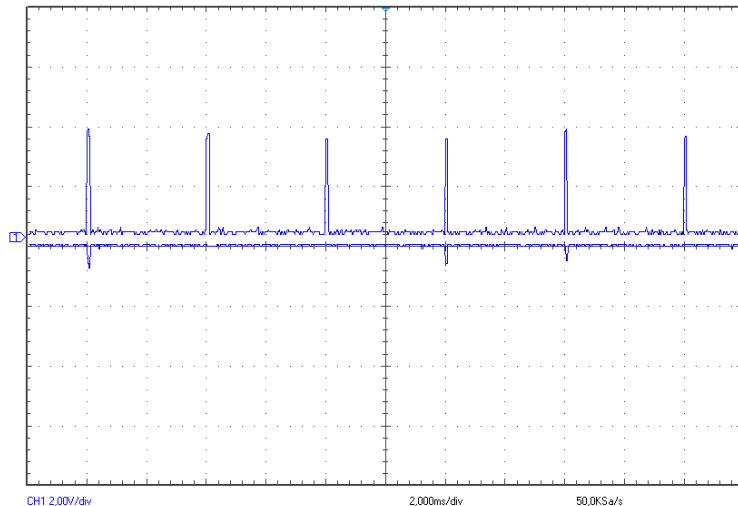
Il problema principale riscontrato in questa fase è che la CP1616 non può essere utilizzata dalla stazione PC in cui è installata poiché non è disponibile alcun tipo di interfaccia né per comunicare con essa né per accedere alla memoria utilizzata. L’unico tipo di accesso correttamente funzionante che si è instaurato con la scheda è stato tramite LAN ethernet e quindi tramite un dispositivo PG/PC esterno collegato via rete alla scheda, con l’ausilio del programma NCM S7 del pacchetto di Step7. Come ulteriore conferma di quanto detto si è tenuto in considerazione quanto detto dai tecnici dell’assistenza Siemens, i quali hanno precisato sia le potenzialità di questa scheda sia i problemi a cui si va incontro non avendo il software di sviluppo adeguato. A tal riguardo si è deciso di procedere utilizzando il pacchetto Zenon senza CP1616, adottando come interfaccia una comune scheda di rete. Tale scelta progettuale non va a scapito della velocità di comunicazione in quanto ProfinetIO RT classe 1 è supportato appieno da una normale scheda di rete e non necessita di hardware dedicato.

## 4.2 Seconda fase: considerazioni preliminari alle misure

Si descrive ora quanto pervenuto in contemporanea alle prime misure di prestazione eseguite sull'AnyBus IC Profinet, il cui datasheet si può trovare in [24]. In [25] capitolo 4 si descrivono le funzionalità e le caratteristiche tecniche del canale SSC dell'ABIC, canale utilizzato nelle prove eseguite in laboratorio. Secondo i suddetti datasheet la comunicazione del canale SSC dispone di una sequenza di inizializzazione che, solo in fase di start-up del chip, esegue una scansione per calcolare la quantità di shift register presenti nel canale e successivamente esegue una scansione di tali registri ogni  $6\text{ ms}$ . Quanto descritto è stato dedotto utilizzando i seguenti componenti in fase di test:

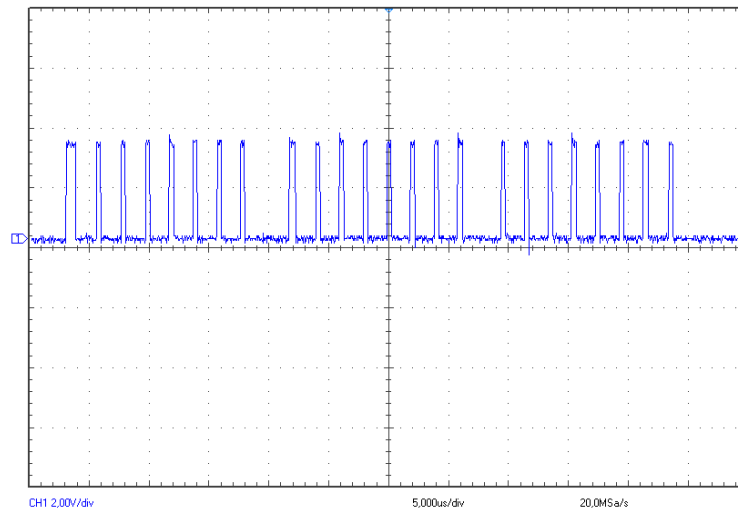
- chip ABIC-PRT 00-30-11-03-60-7E
- chip ABIC-PRT 00-30-11-03-60-80
- scheda ABIC EVB V 2.4

In seguito ad una serie di misurazioni sulle tempistiche di scambio dati tra il fieldbus (Profinet IO) ed il canale SSC (in particolare il led 0 del byte 1) si è verificata la necessità di controllare la veridicità di tali specifiche fornite. Le misurazioni effettuate, con oscilloscopio HP 54504 A, restituivano misure ben più prestanti di quelle che ci si poteva aspettare con un periodo di refresh della memoria di  $6\text{ ms}$ . Collegando la sonda al pin 7 del clock "SSC\_CLK" si sono acquisite le forme d'onda riportate nelle figure 4.2 e 4.3. In 4.2 è rappresentata la distanza temporale tra più scansioni



**Figura 4.2:** Refresh time della memoria.

successive, definito come tempo di refresh della memoria dell'ABIC, descritto nella pagina 4-8 di [25]. Da questa forma d'onda acquisita dall'oscilloscopio è di notevole interesse il fatto che il refresh time della memoria (CLK\_SSC) viene eseguito ogni 4



**Figura 4.3:** *Treno d'impulsi presente in ogni ciclo di refresh.*

*ms* e non ogni *6 ms* come specificato dalle specifiche tecniche.

Per avere un ulteriore approfondimento si è misurato il treno d'impulsi che si ripete ad ogni ciclo. Come si può vedere in figura 4.3 il treno di 8 impulsi è ripetuto 3 volte, poichè l'ABIC è configurato con shift registers, 3 di input e 2 di output.

### 4.3 Terza fase: primo set di misure

Le misure che abbiamo eseguito necessitano di una descrizione separata sia per la diversa tipologia sia per le varie topologie di collegamenti adottati in ogni misura. In ogni sezione sarà quindi riportato sia lo schema di collegamento, sia i grafici con i risultati ottenuti. Nelle seguenti descrizioni con il termine *semiperiodo* indicheremo il tempo necessario perchè si compia un intero ciclo del plc; si parlerà di *semiperiodo positivo* quando l'uscita monitorata avrà livello logico alto; si parlerà invece di *semiperiodo negativo* quando l'uscita monitorata avrà livello logico basso. Questo abuso di notazione è stato adottato per semplificare la successiva spiegazione delle misure.

#### 4.3.1 Misura 1

La prima misura intrapresa, la cui topologia è riportata nello schema di figura 4.4, vuole quantificare il lasso temporale tra l'invio di un segnale all'uscita del PLC e l'effettiva commutazione dell'uscita dell'ABIC. L'uscita in questione è rappresentata dal bit 0 del byte 1 dell'ABIC-EVB (si veda la figura dettagliata 3.8). Per poter effettuare questa misura si è collegato il blocco di ingressi ed uscite digitali SM374 direttamente alla CPU, mentre l'ABIC lo si è collegato tramite rete Profinet IO. Si è quindi scritto un programma di prova e caricato nell'OB35 del plc. Il senso di questa misura è determinare il ritardo introdotto dalla comunicazione Profinet IO. Per eseguire questa misura si sono impostati sia il tempo di schedulazione sia il tempo di aggiornamento dell'ABIC a 16, a 64 e a 256 *ms*. Per poter dare una stima affidabile dei tempi misurati riportiamo le considerazioni fatte dopo una esaustiva documentazione sul funzionamento del plc e sullo scambio dei dati tramite protocollo Profinet IO. Come prima cosa si è schematizzato il sistema di misura raggiunto dalle due sonde e se n'è data una caratterizzazione a livello temporale. La sonda del canale uno è stata collegata ad un led del blocco I/O, tale A0.0, collegato alla cpu. La sonda due è stata collegata ad un led dell'ABIC, tale A1.0. Il programma caricato in OB35 fornisce in contemporanea il medesimo valore alle uscite, poiché vengono settate nell'Immagine di Processo delle Uscite del plc.

Riassumiamo con le seguenti formule i calcoli delle tempistiche relative ai due canali dell'oscilloscopio:

$$T_{S1} = T_{CPU} + T_{EL-374}$$

$$T_{S2} = T_{CPU} + T_{CPU-PNIO} + T_{TX} + T_{PNIO-ABIC} + T_{EL-ABIC} + T_{OFFSET-PNIO}$$

Dove:

- $T_{S1}$  è il tempo di set dell'uscita locale del PLC;
- $T_{S2}$  è il tempo di set dell'uscita remota nell'ABIC;
- $T_{CPU}$  è il tempo impiegato dalla cpu per eseguire il ciclo macchina, sarà circa uguale al valore di schedulazione dell'orologio impostato per il blocco OB35;

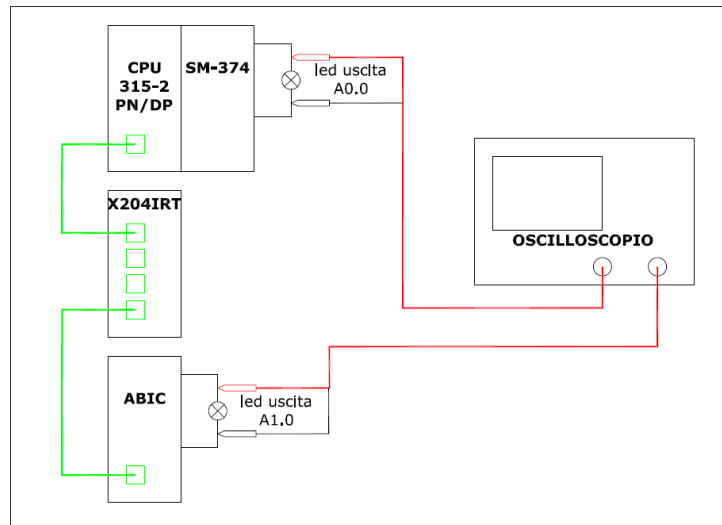


Figura 4.4: Topologia dei collegamenti per la prima misura.

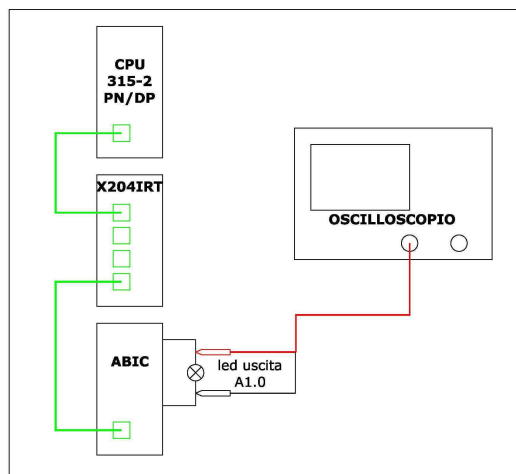
- $T_{EL-374}$  è il tempo di reazione impiegato dalla cpu a trasmettere i dati al blocco I/O 374 fino alla commutazione del led d'uscita A0.0;
- $T_{CPU-PNIO}$  è il tempo di reazione impiegato dalla cpu a scrivere i dati nel buffer di comunicazione del sistema Profinet;
- $T_{TX}$  è il tempo impiegato dalla rete Profinet a trasferire il dato dal buffer della cpu a quello dell'ABIC, compreso il passaggio nello switch;
- $T_{PNIO-ABIC}$  è il tempo di reazione impiegato dal dato a passare dal buffer Profinet dell'ABIC al chip stesso;
- $T_{EL-ABIC}$  è il tempo di reazione impiegato dall'ABIC a trasmettere i dati dal chip fino alla commutazione del led d'uscita A1.0; essendo il periodo di aggiornamento del canale sincrono utilizzato al massimo  $4ms$ , questo tempo sarà nel caso peggiore pari a  $4ms$  (dalle considerazioni fatte in 4.2);
- $T_{OFFSET-PNIO}$  è il tempo che dipende dall'istante in cui si instaura la comunicazione tra CPU ed ABIC; è collegato quindi al ciclo di bus Profinet ed è al massimo uguale al tempo di aggiornamento impostato nell'IO Device.

A causa della variabilità del  $T_{OFFSET-PNIO}$  non è possibile rilevare delle misure che caratterizzino correttamente il sistema: si ha infatti un errore di misura dello stesso ordine di grandezza della misura stessa. Quando la cpu viene posizionata in "Stop" la comunicazione Profinet resta attiva; non appena si passa allo stato "Run" si avvia il dialogo su Profinet: questo dialogo si aggancia al primo ciclo bus utile disponibile (che nel nostro caso è di  $1ms$ ) fino ad avere un ritardo massimo pari al tempo di aggiornamento dell'ABIC. Tali misure sono quindi direttamente correlate

con l'istante in cui l'operatore umano avvia la cpu e, poiché la misura non è ripetibile, il valore misurato non può essere considerato analiticamente rilevante.

### 4.3.2 Misura 2

La seconda configurazione di misura, rappresentata nello schema 4.5, ha lo scopo di misurare la regolarità della comunicazione tra il chip ABIC ed il plc. Si è caricato

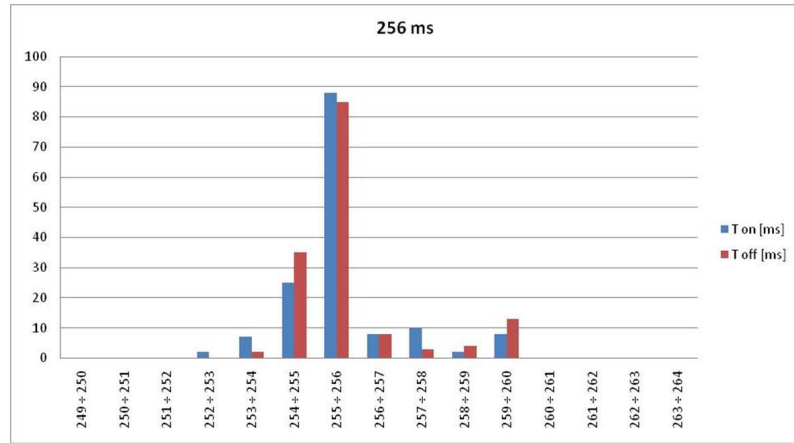


**Figura 4.5:** Topologia dei collegamenti per la seconda misura.

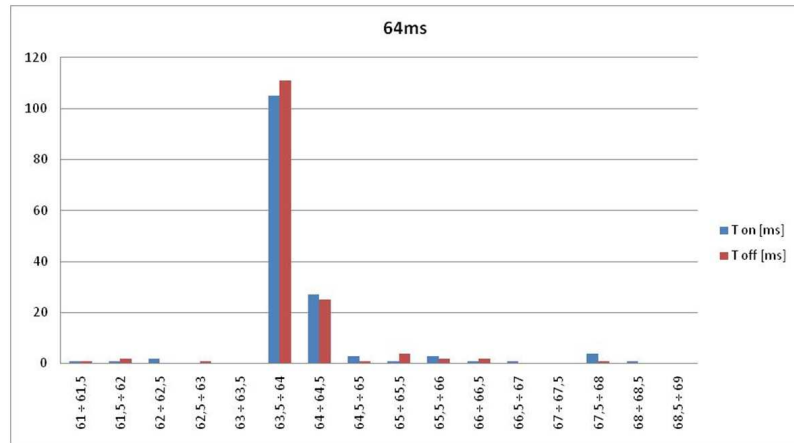
nella CPU un programma che facesse commutare un bit d'uscita ad ogni ciclo di schedulazione del blocco organizzativo OB35. A tale proposito si è settato lo stesso valore al tempo di schedulazione e al tempo di aggiornamento dell'ABIC. Sono state effettuate misure a 16, 64 e 256 ms allo scopo di verificare la regolarità e la costanza delle commutazioni; si sono raccolti 150 campioni per ogni semiperiodo positivo e altrettanti per quello negativo. Nella fase di acquisizione dati si è tenuto conto dell'impossibilità fisica di sincronizzare i periodi di OB35 e i cicli di aggiornamento Profinet dell'ABIC e quindi si sono scartate le misure palesemente errate ai fini del senso della misurazione. Alla luce dei ritardi introdotti dal canale sincrono si è deciso di non effettuare misurazioni a 4 ms in quanto non sarebbe stato possibile differenziare i valori errati introdotti dalla perdita di sincronismo tra schedulazione ed aggiornamento da quelli introdotti dal ritardo di elaborazione del dato del canale sincrono SSC: queste misurazioni non sarebbero quindi state sensibili ai fini della caratterizzazione del chip.

Si riportano nei grafici a figura 4.6 i valori rilevati in fase di misurazione, da questi si evince la buona precisione del chip nell'intorno del valore atteso del tempo impostato da Step7 come aggiornamento dell'ABIC. Come possiamo vedere il ritardo massimo rilevato è stato intorno ai 4 ms.

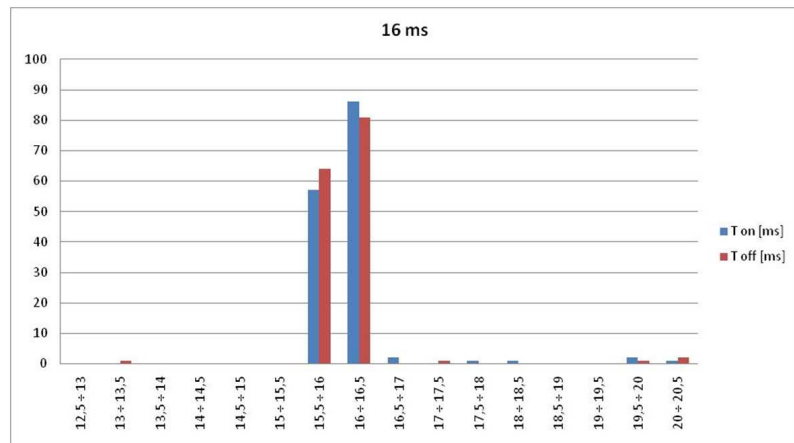




(a) Tempo di scheduazione impostato a 256 ms.



(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

Figura 4.6: Grafici riportanti le misure descritte dalla configurazione 2.

### 4.3.3 Misura 3

La terza configurazione ripropone la stessa tipologia di misura fatta in 4.3.2. In questo caso si è testato il modulo ET200M anziché l'ABIC al fine di poter comparare i due IO device. Anche in questo caso sono state effettuate misure a 16, 64 e 256 ms allo scopo di verificare la regolarità e la costanza delle commutazioni; si sono raccolti 150 campioni per ogni semiperiodo positivo e altrettanti per quello negativo. Nella fase di acquisizione dati si è tenuto conto dell'impossibilità fisica di sincronizzare i periodi di OB35 e i cicli di aggiornamento Profinet dell'ET200M e quindi si sono scartate le misure palesemente errate ai fini della misurazione.

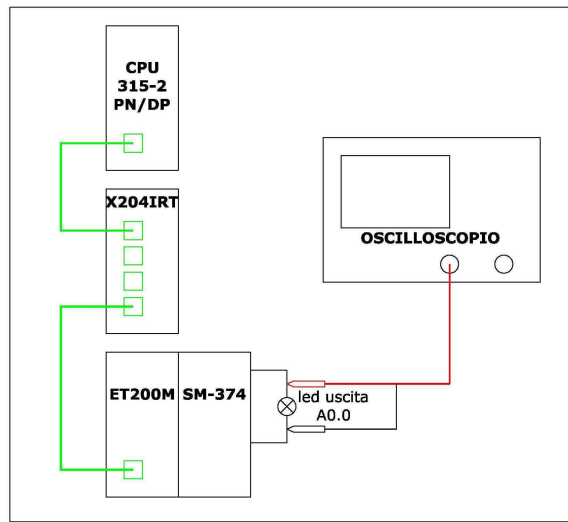


Figura 4.7: Topologia dei collegamenti per la terza misura.

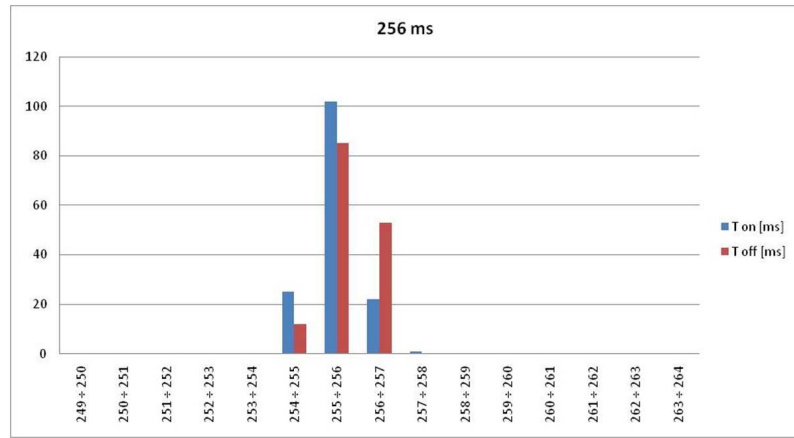
Si riportano i grafici con i valori rilevati in fase di misurazione: si nota come i tempi impiegati dal modulo stiano nell'intorno del valore temporale che ci si poteva aspettare. Questo valore atteso è prossimo al tempo di aggiornamento dell'ET200M impostato da Step7.

Nella tabella sottostante riportiamo le medie e le deviazioni standard delle misurazioni di entrambi i moduli:

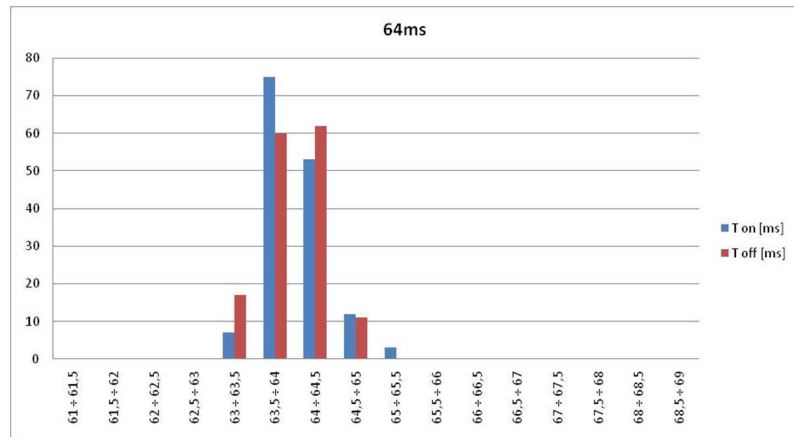
misura 2 - abic			
		$\mu[ms]$	$\sigma[ms]$
Ton	256ms	255.6	1.30
Toff	256ms	255.7	1.42
Ton	64ms	64.0	0.90
Toff	64ms	63.9	0.63
Ton	16ms	16.1	0.59
Toff	16ms	16.0	0.63

misura 3 - ET200M			
		$\mu[ms]$	$\sigma[ms]$
Ton	256ms	255.4	0.60
Toff	256ms	255.7	0.51
Ton	64ms	64.0	0.40
Toff	64ms	63.9	0.41
Ton	16ms	15.9	0.40
Toff	16ms	16.0	0.42

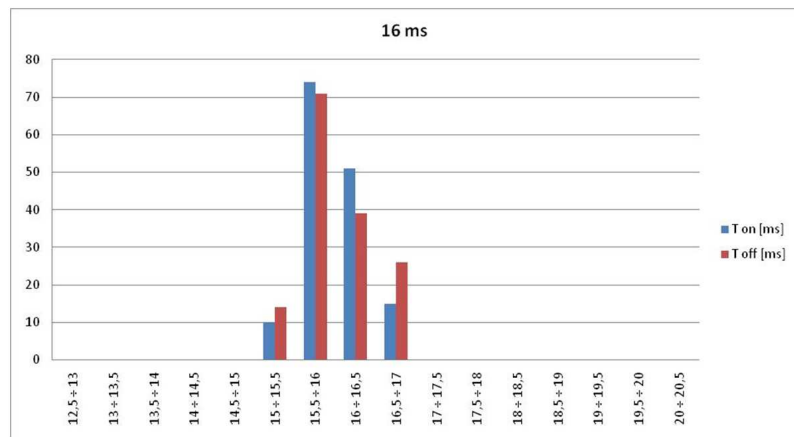
Si nota come le deviazioni standard dell'ET200M variano di poco il loro valore, mentre nell'ABIC si ha una variazione direttamente proporzionale all'aumentare del



(a) Tempo di scheduazione impostato a 256 ms.



(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

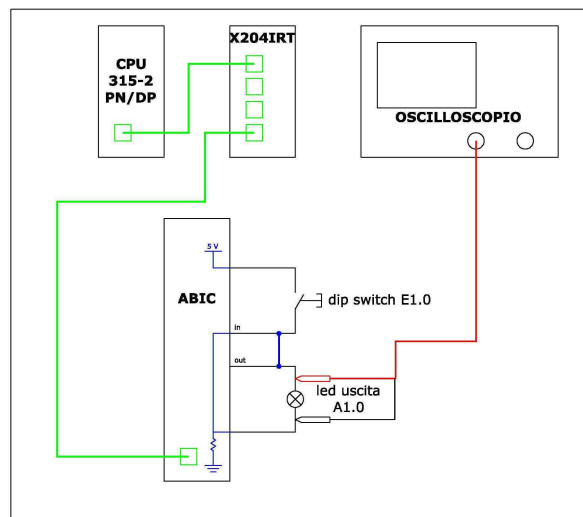
Figura 4.8: Grafici riportanti le misure descritte dalla configurazione 3.

tempo di aggiornamento della schedina.

Dalle tabelle si evince come l'ET200M sia più accurato dell'ABIC: questo risultato era facilmente intuibile in quanto l'ET200M è stato progettato per la comunicazione di classe IRT, quindi molto più prestante che non la classe 1 RT. Si nota inoltre come il valore della deviazione standard misurato sia abbastanza costante nell'ET200M e come tale valore rientri nelle specifiche riportate dal costruttore, che dichiara tempi di commutazioni delle uscite entro i  $500 \mu s$ .

#### 4.3.4 Misura 4

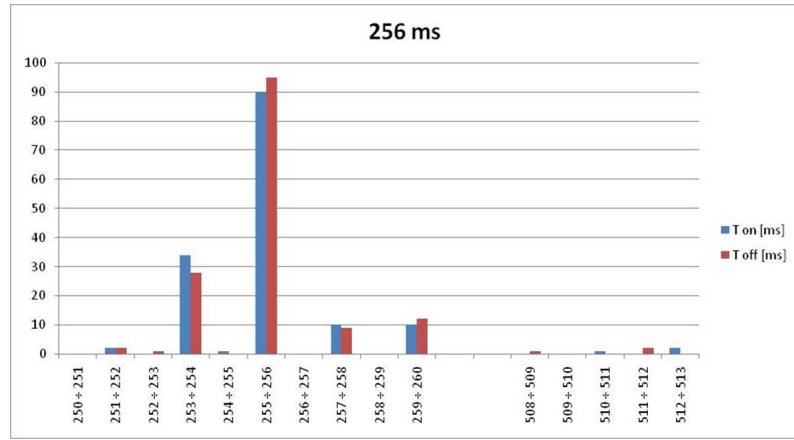
Come quarta misurazione si è voluto testare il tempo complessivo di lettura-scrittura nella scheda ABIC. La topologia è rappresentata nello schema 4.9: nell'EVB è stato effettuato un ponte fisico tra il primo bit del byte 1 d'ingresso ed il primo bit del byte 1 d'uscita. In aggiunta si è caricato nella CPU un programma che settasse ad ogni ciclo l'uscita A1.0 come negazione dell'ingresso letto E1.0. Il programma è stato caricato nell'OB1 e non nell'OB35 in quanto il tempo di schedulazione avrebbe introdotto un ulteriore errore causato dalla mancanza di sincronismo.



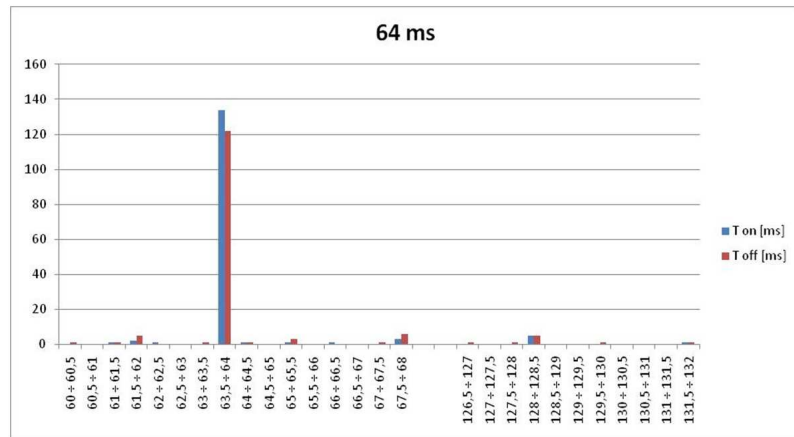
**Figura 4.9:** Topologia dei collegamenti per la quarta misura.

Così facendo si è potuto osservare una commutazione ciclica nel led d'uscita: il risultato di questo andamento è riportato nei grafici a figura 4.10 e 4.11 dove ad ogni istogramma corrisponde un diverso settaggio dei tempi di aggiornamento dell'ABIC.

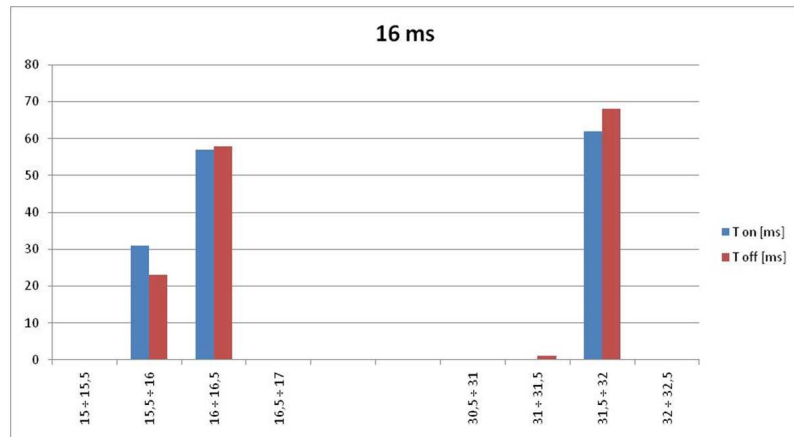
Come si può notare dai grafici riportati alle figure 4.10 e 4.11, al diminuire del tempo di aggiornamento del device, aumentano le occorrenze in cui raddoppia il semiperiodo di accensione/ spegnimento del led di uscita. Questo comportamento, seppure sporadico a tempi di aggiornamento elevati, è risultato ciclico e di costante ripetizione. Per poterlo spiegare a livello analitico si è dovuti scendere nel dettaglio sul funzionamento del chip ABIC e nella scheda EVB in cui è installato. Nelle nostre misurazioni abbiamo utilizzato solo input ed output appoggiati all'interfaccia SSC:



(a) Tempo di scheduazione impostato a 256 ms.

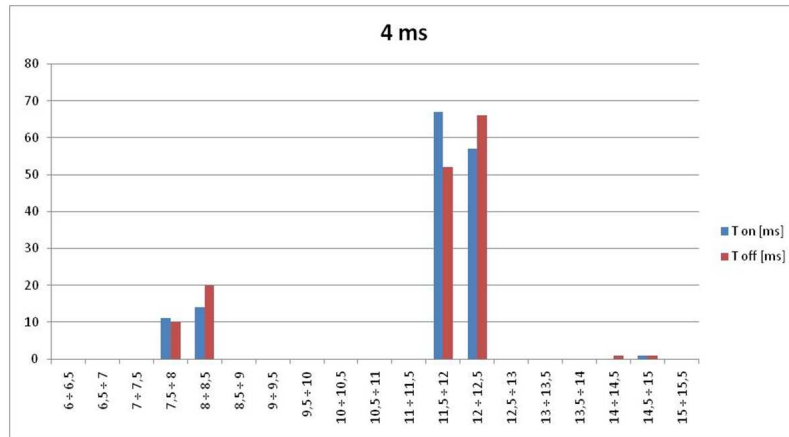


(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

Figura 4.10: Grafici riportanti misure descritte dalla configurazione 4.



**Figura 4.11:** Misura della configurazione 4 con tempo di scheduazione impostato a 4 ms.

questa presenta un funzionamento e delle tempistiche intrinseche non trascurabili ai fini delle nostre misure: il periodo con cui viene eseguito il refresh sulla seriale sincrona è di 4 ms (come evinto dall'analisi fatta in 4.2) e questo dato, ricavato dai test effettuati sul clock del ciclo di refresh della memoria del canale SSC, va ad incrementare i tempi di risposta che ci saremmo aspettati dal chip. La memoria dell'ABIC è gestita in contemporanea sia dal lato fieldbus (Profinet IO nel caso nostro) sia dal lato seriale (canali SSC ed SCI) e viene aggiornata ogni 1 ms (dato fornitoci dal supporto tecnico di HMS). Nel caso della nostra misurazione l'incidenza di questi ritardi si fa sentire almeno raddoppiata (un ciclo di refresh della memoria per la scrittura dell'uscita, un altro per la lettura dell'ingresso ed un altro eventuale ciclo può essere perso per la mancanza di sincronizzazione tra i canali): ciò si nota perfettamente nelle misurazioni eseguite con tempo di aggiornamento di 4 ms. Siamo quindi di fronte ad un fenomeno ciclico con periodo al massimo uguale a 12 ms: risulta quindi evidente come le misure effettuate con tempi di aggiornamento piccoli risentano maggiormente dello sfasamento tra tempo di aggiornamento impostato nell'ABIC ed il tempo impiegato per l'aggiornamento di scrittura/lettura della memoria. Dai grafici delle figure 4.10 e 4.11 si osserva infatti come le occorrenze, in cui il periodo non è coincidente con il tempo di aggiornamento, crescano al decrescere del tempo di aggiornamento impostato.

### 4.3.5 Misura 5

In questa misurazione si è voluto stimare la differenza tra i due istanti in cui avviene la commutazione nei due IO device ET200M e l'ABIC. Per fare ciò abbiamo predisposto la topologia come rappresentata nello schema 4.12: una sonda dell'oscilloscopio è stata collegata ad un'uscita dell'ABIC, l'altra ad un'uscita dell'ET200M. Il ritardo dell'ABIC sull'ET200M l'abbiamo misurato utilizzando una funzione di misura dell'oscilloscopio in dotazione; impostando tempi di schedulazione in OB35 e tempi di

aggiornamento uguali tra loro, abbiamo eseguito tre misurazioni, rispettivamente a 16, 64 e 256 ms raccogliendo 200 campioni per ogni misura effettuata.

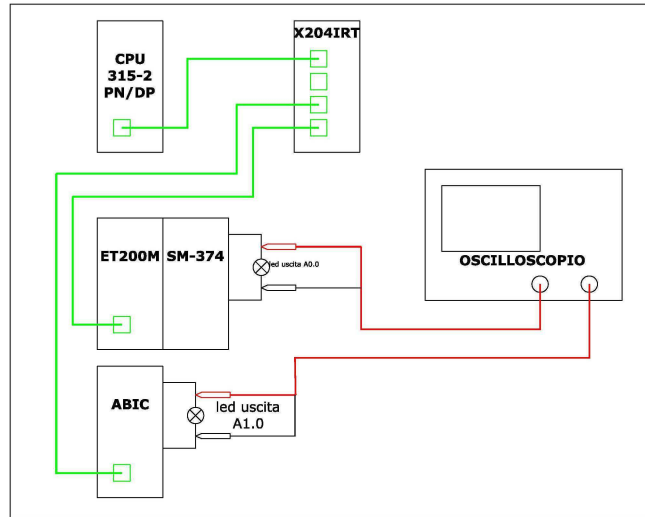


Figura 4.12: Topologia dei collegamenti per la quinta misura.

L'unico grafico riportato in figura 4.13 propone tutti i tempi misurati. Si può notare come la distribuzione sia abbastanza uniforme tra -1 e 5 ms. Il valore negativo viene facilmente spiegato: analizzando il flusso di dati con il programma Wireshark abbiamo visto che il plc comunica con ET200M ed ABIC ad intervallo di un ciclo di bus Profinet IO diverso: in un ciclo di bus viene inviato il frame verso l'ABIC, mentre in quello subito successivo (1 ms dopo) viene inviato all'ET200M.

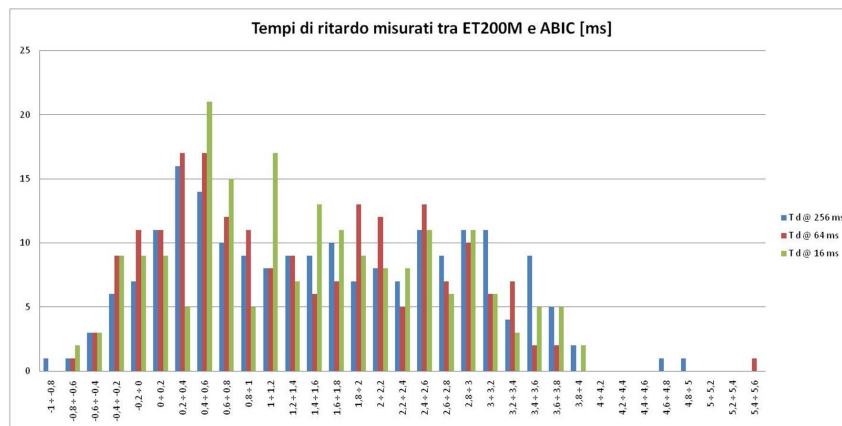


Figura 4.13: Misura della configurazione 5 con i tempi dei ritardi tra i due moduli.

La modalità con cui avviene la comunicazione è decisa automaticamente da Step7 e quindi tale valore non è modificabile; alla luce di ciò si deve pensare che i ritardi dell'ABIC andrebbero tutti traslati di 1 ms portando quindi il ritardo globale tra 0

e 6 ms. Si ha infatti che anche in questo caso i cicli di lettura scrittura del canale seriale sincrono dell'ABIC vanno a modificare sensibilmente i tempi di ritardo tra i due IO Device. La misura infatti è eseguita in modo differenziale, cioè:

$$\begin{aligned} T_{S-ABIC} &= T_{CPU} + T_{CPU-PNIO} + T_{TX} + T_{PNIO-ABIC} + T_{EL-ABIC} \\ T_{S-ET200} &= T_{CPU} + T_{CPU-PNIO} + T_{TX} + T_{PNIO-ET200} + T_{EL-ET200} + T_{\varphi} \end{aligned}$$

Dove il  $T_{\varphi}$  è il tempo di sfasamento introdotto dalla rete Profinet tra i due IO device (nel nostro caso 1 ms). Gli altri tempi sono già stati descritti in 4.3.2. Si è quindi attribuito un ritardo massimo tra i due moduli di circa 6 ms. Considerando che l'ABIC ha come tempo minimo di comunicazione fornito dal costruttore proprio 4 ms si può considerare soddisfacente la misurazione ottenuta, anche alla luce delle specifiche tecniche dell'ET200M che è un dispositivo progettato per applicazioni di rete IRT. È doveroso evidenziare come il limite di funzionamento rilevato dalle misurazioni sia da attribuire alla gestione del canale sincrono SSC, che introduce di default un tempo di ritardo massimo di 4 ms dovuto al tempo di refresh della memoria.

#### 4.3.6 Misura 6

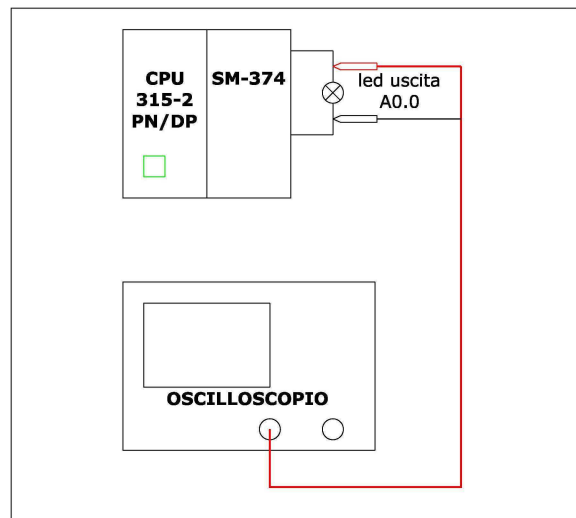


Figura 4.14: Topologia dei collegamenti per la sesta misura.

In questa misurazione abbiamo voluto testare a quanto ammontasse il tempo di ciclo del PLC con caricato il programma utilizzato nelle misure fatte finora. Sono state effettuati tre set di rilievi:

- programma caricato in OB1 senza schedulazione dell'orologio;
- programma caricato in OB35 con schedulazione dell'orologio impostata a 4 ms;



- programma caricato in OB35 con schedulazione dell'orologio impostata a 16 *ms*.

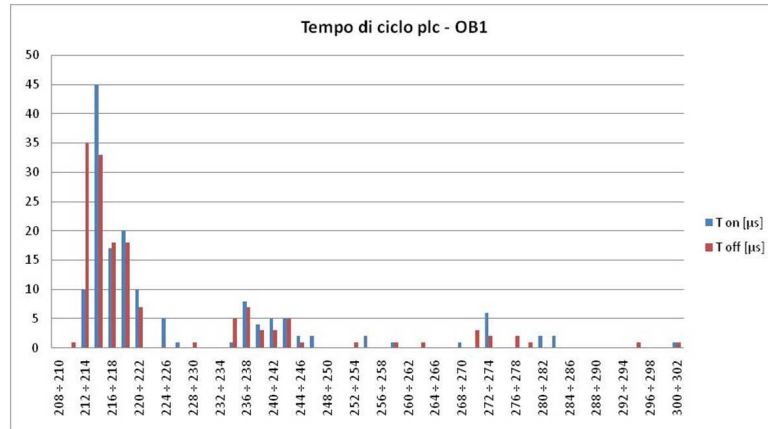
Per fare ciò abbiamo collegato i dispositivi come in figura 4.14. In queste misure non si è potuto tenere presente il ritardo della trasmissione del segnale dal plc al modulo SM374 tramite connettore backplane. Considerando che il programma genera un'onda quadra in un periodo complessivo di due cicli di esecuzione del plc, si sono fatte le seguenti considerazioni per spiegare la scelta fatta:

- l'ordine di grandezza di tale ritardo è trascurabile rispetto a quello delle misure eseguite;
- tale ritardo viene sommato sia al semiperiodo positivo, sia a quello negativo andando così ad annullarsi in una misura differenziale.

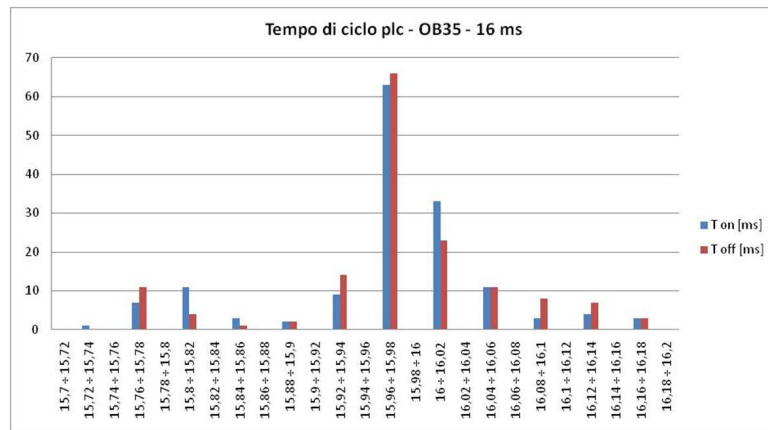
I risultati ottenuti sono riportati nei tre grafici riportati in figura 4.15.

Riportiamo infine in tabella sottostante le medie e le deviazioni standard delle ultime misure effettuate.

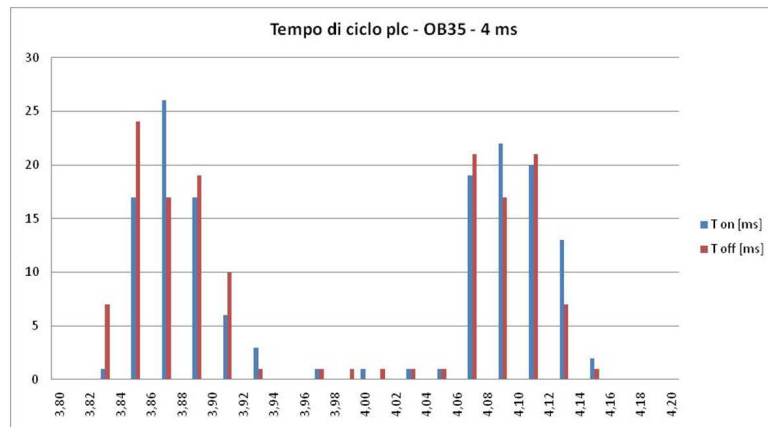
CPU			
		$\mu$	$\sigma$
Ton	OB1	226 $\mu s$	19 $\mu s$
Toff	OB1	224 $\mu s$	18 $\mu s$
Ton	OB35, 4 <i>ms</i>	3.99 <i>ms</i>	0.11 <i>ms</i>
Toff	OB35, 4 <i>ms</i>	3.98 <i>ms</i>	0.11 <i>ms</i>
Ton	OB35, 16 <i>ms</i>	15.97 <i>ms</i>	0.09 <i>ms</i>
Toff	OB35, 16 <i>ms</i>	15.98 <i>ms</i>	0.09 <i>ms</i>



(a) Programma caricato in OB1.



(b) Programma caricato in OB35 e tempi di schedulazione impostato a 16 ms.



(c) Programma caricato in OB35 e tempi di schedulazione impostato a 4 ms.

Figura 4.15: Grafici riportanti le misure descritte dalla configurazione 6.

## 4.4 Quarta fase: secondo set di misure

In una quarta fase del lavoro si è utilizzato come IO controller la stazione PC, con il pacchetto Straton integrato nel software Zenon (vedi 3.5.2). Il secondo set di misure è stato quindi eseguito con lo scopo di paragonare l'efficienza e l'efficacia del soft-PLC proposto da Copadata nei confronti dei già collaudati PLC. La piattaforma Zenon ha infatti integrato il pacchetto Straton che implementa il protocollo Profinet IO fungendo così da vero e proprio PLC.

Il team R&S di Weightpack sta progettando un sistema di controllo delle macchine riempitrici che continui a basarsi sulla CPU 315 ma che, nel caso di sovraccarico di dati provenienti dalla periferia, non costringa alla sostituzione della CPU presente con una più potente. Per questo motivo si è pensato di far svolgere queste attività aggiuntive ad un'unità decentrata, come ad esempio un PC che funge già da SCADA alla macchina: ciò comporterebbe il solo costo aggiuntivo della licenza d'utilizzo del pacchetto Straton senza la costrizione di dover modificare fisicamente l'impianto con i relativi costi d'intervento e di cablaggio. La classe 1 RT, utilizzata per la comunicazione PNIO, non necessita infatti di hardware dedicato per poter funzionare. Ecco spiegato il motivo per cui si è deciso di effettuare un secondo set di misure che possa in qualche modo confrontare il funzionamento tra soft-PLC e PLC, una volta caricati gli stessi programmi e la stessa configurazione di rete nei due IO controller.

### 4.4.1 Misura 7

La settima configurazione di misura è rappresentata nello schema 4.16 e ha lo scopo di misurare la regolarità della comunicazione tra il chip ABIC ed il softplc. Questa misura è l'analoga di quella effettuata in 4.3.2 sostituendo il softPLC al PLC. Si è

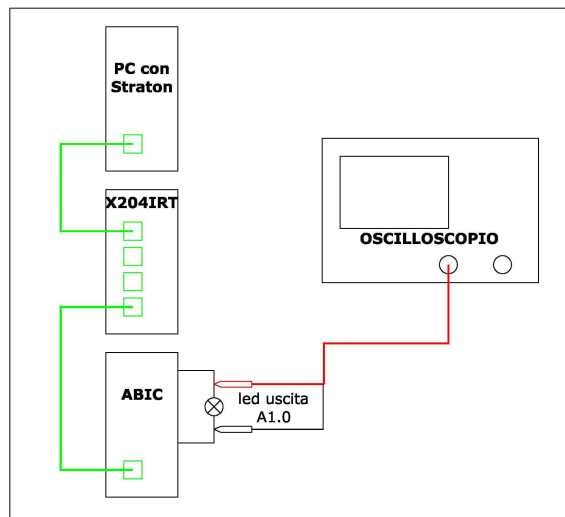


Figura 4.16: Topologia dei collegamenti per la settima misura.

caricato in Straton un programma che facesse commutare un bit d'uscita ad ogni ciclo

di schedulazione impostato manualmente nelle opzioni del software. A tale proposito si è settato lo stesso valore del tempo di schedulazione al tempo di aggiornamento dell'ABIC. Sono state effettuate misure a 16, 64 e 256 *ms* allo scopo di verificare la regolarità e la costanza delle commutazioni; si son raccolti 150 campioni per ogni semiperiodo positivo e altrettanti per quello negativo. Nella fase di acquisizione dati si è tenuto conto dell'impossibilità fisica di sincronizzare i periodi del ciclo di schedulazione di Straton ed i cicli di aggiornamento PNIO dell'ABIC e quindi si sono scartate le misure palesemente errate ai fini del senso della misurazione. Come già detto nella seconda misura (sempre in 4.3.2), conoscendo i ritardi introdotti dal canale sincrono della EVB si è deciso di non effettuare misurazioni a 4 *ms* in quanto non sarebbe stato possibile differenziare i valori errati introdotti dalla perdita di sincronismo tra schedulazione ed aggiornamento da quelli introdotti dal ritardo di elaborazione del dato del canale sincrono: queste misurazioni non sarebbero quindi state sensibili ai fini della caratterizzazione del chip.

Si riportano in figura 4.17 i grafici con i valori rilevati in fase di misurazione, da questi si evince una scarsa precisione nell'intorno del valore atteso, specie nella misura con tempo di aggiornamento dell'ABIC e tempo di schedulazione dell'orologio settati a 256 *ms*, valori impostato dal tool di configurazione di Straton: come si può vedere in figura 4.17(a) lo scostamento massimo rilevato è arrivato ad un massimo di 10 *ms* dal valore atteso.

#### 4.4.2 Misura 8

L'ottava configurazione ripropone la stessa tipologia di misura fatta in 4.4.1. In questo caso si è testato il modulo ET200M anziché l'ABIC sia per comparare i due IO device sia, nel contempo, per fare una verifica incrociata sul funzionamento dell'IO controller. Anche in questo caso sono state effettuate misure a 16, 64 e 256 *ms* allo scopo di verificare l'accuratezza e la regolarità delle commutazioni; si son raccolti 150 campioni per ogni semiperiodo positivo e altrettanti per quello negativo.

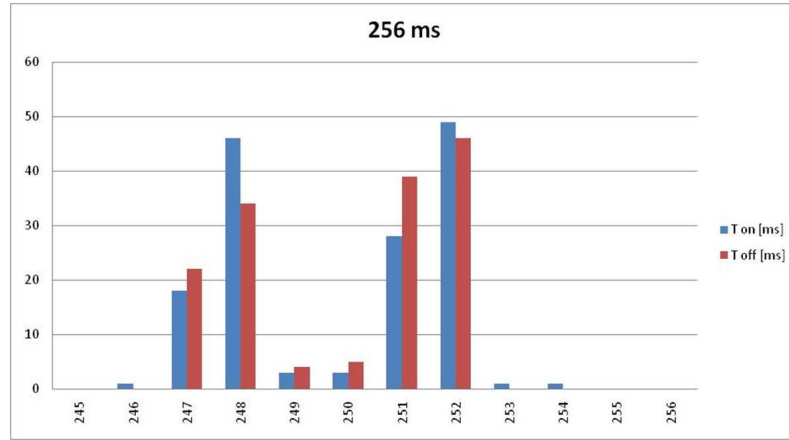
Si riportano in 4.19 i grafici con i valori rilevati durante le misurazioni.

Nella tabella di seguito riportiamo le medie e le deviazioni standard delle misurazioni di entrambi i moduli utilizzando Straton come IO controller.

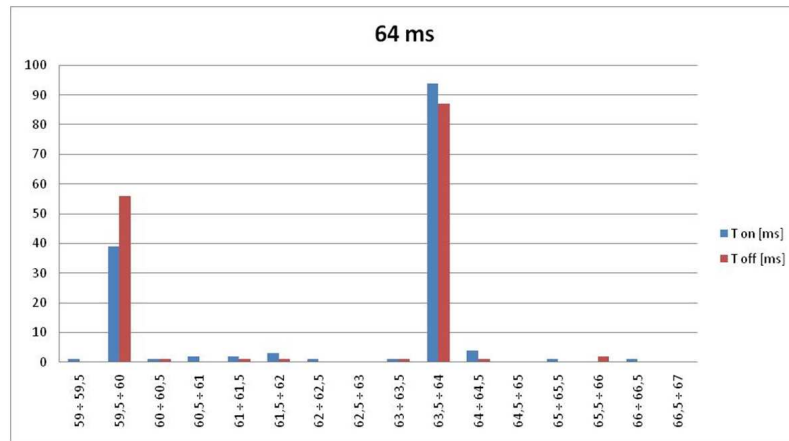
misura 7 - abic			
		$\mu[ms]$	$\sigma[ms]$
Ton	256ms	249.9	2.00
Toff	256ms	250.0	1.92
Ton	64ms	62.6	1.80
Toff	64ms	62.3	1.98
Ton	16ms	15.6	1.18
Toff	16ms	15.6	1.15

misura 8 - ET200M			
		$\mu[ms]$	$\sigma[ms]$
Ton	256ms	250.0	0.50
Toff	256ms	250.0	0.55
Ton	64ms	62,4	0.40
Toff	64ms	62,4	0.45
Ton	16ms	15.6	0.32
Toff	16ms	15.6	0.38

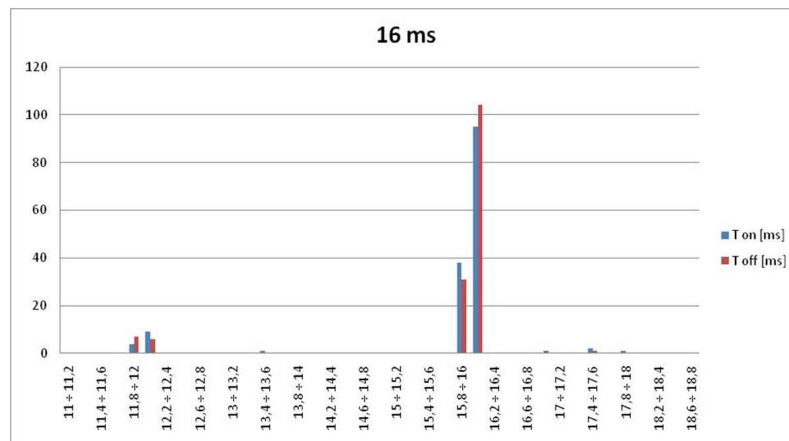
Si nota come le deviazioni standard dell'ET200M variano di poco il loro valore al variare dei tempi di aggiornamento/schedulazione: questo risultato era già stato



(a) Tempo di scheduazione impostato a 256 ms.

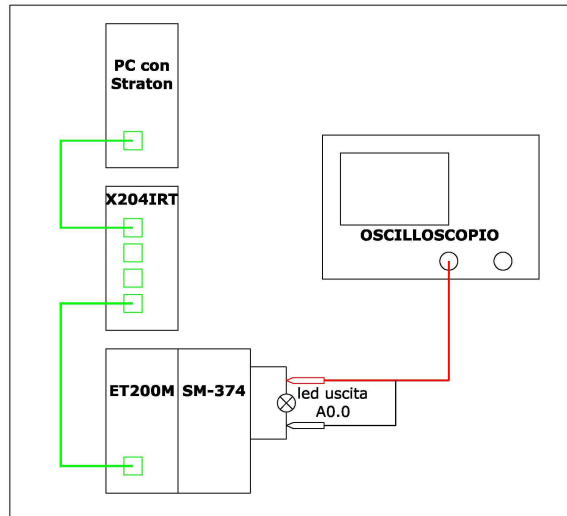


(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

Figura 4.17: Grafici riportanti le misure descritte dalla configurazione 7.



**Figura 4.18:** Topologia dei collegamenti per l'ottava misura.

rilevato nelle misure due e tre (riportate in tabella a pagina 56) e queste misurazioni confermano come tale valore continui a rientrare nelle specifiche riportate dal costruttore, che dichiara tempi di commutazioni delle uscite entro i  $500 \mu s$ .

Nell'ABIC le deviazioni standard crescono all'aumentare del tempo di aggiornamento della schedina; tale comportamento è decisamente più marcato di quello già visto in tabella a pagina 56.

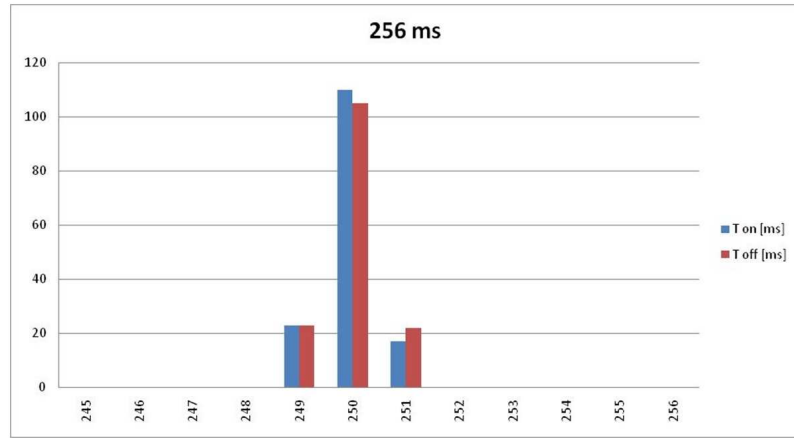
Si nota anche come il valore della media si scosti dal valore atteso settato nella schedulazione dell'orologio: ciò fa pensare ad una scarsa precisione nella gestione del clock che gestisce il tempo di ciclo di Profinet IO all'interno di Straton.

Si evince quindi come l'ET200M risulti più accurato dell'ABIC anche in questa occasione, come la media sia discordante dai valori duali riportati in tabella a par.4.3.3. Tale discordanza è imputabile al tipo di IO controller che gestisce la comunicazione PNIO con il device di turno, sia ABIC o ET200M. Ulteriori considerazioni saranno riprese nel paragrafo 4.5.

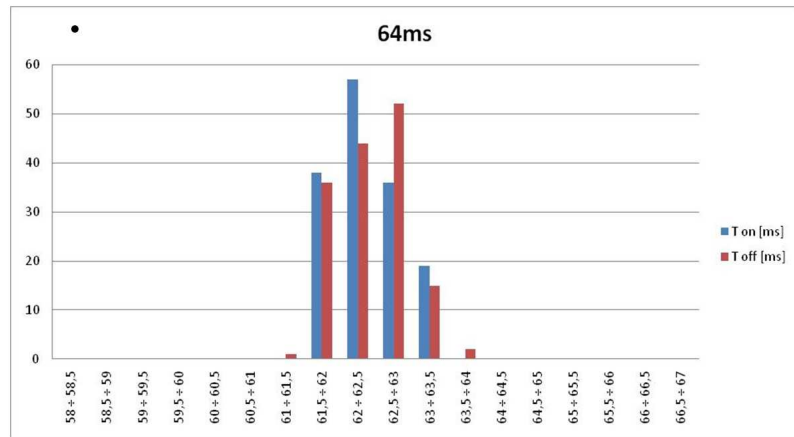
#### 4.4.3 Misura 9

Come corrispettiva della quarta misura, nella nona si è voluto testare il tempo complessivo di lettura-scrittura nella scheda ABIC. La topologia è rappresentata nello schema 4.20: nell'EVB è stato effettuato un ponte fisico tra il primo bit del byte 1 d'ingresso ed il primo bit del byte 1 d'uscita (stesso ponte utilizzato in misura 4). In aggiunta si è caricato nel software Straton un programma che settasse ad ogni ciclo l'uscita A1.0 come negazione dell'ingresso letto E1.0. Il programma è stato caricato senza impostare tempi di schedulazione onde evitare di introdurre misure fallaci.

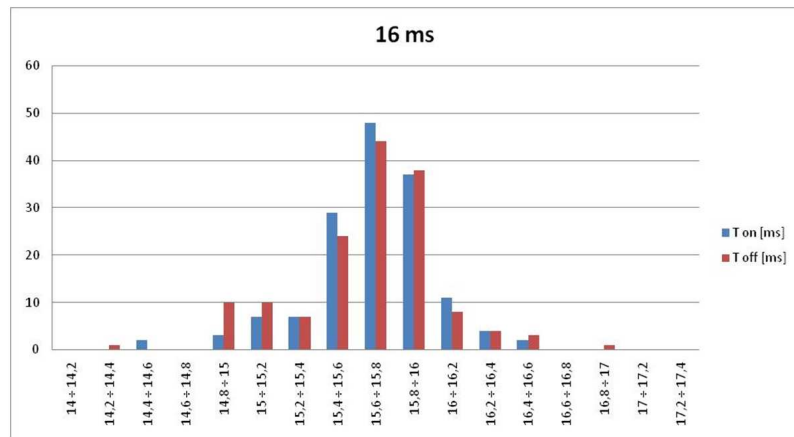
Così facendo si è potuto osservare una commutazione ciclica nel led d'uscita: il risultato di questo andamento è riportato nei grafici 4.21(a), 4.21(b) e 4.10(c) dove



(a) Tempo di scheduazione impostato a 256 ms.

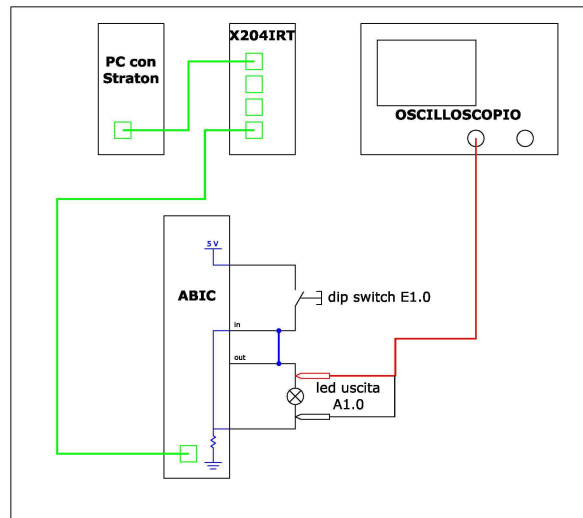


(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

Figura 4.19: Grafici riportanti le misure descritte dalla configurazione 8.



**Figura 4.20:** Topologia dei collegamenti per la nona misura.

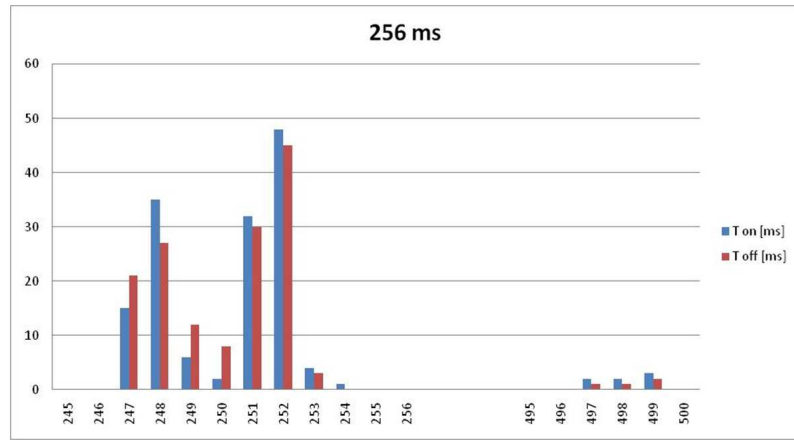
ad ogni istogramma corrisponde un diverso settaggio dei tempi di aggiornamento dell'ABIC, come riportato nel titolo dei vari grafici.

Come si può notare dai grafici 4.21(a), 4.21(b) e 4.21(c), al diminuire del tempo di aggiornamento del Device, aumentano le occorrenze in cui raddoppia il semiperiodo di accensione/ spegnimento del led di uscita. Questo fenomeno, che si è già commentato in 4.3.4, è effetto del tempo di refresh della memoria del canale sincrono di comunicazione dell'ABIC e, nostre misurazioni, abbiamo utilizzato solo input ed output appoggiati all'interfaccia SSC dell'Evaluation Board. Nel caso della nostra misurazione l'incidenza di questi ritardi si fa sentire almeno raddoppiata (un ciclo di refresh della memoria per la scrittura dell'uscita ed un altro per la lettura dell'ingresso). Risulta evidente come le misure effettuate con tempi di aggiornamento piccoli risentano maggiormente dell'asincronismo tra tempo di aggiornamento impostato per l'ABIC ed il tempo impiegato per l'aggiornamento di scrittura/lettura della memoria. Guardando i grafici si osserva infatti come le occorrenze, in cui il periodo non è coincidente con il tempo di aggiornamento impostato, crescano al decrescere del tempo di aggiornamento impostato. Anche in questi grafici, come in quelli di misura 7 ed 8, si evidenzia una scarsa accuratezza del tempo di aggiornamento misurato, dovuto alla scansione temporale dell'IO controller, che è costante in tutte le misure effettuate.

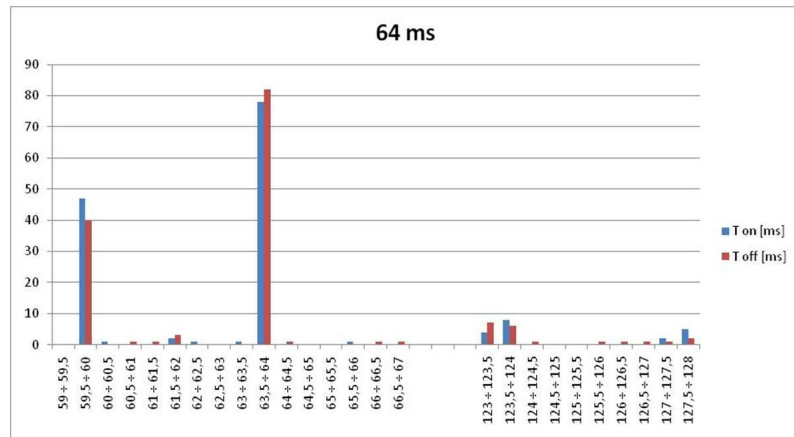
#### 4.4.4 Misura 10

Parallelamente a quanto stimato in misura 5 (4.3.5) si è deciso di misurare la differenza tra i due istanti in cui avviene la commutazione nelle uscite dei due IO device ET200M e l'ABIC. Per fare ciò abbiamo predisposto la topologia come rappresentata nello schema 4.22: una sonda dell'oscilloscopio è stata collegata ad un'uscita dell'ABIC, l'altra ad un'uscita dell'ET200M. Il ritardo dell'ABIC sull'ET200M l'ab-

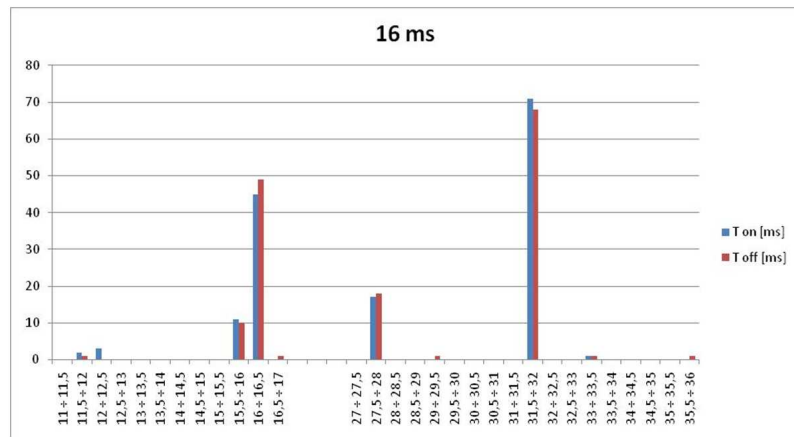




(a) Tempo di scheduazione impostato a 256 ms.



(b) Tempo di scheduazione impostato a 64 ms.



(c) Tempo di scheduazione impostato a 16 ms.

Figura 4.21: Grafici riportanti le misure descritte dalla configurazione 9.

biamo misurato utilizzando una funzione di misura dell'oscilloscopio in dotazione; impostando tempi di schedulazione sul software Straton ed i tempi di aggiornamento uguali tra loro; abbiamo quindi eseguito tre misurazioni, rispettivamente a 16, 64 e 256 ms.

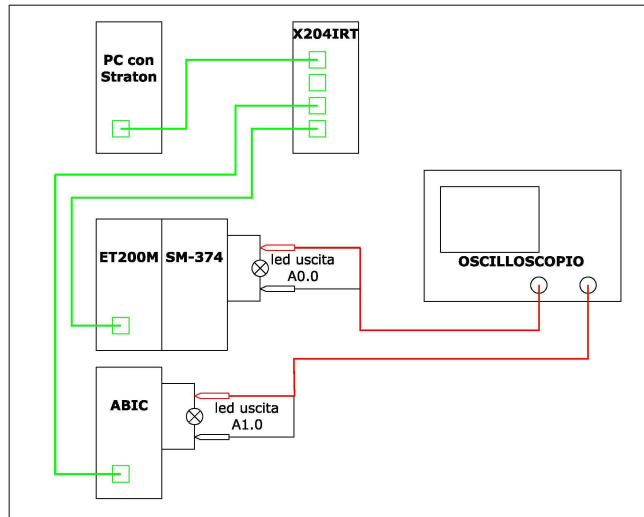


Figura 4.22: Topologia dei collegamenti per la decima misura.

Il grafico riportato in figura 4.23 propone i tempi misurati. Si può notare come la distribuzione sia abbastanza uniforme tra -1 e 5.5 ms. Il valore negativo viene così spiegato: analizzando il flusso di dati con Wireshark si è visto che il PLC comunica con l'ET200M e l'ABIC ad intervallo di un ciclo di bus PNIO diverso: in un ciclo di bus viene inviato il frame verso l'ABIC, mentre in quello subito successivo (1 ms dopo) viene inviato all'ET200M.

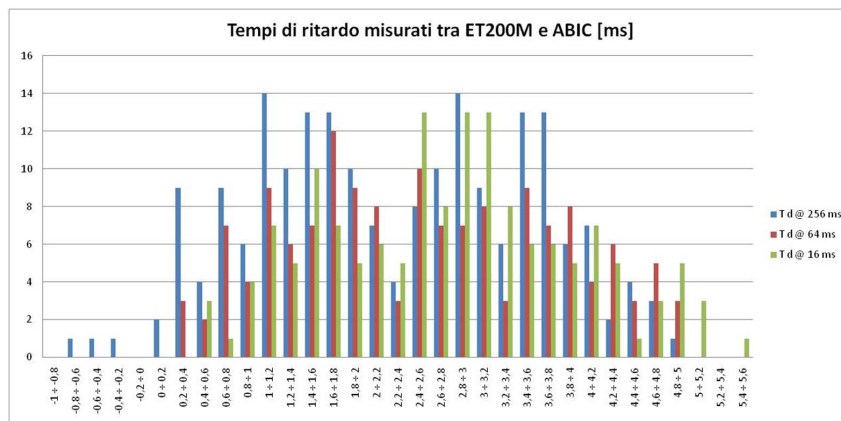


Figura 4.23: Misura della configurazione 10 con i tempi dei ritardi tra i due moduli.

La modalità con cui avviene la comunicazione è decisa automaticamente manualmente nelle Opzioni di Straton e, per creare un'analogia con le misure effettuate nel

primo set, abbiamo impostato il tempo di ciclo PNIO ad 1 *ms*, inserendo un valore di Reduction Ratio pari a 32.

Anche in questa misura i cicli di lettura scrittura del canale seriale sincrono dell'ABIC vanno a modificare sensibilmente i tempi di ritardo tra i due IO Device. La misura infatti è eseguita in modo differenziale, come avevamo già visto in Misura 5 (4.3.5), cioè:

$$\begin{aligned} T_{S-ABIC} &= T_{CPU} + T_{CPU-PNIO} + T_{TX} + T_{PNIO-ABIC} + T_{EL-ABIC} \\ T_{S-ET200} &= T_{CPU} + T_{CPU-PNIO} + T_{TX} + T_{PNIO-ET200} + T_{EL-ET200} + T_{\varphi} \end{aligned}$$

Dove:

- $T_{CPU}$  è il tempo impiegato dalla cpu per eseguire il ciclo macchina, sarà circa uguale al valore di schedulazione dell'orologio impostato in Straton;
- $T_{CPU-PNIO}$  è il tempo di reazione impiegato dalla cpu a scrivere i dati nel buffer di comunicazione del sistema Profinet, in questo caso sarà la scheda di rete Ethernet del PC;
- $T_{TX}$  è il tempo impiegato dalla rete Profinet a trasferire il dato dal buffer della cpu a quello dell'ABIC, compreso il passaggio nello switch;
- $T_{PNIO-ABIC}$  è il tempo di reazione impiegato dal dato a passare dal buffer Profinet dell'ABIC al chip stesso;
- $T_{PNIO-ET200}$  è il tempo di reazione impiegato dal dato a passare dal buffer Profinet dell'ET200M all'unità stessa;
- $T_{EL-ABIC}$  è il tempo di reazione impiegato dall'ABIC a trasmettere i dati dal chip fino alla commutazione del led d'uscita A1.0; essendo il periodo di aggiornamento del canale sincrono utilizzato al massimo 4*ms*, questo tempo sarà nel caso peggiore pari a 4*ms* (dalle considerazioni fatte in 4.2);
- $T_{EL-ET200}$  è il tempo di reazione impiegato dall'ET200M a trasmettere i dati dall'unità di ricezione fino allo slot che gestisce la commutazione del led d'uscita A0.0, il costruttore indica come ritardo massimo 500 *ms* (come riportato già in sezione 3.3);
- $T_{\varphi}$  è il tempo di sfasamento introdotto dalla rete Profinet tra i due IO device (nel nostro caso 1 *ms*).

Si è quindi attribuito un ritardo massimo tra i due moduli di circa 6 *ms*. Considerando che l'ABIC ha come tempo minimo di comunicazione fornito dal costruttore proprio 4 *ms* si può considerare soddisfacente anche questa misurazione ottenuta. È doveroso evidenziare come il limite di funzionamento rilevato dalle misurazioni sia da attribuire ancora una volta alla gestione del canale sincrono SSC, che introduce di default un tempo di ritardo massimo di 4 *ms* sempre dovuto al tempo di refresh della memoria.

## 4.5 Valutazioni

Osservando i risultati delle misurazioni, si ricava immediatamente la differenza nelle prestazioni per quanto riguarda la gestione temporale del ciclo di comunicazione Profinet IO tra il chip ABIC e l'unità di periferia decentrata ET200M, valori riportati nelle tabelle alle pagine 56 e 66. La differenza è dovuta al tipo di tecnologia adottata dall'ET200M: hardware dedicato esclusivamente alla comunicazione con capacità di gestione del traffico isocrono Real Time.

Da un'attenta analisi delle due tabelle si nota un errore costante quando la rete è gestita dallo Straton; tale errore aumenta al crescere del tempo di schedulazione impostato: motivo per cui a 256 ms risulta più evidente in termini assoluti. Tale differenza nelle medie è riportata, insieme all'errore relativo medio, nella tabella riassuntiva 4.1.

MISURA N°		2		3		7		8	
$T_{\text{schedulazione}}$ [ms]		$\mu$ [ms]	$E_{\%}$	$\mu$ [ms]	$E_{\%}$	$\mu$ [ms]	$E_{\%}$	$\mu$ [ms]	$E_{\%}$
Ton	256	255.6	0.16	255.4	0.23	249.9	2.39	250.0	2.34
Toff	256	255.7	0.12	255.7	0.12	250.0	2.34	250.0	2.34
Ton	64	64.0	0.00	64.0	0.00	62.6	2.19	62,4	2.50
Toff	64	63.9	0.16	63.9	0.16	62.3	2.65	62,4	2.50
Ton	16	16.1	0.63	15.9	0.63	15.6	2.50	15.6	2.50
Toff	16	16.0	0.00	16.0	0.00	15.6	2.50	15.6	2.50

**Tabella 4.1:** Tabella riassuntiva con medie ed errori relativi delle misure 2, 3, 7 e 8.

La differenza tra le medie che si riscontra nella tabella 4.1 tra le misure 2-3 e le misure 7-8 è dovuta all'utilizzo di un sistema soft-PLC avente il clock di Profinet poco preciso. Il tool Straton mette a disposizione due tipi di kernel, uno di tipo SRT ed uno HRT. In un primo momento si è provato il kernel HRT ma la sua struttura è risultata troppo stringente per le caratteristiche della comunicazione instaurata tra Straton e l'ABIC. Utilizzando in seguito il kernel SRT di Straton si è riusciti ad instaurare una corretta comunicazione Profinet ed effettuare le misurazioni volute (vedi 4.4). Il SRT kernel viene trattato dalla stazione PC come fosse una task di Windows, non dedicando così le necessarie risorse per garantire una precisione paragonabile a quella del PLC. È da sottolineare che comunque l'errore relativo medio rimane confinato in un margine del 2.65% del valore atteso, come riportato nella tabella riassuntiva 4.1. Altra nota non superfiale e doverosa da segnalare, specie in un lavoro di questo tipo, è l'efficienza dell'assistenza offerta dal costruttore di un prodotto: durante il lavoro sono stati contattati più volte i centri assistenza di Siemens, HMS e Copa-data. Il call-center di Siemens, rodato e ormai maturo dai molti anni di esperienza, è stato impeccabile, sia in tempi di risposta, sia nella qualità delle informazioni date. Il centro assistenza Italia di HMS, seppure quantitativamente più ridotto, è stato molto efficiente e disponibile nel collaborare alla soluzione delle svariate problematiche

insorte nel corso delle misure.

Infine, il centro di assistenza di Copa-data, che è stato contattato in occasione di una problematica sull'interfacciamento tra ABIC e Straton, si è dimostrato poco presente nel supporto al cliente. Il problema suddetto è infine stato risolto senza ricevere alcun aiuto.



## Conclusioni

In questo lavoro sono state realizzate ed analizzate due tipologie di reti di comunicazione Profinet IO, la prima costituita da CPU315-2 PN/DP, Anybus-IC ed ET200M e descritta nel paragrafo 4.3, la seconda composta da Straton su Zenon come soft-PLC, Anybus-IC ed ET200M, descritta in 4.4.

Lo svolgimento del lavoro ha portato a dare maggior attenzione alle seguenti voci:

- la caratterizzazione del chip AnyBus-IC PRT in funzione del tipo di utilizzo che ne vorrà fare Weightpack in un prossimo impiego futuro;
- l'utilizzo in prova e vari test sulle funzioni del pacchetto Zenon+Straton fornito da Copa-data per un eventuale integrazione nelle macchine imbottigliatrici Weightpack.

Sebbene la rete Profinet IO offra prestazioni potenzialmente formidabili, tali performance risultano eccessive se, come nel nostro caso, si pensa al ritardo di ben  $4\text{ ms}$  introdotto dal canale sincrono SSC dell'ABIC; questo spiega il motivo per cui, nella totalità delle misure effettuate sull'ABIC, il semiperiodo di accensione o spegnimento del led avesse talvolta un ritardo di  $4\text{ ms}$  (o multiplo nel caso di circuito con il ponticellamento tra out e in), dovuto alla perdita della periodicità tra il tempo di refresh della memoria del canale SSC ed il tempo di aggiornamento del dispositivo. Sostanzialmente per questo motivo si è deciso di non ritenere affidabili, e quindi di non riportare in questa tesi, quelle misurazioni effettuate con tempi di aggiornamento e di schedulazione orologio di  $4\text{ ms}$ : non si sarebbe potuto scindere se le occorrenze, in cui il semiperiodo raddoppiava, dipendessero dalla perdita di periodicità tra tempo di aggiornamento e tempo di schedulazione orologio oppure dalla perdita di periodicità tra tempo di aggiornamento e tempo di refresh della memoria.

È tuttavia da sottolineare come i risultati delle misure siano qualitativamente soddisfacenti: i valori rientrano appieno sia nelle specifiche tecniche del produttore, confermando la bontà del prodotto, sia nelle specifiche del protocollo Profinet RT classe 1, che consente una variazione massima del 15% rispetto al valor medio (come da figura 2.2).





# Bibliografia

- [1] P. Neumann  
*Communication in industrial automation - What is going on?*  
Elsevier, 2006
- [2] S. Vitturi  
*On the use of Ethernet at low level of factory communication systems*  
Elsevier, 2001
- [3] M. Felser  
*Real-Time Ethernet - Industry Prospective*  
IEEE, 2005
- [4] P. Doyle  
*Introduction to Real-Time Ethernet*  
Dal periodico *Extension*, A technical Supplement to *Control Network*  
2004
- [5] P. Ferrari, A. Flammini, D. Marioli, A. Taroni  
“Valutazioni sperimentali delle prestazioni di Profinet”  
Dal periodico *Brescia Ricerche*, n.52  
2005
- [6] J. Feld  
*PROFINET - Scalable factory communication for all application*  
IEEE, 2004
- [7] F. Venturini  
*I bus di campo isocroni Profibus DP-V2 e Profinet IO*
- [8] PROFINET System Description  
*Manuale di sistema PI*
- [9] P. Ferrari, A. Flammini, D. Marioli, A. Taroni, F. Venturini  
*Experimental analysis to estimate jitter in PROFINET IO Class 1 networks*  
IEEE, 2006

- [10] P. Ferrari, A. Flammini, D. Marioli, A. Taroni  
*An experimental approach to estimate real-time characteristic of PROFINET IO versus PROFIBUS DP-V2*  
2006
- [11] H. Kleines, S. Detert, M. Drochner, F. Suxdorf  
*“Performance aspects of PROFINET IO”*  
IEEE “Transactions on nuclear science”, vol. 55, n°1 Febbraio 2008  
[www.ieee.org](http://www.ieee.org)
- [12] Dal manuale *“Comunicazione PROFIBUS DP/PROFINET IO”*  
Sito web *Siemens*:  
[www.automation.siemens.com](http://www.automation.siemens.com)
- [13] P. Ferrari, A. Flammini, S. Vitturi  
*Performance analysis of PROFINET networks*  
Elsevier, 2005
- [14] P. Neumann, A. Poschmann  
*Ethernet-based Real-Time Communications with PROFINET IO*
- [15] A. Zuin  
*Caratterizzazione e analisi di una rete Ethernet industriale*  
Tesi di laurea, 2008
- [16] D. Cattai  
*Simulazione del funzionamento del protocollo PROFINET IO*  
Tesi di laurea, 2007
- [17] SIEMENS - SIMATIC  
*PROFINET - Descrizione del sistema*  
Manuale di sistema, 2008
- [18] SIEMENS - SIMATIC  
*PROFINET IO - Getting Started: Collection*  
Manuale di sistema, 2008
- [19] SIEMENS - SIMATIC  
*Configurazione dell’hardware e progettazione con Step 7*  
Manuale, 2006
- [20] SIEMENS - SIMATIC  
*Sistema di automazione S7-300 - Getting Started Collection*  
Manuale di sistema, 2008
- [21] SIEMENS - SIMATIC  
*Sistema di automazione S7-300 - Caratteristiche delle unità modulari*  
Manuale del prodotto, 2009

- [22] SIEMENS  
*Switch Industrial Ethernet, SCALANCE X-200*  
Istruzioni Operative, 2008
- [23] SIEMENS - SIMATIC  
*S7-300, CPU 31xC e CPU 31x: Dati Tecnici*  
Manuale del prodotto, 2009
- [24] HMS  
*Anybus-IC PROFINET IO - Fieldbus Appendix, rev 1.02*  
Doc.Id. SCM-1200-036
- [25] HMS  
*Anybus-IC*  
Design Guide, rev 1.59
- [26] HMS  
*Anybus-IC Evaluation Board*  
User Manual, rev 1.01  
Doc. no. ABSICEVB-UM-1.01E
- [27] HMS  
*Anybus-IC - Single Chip Solutions*  
Brochure
- [28] Copa Data  
<http://www.copadata.com>
- [29] Zenon  
*SCADA Systems using IEC 61850*
- [30] Copa Data  
*Zenon Manual*



# Ringraziamenti

Tralasciando ora la terza persona singolare che ha accompagnato questa mia tesi sento come dovere, ma anche come occasione che probabilmente non si ripeterà, di scrivere queste righe. In questo giorno si chiude per me il percorso degli studi, un lungo capitolo segnato da molte tappe, iniziato ai tempi delle elementari fino agli ultimi anni dell'università. Per questo mi sento, a modo mio, di ringraziare tutte quelle persone che ho incontrato nel mio cammino e con cui ho condiviso momenti importanti e che in qualche modo hanno contribuito a formare la persona che oggi sono.

Parto ringraziando mio papà e mia mamma, per come mi hanno tirato su, mi hanno fatto crescere su solidi principi morali, educandomi al rispetto verso ogni persona ed ogni cosa riuscendo a non farmi mai mancare nulla: grazie per avermi insegnato a non dare mai niente per scontato.

Mio fratello, che riesce a starmi vicino con la sua discrezione e che, sotto quel personaggio un po' burbero ed un po' impulsivo, nasconde una sensibilità che solo poche persone hanno. Stringi i denti che tra un po' toccherà a te!

Come non ringraziare i miei cari nonni Bruno e Mercedes, che fin da quando sono piccolo hanno saputo starmi vicino prendendosi cura di me, sgridandomi quando serviva, insegnandomi un sacco di cose; mi avete trasmesso quegli "ormai antichi" sapori della vita che noi giovani d'oggi non abbiamo modo di gustare. Un grazie per avermi saputo guidare in tutti questi anni!

Un caro ricordo va a nonno Mansueto e a nonna Argia che, sebbene non abbia avuto l'opportunità di conoscervi bene, son certo che da lassù mi avete sempre accompagnato e tenuto d'occhio.

Un ringraziamento va ai miei numerosi zii, in particolare a Bruno, Rina, Mariella, Bruna, Ivo, Giancarlo, Antonietta e Gina che hanno saputo supportarmi, ascoltarmi e consigliarmi nei vari momenti della mia vita in cui necessitavo di qualche indirizzata o solo di qualche consiglio; grazie per esserci sempre stati!

Un grazie ai miei cugini, Massimo, Enrico, Francesca, Francesco, Silvia e Benedetta, con i quali, più degli altri, ho avuto la possibilità di creare dei legami, di confrontarmi e di crescere.

Tempo fa mi ero riproposto di evitare di scrivere nome per nome tutte le persone

da ringraziare, forse per la paura di saltare qualcuno, forse per la pigrizia di dover scrivere ben altre due pagine...fatto sta che mi sembra giusto correrlo questo rischio, a costo di rischiare lo scivolone... Partendo un po' dagli albori ringrazio quelli che sono gli amici con cui ho condiviso parte della mia adolescenza e della giovinezza: Alberto, Ale, Alessia, Alice, Bob, Briskije, Ciups, Cri, Daniele, Dario, Elena, Evy, Filippo, Federica, Francesco, Gabri, don Giorgio, Giulio, Irene, Ivana, Laura, Luca, Manuel, Mara, Marchetto, Massimo, Matteo, Mauro, don Marco, Mela, Nicoea, Ponch, Roberto, le 2 Sara, Simone, Tia, Toni, le 3 Vale, Walter e Zeta.

È difficile trarre le fila di tutti questi anni di vita studentesca ma mi è altrettanto impossibile non ricordare le splendide avventure vissute con don Marco, la Mela, Tia, la Vale, Cri ed i "bocioni"; vi ringrazio per esservi messi in gioco con me, per aver impiegato un sacco di tempo a preparare quelle attività e quegli incontri i cui risultati ci riempivano sempre il cuore. Sebbene i tempi siano cambiati e le nostre vite abbiano preso strade diverse vi ringrazio per la vostra costante presenza!

Un grazie a chi mi è stato vicino in questi ultimi mesi, avete saputo a modo vostro sopportarmi, incitarmi ed anche sopprimermi, Cri, Ale, Vale, Tia, Vale, Martina, Mela, Alessia ed Irene.

Un ringraziamento ai numerosi compagni di banco incrociati in questi anni di vita universitaria; mi preme ringraziare soprattutto quelli con cui ho avuto modo di instaurare un rapporto Luca, i ben 4 Marco (el volpe, barchitta, spill e tegon),Ale M., Diego, Valentino, Tiziano, Leo, Stefano S., Paolo, Carlo, Eros, Stefano B., Matteo, Bort, Daniele, Francesco, Daniela, Cristina, Chiara, Sebastiano, Ale R., Gio, Paolo, Max e Francesco.

Un ringraziamento a Luca, mio compagno di tesi, con cui ho avuto modo di spartire gioie alternate a sbattimenti di capo.

Un ringraziamento particolare al Prof. Stefano Vitturi per la costante presenza in tutto il periodo di tesi; la ringrazio per i suoi preziosi consigli, non solo accademici, che ha saputo darmi nel corso di questi mesi.

Allo stesso modo ringrazio l'Ing. Paolo Marogna di Weightpack che mi ha dato la possibilità di svolgere questo lavoro con l'autonomia necessaria riuscendo a non farmi mai mancare la sua presenza nell'affrontare i problemi insorti nel mentre.

Un ringraziamento anche all'Ing. Giorgio Berinzaghi di HMS Italia e all'Ing. Andrea Zuin di Siemens per la loro disponibilità nell'aiutarmi a comprendere la natura dei problemi tecnici che si sono riscontrati specie in fase di configurazione della rete in laboratorio.

Ben conscio che il traguardo raggiunto in questo giorno non è altro che un nuovo punto di partenza, mi auguro di continuare ad essere circondato da tutti voi e da nuove persone in gamba, come quelle che ho avuto il piacere di incontrare fino ad oggi.