Reti Real-Time: Protocolli per l'Automazione Industriale

Nicolò Toscani

UNIVERSITÀ DI PARMA
DIPARTIMENTO DI SCIENZE MATEMATICHE, FISICHE E INFORMATICHE
Corso di Laurea Triennale in Informatica

20 Dicembre 2018

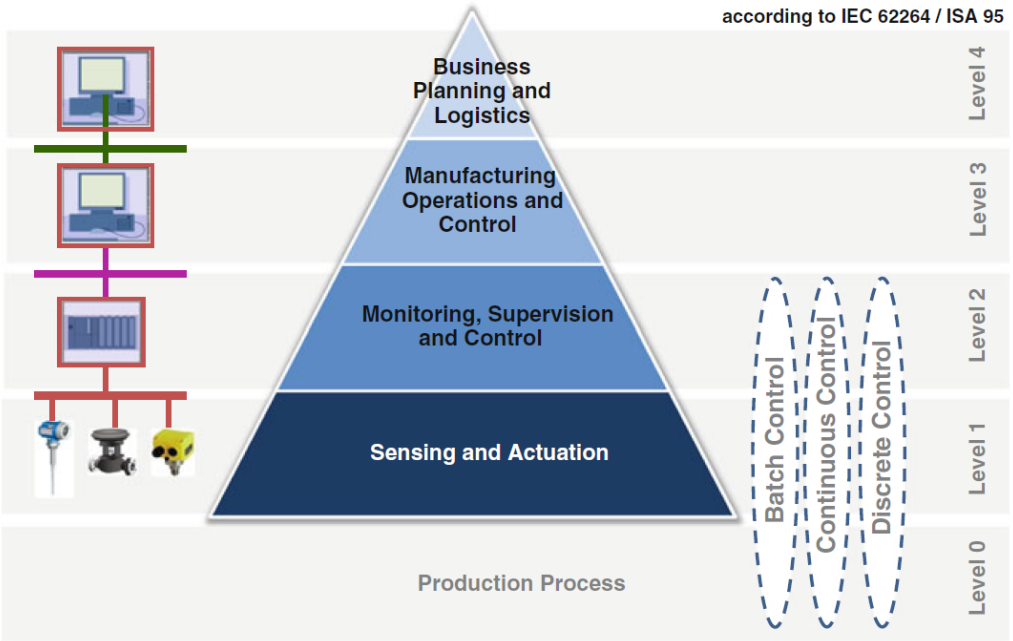
Obiettivi

- Analisi dei requisiti richiesti dalle reti utilizzate nel settore dell'automazione industriale
- Analisi e confronto tra due dei protocolli derivati: EtherNet/IP e EtherCAT
- Verifica delle caratteristiche dei due protocolli attraverso l'analisi dei pacchetti

Il flusso dei dati

Un sistema produttivo industriale può essere definito mediante un modello teorico ***CIM: Computer Integrated Manufacturing***. L'intero sistema deve essere interconnesso per garantire il flusso delle informazioni:

- Comunicazione verticale e orizzontale
- Differenti caratteristiche sui vari livelli



Il sistema di comunicazione

I livelli che richiedono requisiti più stringenti sono:

- Livello di campo: comprende i componenti hardware che eseguono fisicamente le trasformazioni necessarie per la produzione ed il controllo (sensori, attuatori)
- Livello di macchina: controlla e pianifica le azioni da inviare al livello di campo (PLC)

Principali requisiti:

- Comunicazione real-time
- Dati di piccola dimensione, non strutturati e con elevata frequenza di trasmissione

Real-Time

In un processo industriale lo scambio di informazioni avviene tra *task* in esecuzione su un sistema real-time.

Un *sistema real-time* è un sistema la cui correttezza logica non dipende solo dal risultato in uscita ma anche dall'istante temporale in cui tale risultato viene reso disponibile.

Principali requisiti per la comunicazione:

- Determinismo: è possibile calcolare un tempo massimo entro il quale l'informazione verrà trasmessa
- Isocronia: garantire tempi di consegna dei pacchetti ripetitivi ed equidistanti tra loro

Comunicazione real-time

Possiamo identificare tre tipi di messaggi che viaggiano sulla rete:

- **Messaggi impliciti**: generati e utilizzati da task periodici
- **Messaggi espliciti**: generati e usati da task aperiodici
- **Messaggi sporadici**: generati da task aperiodici con vincoli hard real-time

Le specifiche di una rete RT vengono rilassate definendo delle soglie di tollerabilità:

- Missed Packet Rate: pacchetti non arrivati in tempo
- Lost Packet Rate: percentuale dei pacchetti persi
- Delay Jitter: indice di valutazione sulla varianza del ritardo medio subito dai pacchetti della rete

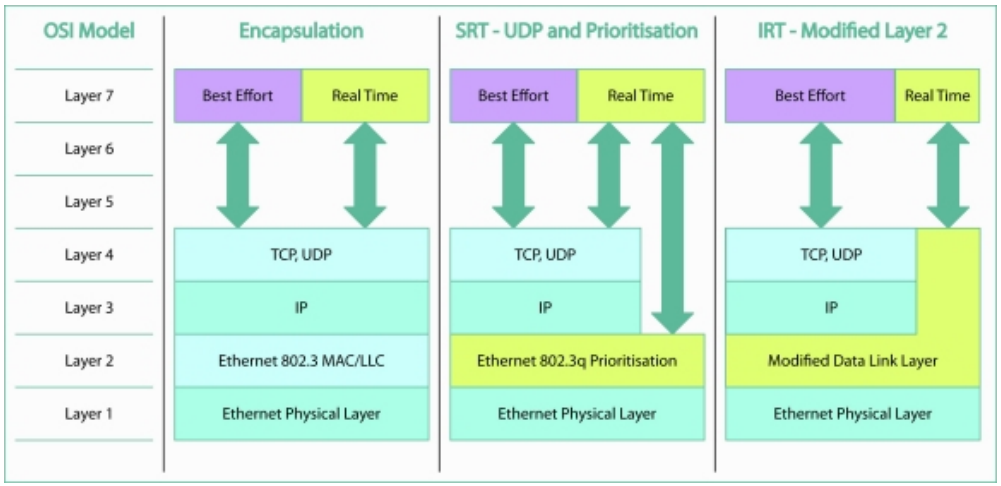
La tradizionale tecnologia Ethernet non rispetta questi requisiti. Per raggiungere le prestazioni richieste in applicazioni industriali è necessario :

- Fast Ethernet
- Trasmissione full-duplex
- Priorità dei messaggi
- Switch
- Sincronizzazione dei Clock di sistema (IEEE 1588 - Precision Time Protocol)

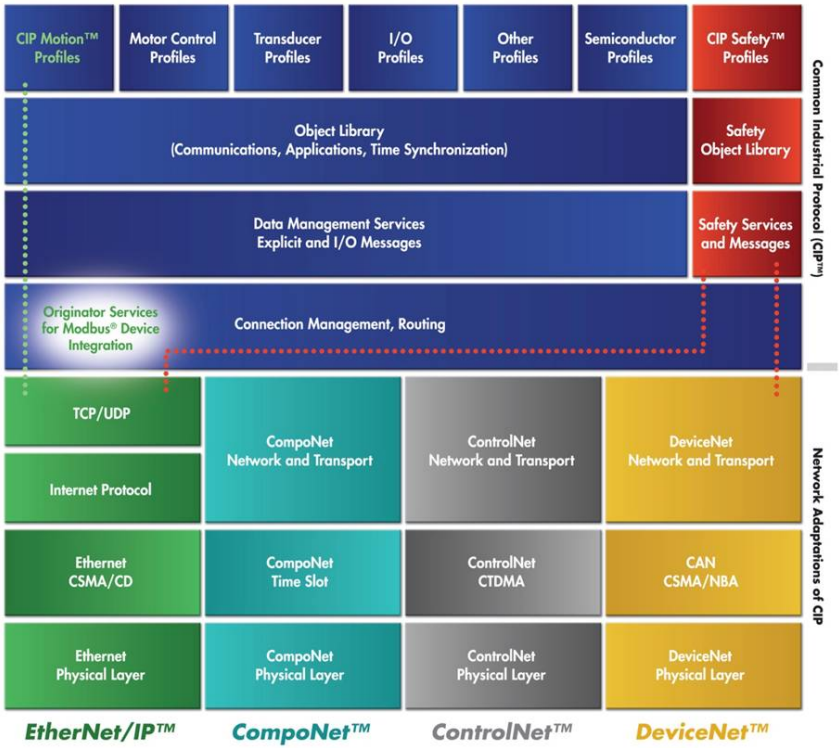
Aspetti implementativi

I diversi ambienti applicativi di una rete RTE hanno portato a differenti approcci implementativi:

- **Classe A - Soluzione basata su TCP/IP**: vengono aggiunte funzionalità a livello applicazione
- **Classe B - Ethernet MAC**: i dati vengono inseriti direttamente a livello di collegamento
- **Classe C - Ethernet modificata**: viene modificata la funzionalità del protocollo Ethernet (risposta < 1 ms)

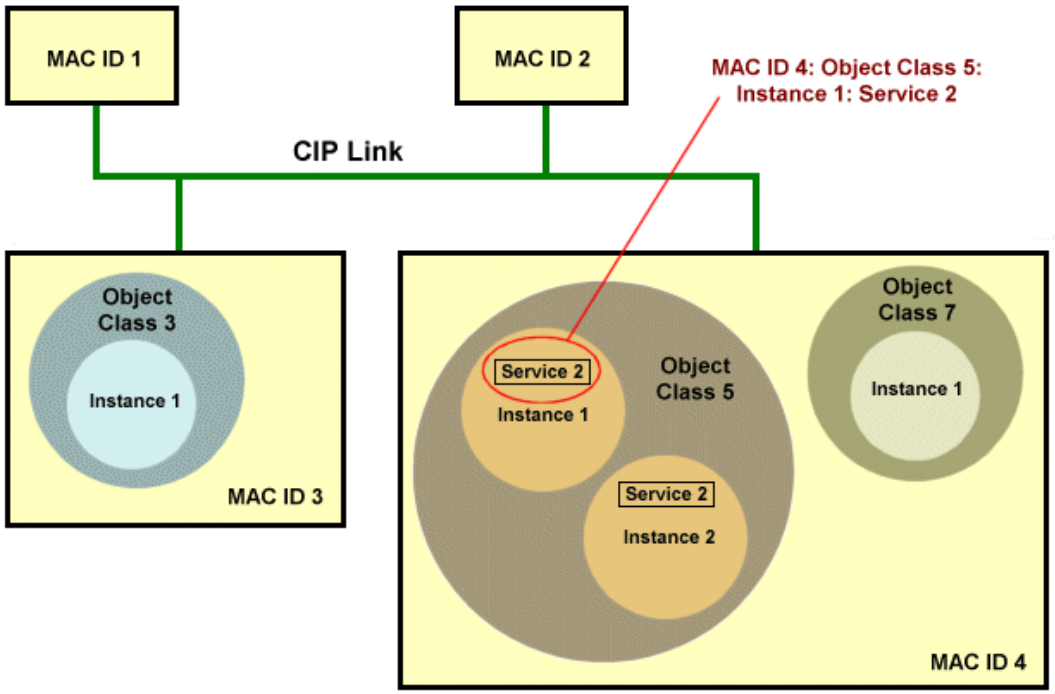


L'idea più semplice per utilizzare Ethernet come bus di campo è quella di incapsulare un protocollo per la gestione della rete nel campo dati di un *pacchetto* a livello di trasporto.
EtherNet/IP è un protocollo che utilizza a livello applicazione il *Common Industrial Protocol* (CIP).



Orientamento ad oggetti

Ogni nodo della rete CIP viene definito come un insieme di oggetti, ovvero delle rappresentazioni astratte di un particolare componente o funzionalità del dispositivo. Gli oggetti sono strutturati in: *classe*, *istanza*, *attributi* e *servizi*.

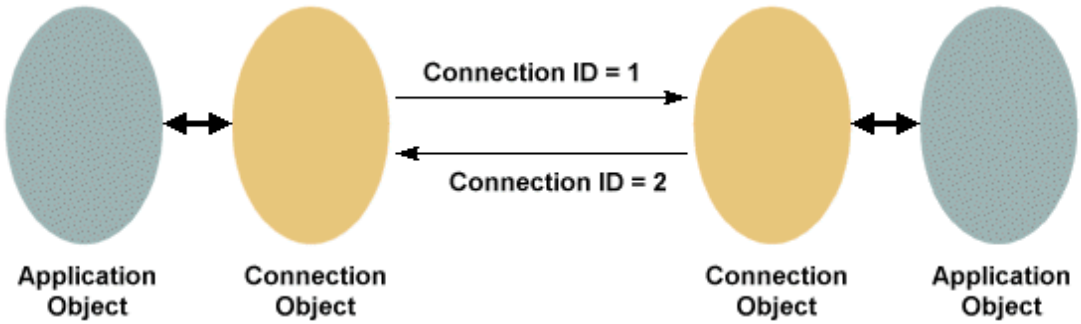


Scambio dei messaggi

La comunicazione è di tipo *produttore-consumatore:*

- Connessioni I/O: comunicazione tra un nodo produttore e i consumatori per lo scambio dei dati di processo
- Messaggi espliciti: comunicazione di tipo *request-response* tra due dispositivi

Ad ogni connessione attiva sulla rete viene associato un identificativo *CID* univoco in base al quale i nodi destinatari verificano la necessità di processare un messaggio.

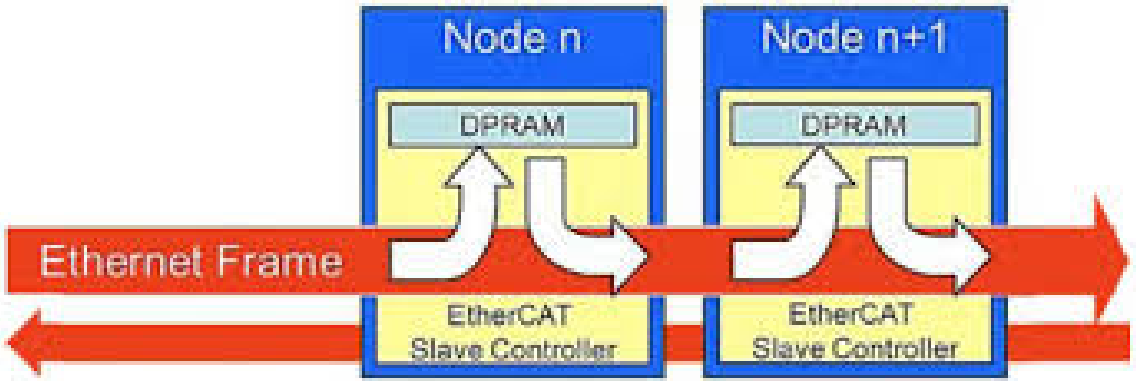


EtherCAT (Ethernet for Control Automation Technology) viene implementato modificando le funzionalità di Ethernet a livello di collegamento. I livelli implementati sono:

- **Livello fisico e di collegamento:** gestiscono l'elaborazione dei frame, il controllo degli errori e l'accesso alla memoria del dispositivo
- **Livello applicazione:** gestisce la macchina a stati del dispositivo e lo scambio dei messaggi ciclici e aciclici

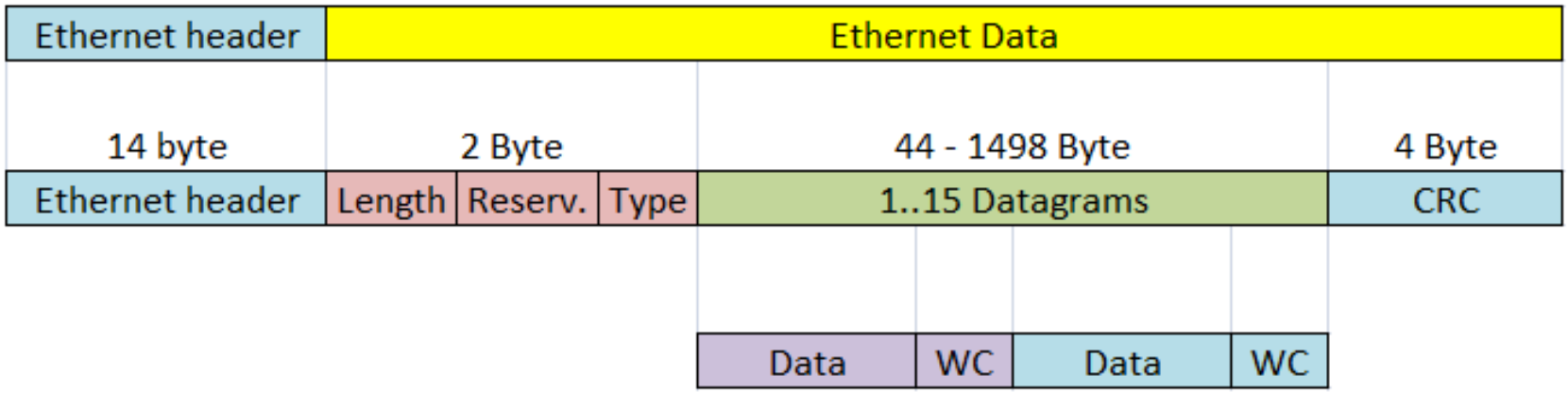
Dati di processo "on the fly"

La particolarità di questo protocollo è quella di non scambiare un singolo frame per ogni nodo.
Un dispositivo *Master* invia un telegramma che attraversa tutti i nodi della rete. Ogni *Slave* legge / scrive i dati di suo interesse al volo presenti nel frame.
Raggiunto l'ultimo nodo della rete, il frame ritorna al master sfruttando la comunicazione full-duplex.



Trasferimento dei dati

Il protocollo è ottimizzato per trasportare i *Dati Di Processo (PDO)* indirizzati agli slave.
Il frame Ethercat è costituito da diversi *Datagrammi* ognuno dei quali serve le aree di memoria dei dispositivi slave interessati.
Il tutto viene inglobato nel payload Ethernet ed identificato da un particolare EtherType 0x88A4.



Obiettivo

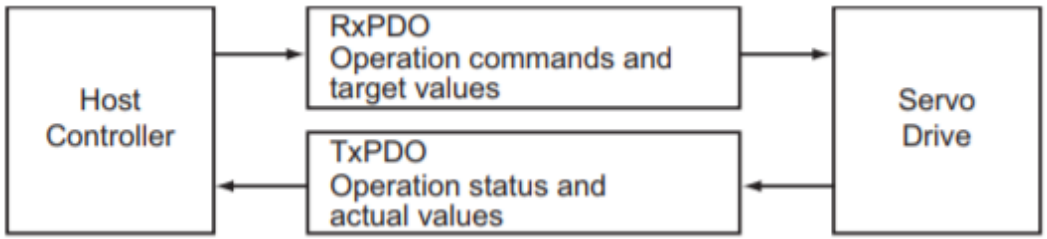
Catturare, tramite l'utilizzo di Wireshark, i dati che viaggiano sulla rete allo scopo di:

1. **Analizzare** la composizione dei pacchetti scambiati
2. **Identificare** il motivo di eventuali condizioni anomale

Contesto applicativo

Analizzare il telegramma trasmesso da un dispositivo *master* verso uno *slave*:

- Il *master* invia un telegramma che contiene i comandi per un posizionamento
- Lo *slave* estrae i dati necessari, traduce i segnali digitali in segnali elettrici per comandare l'attuatore ed inserisce i dati destinati al *master*
- Il telegramma ritorna al *master*



Comando

```
Frame 1: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface 0
Ethernet II, Src: OmronTat_9a:ba:3b (00:00:0a:9a:ba:3b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
EtherCAT frame header
EtherCAT datagram(s): 3 Cmds, 'LRW': len 26, 'LRD': len 1, 'ARMW': len 8
  EtherCAT datagram: Cmd: 'LRW' (12), Len: 26, Addr 0x0, Cnt 0
    Header
    Data: 0f00fcb1e62600000000f4ff08000000000014b80bb80b00...
    Working Cnt: 0
  EtherCAT datagram: Cmd: 'LRD' (10), Len: 1, Addr 0x80000, Cnt 0
    Header
    Data: 00
    Working Cnt: 0
  EtherCAT datagram: Cmd: 'ARMW' (13), Len: 8, Adp 0x0, Ado 0x910, Cnt 0
    Header
    DC SysTime (0x910): 0x0000000000000000
    DC SysTime L (0x910): 0x00000000
    DC SysTime H (0x914): 0x00000000
    Working Cnt: 0
```

```
0000  ff ff ff ff ff ff 00 00  0a 9a ba 3b 88 a4 47 10   ........ ...;hu.·
0010  0c 00 00 00 00 00 1a 80  00 00 0f 00 fc b1 e6 26   ........ .......W&
0020  00 00 00 00 f4 ff 08 00  00 00 00 00 14 b8 0b b8   ····4··· ·····.··
0030  0b 00 00 00 00 00 0a 00  00 00 08 00 01 80 00 00   ····.... .....··
0040  00 00 00 0d 00 00 00 10  09 08 00 00 00 00 00 00   ........ ........
0050  00 00 00 00 00 00 00                               .......
```

Stato

```
Frame 2: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface 0
Ethernet II, Src: 02:00:0a:9a:ba:3b (02:00:0a:9a:ba:3b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
EtherCAT frame header
EtherCAT datagram(s): 3 Cmds, 'LRW': len 26, 'LRD': len 1, 'ARMW': len 8
  EtherCAT datagram: Cmd: 'LRW' (12), Len: 26, Addr 0x0, Cnt 3
    Header
    Data: 000037123da6e626f4ff0800000000000000000000000000...
    Working Cnt: 3
  EtherCAT datagram: Cmd: 'LRD' (10), Len: 1, Addr 0x80000, Cnt 1
    Header
    Data: 00
    Working Cnt: 1
  EtherCAT datagram: Cmd: 'ARMW' (13), Len: 8, Adp 0x1, Ado 0x910, Cnt 1
    Header
    DC SysTime (0x910): 0x000007b48342be42
    DC SysTime L (0x910): 0x8342be42
    DC SysTime H (0x914): 0x000007b4
    Working Cnt: 1
```

```
0000  ff ff ff ff ff ff 02 00  0a 9a ba 3b 88 a4 47 10   ........ ...;hu.
0010  0c 00 00 00 00 00 1a 80  00 00 00 00 37 12 3d a6   ........ ....7.=
0020  e6 26 f4 ff 08 00 00 00  00 00 00 00 00 00 00 00   .&...... ........
0030  00 00 1c 01 03 00 0a 00  00 00 08 00 01 80 00 00   ........ ........
0040  00 01 00 0d 00 01 00 10  09 08 00 00 00 42 be 42   ........ .....B.
0050  83 b4 07 00 00 01 00                               .......
```

```
▷ Frame 83: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▷ Ethernet II, Src: OmronTat_a8:4e:d7 (00:00:0a:a8:4e:d7), Dst: IPv4mcast_40:02:20 (01:00:5e:40:02:20)
▷ Internet Protocol Version 4, Src: 192.168.250.10, Dst: 239.192.2.32
▷ User Datagram Protocol, Src Port: 2222, Dst Port: 2222
◢ EtherNet/IP (Industrial Protocol)
  ◢ Item Count: 2
    ◢ Type ID: Sequenced Address Item (0x8002)
        Length: 8
        Connection ID: 0x5f8e0034
        Encapsulation Sequence Number: 35238
    ◢ Type ID: Connected Data Item (0x00b1)
        Length: 6
        Data: 7003f4046419
```

```
0000  01 00 5e 40 02 20 00 00  0a a8 4e d7 08 00 45 bc   ··; · ·· ·y+P··..
0010  00 34 a0 af 00 00 01 11  6b ba c0 a8 fa 0a ef c0   ·4·····  ,.{y..{
0020  02 20 08 ae 08 ae 00 20  4d 32 02 00 02 80 08 00   · ·.·.·  (2·····
0030  34 00 8e 5f a6 89 00 00  b1 00 06 00 70 03 f4 04   4·.^wi··  .···.·4·
0040  64 19                                             ..
```

Conclusioni e Sviluppi Futuri

Conclusioni

In questo lavoro sono stati analizzati i requisiti richiesti da una rete di comunicazione real-time basata su Ethernet, analizzando le possibili implementazioni e la composizione dei dati scambiati.

Sviluppi Futuri

- Estendere l'analisi agli ulteriori protocolli sviluppati (PROFINET, Ethernet Powerlink, SERCOS III, ...)
- Analizzare le prestazioni tra differenti protocolli implementati sulla stessa applicazione (ad esempio: posizionamento di un braccio robotizzato)

Grazie per l'attenzione.