



ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Napoli

INFN/TC-07/07

14 Giugno 2007

LA SCHEDA E-TIBBO

V. Masone, P. Parascandolo

INFN – Sezione di Napoli, Via Cintia, I-80126 Napoli, Italy

Abstract

Questa nota descrive una possibile applicazione del modulo EM202 della Tibbo Technology come analizzatore di rete. Tramite la scheda E-Tibbo, che alloggia in un crate VME, è possibile spedire e ricevere dati sulla rete Ethernet 100 BaseT.

*Published by **SIS-Pubblicazioni**
Laboratori Nazionali di Frascati*

INTRODUZIONE

Un'architettura Internet [1] consiste di un insieme di reti interconnesse [Fig. 1] per gran parte in fibra ottica che possono essere considerate come entità unica.



Fig. 1 - Reti e Routers.

I calcolatori (Host) che si collegano alla rete devono eseguire dei protocolli specifici per controllare l'invio e la ricezione di informazioni. Un protocollo definisce il formato e l'ordine dei messaggi scambiati tra due dispositivi che comunicano tra loro

Ciascuna rete ha un solo canale trasmissivo ad alta velocità condiviso nel tempo da tutti i sistemi collegati. Un singolo sistema cioè, diventa proprietario temporaneamente (per la durata di uno o più pacchetti) dell'intera capacità trasmissiva della rete. La trasmissione è sempre di tipo broadcast: un sistema invia dati e tutti gli altri ricevono. Tale organizzazione implica la presenza di indirizzi per stabilire chi è il destinatario e chi il mittente del trasferimento. Inoltre, occorre arbitrare l'accesso all'unico mezzo trasmissivo tra tutti i sistemi che hanno necessità di trasmettere.

L'Host sorgente che deve inviare i dati sulla rete suddivide il messaggio in piccoli blocchi indicati con il nome di pacchetti. Questi pacchetti viaggiano lungo il Link di comunicazione e attraverso i Routers alla velocità massima consentita. I Routers utilizzano un meccanismo "store and forward" attraverso cui immagazzinano l'intero pacchetto prima di procedere alla ritrasmissione. Poiché in genere un Router è connesso a molti canali, è sempre presente un buffer per ciascun Link di connessione.

In una rete a commutazione di pacchetto quindi, il protocollo deve garantire che il destinatario possa riassemblare i pacchetti fino ad ottenere il messaggio originale.

La frammentazione del messaggio in pacchetti riduce di un fattore tre i ritardi di propagazione rispetto alla trasmissione del messaggio intero e risulta inoltre più sicura.

I Routers stabiliscono l'instradamento dei diversi pacchetti basandosi esclusivamente sull'indirizzo di destinazione degli stessi.

La pila di protocolli Internet (Fig. 2) è costituita da 5 strati: Fisico (Physical Layer), Collegamento (Data Link Layer), Rete (Network Layer), Trasporto (Transport Layer) ed Applicazione (Application Layer) [2].

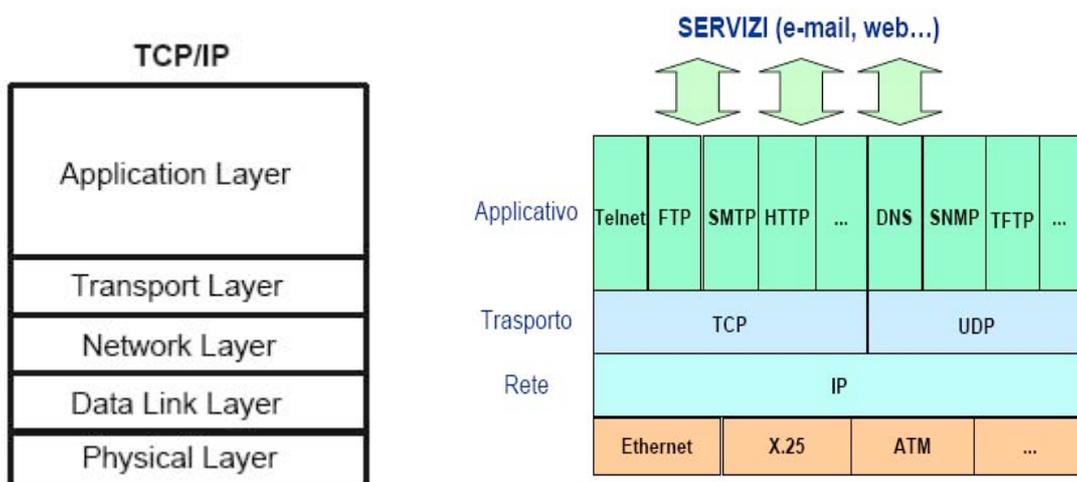


Fig. 2 - La pila protocollare.

Per quattro di questi cinque strati vengono comunemente usati i seguenti acronimi: Frame per il Data Link Layer; Datagram per il Network Layer; Segment per il Transport Layer e Messaggio per l'Application Layer.

Le relazioni tra i livelli protocollari si possono schematizzare come segue:

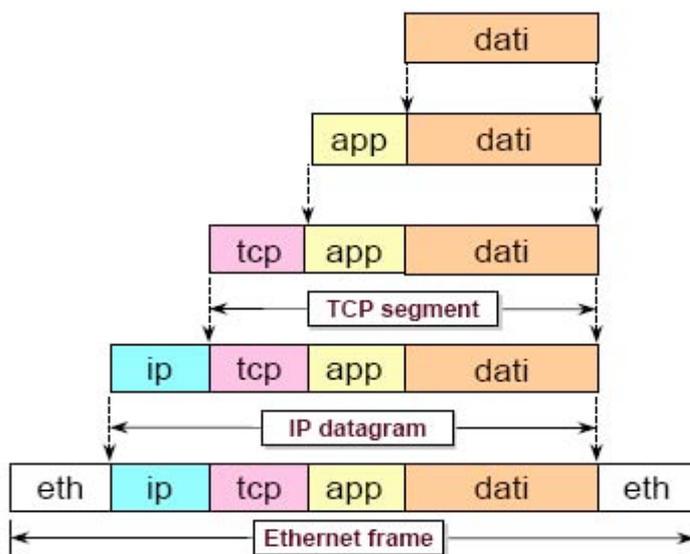


Fig. 3 - Relazioni tra i livelli protocollari.

Come si vede dalla figura precedente ciascun strato introduce un Header. Al livello sottostante l'Header superiore viene trattato come un dato. In ricezione invece, avviene un processo inverso in cui ciascun strato controlla e sopprime il corrispondente Header.

Sulla rete un frame Ethernet assume il formato indicato in figura:

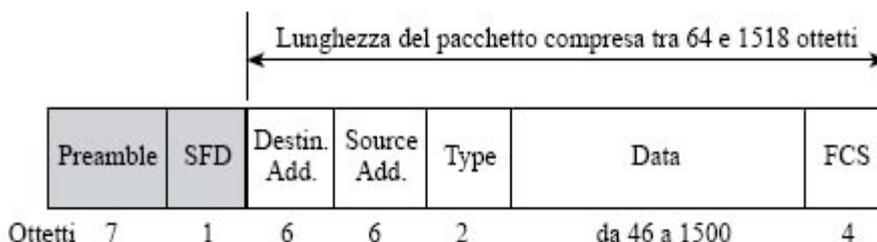


Fig. 4 - Formato del frame Ethernet.

In testa al pacchetto troviamo un preambolo di 7 ottetti (ciascuno costituito da un'alternanza di uni e zeri) che serve alla stazione ricevente per sincronizzarsi sul clock di quella trasmittente. Immediatamente dopo c'è un ottetto di SFD (Start Frame Delimiter, caratterizzato dagli ultimi due bit ad uno logico) che indica l'inizio del pacchetto.

Nel campo Source e Destination ci sono i due MAC address. Gli indirizzi MAC sono lunghi 6 byte, si scrivono per convenzione in esadecimale e sono univoci a livello mondiale. Essi sono scritti in una ROM dal costruttore della scheda di rete e possono essere eventualmente sostituiti via software da indirizzi scritti in un apposito buffer. I MAC si compongono di due parti di 3 byte ciascuna: i 3 byte più significativi, detti OUI (Organization Unique Identifier), indicano il lotto di indirizzi assegnato al costruttore della scheda di rete locale o all'organizzazione che ha progettato una data architettura di rete; i 3 byte meno significativi costituiscono una numerazione interna progressiva decisa dal costruttore stesso.

Nel campo Type è contenuto il codice associato al protocollo di livello superiore che ha prodotto i Data. Il campo FCS (Frame Check Sequence) contiene il valore di CRC calcolato sulla base dei campi menzionati in precedenza. Non esiste un segnalatore di fine pacchetto.

Il livello rete tiene traccia dell'Host mittente, dell'Host destinatario e del protocollo di trasporto (TCP o UDP) coinvolto nella comunicazione; il livello di trasporto tiene traccia dell'applicazione mittente e dell'applicazione destinataria.

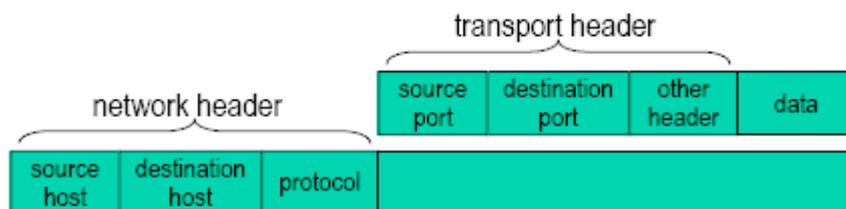


Fig. 5 - Formato a livello rete.

A livello datagram IP il formato dell'Header è il seguente:

0	8	16	24	31
version	header length	type of service	datagram length	
Identification (ID)		flags	fragment offset	
time to live	next protocol	header checksum		
source IP address				
destination IP address				
options, se presenti (lunghezza variabile)				(riempimento)

Fig. 6 - Header a livello Datagram.

Le funzioni principali del protocollo IP in trasmissione sono: l'incapsulamento in datagram dei dati provenienti dal livello di trasporto; l'applicazione dell'intestazione; l'applicazione dell'algoritmo di routine per inviare i dati verso l'interfaccia di rete. Le funzioni principali del protocollo IP in ricezione sono: la verifica della validità dei datagram in arrivo; l'esame dell'intestazione; la formattazione dei dati per lo strato di trasporto.

Il datagram IP ha una lunghezza variabile. La somma degli ottetti Dati + Header deve essere inferiore alla MTU (Maximum Transmission Unit), altrimenti si ha la frammentazione. Nel campo Protocol è inserito il protocollo di trasporto. Il Fragment Offset indica il frammento nell'ambito del totale dei dati da trasmettere. Identification è l'identificativo del datagram. Flag indica se il datagram rappresenta o meno un frammento. Time to live (TTL) indica il numero di routers che un datagram può attraversare. Header Checksum è il controllo dell'errore sull'Header. IHL è la lunghezza dell'Header e il Total Length è la lunghezza totale.

1. IL LAYER FISICO

La trasmissione e la ricezione in Ethernet su 100 BaseTX o 10 BaseT devono rispettare lo standard IEEE 802.3. Diverse case costruttrici offrono dispositivi che incorporano tutta l'interfaccia analogica con il ricevitore per coppia twistata ed il driver differenziale.

Un circuito integrato tipico per lo svolgimento di queste funzioni è l'80225 della LSI Logic il cui schema a blocchi è rappresentato in figura 7. Al suo interno il dispositivo comprende un equalizzatore, un encoder e un decoder 4B5B, uno scrambler e descambler, un circuito di autonegoziazione, un'interfaccia per il controller MII, una porta seriale ed un'interfaccia di rete per il formato ternario MLT-3.

In ricezione il clock viene ricavato dagli stessi dati ed utilizzato per immagazzinare le informazioni che successivamente vengono convertite nel formato NRZ. I dati sono poi trattati dal decoder 4B5B ed inviati al controllore Ethernet.

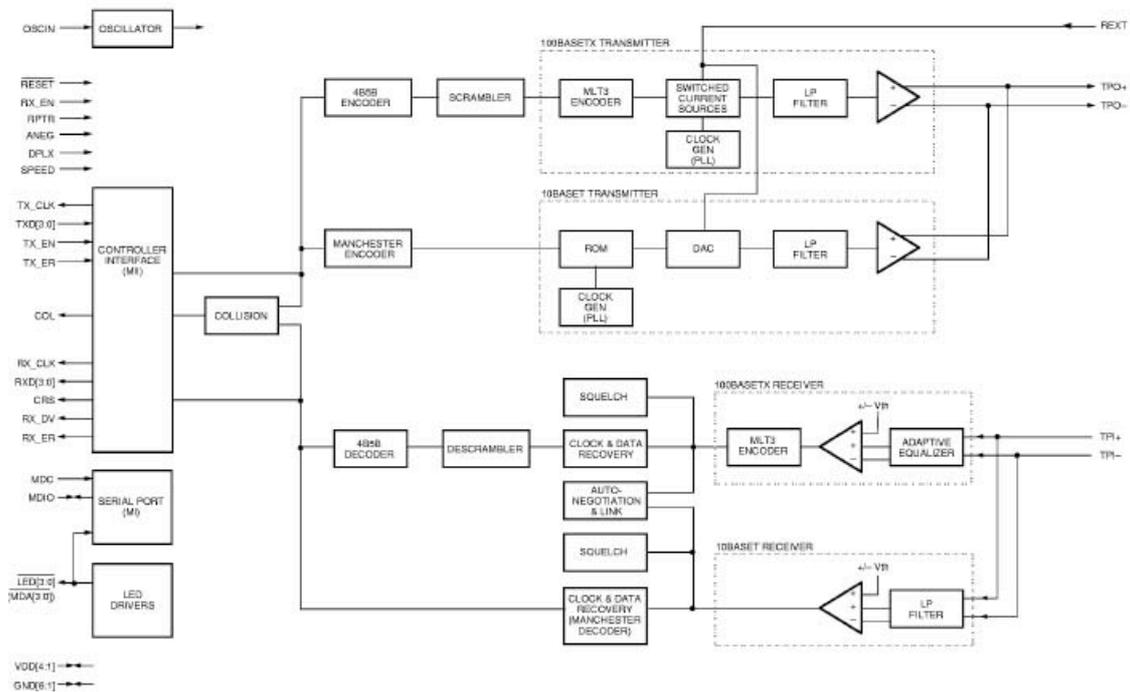


Fig. 7 - Schema a blocchi dell'80225 della LSI Logic.

2. LA SCHEDE E-TIBBO

La scheda E-Tibbo si basa sul modulo EM202 [3] della Tibbo Technology che è a tutti gli effetti un Device Server (DS). Il dispositivo si presenta come un semplice connettore RJ45; al suo interno però, è racchiusa tutta la circuiteria – comprese le parti magnetiche - che lo rende una perfetta interfaccia tra una porta seriale TTL ed una porta Ethernet 100 BaseT. I pin corrispondenti alle porte di comunicazione seriale possono essere interfacciati direttamente ai microprocessori più comuni ed alle FPGA.

Il Tibbo, così come qualsiasi altro dispositivo Ethernet, ha un MAC Address univoco assegnatoli direttamente dalla casa costruttrice. L'utente, invece, si deve preoccupare di trovare un indirizzo IP valido in modo da poterlo interfacciare alla rete Ethernet. Per collegarsi ad un Host bisogna inoltre utilizzare un valido protocollo di trasmissione tra quelli supportati dal dispositivo (UDP, TCP, ARP, ICMP, DHCP, PPPoE ed LCP).

La gran parte della funzionalità dell'EM202 è svolta dal firmware proprietario residente nel dispositivo. Esso fa sì che i dati seriali d'ingresso vengano convertiti nel formato compatibile con il protocollo prescelto. In alternativa è possibile caricare al suo interno un completo sistema operativo proprietario che lo rende completamente programmabile dall'utente.

Il Tibbo possiede al suo interno 8KBx2 di memoria e quattro LED di stato che indicano le diverse modalità di operazione.

Per testare la funzionalità del modulo abbiamo realizzato una scheda doppio Eurocard sulla quale è alloggiata una FPGA della Xilinx (Spartan-II XC2S100-5PQ208) che produce a cadenza prefissata un pattern noto da trasferire serialmente all'EM202 ad una velocità di 115.200 baud e da qui, tramite rete Ethernet, ad un PC. E' naturalmente possibile eseguire il percorso inverso che consiste nel trasmettere dati all'FPGA tramite rete e rileggerli via RS232 attraverso un terminale opportunamente configurato.

La scheda possiede inoltre un'interfaccia seriale realizzata tramite un ADM3311 della Analog Devices che viene alimentato a 3.3Volt e realizza i livelli compatibili con lo standard RS232 garantendo un isolamento di 15KV. Il dispositivo va automaticamente in uno stato di basso assorbimento allorquando non rileva traffico di dati seriali con l'FPGA.

La scheda realizzata è raffigurata nell'immagine seguente:

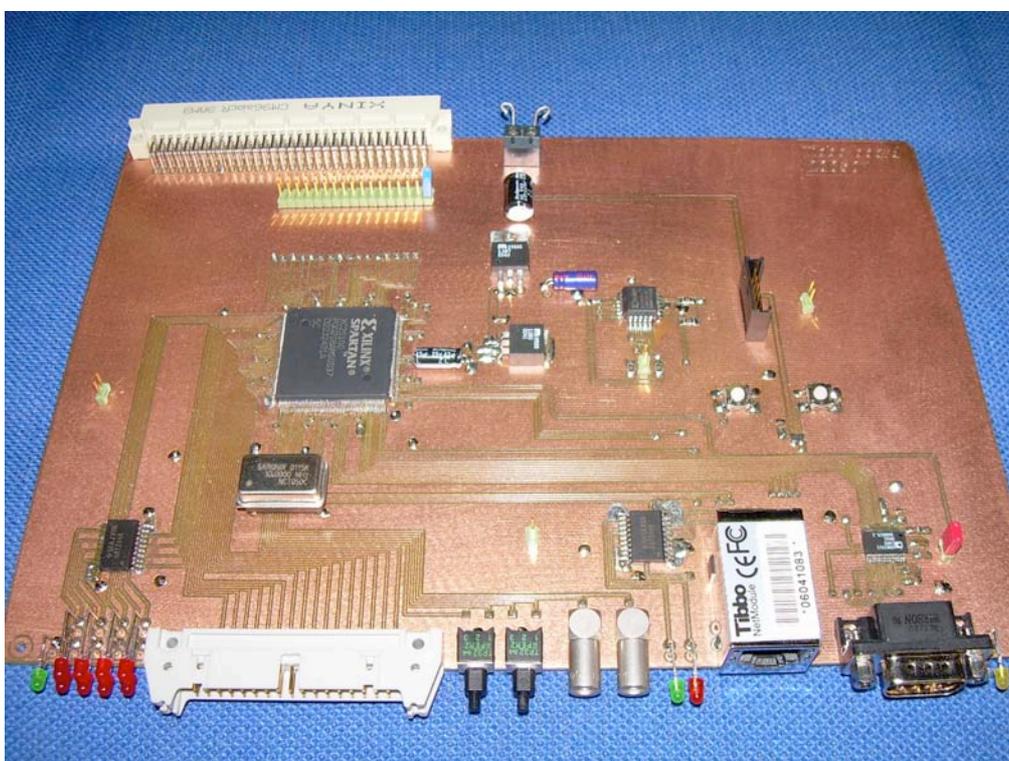


Fig. 8 - La scheda E-Tibbo.

3. IL MINI-ANALIZZATORE DI RETE

Dei due indirizzi (MAC ed IP) che usa ogni computer connesso su una LAN, il MAC è quello che l'Ethernet mette in testa al frame per trasferire i dati tra due dispositivi. Tutti i computer che stanno sulla rete analizzano il MAC address di destinazione per vedere se corrisponde con il loro; in caso negativo il frame viene soppresso. Basandoci su questo principio, abbiamo realizzato un piccolo analizzatore di rete che intercetta il MAC address del Tibbo e, a partire da questo, comincia a registrare il traffico dati [Fig. 9]. L'analizzatore attualmente è costituito da due diverse schede.

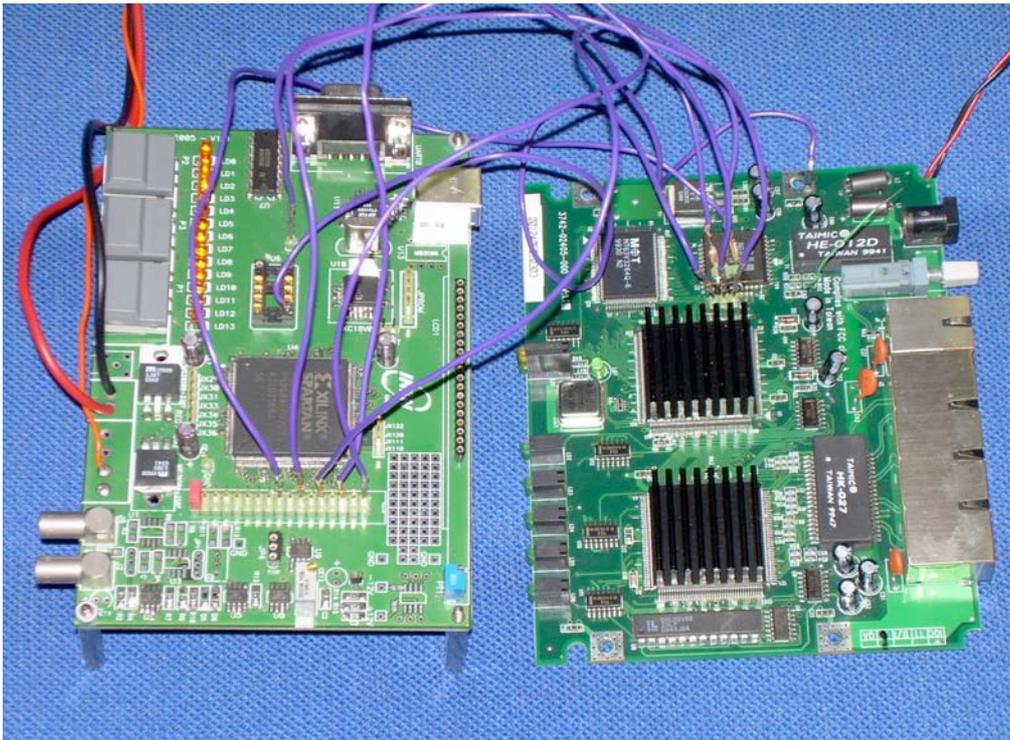


Fig. 9 - Mini analizzatore di rete.

Su una delle schede si trova l'interfaccia Ethernet verso il layer fisico basato sull'80225 della LSI-LOGIC. La seconda scheda è basata su di una FPGA della Xilinx (Spartan-II XC2S100-5PQ208) che riceve i dati dal layer fisico, si aggancia al MAC address del Tibbo e comincia a registrare il traffico funzionando semplicemente da buffer di memoria.

Per stabilire una connessione client/server è necessario il susseguirsi di tre messaggi: il client invia un segmento SYNC contenente il suo numero di sequenza; il server deve rispondere con un segmento SYNC/ACK contenente il suo numero di sequenza e l'acknowledgment; infine il client invia a sua volta al server un segmento che indica l'acknowledgment del numero di sequenza del server.

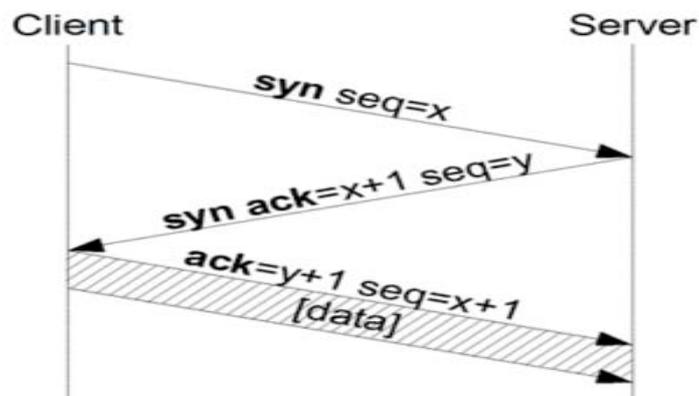


Fig. 10 - Il traffico di rete iniziale.

Quando l'FPGA riconosce il termine del pacchetto aggiunge un byte di fine frame (EE) che riconosciamo come trailer. Una volta terminato il traffico tramite un pulsante è poi possibile leggere il contenuto della memoria interna della FPGA mediante un'interfaccia seriale RS232 realizzata anch'essa all'interno della Xilinx.

Il numero dei pacchetti inviati dipende dalla lunghezza del file da trasferire. Nella seguente tabella riportiamo il traffico che abbiamo registrato inviando un file costituito da 100 Byte tutti uguali tra loro (CB).

Primo pacchetto

```
00000: 00 0C 6E D9 63 18 00 CA 00 00 07 62 08 00 45 00
00010: 00 78 00 22 40 00 FF 06 FA 05 C0 A8 00 05 C0 A8
00020: 00 02 03 E9 06 38 04 E9 D0 CA 66 DB B1 EC 50 18
00030: 04 00 59 B0 00 00 CB CB CB CB CB CB CB CB CB CB
00040: CB CB
00050: CB CB
00060: CB CB
00070: CB CB
00080: CB CB CB CB CB CB CB 51 5F CB EE
```

Secondo pacchetto

```
00090: 00 0C 6E D9 63 18 00 CA 00 00 07 62 08 00 45 00
000A0: 00 78 00 22 40 00 FF 06 FA 05 C0 A8 00 05 C0 A8
000B0: 00 02 03 E9 06 38 04 E9 D0 CA 66 DB B1 EC 50 18
000C0: 04 00 59 B0 00 00 CB CB CB CB CB CB CB CB CB CB
000D0: CB 51 5F CB EE
```

Terzo pacchetto

```
000E0: 00 0C 6E D9 63 18 00 CA 00 00 07 62 08 00 45 00
000F0: 00 28 00 24 40 00 FF 06 FA 53 C0 A8 00 05 C0 A8
00100: 80 02 03 E9 86 38 04 E9 D1 2E 66 DB B1 EC 50 10
00110: 04 00 31 7C 00 00 CB CB CB CB CB CB CB AA 8D 79 EE
```

Dall'esame dei pacchetti si individuano facilmente:

MAC PC → 00 0C 6E D9 63 18	MAC TIBBO → 00 CA 00 00 07 62
IP TIBBO → C0 A8 00 05	IP PC → C0 A8 00 02
Versione (IPV4) → 4	Header Length → 5
Service Type → 00	Total Length → 78
Identificativo → 00 22	Time To Live → FF
Protocollo TCP → 06	Checksum → FA 05
Source Port → 03 E9	Destination Port → 06 38
Numero di Sequenza → 04 E9 D0 CA	Nr. Acknowledgment → 66 DB B1 EC

4. CONCLUSIONI

L'utilizzo del Tibbo consente di modificare via Ethernet le impostazioni dei moduli di acquisizione dati negli esperimenti ed il trasferimento dei dati stessi via rete.

BIBLIOGRAFIA

- [1] J.F. Kurose, K.W. Ross, "Internet e Reti di Calcolatori", 2^a edizione, McGrawHill Italia, 2003;
- [2] R. Braden, J. Postel and Y. Rekhter, "RFC 1620: Internet Architecture Extension for Shared Media", May 1994;
- [3] <http://www.tibbo.com/em202.php>