

Modello USB

➤ Livello fisico

- Il *serial interface engine* (SIE) codifica i dati, rileva gli errori e rileva il *packet identifier* (PID).
- L'host controller gestisce il polling delle periferiche

➤ Protocol engine layer

- Si occupa della traduzione dei dati provenienti dallo strato superiore in formato USB

Modello USB

➤ Livello di applicazione

Ogni comunicazione è composta da 3 fasi:

- ❑ Token phase

L'host invia un pacchetto per indicare il tipo di comunicazione

- ❑ Data phase

I dati vengono inviati raggruppati in pacchetti

- ❑ Handshake phase (opzionale)

Viene inviato un pacchetto per segnalare il successo o il fallimento della comunicazione

Campi del pacchetto

☐ Sync field

☐ Packet identifier field

I. Token

- a. Out token
- b. In token
- c. Start of frame token
- d. Setup token

Usato per la configurazione del dispositivo

II. Dati

Descrive il tipo di dati come ad esempio la velocità con cui vengono inviati

Campi del pacchetto

III. Handshake

- a. Acknowledge
- b. Not acknowledge
- c. Stall

PID_0	PID_1	PID_2	PID_3	$\overline{PID_0}$	$\overline{PID_1}$	$\overline{PID_2}$	$\overline{PID_3}$
---------	---------	---------	---------	--------------------	--------------------	--------------------	--------------------

Campi del pacchetto

☐ Address field

I. ADDR

$ADDR_0$	$ADDR_1$	$ADDR_2$	$ADDR_3$	$ADDR_4$	$ADDR_5$	$ADDR_6$
----------	----------	----------	----------	----------	----------	----------

II. ENDP

$ENDP_0$	$ENDP_1$	$ENDP_2$	$ENDP_3$
----------	----------	----------	----------

Campi del pacchetto

- ❑ **Number field**

Contiene il numero d'ordine del frame

- ❑ **Data field**

Può contenere fino a 1023 byte di payload

- ❑ ***Cyclical redundancy check* (CRC) field**

Formato dei pacchetti

☐ Token

PID	ADDR	ENDP	CRC
-----	------	------	-----

☐ Start of frame (SOF)

PID	Frame number	CRC
-----	--------------	-----

☐ Data

PID	Data	CRC
-----	------	-----

Formato dei pacchetti

☐ Handshake

PID	Frame number	CRC
-----	--------------	-----

Protocollo di comunicazione

- Le comunicazioni USB sono basate sul *polling* dei dispositivi connessi al root-hub. Tutti i trasferimenti di dati vengono iniziati dall'host-controller che periodicamente interroga i dispositivi connessi alla tiered star.
- La maggior parte delle transazioni sul bus prevede l'invio di un certo numero di frame (al massimo 3 consecutivi). Una transazione ha inizio quando l'host controller, sulla base di un criterio di scheduling, invia un frame USB che descrive il tipo ed il verso della transazione, l'indirizzo del device USB ed il numero dell'endpoint. Tale frame viene chiamato token packet. Il dispositivo USB che è stato indirizzato dall'host-controller si riconosce come destinatario di quella comunicazione interpretando il campo indirizzo.

Protocollo di comunicazione

- In una transazione i dati vengono trasferiti o dall'host verso il device o dal device verso l'host; il verso del trasferimento viene specificato nel token packet. In altre parole la comunicazione è di tipo half-duplex.
- Per comunicare con un device USB, il software sull'host apre una serie di pipe logiche che vengono impiegate per trasportare dati e segnali di controllo.
- Le pipe corrispondono ad endpoint di tipo hardware, dietro ai quali ci sono canali individuali, tipicamente con buffer dedicati.

Data transfer

- Un trasferimento dati è costituito da un certo numero di frame che vengono scambiati tra host e device.
- Per comunicare con un device USB, il software sull'host apre una serie di pipe logiche che vengono impiegate per trasportare dati e segnali di controllo.
- Le pipe corrispondono ad endpoint di tipo hardware, dietro ai quali ci sono canali individuali, tipicamente con buffer dedicati.

Enumeration di un dispositivo USB

- Si tratta del processo con cui il sistema operativo riconosce un nuovo dispositivo connesso al bus USB e determina i suoi requisiti. Dopodichè il sistema operativo carica il driver più appropriato ed attribuisce al dispositivo un indirizzo (temporaneo) univoco.
- Si articola nei seguenti step:
 1. Il device viene connesso alla porta dell'hub.
 2. L'hub notifica la modifica nello stato della porta(vi è un “nuovo” assorbimento di corrente) quando viene interrogato dall'host controller durante la fase di “status update”.
 3. L'host richiede informazioni su tale variazione e invia un “port enable” ed un “reset” per quella porta.

Enumeration di un dispositivo USB

4. Il device è adesso nello stato “powered” e risponde al default address (si ricordi l’endpoint 0).
5. L’host
 - a. Recupera il device descriptor e determina la massima dimensione dei pacchetti per quell’endpoint .
 - b. Sulla base del vendor identifier e del product identifier del device, decide quale driver caricare.
 - c. Recupera il configuration descriptor e assegna un indirizzo univoco al dispositivo USB e, sulla base della potenza e della banda disponibile, assegna una configurazione al device.
6. Il dispositivo è così indirizzato, configurato e pronto all’uso .

Riferimenti bibliografici

- www.usb.org
- www.bitportal.it/tutorial