

Corso di Sistemi Automatici di Misura

SETTIMA LEZIONE:

**TRASMISSIONE DI DATI E COMANDI
NEL SISTEMA IEEE-488**

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

Lo **scambio di dati** tra le periferiche o l'invio di **comandi indirizzati** da parte del *controller*, avvengono attraverso una procedura fissata dallo standard IEEE-488, che prevede il succedersi di alcune fasi caratteristiche

A seconda che la trasmissione avvenga tra periferiche *talker* e *listener* o tra *controller* e *listener*, il protocollo di trasmissione prevede fasi diverse

CASO 1: TRASMISSIONE DI COMANDI INDIRIZZATI AD OPERA DEL *CONTROLLER*

CASO 2: TRASFERIMENTO DI DATI FORNITI DA UN *TALKER*

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 1: TRASMISSIONE DI COMANDI INDIRIZZATI AD OPERA DEL *CONTROLLER*

Fase 1: selezione delle interfacce che dovranno ricevere la comunicazione (indirizzamento per funzionamento *listener*)

In modalità $ATN = TRUE$ (mod. *command*) il controller invia un comando di deselegazione UNL che disattiva le periferiche che erano state indirizzate come *listener* in una operazione precedente

In modalità $ATN = TRUE$ (mod. *command*) il controller provvede a selezionare ed attivare le periferiche che prenderanno parte alla successiva operazione, trasmettendo i corrispondenti *listen address*

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 1: TRASMISSIONE DI COMANDI INDIRIZZATI AD OPERA DEL *CONTROLLER*

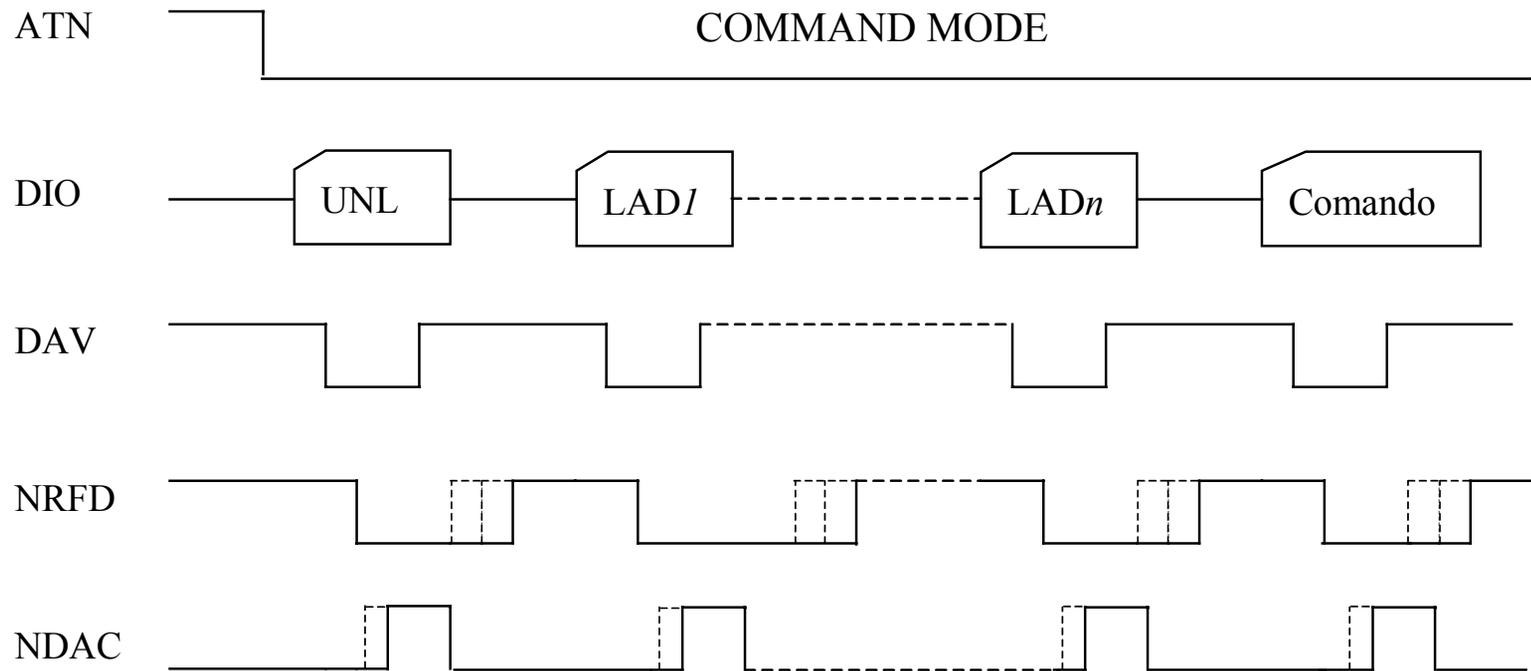
Fase 2: invio del comando multilinea indirizzato ai *listener* selezionati

In modalità $ATN = TRUE$ (mod. *command*) il controller pone il comando sulle linee DIO1-8 e successivamente pone fine alla trasmissione

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 1: TRASMISSIONE DI COMANDI INDIRIZZATI AD OPERA DEL *CONTROLLER*



Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 2: TRASMISSIONE DI DATI FORNITI DA UN TALKER

Fase 1: selezione delle interfacce che dovranno ricevere la comunicazione (indirizzamento per funzionamento *listener*) e di quella che sarà sorgente della comunicazione (indirizzata come *talker*)

In modalità ATN = TRUE (mod. *command*) il controller invia un comando di deselegazione UNL che disattiva le periferiche che erano state indirizzate come *listener* in una operazione precedente

In modalità ATN = TRUE (mod. *command*) il controller provvede a selezionare ed attivare le periferiche che prenderanno parte alla successiva operazione, trasmettendo i corrispondenti *listen address*

In modalità ATN = TRUE (mod. *command*) il controller provvede a indirizzare la periferica che sarà sorgente di dati, trasmettendo il corrispondente *talken address*

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 2: TRASMISSIONE DI DATI FORNITI DA UN TALKER

Fase 2: il controller cede la trasmissione alla interfaccia indirizzata come *talker* nella fase 1

Questa fase ha inizio quando il controller impone $ATN = FALSE$ (mod. *data*)

Il *talker* porta sulle linee del data-bus il primo carattere da trasmettere ed inizia ad operare come sorgente di handshake

Le sole periferiche indirizzate come *listener* (fase 1) partecipano alla temporizzazione della comunicazione operando come ricevitori di handshake

La trasmissione dei caratteri costituenti il messaggio ha luogo in maniera analoga al caso precedente fino a quando il *talker* ha trasmesso l'ultimo carattere

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 2: TRASMISSIONE DI DATI FORNITI DA UN TALKER

Fase 3: una volta terminata la trasmissione di dati da parte del *talker*, questo deve segnalare la terminazione dell'operazione di trasmissione per consentire al controller di riprendere la propria attività

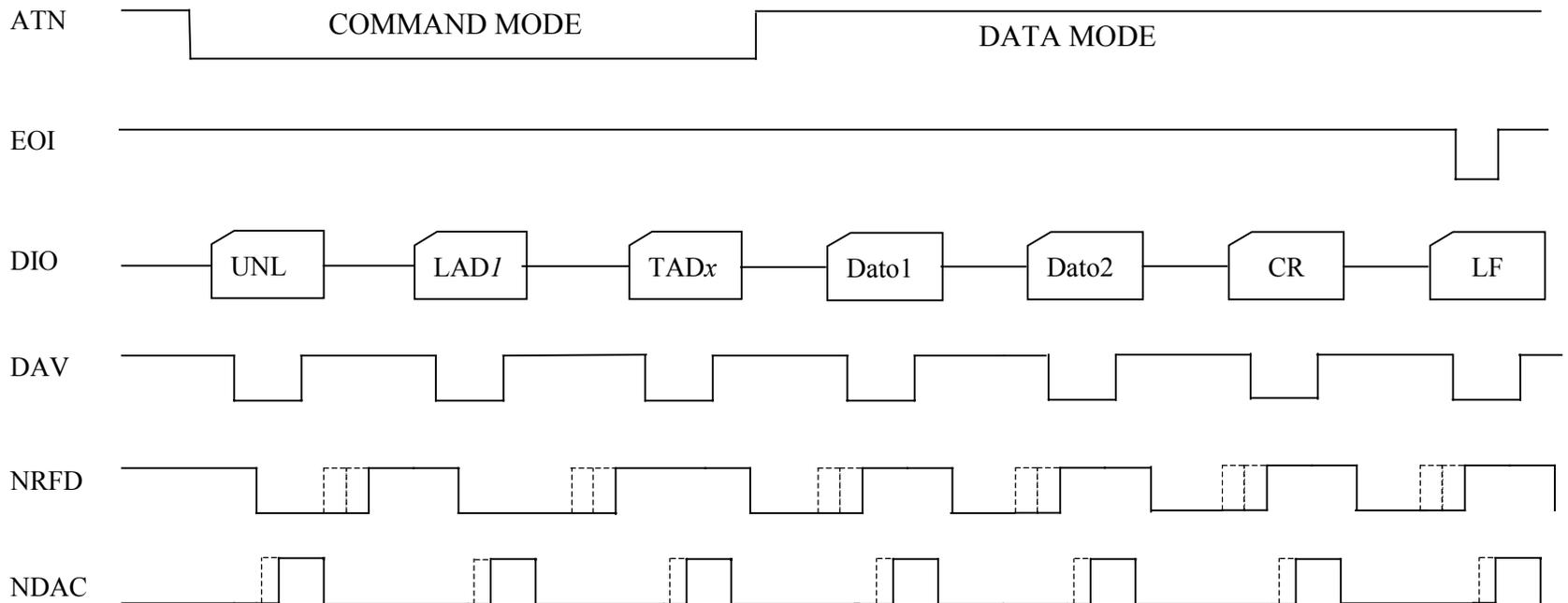
La segnalazione di fine trasmissione può essere fatta in due modi:

- 1) Il *talker* può fare uso della linea EOI: non è necessario che il controller si sia indirizzato come *listener* e pertanto non partecipa all'handshake, potendosi dedicare ad altre attività. Esso riprenderà il controllo del sistema in conseguenza della segnalazione inviata dal *talker* tramite l'EOI
- 2) Il *talker* può segnalare la fine trasmissione aggiungendo in coda al messaggio trasmesso una sequenza di caratteri prestabiliti che fungano da terminatore (solitamente LF+CR). In tal modo, per accorgersi della fine della trasmissione, il controller deve essersi indirizzato come *listener*, partecipando all'handshake

Trasmissione di dati e comandi

PROTOCOLLO DI TRASMISSIONE

CASO 2: TRASMISSIONE DI DATI FORNITI DA UN TALKER



Trasmissione di dati e comandi

STRUTTURA DEI MESSAGGI

I messaggi che vengono trasmessi tramite il bus hanno una struttura che è solo parzialmente definita dallo standard IEEE-488

Interface Message: Comandi per le interfacce, codificati dallo standard

Device Message: Comandi e dati scambiati fra le periferiche, dipendenti dai progettisti delle periferiche. L'IEEE-488 si limita a consigliare l'uso dei caratteri ASCII per la trasmissione di dati alfanumerici, e nel caso di trasmissione dati in forma diversa (ad es. binaria a 8 bit), a prescrivere che il bit più significativo venga trasmesso mediante la linea DIO8

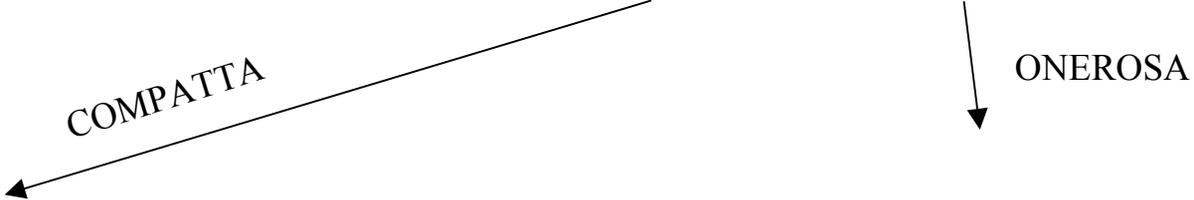
Trasmissione di dati e comandi

STRUTTURA DEI MESSAGGI

Risultati di misura

La comunicazione dei risultati di misura può avvenire in due modi, a seconda del tipo di codifica che può essere *binaria* o *decimale*

COMPATTA



Il risultato di misura viene discretizzato su un certo numero di livelli dipendenti dal numero di bit utilizzato per la codifica

Ad ogni carattere trasmesso (ad es. 8 bit) è associato un livello di tensione e quindi un risultato di misura (oscillosopi)

ONEROSA

Il risultato di misura viene inviato attraverso una sequenza di caratteri ASCII che compone un numero decimale

Ad una sequenza di caratteri trasmessi (ad es. 1.356V) è associato un risultato di misura (multimetri, contatori, ecc.)

Trasmissione di dati e comandi

STRUTTURA DEI MESSAGGI

Risultati di misura

Quando la trasmissione avviene secondo la codifica decimale, il *talker* può segnalare al controller la fine della propria attività facendo uso, indifferentemente, della commutazione della linea *EOI* oppure della *sequenza dei caratteri terminatori prestabiliti*

Se la trasmissione avviene secondo la codifica binaria, la terminazione della trasmissione da parte del *talker* non può avvenire tramite caratteri terminatori. In tal caso infatti i risultati di misura sono codificati come caratteri ASCII (livelli) e quindi ad una sequenza di caratteri (ad es. CR LF) non è associata la terminazione del messaggio ma due livelli di tensione (risultati di misura)

Richiesta di servizio e procedure di interrogazione

RICHIESTA DI SERVIZIO

Le interfacce di alcune periferiche hanno la capacità di effettuare una richiesta di servizio attraverso la linea SRQ che viene monitorata dal controller

La possibilità di generare una richiesta di servizio da parte di una periferica consente di interrompere l'attività del controller in maniera asincrona, anche quando l'interfaccia non è stata abilitata a comunicare mediante il data bus

Le cause che possono provocare una richiesta di servizio sono molteplici:

1) alcune sono legate al verificarsi di condizioni anormali che, se ignorate, potrebbero ostacolare il corretto funzionamento del sistema: acquisizione di una misura fuori scala da parte di uno strumento

2) altre invece potrebbero derivare dal verificarsi di particolari condizioni del sistema: l'ampiezza del segnale misurato supera una soglia predeterminata provocando la necessità di una immediata azione correttiva da parte del controller

Richiesta di servizio e procedure di interrogazione

RICHIESTA DI SERVIZIO

Quando una periferica di misura viene comandata per eseguire una misura, non si conosce in generale il tempo necessario per ottenere il risultato di misura. Pertanto, il bus verrebbe bloccato per tutto il tempo necessario allo strumento per eseguire e restituire la misura al controller

Se invece si ricorre alla procedura di richiesta di servizio, il controller può dare inizio alla misura e, immediatamente dopo, ad altre attività sul bus (senza bloccarlo); lo strumento nel frattempo prosegue nelle misure e non appena il risultato sarà disponibile, lo segnala al controller attraverso la linea SRQ. A questo punto, alla prima occasione, il controller si farà restituire il dato indirizzando la periferica di misura come *talker*

Richiesta di servizio e procedure di interrogazione

PROCEDURE DI INTERROGAZIONE

Tutte le periferiche in grado di operare una richiesta di servizio agiscono sulla stessa linea SRQ (struttura party-line)

Quindi il controller non può direttamente individuare quale periferica ha effettuato la richiesta

Si rende necessaria una procedura di *poll*:

1) Sequenziale (*serial poll*): privilegia la capacità diagnostica del sistema

2) Contemporanea (*parallel poll*): privilegia la velocità

Richiesta di servizio e procedure di interrogazione

SERIAL POLL

Il controller si rivolge in maniera sequenziale alle diverse interfacce connesse al bus ricevendo da ciascuna il *byte di status* che ne riassume la condizione operativa

Il serial poll è attivato dal controller mediante il comando (universale multilinea) SPE. A seguito di tale comando e fino all'invio dell'SPD, ogni periferica in grado di operare come *talker* risponde alla trasmissione del proprio TAD con un carattere che è il *byte di status*

Lo standard IEEE-488 prevede che il carattere trasmesso dalla periferica che ha attivato la linea SRQ abbia il bit 7 (bit RQS) allo stato logico vero. Gli altri bit descrivono il motivo della richiesta di servizio mediante codifiche dipendenti dal tipo di periferica

Richiesta di servizio e procedure di interrogazione

SERIAL POLL

Il controller definisce la priorità da associare alle richieste, semplicemente stabilendo l'ordine con cui indirizzare come *talker* le periferiche connesse al bus

Principali caratteristiche del *serial poll*:

- 1) possibilità di interrogare tutte le periferiche collegate al sistema;
- 2) possibilità di capire (una volta individuata la periferica) le cause della richiesta di servizio (potente funzione diagnostica)
- 3) lenta se le periferiche da interrogare sono molte

Richiesta di servizio e procedure di interrogazione

PARALLEL POLL

La procedura di interrogazione è contemporanea per al massimo 8 periferiche

Durante il *parallel poll* ogni interfaccia risponde alla interrogazione del controller riportando il proprio *bit di status* (RQS) su di una linea del data bus (ogni periferica può essere associata ad una data linea)

Il controller, esaminando il data bus e conoscendo la corrispondenza fra le linee e le periferiche, individua l'interfaccia che ha attivato la richiesta di servizio

Richiesta di servizio e procedure di interrogazione

PARALLEL POLL

I passi da svolgere per attivare il *parallel poll*:

1) Il controller deve trasmettere ad ogni periferica da configurare il comando indirizzato PPC seguito dal comando secondario PPE

In tal modo per ogni interfaccia viene assegnata la linea e il corrispondente livello logico da assegnarle in caso si richiama di servizio

2) Con $ATN=True$ il controller porta la linea EOI allo stato logico “vero”

Le interfacce precedentemente configurate rispondono in un tempo < 200 ns e il controller dopo $2 \mu s$ va a leggere il data-bus

Richiesta di servizio e procedure di interrogazione

PARALLEL POLL

Fino a quando il comando indirizzato PPD non disabiliterà singolarmente le interfacce, queste risponderanno al *parallel poll* in base alla configurazione ricevuta nella fase di abilitazione

La trasmissione del comando universale PPU provoca la perdita della configurazione di tutte le periferiche precedentemente abilitate