



Progettazione di un'interfaccia UART per comunicazione RS-232

Corso di Laurea Magistrale in Ingegneria Elettronica

Studenti

Goti Gianluca - 269825

Chisciotti Laura

Cerbai Matilde

Amato Giovanni Luca

Indice

1	Il protocollo RS-232	2
1.1	Caratteristiche generali	2
1.2	Descrizione del segnale	2
2	Trasmettitore	3
2.1	Struttura e caratteristiche principali	3
2.1.1	Data Path	4
2.2	Control Unit	5
2.3	Diagramma di stato	5
2.4	Timing Diagram	5
3	Ricevitore	5
4	Uart	5

1 Il protocollo RS-232

1.1 Caratteristiche generali

L'RS-232 è uno standard che permette lo scambio di informazioni a bassa velocità tra dispositivi digitali. Tale protocollo presenta tre caratteristiche principali:

- Seriale;
- Full-duplex;
- Asincrono;

Questo significa che è possibile inviare solamente un bit alla volta, è possibile trasmettere e ricevere contemporaneamente e che non viene scambiato alcun tipo di clock. Nella sua forma più semplice, tale protocollo ha un supporto fisico costituito da due fili, il "Tx" e l' "Rx" (in realtà ce ne sono molti di più, in quanto è stato progettato per comunicazioni a lunga distanza, per esempio 15 m). Per quanto riguarda i livelli logici, non vengono usati dei valori di tensione comuni, tale protocollo è molto robusto, infatti si può avere fino a 9V di margine di rumore.

In uscita si trova:

- $+12\text{ V} = 0\text{ Logico } "V_{OL}"$
- $-12\text{ V} = 1\text{ Logico } "V_{OH}"$

In ingresso si trova fino a:

- $+3\text{ V} = "V_{IL}"$
- $-3\text{ V} = "V_{IH}"$

Ovviamente queste non sono tensioni logiche normali, esistono dei circuiti (MAX232) che trasformano tali livelli logici in TTL o CMOS compatibili e viceversa.

1.2 Descrizione del segnale

Lo standard trasmette un bit alla volta, ogni word da 8 bit è incapsulata (framing) con uno "START" (0 logico) bit e uno "STOP" (1 logico) bit, il livello "neutro" è l'IDLE (1 logico). In generale le combinazioni possono essere molteplici, ci deve essere un accordo esplicito sulla lunghezza della word trasmessa, il bit di stop può durare per uno o più tempi bit, ci può essere parità o meno. Nel caso sotto esame il protocollo prevedeva:

- 8 bit (word)
- No parity
- 1 bit di STOP

Quando il sistema è in quiete è nello stato di "IDLE", che equivale all' 1 logico. Quando l'utente vuole iniziare una trasmissione ci deve essere sempre una transizione, infatti da IDLE si passa a START ottenendo così una transizione dall' 1 logico allo 0 logico. Il bit di STOP è importante in quanto se l'utente necessita di effettuare una trasmissione continua, la transizione 1 logico e 0 logico è sempre garantita, in quanto lo STOP riporta a livello alto la linea garantendo così la corretta transizione. Il timing diagram di una generica trasmissione RS-232 si presenta come in figura 1. Come si nota nella figura, il sistema

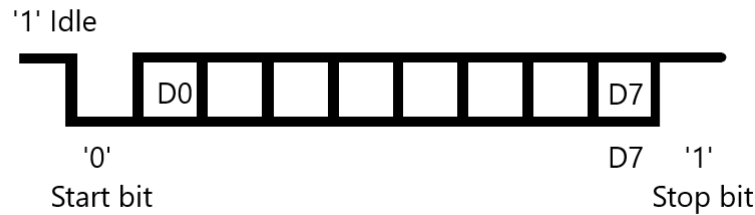


Figura 1: Timing diagram di una generica trasmissione RS-232

passa da uno stato di quiete, ovvero IDLE, al bit di START, eseguendo così la transizione che farà capire al ricevitore che è stata iniziata una nuova trasmissione. Infine il sistema passa in STOP e dopodichè va in IDLE.

2 Trasmettitore

2.1 Struttura e caratteristiche principali

In figura (INSERIRE FIGURA DELLA UART) si può vedere i principali segnali che arrivano e che partono dalla UART e che interessano il trasmettitore.

- DOUT: Questo è un bus da 8 bit che carica in modo parallelo il dato che deve essere trasmesso
- Data Valid: E' un segnale in ingresso che dice al trasmettitore quando campionare il dato da trasmettere
- TX RDY: E' il segnale di stato del trasmettitore, comunica all'esterno se il sistema è in trasmissione (busy) o meno
- TX: E' l'uscita seriale del trasmettitore

Quando il trasmettitore non lavora, il TX RDY è tenuto alto, quando è in trasmissione tale segnale è basso (busy). Il funzionamento di tale ricevitore è abbastanza semplice:

- L'utilizzatore esterno, per esempio un μC manda il dato che vuole trasmettere sulla seriale al DOUT;
- Dopo aver mandato il dato da trasmettere asserisce il Data Valid, in questo modo il trasmettitore campiona i dati che dovrà trasmettere;
- Il trasmettitore acquisisce i dati e li trasmette seguendo la logica del protocollo e pone al livello logico basso la linea TX RDY;
- Dopo che la trasmissione è finita il TX RDY è posto a livello logico alto e il sistema è pronto per una nuova trasmissione;

Il sistema lavora con un clock da 25 MHz, la trasmissione seriale è a 9600 baud, ciò significa che il tempo di bit è circa $300 \mu s$.

2.1.1 Data Path

Il datapath del trasmettitore è mostrato in figura 2.

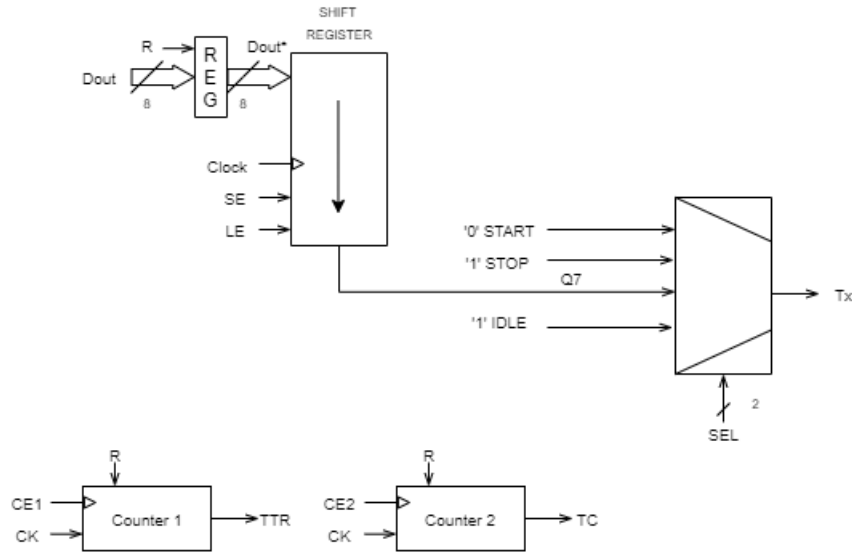


Figura 2: Datapath del trasmettitore

Lo shift register è il cuore di questo sistema, è lui che esegue la "serializzazione" del dato, in particolare questo ha un caricamento di tipo parallelo e una porta che abilita lo shift "SE" e una porta che abilita il load "LE". Idealmente appena il sistema avverte il Data Valid, lo shift register dovrebbe caricare i dati all'interno, questo però comporterebbe la creazione di una macchina di Mealy, in quanto un segnale esterno provoca un'immediata reazione nella macchina. Questo ovviamente si vuole evitare ad ogni costo, la soluzione adottata è la macchina di Moore, per questo il sistema risponde al Data Valid con un comando di "load" dello Shift register al colpo di clock successivo. Risulta quindi immediatamente evidente la necessità di un registro che permetta di ritardare del tempo necessario (in questo caso un colpo di clock) per permettere al sistema di rispondere correttamente e quindi di campionare il dato, caricandolo così all'interno dello shift register. La temporizzazione della trasmissione è assicurata da due contatori, il "Counter 1" assicura la corretta temporizzazione tra la trasmissione di un bit e l'altro e il "Counter 2" (SICURI CHE SIANO QUESTI NUMERI?) conta il numero di bit trasmessi durante la comunicazione, decretandone in seguito la fine. Entrambi i contatori hanno un terminal count che viene mandato alla Control Unit (CU) la quale risponde al colpo di clock dopo col comando corretto (può per esempio abilitare lo shift dello shift register oppure girare il mux).

2.2 Control Unit

La control unit utilizzata per questo progetto è molto semplice, è costituita da una parte sequenziale che fa progredire gli stati, occupandosi quindi anche di eventuali salti e una parte puramente combinatoria che manda i comandi al datapath. In figura 3 si può vedere uno schema di principio della CU utilizzata. Come si nota dallo schema, gli input entrano nella "next state logic" ma sono filtrati da un registro,

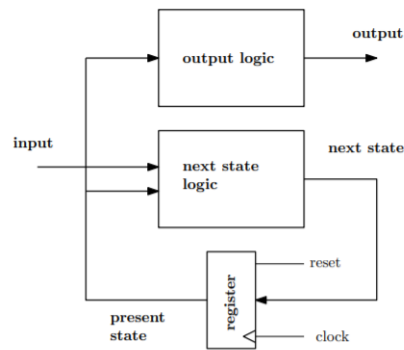


Figura 3: FSM a due processi

questo permette di svincolarsi dall'esterno, evitando di creare una macchina di Moore. Nel progetto, ogni volta che un segnale arriva dal datapath ed entra nella CU, questa risponde con un adeguato comando solamente al colpo di clock successivo.

2.3 Diagramma di stato

In figura (REF) si nota il diagramma di stato del ricevitore. ANDRA' COMMENTATO????

2.4 Timing Diagram

2.5 Test e risultati

3 Ricevitore

4 Uart