



di GIORGIO OBER (GIOBE2000)

l'angolo di Mr A.KEER



(parte settima)

PROGETTARE con le PORTE LOGICHE

Decoder/Demultiplexer

Prosegue in questo numero l'analisi dei principali dispositivi combinatori; è la volta dei Decoder/Demultiplexer, di cui scopriremo i segreti e tutti i componenti resi disponibili sul mercato.

Nella precedente puntata abbiamo analizzato i **multiplexer**, le *macchine combinatorie* in grado di spostare su un'unica uscita uno degli ingressi di un dato a più bit; è verosimile poter disporre anche di un componente in grado di assicurare l'operazione opposta.

Di questo ci occuperemo tra un po', non senza evidenziare delle suggestive scoperte; per tenere alta la *souspance* voglio però partire da altre fondamentali macchine combinatorie: i **decoder binari**.

DECODER BINARI (TTL) : 74LS139, 74LS138, 74LS154

La teoria booleana tratta con cura i *codici binari* e definisce **binario puro a n bit** quello che raccoglie tutte le possibili combinazioni delle n variabili coinvolte, in sostanza 2^n parole a n bit; se abbiamo a che fare con 4 variabili d'ingresso con esse potranno essere definite 16 *parole a 4 bit*, diverse tra loro.

Naturalmente questo *codice nativo* non è che la *raccolta* delle possibili *parole* a cui fanno riferimento le definizioni dei codici veri e propri, creati per garantire la sicurezza e la gestione dell'informazione binaria; tra di essi uno dei più importanti e utilizzati è il **codice BCD (Binary Decimal Code)**, formato (come molti altri) da 10 *parole di 4 bit* e pensato per *codificare in binario* ciascuno dei 10 *simboli del sistema di numerazione decimale*, fornendo in questo modo una concreta interfaccia tra i dati (binari) prodotti dalle macchine digitali e quelli (decimali) graditi a noi poveri umani.

La *Figura 1* mostra le tabelle che li rappresentano entrambi e sottolinea come il **BCD** sia un sottoinsieme del *codice binario puro a 4 bit*.

I **Decoder** sono dispositivi chiamati a scoprire (= *decodificare*) quale tra le possibili *parole* di questi codici è stata predisposta sulle loro linee d'ingresso (di solito 2, 3 o 4) attivando *la sola uscita* (tra le 4, 8 o 16, pari alla potenza di 2 del numero degli ingressi) che corrisponde alla *parola* stessa.

Quelli chiamati ad interpretare il *codice BCD* sono detti **Decoder decimali** (e saranno oggetto della nostra analisi in una prossima occasione) mentre quelli che trattano l'informazione *pura a n bit* sono detti **Decoder Binari**.

Codice Binario Puro a 4 bit					Codice BCD				
n	D	C	B	A	n	D	C	B	A
0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	1
2	0	0	1	0	2	0	0	1	0
3	0	0	1	1	3	0	0	1	1
4	0	1	0	0	4	0	1	0	0
5	0	1	0	1	5	0	1	0	1
6	0	1	1	0	6	0	1	1	0
7	0	1	1	1	7	0	1	1	1
8	1	0	0	0	8	1	0	0	0
9	1	0	0	1	9	1	0	0	1
10	1	0	1	0	10	1	0	1	0
11	1	0	1	1					
12	1	1	0	0					
13	1	1	0	1					
14	1	1	1	0					
15	1	1	1	1					

Figura 1 - a) codice binario puro a 4 bit; b) codice BCD

I 3 componenti principali previsti dalla serie **TTL LS** per questo dispositivo, descritto dai *datasheet* come **Decoder/Demultiplexer**, sono il **74LS139**, che ne contiene 2 a 2 ingressi (Figura 2), il **74LS138**, che ne contiene 1 a 3 ingressi (Figura 3) e il **74LS154**, un dispositivo a 4 ingressi, Figura 4, come il **74LS159**, ad esso *pin-out compatibile* ma a *collettore aperto*; scontata in quest'ultimo caso la necessità di un contenitore a 24 piedini, dato il considerevole numero di segnali coinvolti.

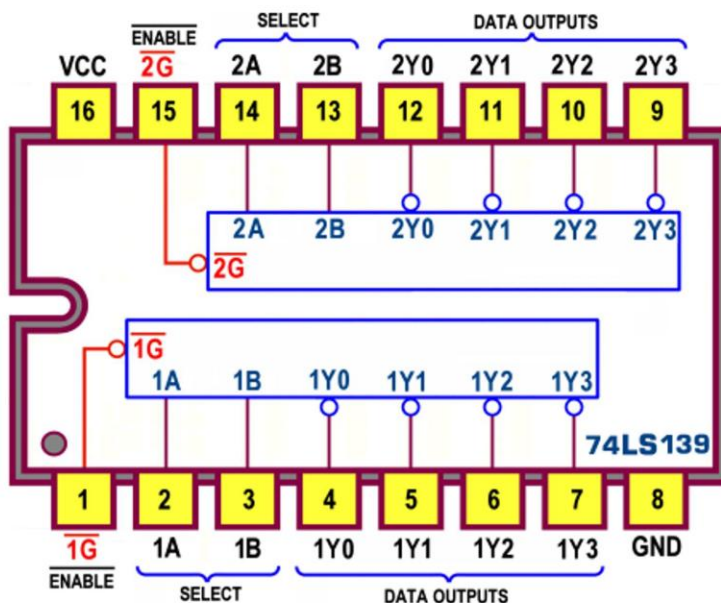


Figura 2 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Pin-out

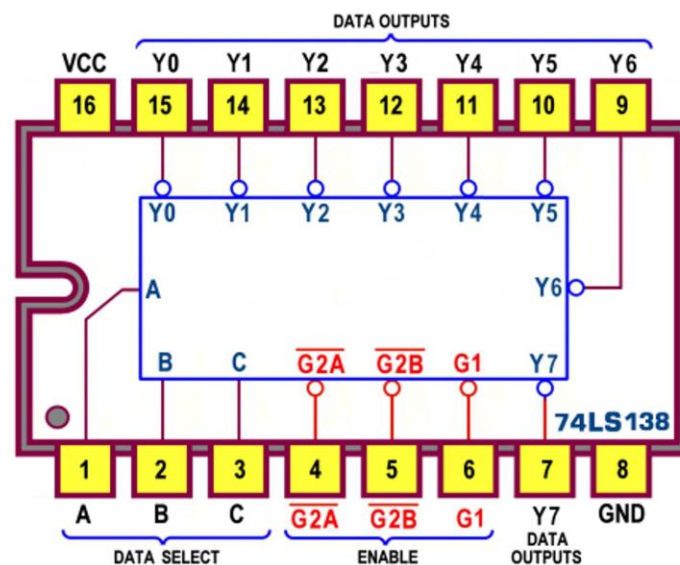


Figura 3 - 3-Line To 8-Line Decoders/DMUXs 74LS138: Pin-out

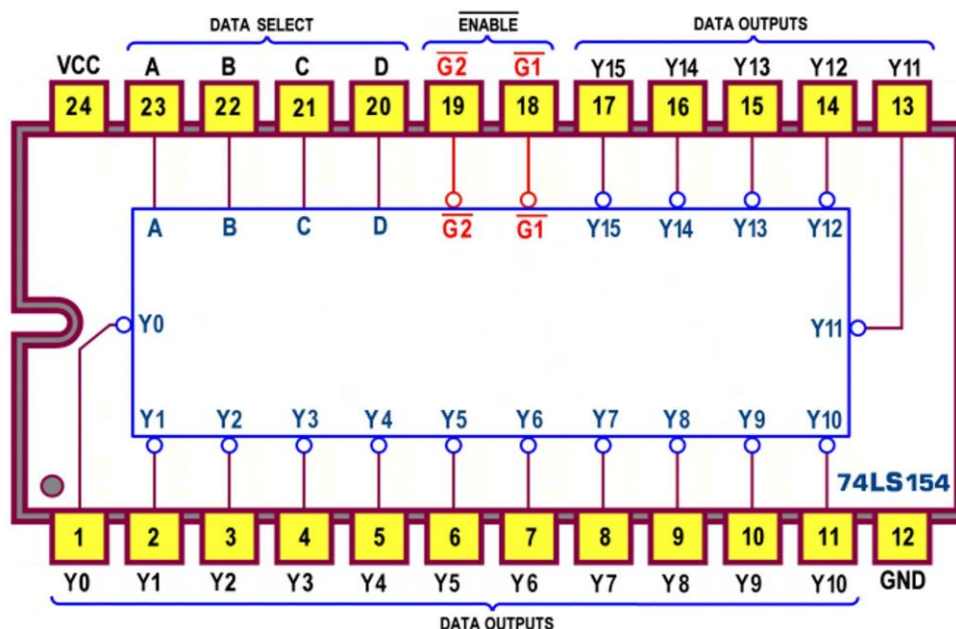


Figura4 - 4-line To 16-Line Decoders/DMUXs 74LS154: Pin-out

Osserviamo che già la definizione dei *datasheet* lascia intendere una disponibilità, non ancora del tutto evidente, di questi componenti a comportarsi come **DMUX**; ma per ora analizziamo la loro funzione come decoder binari.

Lo *schema funzionale* dei componenti combinatori (proposto talvolta come *schema logico*, *Logic Diagram* o *schematic circuit*) è decisamente interessante: di solito è piuttosto elaborato ma consente un'analisi dettagliata del loro funzionamento, aiutandoci a capire l'influenza di ciascun segnale, seguendone il percorso. La *Figura 5* mostra i dettagli del decoder **74LS139**

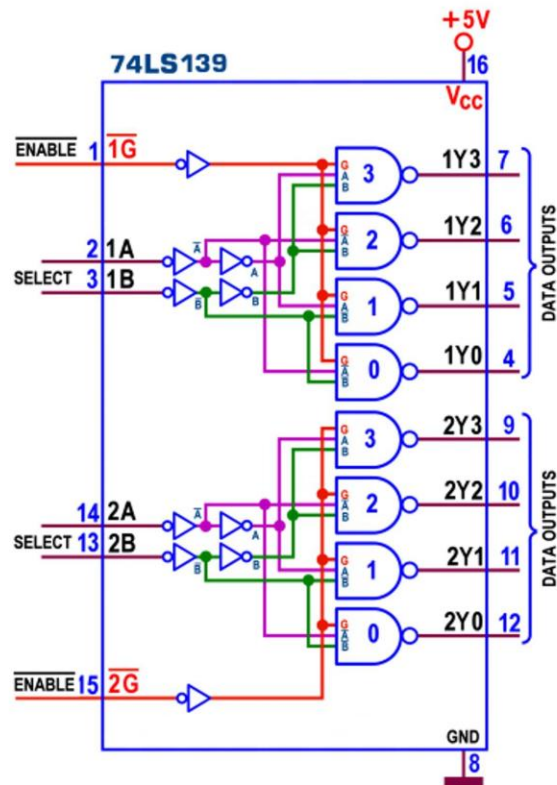


Figura 5 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Schema funzionale

Ciascuna delle due metà (identiche e tra loro indipendenti) mette immediatamente in risalto la struttura dei componenti di questa categoria: essi hanno tante uscite quante sono le possibili combinazioni binarie degli ingressi; in particolare il **74LS139** dispone di 2 ingressi di selezione (*select*), A e B, per cui avrà $2^2=4$ uscite.

Il loro funzionamento come **Decoder** consiste nell'attivazione di *una sola* linea d'uscita alla volta: come si vede ciascuna di esse è supportata da una NAND per cui è facile capire che sarà attiva (*bassa*) solo quella la cui NAND *dispone di un 1* su tutti e 3 i suoi ingressi.

In pratica, in virtù della presenza di una rete logica (detta appunto di *decodifica*) appositamente studiata per questo scopo, sarà attiva solo quella identificata dal numero decimale 2^n corrispondente al codice binario fornito sugli n ingressi di selezione.

La caratteristica, comune a tutti i **Decoder binari**, è dunque quella di avere *tutte le uscite a 1 meno una*, mostrando la capacità di saper riconoscere uno tra i possibili codici binari forniti in ingresso.

Tutto questo è vero solo se è attivo (*basso*) anche il segnale di controllo (*enable*); esso influenza contemporaneamente tutte le porte di ogni dispositivo abilitando l'azione di riconoscimento (*decodifica*) solo se posto a **0**; in caso contrario *tutte le uscite* saranno *forzate a 1*, compresa quella corrispondente al codice binario presente sugli ingressi, vanificando (*disabilitando*) lo scopo del dispositivo.

La *Figura 6* presenta la *Tabella di Verità (Function Table)*, chiamata a sintetizzare le considerazioni appena fatte.

74LS139						
INPUT			OUTPUT			
SELECT		ENABLE	Y0	Y1	Y2	Y3
A	B	G				
X	X	H	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
H	L	L	H	H	L	H
H	H	L	H	H	H	L

H livello logico **alto**
 L livello logico **basso**
 X livello logico **irrelevante**

Figura 6 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Tabella di verità

Lo *schema pratico* (Figura 7) offre il simbolo del **74LS139**, da utilizzare direttamente nel contesto di un progetto, evidenziando il nome dei *segnali logici* che entrano e di quelli che escono e il *numero* dei piedini ad essi riservati.

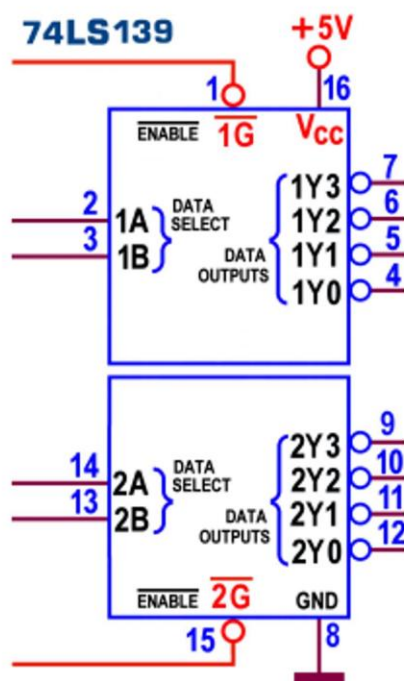


Figura 7 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Schema pratico

Fin dall'inizio abbiamo anticipato che questa categoria di componenti può essere utilizzata con successo anche come **Demultiplexer** o **Selettore d'uscita** o **DMUX**, dispositivo chiamato a trasferire l'informazione dall'unico ingresso previsto verso più uscite (solo una alla volta, di norma in numero di 2, 4, 8 o 16).

La Figura 8 mostra una *interpretazione logica* del funzionamento del **74LS139** e aiuta a capire questa particolarità.

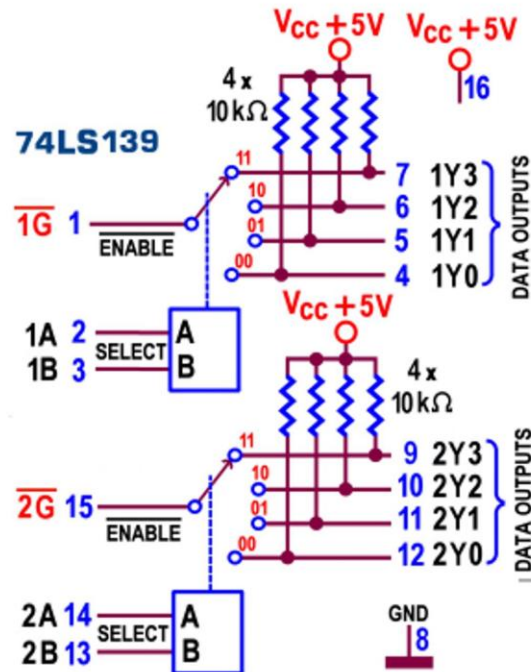


Figura 8 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Schema equivalente

La sensazione che produce questo schema è istintivamente quella di un *commutatore "a 2 vie - 4 posizioni"*, analogo a quello proposto per i *multiplexer*, la scorsa puntata, ma *"alla rovescia"*, per configurare ciascuna *via* come dotata di 1 ingresso e di 4 uscite.

Ma di fatto si tratta dello stesso *decoder* di prima, analizzato da un altro punto di vista: gli ingressi di selezione mantengono la stessa funzione esercitata nel funzionamento come *decoder*, consentendo di selezionare una delle 4 possibili uscite, mentre l'unica abilitazione (G1 o G2, attiva *bassa*) è a tutti gli effetti l'ingresso del **DMUX**.

E' facile dimostrare infatti che il valore logico posto su questo piedino passa inalterato sull'uscita selezionata, sia quando G1 (o G2) vale **0** (componente *abilitato*) che quando vale **1** (componente *non abilitato*).

La *Figura 8* mostra sulle uscite una batteria di 4 resistori di *pull-up*: essi non sono fisicamente presenti negli integrati ma servono *per dare rigore* allo schema equivalente: la Tabella di verità prevede infatti che tutte le uscite non selezionate siano a **1** logico e, nello schema questo non sarebbe possibile in assenza dei resistori, perché i contatti *aperti* non avrebbero valore logico.

In chiusura è interessante sottolineare l'attenzione di molti costruttori nei confronti di questa duplice funzionalità: nei *datasheet* questo componente è propriamente definito "*Dual 2-Line-to-4-Line Decoder or 1-Line-to-4-Line Demultiplexer*".

Essa è ribadita anche dai *simboli logici* predisposti dallo *standard IEEE*; è subito chiaro che le norme internazionali sottolineano ufficialmente una possibile differenza funzionale, prevedendo per lo stesso componente 2 diverse proposte (vedi *Figura 9*); come *decoder* la notazione identifica i 2 ingressi di selezione A e B con il numero che indica il rispettivo *peso*, cioè l'esponente della potenza di 2 che stabilisce quale delle possibili uscite sarà abilitata (se il segnale G, indicato con EN, è attivo); come *demultiplexer* i 2 ingressi sono raccolti da una parentesi graffa che li fa corrispondere proprio a G, alla destra del quale la grafica $0/(2^n-1)$ non è una frazione ma indica l'intervallo previsto per le uscite, nel nostro esempio da 0 a 3.

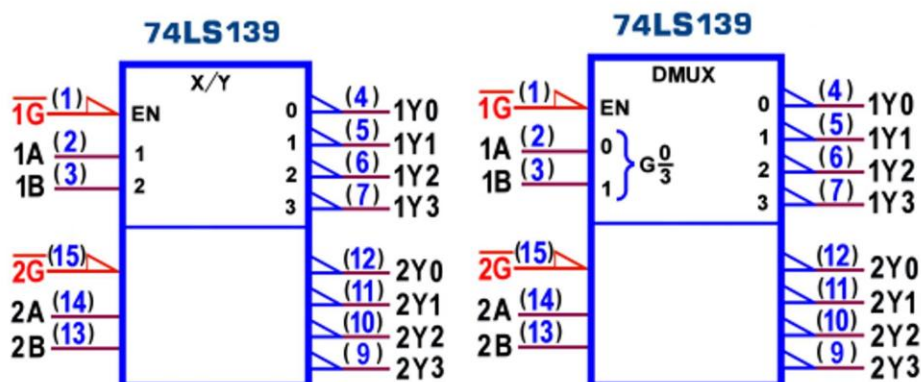


Figura 9 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS139: Simbolo logico ANSI/IEEE Std. 91-1984

La serie **TTL LS** prevede altri integrati per la categoria dei decoder "da 2 a 4": il **74LS539** (una versione a 20 piedini con uscite *attive basse, three-state*) e i **74LS155** e **74LS156** (quest'ultimo a *collettore aperto*), diversi dal **74LS139** ma più versatili (vedi Figura 10): avendo le 2 linee A e B di selezione in comune, le due metà possono disporre ciascuna di un terzo ingresso C (uno negato e uno diretto) che uniti insieme trasformano il componente in decoder "da 3 a 8".

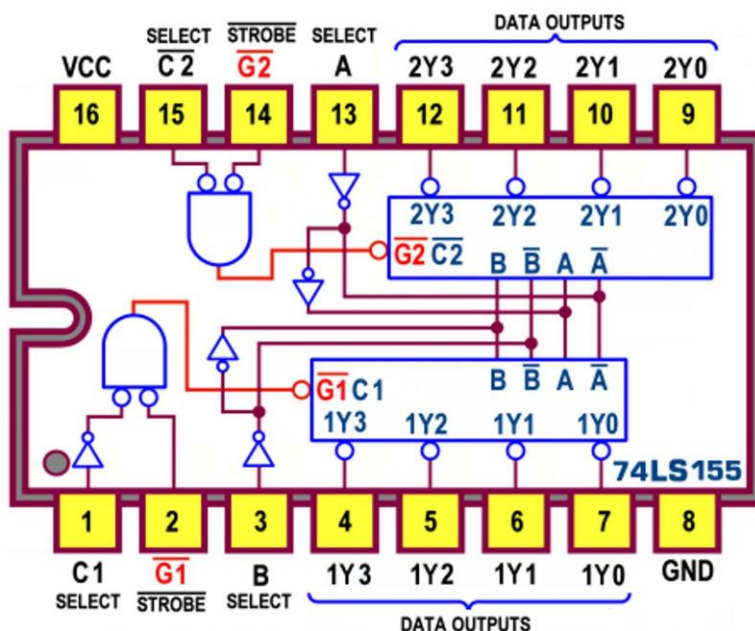


Figura 10 - Dual 2-Line To 4-Line Decoders/DMUXs 74LS155: Pin-out

Naturalmente, all'occorrenza, anche un doppio decoder "da 2 a 4" come il **74LS139** può diventare un decoder "da 3 a 8", rinunciando alla possibilità di poterne controllare l'abilitazione; la Figura 11 mostra la tecnica per disporre di questo tipo di *espansione logica*, applicabile a tutti i componenti della famiglia.

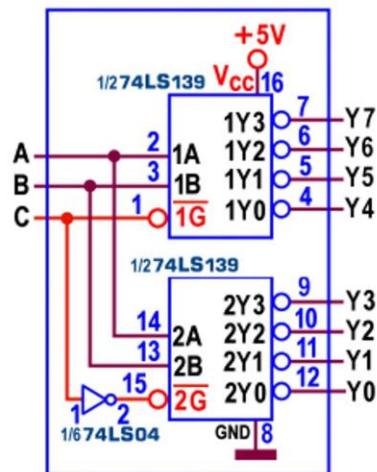


Figura 11 - Espansione di un Decoder 2-To-4-Line in un Decoder 3-To-8-Line

Per i 2 rimanenti dispositivi **Decoder/DMUX** valgono le medesime considerazioni; a causa della maggiore quantità di uscite (8 per il **74LS138** e 16 per il **74LS154**) i rispettivi *schemi funzionali* sono decisamente *ingombranti*; anche per questo (sebbene essi siano sempre disponibili nei *datasheet*) non ritengo necessario ridisegnarli qui, ma la struttura di entrambi è sempre quella del **74LS139**, basata su una rete logica in grado di *decodificare* le linee di selezione (ora rispettivamente 3 e 4) al fine di aprire *una sola* delle NAND associate a ciascuna delle uscite previste.

Questo vale anche per le loro *Tabelle di verità*, chiamate a ribadire la caratteristica (già rimarcata) di avere *tutte le uscite a 1 meno una*, fatto ben evidente, indicato dalla L (*Low*) in diagonale sulle colonne delle uscite, attorniate solo da H (*High*), vedi *Figura 6*.

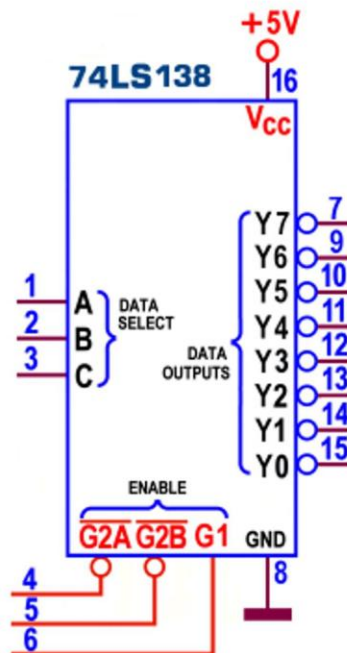


Figura 12 - 3-Line To 8-Line Decoders/DMUXs 74LS138: Schema pratico

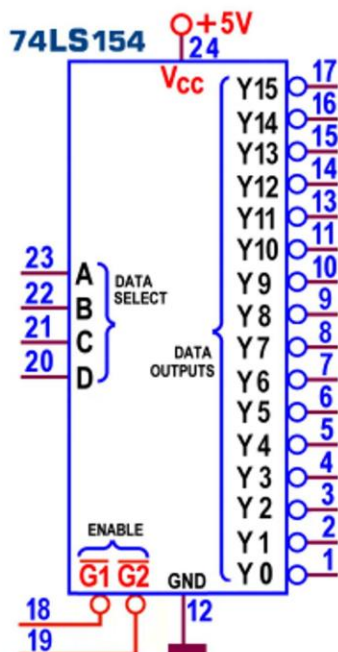


Figura 13 - 4-Line To 16-Line Decoders/DMUXs 74LS154: Schema pratico

Le Figure 12 e 13 suggeriscono invece per entrambi lo *schema pratico*, da utilizzare nei progetti, consigliato per la sua sintetica completezza.

Vale la pena sottolineare la grande versatilità di *controllo* garantita da entrambi, per la presenza rispettivamente di 3 (una delle quali *attiva alta*) e 2 linee di abilitazione; è importante ricordare però che il funzionamento come **Decoder** è subordinato alla presenza di segnali attivi su tutte le linee di ciascun gruppo. Così il **74LS138** funzionerà solo se $G1=1$, $G2A=0$ e $G2B=0$.

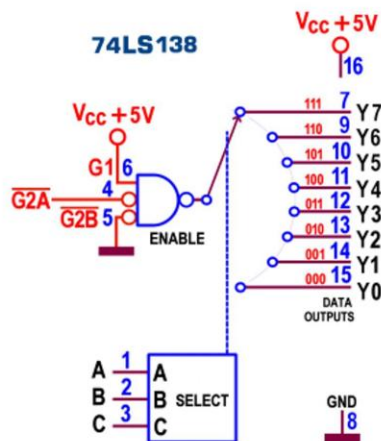


Figura 14 - 3-Line To 8-Line Decoders/DMUXs 74LS138: Schema funzionale

Per questi dispositivi la nativa predisposizione ad essere utilizzati come **DMUX** riconduce entrambi alle funzioni di un *commutatore "ad una via - 8 posizioni"* (per il **74LS138**) o "*ad una*

via - 16 posizioni" (per il **74LS154**); di sicuro la presenza di più linee di *enable* rende versatile anche la scelta dell'unico ingresso.

La Figura 14 mostra l'interpretazione logica del funzionamento del **74LS138**: per non renderla pesante ho evitato di collegare gli 8 resistori di *pull-up* (per altro non presenti nell'integrato) dando per scontato che le uscite non selezionate siano comunque a **1** logico.

La presenza di ben 3 linee di abilitazione rende necessario fare una scelta per stabilire quale delle 3 si debba ritenere *ingresso* del **DMUX** e, in ogni caso, i rimanenti 2 devono essere resi *ininfluenti*; la figura mostra come comportarsi se si desidera affidare il compito a G2A; la presenza di un 1 su G1 e di uno 0 su G2B tiene aperta (per la parte di loro competenza) la porta NAND, cosicché il valore della sua uscita è logicamente identico a quello del suo terzo ingresso G2A che passa inalterato anche sull'uscita selezionata del **DMUX**.

I simboli logici predisposti dallo standard IEEE per il **74LS138** sono ancora 2 (vedi Figura 15), uno come *decoder* e uno come *demultiplexer*; in entrambi i casi per la notazione utilizzata valgono le stesse considerazioni fatte poco fa per il **74LS139**, ovviamente applicabili anche al **74LS154**, i cui simboli sono facilmente estrapolabili da quelli ora proposti.

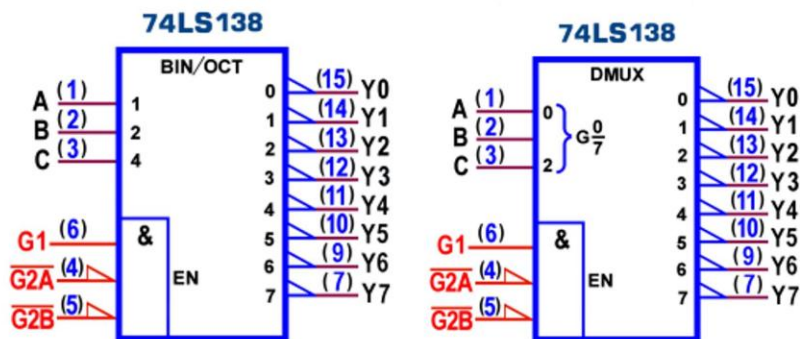


Figura 15 - 3-Line To 8-Line Decoders/DMUXs 74LS138: Simbolo logico ANSI/IEEE Std. 91-1984

Una curiosità: la scorsa puntata abbiamo visto che tra le configurazioni tipiche dei *multiplexer* c'è anche quella che contiene 4 **MUX** a 2 ingressi e un'uscita, disponibile nel **74LS157**; ebbene, non esiste integrato che garantisca la struttura complementare, cioè un **DMUX** "da 1 a 2", sebbene essa (in sintonia con gli schemi già proposti) sia facilmente realizzabile con poche porte logiche (vedi Figura 16).

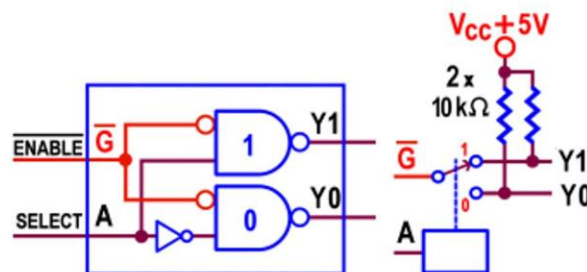


Figura 16 - 1-Line To 2-Line Decoders/DMUXs: Schema funzionale

La serie **TTL LS** prevede altri integrati per la categoria dei decoder "da 3 a 8", tutti *pin-out compatibili* con il **74LS138**: il **74LS238** è ad esso simile, ma con uscite *attive alte*; altri 3 componenti dispongono anche di memoria e utilizzano una delle abilitazioni come sincronismo di memorizzazione: nel **74LS237** (con uscite *attive alte*) e nel **74LS137** (con uscite *attive*).

basse) la linea sul pin4 è utilizzata come *latch enable*: quando è a *livello basso* funzionano da decoder; nel passaggio *da basso a alto* il dato posto sugli ingressi viene memorizzato, rendendo stabili le uscite per tutto il tempo che rimane a *livello alto*, ignorando ogni eventuale variazione degli ingressi stessi. Il **74LS131** (con uscite *attive basse*) utilizza la linea sul pin4 come *clock enable*: la memoria coinvolta è dunque di tipo *d-type* e consente la memorizzazione dei dati in ingresso sul *fronte di salita*, rendendo stabili le uscite fino all'arrivo di un nuovo fronte. Il **74LS538** e il **74LS259** sono invece versioni non *pin-out compatibili*: il primo (dotato di 20 piedini) ha uscite *attive basse* di tipo *three-state* e il secondo è propriamente una memoria a 8 bit di tipo *latch* ma si presta anche ad essere usato come decoder "da 3 a 8" con *uscite attive alte* con memoria.

Ricordiamo ora alcune applicazioni per i **Decoder Binari**; un utilizzo curioso è quello di *generatore di funzioni logiche*: poichè ciascuna delle 2^n uscite corrisponde senza dubbio ad *una e una sola* combinazione delle n variabili d'ingresso è facile pensare che esse possano essere associate anche al rispettivo *mintermine* (cioè all'espressione che esprime il prodotto delle n variabili prese in forma diretta o in forma negata a seconda del loro valore corrente).

La teoria digitale prevede che ogni possibile funzione booleana sia rappresentabile (in forma *canonica disgiuntiva*) dalla **somma** di tutti e soli i mintermini per i quali essa vale 1; se le uscite fossero *attive alte* per realizzare un determinato progetto logico sarebbe dunque sufficiente raccogliere con una **OR** tutte e sole quelle corrispondenti alle combinazioni per le quali la *Tabella di verità* prevede valore uguale a 1.

La *Figura 17* utilizza 1/2 **74LS139** per simulare una **OREX** e, poichè le sue uscite sono *attive basse*, si rende necessario un inverter su tutte quelle coinvolte nel progetto; per evitare questo spreco è consigliabile sostituire la rete **NOT-OR** con una **NAND** (vedi dettaglio sottostante).

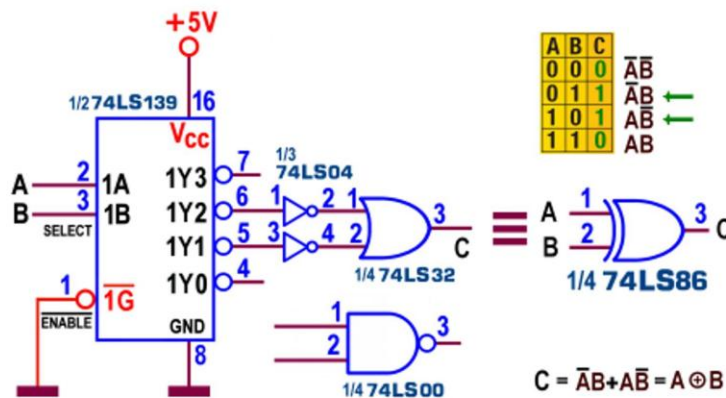


Figura 17 - 1/2 2-Line To 4-Line Decoders/DMUXs 74LS139: generatore di funzioni

Con questa tecnica è quindi possibile realizzare qualsiasi *Tabella di verità*, in particolare quelle associate a funzioni che non è possibile minimizzare oppure quelle, con numerose variabili in ingresso, per le quali le tecniche di minimizzazione risultano improbabili o complesse; applicando a 2 **74LS154** la tecnica di *espansione logica* proposta in *Figura 11* è per esempio possibile realizzare funzioni a 5 variabili.

I **Decoder Binari** sono particolarmente utili anche nel progetto digitale dedicato ai microprocessori; con essi infatti si realizzano le importanti reti che si occupano di interpretare gli indirizzi e i segnali di controllo destinati a localizzare senza ambiguità le periferiche di *Input/Output* e le *locazioni di memoria* di un computer.

La *Figura 18* utilizza un **74LS138** per creare 8 segnali di sincronismo adatti a controllare altrettante periferiche d'ingresso: essi sono in ogni caso dei *brevi segnali attivi bassi*, presenti

sulle uscite del decoder solo se il processore decide di riservare attenzione all'I/O (cioè se **M/IO=0**), se intende *leggere* le periferiche (**RD=0**) e se il nome dei dispositivi è compreso nell'intervallo da **03F8H** a **03FFH**; tutte e sole queste condizioni concorrono ad *abilitare* il decoder e coinvolgono (per scelta) solo le prime 10 linee del *bus indirizzi*, affidando alle 3 meno significative il compito di stabilire quale degli 8 segnali sarà attivo. Da notare che con esse, ricombinando con degli inverter i rispettivi valori, è possibile concedere alla CPU la possibilità di *leggere* (**RD**) fino a $2^{10}=1024$ diversi dispositivi e, sostituendo **RD** con **WR**, possibilità di *scrivere* altrettante.

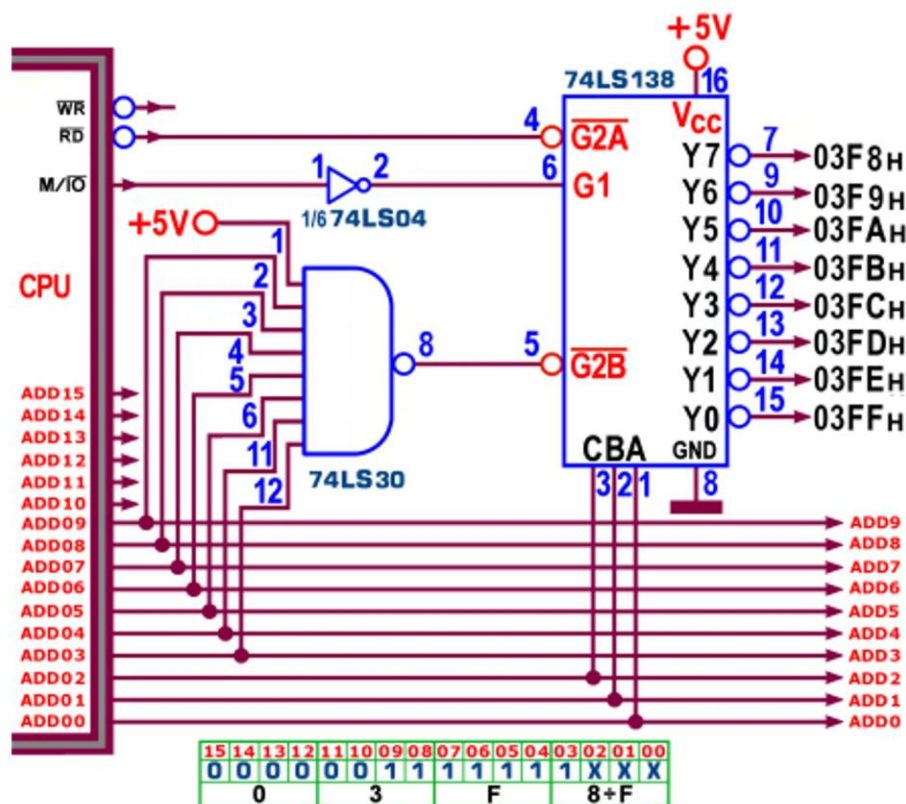


Figura 18 - 3-Line To 8-Line Decoders/DMUXs 74LS138: Decodifica indirizzi di I/O

La Figura 19 mostra un processore a 8 bit con 16 linee d'indirizzo, in grado quindi di indirizzare $2^{16}=65536$ (=64kbytes) locazioni di memoria, organizzata in 4 chip (una ROM e 3 RAM) ciascuno da 16Kbytes; per accedere ad ogni cella la CPU si avvale di una rete di decodifica basata su 1/2 74LS139, utilizzando le sue uscite per generare i 4 brevi impulsi attivi bassi utilizzati, uno in alternativa all'altro, come *Chip Enable* della memoria che contiene le locazioni puntate dal valore corrente delle 16 linee d'indirizzo. Le 14 linee meno significative sono spedite a tutte e 4 contemporaneamente, localizzano in ciascuna la stessa cella, tra le $2^{14}=16384$ possibili, ma solo quella abilitata dall'uscita del decoder (corrispondente al valore delle linee ADD14 e ADD15) sarà a disposizione della CPU stessa, in scrittura (se **WR**=0) o altrimenti in lettura; si può notare che comunque tutto ciò è possibile solo se **M/IO**=1, cioè se la CPU intende coinvolgere memoria e non I/O.

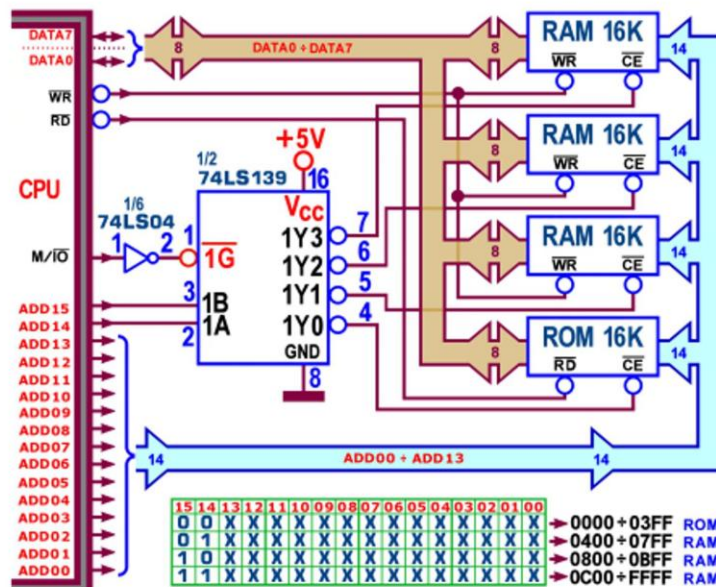


Figura 19 - 1/2 2-Line To 4-Line Decoders/DMUXs 74LS139: Decodifica chip di Memoria

In entrambi gli schemi appena proposti ho predisposto una tabellina in grado di aiutare a ricostruire rispettivamente il *nome* (binario) della periferica e il *range* di indirizzi di ciascun chip di memoria: si tratta di una buona abitudine che aiuta a documentare e comprendere al meglio progetti di questo tipo.

In chiusura ricordiamo che un **DMUX** (in coppia con un **MUX**, vedi puntata precedente) è molto adatto per organizzare la *trasmissione a distanza* dell'informazione binaria a più bit, utilizzando una sola linea: in partenza il **MUX** serializza uno dopo l'altro gli *n* bit dei suoi *n* ingressi; in arrivo il **DMUX** riconverte *in parallelo* gli *n* bit ricevuti sul suo unico ingresso. Sebbene il principio sia molto intuitivo bisogna sottolineare che tutto funzionerà solo se la ritrasmissione è corredata da sincronismi in grado di far capire con certezza a chi riceve quando inizia la ricezione e solo se la *frequenza di selezione* è la stessa da ambo i lati.

I componenti descritti da questo articolo possono dunque concorrere solo in parte alla realizzazione di questo progetto; la loro versione integrale, usata da tutti i microcontrollori, è offerta sotto forma di **UART** (Universal Asynchronous Receiver Transmitter), il noto *ricevitore trasmettitore asincrono universale* indispensabile per gestire la comunicazione seriale.

La tecnica di *multiplexing* e *demultiplexing* è diffusa in molti settori (per esempio quello telefonico, per trasferire a distanza, ad intervalli regolari, numerosi *canali*): in questa occasione voglio ricordare anche la *gestione dei digit* di un visualizzatore, sempre economicamente vantaggiosa.

DECODER BINARI (CMOS) : 4514, 4515, 4555, 4556

La serie **CMOS 4000** prevede il **4556** (con uscite *attive basse*, Figura 20) e il **4555** (idem con uscite *attive alte*) contenenti 2 **Decoder/DMUX** "da 2 a 4", entrambi *pinout* compatibili con il **74LS139**; e il **4515** (con uscite *attive basse*, Figura 21) e il **4514** (idem con uscite *attive alte*), contenente un **Decoder** "da 4 a 16" **con una memoria**, di tipo *latch RS*, su ciascun ingresso, in grado di bloccare l'informazione presente su di esso sul fronte di discesa dello

strobe e di mantenerla finché questo segnale rimane a 0; il segnale di *inhibit* provvede a forzare tutte le uscite a 1 (4515) o a 0 (4514) indipendentemente dallo stato dei dati d'ingresso e dello *strobe*.

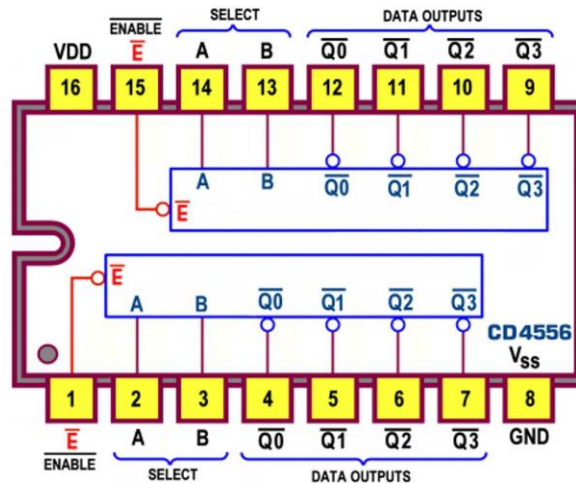


Figura 20 - Dual 2-Line To 4-Line Decoders/DMUXs 4556: Pin-out

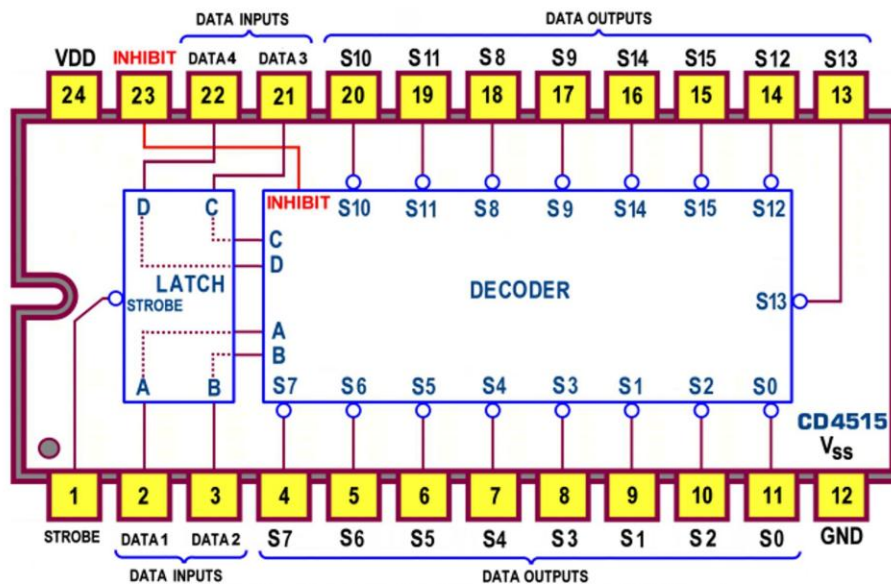


Figura 21 - 4-line To 16-Line Decoders with Latches 4515: Pin-out

Questa serie prevede anche numerosi *demultiplexers analogici* (che funzionano perfettamente anche in *digitale*) dotati di interruttori (*switch*) bidirezionali controllati digitalmente, come il **4053** (triplo **DMUX** "da 1 a 2"), il **4052** (doppio a "da 1 a 4"), il **4051** (singolo "da 1 a 8"), il **4067** (singolo "da 1 a 16") e il **4097** (doppio "da 1 a 8").

Ciascuno degli interruttori bidirezionali analogici presenti (2, 4, 8 o 16) è collegato da una parte ad una linea indipendente e dall'altra ad una linea in comune con gli altri *switch*; ogni linea è intrinsecamente di *input/output* ed è quindi possibile scambiarne i ruoli, facendo assumere agli ingressi la funzione di uscite e alle uscite quelle d'ingresso.