



di GIORGIO OBER (GIOBE2000)

l'angolo di Mr A.KEER



(parte venticinquesima)

PROGETTARE con le PORTE LOGICHE

Registers File

Continua in questa puntata la descrizione dei componenti all'elemento di memoria di tipo D; la struttura del Registro, fondamentale nel progetto digitale, viene ora organizzata in batteria, al fine di disporre di dispositivi che possono essere scritti e letti contemporaneamente.

L'importanza funzionale degli elementi di memoria di tipo **D** è decisamente degna di nota: la loro presenza sta alla base di numerosi dispositivi, a partire da quelli chiamati **Registri** (che contengono da 4 fino a 64 **D-Type**), trattati in dettaglio la volta scorsa; in sostanza ciascuno di essi può essere inteso come una memoria sulla quale può essere letta o scritta una sola parola binaria, di solito formata da tanti bit quanti sono i **Flip-Flops** coinvolti.

Abbiamo sottolineato come questa struttura sia fondamentale nell'ambito dei microcontrollori, non solo come *porta d'uscita* (per assicurare in modo stabile l'informazione alle periferiche specializzate a ricevere dati, come una batteria di Led o di Relè) ma anche come *memoria temporanea interna* agli stessi processori (al fine di facilitare l'elaborazione aritmetica o logica dell'informazione assunta o lasciata sul Bus Dati).

In questa puntata ci occuperemo di memorie più grandi, realizzate raggruppando più **Registri** e organizzandoli in modo da poter essere letti e/o scritti a piacere; le strutture così ottenute sono piuttosto numerose e sono catalogate in due categorie: i **Registers File** e le **Random Access Memories** (sinteticamente note come **RAMs**).

Registers File (TTL): 74170, 74670, 74870, 74871, 74172

Un **Register File** è una matrice bidimensionale (*array*) di **MxN** memorie elementari di tipo **D**; può essere definito "Banco di Registri" e, in sostanza, è un insieme di **M** Registri (le colonne della matrice) ciascuno dei quali è in grado di memorizzare **N** bit (le righe della matrice); in questo assomiglia molto alle memorie RAM, nelle quali la lettura o la scrittura dei dati in esse contenuti può avvenire in qualsiasi ordine, puntando a caso una qualsiasi locazione, cioè (come si suol dire) "ad accesso casuale" (*Random Access*).

Per consentire la localizzazione del Registro in cui inserire (o da cui estrarre) un dato è necessario disporre di un certo numero di linee di controllo; alcune di esse servono per indicarne il numero (espresso in binario) e per questo sono dette *linee d'indirizzo*: se il numero dei Registri contenuti nel Banco è pari a **M=8** saranno necessarie 3 linee, cioè sempre in

quantità pari all'esponente della base 2 necessario per esprimere M (nel nostro esempio, appunto, $2^3=8$).

Stabilito l'indirizzo della locazione (Registro) è necessario attendere che il valore posto sulle linee di indirizzo sia stabile, e poi abilitare la scrittura o la lettura, con l'aiuto di una o più linee di sincronismo; di solito (come vedremo in altra puntata) alle memorie RAM ne basta una, il cui livello attivo (basso) abilita la scrittura, assicurando invece automaticamente la lettura delle celle puntate quando è disattiva (alta).

I **Registers File** sono, invece, componenti speciali, molto versatili perché dispongono di linee separate e autonome (d'indirizzo e di abilitazione) sia per la scrittura che per la lettura; questo significa che ognuno degli M Registri può essere letto in qualsiasi momento anche se, contemporaneamente, è in atto una sua scrittura, cosa impossibile con le **RAMs**.

Per i dettagli funzionali è conveniente far riferimento alla mancata di componenti reali resi disponibili dalla famiglia TTL, a cominciare dal **74LS670**, definito **4-by-4 Register File with 3-state outputs**, che permette la scrittura e la lettura simultanea di quattro diversi Registri; la *Figura 1* mostra il suo *pin-out*, dal quale appare evidente la possibilità di controllare le 4 parole da 4 bit ciascuna con tre linee per la loro scrittura (*Write Select*, W_A e W_B , e *Write Enable*, G_W) distinte dalle tre linee necessarie per la loro lettura (*Read Select*, R_A e R_B , e *Read Enable*, G_R).

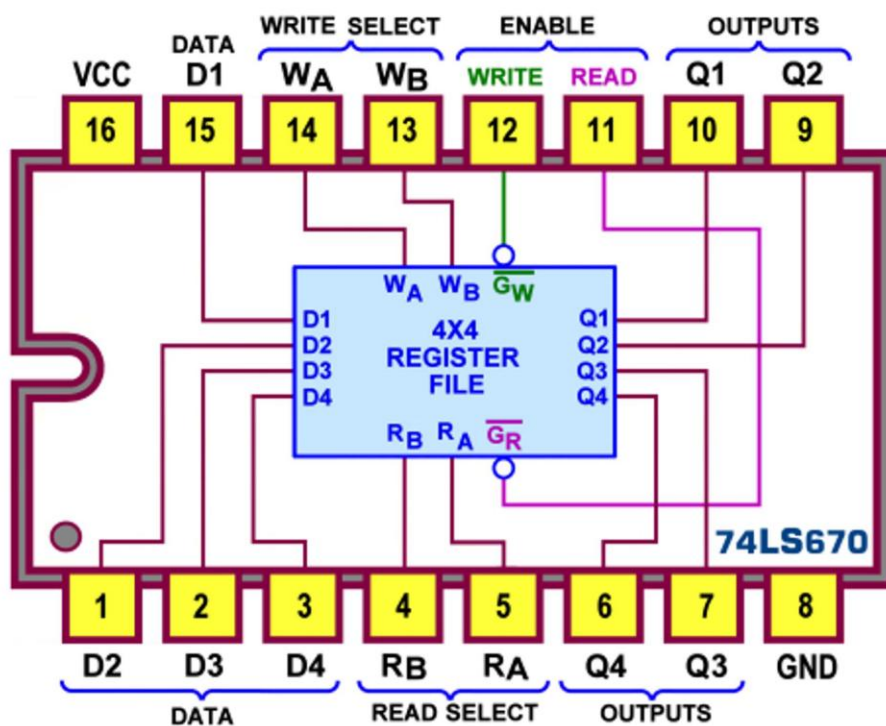


Figura 1 - 4-by-4 Register File, 74LS670: Pin-out

Lo *schema funzionale* (*Figura 2*) è particolarmente impegnativo ma altrettanto utile per capire i segreti di questo dispositivo; intanto appare evidente che i 16 elementi di memoria utilizzati sono **D-Latch** (contrariamente a quelli usati nelle memorie RAM, di tipo **D-Type**), cioè tali da "inseguire" in uscita il valore dell'ingresso D per tutto il tempo in cui loro abilitazione E rimane a 1, memorizzando quello presente nell'istante corrispondente al fronte di discesa di E.

Si tratta dunque di memorie attive sui livelli (**Level Triggered**) del segnale di controllo (.. un'abilitazione, non un *Clock*) e il loro tipico "effetto trasparenza" obbliga (al fine di evitare

cambiamenti di stato indesiderati) ad essere ben certi che l'ingresso D sia stabile e a pensare al segnale di abilitazione E come ad un brevissimo impulso a 1 (sostanzialmente racchiuso dai suoi 2 fronti di salita e di discesa, così da essere ritenuti pressoché coincidenti), breve ma non troppo, per dare alla memoria almeno il tempo per commutare.

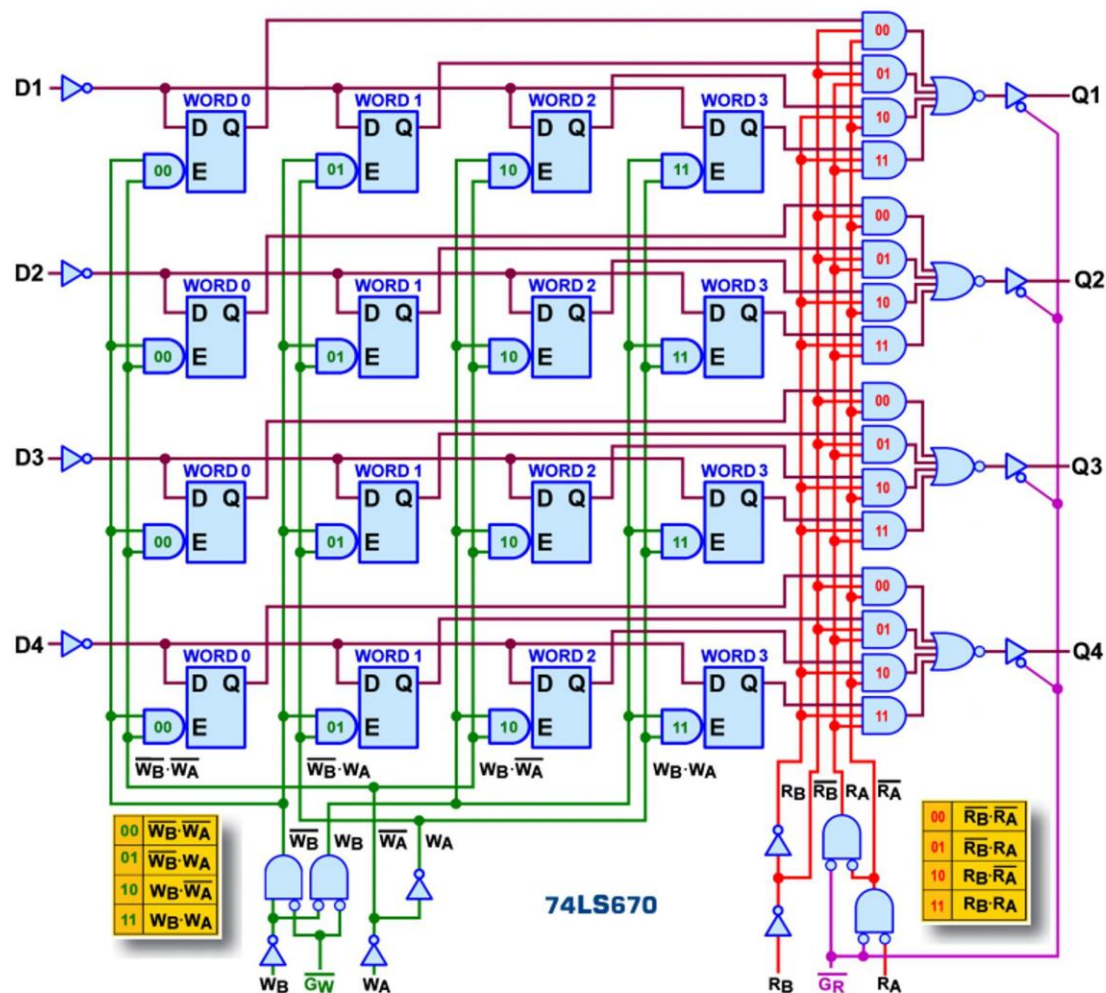


Figura 2 - 4-by-4 Register File, 74LS670: Schema funzionale

Entrambe le condizioni sono facilmente assicurabili e comunque sono relative alla sola fase di scrittura; per ognuno dei **D-Latch** l'abilitazione E è assicurata dal prodotto logico (AND) di due segnali, ottenuti dalla decodifica del numero binario a 2 bit predisposto sulle linee d'indirizzo in scrittura, W_A e W_B , abilitata dal breve impulso attivo basso fornito su G_W ; la *Figura 3a* mostra due varianti del dettaglio relativo al circuito di controllo della scrittura offerto da diversi datasheet e la *Figura 3b* offre un paio di schemi, equivalenti ad entrambi i precedenti.

La parola a 4 bit da memorizzare è presentata ai quattro ingressi di dato (da D1 a D4) e finisce in parallelo sugli ingressi interni D di ciascun latch, ma ovviamente solo uno dei 4 Registri del Banco sarà autorizzato a riceverla; a questo provvede il decoder che, in funzione del valore corrente delle linee d'indirizzo W_A e W_B , farà in modo di aprire (forzando un 1 logico su entrambi i suoi ingressi) le sole porte AND ad esso relative.

In realtà le porte autorizzate dalla decodifica del codice binario rimangono aperte solo per il brevissimo istante durante il quale il segnale G_W (il vero sincronismo di scrittura) viene tenuto basso, rendendo la locazione indirizzata trasparente ai dati, e memorizzando quello presente nell'istante corrispondente al fronte di salita di G_W ; da questo momento, per tutto il tempo durante il quale quest'ultimo rimane alto, le linee di ingresso dati sono inibite e il loro livello non potrà causare alcun cambiamento alle informazioni memorizzate sulle uscite dei latches.

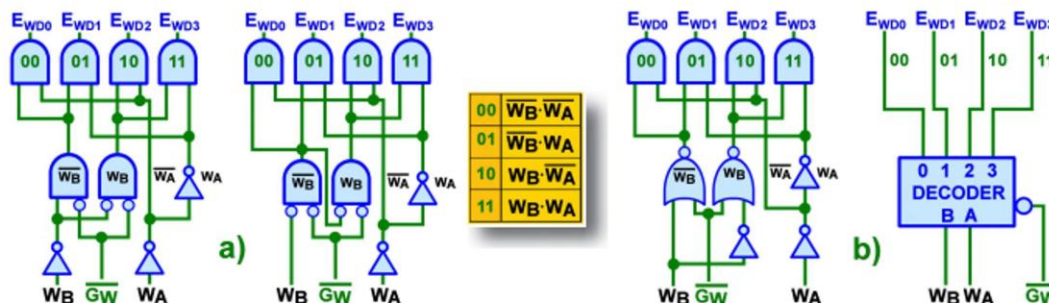


Figura 3 - 4-by-4 Register File, 74LS670: Ingressi per la scrittura

La parte circuitale destinata alla lettura è affidata ad un multiplexer "da 4 a 1" a quattro vie, una per ciascun gruppo di uscite interne Q dei 4 Latches interessati dai bit aventi lo stesso peso, nell'ambito dei Registri del Banco; la *Figura 4a* mostra il dettaglio di una delle quattro vie, in due varianti estratte da diversi datasheet; vale la pena evidenziare la presenza in forma negata dei segnali coinvolti in ingresso (le uscite dei 4 Latch): ciò è dovuto alla presenza dell'inverter su ciascuna linea d'ingresso Dato D (vedi *schema funzionale* di *Figura 2*) ed il corretto valore viene poi recuperato dall'inverter presente nella tipica forma AND-OR-NOT usata per il MUX. La *Figura 4b* ne è una buona e semplice simulazione.

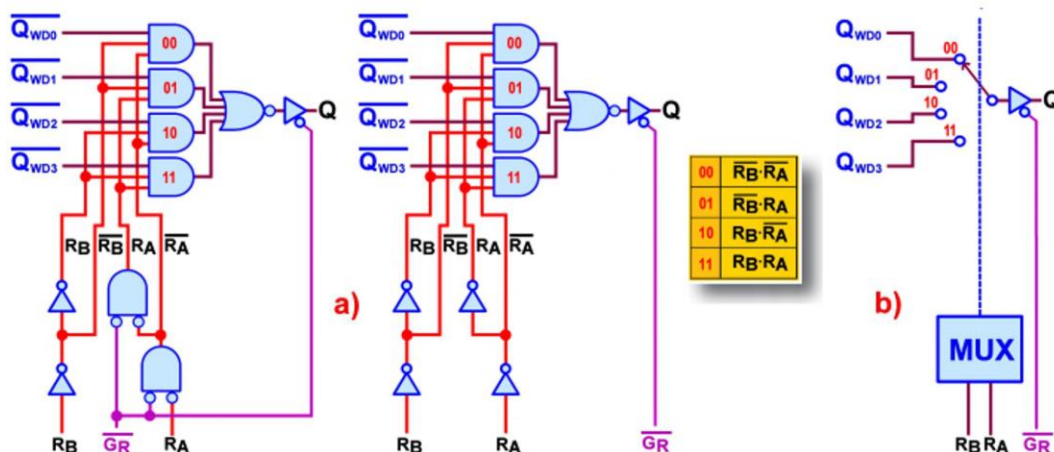


Figura 4 - 4-by-4 Register File, 74LS670: Ingressi per la lettura

Tutti i bit presenti sulle uscite interne Q dei 16 latches sono contemporaneamente presenti sugli ingressi del Multiplexer, ma solo 4 saranno trasferiti sulle uscite esterne Q del **Register File**; per la sua natura di "commutatore rotativo digitale" il MUX è chiamato a catturare simultaneamente tutti e 4 i bit del solo Registro puntato dalle sue linee di selezione, in sostanza costituite dalle linee d'indirizzo R_A e R_B , dedicate esclusivamente alla lettura e del tutto distinte da quelle utilizzate nella fase di scrittura.

In realtà la parola contenuta nel Registro indirizzato appare sulle quattro uscite esterne solo quando anche il segnale di abilitazione alla lettura, G_R , viene tenuto basso, aprendo i buffers 3-

state ad esse associati; se G_R è alto, le 4 uscite Q esterne sono inibite e passano allo stato di alta impedenza.

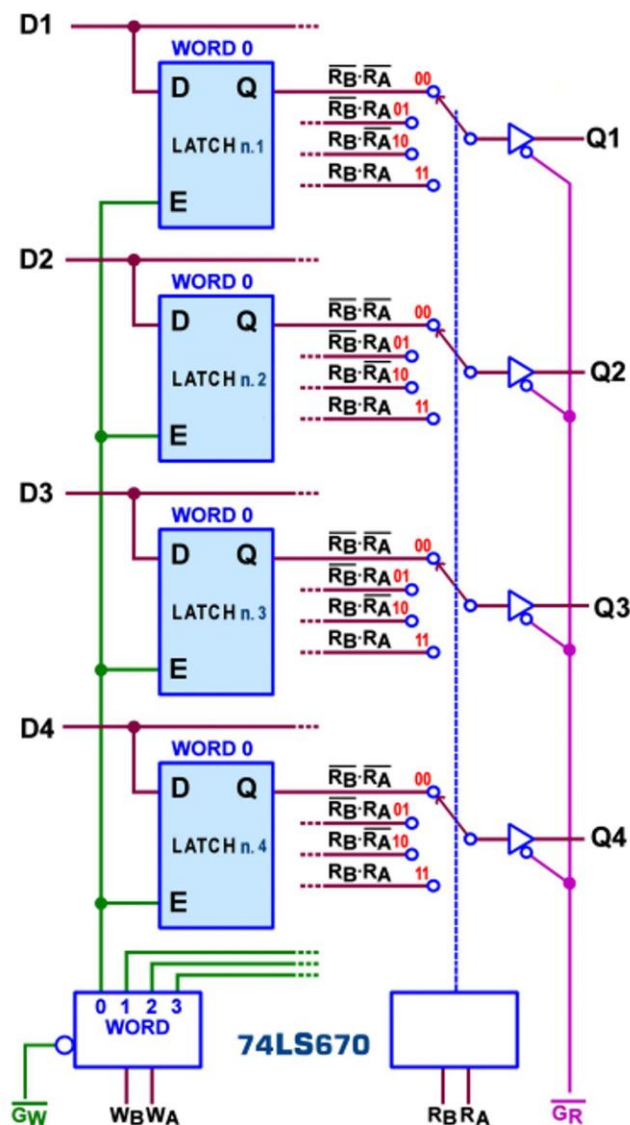


Figura 5 - 4-by-4 Register File, 74LS670: Gestione del Registro Word0

La *Figura 5* sintetizza l'azione delle linee di controllo sui 4 latches del primo Registro del Banco, chiamato a memorizzare la parola 0 (Word0); naturalmente, come detto, la fase di scrittura può avvenire mentre viene esercitata una lettura, anche da un Registro diverso da quello che si sta scrivendo; sebbene possa succedere che lo stesso Registro sia letto e scritto nello stesso istante, se i dati proposti in ingresso non sono mantenuti stabili, prima e durante dell'arrivo dell'impulso di scrittura G_W , è consigliabile (a causa della "trasparenza" dei Latches) esercitare la lettura in tempi diversi.

La *Figura 6* propone lo *schema pratico* mentre la *Figura 7* mostra la *Tabella di Verità*, articolata in due parti per sottolineare l'attitudine di questo componente a essere scritto e letto contemporaneamente; in entrambe il pedice n (in questo componente uguale da 0 a 3) fa le

vece delle linee d'indirizzo, non presenti in modo esplicito, ed associa i segnali che ne dispongono (D_n e Q_n) a tutti gli n bit del Registro indirizzato.

A corollario della descrizione notiamo che: **a)** nella fase di scrittura (*Write Mode Select*) viene sottolineata la necessità di mantenere stabili anche le linee d'indirizzo W_A e W_B mentre il segnale G_W è a basso livello; **b)** nella fase di lettura (*Read Mode Select*) si conferma che l'indirizzamento (tramite R_A e R_B) delle uscite interne Q può avvenire indipendentemente dall'azione esercitata dai segnali di abilitazione segnale G_W e G_R .

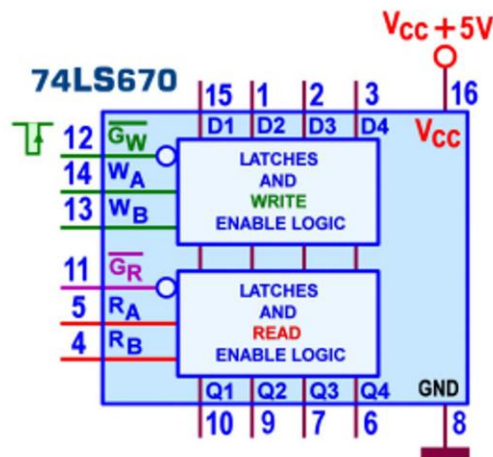


Figura 6 - 4-by-4 Register File, 74LS670: Schema pratico

WRITE MODE SELECT TABLE			
INPUTS		OUTPUT	
D_n	$\overline{G_W}$	Q_n INTERNAL LATCHES	MODO
0	0	0	Scrittura del Dato
1	0	1	
X	1	non cambia	Memorizzazione

READ MODE SELECT TABLE			
INPUTS		OUTPUT	
Q_n INTERNAL LATCHES	$\overline{G_R}$	Q_n	MODO
0	0	0	Lettura del Dato
1	0	1	
X	1	Hi-Z	Alta Impedenza

Figura 7 - 4-by-4 Register File, 74LS670: Tabella di Verità

L'estrema versatilità di questo componente permette di coinvolgerne più di uno "in batteria", al fine di garantire la disponibilità di Registri contenenti parole più grandi (portando la loro dimensione a 8 o 12 o 16 bit, comunque multipla di 4) oppure per aumentare il numero di Registri nel Banco (portandolo a 8 o 12 o 16 bit, sempre multiplo di 4), o per l'una e l'altra opportunità.

In *Figura 8* è disponibile il *simbolo logico* predisposto dallo *standard IEEE*; i segnali presenti nel blocco di controllo comune (posto in alto, nel simbolo) sono quelli che si ritengono tali da influenzare tutti gli altri elementi presenti nel dispositivo: **a)** la prima notazione raccoglie i 2 ingressi di selezione A e B del *decoder* (quelli d'indirizzo in scrittura) in una parentesi graffa,

che li fa corrispondere al simbolo **1A** per identificare l'etichetta dei segnali (appartenenti alla serie dei blocchi sottostanti) da essi influenzati (cioè gli ingressi di dato D_n); il numero **0** e **1** indica il *peso* del bit nell'ambito della parola d'indirizzo, e la grafica $0/(2^n-1)$ alla destra della grafa non è una frazione ma indica l'intervallo previsto per gli ingressi sotto controllo, nel nostro esempio da 0 a 3; **b**) la seconda notazione è analoga alla prima ma rappresenta ora i 2 ingressi di selezione A e B del *multiplexer* (quelli d'indirizzo in lettura): il simbolo **2A** identifica l'etichetta dei segnali da essi influenzati (ora le uscite Q_n) mentre la "frazione" alla destra della grafa ha la stessa funzione di prima; **c**) poiché i segnali di abilitazione (indicati con **C4** e con **EN**) sono unici per tutte le quattro sezioni sottostanti vengono inclusi nella metà superiore (altrimenti ciascuna sezione avrebbe i propri).

La parte inferiore del simbolo è composta da 4 blocchi (uno per **D latch**), identici tra loro e per i quali solo il primo deve essere redatto con dettaglio; nel nostro caso esso contiene: **a**) l'etichetta **1A,4D** chiaramente riferita ai Dati in ingresso, con davanti il numero **1A** per far capire a quale segnale del blocco di controllo è da ritenersi correlata (appunto quello del *decoder*); **b**) l'etichetta **2A**, in prossimità delle uscite Q, per correlarle al blocco di controllo *multiplexer*; **c**) un triangolo rivolto verso il basso, per indicare che le uscite sono 3-state e che l'azione del comando in relazione con esse sarà quella di abilitare la disponibilità dei dati, oppure di disabilitarle in blocco (cioè di porle nello stato di alta impedenza).

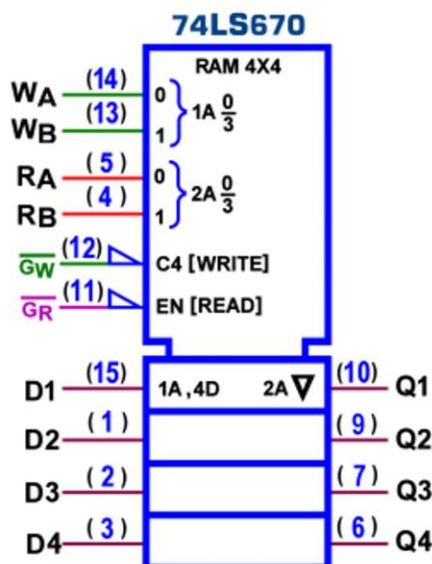


Figura 8 - 4-by-4 Register File, 74LS670: Simbolo logico ANSI/IEEE Std. 91-1984

La potenza massima dissipata dal **74LS640** è di **250 mW**; la corrente erogata tipica in uscita (I_{OH} , con $Q=1$) e di **2,6 mA** mentre quella assorbita massima in uscita (I_{OL} , con $Q=0$) è di ben **24 mA**; in regime dinamico (con carico di **45pF/667ohm**) i ritardi di propagazione massimi, per entrambe le transizioni t_{PLH} e t_{PHL} e per ogni ingresso (dati, selezione in lettura e abilitazione in scrittura) rispetto ad ogni uscita Q, sono mediamente di **45 ns**; il valore minimo di *Setup Time*= $t_s=10$ ns e quello di *Hold Time*= $t_h=15$ ns. In questo caso viene fornito anche il tempo di latch ($t_{LATCH}=25$ ns, massimo), cioè il tempo concesso all'uscita interna Q del latch per assumere il valore di nuovi dati; esso è importante solo quando si cerca di leggere da una locazione immediatamente dopo la sua scrittura.

Del Banco di Registri **74LS670** (4-By-4 Register Files) esiste una versione del tutto identica, il **74LS170**, ma con uscite di tipo Open-Collector; tutte le considerazioni appena fatte rimangono dunque valide, compresi gli schemi, ovviamente con le piccole differenze dovute alla nuova natura delle uscite: per esempio nello *schema funzionale* di *Figura 2* mancheranno i 4 buffer 3-state e il segnale di abilitazione alla lettura G_R si limiterà a tenere aperte le due AND che governano l'indirizzo R_A .

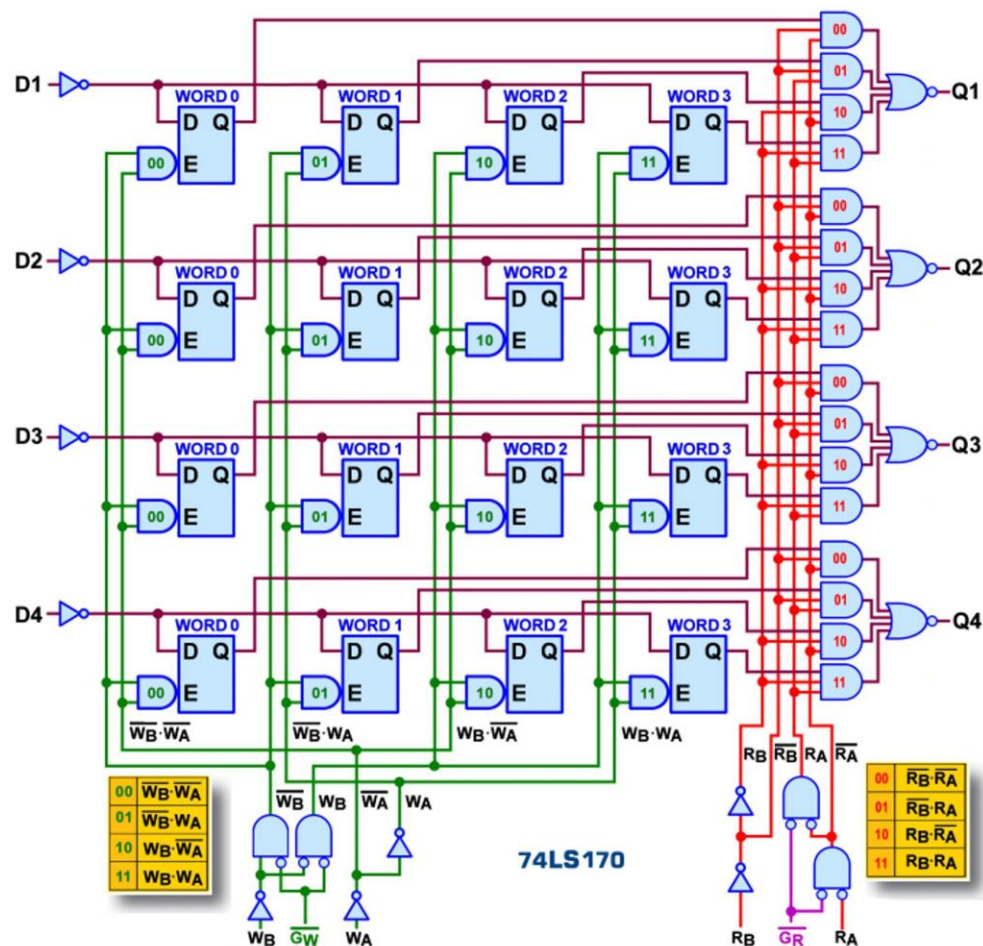


Figura 2 bis - 4-by-4 Register File, 74LS170: Schema funzionale

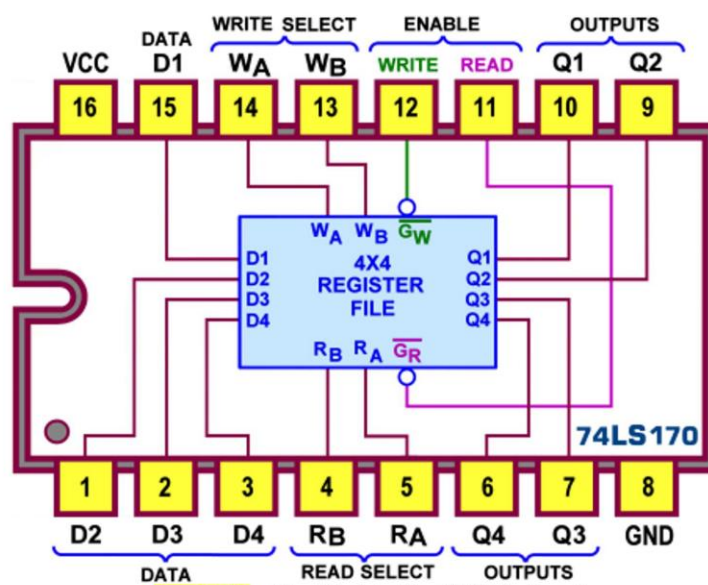


Figura 1 bis - 4-by-4 Register File, 74LS170: Pin-out

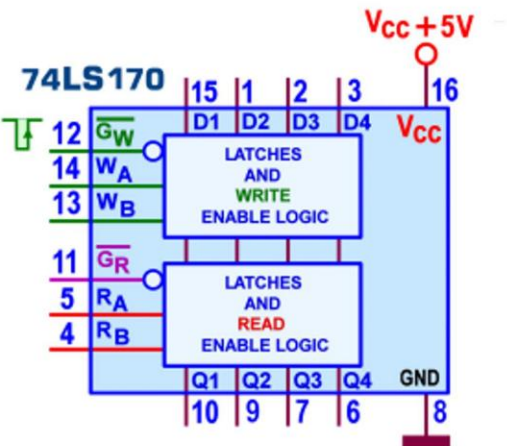


Figura 6 bis - 4-by-4 Register File, 74LS170: Schema pratico

WRITE MODE SELECT TABLE

INPUTS		OUTPUT	
D_n	$\overline{G_W}$	Q_n <small>INTERNAL LATCHES</small>	MODO
0	0	0	Scrittura del Dato
1	0	1	
X	1	non cambia	Memorizzazione

READ MODE SELECT TABLE

INPUTS		OUTPUT	
Q_n <small>INTERNAL LATCHES</small>	$\overline{G_R}$	Q_n	MODO
0	0	0	Uscite
1	0	1	Lettura del Dato
X	1	1	Uscite disabilitate

Figura 7 bis - 4-by-4 Register File, 74LS170: Tabella di Verità

Anche il *simbolo logico* predisposto dallo *standard IEEE* dovrà contenere un dettaglio differente: nella parte inferiore del simbolo (quella dei 4 blocchi che rappresentano i **D latch** del dispositivo) sull'estrema destra del primo apparirà ora un piccolo rombo, destinato a giustificare le uscite open collector, al posto del "triangolo rivolto verso il basso", tipico delle uscite sono 3-state.

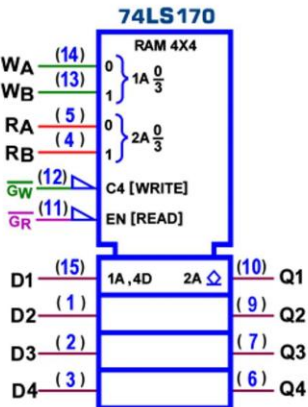
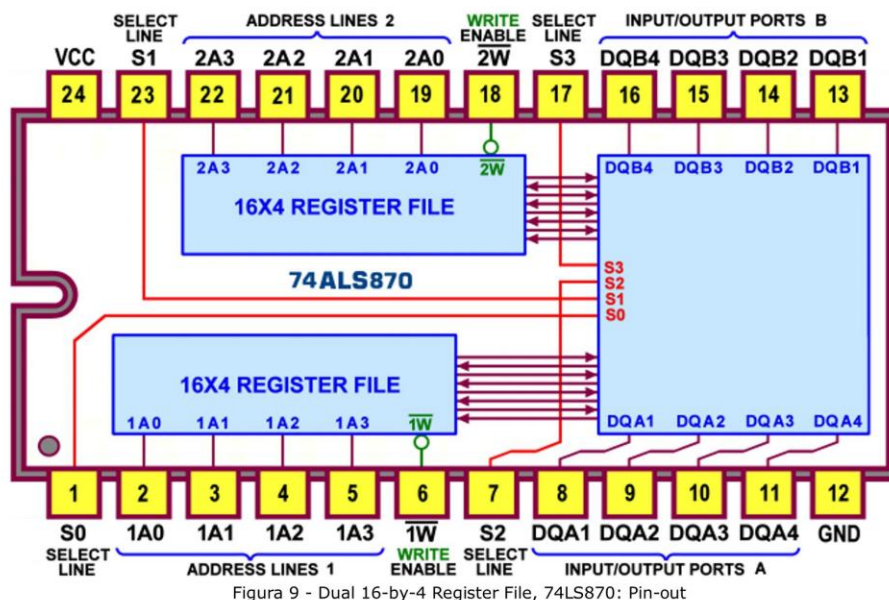


Figura 8 bis - 4-by-4 Register File, 74LS170: Simbolo logico ANSI/IEEE Std. 91-1984

La potenza massima dissipata dal **74LS170** è di **200 mW** e la corrente assorbita in uscita (I_{OL} , con $Q=0$) può essere al massimo di **8 mA**; il comportamento in regime dinamico (con carico di **15pF/2kohm**) è di fatto lo stesso di quello con uscite 3-state.

In regime dinamico i ritardi di propagazione massimi, per entrambe le transizioni, t_{PLH} e t_{PHL} , e per ogni ingresso (dati, selezione in lettura e abilitazione in scrittura) rispetto ad ogni uscita Q , sono mediamente dai **30 ns** ai **45 ns**; il valore minimo di *Setup Time* $=t_s=10$ ns e quello di *Hold Time* $=t_H=15$ ns. In questo caso viene fornito anche il tempo di latch ($t_{LATCH}=25$ ns, massimo), cioè il tempo concesso all'uscita interna Q del latch per assumere il valore di nuovi dati; esso è importante solo quando si cerca di leggere da una locazione immediatamente dopo la sua scrittura.

La serie TTL (per altro nella sua versione avanzata, **ALS**, *Advanced Low Power Schottky*) prevede un altro componente di questo tipo, molto sofisticato: si tratta del **74ALS870** (un **Dual 16-By-4 Register File with 3-state outputs**) dotato 2 Banchi, ciascuno organizzato con 16 Registri a 4 bit; ogni Registro dispone delle proprie 4 linee d'indirizzo (da A0 a A3, decodificate da un decoder interno da 4 a 16, funzionalmente simile al 74LS154, per poter puntare *ad accesso casuale* le $2^4=16$ locazioni) e della propria linea di abilitazione alla scrittura (W , *Write Enable*, attiva bassa); la Figura 9 mostra il suo pin-out.



Questo dispositivo è poi dotato di due porte bidirezionali di input/output a 4 bit (DQA1-DQA4 e DQB1-DQB4) e di una complicata rete combinatoria che ha il compito di organizzare il trasferimento dell'informazione in transito su di esse dalle uscite interne o verso gli ingressi interni dei 2 Banchi di Registri (*Registers File*), in funzione del valore assunto da ulteriori 4 linee di controllo, da S0 a S3.

In sostanza il componente è progettato specificamente per Architetture Multibus: di fatto le due porte sono collegate a due distinti bus, e sono in grado di pilotarli direttamente, disponendo di uscite bufferizzate di tipo 3-state; la destinazione di ciascuna possibile *parola a 4 bit* dipende dal valore del codice binario a 4 bit presente sulle linee di selezione: **a)** potrà essere trasmessa in uscita su uno dei 2 *bus dati*, dopo essere stata letta da uno dei Registri; **b)** potrà essere ricevuta da uno dei *bus dati* e poi scritta in uno dei Registri; **c)** potrà essere ricevuta in ingresso da uno dei due bus A e spedita in uscita sull'altro bus, e viceversa.

Per la numerosa quantità di porte logiche "di contorno" coinvolte (in aggiunta ai 32 Latches di tipo D, necessari per i 2 Registers File) lo *schema funzionale* è decisamente troppo complesso;

anche i datasheet si limitano a fornirne una versione molto schematica, che rischia di rimanere piuttosto oscura e di difficile comprensione: in questo caso il suo disegno diventa controproducente e per sintetizzarne il funzionamento risulta più convincente la Tabella di Verità, proposta in *Figura 10*.

REGISTERS FILE SELECT				INPUT/OUTPUT SELECT	
S0	S1	Selezione del Registro coinvolto	S2	S3	Flusso Dati sul Bus
0	0	Registro1 verso A, Registro1 verso B	0	0	Porta A in OUTPUT Porta B in OUTPUT
1	0	Registro2 verso A, Registro1 verso B			
0	1	Registro1 verso A, Registro2 verso B			
1	1	Registro2 verso A, Registro2 verso B			
0	0	A verso Registro1, Registro1 verso B	1	0	Porta A in INPUT Porta B in OUTPUT
1	0	A verso Registro2, Registro1 verso B			
0	1	A verso Registro1, Registro2 verso B			
1	1	A verso Registro2, Registro2 verso B			
0	0	Registro1 verso A, B verso Registro1	0	1	Porta A in OUTPUT Porta B in INPUT
1	0	Registro2 verso A, B verso Registro1			
0	1	Registro1 verso A, B verso Registro2			
1	1	Registro2 verso A, B verso Registro2			
0	0	B verso Registro1	1	1	Porta A in INPUT Porta B in INPUT
1	0	A verso Registro2, B verso Registro1			
0	1	A verso Registro1, B verso Registro2			
1	1	B verso Registro2			

Figura 10 - Dual 16-by-4 Register File, 74LS870: Tabella di Verità

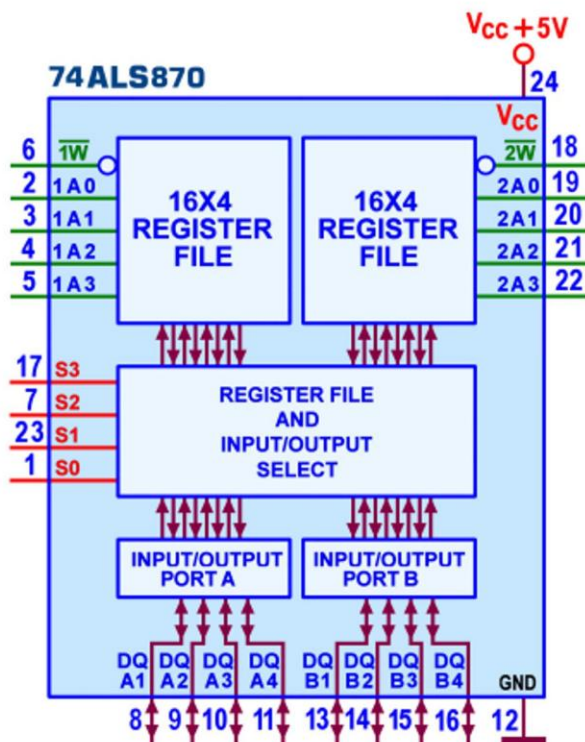


Figura 11 - Dual 16-by-4 Register File, 74LS870: Schema Pratico

Da essa si capisce che due delle linee di selezione, S0 e S1, hanno il compito di stabilire quale delle 2 Porte di I/O e a quale dei 2 Banchi di Registri appartiene la locazione indirizzata che verrà messa in contatto con la porta, al fine di scambiarsi il dato a 4 bit: il flusso dei dati potrà avvenire in quattro modi diversi, tante quante sono le codifiche binarie possibili sulle 2 linee S0 e S1.

La linea S2 stabilisce se la **porta A** di I/O è programmata in ingresso (S2=1) o in uscita (S2=0) e la linea S3 fa lo stesso servizio per la **porta B** di I/O, programmata in ingresso (S3=1) o in uscita (S3=0).

Osservando le ultime quattro righe della tabella, la prima e l'ultima sembrano incomplete; a stretto rigor di logica la combinazione S0,S1=00 dovrebbe selezionare "A verso Registro1, B verso Registro1" ma la cosa è fisicamente intollerabile; la stessa considerazione vale per la combinazione S0,S1=11 che dovrebbe orientare entrambe le porte A e B verso lo stesso Registro2. Per evitare questo conflitto in scrittura, tipico solo durante la modalità a doppio ingresso, le specifiche del dispositivo prevedono che la Porta d'ingresso B abbia la priorità sulla Porta d'ingresso A.

In Figura 11 è disponibile lo *schema pratico*, utile per l'impiego diretto nel contesto di un progetto.

La potenza massima dissipata dal **74ALS870** è di **550 mW** e la corrente massima in uscita è tipicamente pari a **2,6 mA** (erogata, I_{OH} con $Q=1$) e di **24 mA** (assorbita, I_{OL} con $Q=0$), decisamente degna di nota; il valore minimo di *Setup Time* prima del sincronismo di scrittura, t_s , va da **5 ns** (indirizzi) a **15 ns** (dati), passando per i **12 ns** (linee di selezione); quello di *Hold Time*, t_h , è sostanzialmente nullo; i tempi d'accesso massimi, per ogni ingresso (dati, selezione, indirizzi) rispetto ad ogni uscita DQ, sono mediamente di **16 ns**.

Il componente **74ALS871** è una versione a 28 pin del precedente, funzionalmente identico ma con 4 piedini in più, per consentire una migliore disponibilità dei segnali per la porta A, ora gestibile in modo unidirezionale, separatamente per l'ingresso (DA1 - DA4) e per l'uscita (QA1 - QA4); l'altra porta B rimane bidirezionale (DQB1 - DQB4) e comunque prioritaria sulla porta A, quando entrambe sono programmate in ingresso; la Figura 12 mostra il suo pin-out e la Figura 13 il suo *schema pratico*.

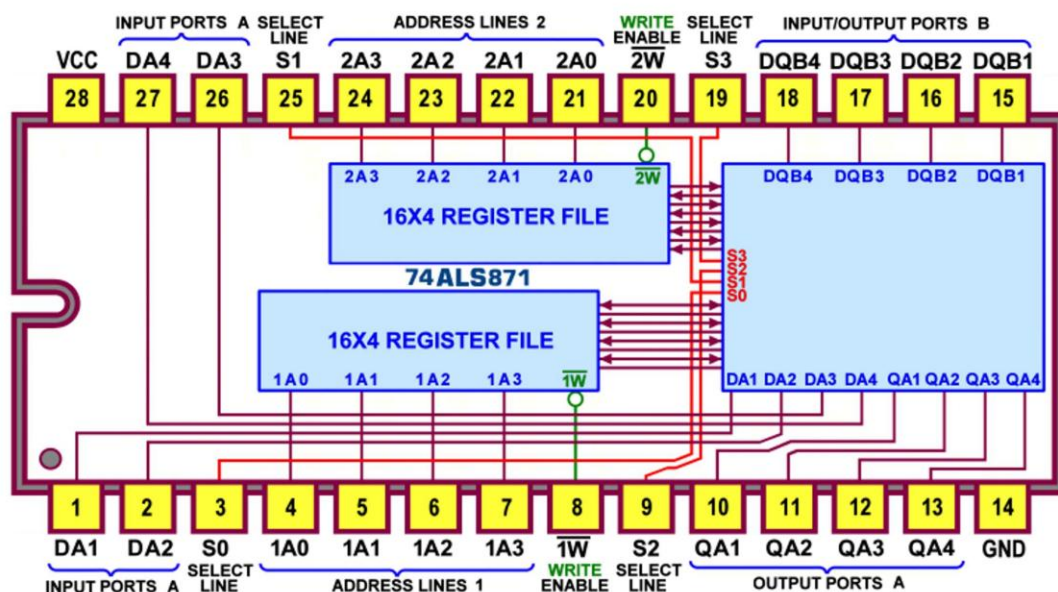


Figura 12 - Dual 16-by-4 Register File, 74ALS871: Pin-out

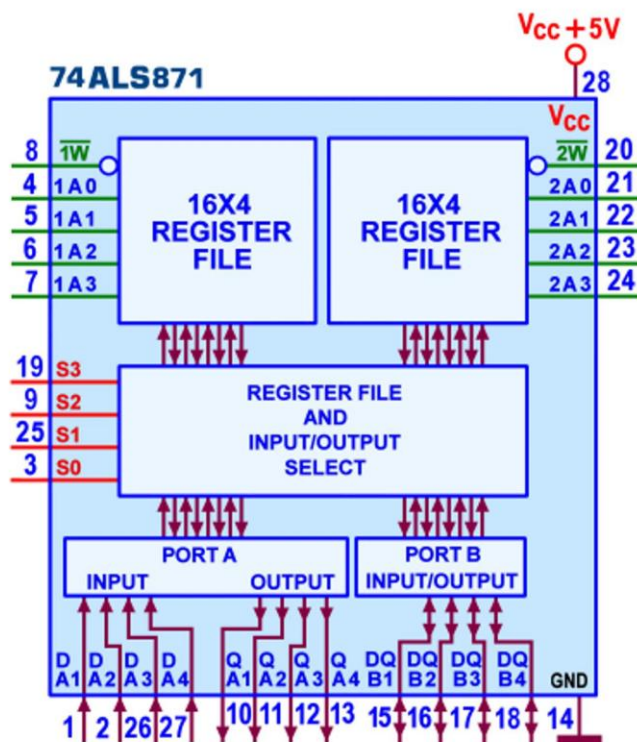


Figura 13 - Dual 16-by-4 Register File, 74LS871: Schema Pratico

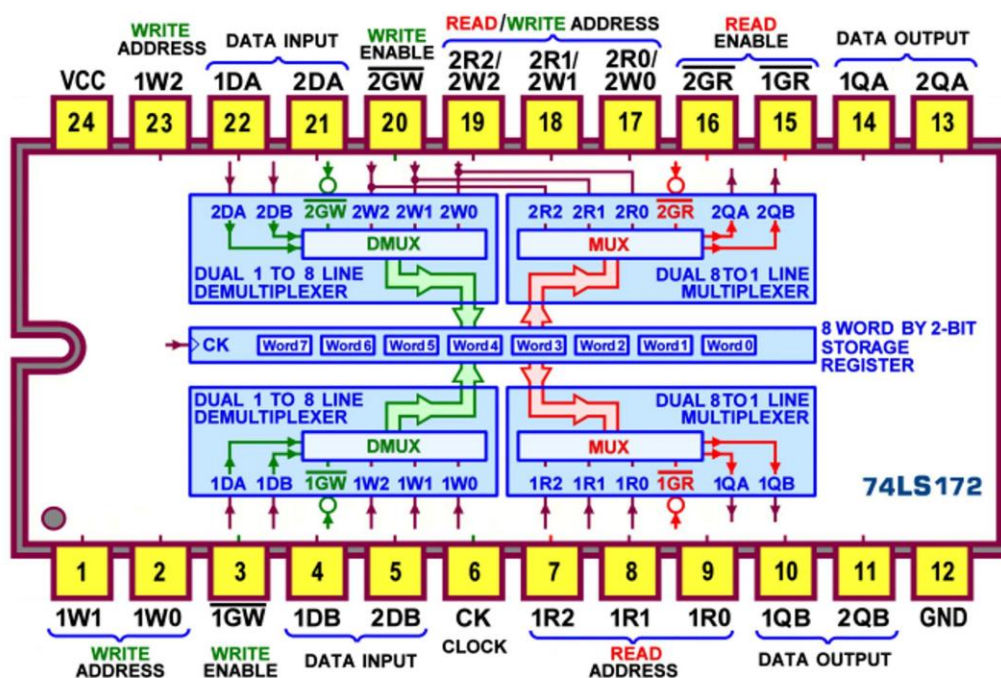


Figura 14 - 16 bit Multiple Port Register File, 74LS172: Pin-out

L'ultimo componente della categoria è ancora più particolare; si tratta del **74LS172**, definito **16 bit Multiple Port Register File with 3-state outputs**; esso dispone di due sezioni identiche ciascuna in grado di gestire le medesime 8 parole da 2 bit della piccola memoria, simultaneamente sia in lettura che in scrittura; ogni sezione dispone infatti di un proprio apparato separato per la lettura e per la scrittura, indipendente ed autonomo rispetto ad ogni altro.

La *Figura 14* mostra il suo complesso pin-out.

La presenza di una linea di Clock sottolinea che, in questo caso, si tratta a tutti gli effetti di un dispositivo assimilabile ad una **Ram 8x2**, realizzato cioè con 16 **Flop-Flop D-Type**; questo sincronismo è in comune ad entrambe le sezioni e provvede a memorizzare i nuovi dati nelle locazioni indirizzate sul suo fronte di salita, comunque se contemporaneamente la rispettiva linea di abilitazione alla scrittura, 1GW e/o 2GW, è bassa.

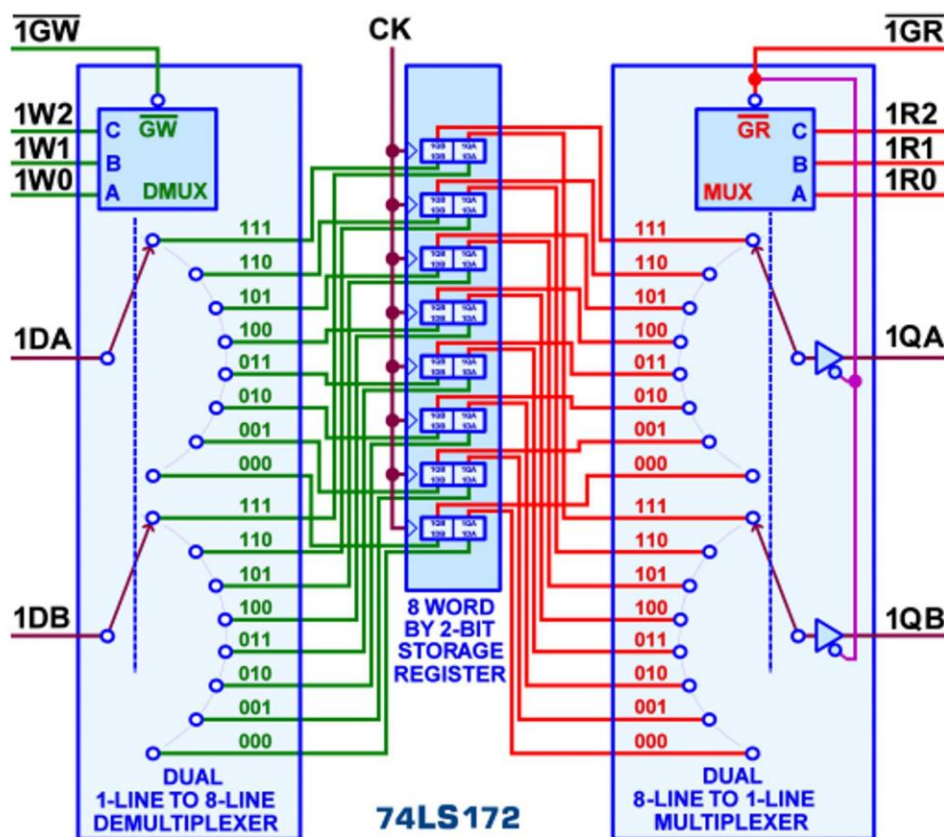


Figura 15 - 16 bit Multiple Port Register File, 74LS172: Dettaglio Funzionale

La *Figura 15* espande in dettaglio una delle due sezioni (per altro uguali tra loro) che concorrono sulle 16 locazioni del Banco degli 8 Registri a 2 bit, evidenziandone il sofisticato funzionamento; per esempio ci riferiamo alla prima delle due sezioni per cui tutti i segnali avranno il prefisso "1" (va da sé che ogni descrizione vale anche per l'altra sezione, sostituendo "2" al posto di detto prefisso): **a)** i dati in ingresso esterni, 1DA, 1DB, sono forniti alle due linee di input di un *doppio demultiplexer* "da 1 a 8" e da questo presentati sulle linee interne D dei 2 Flip-Flops corrispondenti ai due bit della parola indirizzata dalle linee 1W0, 1W1, 1W2; se l'abilitazione alla scrittura 1GW è bassa questa informazione viene memorizzata sulla transizione positiva dell'ingresso di clock; **b)** se l'abilitazione alla lettura 1GR è bassa, con l'aiuto di un *doppio multiplexer* "da 8 a 1", il contenuto di una delle 8 parole del

Banco di Registri potrà essere letto (anche in concomitanza con l'eventuale fase di scrittura) sulle uscite 1QA,1QB, che assumeranno i livelli logici memorizzati sulle linee d'uscita interne Q dei 2 Flip-Flops corrispondenti ai due bit della parola indirizzata dalle linee 1R0,1R1,1R2; se l'abilitazione alla lettura 1GR è alta, le uscite associate alle locazioni indirizzate si trovano nello stato di alta impedenza.

Poiché le due sezioni sono indipendenti, può succedere che entrambe stiano puntando (rispettivamente con 1W0,1W1,1W2 e con 2W0,2W1,2W2) la medesima word (Registro a 2 bit), cercando di memorizzare in essa l'informazione presente sui propri ingressi di dato (rispettivamente 1DA,1DB e 2DA,2DB); se si verifica che l'informazione non è la stessa per entrambe le sezioni (cioè 1DA diverso da 2DA e/o 1DB diverso da 2DB) in ogni bit da memorizzare sarà predominante il dato a livello basso.

Da notare che, per evidenti motivi di spazio (il chip ha solo 24 piedini ..), nella seconda sezione le linee d'indirizzo utilizzate per la scrittura e per la lettura sono le stesse.

Per ogni ingresso il valore minimo t_s di *Setup Time* è mediamente di **35** ns e quello t_h di *Hold Time* è nullo; il tempo d'accesso tipico è di **15** ns in lettura e di **33** ns in scrittura.