

Per l'implementazione sono stati seguiti i seguenti passi:

1. costruzione modulo complementatore;
2. costruzione sommatore/sottrattore;
3. costruzione moltiplicatore con 4+1 bit in ingresso, 8+1 bit in uscita;
4. inglobare i moduli precedenti nell' Unità Aritmetico-Logica;
5. costruzione registro a 8 bit + segno ;
6. inglobare i moduli precedenti nel Microprocessore;
7. costruzione modulo R.A.M.;
8. codifica delle istruzioni;
9. test del mini-computer tramite switch, tastiere e display.

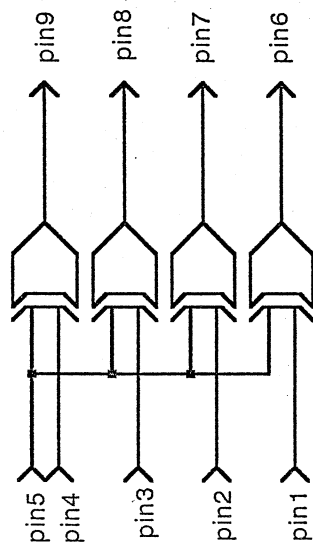
## 1. COMPLEMENTATORE

E' un modulo in grado di invertire il valore dei bit di dato se il segno corrispondente è negativo (pin5 =1), oppure mantenerlo inalterato se il segno è positivo.

A questo scopo sono sufficienti 4 porte XOR, come visibile nello schema elettrico allegato.

<b>Valori logici presenti in uscita in base al valore assunto dal pin5:</b>		
<b>pin5</b>	<b>0</b>	<b>1</b>
pin6	pin1	pin1*
pin7	pin2	pin2*
pin8	pin3	pin3*
pin9	pin4	pin4*

# compl. 4-bit



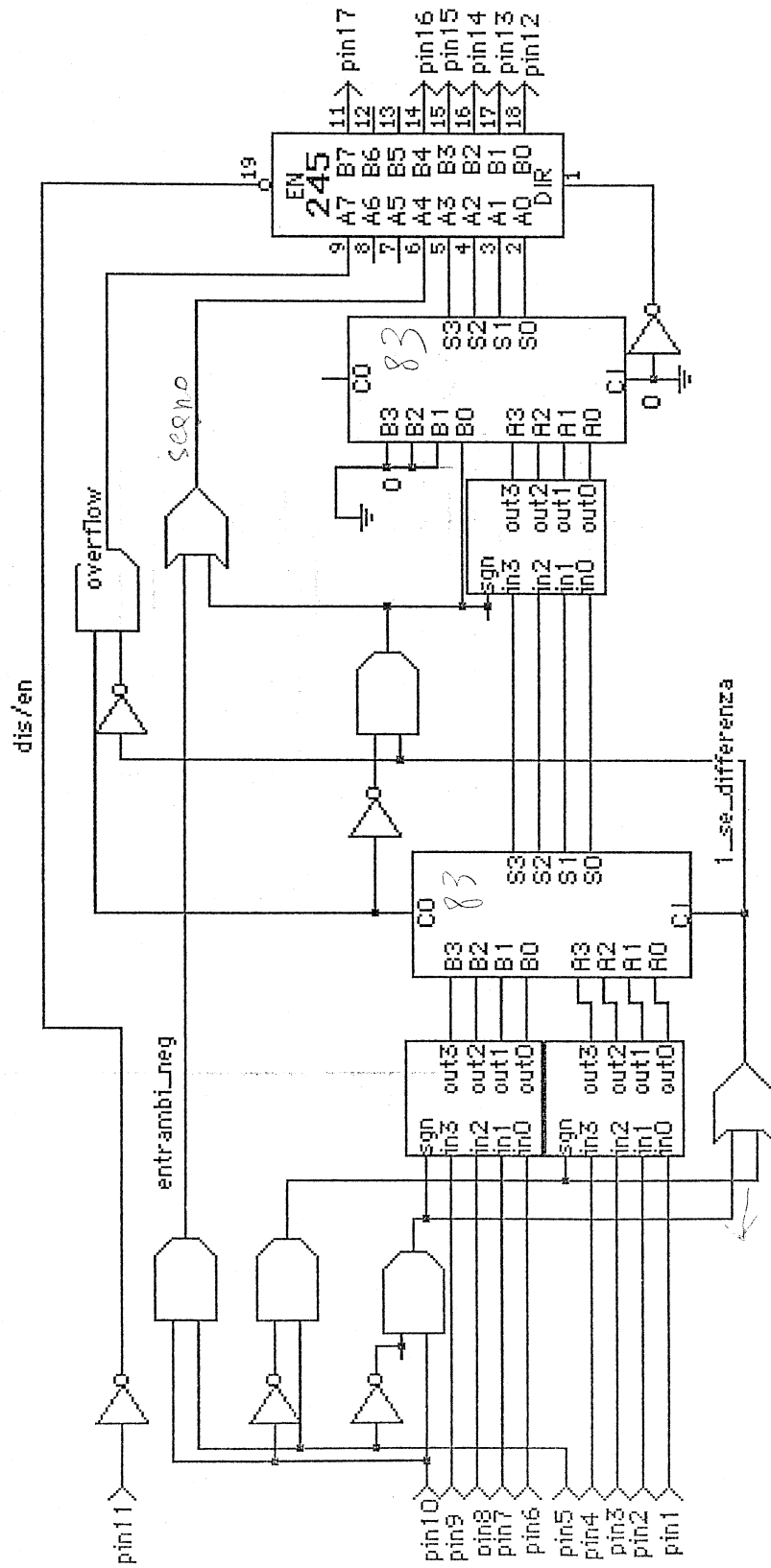
## 2. SOMMATORE/SOTTRATTORE

Una logica costituita da 3 porte AND verifica se si tratta di una differenza (numero positivo + numero negativo); in questo caso viene complementato il numero negativo, sommato all'altro e incrementato di 1.

L'uscita del full-adder in certi casi deve essere nuovamente complementata e incrementata (per l'incremento viene usato un altro full-adder, ma si potrebbe creare una logica adeguata).

Il buffer tipo 74LS245 porta l'uscita sul BUS in 3-state.

# somma/diff 4-bit



### 3. MOLTIPLICATORE

Per eseguire la moltiplicazione  $n \times m$  viene fatta la somma di  $m$  volte su  $n$ .

Sono stati utilizzati 2 circuiti integrati tipo 74LS83 in quanto la moltiplicazione di 2 numeri a 4 bit da come risultato un numero a 8 bit (senza overflow).

Il numero di somme da eseguire è gestito da un 74LS193 settato come Down-counter, che possiede già all'interno il modulo di ZERO-DETECT (segnale BOR).

Il segno del risultato è ottenuto da una porta XOR che ha la seguente tabella di verità:

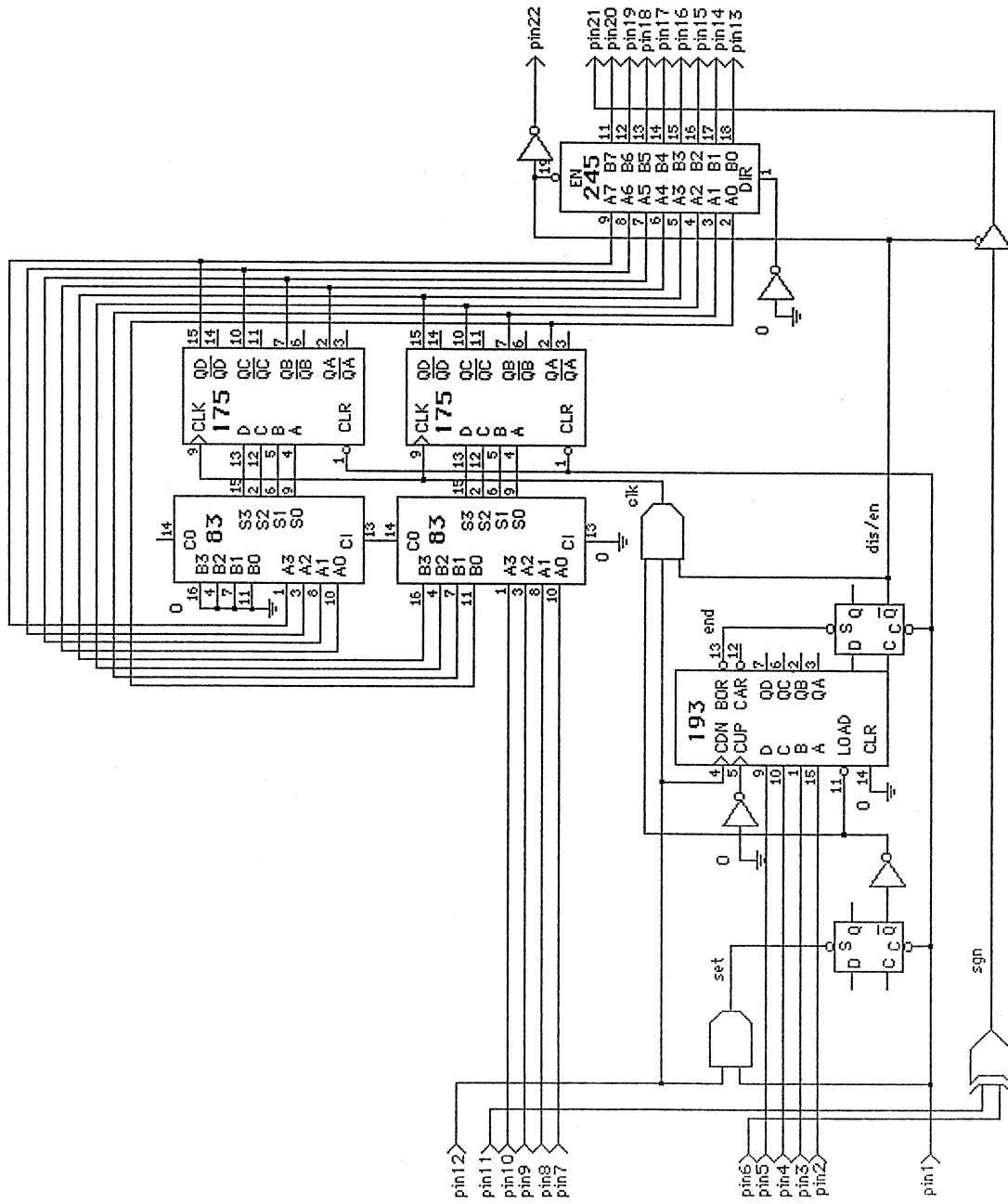
IN	OUT
11	0
01	1
10	1
00	0

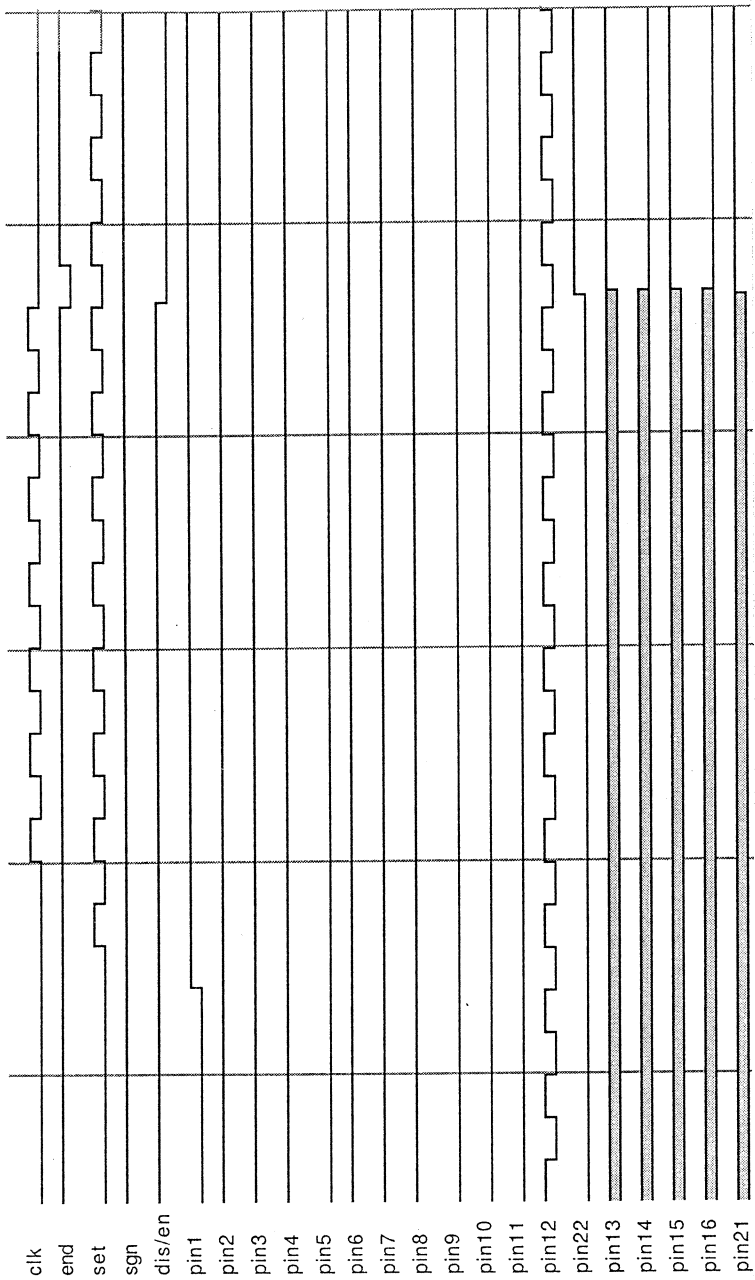
Il pin 1 è il segnale di enable/clear\*;

il pin 12 è il segnale di clock che verrà fornito dall'esterno del modulo.

All'esterno viene anche portato il segnale di END (pin 22) che sarà utile successivamente perchè la durata dell'operazione di moltiplicazione dipende dai numeri da moltiplicare.

# moltiplicatore parallelo







## 4. A.L.U.

Queste sono le operazioni consentite:

00 - non fa niente

01 - tramite la porta XOR viene abilitata la somma tra i due numeri

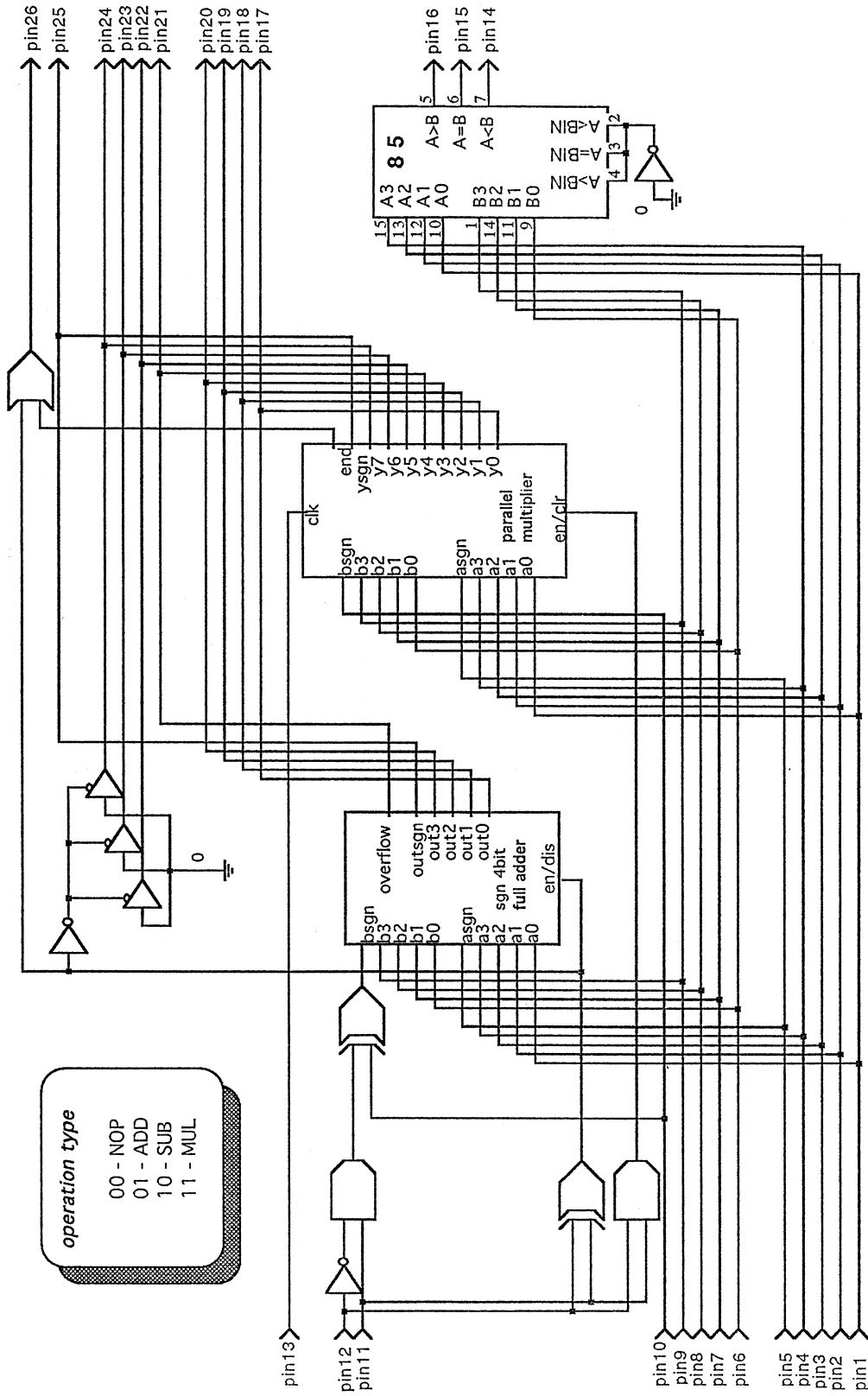
10 - viene abilitata la somma tra il 1° numero ed il 2° invertito di segno

11 - abilita il modulo moltiplicatore che fornisce in uscita 8 bit + 1 senza overflow.

Il bit di overflow della somma è utilizzato come quinto bit di dati sul BUS; gli altri 3 bit in questo caso sono posti a massa.

Il modulo comparatore 74LS85 è sempre attivo e fornisce in uscita 3 bit di stato che possono essere utilizzati in futuro per la costruzione di particolari istruzioni di test condizionali.

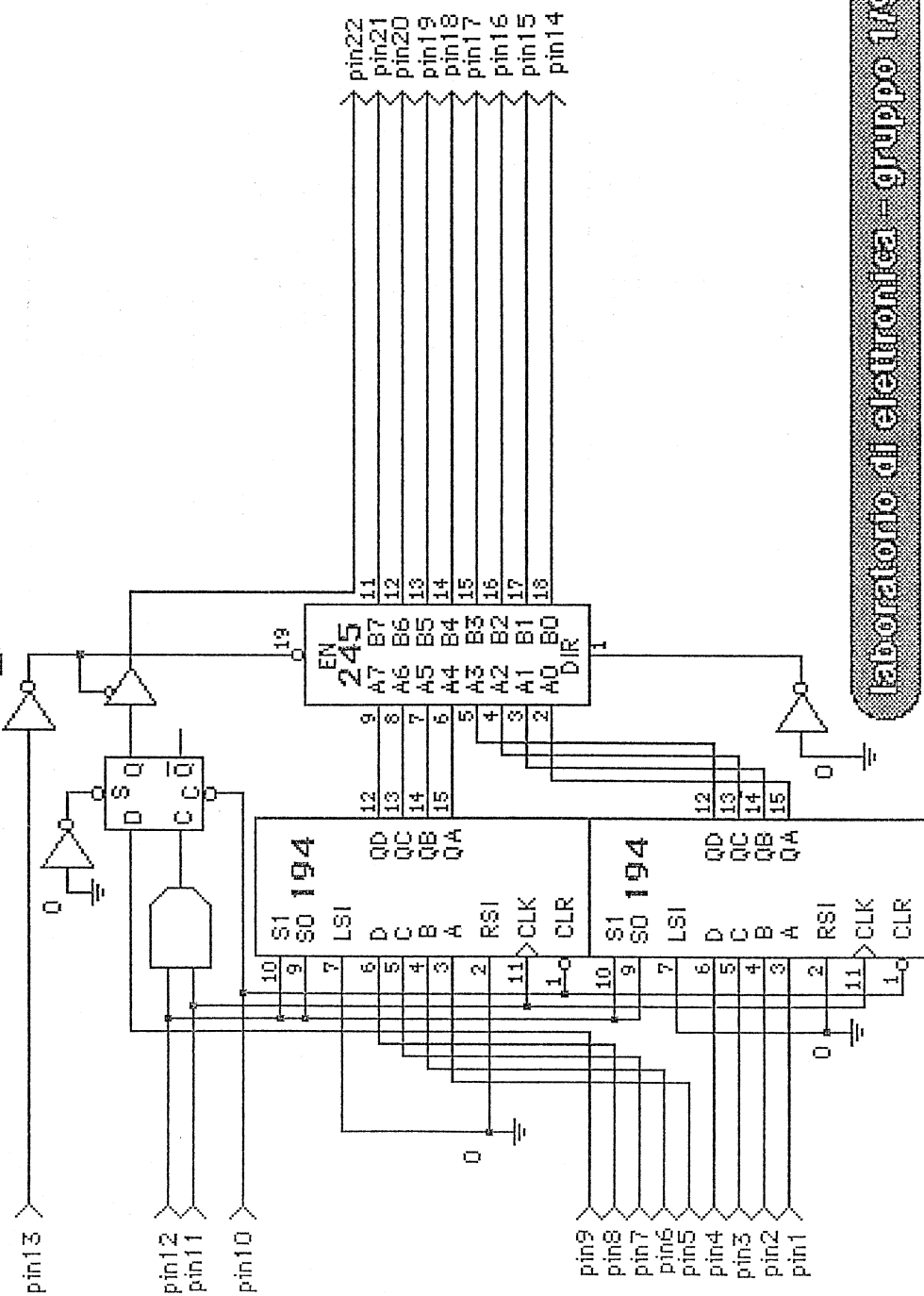
# A.L.U.



## 5. REGISTO 9-bit

È costituito da 2 integrati 74LS194 che contengono entrambi 4 flip-flop, + 1 flip-flop esterno per memorizzare il dato, e un buffer 74LS245 per presentare il dato memorizzato precedentemente. Gli ingressi LSI, RSI sono posti a massa (disabilitati), in quanto in questa versione semplificata, non vengono previsti SHIFT a sinistra o destra sul dato nel registro.

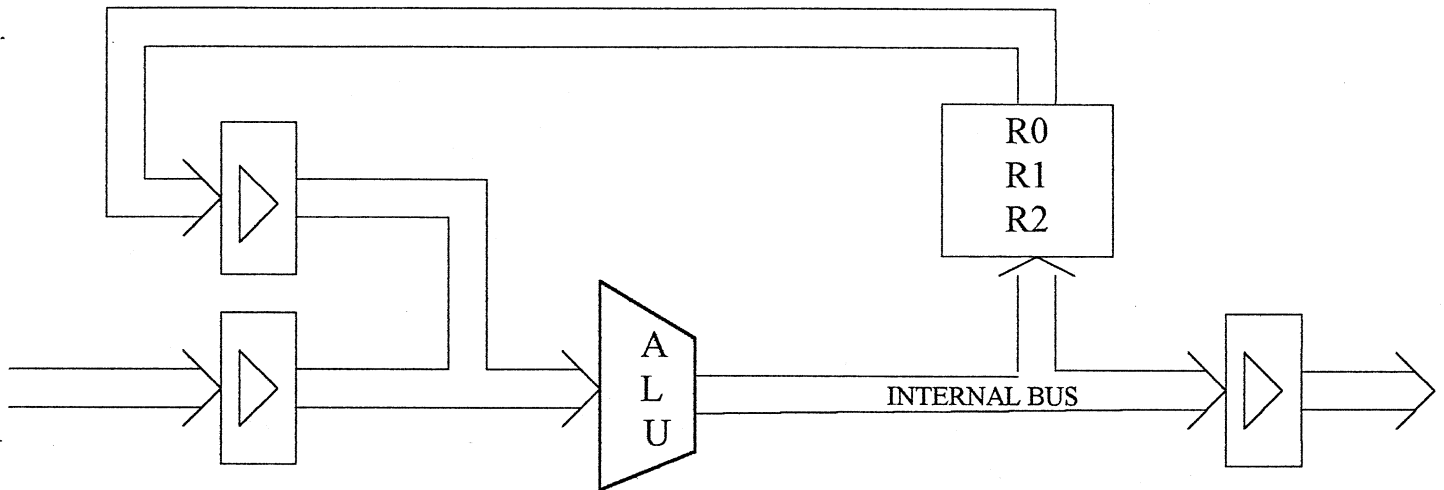
# registro 9-bit



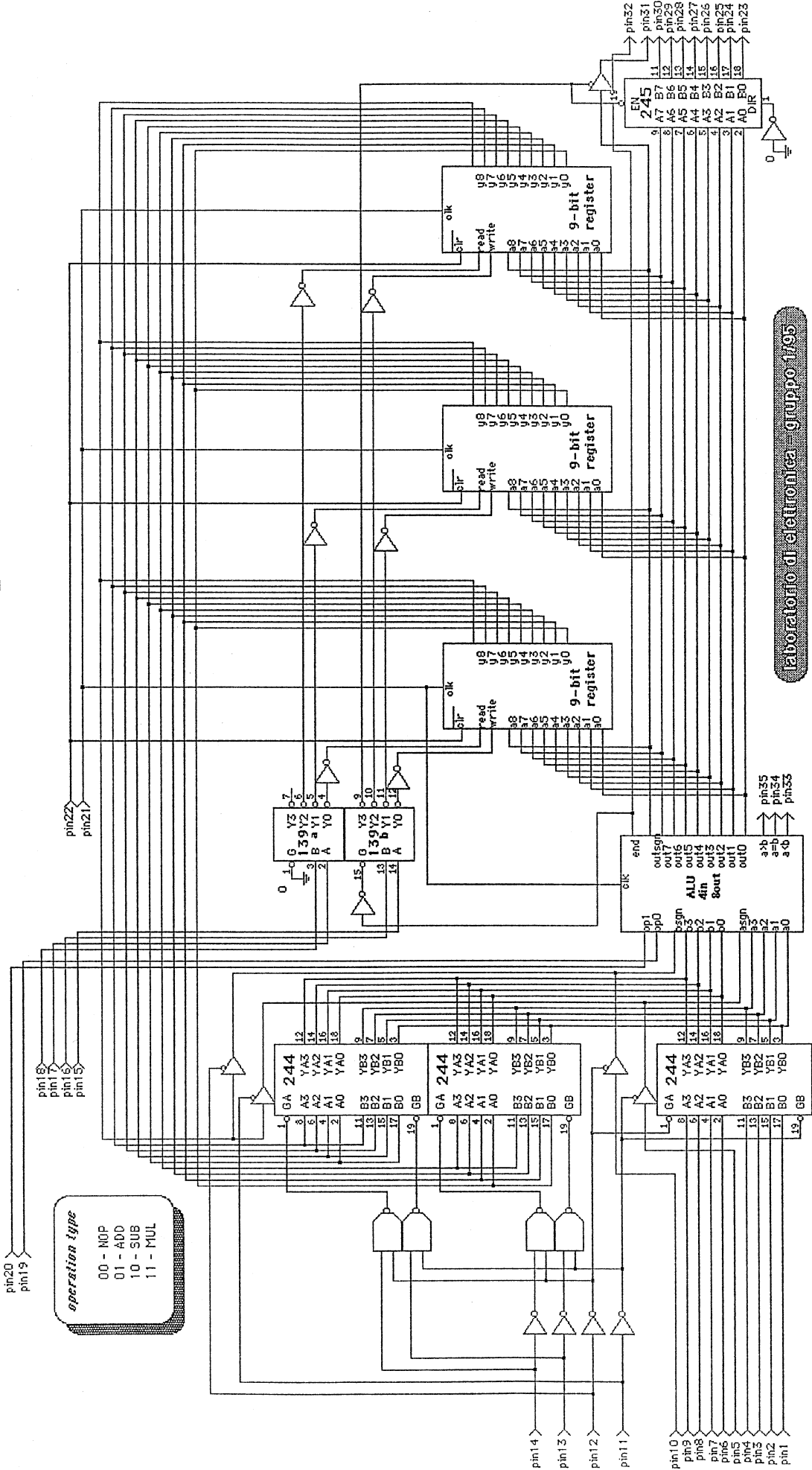
laboratorio di elettronica - gruppo 1195

## 6. MICRO 1/95

Sono previsti 3 registri interni al microprocessore, su cui memorizzare i dati temporaneamente, e dei buffers che permettono di selezionare il tipo di ingressi alla ALU se da registro o esterni da tastiera.



# micro 1/95



**operation type**

- 00 - NOP
- 01 - ADD
- 10 - SUB
- 11 - MUL

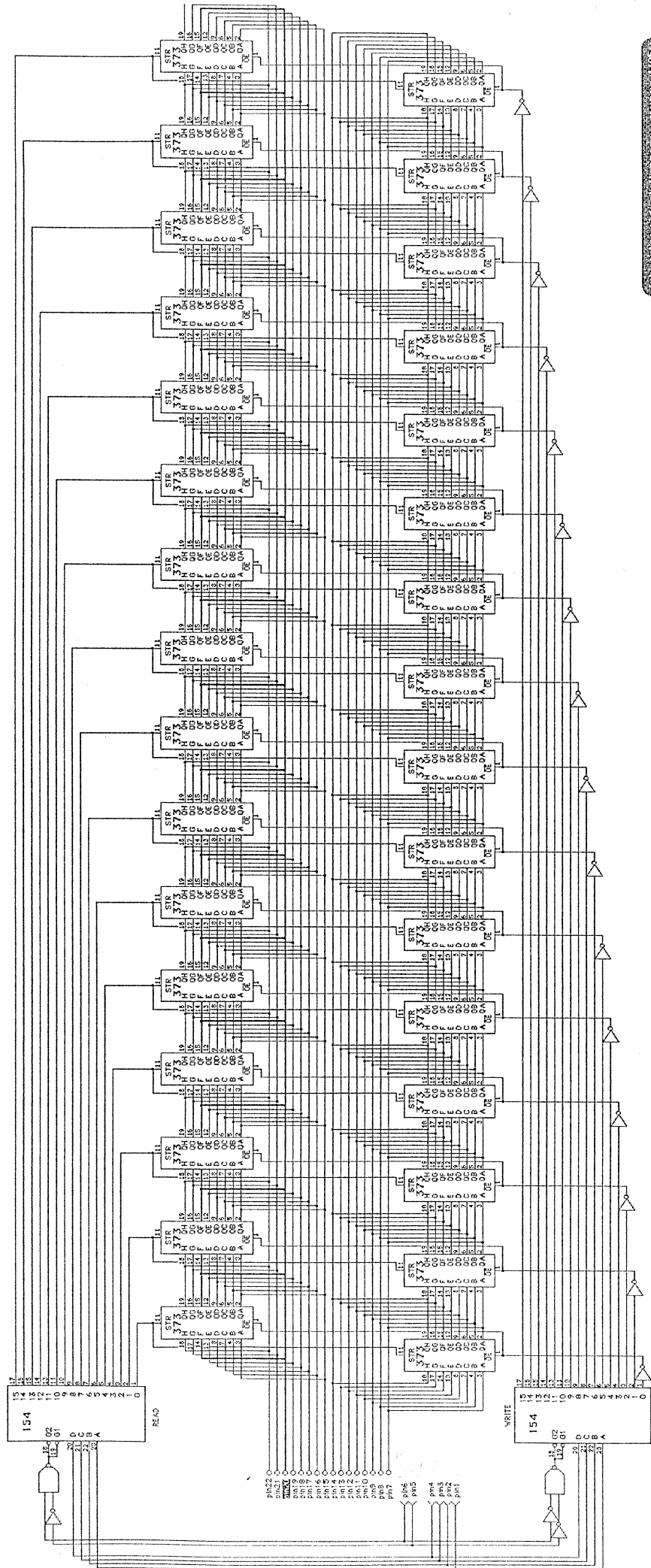
**Laboratorio di elettronica - gruppo 1/95**

## 7. R.A.M.

E' costituita da 16x2 integrati tipo 74LS373 che contengono 8 LATCH tipo D ciascuno; quindi è una RAM 16x16 bit per un totale di 256 bit.

I 4 bit di indirizzo A0-A4 vengono decodificati a 16 tramite 2 integrati 74LS154 che vengono abilitati separatamente nel caso di READ o WRITE; è previsto inoltre il pin di CHIP-SELECT, per eventuali espansioni.

I pin 1-6 sono segnali di ingresso, 7-22 sono ingressi/uscita per i dati.



Fabbricatore di elettronica di gruppo IAS



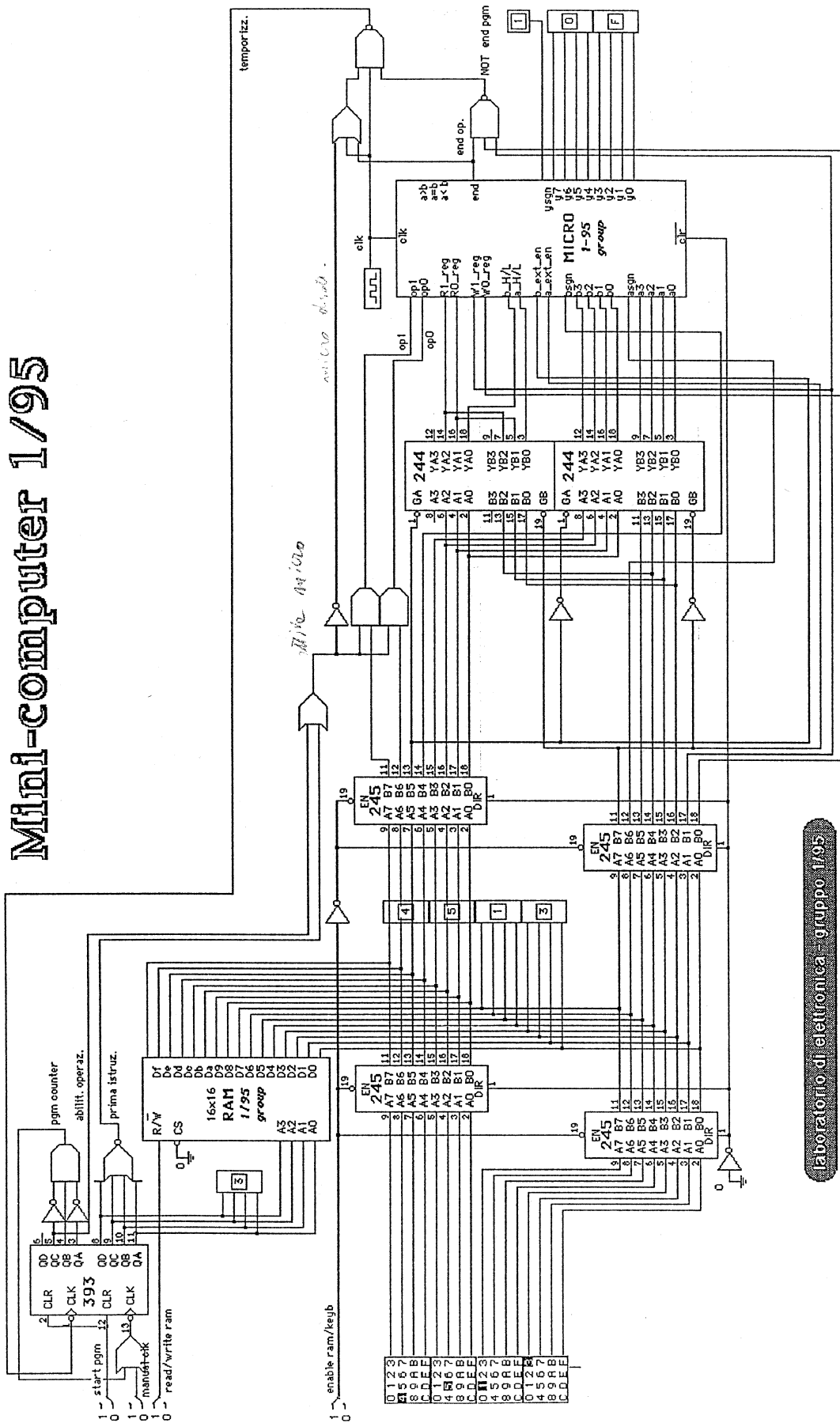
## 8. CODIFICA ISTRUZIONI

Occorrono 16 bit per la codifica delle istruzioni che possono così essere fornite alla RAM in D.M.A. tramite 4 tastiere.

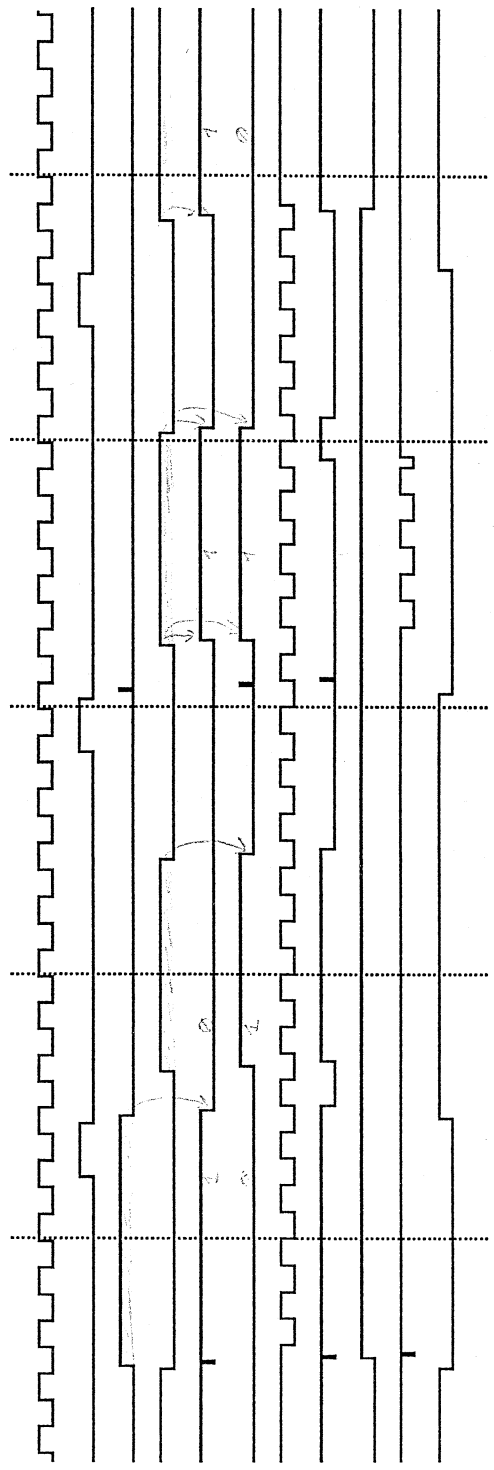
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2 bit		6 bit 1° operando						6 bit 2° operando						2 bit	
per il tipo di istruzione:		per il tipo di indirizzamento:										per il registro destinazione:			
00 nop		1) Indirizzamento diretto (dato da registro): preimi 3 bit =0, 2 bit che definiscono il registro, 1 bit per definire quale gruppo di 4 bit (0=Low, 1=High), perchè i registri sono a 8 bit;										00 = r0			
01 add		2) Indirizzamento immediato (dato definito nell'istr.):										01 = r1			
10 sub		preimo bit=1, 1 bit per il segno, 4 bit per il dato.										10 = r2			
11 mul												11 = r3			

Se la destinazione è definita sul registro r3, allora il programma viene automaticamente terminato, e abilitata la visualizzazione del risultato sul display.

# Mini-computer 1/95



laboratorio di elettronica - gruppo 1/95



clk  
 pgm counter  
 prima istruz.  
 abilit. operaz.  
 op0  
 op1  
 temporizz.  
 end op.  
 NOT end pgm

Laboratorio di Elettronica-progetto di mini-computer  
Gruppo 1/95

Programma di prova

add #+ 3, #- 5, r0 r0=(+3)+(-5)=-02  
01 10 0011 11 0101 00  
└───┘ └───┘ └───┘ └───┘  
6 3 D 4

sub r0.L, #- 5, r1 r1=(-5)-(r0.L)=-03  
10 00000 0 11 0101 01  
└───┘ └───┘ └───┘ └───┘  
8 0 D 5

mul #+ A, r1.L, r2 r2=(+A)\*(r1.L)=-1E  
11 10 1010 00001 0 10  
└───┘ └───┘ └───┘ └───┘  
E A 0 A

add r2.H, r2.L, r3 r3=(r2.H)+(r2.L)=(-1)+(-E)=-0F  
01 00010 1 00010 0 11  
└───┘ └───┘ └───┘ └───┘  
4 5 1 3

*fine pg e visualizza risultato*