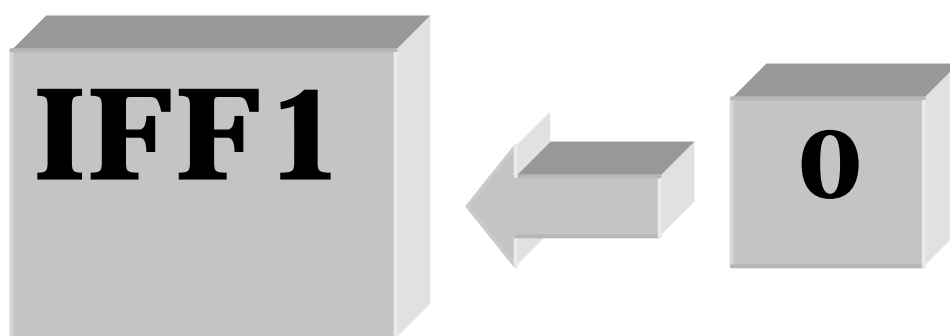


*Istruzioni di
gestione delle
interruzioni*

DI

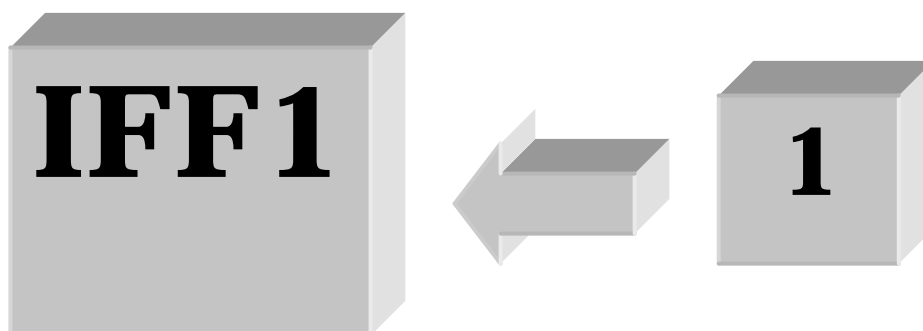
E' l'istruzione di disabilitazione delle interruzioni mascherabili (Disable Interrupt).

Il microprocessore Z80 non viene interrotto da un'interruzione mascherabile se il registro IFF1 si trova a zero. Lo scopo dell'istruzione DI è proprio quello di porre a zero tale registro.

**EI**

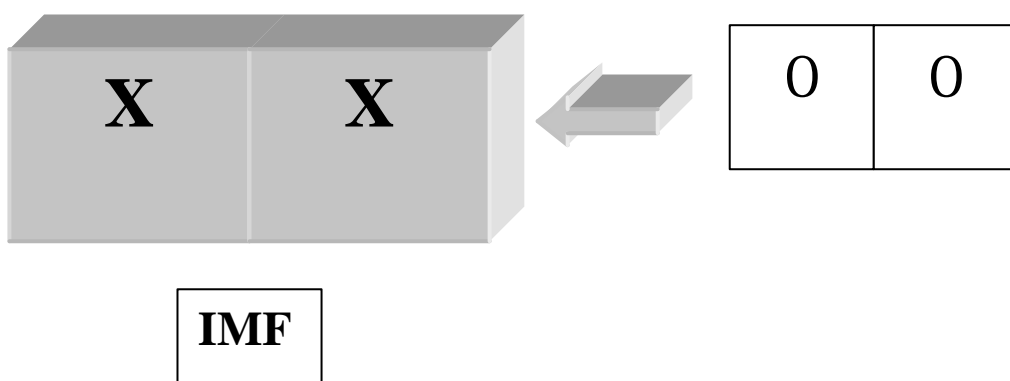
Quest'istruzione ha lo scopo di abilitare le interruzioni mascherabili in modo che possano causare un'interruzione del microprocessore. Essa agisce settando il registro

IFF1



IM0

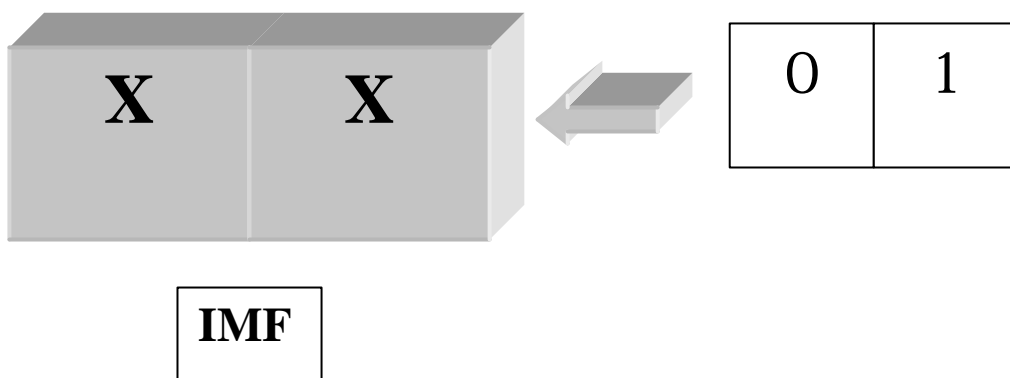
Quest'istruzione ha lo scopo di costringere il microprocessore a gestire le interruzioni mascherabili in modo zero. Essa agisce sul registro IMF dello Z80 ponendolo a zero



Non modifica il registro dei flag

IM1

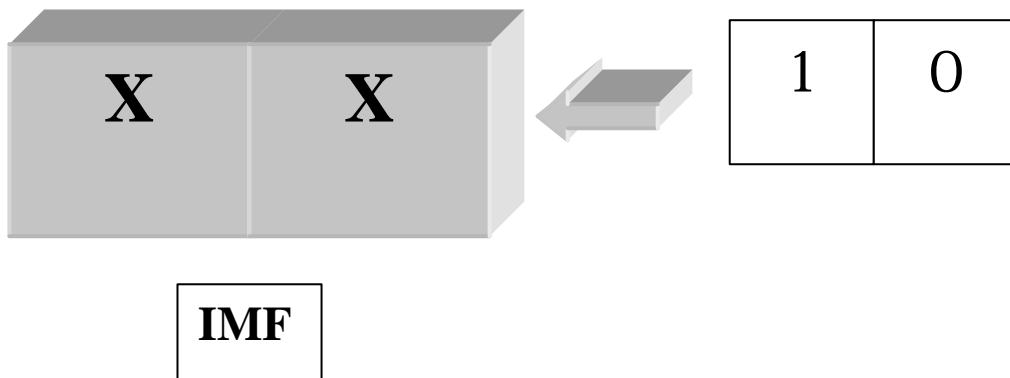
Quest'istruzione ha lo scopo di costringere il microprocessore a gestire le interruzioni mascherabili in modo uno. Essa agisce sul registro IMF dello Z80 ponendolo a 01



Non modifica il registro dei flag

IM2

Quest'istruzione ha lo scopo di costringere il microprocessore a gestire le interruzioni mascherabili in modo due. Essa agisce sul registro IMF dello Z80 ponendolo a 10



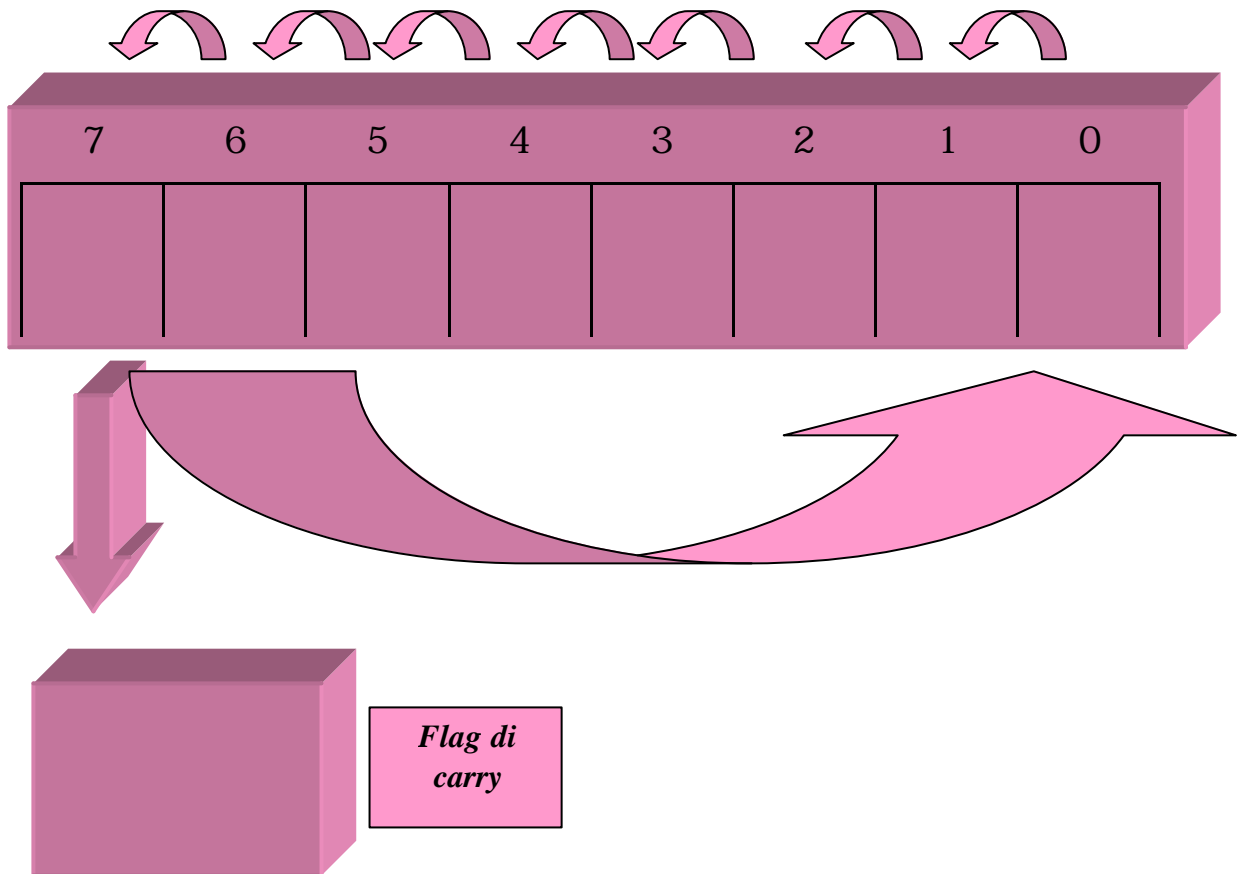
Non modifica il registro dei flag



*Istruzioni
di
rotazione*

RLC s

Quest'istruzione ruota verso sinistra l'operando s: ogni bit si sposta nella cella alla sua sinistra. Il bit 7 va ad occupare il posto del bit zero e contemporaneamente va nel flag di carry.

**Modifiche al registro dei flag**

S	Z		H		P/V	N	C
●	●		0		●	0	●

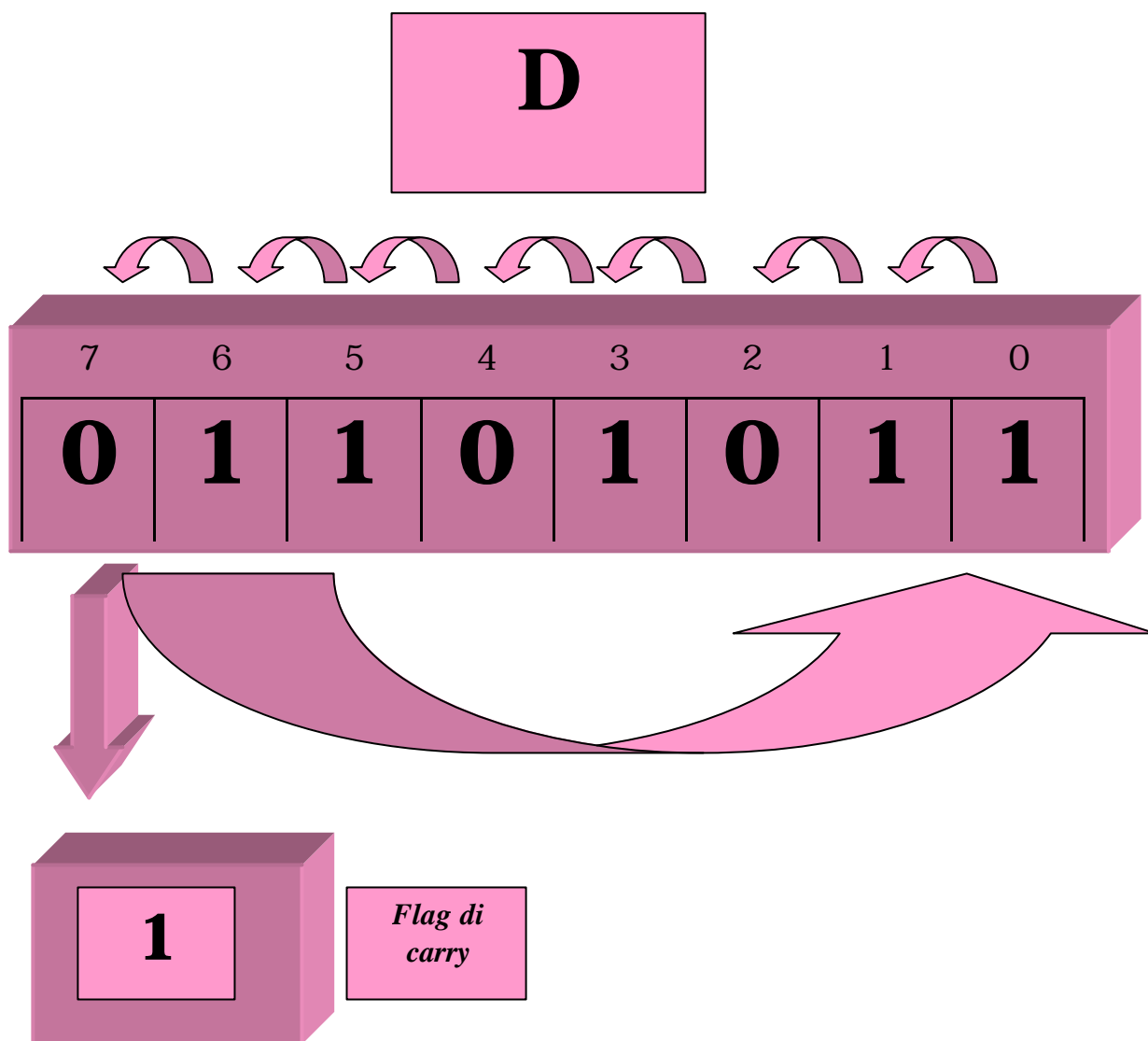
I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 7 dell'operando.

L'operando s può essere

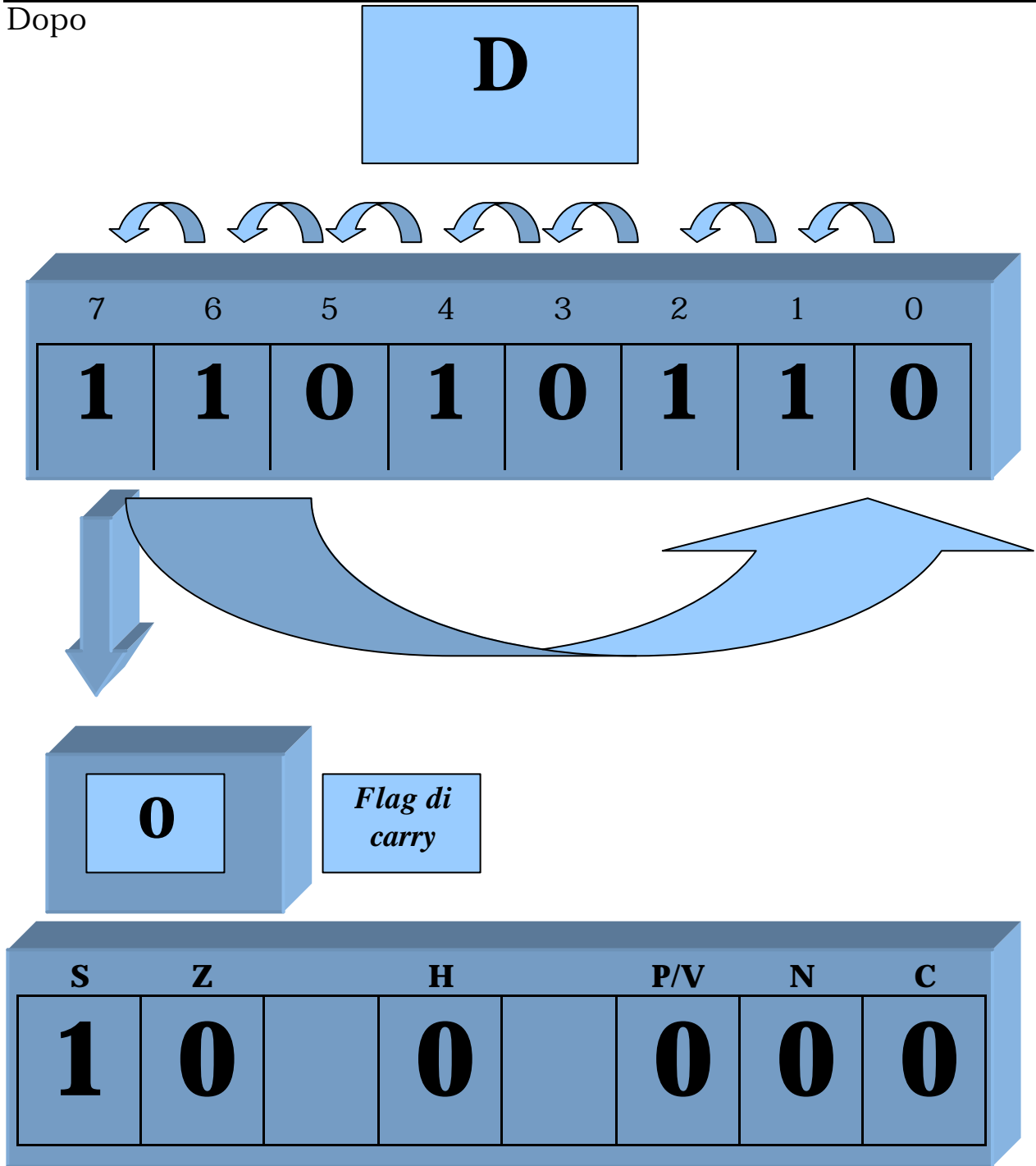
- ✓ Un registro ad otto bit del microprocessore A, B, C, D, E, H, L

Es. RLC D

Prima

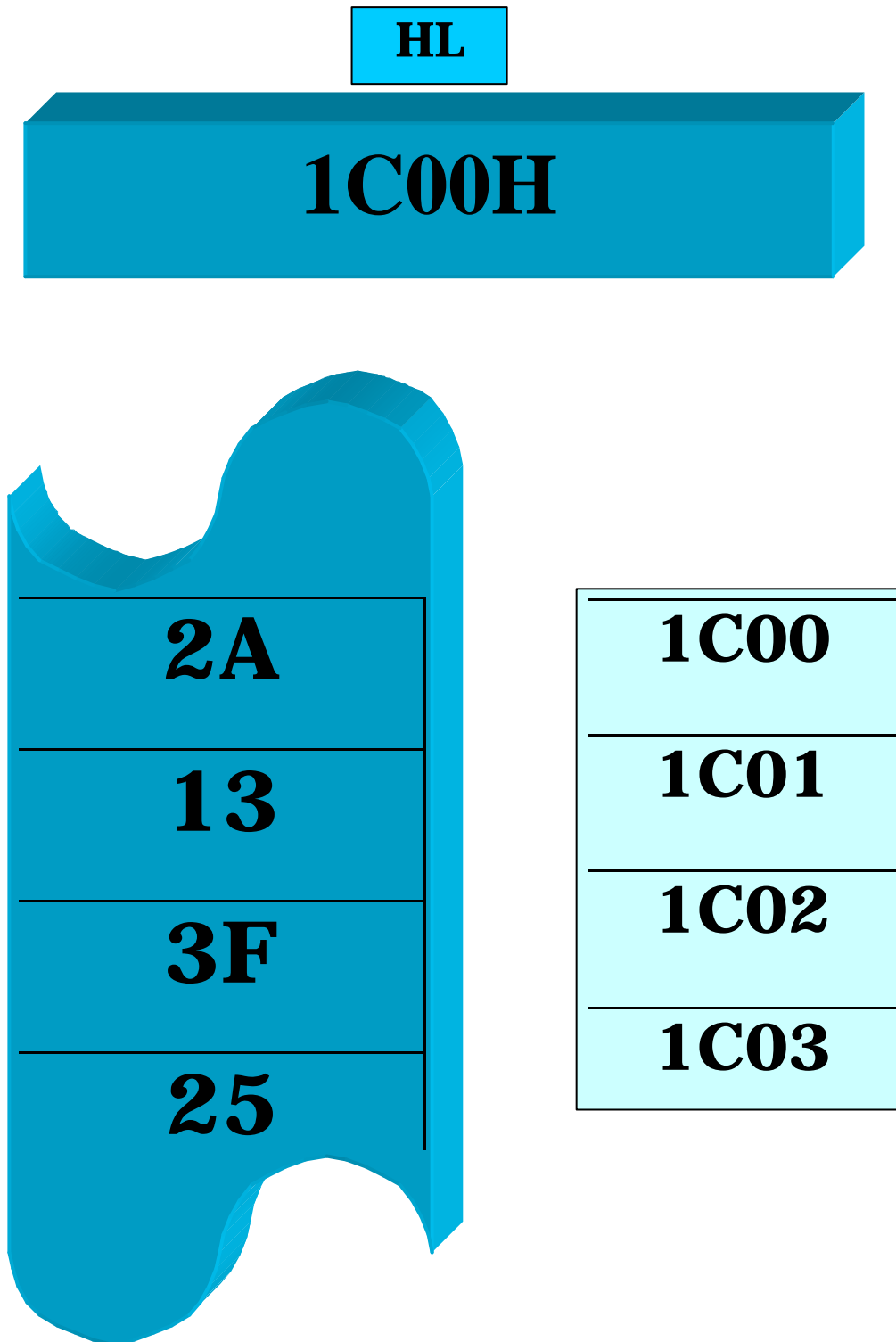


Dopo

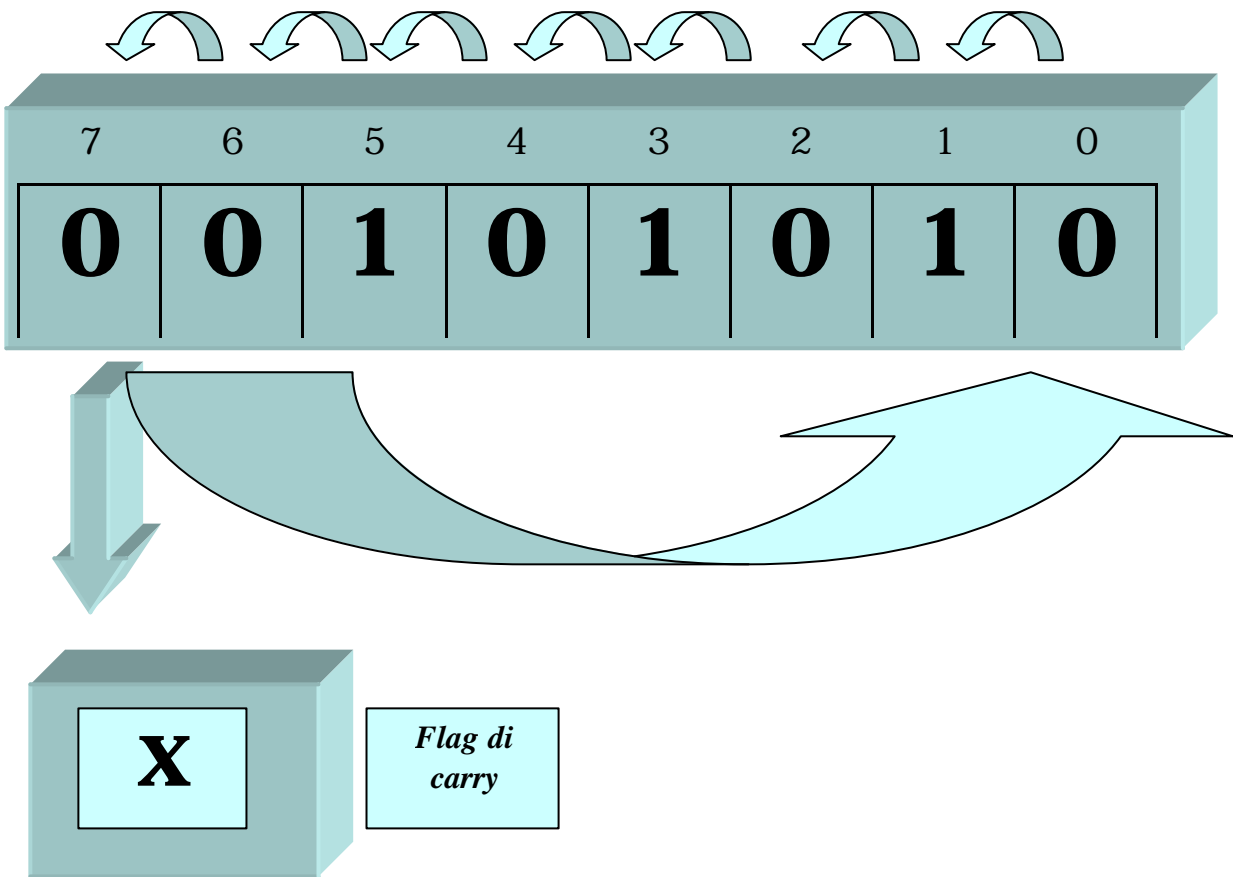


Il risultato della rotazione ha il bit 7 pari ad 1 quindi è un numero negativo (S=1), diverso da zero (Z=0), il numero di bit ad 1 è di cinque cioè è dispari (P=0), il bit 7 prima della rotazione era 0 (C=0).

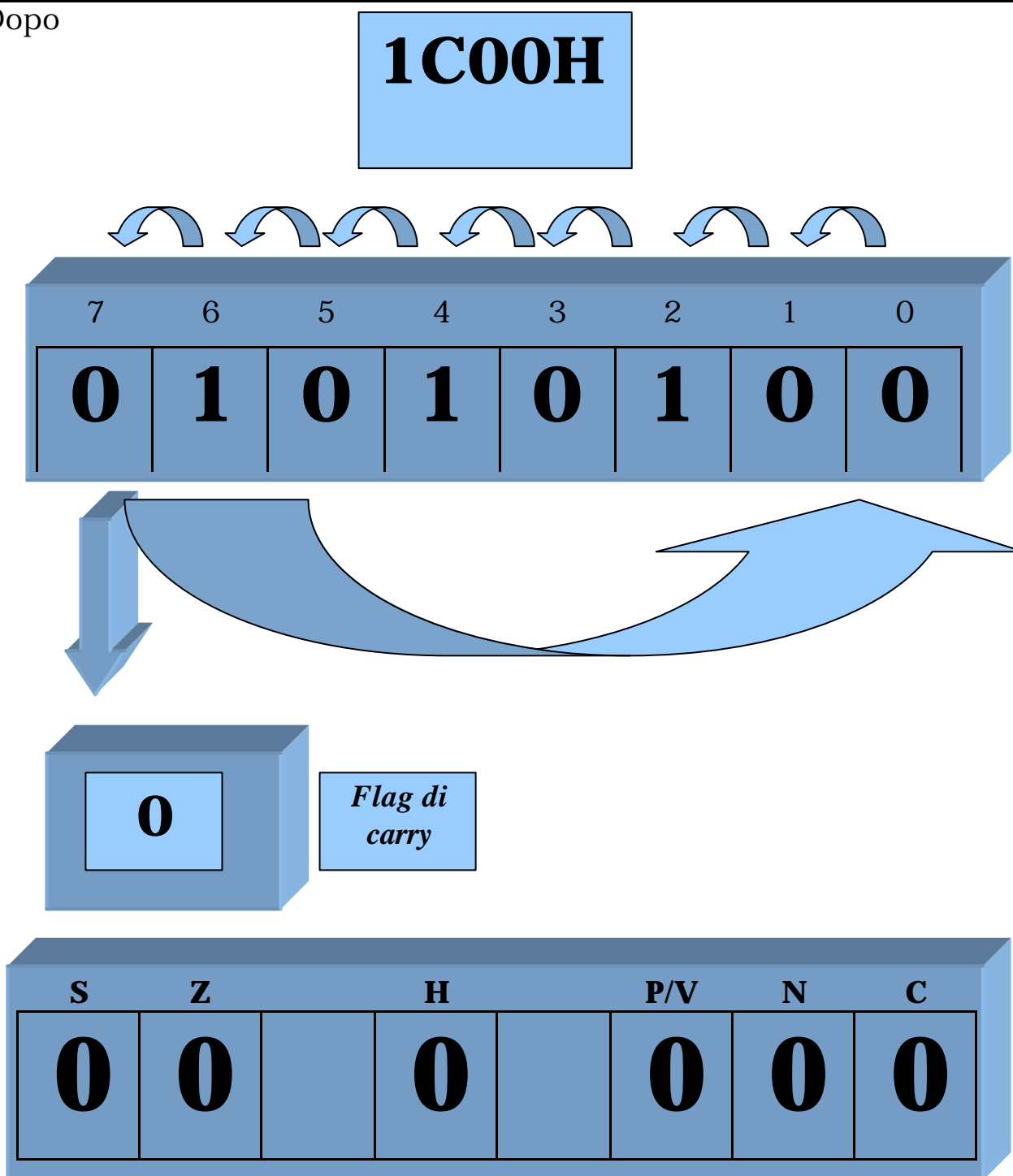
-
- ✓ L'operando s può essere il contenuto di una locazione di memoria il cui indirizzo è contenuto nel registro HL.



1C00H



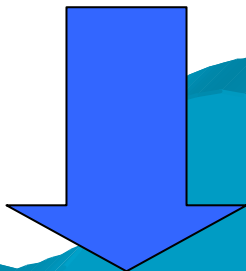
Dopo



Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo ($S=0$), diverso da zero ($Z=0$), il numero di bit ad 1 è di tre cioè è dispari ($P=0$), il bit 7 prima della rotazione era 0 ($C=0$).

HL

1C00H



54

13

3F

25

1C00

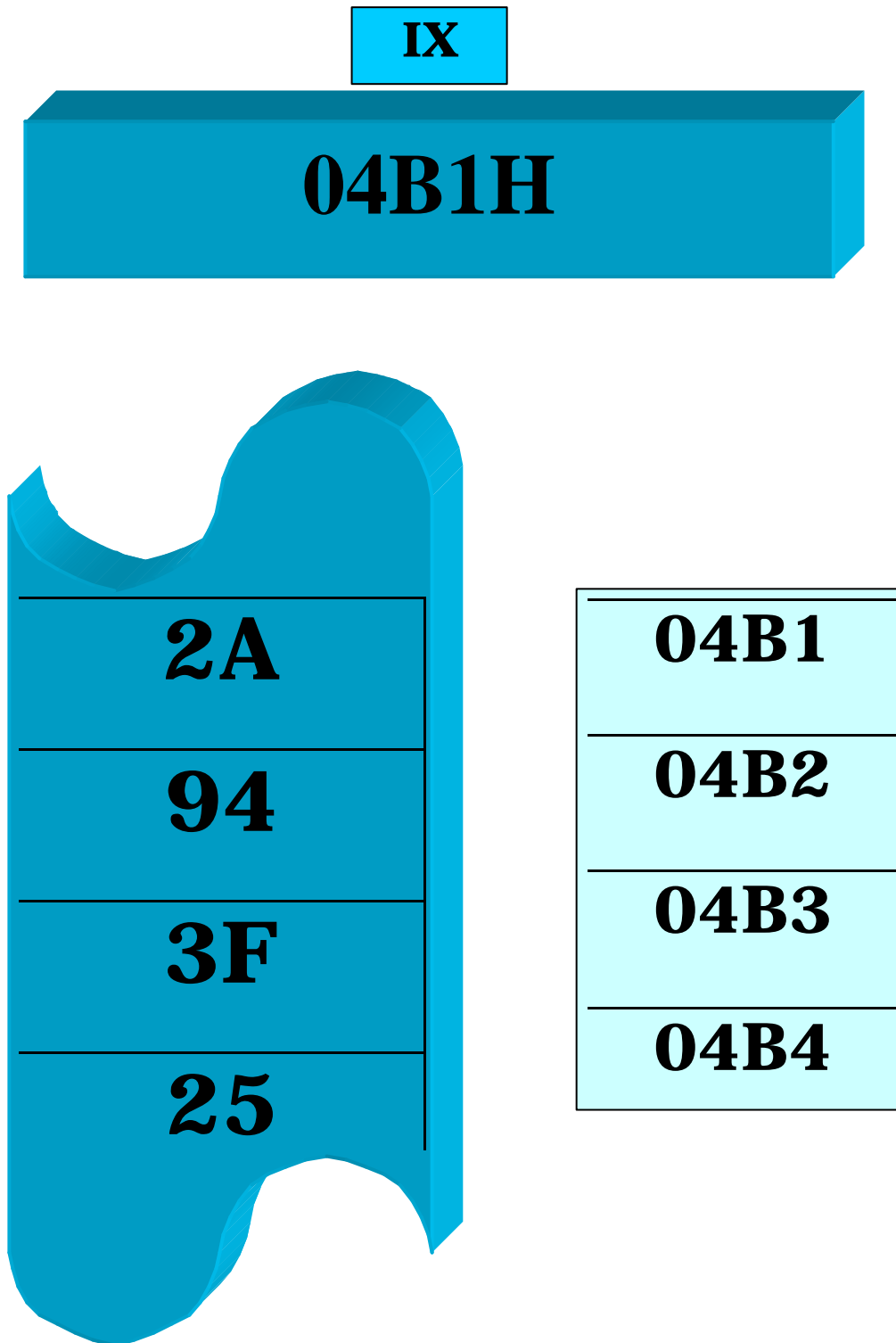
1C01

1C02

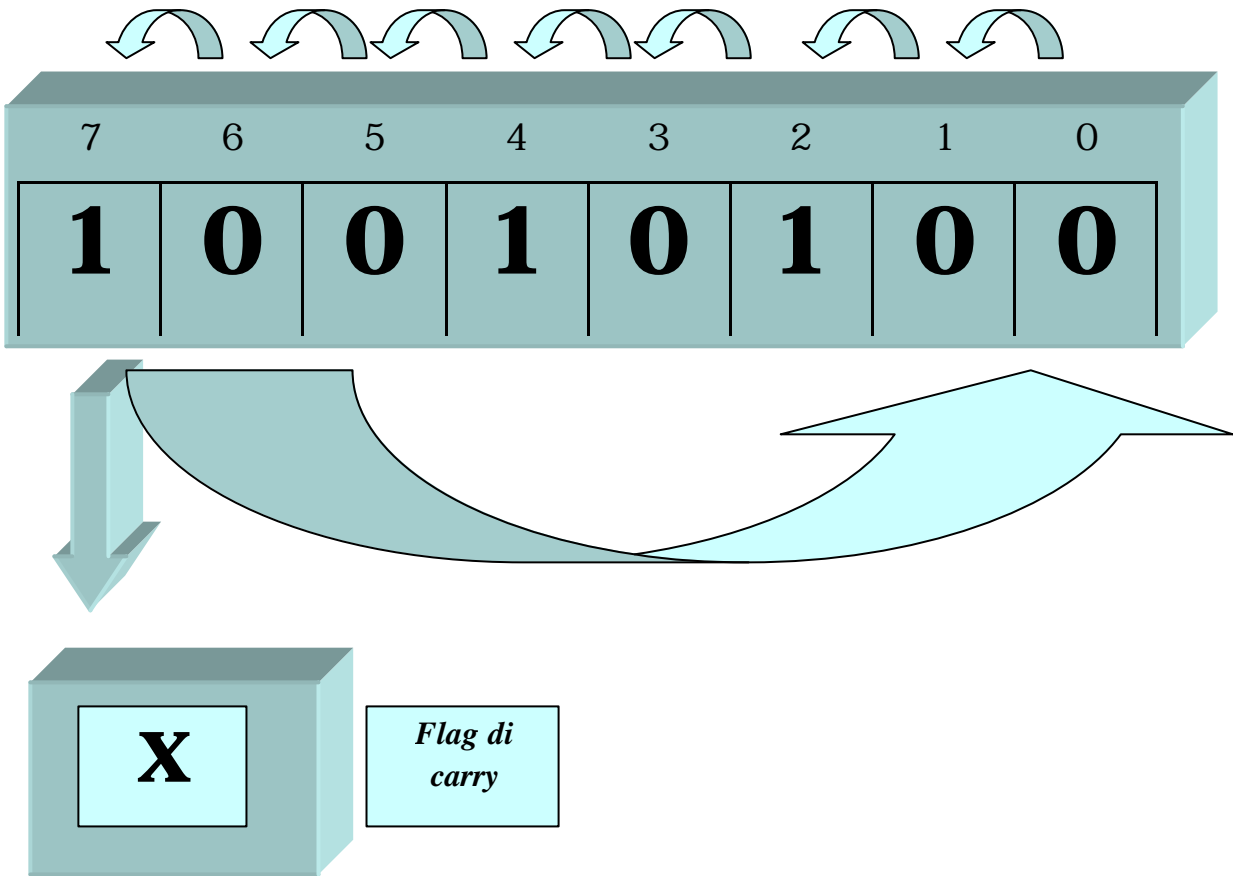
1C03

-
- ✓ L'operando s può essere il contenuto di una locazione di memoria indirizzata con indirizzamento indicizzato

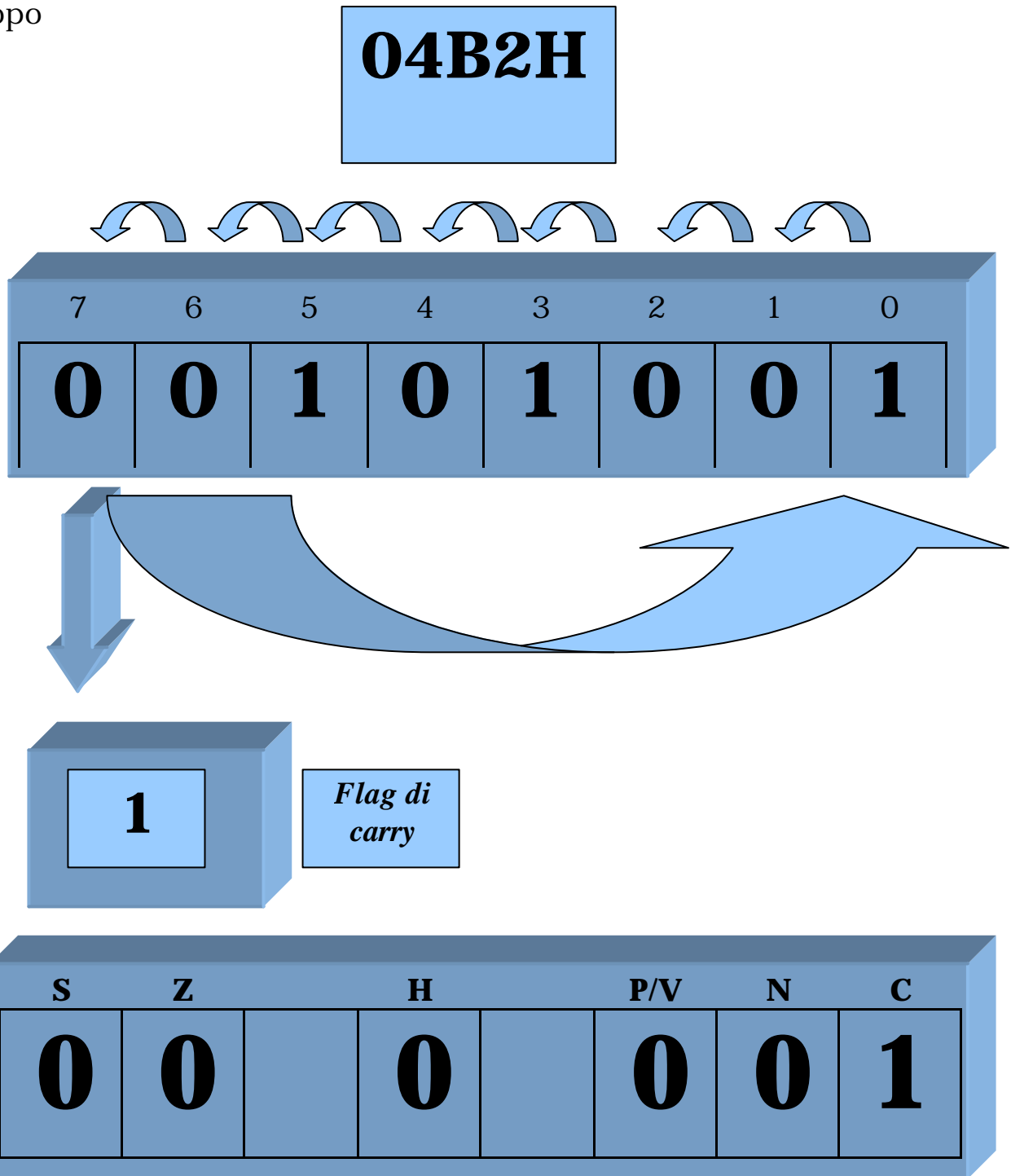
Es. RLC (IX+1)



04B2H



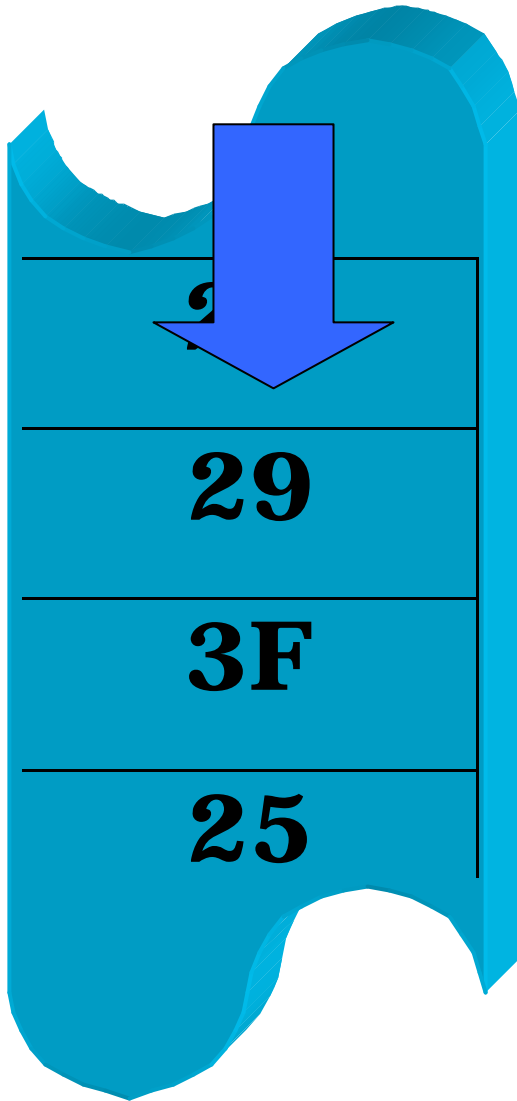
Dopo



Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo (S=0), diverso da zero (Z=0), il numero di bit ad 1 è di tre cioè è dispari (P=0), il bit 7 prima della rotazione era 1 (C=1).

IX

04B1H



04B1

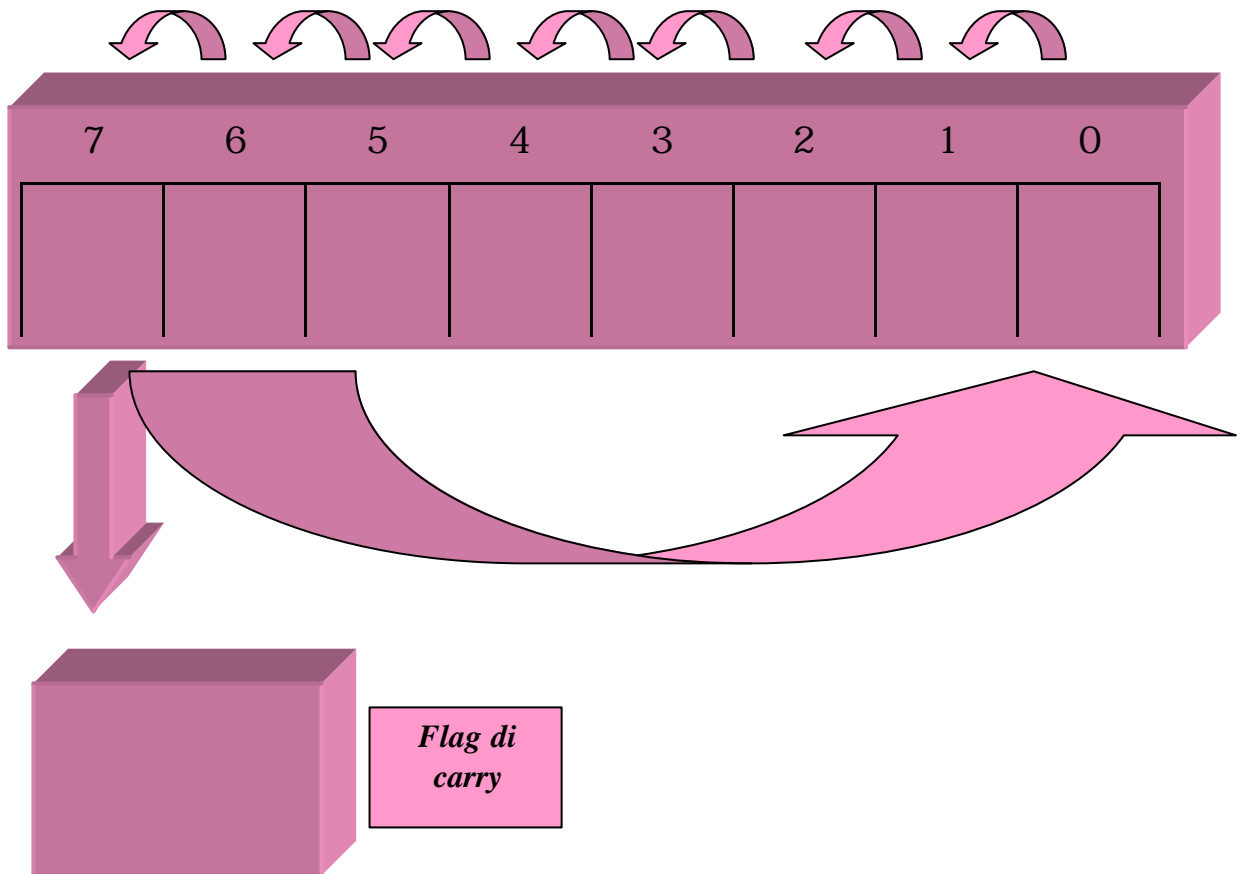
04B2

04B3

04B4

RLCA

Quest'istruzione ruota verso sinistra l'accumulatore: ogni bit si sposta nella cella alla sua sinistra. Il bit 7 va ad occupare il posto del bit zero e contemporaneamente va nel flag di carry.

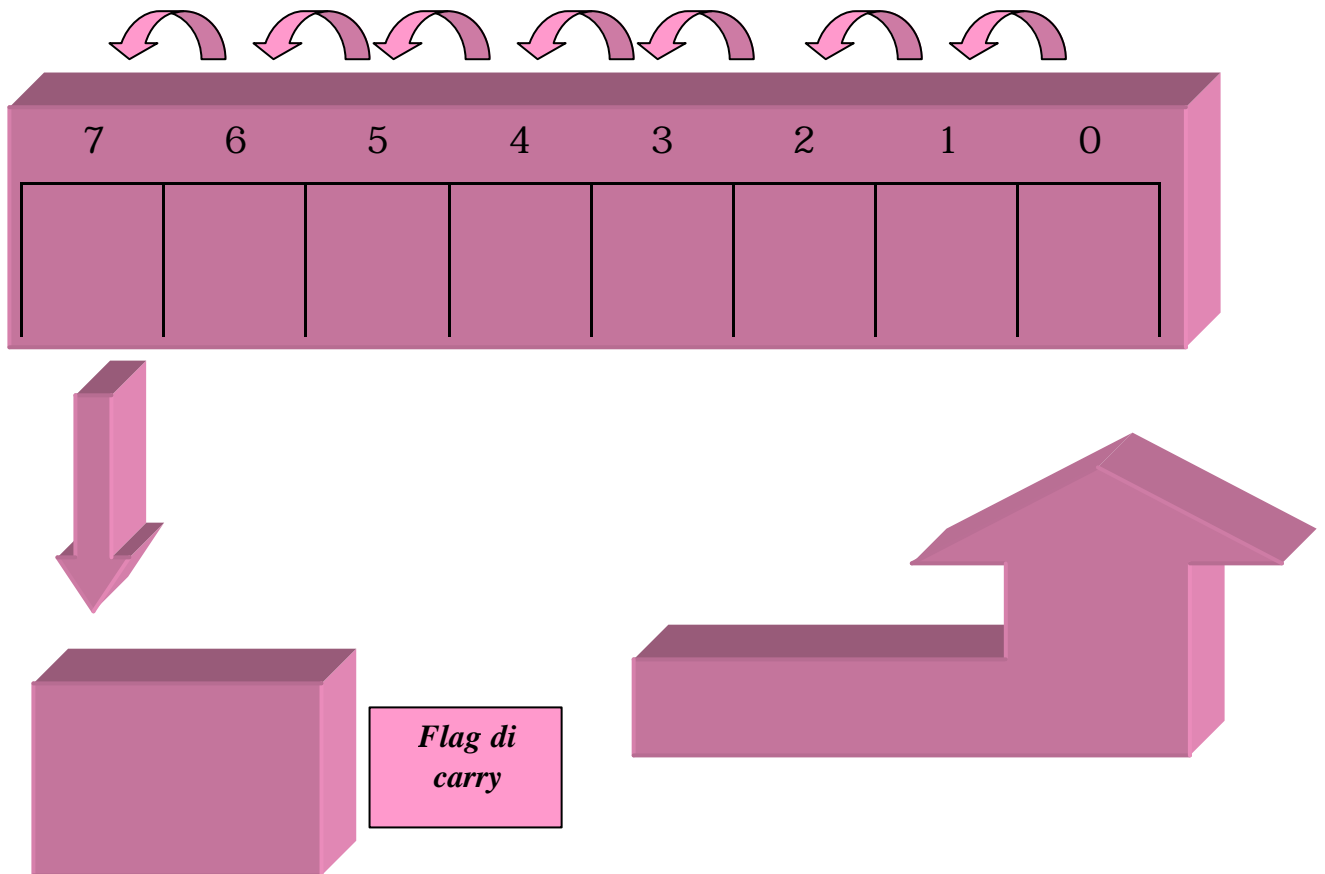
**Modifiche al registro dei flag**

S	Z		H		P/V	N	C
●	●		0		●	0	●

I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 7 dell'operando.

RL s

Quest'istruzione ruota verso sinistra l'operando s: ogni bit si sposta nella cella alla sua sinistra. Il bit 7 va ad occupare il posto del flag di carry, il vecchio contenuto del flag di carry va ad occupare il bit zero.



Modifiche al registro dei flag

S	Z		H		P/V	N	C
●	●		0		●	0	●

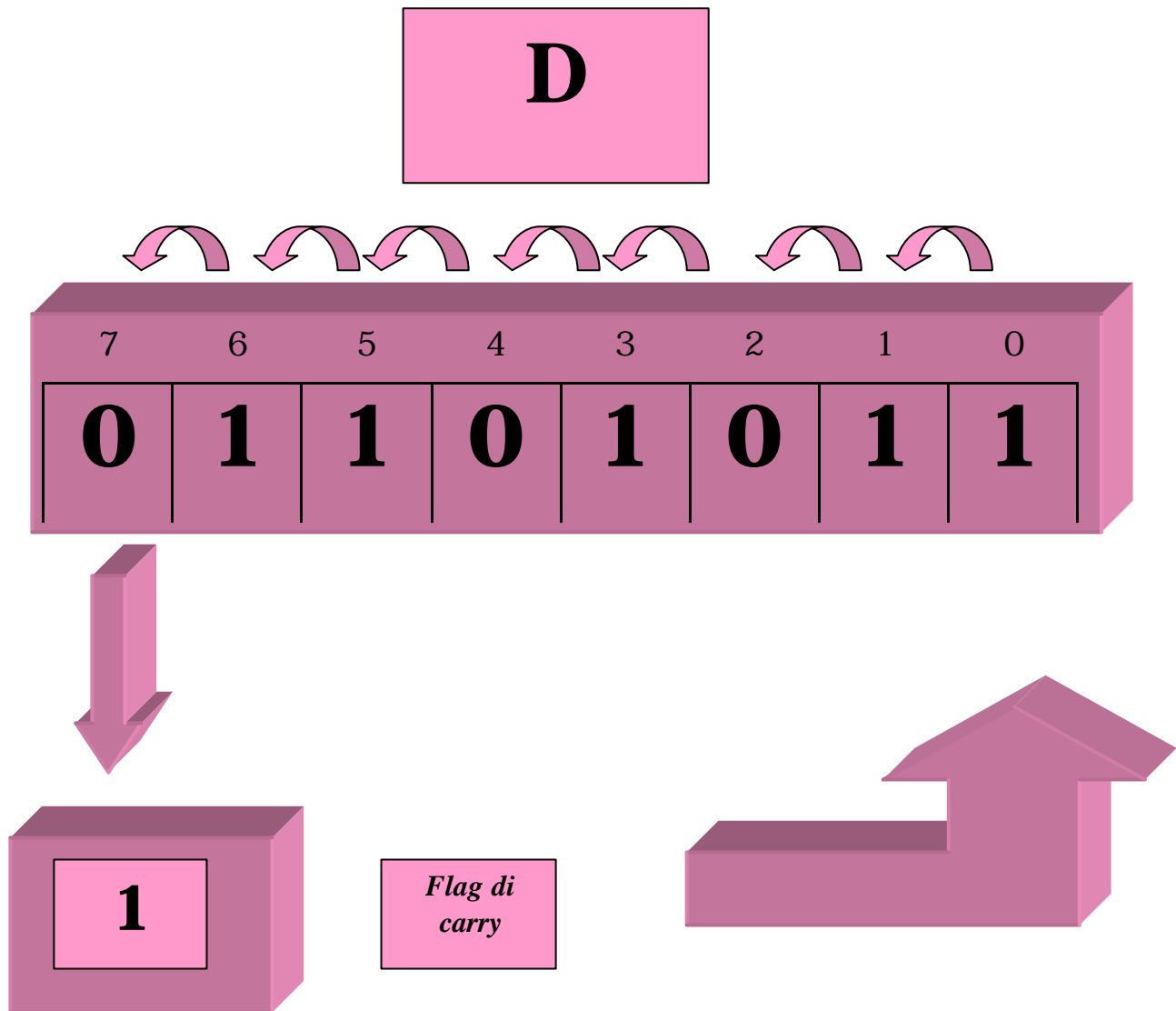
I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 7 dell'operando.

L'operando s può essere

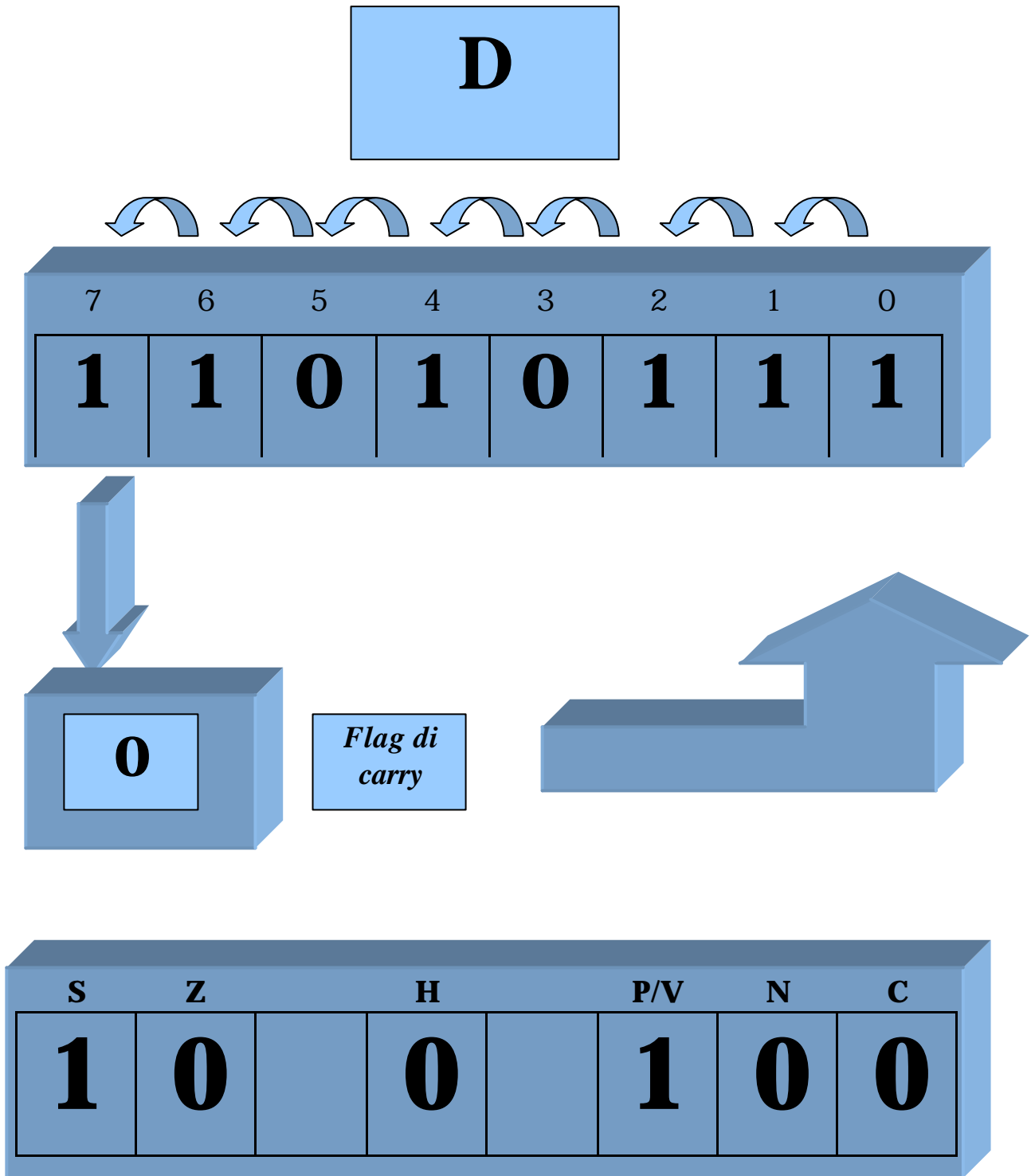
- ✓ Un registro ad otto bit del microprocessore A, B, C, D, E, H, L

Es. RLC D

Prima

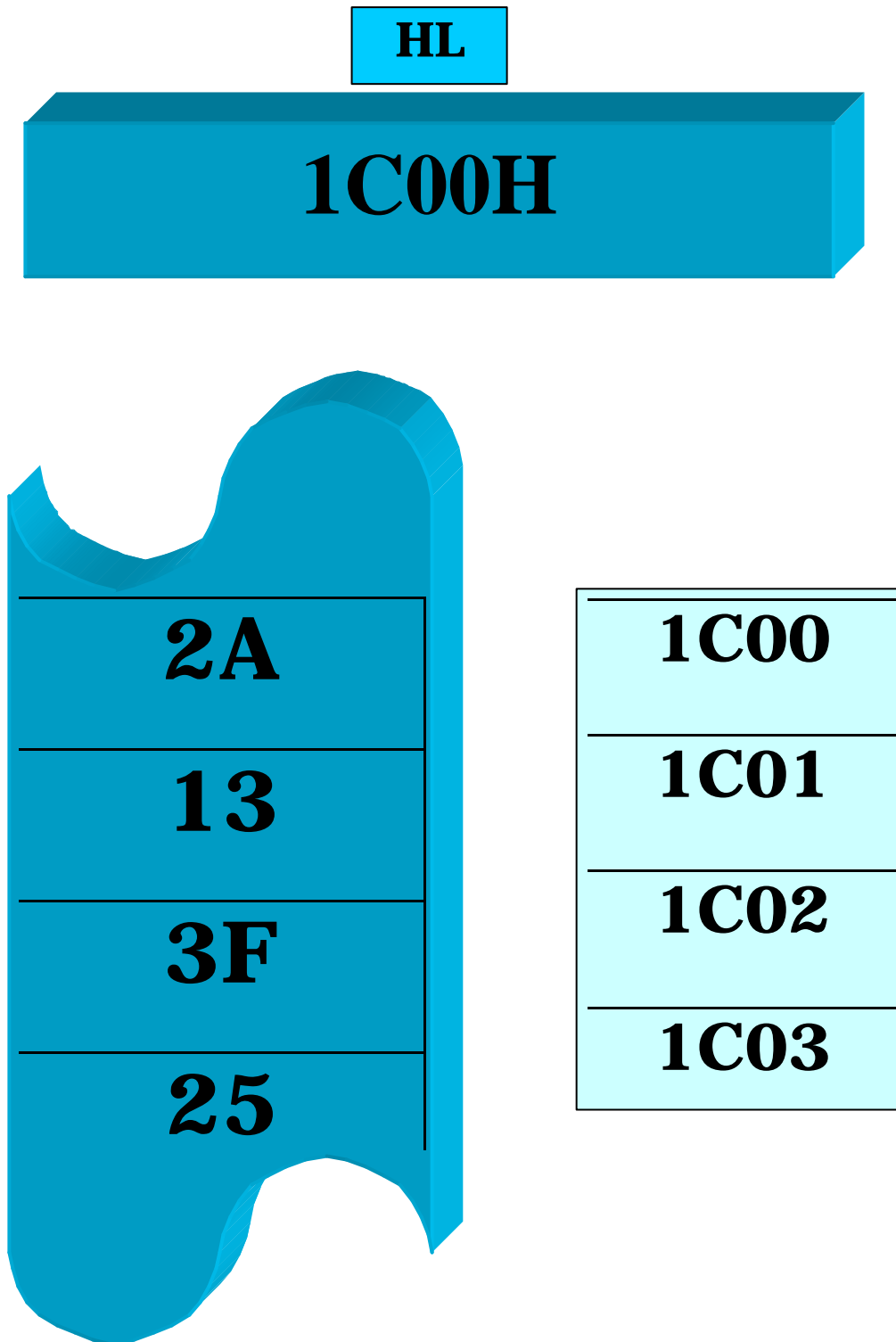


Dopo

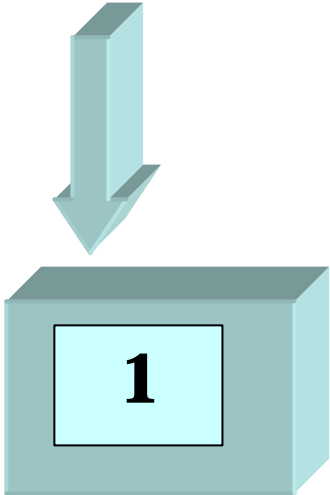
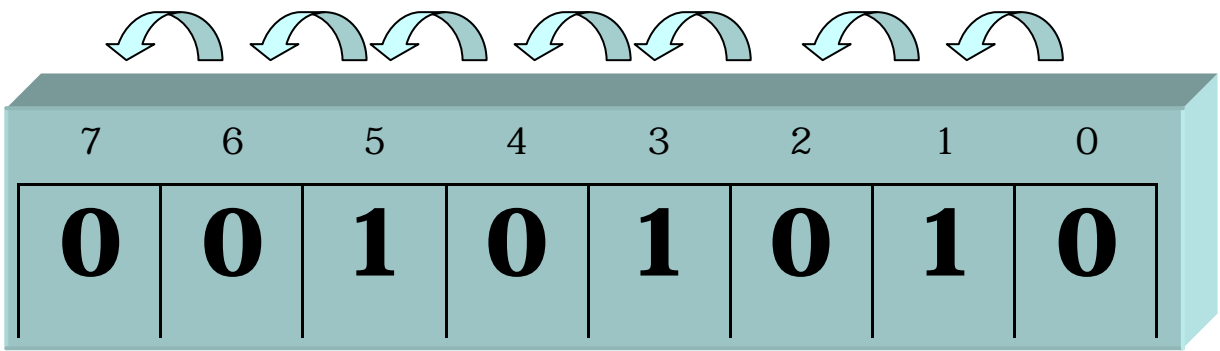


Il risultato della rotazione ha il bit 7 pari ad 1 quindi è un numero negativo (S=1), diverso da zero (Z=0), il numero di bit ad 1 è di sei cioè è pari (P=1), il bit 7 prima della rotazione era 0 (C=0).

-
- ✓ L'operando s può essere il contenuto di una locazione di memoria il cui indirizzo è contenuto nel registro HL.



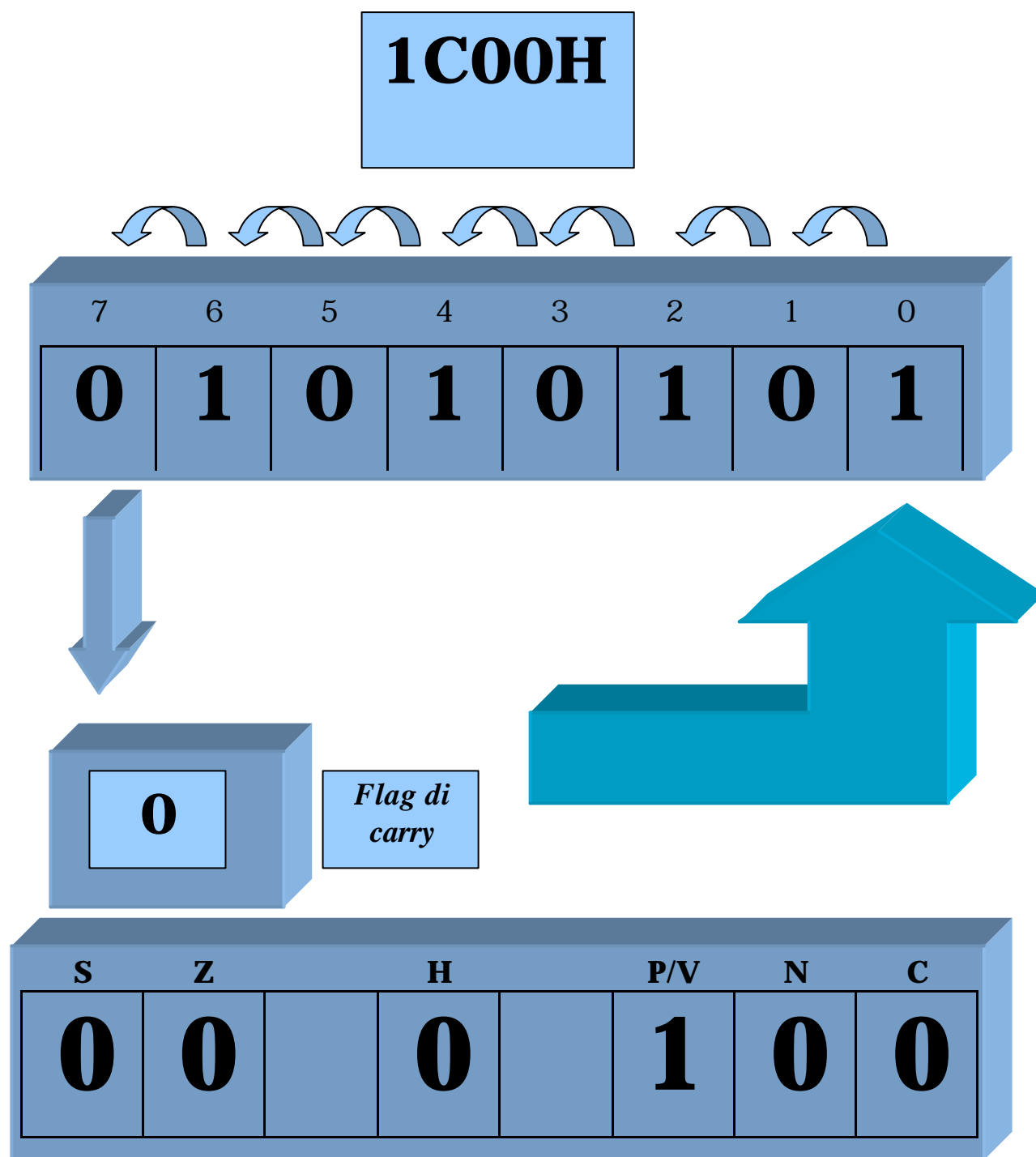
1C00H



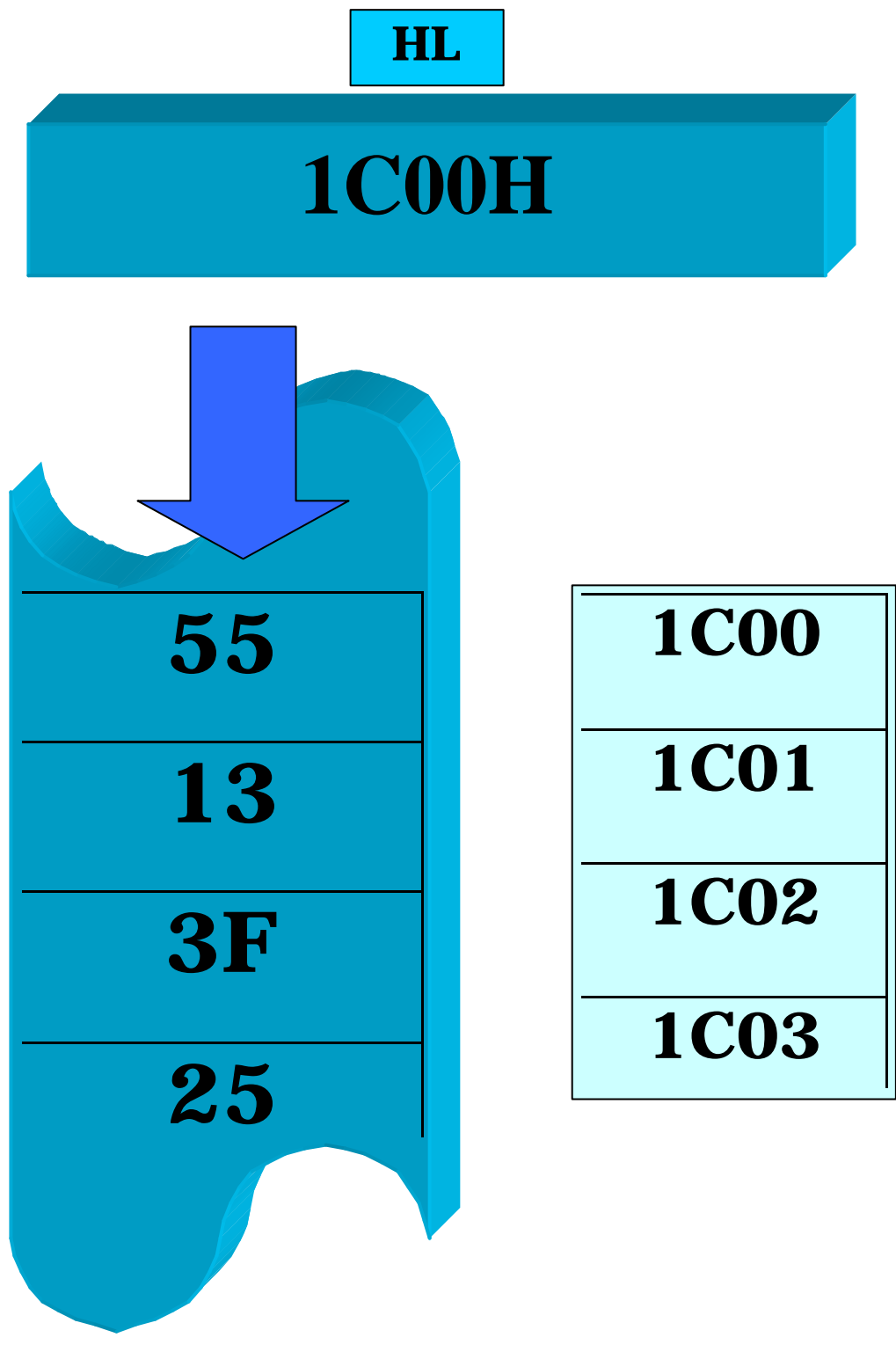
Flag di carry



Dopo

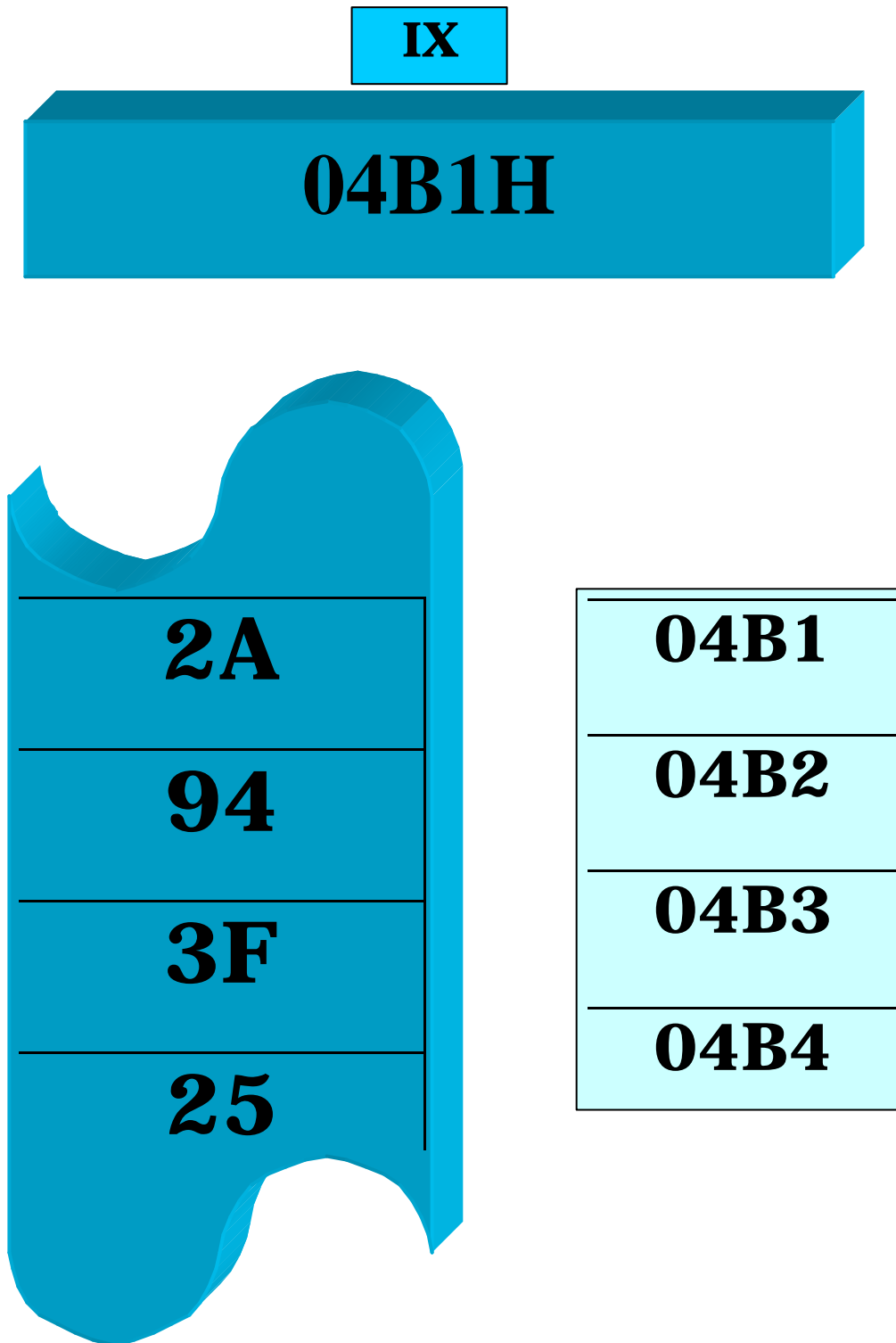


Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo (S=0), diverso da zero (Z=0), il numero di bit ad 1 è di quattro cioè è pari (P=1), il bit 7 prima della rotazione era 0 (C=0).

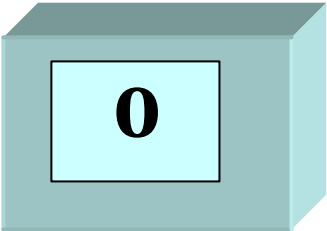
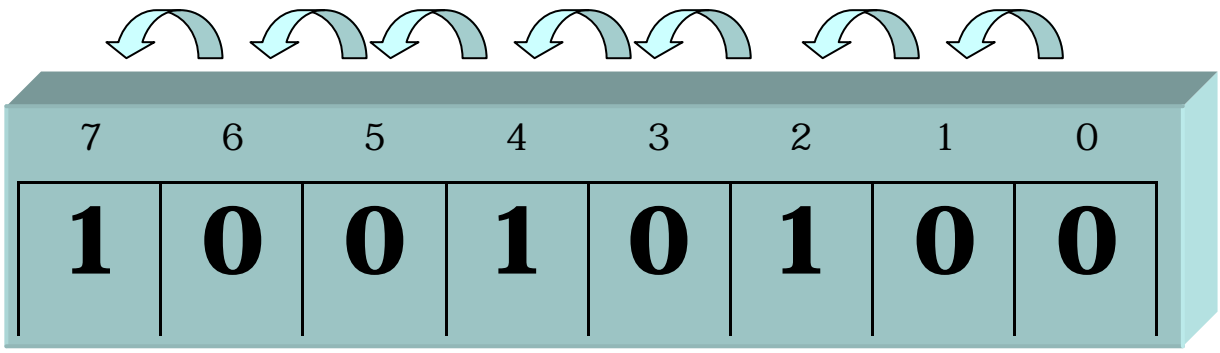


- ✓ L'operando s può essere il contenuto di una locazione di memoria indirizzata con indirizzamento indicizzato

Es. RLC (IX+1)



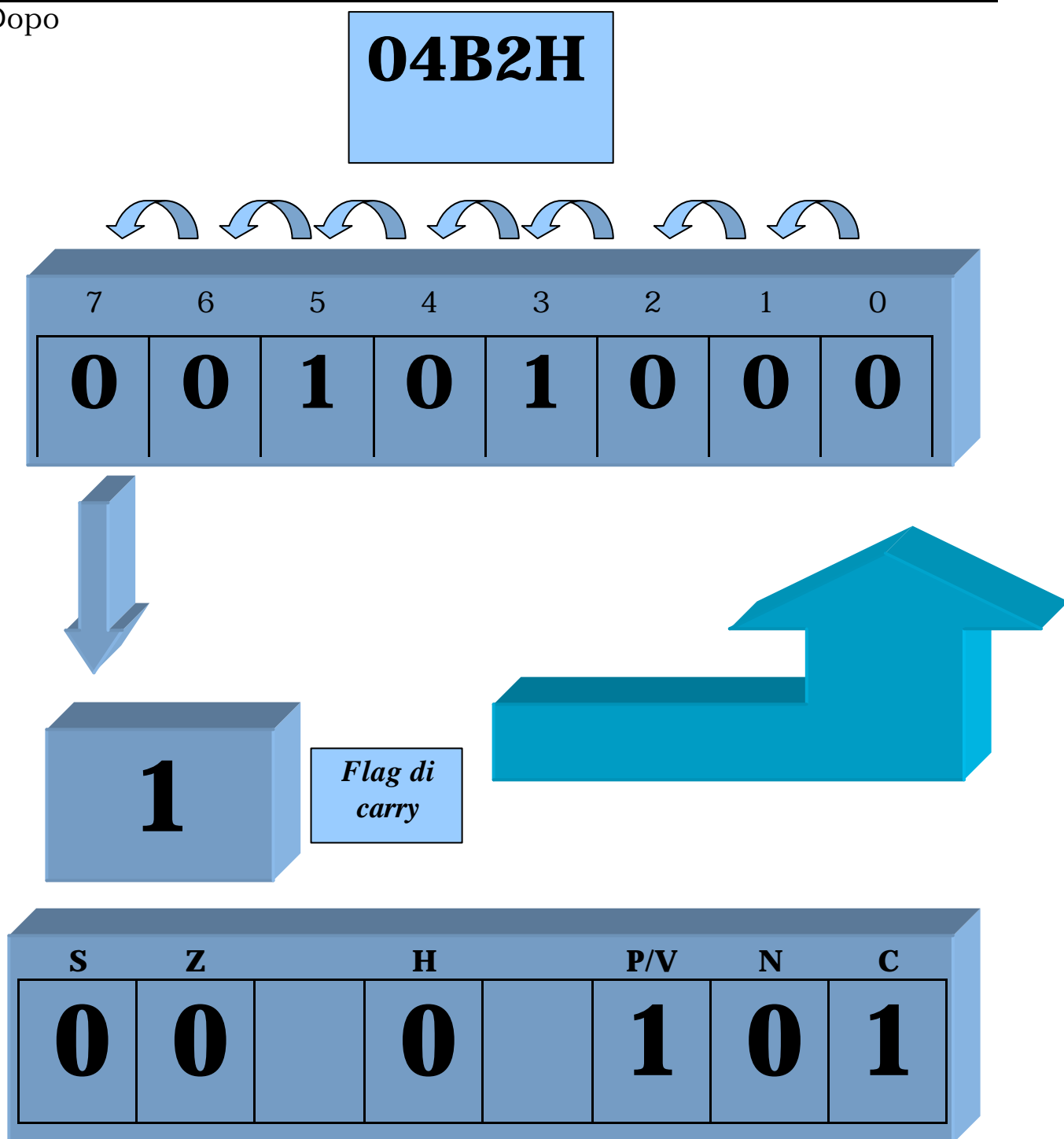
04B2H



Flag di carry



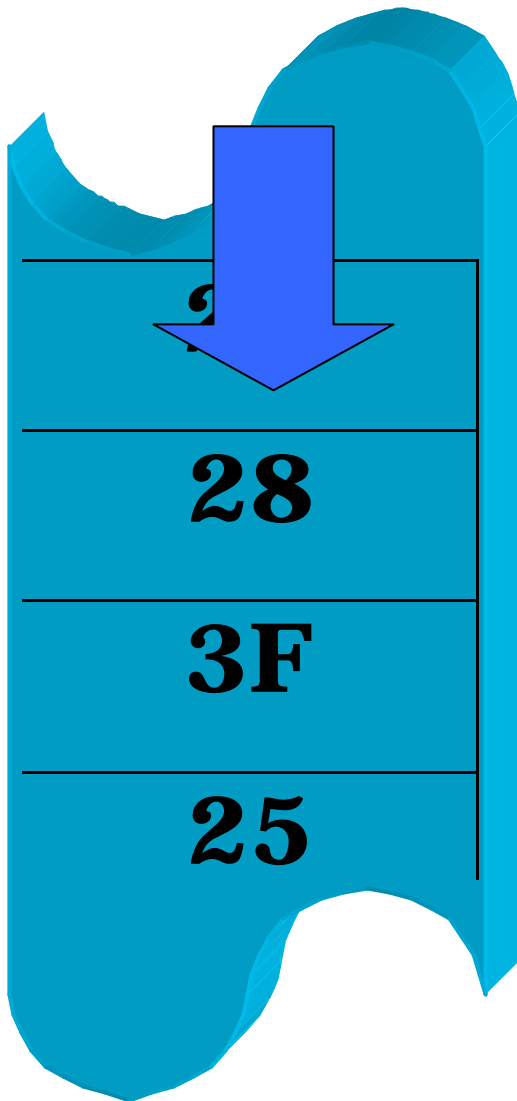
Dopo



Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo ($S=0$), diverso da zero ($Z=0$), il numero di bit ad 1 è di tre cioè è dispari ($P=0$), il bit 7 prima della rotazione era 1 ($C=1$).

IX

04B1H



04B1

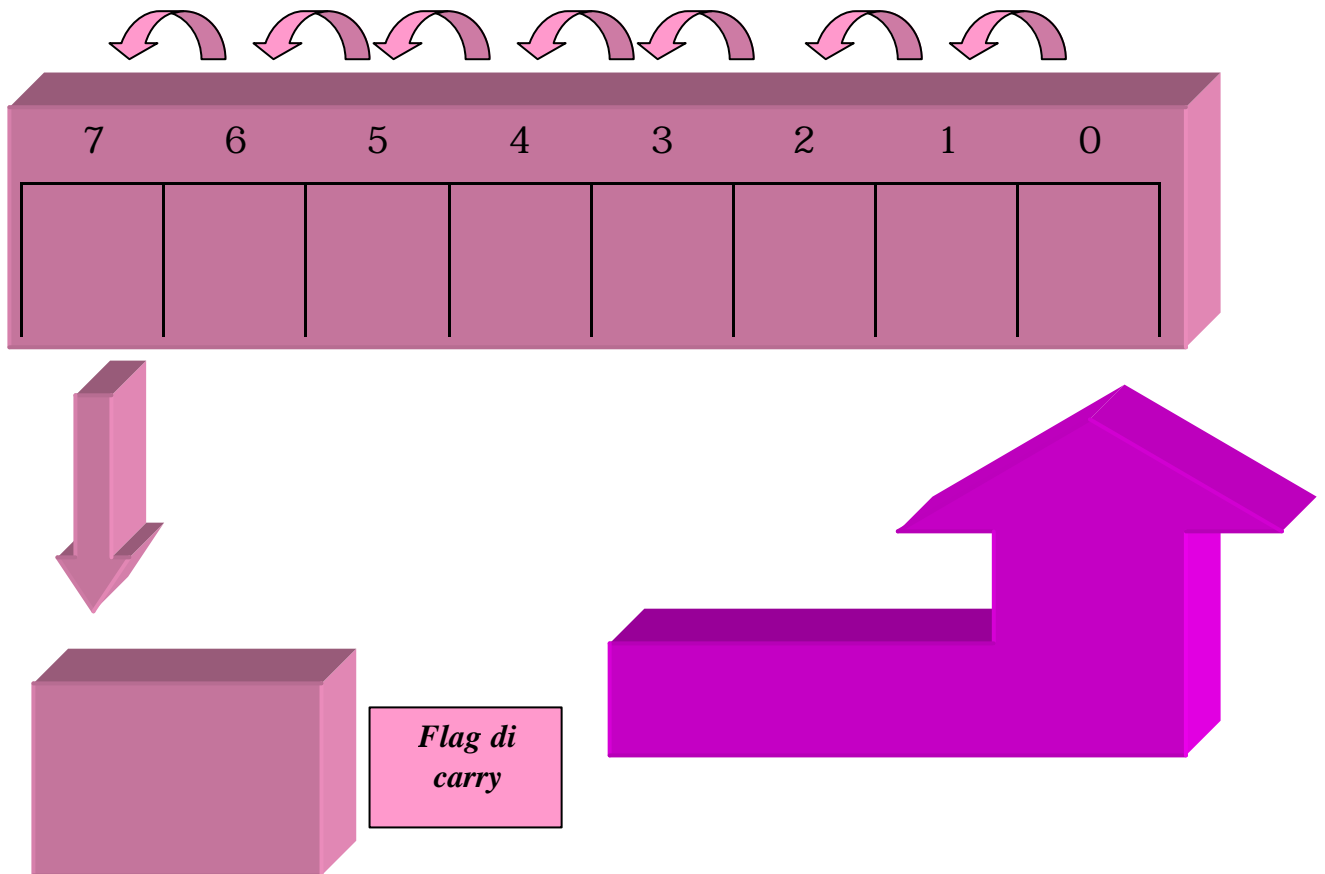
04B2

04B3

04B4

RLA

Quest'istruzione ruota verso sinistra l'accumulatore: ogni bit si sposta nella cella alla sua sinistra. Il bit 7 va ad occupare il flag di carry, il vecchio contenuto del flag di carry va ad occupare il posto del bit zero.

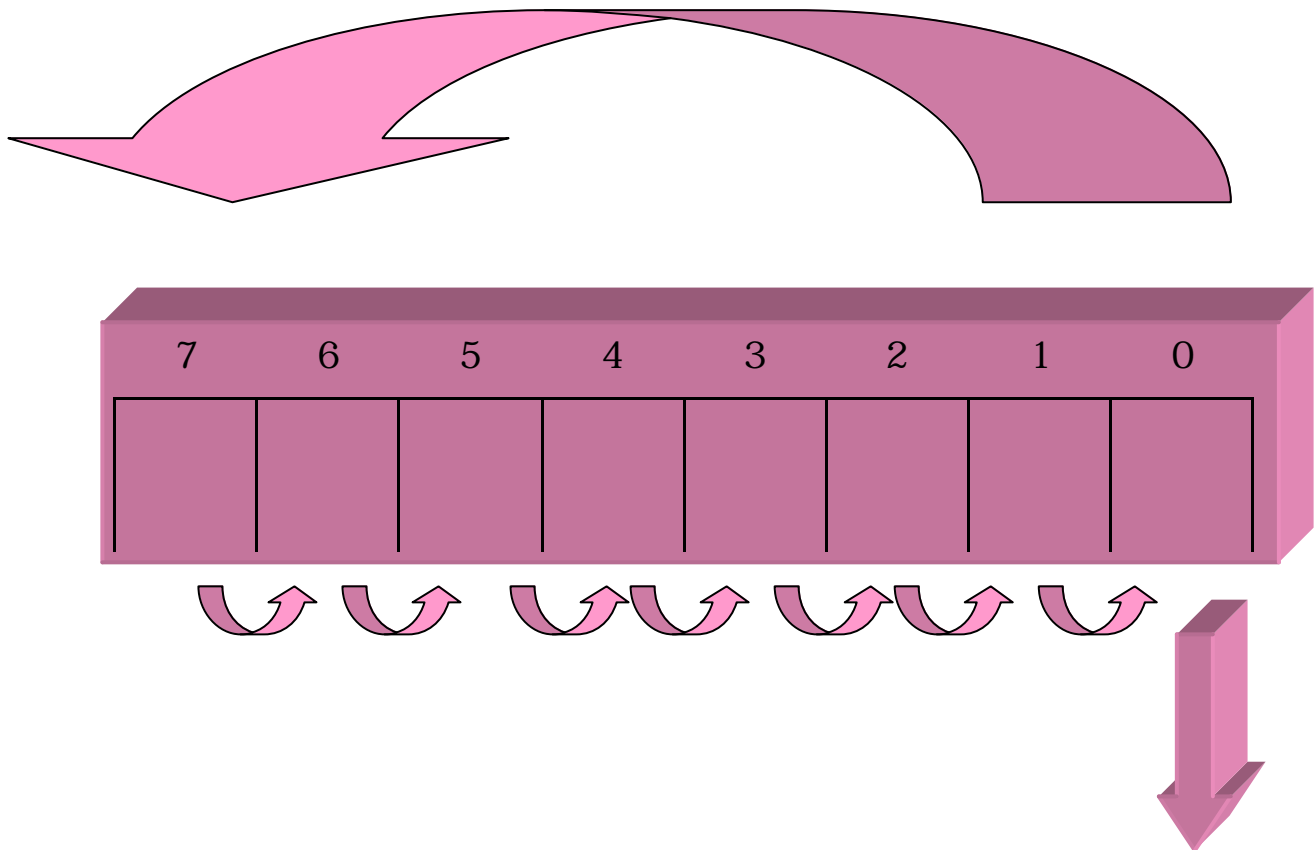
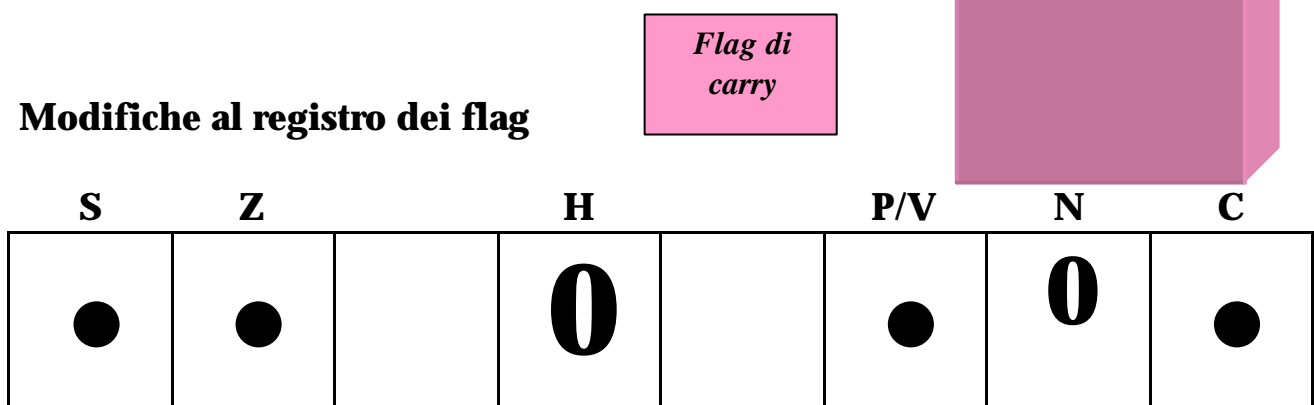
**Modifiche al registro dei flag**

S	Z		H		P/V	N	C
●	●		0		●	0	●

I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 7 dell'operando.

RRC s

Quest'istruzione ruota verso destra l'operando s: ogni bit si sposta nella cella alla sua destra. Il bit 0 va ad occupare il posto del bit 7 e contemporaneamente va nel flag di carry.

**Modifiche al registro dei flag**

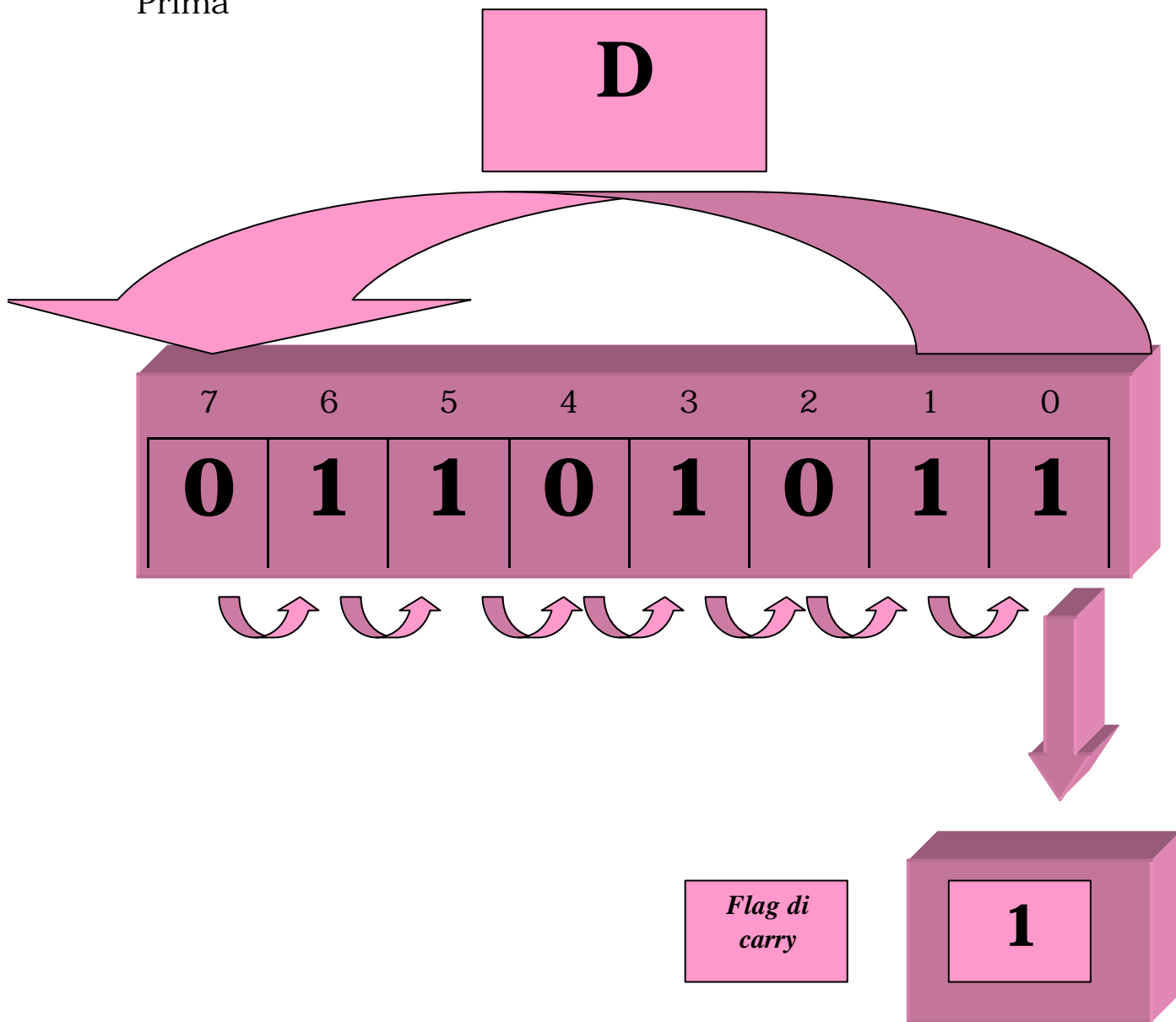
I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 7 dell'operando.

L'operando s può essere

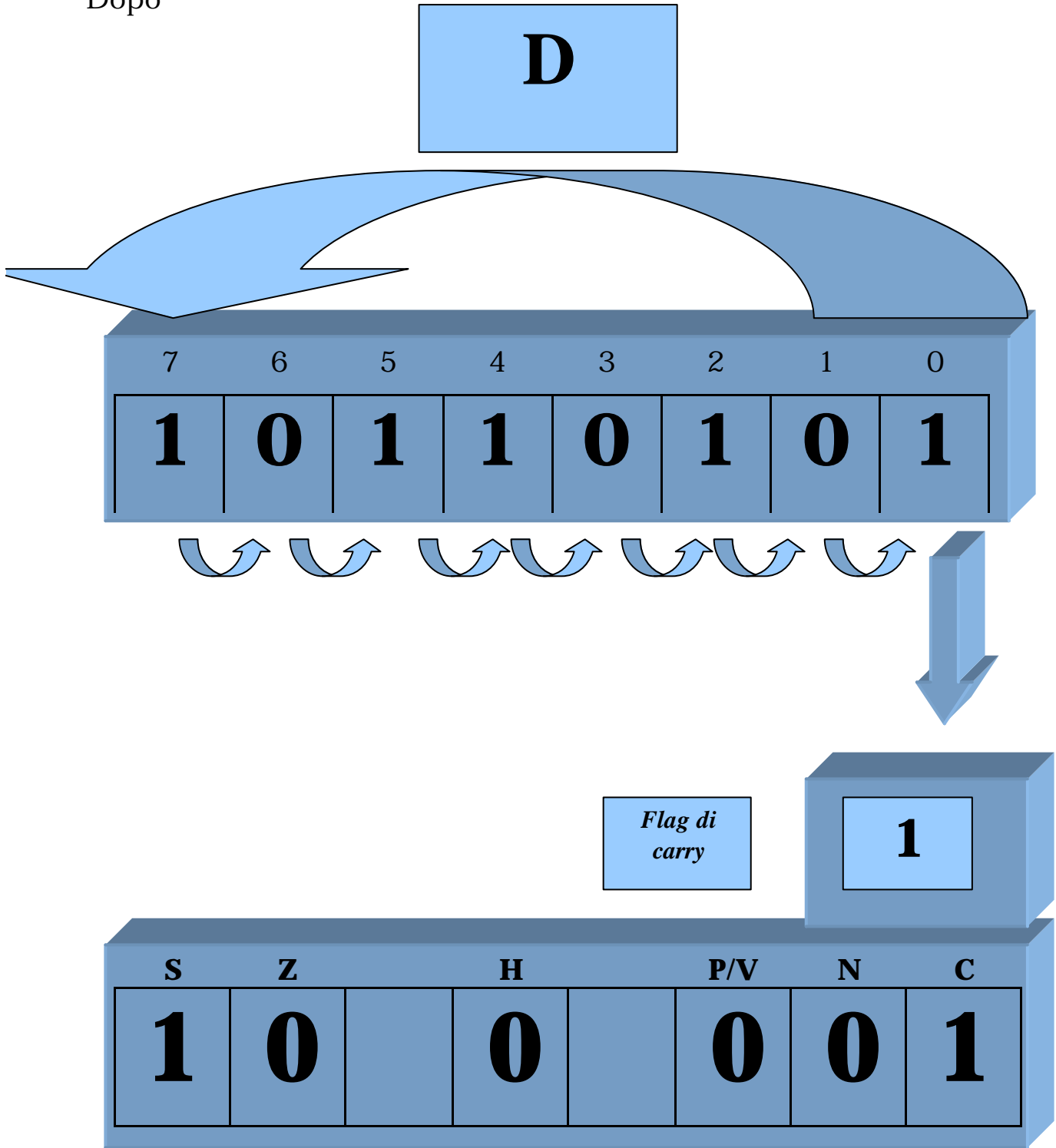
- ✓ Un registro ad otto bit del microprocessore A, B, C, D, E, H, L

Es. RLC D

Prima

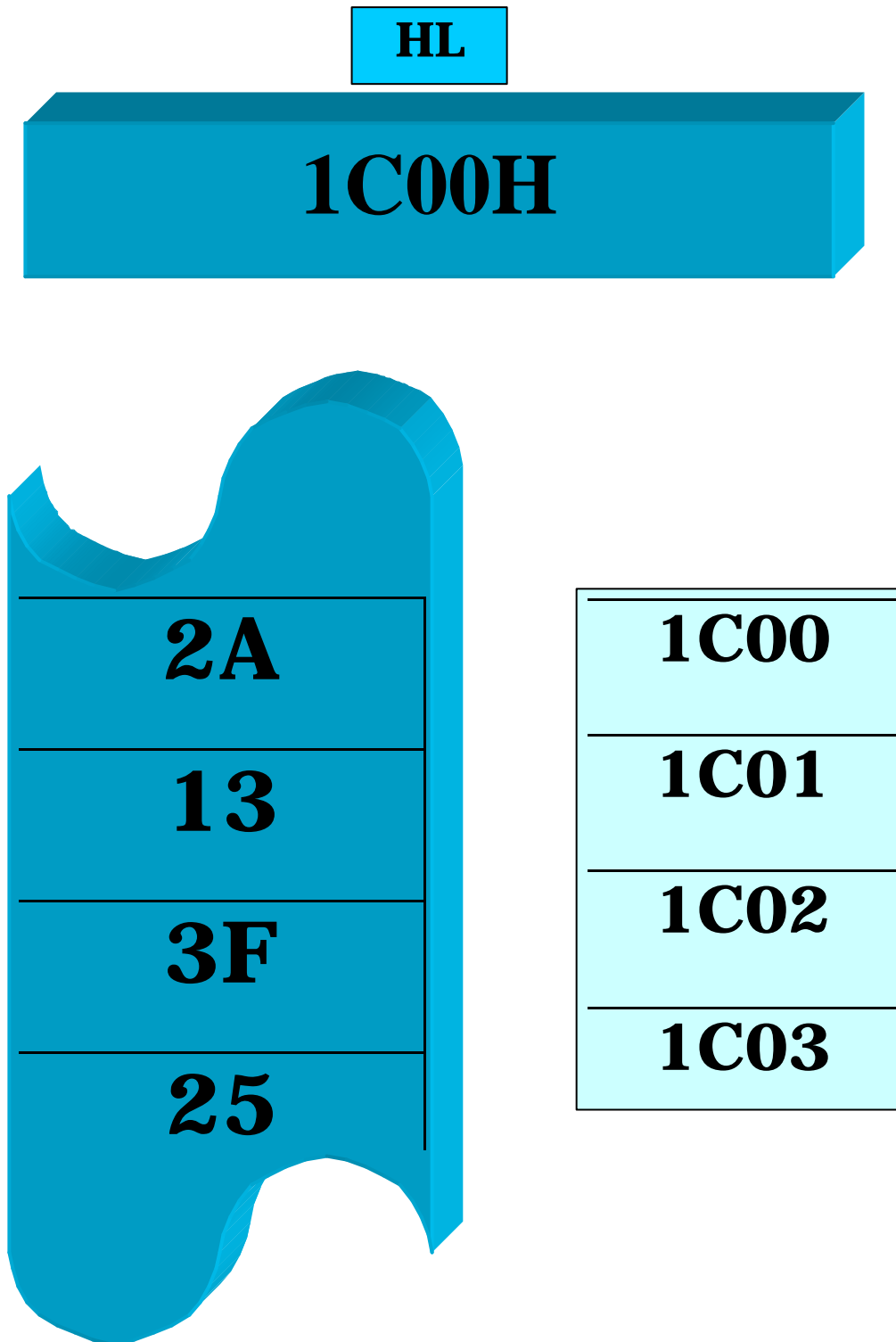


Dopo

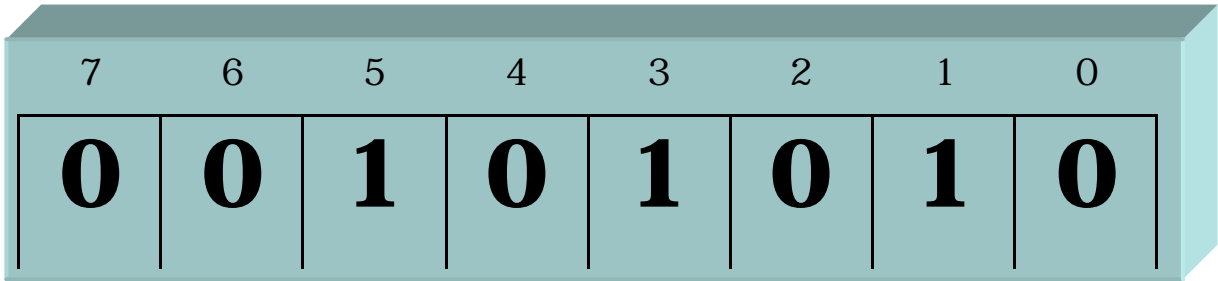
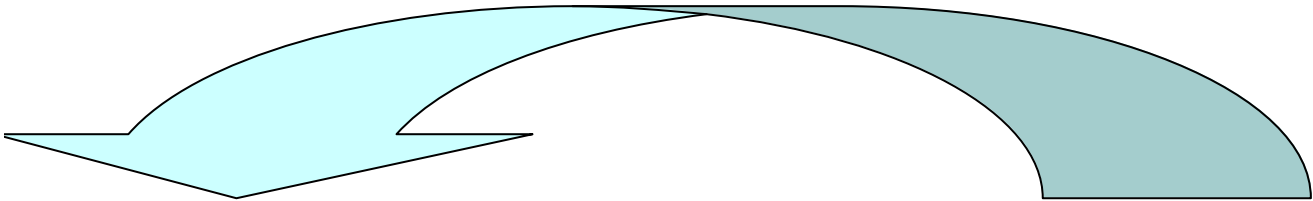


Il risultato della rotazione ha il bit 7 pari ad 1 quindi è un numero negativo (S=1), diverso da zero (Z=0), il numero di bit ad 1 è di cinque cioè è dispari (P=0), il bit 0 prima della rotazione era 1 (C=1).

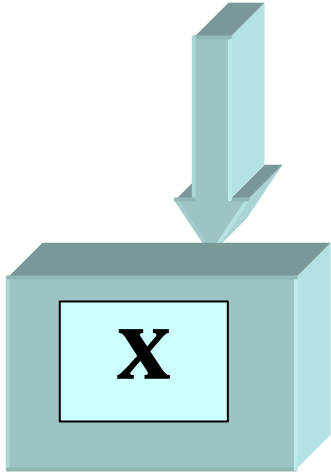
- ✓ L'operando s può essere il contenuto di una locazione di memoria il cui indirizzo è contenuto nel registro HL.



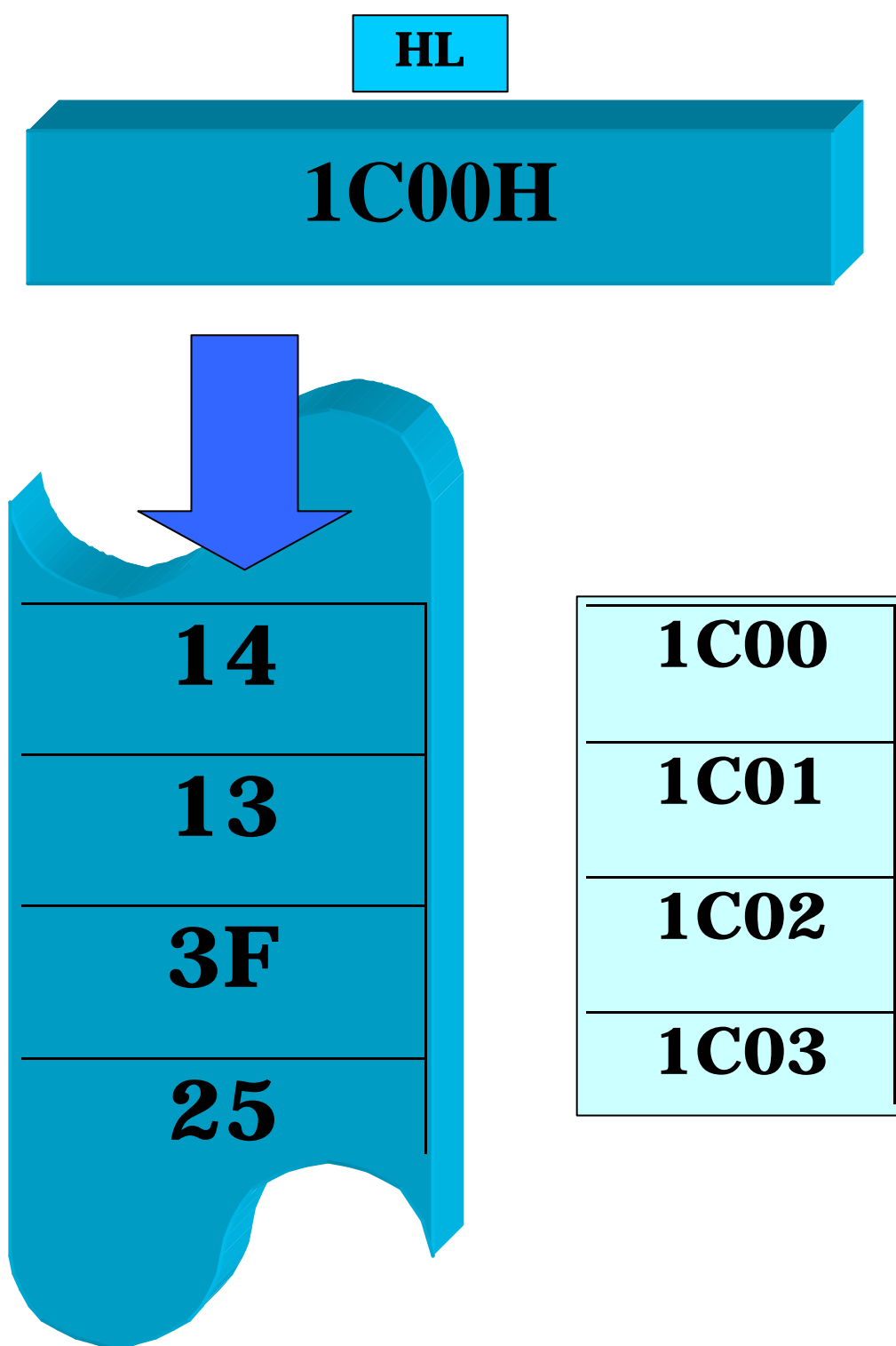
1C00H



Flag di carry

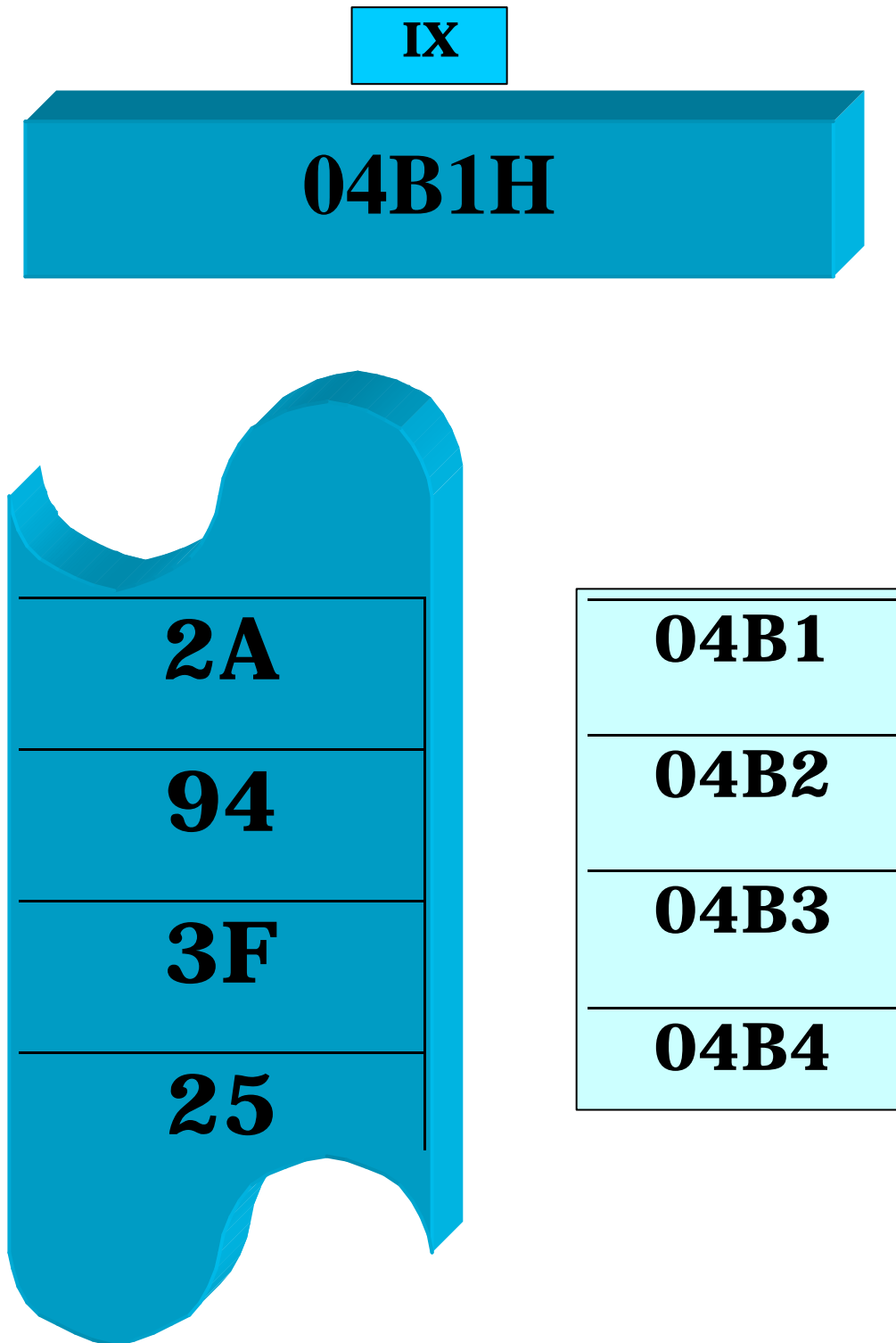


Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo ($S=0$), diverso da zero ($Z=0$), il numero di bit ad 1 è di tre cioè è dispari ($P=0$), il bit 0 prima della rotazione era 0 ($C=0$).

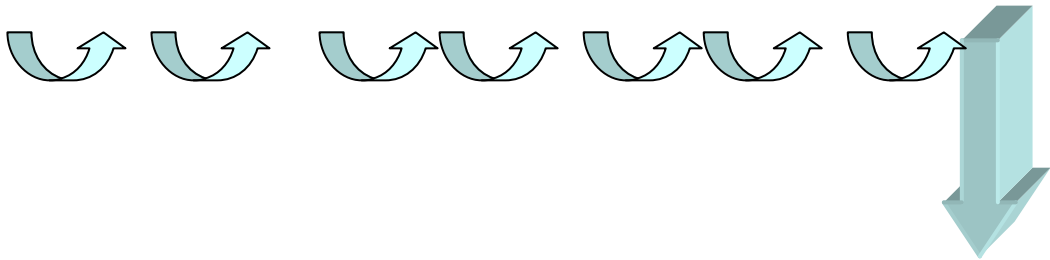
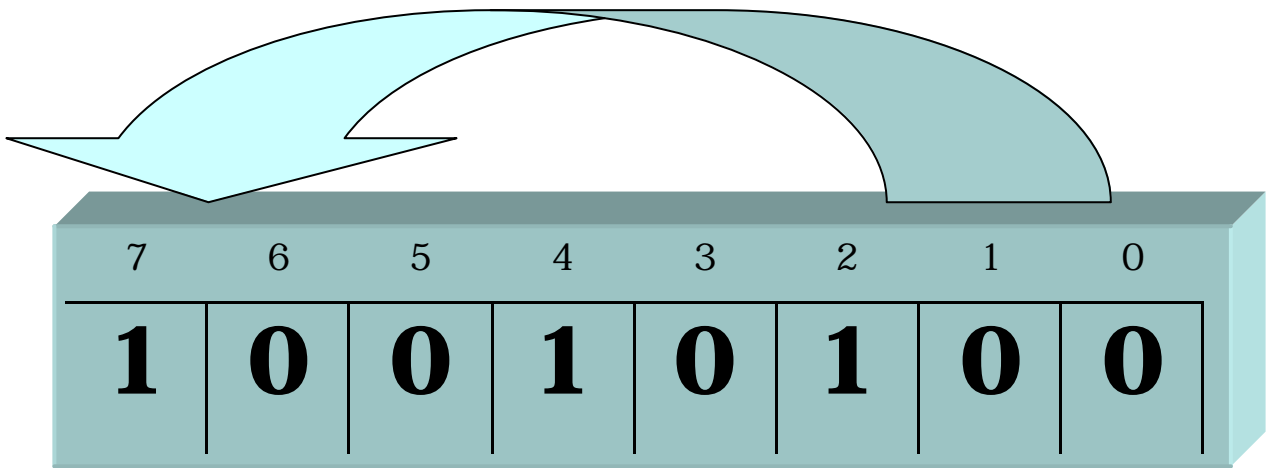


- ✓ L'operando s può essere il contenuto di una locazione di memoria indirizzata con indirizzamento indicizzato

Es. RLC (IX+1)



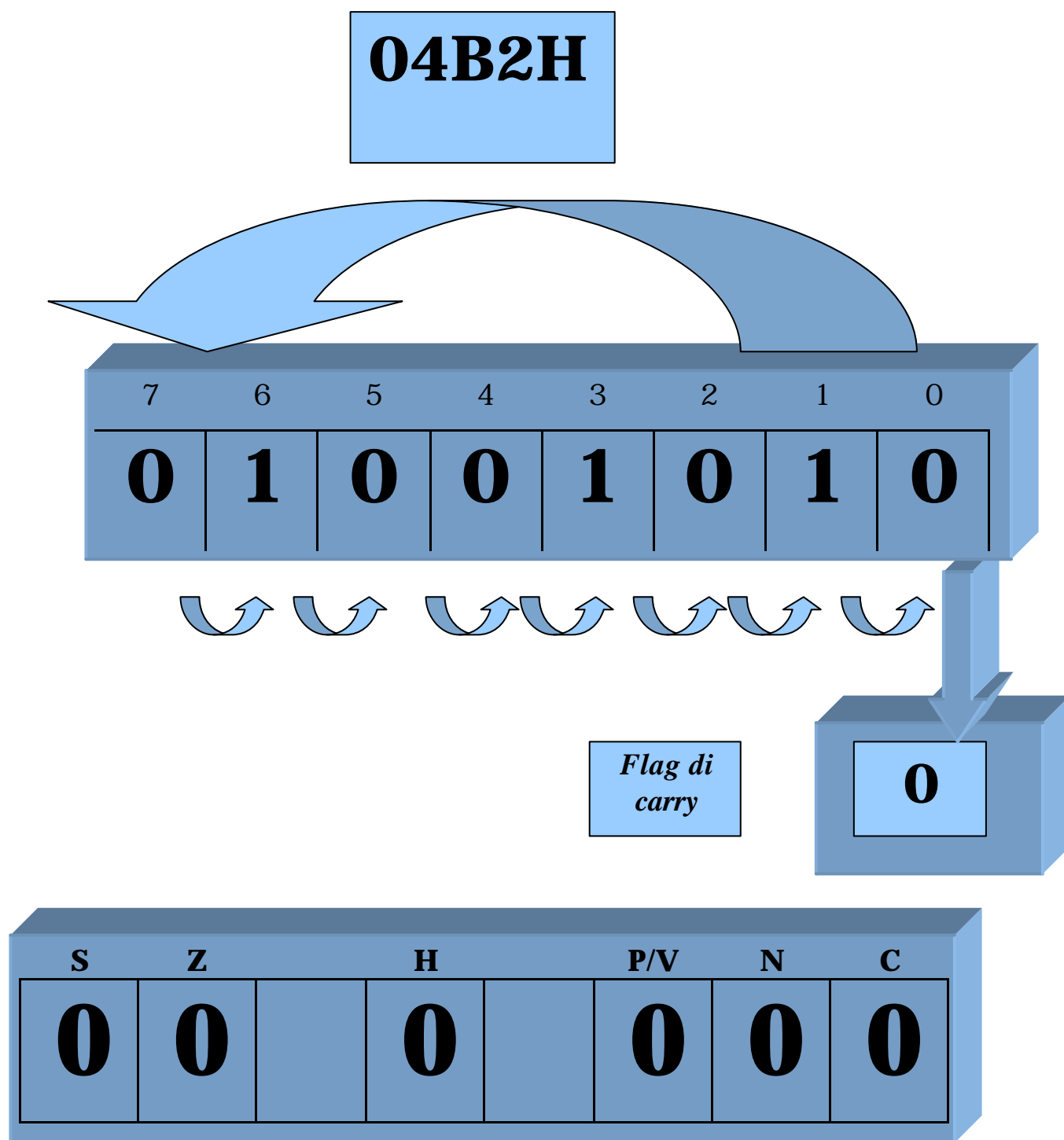
04B2H



Flag di carry

X

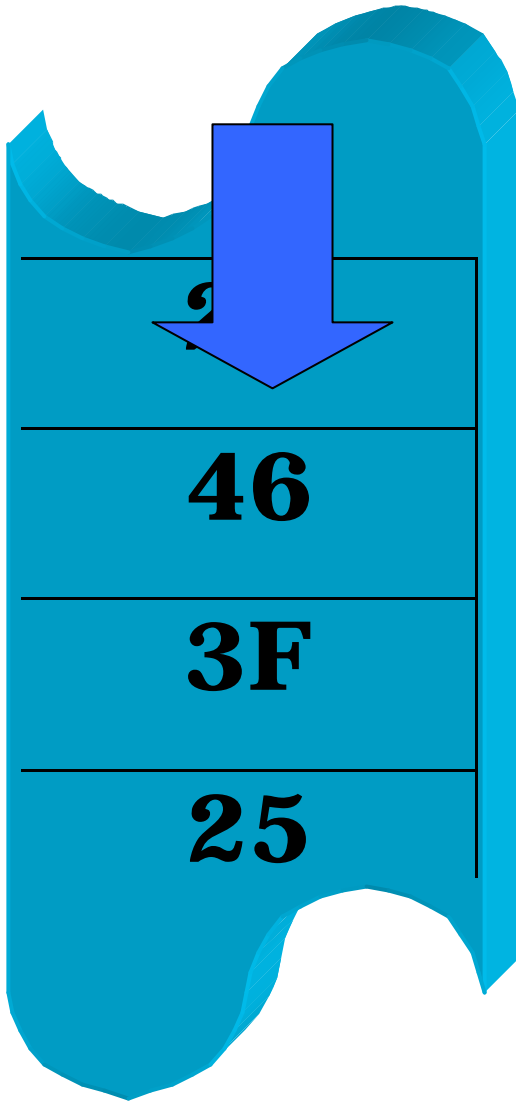
Dopo



Il risultato della rotazione ha il bit 7 pari ad 0 quindi è un numero positivo (S=0), diverso da zero (Z=0), il numero di bit ad 1 è di tre cioè è dispari (P=0), il bit 0 prima della rotazione era 0 (C=0).

IX

04B1H



04B1

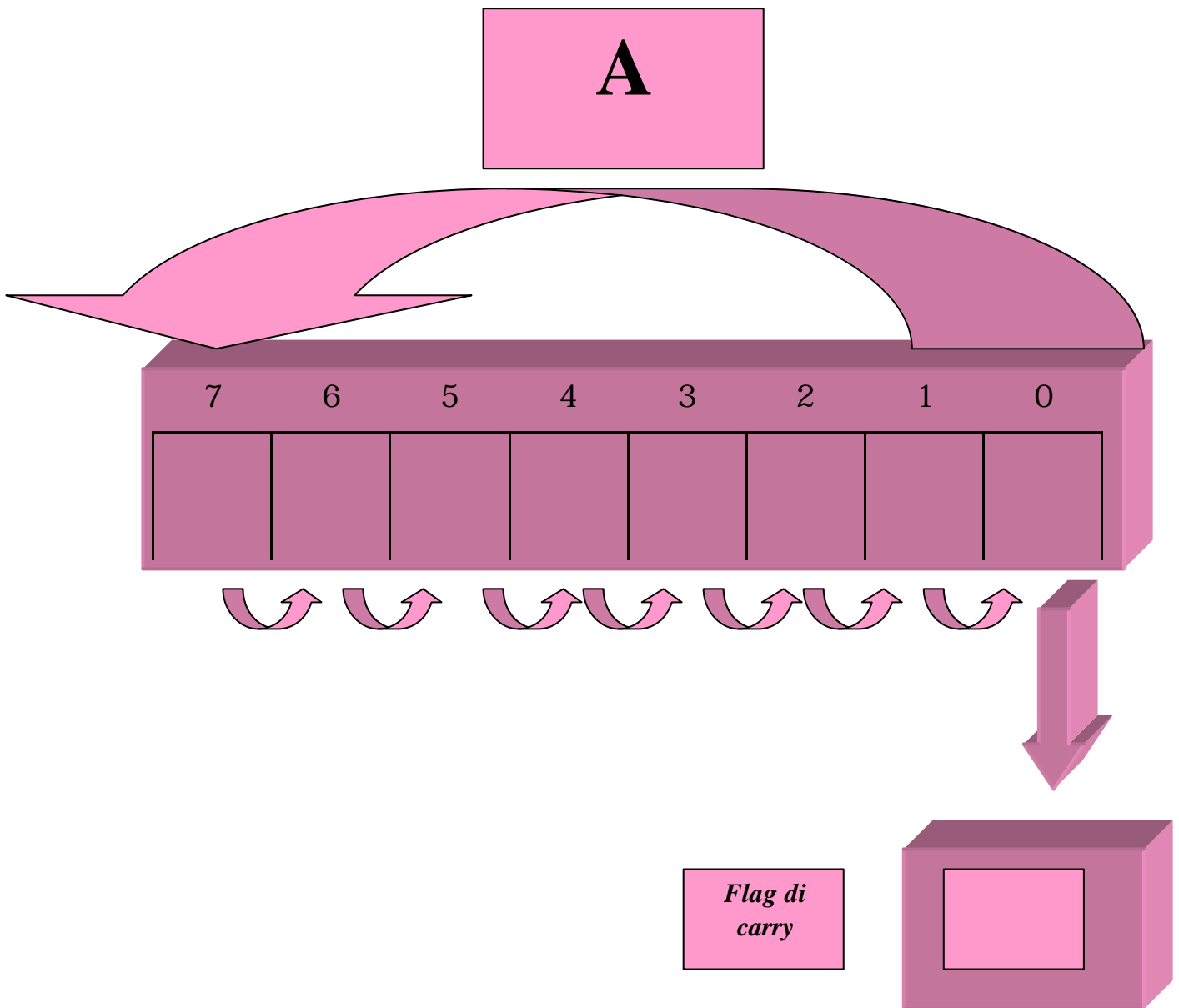
04B2

04B3

04B4

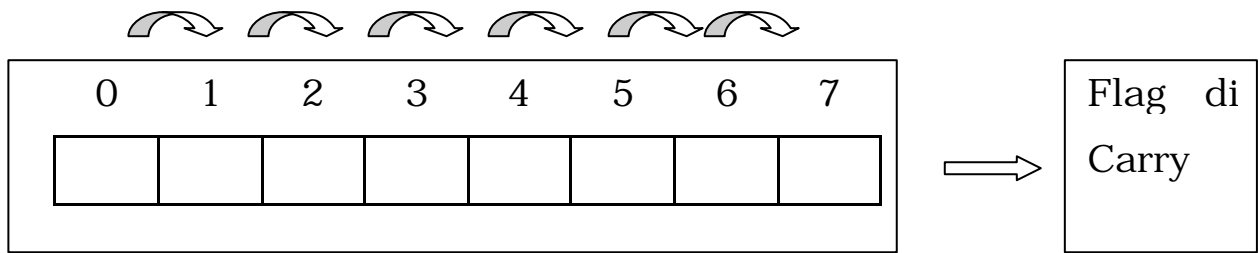
RRCA

Con quest'istruzione viene ruotato l'accumulatore verso destra, il bit zero va ad occupare il posto del bit 7 e contemporaneamente va ad occupare il flag di carry.

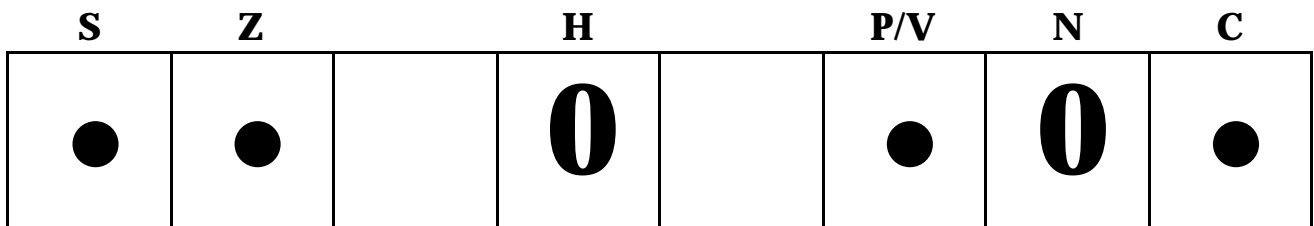


RR s

Quest'istruzione ruota a destra l'operando s: ogni bit occupa il posto di quello alla sua destra. Il bit zero occupa il posto del flag di carry, il vecchio contenuto del flag di carry occupa il posto del bit 7.



Modifiche del registro dei flag

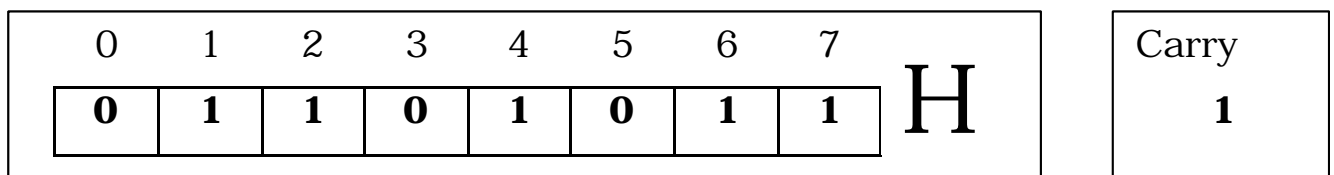


I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 0 dell'operando. I flag H e N sono sempre a zero

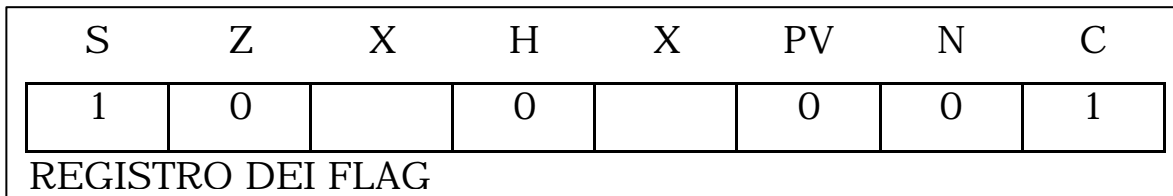
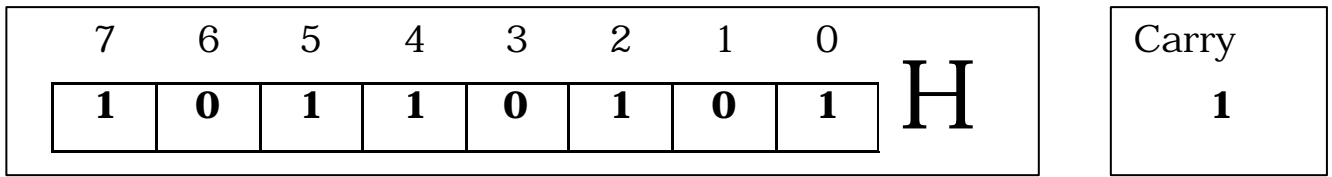
✓ L'operando s può essere un registro ad otto bit.

Es. RR H

Prima



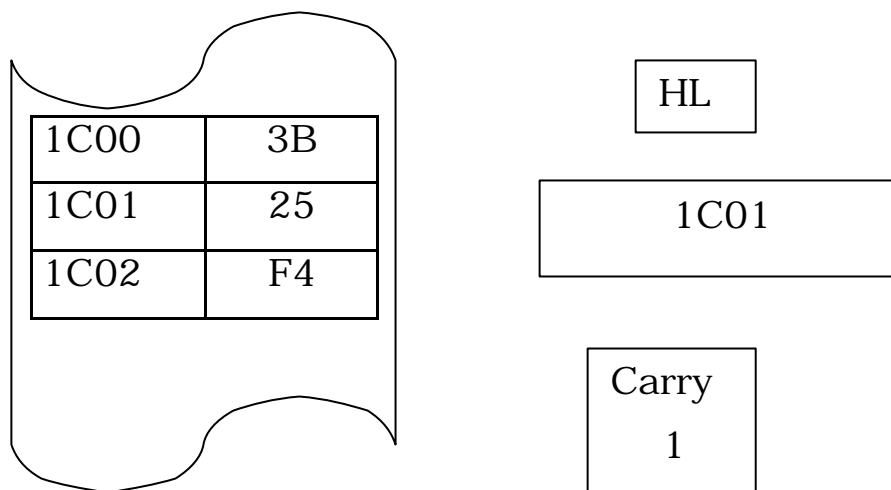
Dopo



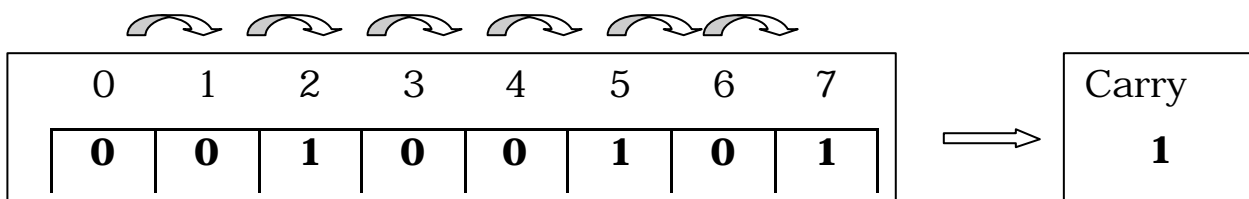
Il risultato della rotazione è un numero negativo (S=1), è un numero diverso da zero (Z=0), esso presenta un numero di bit ad uno pari a cinque, cioè dispari (PV=0), il bit zero prima della rotazione era pari a 1 (C=1).

- ✓ L'operando s può essere il contenuto di una locazione di memoria il cui indirizzo è contenuto nel registro HL (indirizzamento indiretto)

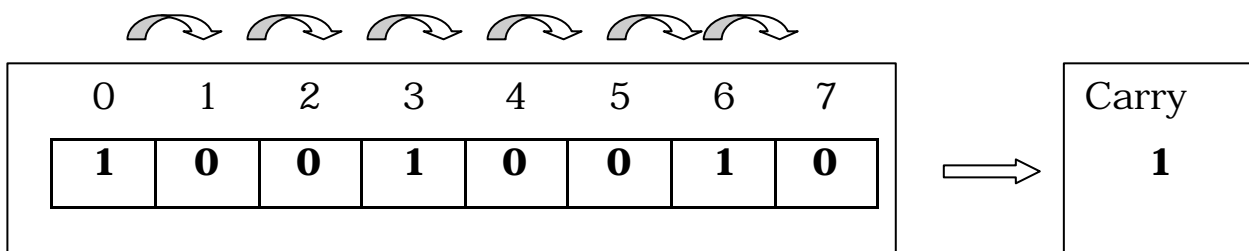
Es. RR (HL)



$$25|_{16} = 00100101|_2$$



Dopo



1C00	3B
1C01	52
1C02	F4

HL

1C01

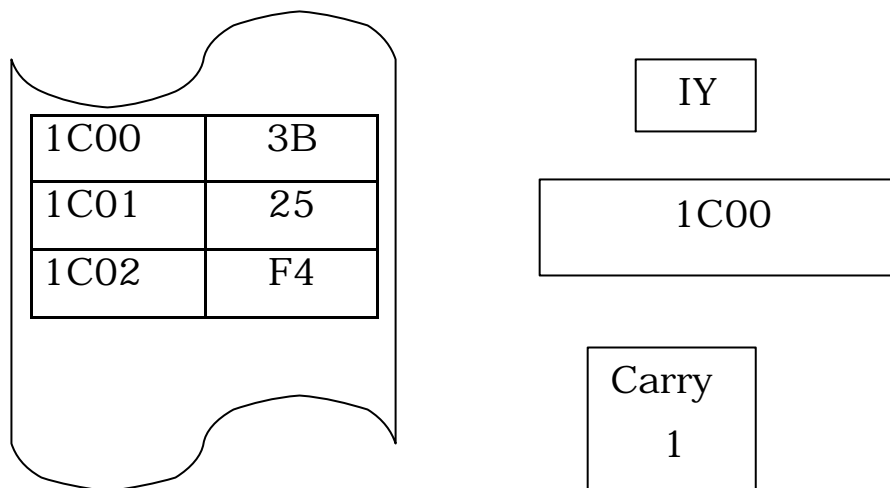
Carry
1

S	Z	X	H	X	PV	N	C
1	0		0		0	0	1
REGISTRO DEI FLAG							

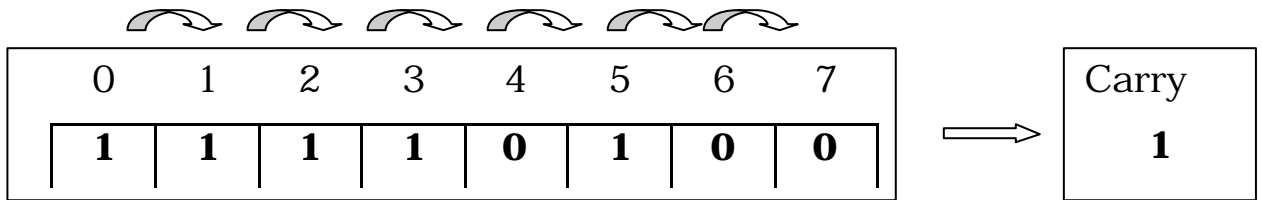
Il risultato della rotazione è un numero negativo (S=1), è un numero diverso da zero (Z=0), esso presenta un numero di bit ad uno pari a tre, cioè dispari (PV=0), il bit zero prima della rotazione era pari a 1 (C=1).

- ✓ L'operando s può essere il contenuto di una locazione di memoria il cui indirizzo si può ottenere sommando il contenuto del registro IX o IY ed uno spiazzamento (indirizzamento indicizzato)

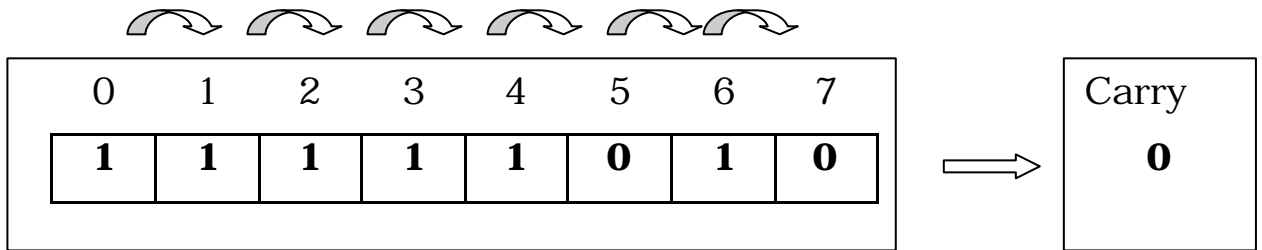
Es. RR (IY +2)



$$F4|_{16} = 11110100|_2$$



Dopo



1C00	3B
1C01	52
1C02	F8

IY

1C00

Carry
1

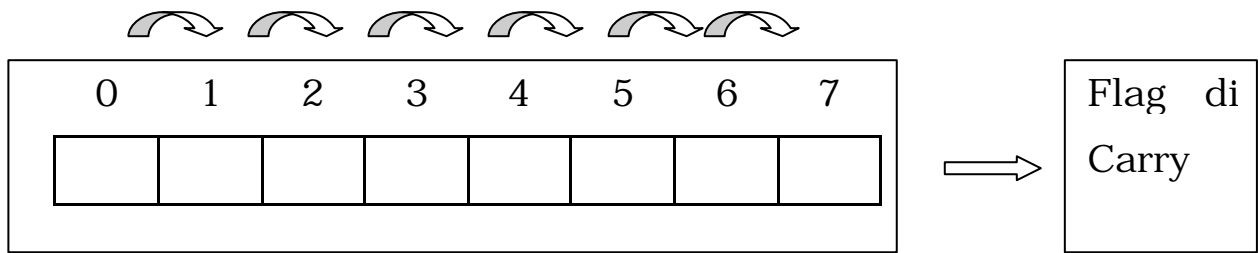
S	Z	X	H	X	PV	N	C
1	0		0		1	0	0

REGISTRO DEI FLAG

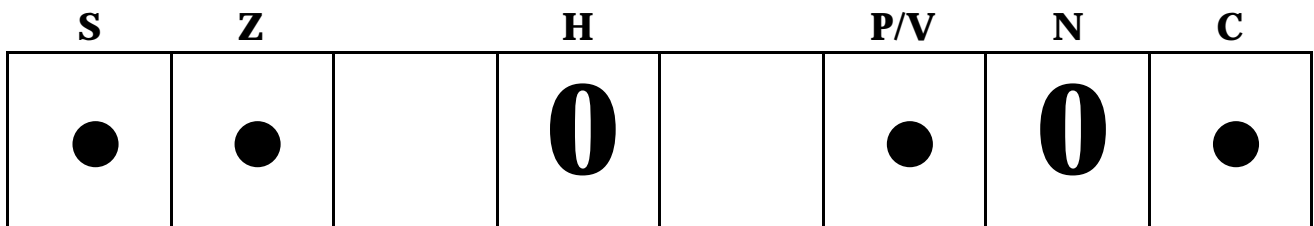
Il risultato della rotazione è un numero negativo (S=1), è un numero diverso da zero (Z=0), esso presenta un numero di bit ad uno pari a SEI, cioè pari (PV=1), il bit zero prima della rotazione era pari a 0 (C=0).

RRA

Quest'istruzione ruota a destra l'accumulatore: ogni bit occupa il posto di quello alla sua destra. Il bit zero occupa il posto del flag di carry, il vecchio contenuto del flag di carry occupa il posto del bit 7.



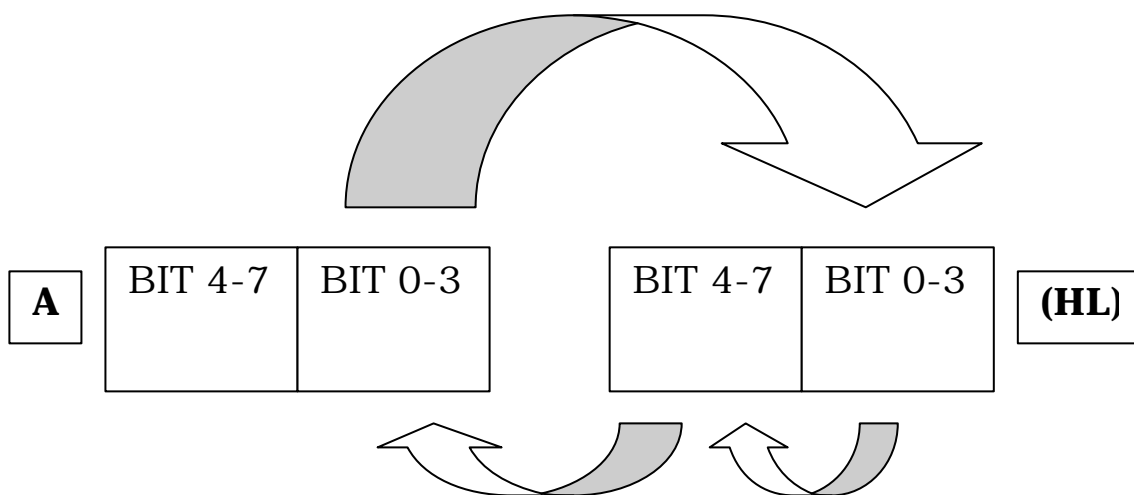
Modifiche del registro dei flag



I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità, C conterrà il bit 0 dell'operando. I flag H e N sono sempre a zero

RLD

Rotazione decimale a sinistra. L'operando è implicitamente una locazione di memoria il cui indirizzo è contenuto nel registro HL. Quest'istruzione presuppone che il contenuto dell'operando sia rappresentato in codice BCD, per cui ogni nibble rappresenta una cifra decimale. Il nibble inferiore dell'operando va a sostituire il nibble superiore dell'operando stesso. Il nibble superiore va a sostituire il nibble inferiore dell'accumulatore, e questo va a sostituire il nibble inferiore dell'operando.

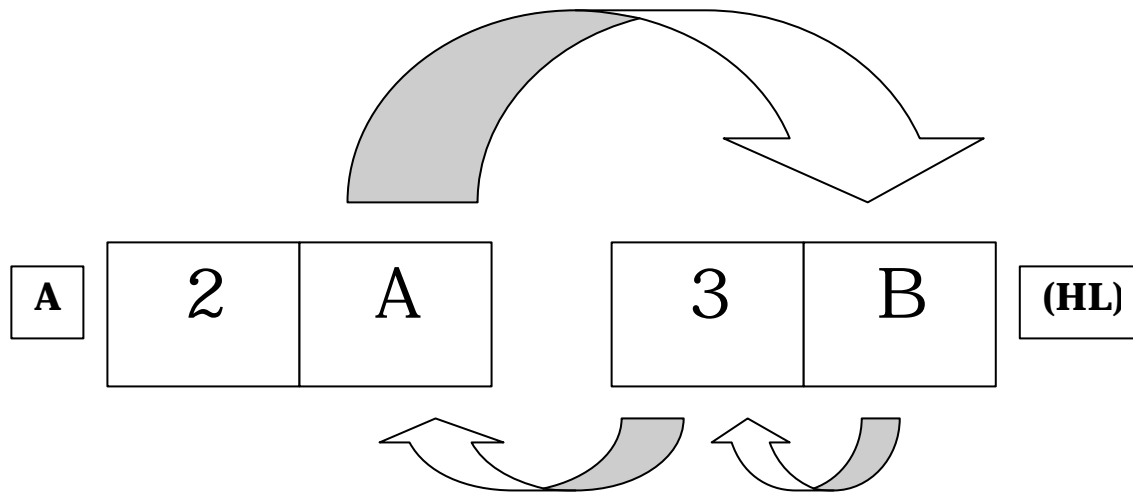
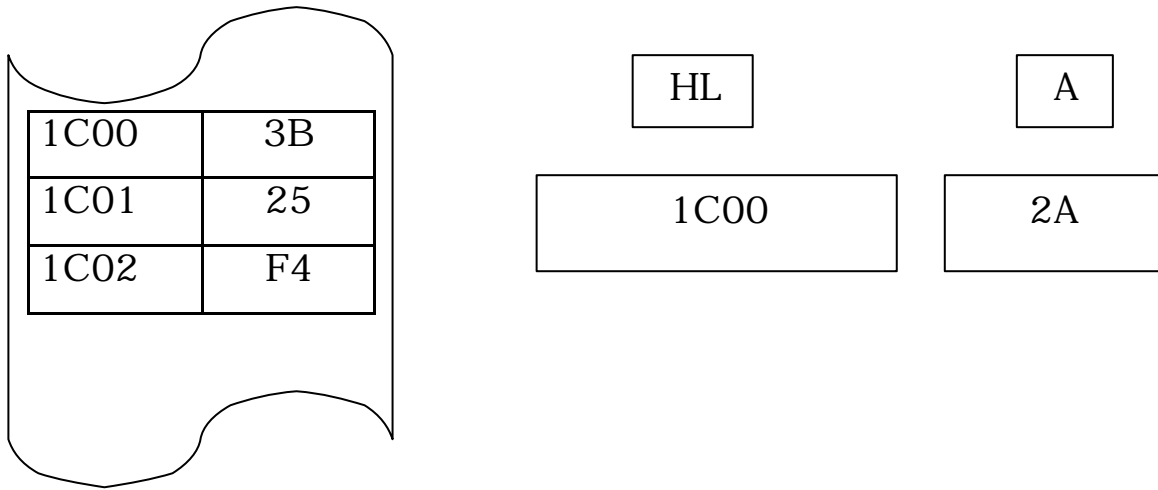


Modifiche del registro dei flag

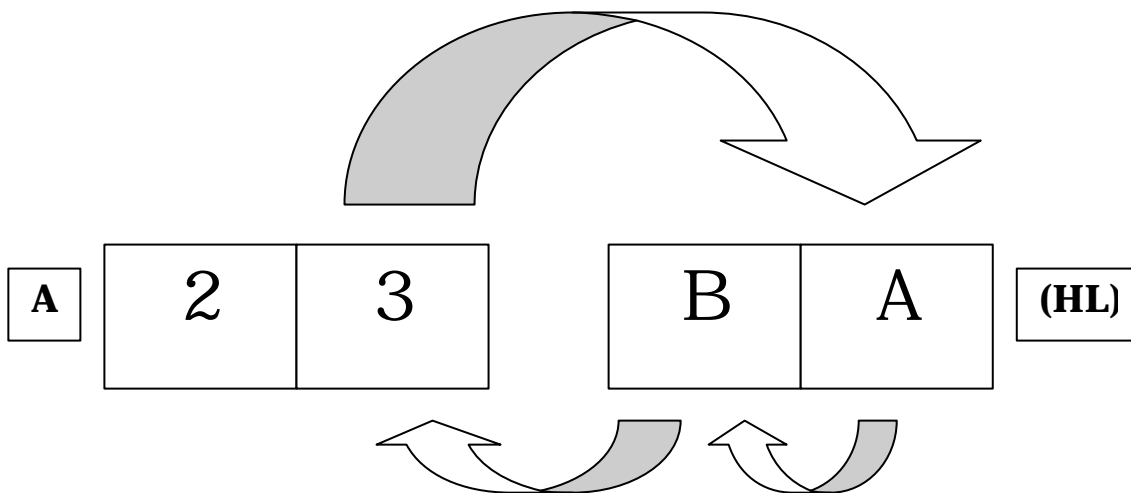
S	Z		H		P/V	N	C
●	●		0		●	0	

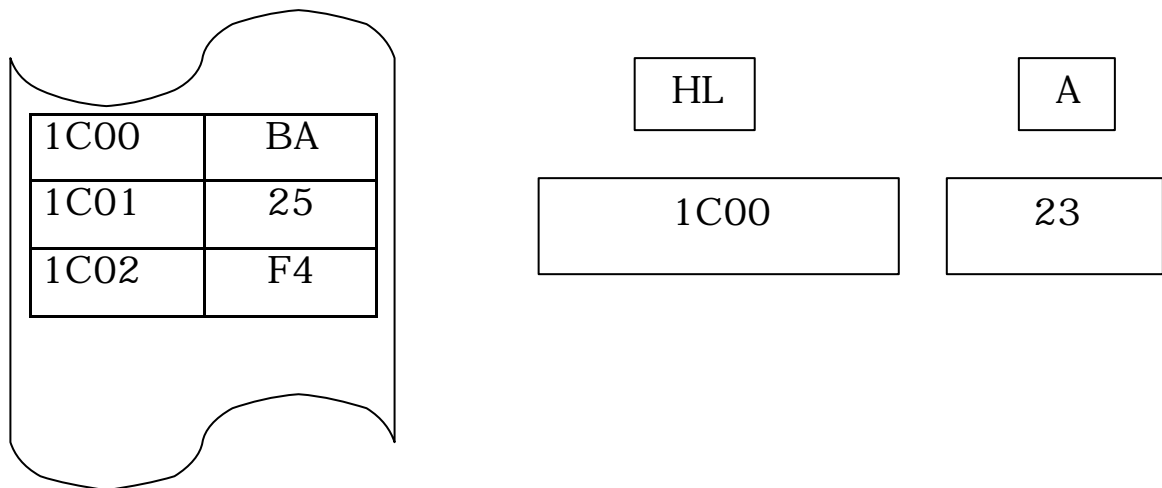
I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità. I flag H e N sono sempre a zero. C non viene modificato.

Es.



Dopo





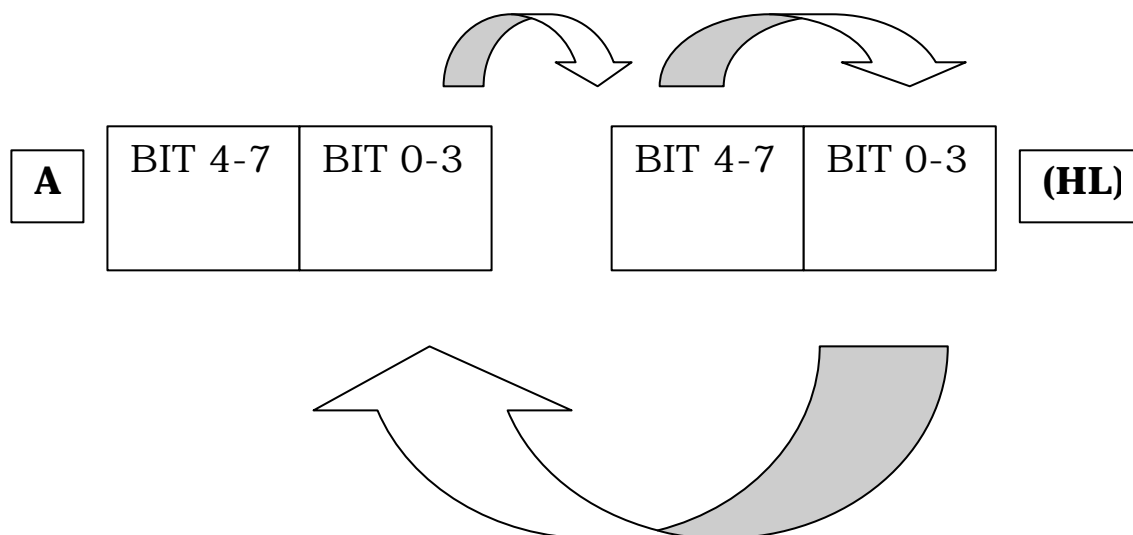
$$\mathbf{BA}_{16} = \mathbf{10111010}_2$$

S	Z	X	H	X	PV	N	C
1	0		0		0	0	
REGISTRO DEI FLAG							

Il risultato della rotazione è un numero negativo ($S=1$), è un numero diverso da zero ($Z=0$), esso presenta un numero di bit ad uno pari a CINQUE, cioè dispari ($PV=0$).

RRD

Rotazione decimale a destra. L'operando è implicitamente una locazione di memoria il cui indirizzo è contenuto nel registro HL. Quest'istruzione presuppone che il contenuto dell'operando sia rappresentato in codice BCD, per cui ogni nibble rappresenta una cifra decimale. Il nibble superiore dell'operando va a sostituire il nibble inferiore dell'operando stesso. Il nibble inferiore va a sostituire il nibble inferiore dell'accumulatore, e questo va a sostituire il nibble superiore dell'operando.

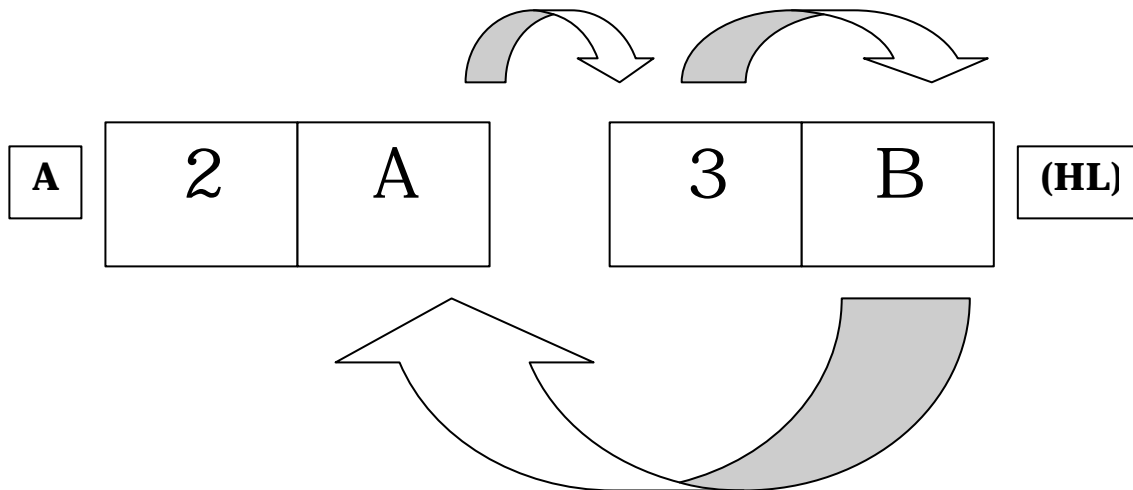
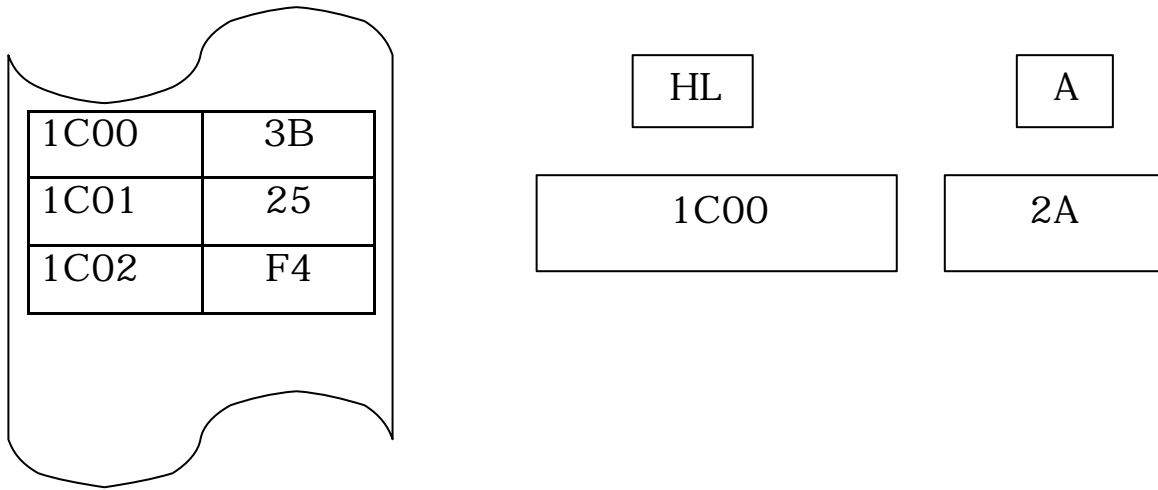


Modifiche del registro dei flag

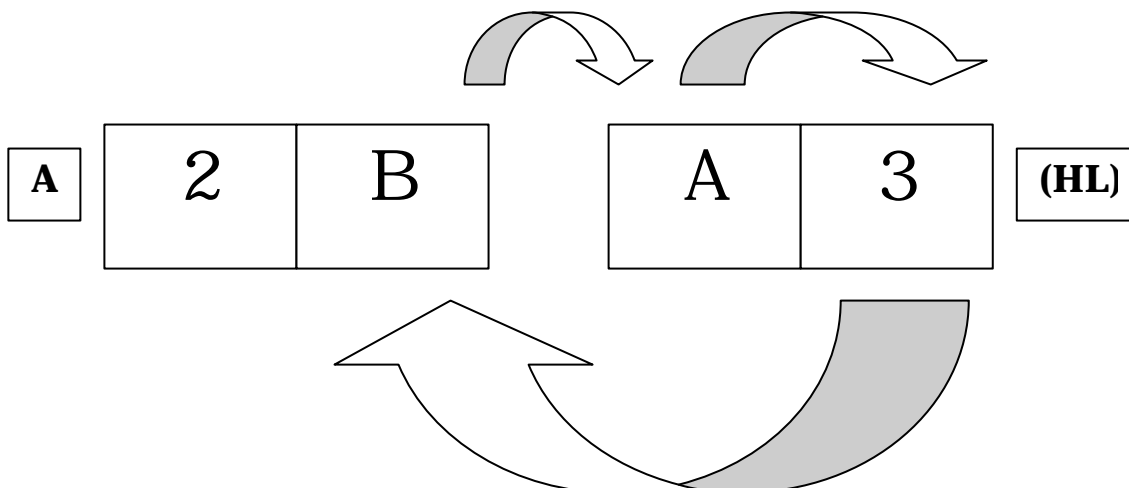
S	Z		H		P/V	N	C
●	●		0		●	0	

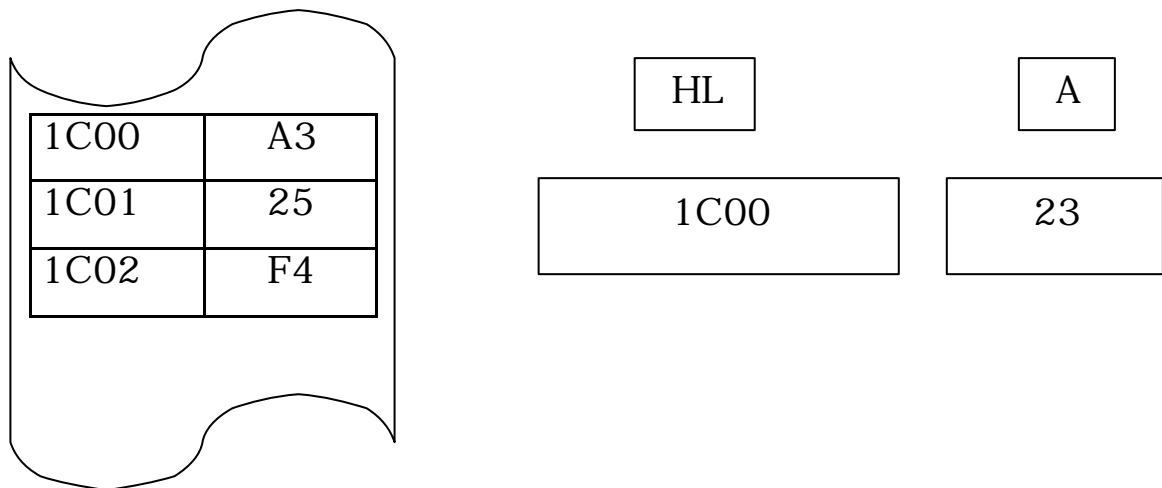
I flag S e Z sono modificati secondo la solita legge, il flag PV funziona da flag di parità. I flag H e N sono sempre a zero. C non viene modificato.

Es.



Dopo





$$\mathbf{BA}_{16} = \mathbf{10111010}_2$$

S	Z	X	H	X	PV	N	C
1	0		0		0	0	

REGISTRO DEI FLAG

Il risultato della rotazione è un numero negativo ($S=1$), è un numero diverso da zero ($Z=0$), esso presenta un numero di bit ad uno pari a CINQUE, cioè dispari ($PV=0$).