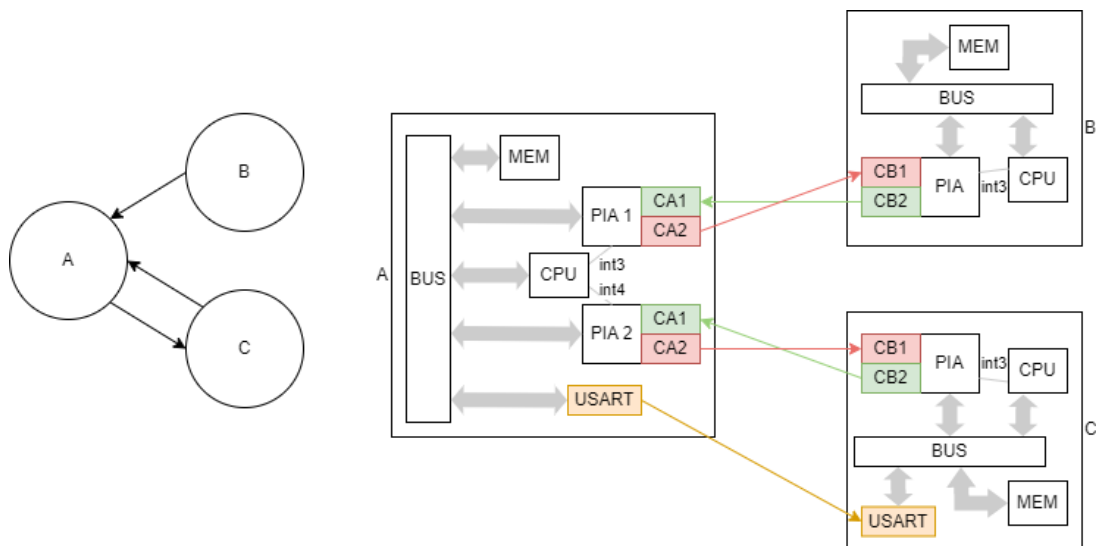




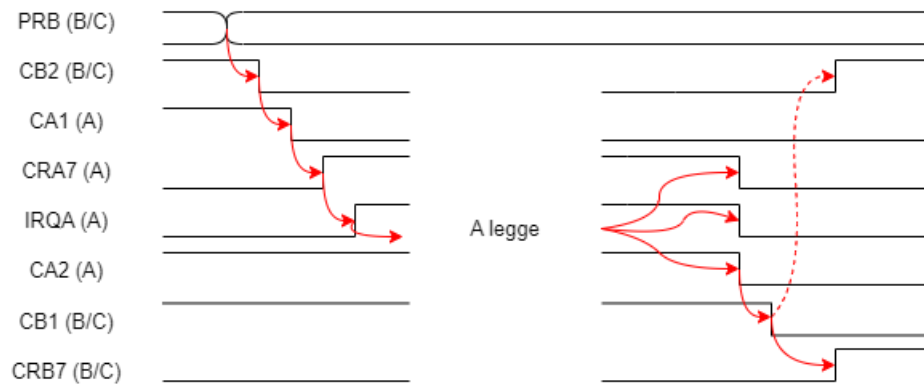
Traccia

Un sistema è composto da tre unità, A, B e C. A riceve una sequenza di M messaggi da N caratteri ciascuno mediante una periferica parallela P1. Se un messaggio è composto da tutti caratteri nulli, A invia a C un carattere di controllo \$FF mediante una seriale e continua a ricevere nuovi messaggi. Si suppone che dopo aver ricevuto tale carattere, C effettui una elaborazione, ed **eventualmente** invia ad A un carattere \$FF mediante una seconda parallela P2. Alla ricezione di tale carattere da C, A deve sospendere la ricezione dei caratteri, dopo aver completato l'eventuale messaggio in corso.

Architettura



Protocolli



Mappa della memoria

	MEMORIA RAM	
\$8000	COUNT_CAR	COUNT_MESS
\$8002	CHECK	MESSAGE
\$8004	MESSAGE	MESSAGE
\$8006	MESSAGE	
	...	
\$8100	MAIN	
	...	
\$8500	ISR_B	
	...	

	MEMORIA ROM
\$6C	00 00 85 00
\$70	00 00 87 00

Pseudocodice

```

1  //----- NODO A -----
2  MAIN(){
3      inizializza PIA 1 in lettura;
4      configurazione seriale in TX;
5      inizializza PIA 2 in lettura;
6      Enable stato utente;
7      Enable interruzioni;
8      while(true);           //attendo l'interruzione del sistema B
9  }
10
11  ISR_B(){
12      if(TAS lock){
13          char = Leggi da PIA1;
14          count_car ++ ;
15          if (char == 0){
16              num_zero ++ ;
17          }

```

```

18         if(count_car == N){
19             count_car = 0;
20             count_mess ++ ;
21             if(count_mess == M OR C_attivo == 1){
22                 disattiva PIA1;
23             }
24             else if(num_zero == N){
25                 num_zero = 0;
26                 Invia $FF a C con USART;
27             }
28         }
29         TAS unlock;
30     }
31     else{
32         sospendi_B = 1
33     }
34     return from ISR;
35 }
36
37 ISR_C(){
38     if(TAS lock){
39         car_ctrl = Leggi da PIA2;
40         C_attivo = 1;
41         if(sospendi_B == 1){
42             sospendi_B = 0;
43             char = Leggi da PIA1;
44             count_car ++ ;
45             if(char == 0){
46                 num_zero ++ ;
47             }
48         }
49         TAS unlock;
50     }
51     else{
52         C_attivo = 1;
53     }
54     return from ISR;
55 }

```

Programma assembly

```

1
2 IL PROGRAMMA NON E' CORRETTO
3
4         ORG      $8000
5 N       EQU      4
6 M       EQU      10
7
8 COUNT_CAR DC.B    0
9 COUNT_MESS DC.B    0
10 MESSAGGE DS.B    N
11

```

```

12      ORG      $8100
13  PIA1D      EQU      $2004      *registro dato di PIA1
14  PIA1C      EQU      $2005      *registro controllo di PIA1
15  PIA2D      EQU      $2006      *registro dato di PIA2
16  PIA2C      EQU      $2007      *registro controllo di PIA2
17
18  MAIN      JSR      SETUP_PIA
19            MOVE.W   SR,D0        *salvo SR in D0 per modificarlo
20            ANDI.W   #$D8FF,D0    *imposto lo stato utente e le interruzioni in D0
21            MOVE.W   D0,SR        *pongo il nuovo SR nel registro
22  LOOP      END      LOOP
23
24  SETUP_PIA  MOVE.B   #0,PIA1C      *invio con PIA1
25            MOVE.B   #$00,PIA1D
26            MOVE.B   #%00100101,PIA1C
27
28            MOVE.B   #0,PIA2C      *ricevo con PIA2
29            MOVE.B   #$FF,PIA2D
30            MOVE.B   #%00100101,PIA2C
31            RTS
32
33  ISR_B      ORG      $8500
34            MOVEA.L  A0,-(A7)
35            MOVEA.L  A1,-(A7)
36            MOVEA.L  A2,-(A7)
37            MOVEA.L  A3,-(A7)
38            MOVE.L   D0,-(A7)
39            MOVEA.L  A4,-(A7)
40
41            MOVE.L   #PIA1D,A0
42            MOVE.L   #PIA1C,A1
43            MOVE.L   #PIA2D,A2
44            MOVE.L   #PIA2C,A3
45            MOVE.L   #MESSAGE,A4
46
47            MOVE.B   COUNT_CAR,D0
48            MOVE.B   (A0),(A4,D0)
49            ADD.B    #1,D0
50            MOVE.B   D0,COUNT_CAR
51            CMP.B    #N,D0
52            BNE      FINE
53            MOVE.B   #0,COUNT_CAR
54            MOVE.B   COUNT_MESS,D0
55            ADD.B    #1,D0
56            MOVE.B   D0,COUNT_MESS
57            CMP.B    #M,D0
58            BEQ      ELSE
59            CMP.B    #1,CHECK
60            BEQ      ELSE
61  INVIA      MOVE.B   COUNT_CAR,D0
62            MOVE.B   (A4,D0),(A2)
63            ADD.B    #1,D0
64            CMP.B    #N,D0
65            BNE      INVIA

```

```
66 ELSE          disattiva PIA
67
68 FINE          MOVEA.L (A7)+,A4
69              MOVE.L (A7)+,D0
70              MOVEA.L (A7)+,A3
71              MOVEA.L (A7)+,A2
72              MOVEA.L (A7)+,A1
73              MOVEA.L (A7)+,A0
74              RTE
75
76
77              END      MAIN
```
