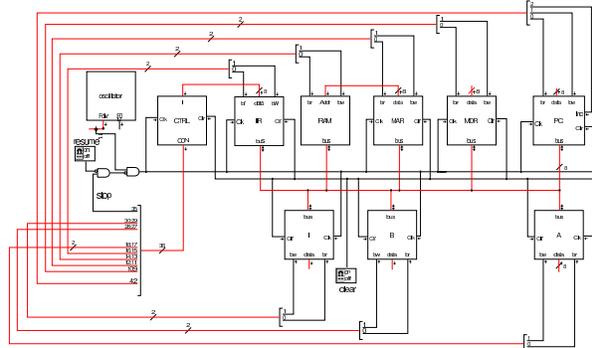


Nella terza versione della CPU dimostrativa, vengono aggiunti due registri che per il momento non hanno alcuno scopo particolare: *A* e *B*. Tali registri sono realizzati nello stesso modo di *I*, *MDR*, *MAR* e *IR*.

Figura u108.1. Il bus della CPU nella sua terza fase realizzativa.



Nel codice che descrive i campi del bus di controllo, si aggiungono quelli seguenti, i quali servono specificatamente a gestire i registri *A* e *B*:

```
field a_br[17];           // A <-- bus
field a_bw[18];           // A --> bus
field b_br[27];           // B <-- bus
field b_bw[28];           // B --> bus
```

Nell'elenco dei codici operativi si aggiungono istruzioni nuove e lo stesso poi nella descrizione del microcodice:

```
op move_mdr_a {
  map move_mdr_a : 5;           // move MDR to A
  +0[7:0]=5;
  operands op_0;
};
op move_a_mdr {
  map move_a_mdr : 6;           // move A to MDR
  +0[7:0]=6;
  operands op_0;
};
op move_mdr_b {
  map move_mdr_b : 7;           // move MDR to B
  +0[7:0]=7;
  operands op_0;
};
op move_b_mdr {
  map move_b_mdr : 8;           // move B to MDR
  +0[7:0]=8;
  operands op_0;
};
```

```
begin microcode @ 0
...
move_mdr_a:
  a_br mdr_bw;                 // A <-- MDR
  ctrl_start ctrl_load;       // CNT <-- 0
//
move_a_mdr:
  mdr_br a_bw;                 // MDR <-- A
  ctrl_start ctrl_load;       // CNT <-- 0
//
move_mdr_b:
  b_br mdr_bw;                 // B <-- MDR
  ctrl_start ctrl_load;       // CNT <-- 0
//
move_b_mdr:
  mdr_br b_bw;                 // MDR <-- B
  ctrl_start ctrl_load;       // CNT <-- 0
...
end
```

©2- 2013.11.11 -- Copyright © Daniele Giacomini -- appunti2@gmail.com http://informaticalibera.net

Figura u108.5. Corrispondenza con il contenuto della memoria che rappresenta il microcodice (la coppia *m1* e *m2* dell'unità di controllo).

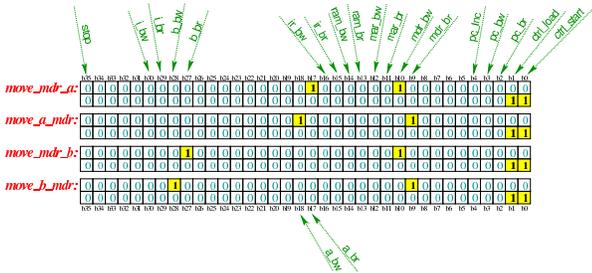


Tabella u108.6. Elenco delle macroistruzioni aggiunte in questa versione della CPU dimostrativa.

Sintassi	Descrizione
move_mdr_a	Copia il contenuto del registro <i>MDR</i> nel registro <i>A</i> .
move_a_mdr	Copia il contenuto del registro <i>A</i> nel registro <i>MDR</i> .
move_mdr_b	Copia il contenuto del registro <i>MDR</i> nel registro <i>B</i> .
move_b_mdr	Copia il contenuto del registro <i>B</i> nel registro <i>MDR</i> .

Come esempio viene proposto il macrocodice seguente:

```

begin macrocode @ 0
start:
    load_imm #data_1
    move_mdr_a
    load_imm #data_2
    move_mdr_b
stop:
stop
data_1:
.byte 17
data_2:
.byte 11
end
    
```

In pratica, viene caricato nel registro *MDR* il valore corrispondente all'indirizzo in cui si trova l'etichetta '*data_1*:' (facendo i conti si tratta dell'indirizzo 7); successivamente, il valore di *MDR* viene copiato nel registro *A*; quindi viene caricato nel registro *MDR* quanto contenuto nell'indirizzo di memoria corrispondente all'etichetta '*data_2*:' (indirizzo 8) e poi copiato nel registro *B*. Il file completo che descrive le memorie per Tkgate dovrebbe essere disponibile all'indirizzo allegati/circuiti-logici/scpu-sub-c.gm

Figura u108.8. Contenuto della memoria RAM. Le celle indicate con «xx» hanno un valore indifferente.

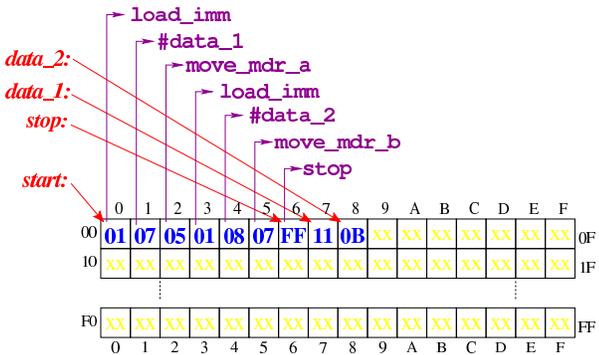
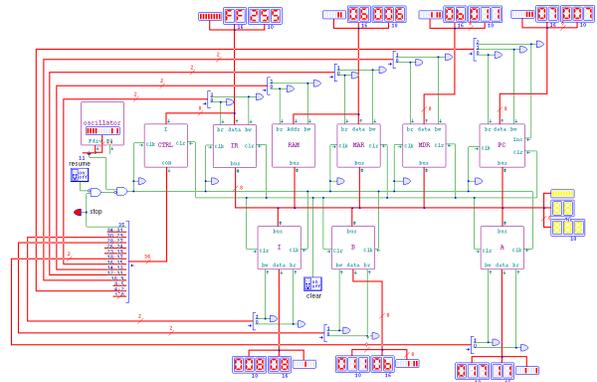


Figura u108.9. Situazione conclusiva del bus dati, dopo l'esecuzione delle istruzioni copia nei registri *A* e *B*. Video: <http://www.youtube.com/watch?v=9qVsCKmxcdk>



Dalle istruzioni introdotte in questa versione della CPU dimostrativa, si può intendere che i dati contenuti nei registri possano essere copiati soltanto con la mediazione del registro *MDR*; pertanto non esiste un'istruzione *move_a_b*. Questa è una semplificazione per evitare di dover dichiarare tante istruzioni nel macrocodice, ma in condizioni normali, tale scelta non sarebbe utile.

