



МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Лабораторна робота №4

з дисципліни «Технології проектування
комп'ютерних систем»
на тему: «Регістрова пам'ять»

Виконав:
студент 4-го курсу
факультету ІОТ
групи ІО-41
Демчик В. В.
НЗК 4111

Перевірив:
проф. Сергієнко А. М.

Київ 2017

Тема: Регістрова пам'ять.

Мета та основні завдання роботи: оволодіти знаннями і практичними навичками з проектування пристроїв пам'яті, таких як регістрова ОЗП (FM). Лабораторна робота також служить для оволодіння навичками програмування та налагодження опису RAM на мові VHDL.

Завдання на лабораторну роботу: розробити FM блок за наведеними нижче умовами:

Розрядність слів – 16 біт.

Об'єм пам'яті – 64 слова.

Каналів – 2.

Заборонено запис в перший регістр.

Виконати описання поведінкової моделі. Провести аналіз отриманих графіків роботи схем.

Хід проектування:

Виконаємо опис входів та виходів RAM:

CLK	Синхросигнал
WR	Запис
AB	Адреса каналу B
AD	Адреса каналу D
B	Дані каналу B
D	Дані каналу D

Оскільки за варіантом FM повинна вміщувати 64 слів, то для їх індексації знадобляться ($63_{10} = 11\ 1111_2$) - 6-бітні адреси.

Код програми:

Поведінкова модель:

```
use CNetwork.all;
entity FM is
port(
  CLK:in BIT; -- synchro
  WR:in BIT; -- write
  --B for read from FM
  --D for write to FM
  AB:in BIT_VECTOR(5 downto 0);-- channel B address
  AD:in BIT_VECTOR(5 downto 0);-- channel D address
  B: out BIT_VECTOR (15 downto 0);-- channel B data
  D: in BIT_VECTOR(15 downto 0);-- channel D data
end FM;

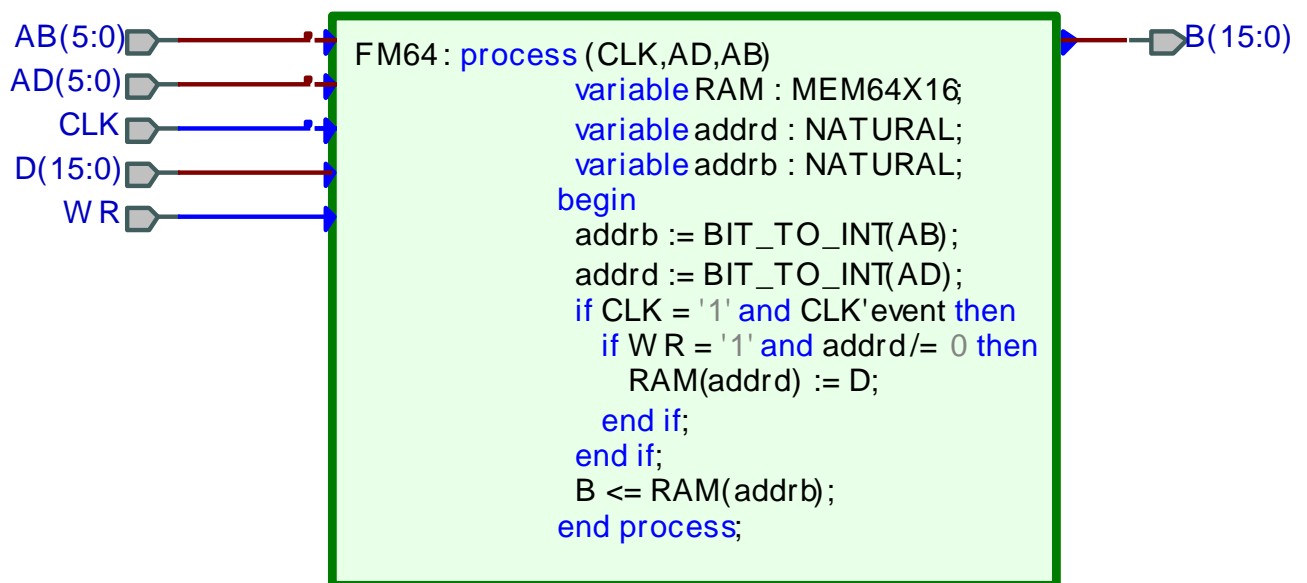
architecture BEH of FM is
type MEM64X16 is array(0 to 63) of BIT_VECTOR(15 downto 0);
signal addr,do: BIT_VECTOR(15 downto 0);
begin
  ---- Block of register memory -----
```

```

FM64:process(CLK,AD,AB)
variable RAM: MEM64x16;
variable addrd,addrb:NATURAL;
begin
    addrd:= BIT_TO_INT(AB);
    addrb:= BIT_TO_INT(AD);
    if CLK='1' and CLK'event then
        if WR = '1' and addrd /= '0' then
            RAM(addrd):= D; -- read data from D channel
        end if;
    end if;
    B<= RAM(addrb); -- write data to channel B
end process;
end BEH;

```

Згенерована схема на основі описаної архітектури:



Design Unit Header

```

library fm;
use fm.cnetwork.all;

```

Architecture Declaration

```

--Added by Active-HDL. Do not change code inside this section.
type MEM64X16 is array(0 to 63) of BIT_VECTOR(15 downto 0);
--End of extra code.

```

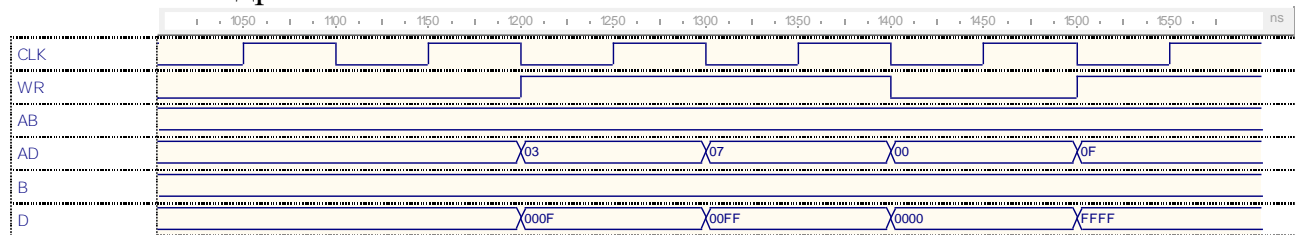
Результати симуляції:

Використовуючи канал D запишемо:

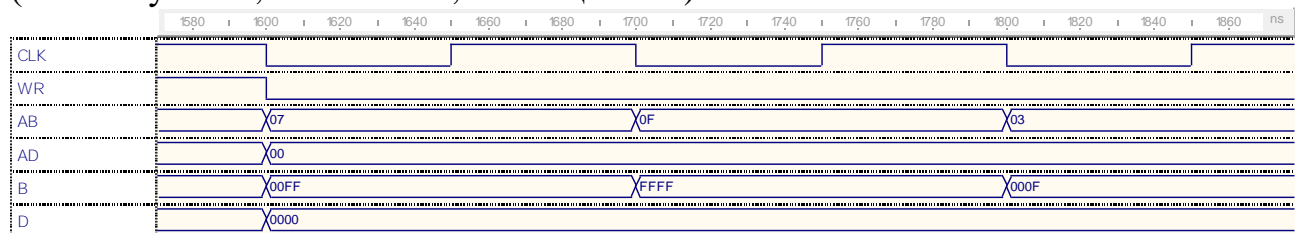
за адресою 03 слово 000F,

за адресою 07 слово 00FF, і через такт (шляхом відключення W)

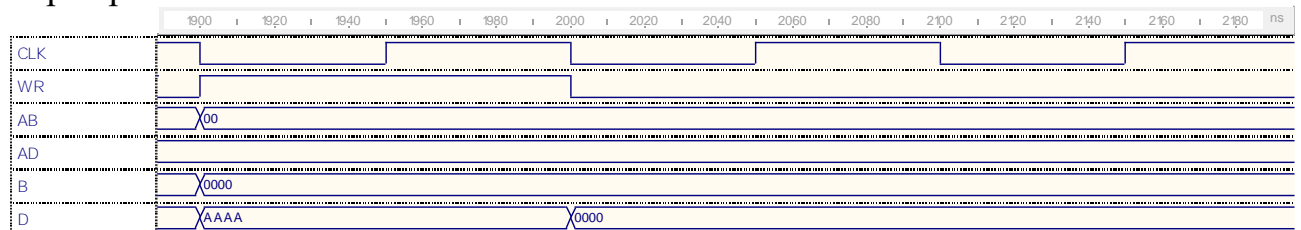
за адресою 0F слово FFFF.



Тепер за допомогою каналу B прочитаємо ці слова в тасованому порядку (спочатку за 07, потім за 0F, і в кінці за 03):

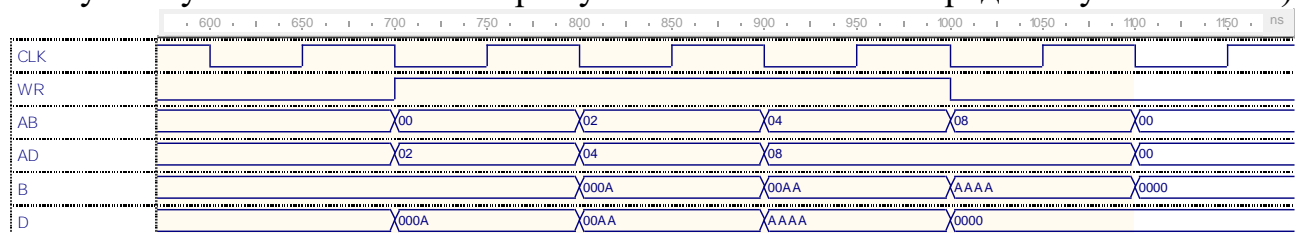


Спробуємо здійснити запис слова AAAA до нульового регістру, і через такт перевірити його вміст:



Як бачимо, перезапису не відбулося.

Спробуємо одночасно записати через канал D три нові значення (000A за адресою 02, 00AA за адресою 04, AAAA за адресою 08, при цьому на кожному наступному такті на каналі B отримувати записане на попередньому такті слово)



Остання діаграма демонструє головну перевагу регістрової пам'яті – швидкість. Оскільки наявні два канали, то за один такт можливо вести як запис нових значень, так і видачу поточних. Крім того, це дає можливість організувати конвеєрний принцип роботи – шляхом подачі однакових адрес на канали B та D ми можемо за один такт отримувати в канал B попереднє значення комірки пам'яті та одразу із каналу D записати в нього наступне.