Computer architecture

MIPS PROCESSOR
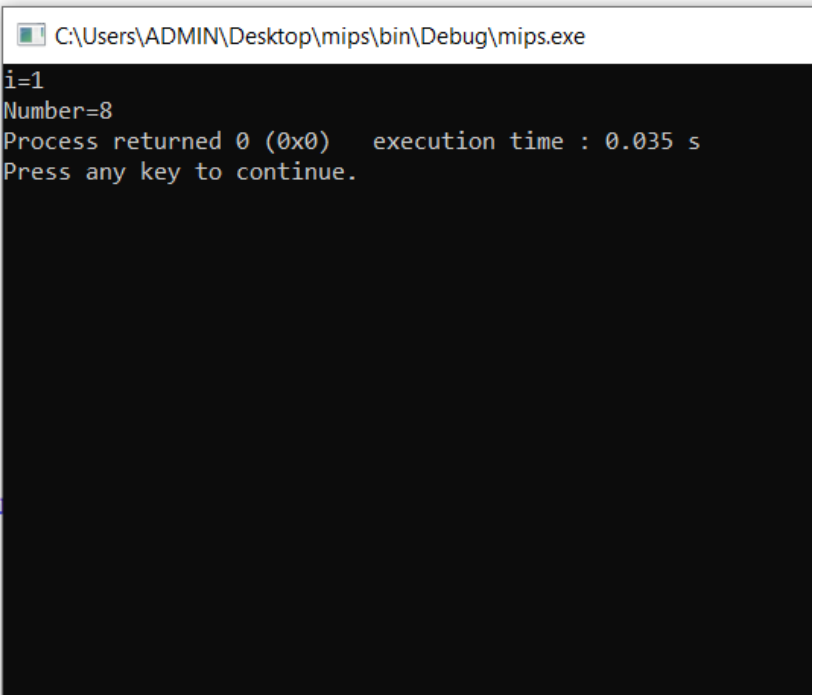
Nermeen Mohammed

Alaa Morsy

C Code :

```c
1   #include <stdio.h>
2   #include <stdlib.h>
3
4   int main()
5   {
6       int Number=7;
7       int i;
8       i=Number;
9
10      while(i>1)
11      {
12          i=i-2;
13      }
14      if(i!=0)
15      {
16          Number=Number+1;
17      }
18      printf("i=%d \nNumber=%d",i,Number);
19
20      return 0;
21  }
22
```

```c
1   #include <stdio.h>
2   #include <stdlib.h>
3
4   int main()
5   {
6       int Number=7;
7       int i;
8       i=Number;
9
10      while(i>1)
11      {
12          i=i-2;
13      }
14      if(i!=0)
15      {
16          Number=Number+1;
17      }
18      printf("i=%d \nNum
19
20      return 0;
21  }
22
```

```
C:\Users\ADMIN\Desktop\mips\bin\Debug\mips.exe
i=1
Number=8
Process returned 0 (0x0)   execution time : 0.035 s
Press any key to continue.
```

The code checks if the input number is odd or even.it adds 1 if the number is odd.

We store the input number in another variable ' i ' and subtract 2 till the ' i ' become 0 or 1.

If ' i ' equal 1 so the number is odd and if ' i ' is 0 so the number is even.

if the input number is odd the program will add 1 to it.

Assembly code:

```
Edit    Execute

mips2.asm*

1   .text
2   main:
3   #s0=number ,$t0=i ,$t1=1
4   addi $s0,$0,7   #number=7
5   add $t0,$s0,$0 #i=number
6   addi $t1,$0,1
7
8   while:
9   slt $t2,$t1,$t0 #if (1<i) ,$t2=1 ,else $t2=0
10  beq $t2,$0,done_while #if ($t2=1)
11  addi $t0,$t0,-2 #i=i-2
12  j while
13
14  done_while:
15  beq $t0,$0,done #if(i=0) "even number"
16  add $s0,$s0,$t1 #number=number+1
17  sw $s0,26($0)   #store the output in memory
18
19  done:
20
```

Registers	Coproc 1	Coproc 0		
Name	Number	Value		
$zero	0	0		
$at	1	0		
$v0	2	0		
$v1	3	0		
$a0	4	0		
$a1	5	0		
$a2	6	0		
$a3	7	0		
$t0	8	1		
$t1	9	1		
$t2	10	0		
$t3	11	0		
$t4	12	0		
$t5	13	0		
$t6	14	0		
$t7	15	0		
$s0	16	8		
$s1	17	0		
$s2	18	0		
$s3	19	0		
$s4	20	0		
$s5	21	0		
$s6	22	0		
$s7	23	0		
$t8	24	0		
$t9	25	0		
$k0	26	0		
$k1	27	0		
$gp	28	268468224		
$sp	29	2147479548		
$fp	30	0		
$ra	31	0		
pc		4194344		
hi		0		
lo		0		

Registers	Coproc 1	Coproc 0
Name	Number	Value
$8 (vaddr)	8	26
$12 (status)	12	65299
$13 (cause)	13	20
$14 (epc)	14	4194340

Modelsim :

- **Testbench**
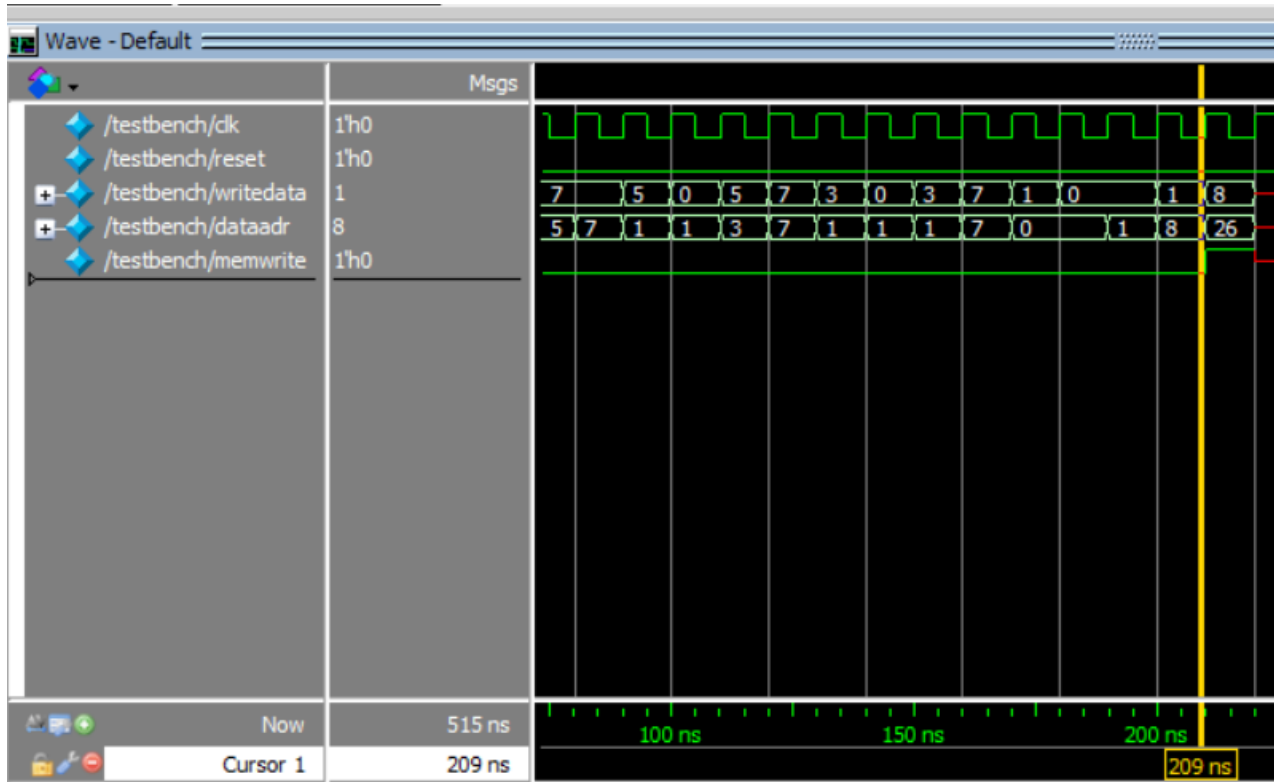
- **Register file**
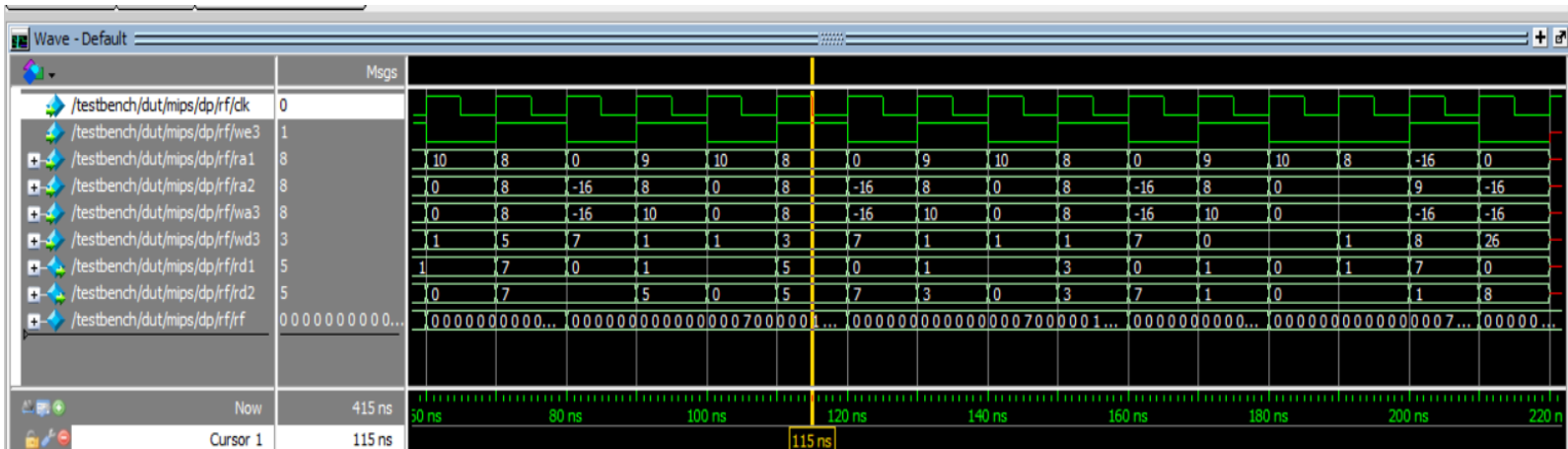
time	r_en	ral	rd1	wa3	wd3
200	1	16	7	16	8

At time 200 ns , the write enable "we3" is 1.

So, it writes 8 in register address 16 -> $S0 which means that the program run successfully

Knowing that no. 7 is odd then add a one to it which make it 8 and store it to the memory.