

メモリの記述

reg文を使ってメモリの宣言も行います。

```
reg [15:0] dmem [0:255];
```

幅16ビット、深さ256のメモリ宣言

```
assign do = dmem[daddr];
```

アドレスdaddrからのデータ読み出し

```
always @(posedge clk)
  if(we) dmem[daddr] <= ddataout;
```

we=1の時のクロック立ち上がりでデータの書き込み
reg同様、常にalways文中で<=で行う

2番地の上位8ビットは？

```
dmem[2][15:8]
```

メモリは通常、合成の対象としない→テストベンチで記述
原則として深さは2のべき乗にする。しなくてもいいが無駄になる

テストベンチでのメモリの初期化

メモリの初期化はinitial文中で、\$readmemh(“ファイル名”,メモリ名);で行います。readmembは2進数、readmemhは16進数で書きます。

```
reg [`DATA_W-1:0] dmem[0:`DEPTH-1];
```

```
reg [`INST_W-1:0] imem[0:`DEPTH-1];
```

```
initial begin
```

```
....
```

```
$readmemh(“dmem.dat”,dmem);
```

```
$readmemb(“imem.dat”,imem);
```

```
0001_00000000  
0110_00000001  
1000_00000000  
0001_00000010  
0111_00000011  
1000_00000010  
...
```

imem.dat:12bit

```
0000  
0002  
0003  
0001  
...
```

dmem.dat:16bit