

2012 Fall CSE140L

Digital Systems Laboratory

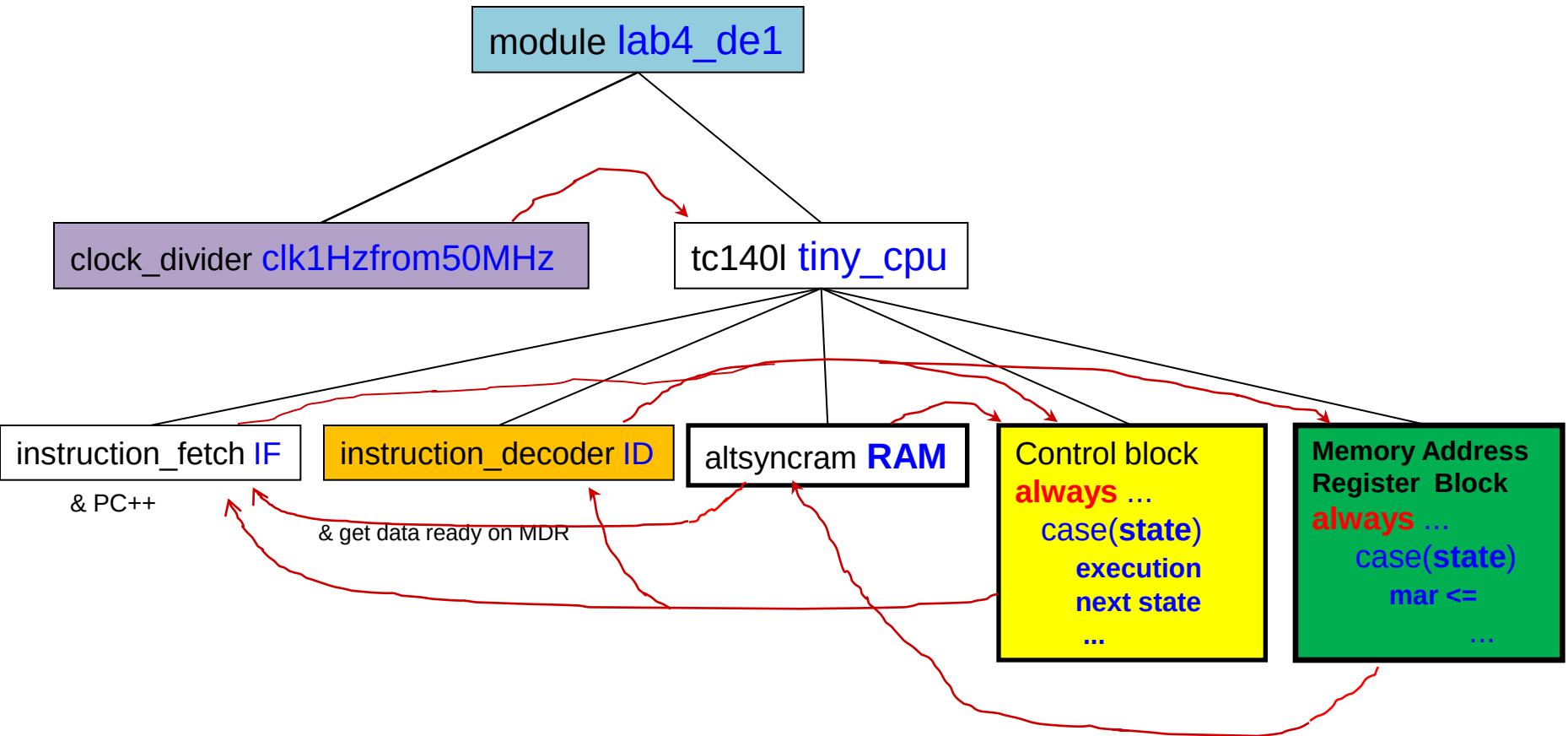
Lecture #9 & 10

by

Dr. Choon Kim

CSE Department, UCSD

Our TinyComputer Verilog Code Major Flow



Question: Which block(s) should be updated?

Practical Sequential Logic Design

[LAB4.pdf](#)

Memory Direct Addressing Mode("add.mif")

```

DEPTH = 256;           % Memory depth and width are required      %
WIDTH = 16;            % Enter a decimal number      %

ADDRESS_RADIX = HEX;   % Address and value radices are optional    %
DATA_RADIX = HEX;      % Enter BIN, DEC, HEX, or OCT; unless      %
                        % otherwise specified, radices = HEX      %

-- Specify values for addresses, which can be single address or range
-- program add
CONTENT
    BEGIN
[00..FF]                : 0000;    % Range--Every address from 00 to FF = 0000 (Default) %
                                % Warning: Comments may or may not be correct! You must confirm %
                                % each instruction with definition in Verilog source code. %

    00 :                  0211;    % LOAD AC with MEM(11) -- initialize AC %
    01 :                  0012;    % ADD MEM(12) to AC %
    02 :                  0C00;    % OUT AC %
    03 :                  0113;    % STORE AC to MEM(13) %
    04 :                  0300;    % JUMP to 00 (loop forever) -- reset AC %
    10 :                  0000;
    11 :                  5555;
    12 :                  1111;
    13 :                  0000;

    END ;

```

"5555 6666..."

The diagram illustrates the memory content at specific addresses. Blue arrows point from the addresses 00, 01, 11, and 12 to the value 5555, indicating that these memory locations contain the hexadecimal value 5555.

Random Number Generation Instruction("random.mif")

```
DEPTH = 256;           % Memory depth and width are required      %
WIDTH = 16;            % Enter a decimal number      %

ADDRESS_RADIX = HEX;   % Address and value radices are optional    %
DATA_RADIX = HEX;      % Enter BIN, DEC, HEX, or OCT; unless      %
                      % otherwise specified, radices = HEX        %

-- Specify values for addresses, which can be single address or range
-- program pseudo random sequencer
CONTENT
    BEGIN
[00..FF]                : 0000;    % Range--Every address from 00 to FF = 0000 (Default) %
                                % Warning: Comments may or may not be correct! You must confirm %
                                % each instruction with definition in Verilog source code. %
    00 : ← 1500;                % LOAD AC with Random number %
    01 : 0300;                  % JUMP to 00 (loop forever) %
    10 : 0000;
    11 : 5555;
    12 : 1111;
    13 : 0000;
    END ;
```

Memory Indirect Addressing Mode("addind.mif")

```

DEPTH = 256;           % Memory depth and width are required      %
WIDTH = 16;             % Enter a decimal number                  %

ADDRESS_RADIX = HEX;   % Address and value radices are optional  %
DATA_RADIX = HEX;      % Enter BIN, DEC, HEX, or OCT; unless     %
                        % otherwise specified, radices = HEX      %

-- Specify values for addresses, which can be single address or range
-- program addind
CONTENT
    BEGIN
[00..FF]                : 0000;    % Range--Every address from 00 to FF = 0000 (Default) %
                                % Warning: Comments may or may not be correct! You must confirm %
                                % each instruction with definition in Verilog source code. %

00:                      0210;      % LOAD AC with MEM(10) -- initialize AC %
01:                      1611;      % ADD IND memory contents to AC %
02:                      1613;      % ADD IND memory contents to AC %
03:                      1611;      % ADD IND memory contents to AC %
04:                      1613;      % ADD IND memory contents to AC %
05:                      1611;      % ADD IND memory contents to AC %
06:                      0300;      % JUMP to 00 (loop forever) %
10:                      0002;
11:                      0012;
12:                      0009;
13:                      0010;
14:                      0005;

    END ;

```

"2 b(=2+9) d(=b+2) 16(=d+9) 18(=0x16+2) 21(=0x18+9) 2"

Memory PC Reference Addressing Mode("addpcr.mif")

```
DEPTH = 256;           % Memory depth and width are required      %
WIDTH = 16;             % Enter a decimal number                  %
```

```
ADDRESS_RADIX = HEX;   % Address and value radices are optional   %
DATA_RADIX = HEX;       % Enter BIN, DEC, HEX, or OCT; unless     %
                        % otherwise specified, radices = HEX       %
```

```
-- Specify values for addresses, which can be single address or range
```

```
-- program addpcr
```

```
CONTENT
```

```

[00..FF] BEGIN
:          0000;           % Range--Every address from 00 to FF = 0000 (Default)      %
00:        0210;           % LOAD AC with Mem(10) -- initialize AC %
01:        1710;           % ADD PCR memory content to AC %
02:        1712;           % ADD PCR memory content to AC %
03:        1714;           % ADD PCR memory content to AC %
04:        0300;           % JUMP to 00 (loop forever) %
10:        0009;
11:        0001;
12:        0005;
13:        0003;
14:        0004;
15:        0007;
16:        0006;
17:        0007;
18:        000B;
END ;
```

"9 E(=9+5) 15(=E+7) 16(=0x15+B) 20(=0x16+B) 9"