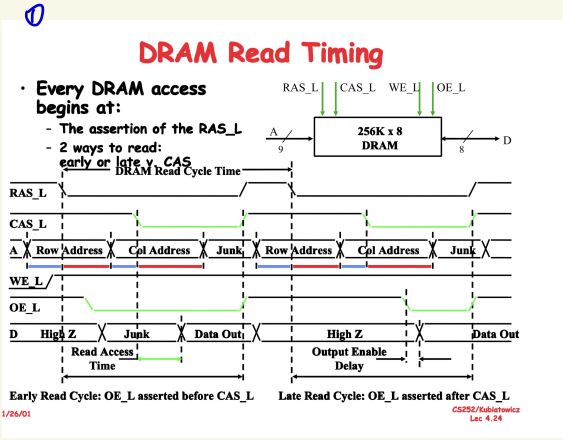
Week 9 Day 2 / 11月9日

①



High 2 高阻态

RAS: row access strobe.

CAS: column access strobe

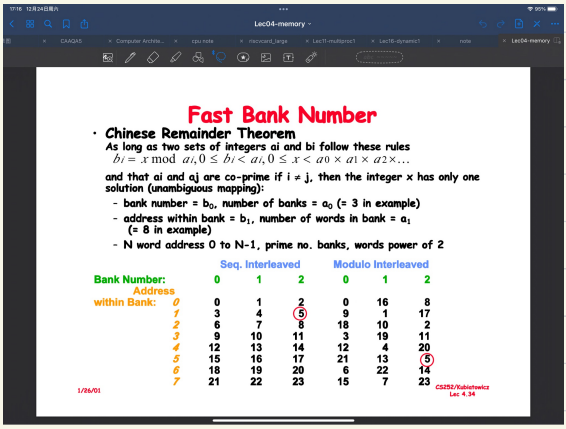
WE: write enable

OE: out enable

Main Memory Performance $AMAT_{mem} = T_{addr} + T_{access} + T_{transfer\ and\ send\ date}$

②

MutiBank



module $P = 2^l - 1$ 或 $P = 2^l + 1$

从address d 到 interger pair (u,v)

$0 \le d \le pm - 1$

$0 \le n \le m-1$ $0 \le v \le p-1$

PMS1 $v = d \bmod p$ $u = [\frac{d}{p}]$

缺点是除法运算复杂度高

PMS2. $v = d \bmod p$ $u = [\frac{d}{p-1}]$

其中 $p = 2^l + 1$ 可以用shift计算除法但很复杂

PDMS 中国剩余定理 $v = d \bmod p$.

并且 $a = d \bmod m$.

12月5日

· Week13 Day1

Running Example

- This code, a scalar to a vector:

```
for (i=1000; i>0; i=i-1)
    x[i] = x[i] + s;
```

- Assume following latency all examples

A 指令所需周期 B 下一条执行B的话要间隔多少时间

Instruction producing result	Instruction using result	Execution in cycles	Latency in cycles
FP ALU op	Another FP ALU op	4	3
FP ALU op	Store double	3	2
Load double	FP ALU op	1	1
Load double	Store double	1	0
Integer op	Integer op	1	0

延迟 需要 stall (暂停)

```
Loop:  L.D     F0,0(R1)  ;F0=vector element
       ADD.D   F4,F0,F2  ;add scalar from F2
       S.D     0(R1),F4  ;store result
       DSUBUI  R1,R1,8   ;decrement pointer 8B (DW)
       BNEZ    R1,Loop   ;branch R1!=zero
       NOP               ;delayed branch slot
```

调整指令顺序减少 stall

改变编译模式

Hardware scheduling

unrolling { 优点 并行，大大减少 overhead 时间
缺点 会要求大量 reg 代码膨胀

Unrolling 条件 每段代码没有相关性

Software pipeline

基本思路：把循环用 unrolling 展开，然后分析 pipeline

Super scalar (多发射) - 个部件一个时刻可以处理多个数据

VLIM (Very long Instruction Word)

· History
  - TI-DSP
  - unrolling
  - S/W pipeline

Kill Branch Instruction

- ① 用 LWC 条件执行指令
- ② superblock
- ③ trace cache