

5.2 MAB ต้องฉันทนความจำ 64 บิตเพื่อเก็บข้อมูล  
เหมือนกับ MDR

5.3 Sentinel ไม่ใช่ใช้ ตัวมันเองเป็น input  
เพราะเป็นการสิ้นสุดของลูป

5.4 a. 8 bits.  
b. 6 bits.  
c.  $10 - (3+1) = 6$



5.7 [15:12] บิตระบุข้อ IR  
[11:9] สัญรับ register ปลายทาง  
[8:6] สัญรับ register ตัวดำเนินการตัวแรก  
[5] ใช้สำหรับค่าทางอักษร (ตัวดำเนินการตัวที่ 2)  
[4:0] สัญรับค่าทางตัวอักษร  
ตัวมันใน 5 บิตที่เป็นค่า 2 complement  
จำนวนบวกที่สามารถแทนได้สูงสุดคือ 15

5.8 บิตใน register ไม่พอ register 32 บิต  
ต้องการอีก 4 บิต  
เราต้องการ 15 บิต แต่เรามีแค่  $16-4=12$  บิต  
สำหรับการกำหนดที่อยู่ของ register



5.10 บิตแรกและไบนารี การกำหนดที่อยู่ หรือไบนารี ตัวดำเนินการ  
ที่พัฒนาได้แตกต่างกัน ซึ่งอาจทำให้ คำสั่งมีความแตกต่างกันได้

A ตัวดำเนินการ เพิ่ม ค่าใน register  
B ตัวดำเนินการ โยนค่า ความจำจากที่อยู่ที่ตั้งค่าไว้ได้

5.13 a. (AND) 0001 (R3) 011 (R2) 010 (ค่าจริง) 1 (ค่าจริง) 00000  
b. - (AND) 0101 (R1) 001 (R1) 001 (ค่าจริง) 1 00000  
- (NOT) 1001 (R3) 011 (R3) 011 11111  
- (AND) 0001 (R3) 011 (R3) 011 (ค่าจริง) 1 00001  
- (AND) 0001 (R1) 001 (R2) 010 (ค่าจริง) 0 000000 011  
c. (AND) 0001 (R1) 001 (R1) 010 (ค่าจริง) 0 00000  
d. ทำไม่ได้ เพราะไม่มีทางที่ จำนวน n และ 0 ในเวลาเดียวกันได้  
e. (AND) 0101 (R2) 010 (R2) 010 (ค่าจริง) 00000

5.21 ประทศไปดอย 8 ปี ตั้งแต่วันที่ [7] [0] ดังนั้น LC-3 จึงรองรับได้สูงสุด 256 ปี

นายธนกร พัทธกิจไพศาล 6710301007



5.23  $\times 1482$  71216

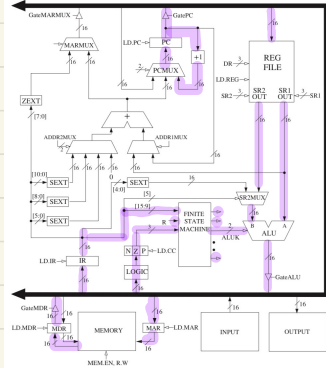
ORg ~~#2~~      " }  
 AND R0, R0, #0  
 DRneg ~~#2~~      " }  
 AND R0, R0, #0  
 ADD R0, R0, #-1  
  
 -      0      +  
 n=0, z=0, p=0  
  
 n=1, z=0, p=0

5.50 BK  $\bar{n}_v \gamma_2$  PC

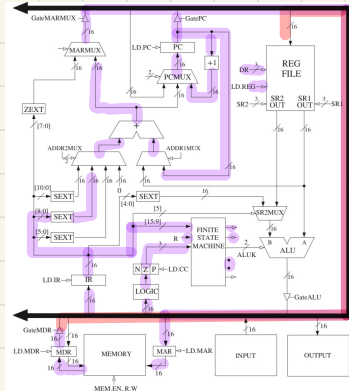
LEA 000000 Reg File

LD เก็บใน MAR

ALU ใช้ในการคำนวณ  $SR_1, SR_2, DR$   
 9 บิต สำหรับการเคลื่อนย้ายข้อมูลในการ ADD



วิธี ALU ในการคำนวณค่า offset  
วิธี: LDI ส่วนรับในดักขั้ว



5.37