5.15

R1: 0x3121

R2: 0x4566

R3: 0xABCD

R4: 0xABCD

5.16

a. PC-relative mode. The offset can be represented by 9 bits. So LD instruction can be used.

b. Indrect mode. The offset cannot be represented by 9 bits, it can be store in a register and then we can load from the address by LDI instruction.

c. Base+Offset mode. We can store the head address of the array in the base register and keep incresing the offset to load sequentially.

5.37

LDI: PC register, general register, memeory and ALU.

5.39

LEA: general register and ALU.

5.50

The Conditional Branch: PC register.

The Load Effective Address: general registers.

The LD instruction: the MAR register.

6.9

```
AND R0, R0, #0 0101 000 000 1 00000
AND R1, R1, #0
ADD R0, R0, #90
AND R1, R1, #100
LOOP TRAP x21
AND R1, R1, #-1
BRz END
BRznp LOOP
END TRAP x25
```

the machine language:

```
0101 000 000 1 00000
0101 001 001 1 00000

```

```
; R0 <- R0 + 15 for 6 times
0001 000 000 1 01111
0001 000 000 1 01111
0001 000 000 1 01111
0001 000 000 1 01111
0001 000 000 1 01111
0001 000 000 1 01111

; R1 <- R1 + 100
0001 001 001 1 01111
0001 001 001 1 01111
0001 001 001 1 01111
0001 001 001 1 01111
0001 001 001 1 01111
0001 001 001 1 01111
0001 001 001 1 01010

1111 0000 0010 0001 ; TRAP x21
0001 001 001 1 11111 ; R1 <- R1 - 1

0000 010 000000001 ; BRz 1
0000 111 111111100 ; BRnzp -4

1111 0000 0010 0101 ; TRAP x25
```

6.10

If R2 is odd, R1 will be 1, else R1 is 0.

```
AND R1, R1, #0
AND R0, R2, #1
BRz END
ADD R1, R1, #1
END TRAP x25
```

the machine code:

```
0101 001 001 1 00000
0101 010 010 1 00001
0000 010 000000001
0001 001 001 1 00001
1111 0000 0010 0101
```