One way you could do this is to move the constant to register A. This can be done via (irmovl V, rA). After this, you need to add A and B. This can be done via (addl rA, rB). However, the “meat” of the problem is below:

In strict computation form as seen in 4.18 and asked in the question, iaddl V, rB would be accomplished as such:

Stage:
iaddl V, rB

Fetch:
//(M1, M1, and M4 have their integers underscored)
icode:ifun ← M1[PC]
rA:rB ← M1[PC+1]
valC ← M4[PC+2] //set up normal/standard fetch setup
valP ← PC+6

Decode:
valB ← R[rB] //get original rB value (from rB) by decoding

Execute:
valE ← valB + valC //add in rB val (valB) to the execution

Memory:
N/A

Write Back:
R[rB] ← valE //write back the new value we got from the addition of execute

PC Update:
PC ← valP //update the new value on computer