1.

Instruction	Fetch	Decode	Execute	Memory	Write Back
Addq %rax, %rbx	11, 12, 14, 15, 16, 17, 19	5, 21, 22, 26, 27	31, 32, 33, 34, 35, 36, 37, 41	---	38, 22
Rrmovq %rax, %rbx	11, 12, 14, 15, 16, 17, 19	5, 21, 22, 26, 27	31, 32, 33, 34, 36, 37	39, 46	38, 22
Irmovq 5, %rbx	11, 12, 14, 15, 16, 17, 18	4	33, 37	--	38, 22
Mrmovq 5(%rdx), %rax	11, 12, 14, 15, 16, 17, 19	21, 5, 22, 27	21, 32, 34, 36, 37	39, 40, 46	3
Rmmovq %rax, 5(%rdx)	11, 12, 14, 15, 16, 17, 19	21, 22, 27	31, 32, 33, 34, 36, 37	39, 46	38, 22
Push %rax	11, 12, 14, 15, 16, 17, 19	5, 21, 24, 26, 27	31, 33, 32, 37, 38	---	38, 3
Pop %rax	11, 12, 14, 15, 16, 17, 19	5, 21, 24, 26, 27	31, 33, 32, 37, 38	---	38, 3
Jne loop	11, 12, 14, 15, 16, 17, 19	5, 21, 24, 26, 27	31, 33, 32, 37, 38	39, 40, 46	38
Call func	11, 12, 14, 15, 19	24, 26, 27	31, 32, 33, 34, 37	39, 40, 46	38
ret	11, 12, 14, 15, 16, 17	5, 21, 22, 26, 27	31, 32, 33, 34, 36, 37	---	38, 22
Cmovne %rax, %rbx	11, 12, 14, 15, 16, 17, 19	5, 21, 22, 26, 27	31, 32, 33, 34, 36, 37	---	38, 22

2.

Instruction	Icode	Ifun	rA	rB	Val C	Val A	Val B	Dst E	Dst M	Src A	Val E	Val M	Cnd
Addq %rax, %rbx	6	0	0	3	-	1	2	3	-	0	1	-	0
Rrmovq %rax, %rbx	2	0	0	3	-	1	-	3	-	0	2	-	-

Irmovq 5, %rbx	3	0	F	3	5	-	-	3	-	-	5	-	-
Mrmovq 5(%rbx), %rax	5	0	0	3	5	-	2	0	-	-	7	0	-
Rmmovq %rax, 5(%rdx)	4	0	0	2	5	1	0x1	-	-	-	0x1005	0	-
Push %rax	A	0	0	F		1	-x2	4		0	0x1%%8	1	-
Pop %rax	B	0	0	F		0x2	0x2	4	2	4	0x2008	2	-
Jmp loop	7	0	-	-	0x3	-	-	-	-	-	-	-	0
Call func	8	0	-	-	0x4	-	0x2	4	-	-	0x1FF8	PC +9	-
Ret	9	0	0	0x2	-	0x2	0x2	4	-	4	0x2008	2	3
Cmove %rax, %rbx	7	3	0	3	-	1	0	3	-	-	1	-	3