Hien Duong
Cpts 460 Lab3

2.

In (6) at unlock() ====> Reason: allow CPU to accept IRQ interrupts

In (6) at kgetc() =====> (5) Reason: CPU executes this function

In (5) at while(hasData==0); ====> (1) Reason: Interrupt has occurred

In (1) at 0x18: LDR PC, irq_handler_addr irq_handler_addr: .word irq_handler ===> (2)
Reason: CPU follows the IRQ vector at 0x18 to enter the irq_handler.

In (2) at bl IRQ_handler ====> (3) Reason : After pushing the registers r0-r12 and lr into the
IRQ stack it will call the IRQ handler in C.

In (3) at if (VIC.statusBit31 && SIC.statusBit3) {kbd_handler(); } =====> (4) Reason: After
reading the status registers of PIC and SIC it calls the kbd_handler()

In (4) get scancode; c = ASCII char mapped by scancode; hasData = 1; =====> (3) ====> (2)
Reason: After getting the scan code and the ASCII character mapped by the scan code, it will
return to where it was originally called in irq_handler.

In (2) at ldmfd sp!, {r0-r12, pc}^ =====> (5) Reason: After popping the stack it will return to the
original point of interruption.

In (5) at hasData = 0; return c; ====> (6) Reason: After the return from the interrupt it now has
the character stored in c from the handler so now it will return to main with that character.