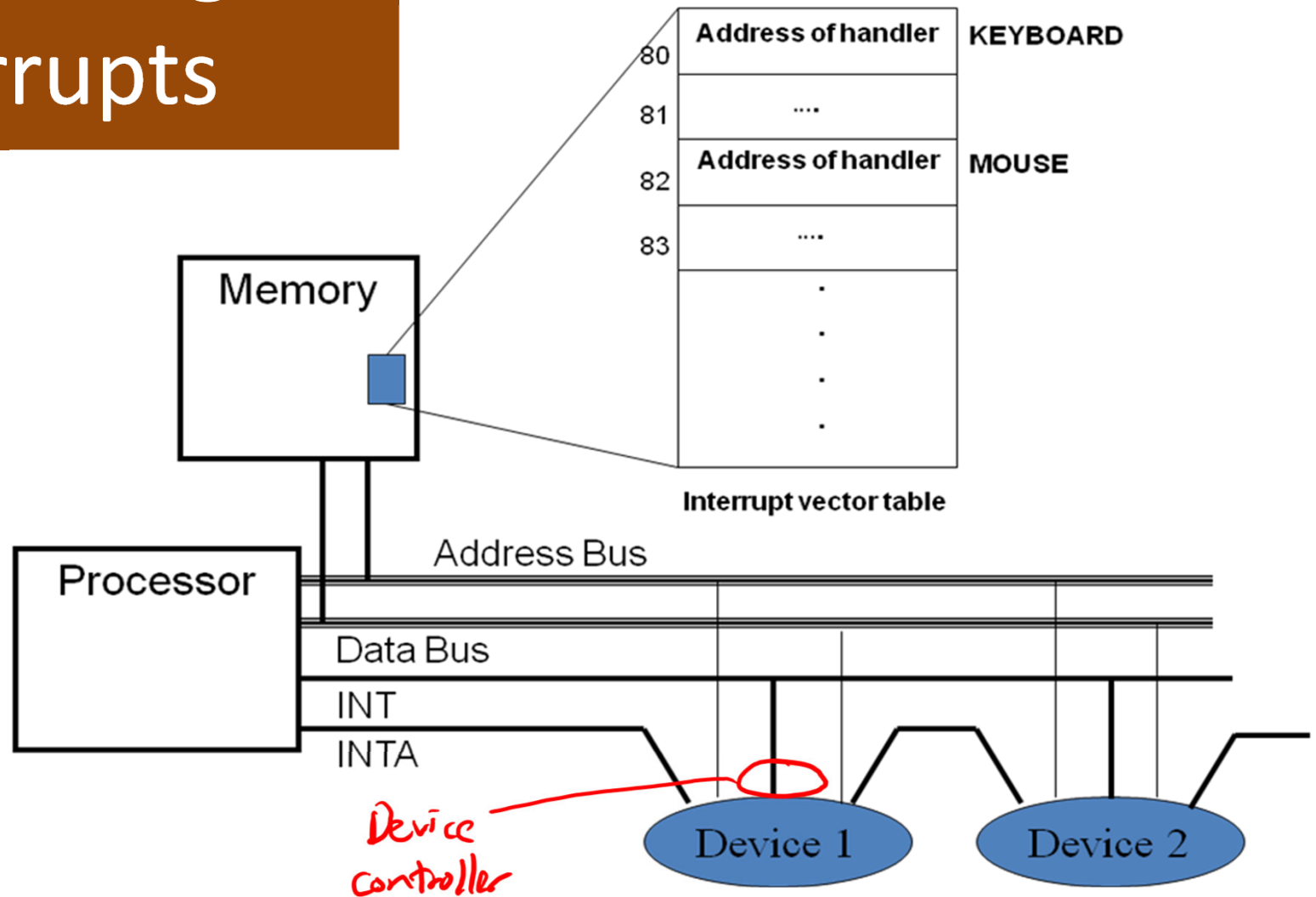
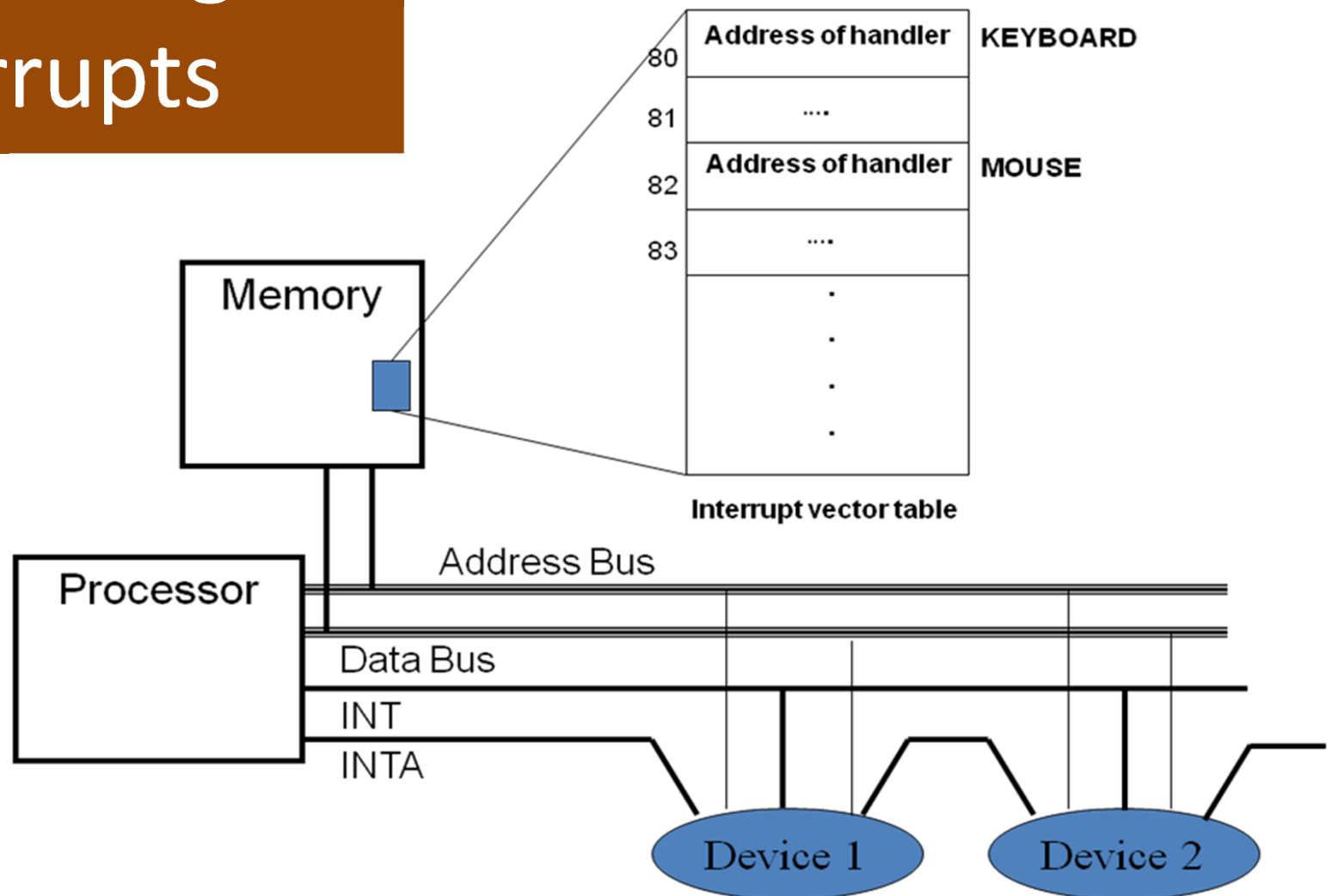


Processing Interrupts



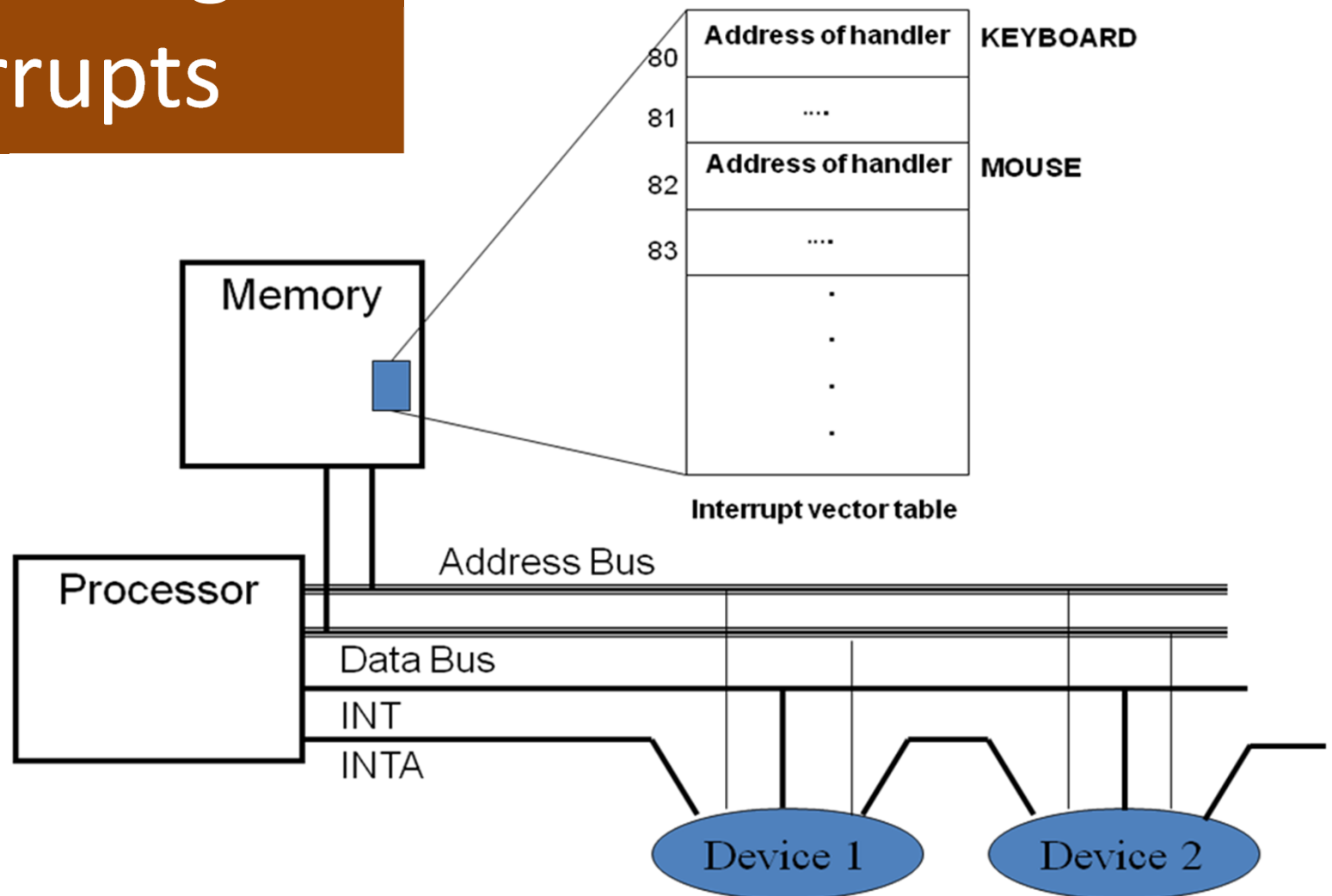
1. The device controller asserts an interrupt line.

Processing Interrupts



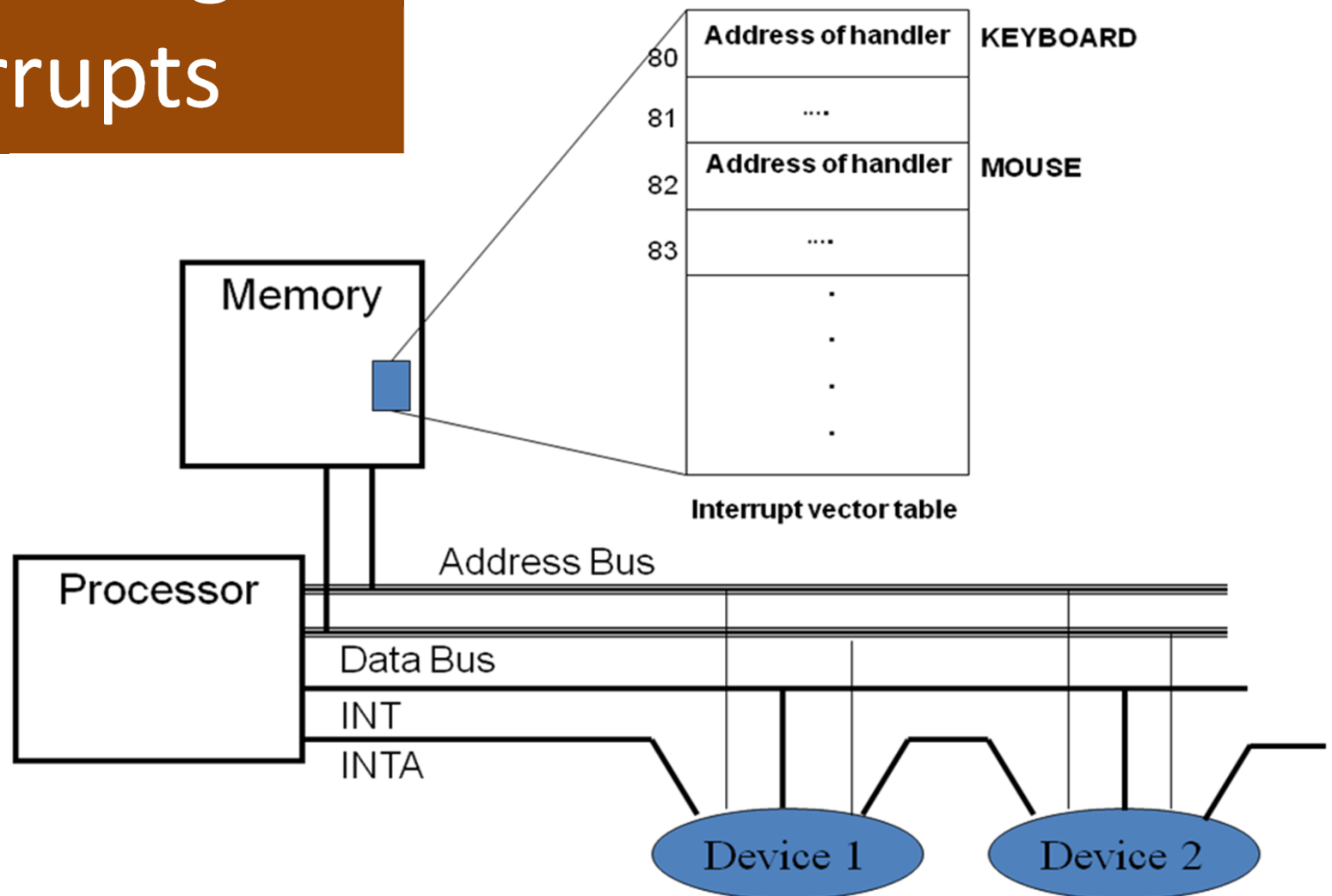
2. Once the CPU is ready to handle the interrupt, it will acknowledge the interrupt on the bus.

Processing Interrupts



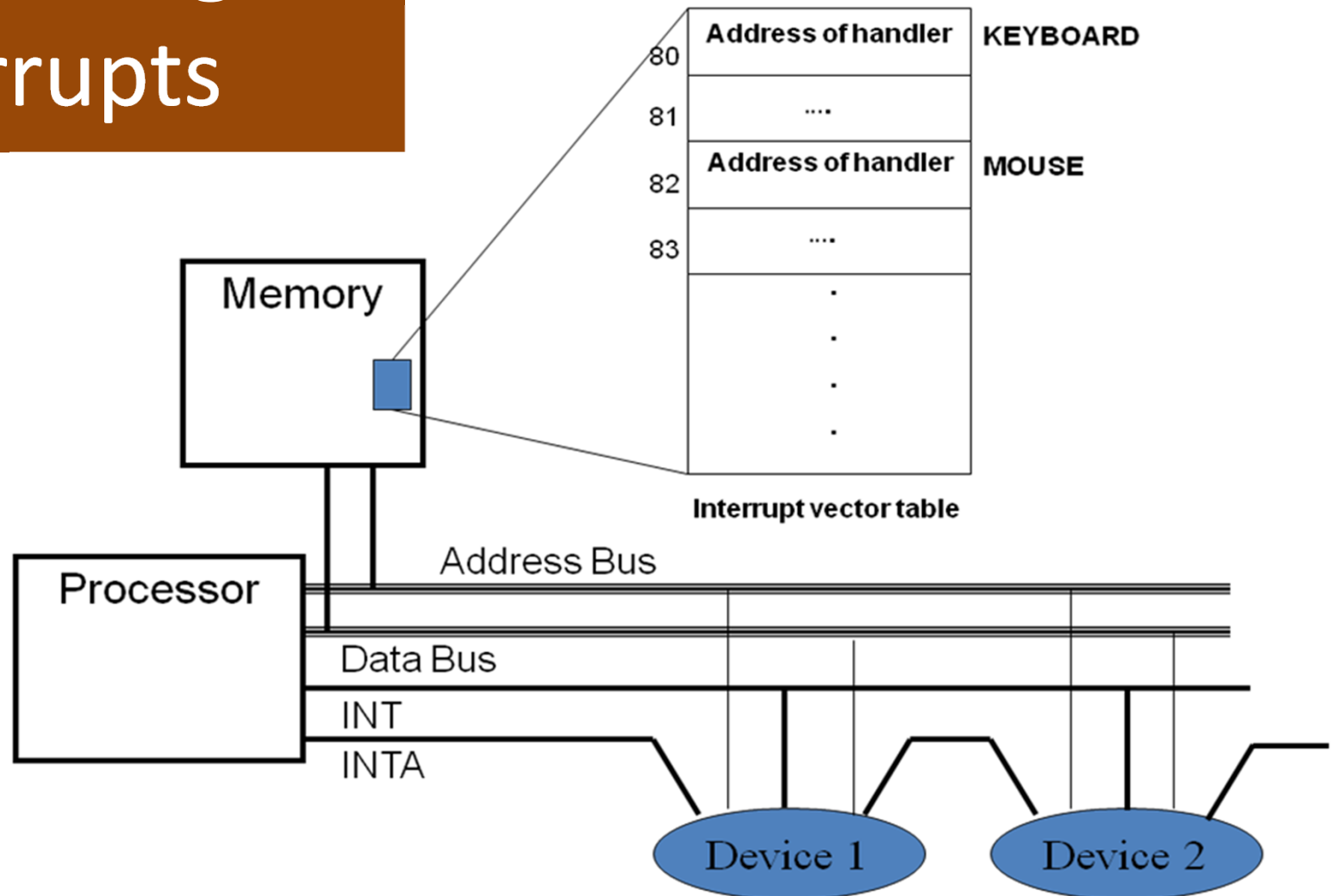
3. Upon seeing the acknowledgment, the device controller will put a small integer on the system bus data lines. This integer, called the *interrupt vector*, corresponds to the type of interrupt.

Processing Interrupts



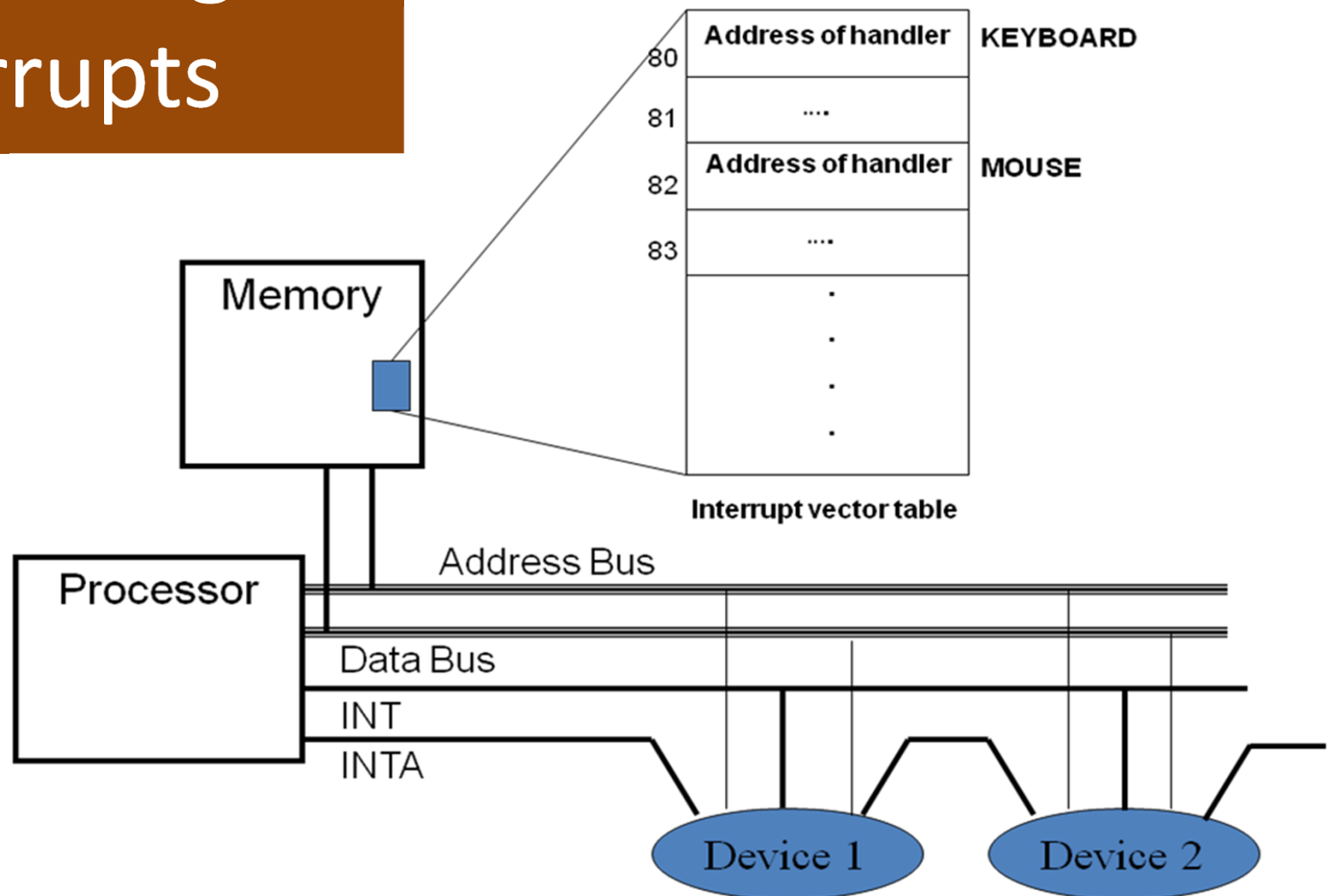
4. CPU temporarily saves the interrupt vector from the bus.
5. CPU will save the current values of PC and other inaccessible system registers (such as the status word) on the stack.

Processing Interrupts



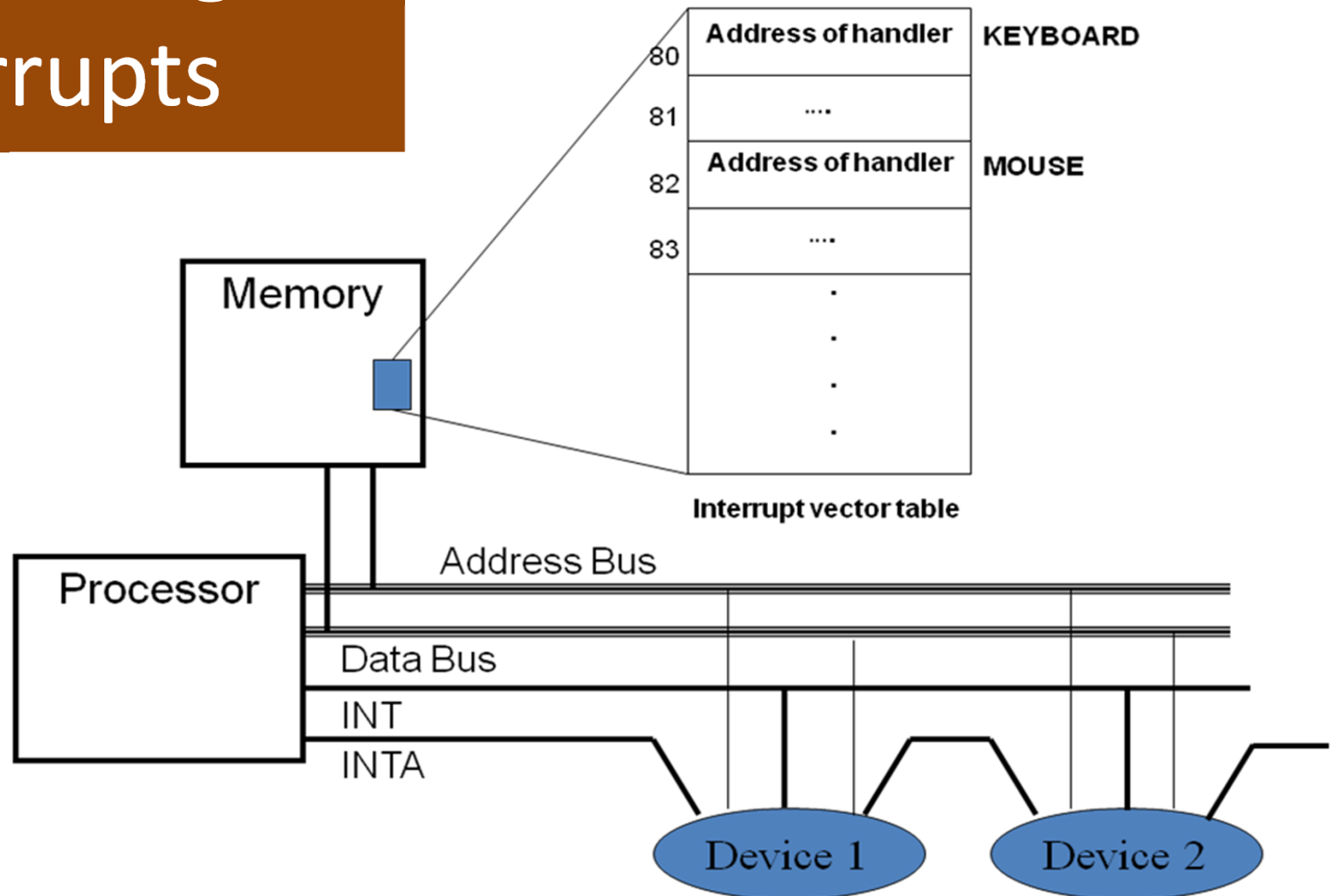
6. CPU uses the interrupt vector as an index the interrupt vector table stored in memory. The interrupt vector table stores the starting addresses of interrupt service routines.

Processing Interrupts



7. Jump to the interrupt service routine. Different interrupt vectors (corresponding to different interrupt types) will jump to different interrupt service routines.

Processing Interrupts



8. Once the interrupt service routine is completed, it must execute a special "return from interrupt" instruction. This instruction restores the saved values of PC and system registers (saved in step 5) and resumes execution of the program where it left off.