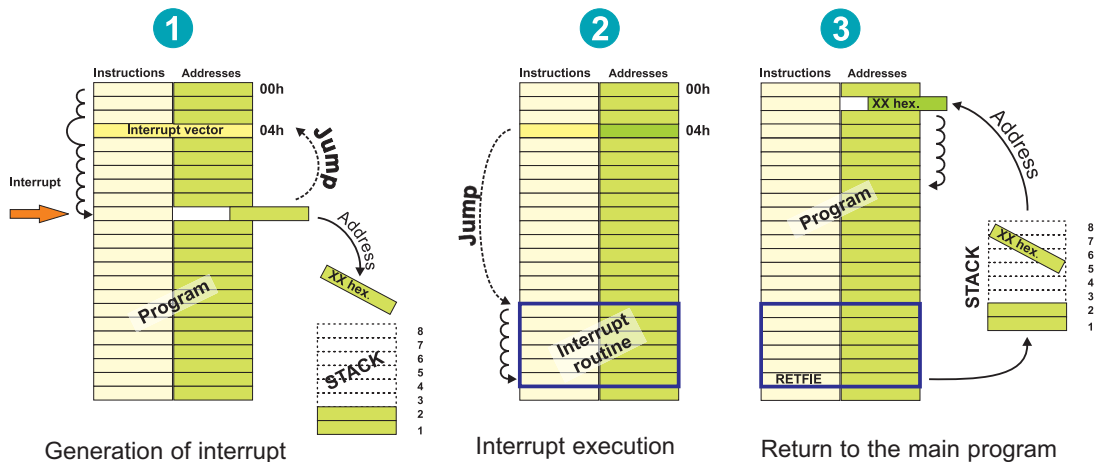


INTERRUPT SYSTEM

The first thing to be done by the microcontroller, when an interrupt request arrives, is to execute the current instruction, then to stop the regular program execution. The current program memory address is automatically pushed onto the stack and the default address (predefined by the manufacturer) is written to the program counter. The location from where the program proceeds with execution is called an interrupt vector. For the PIC16F887 microcontroller, the address is 0004h. As seen in figure below, the interrupt vector should be skipped during regular program execution.

A part of the program to be executed when an interrupt request arrives is called an interrupt routine (it is a subroutine in fact). The first instruction of the interrupt routine is located at the interrupt vector. How long will it take to execute the subroutine and what it will be like, depends on the skills of the programmer as well as on the interrupt source itself. Some microcontrollers have a couple of interrupt vectors (every interrupt request has its vector), whereas this microcontroller has only one. This is why the first part of every interrupt routine should be interrupt source detection. When the interrupt source is known and interrupt routine is executed, the microcontroller reaches the `RETFIE` instruction, pops the address from the stack and proceeds with program execution from where it left off.



MikroBasic recognizes an interrupt routine to be executed by means of the *interrupt* keyword. The interrupt routine should be written by the user.

Example

```
sub procedure interrupt      ' Interrupt routine
cnt = cnt + 1 ;             ' Interrupt causes variable cnt to be incremented by 1
end sub
```