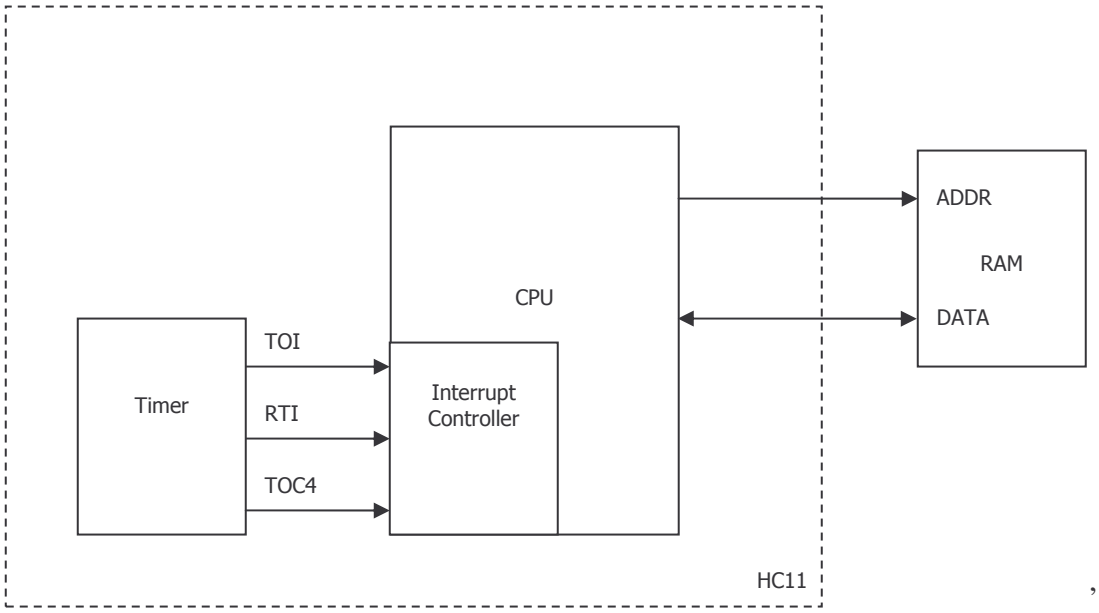
HC11 Interrupt processing

1. Device asserts interrupt signal.
2. CPU finishes current instruction.
3. CPU stacks machine state.
4. Interrupt source discovery
5. PC = mem_{vector_table}[vector]
6. ISR runs. Last instruction is RTI
7. Machine state unstacked. PC restored.

Not all microprocessors and microcontrollers automatically stack the machine state and go ahead and execute the ISR



If your processor does not autostack, you must push and pull the register's values at beginning and end of the ISR. It is good "do no damage" practice to stack the entire state of the machine, as opposed to only those registers which you will use in the ISR.

When an interrupt occurs, all other interrupts are disabled. Don't put a CLI in the ISR.

The Timer Interrupt Mask Register (TMSK2) must be set within the first 64 clock cycles of the RESET signal to the processor.

Basic Interrupt Programming

Two steps to enabling interrupts:

1. Unmask interrupt you would like to use
2. Tell CPU to stop ignoring interrupts

Write a 1 to clear flag strategy.