- **More in Interrupts**
- Huang Sections 6.1-6.4
  - Using interrupts on the 9S12
  - The 9S12 registers and stack when a TOF interrupt is received
  - The 9S12 registers and stack just after a TOF interrupt is received
  - Interrupt vectors for the MC9S12DP256

EXCEPTIONS ON THE HCS12

• Exceptions are the way a processor responds to things other than the normal sequence of instructions in memory.

• Exceptions consist of such things as Reset and Interrupts.

• Interrupts allow a processor to respond to an event without constantly polling to see whether the event has occurred.

• On the HCS12 some interrupts cannot be masked — these are the Unimplemented Instruction Trap and the Software Interrupt (SWI instruction).

• **XIRQ** interrupt is masked with the **X** bit of the **Condition Code Register**.
Once the X bit is cleared to enable the XIRQ interrupt, it cannot be set to disable it.
  – The XIRQ interrupt is for external events such as power fail which must be responded to.

• The rest of the **HCS12 interrupts** are masked with the **I** bit of the **CCR**.
– All these other interrupts are also masked with a specific interrupt mask.
– This allows you to enable any of these other interrupts you want.
– The I bit can be set to 1 to disable all of these interrupts if needed.

What happens when the HCS12 receives an unmasked interrupt?

1. Finish current instruction

2. Push all registers onto the stack

3. Set I bit of CCR

4. Load Program Counter from interrupt vector for particular interrupt

Most interrupts have both a specific mask and a general mask. For most interrupts the general mask is the I bit of the CCR. For the TOF interrupt the specific mask is the TOI bit of the TSCR2 register.

Before using interrupts, make sure to:

1. Load stack pointer
- Done for you in C by the C startup code

2. Write Interrupt Service Routine
- Do whatever needs to be done to service interrupt
- Clear interrupt flag
- Exit with RTI
  - Use the INTERRUPT definition in the Gnu C compiler

3. Load address of interrupt service routine into interrupt vector

4. Do any setup needed for interrupt

- For example, for the TOF interrupt, turn on timer and set prescaler

5. Enable specific interrupt.

6. Enable interrupts in general (clear I bit of CCR with cli instruction or enable() function

Can disable all (maskable) interrupts with the sei instruction or disable() function.