Name: _____ Student #: _____ Signature: _____

1. (2 mark) What is an interrupt vector table and does the NIOS II processor have one?

In many processors, different interrupt service routines can be invoked directly by interrupt number. The address of each ISR is loaded into a long table so that when interrupt-N occurs, the appropriate routine can be called directly by jumping to the address found in the Nth entry in the interrupt vector table. The NIOS II processor does not have an interrupt vector table and there is only one destination for all interrupts.

2.   (5 marks) List (in sequence) the 5 events that occur when an interrupt is generated and acknowledged on the NIOS II processor:

- Stop the current instruction immediately (the instruction is left unfinished)

- Save the Instruction Pointer (IP) to the processor exception return address register (ea = r29)

- Save the current processor status (including the PIE bit) to the processor estatus register

- Set PIE=0 in the processor status register to disable interrupts while this interrupt is being serviced

- Branch to address 0x00000020

3. (1 mark) What is the name of the subroutine that is executed when an interrupt occurs?

Interrupt Service Routine (ISR)

4. (1 mark) The **eret** instruction is used to return from the interrupt "subroutine". What must first be done to ensure that the NIOS II processor returns to the proper instruction?

Must first subtract four from the ea register to complete the instruction that was left unfinished

5.   (1 mark) What does the term **interrupt latency** refer to?

The term interrupt latency refers to how long it takes a device to be serviced after an interrupt is generated.

Name: _____ Student #: _____ Signature: _____

6. (4 marks) In each of the separate C-language code segments below, give the final value of a.

```c`
unsigned int a=1, b=0;
if(!b)
{
   a++;
   b++;
}
``
	a = 2

```c`
unsigned int a=2, b=1;
while(b) b = 3;
a = 0;
``
	a = 2

```c`
unsigned int a;
a = 5;
a = a & 0x0010;
``
	a = 0x0000 = 0

```c`
unsigned int a=0, b=15;
while(b){b = b>>1; a++;}
``
	a = 4