3. Interrupt Controller Question: In the table below, identify the registers that need to be initialized, and give values for each. In this interrupt system, there are four interrupt sources, starting in the least significant bit position, and all are to be enabled. Also, the software activation of interrupts is not to be utilized. Assume that you want to assert the appropriate bits to reset any flags that may have remained from an earlier program.

don't write "1's"

read only

Addr Offset	Register	Bit Pattern
0x00	ISR	0000→0
0x04	IPR	00000→0
0x08	IER	0000→0 ?
0x0C	IAR	1111
0x1C	MER	0x03

not int controlled

int

Now, in the space provided below, give instructions that will establish the bit patterns given above as well as to a)set up the vector register (to **0x000A0000**) and b)set up any enabling activity needed to allow interrupts in general. Assume that the interrupt controller has been located at address **0x84440000.**

```
.set ISR, 0x00
.set IPR, 0x04
.set IER, 0x08
.set IAR, 0x0C
.set MER, 0x1C
lis r2, 0x000A
lo r3, 0x8444
mtevpr r2
li r4, 0
li r5, 0xf
stw r4, IER(r3)
stw r5, IAR(r3)
li r6, 0x03
stw r6, MER(r3)

writee
```

Ne

It depends because you read "1's" for read only and "0's" for read/write