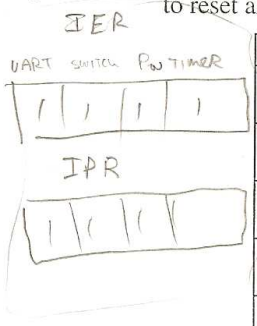
3. Interrupt Controller Question: In the table below, identify the registers that need to be initialized, and give values for each. In this interrupt system, there are four interrupt sources, starting in the least significant bit position, and all are to be enabled. Also, the software activation of interrupts 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.



Addr Offset	Register	Bit Pattern
0x00	ISR	0000
0x04	IPR	xxxx 1111 ← bad, read only!
0x08	IER	xxxx 1111
0x0C	IAR	0000 need this to be 1111 to clear flags
0x1C	MER	~~1100~~ xxxx 11

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 and b)set up any enabling activity needed to allow interrupts in general. Assume that the interrupt controller has been located at address 0x82340000.

Plus the offset

NOTE: EVERY TIME I ATTEMPT TO WRITE CODE ON THE SPOT IT IS WRONG. THIS IS BECAUSE I HAVE LEARNED EVERYTHING I KNOW ABOUT CODING IN THE LAB VIA TRIAL AND ERROR SO I WILL FOCUS ON CONCEPTUAL EXPLANATIONS

The IPR is going to indicate actively interrupts are taking place by having a bit pattern where a 1 represents an interrupt condition. The IER will indicate which interrupts I will be paying attention to. For example if the IER BIT PATTERN is 1011 and the user hits the switch to cause an interrupt, it's not going to happen because I never enabled the bits to enable that interrupt.

IM GOING TO SET A REG TO HOLD VALUE FOR THE ISR WHICH IS DETERMINED BY THE BIT PATTERNS ABOVE. FOR EACH CASE.