Lesson 4 Exercise: Make Blinky

Assignment:

On your final project board, make blinky for yourself. Then add a button to turn the LED on and off. Bonus points for making the button cause an interrupt. Triple bonus points for debouncing the button signal.

Question 1:

What are the hardware registers that cause the LED to turn on and off?
(From the processor manual, don't worry about initialization.)

LED Port, PIN: 0x48000000,0x00000100

Question 2:

What are the button registers that you read?

Button Port, Pin: 0x48000000,0x00000200

Question 3:

Can you read that memory directly and see the button change in a debugger or by printing? out the associated memory?

I believe this is possible, I'm not able to read that memory address with gdb, this might be user error.

Turn in your code with a comment or additional file answering the questions.

Exercise 4: Rubric for Peer Grading Diagrams

Review code from your peers.

Grade them according to the following rubric. For an assignment that truly exceeds expectations, give it the maximum score. A "Meets Expectations" score should be in the 50-70% range and a "Needs Improvement" would get 0-20%. Note that the scale is flexible and an assignment may be between levels.

When giving feedback, remember that you are talking to a person who worked on the assignment, not a robot who needs correcting. The goal is to help them understand how it would work better for you.

Criteria	Needs Improvement	Meets Expectations	Exceeds Expectations	Maximum Score
Turned in	Turned in late or nothing at all.	Turned in on time, mostly complete.	Turned in on time and completed.	10
Blinks	This code does not turn an LED off and on via a button.	This code does turn an LED off and on via a button.	This code does turn an LED off and on via a button. It is very easy to follow.	30
Registers	Questions about registers incompletely or incorrectly answered.	Register questions answered but left at the level of macros and HAL structures.	Clear description of which values to read and write to which addresses and why.	30
Clarity	This code and description does not make sense to me.	Code and description make sense after some effort to understand it or get an explanation.	Code and description are self-explanatory and an example of great coding.	20
Reusing code	This code does not re-use any code, including a HAL. Or this code is entirely an example with no modification.	This code is based on an example but effort has been put into making the code better. HAL is used or there is an explanation for why not.	Perfect balance between using available code to get the job done quickly and putting in effort to make the code better.	10
Bonus: Interrupt		Uses an interrupt when button is pressed to turn on/off the LED	Interrupt sends an event to the main loop to handle turning on and off the LED	5
Bonus: Debounce		Uses a timer to debounce, causing a button press event.	Uses a timer to debounce both high and low causing a button press and a	15

			button release event.	

Discussion Points:

- Why does the author use “marketing comes to you” through this chapter? How does it relate to previous chapters?
- Would you rather use a HAL or not? Why? What are the advantages either way?
- Peer review button blinkies
- Given an input interrupt, output(s), and timers, what could you build? How many things are just a combination of these?