Alfredo Romero

SNHU

CS 350 emerging system

6/8/2024

5-1 Milestone Three Submission

1. What is the purpose of the `timerCallback()` function?

The `timerCallback()` function is like an alarm clock that goes off at regular intervals. Each time it rings, it updates the LEDs to the next part of our Morse code message. So, if we're blinking out "SOS," this function moves us from one dot or dash to the next, making sure our message is displayed correctly and on time.
Lines of code: 80 – 122 and as parameter 138.

2. What does period mean in this context?

In this context, the period is the amount of time between each "alarm" from our timer. It's like setting a timer to ring every 0.5 seconds (500,000 microseconds). This period determines how quickly we move from one part of our Morse code message to the next. By setting the period, we control the pace at which our LEDs blink.

3. How does the `Timer_CONTINUOUS_CALLBACK` parameter impact the driver?

The `Timer_CONTINUOUS_CALLBACK` parameter tells the timer to keep going off over and over again, like a repeating alarm. This means that the timer will continuously call our `timerCallback()` function at the interval we've set. This keeps our Morse code message blinking steadily and reliably without stopping.

Lines of code: 137

4. What is `gpioButtonFxn0()` used for?

The `gpioButtonFxn0()` function is like a doorbell that rings whenever we press a specific button on our board. When this happens, the function sets a flag to let our program know the button was pressed. In our project, pressing the button switches the Morse code message from "SOS" to "OK" or vice versa, but only after the current message finishes blinking.

Lines of code: 157-161 and as parameter 191.

5. What is the purpose of `GPIO_CFG_IN_INT_FALLING`?

`GPIO_CFG_IN_INT_FALLING` is a setting that tells our program to watch for a button press. Specifically, it looks for the moment when the button is pressed down, which creates a signal drop. When this drop (or "falling edge") happens, it triggers an interrupt, which is like an urgent message to our program to run the `gpioButtonFxn0()` function and respond to the button press.

Lines of code: 184 and line 203