UNIVERSITY OF WATERLOO

ECE750 Real-time Embedded System

Lab #4

Name: Shaocong Ren

Student number: 20478300

E-mail: s7ren@uwaterloo.ca

Q1. Briefly explain your program implementation in the following areas:

1. **Which peripherals did you use in your program?**

   GPIO Port A Pin1, GPIO Port B Pin0, a hardware timer TIM2 and an external interrupt EXTI0 are used in my program.

2. **How do you keep track of overall game time?**

   TIM2 is used for four intervals in one game and during these four intervals 2s, 2.5s, 2.5s and 2.5s are accurately recorded. Hence, 9.5s from beginning is kept track of and at this time I pass the potato to guarantee to beat the AI.

   Specifically, I initialize the TIM2 to time 2s and in the first TIM2's ISR to modify prescaler to time 2.5s in the next time. When the game starts again I will initialize TIM2 again to time 2s.

3. **How do you track the amount of time you've held onto the potato?**

   In my strategy, student will hold the potato four times in total. The first time student holds it 2s (actually 1.999s to guarantee no more than 2s). For the remaining three times just make sure the time AI holding adding the time student holding equals 2.5s, no matter how long the AI holding student just passing it at the end of the interval 2.5s which can guarantee that student will not lose. And for the remaining 0.5s the potato is in AI's hand, since it cannot pass it within 0.5s it will lose.

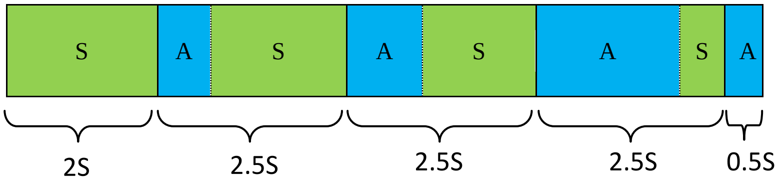
   Technically, the four times is realized by hardware timer TIM2 and external interrupt EXTI0. In the ISR of EXTI0 we can count the times of potato coming, if the count is on a multiple of 4 and we judge it is the beginning of game and TIM2 is initialized as 2s, else TIM2 is initialized as 2.5s.

4. **How do you detect the potato being passed to you?**

   This can be realized by external interrupt EXTI0 which initialized to detect the rising edge of GPIO Port B Pin0. As long as the interrupt is triggered, it means the potato is being passed to me.

Q2. Describe the algorithm used by your program to win against the AI. What strategy did you use to determine how long to hold onto the potato?

In my strategy, student will hold the potato four times in total. The first time student holds it 2s (actually 1.999s to guarantee no more than 2s). For the remaining three times just make sure the time AI holding adding the time student holding equals 2.5s, no matter how long the AI holding student just passing it at the end of the interval 2.5s which can guarantee that student will not lose. And for the remaining 0.5s the potato is in AI's hand, since it cannot pass it within 0.5s it will lose.



This figure shows the holding time by student (S) and AI (A) respectively. In the middle three intervals, do not care about how long the AI holding on to the potato, just pass it in the end of 2.5s.

Q3. Can your program guarantee victory against the random AI (i.e. will it win for all sets of valid random values generated by the AI) under the current game rules? Why or why not?

Theoretically, I can always beat the AI if we both respect the rules under my strategy, because I keep 9.5s not losing and give AI no more than 0.5s to hold on it.

However, in practice, there exists a little possibility (about 0.01%) that I will lose since executing the other clauses in the program will take tiny time which leads to increase the error of hardware timer. Besides, I make an interesting experiment that I hold onto the potato a little longer than 2s and the system shows I hold it 2001ms but it does not shows my failure, so the accuracy of judging system will affect the result of game.