Experiment 5: Using Interrupts on Arduino

Due date : 14.12.2020 12:00

Abdullah Ekrem Okur

okurabd@itu.edu.tr

1 Introduction

In this experiment, you will generate a counter and various number games using external interrupts. Circuit designs of each part will be given in sections below.

- **You can not use any loop statement and recursive functions.**

- **You can not use block programming provided by tinkercad.**

- **You should add comments for necessary cases. Otherwise, you will get zero point.**

- **You can not use digitalWrite and pinMode functions.**

- **You can not use timer interrupt and delay function. Use millis or micros instead of delay.**

For your questions: Abdullah Ekrem Okur (okurabd@itu.edu.tr)

2 Part-1

In this part, you will program the circuit whose design is presented in Figure 1 (Design link). You will implement a simple two-digit decimal timer using two seven-segment displays. The first seven-segment display will show the tens digit and the second seven-segment display will show the ones digit. The counter varies between 0 and 99. The initial value of the counter is 0. Your program should reset the timer as soon as the first button is clicked. Also, your program should have two modes which are counting up and counting down. The timer should count up by default and change the mode whenever the second push-button is clicked. Example sequences of the counter are shown in the table below. Your timer should run in real time, in seconds.

Note that, we expect you to implement interrupt subroutine for button press.

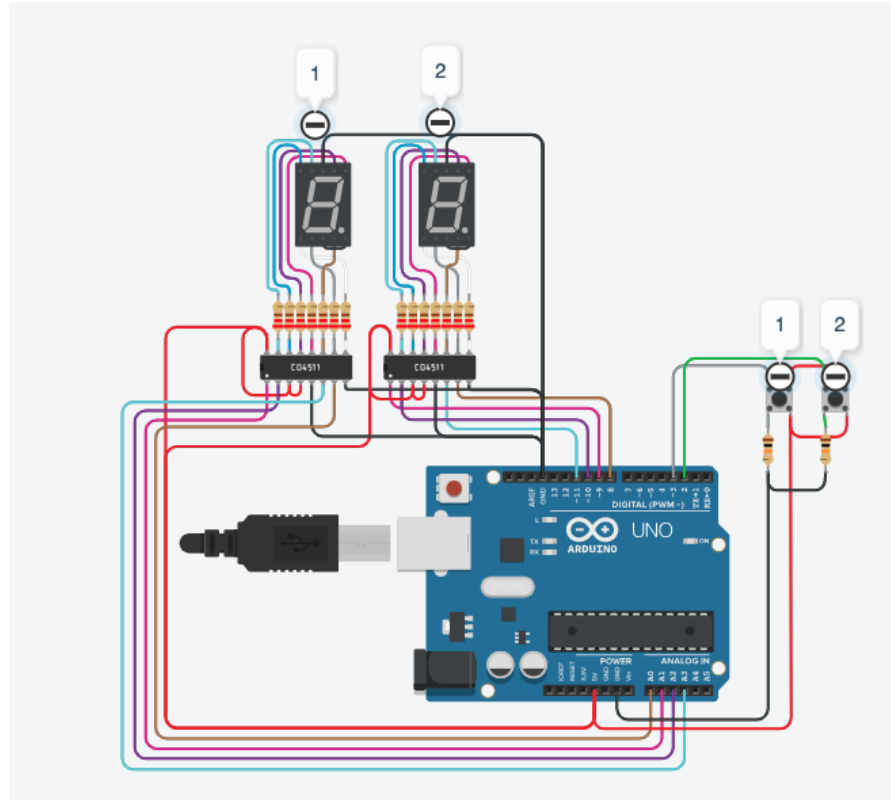| Mode 1	Mode 2
98	2
99	1
0	0
1	99
2	98
...	...

Table 1: Example sequences of timer

Figure 1: Circuit diagram of Part-1

3 Part-2

In this question, you are required to implement a number game on the design is given in Figure 2 (Design link). The aim of the game is to guess the target number (i.e., 3 digits long) by means of push button presses For this game, firstly create a 4-bit counter (0-F) that is displayed on the left-most 7-segment display. Counter should display the numbers at a rate close to 1 number per second. The player sees the number on the 7-segment

display and presses the button to select the digit on the counter. After the selection, the selected number should be shown on the 7-segment display on the right. By means of the following selections the displayed numbers on the three rightmost 7-segment displays should be slid as shown in the example below:

Assume that the current selected digit from the 7-segment display is n(t) , where t represents the time, then the 7-segment displays should have the following configuration:

counter digit $n(t)$ $n(t-1)$ $n(t-2)$
after a push button press, the configuration changes as:
counter digit $n(t+1)$ $n(t)$ $n(t-1)$
after another push button press, the configuration changes as:
counter digit $n(t+2)$ $n(t+1)$ $n(t)$
Until the new selected number matches with the target, game continues. When they match, game stops. Target number should be defined as a variable. should
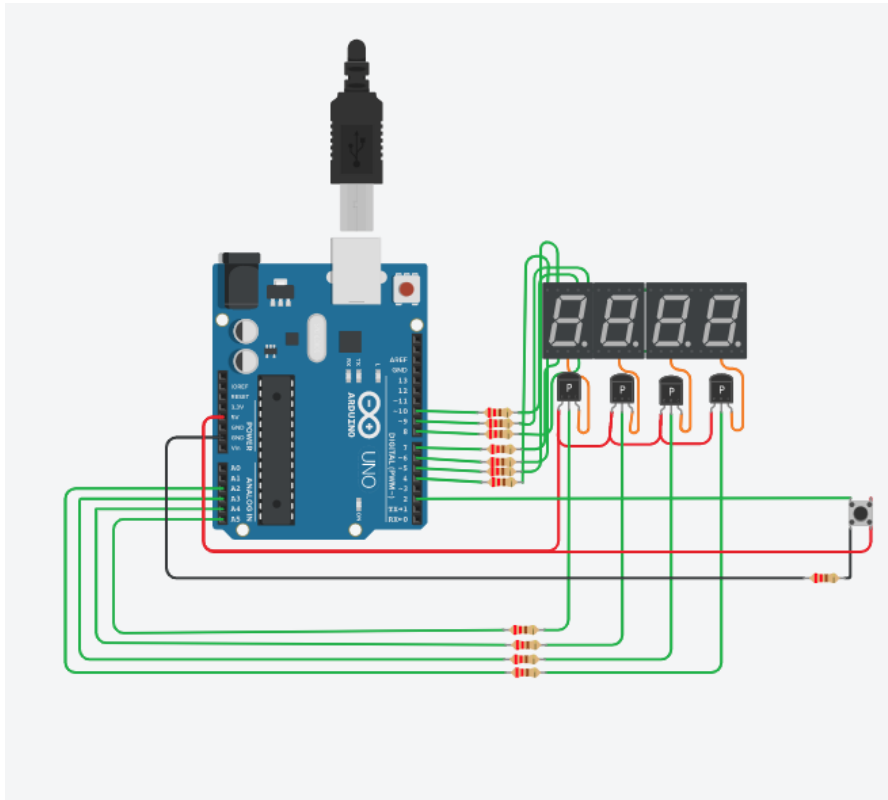Note that, we expect you to implement interrupt subroutine for button press.

Figure 2: Circuit diagram of Part-2

3

4 Part-3

In this part, you should use the design is given in the Figure 3 (Design link). You are required to develop a 2-player number game using push-buttons. The aim of the game is to try and catch the number of the other player. At the beginning of the game, a counter that counts continuously from 1 to 20 should start on two left-most 7-segment display. The counter should display the numbers at a rate close to 1 number per 200 ms. When the first player presses the first button, the selected number is shown on two left-most 7-segment display. After the number chosen by the first player is shown on the screen, similar counter will start on two right-most 7-seven segment display. Second player will press the second button to catch the number chosen by the first player. If the number chosen by second player is the same as the first number, then player 2 wins the game. Otherwise, the first counter will start and the first player try to catch the number chosen by the second player. The game continues until one of the players catches the number chosen by the other players. When the game is over, the winner of the game is indicated on all 7-segment display in the form of "1111" or "2222".
Note that, we expect you to implement interrupt subroutine for button press.
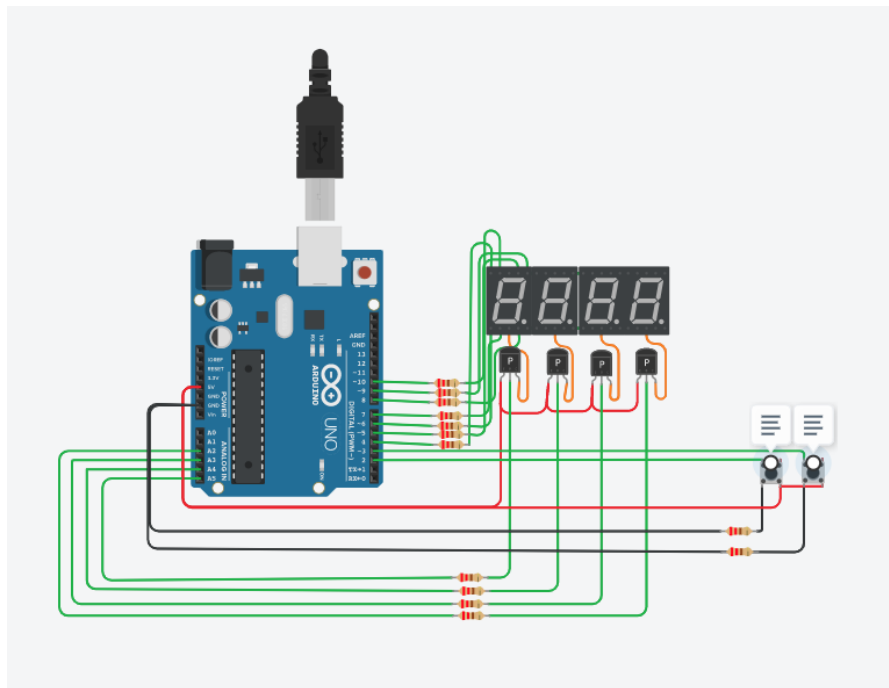

Figure 3: Circuit diagram of Part-3

5 Submission

- You should upload your experiment codes and report on Ninova, and please, do not send your experiment files via e-mail.

- You must upload each part's code seperately to the ninova.

- Your reports must be written with Latex format. Latex report template is available on Ninova. You can use any Latex editor whichever you want. If you upload your report without Latex file, you directly get 0 as your report grade. You should upload both .pdf and .tex files of your report.

- Finally, please do not forget that late submissions are not accepted.