COUNT-DOWN TIMER

Bodhiswattwa Basu
PN: 50266718

FALL - NOVEMBER 5, 2021

CSE 321 – REAL-TIME EMBEDDED SYSTEMS
State University of New York at Buffalo

INTRODUCTION

Countdown systems are integrated in our daily lives – in our phones, watches, microwave and conventional ovens, coffee machines, etc. This project is an implementation of an all-in-one count-down timer with a 4x4 keypad, a liquid crystal display (LCD) and a Nucleo-L4R5ZI. The count-down timer is wired to a solderless breadboard with seven light emitting diodes (LEDs).

SPECIFICATIONS

There are three connections made from the Nucleo-L4R5ZI, one to the LCD, one to the 4x4 keypad and one to the solderless breadboard.

Figure 1 is schematic image of the connections from the Nucleo-L4R5ZI to the 4x4 keypad.

Each of the columns have an LED in series connection – once a key is pressed, an LED lights up in response.

There are three other LEDs that are connected to PA_3, PC_0 and PC_3.

These LEDs are dependent on when the timer reaches the value of 0 minutes and 0 seconds, once reached, they all illuminate.

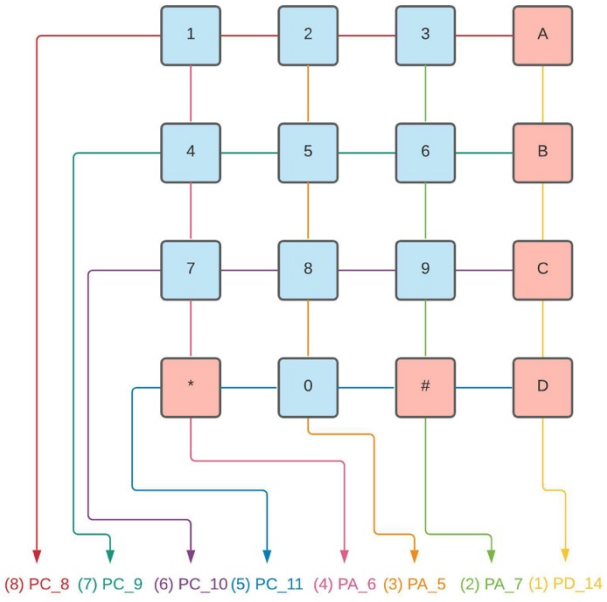

Figure 1 4x4 Keypad Connections with Nucleo-L4R5ZI

The user is permitted to only enter three digits to preserve the *m:ss* format. The constraint of this system is that it is not able to set a timer value past 9 minutes and 59 seconds.

Once the count-down timer begins, its termination can occur at any time by pressing 'B'. This will stop the timer and force the user to restart the program to enter in timer values again.

FEATURES

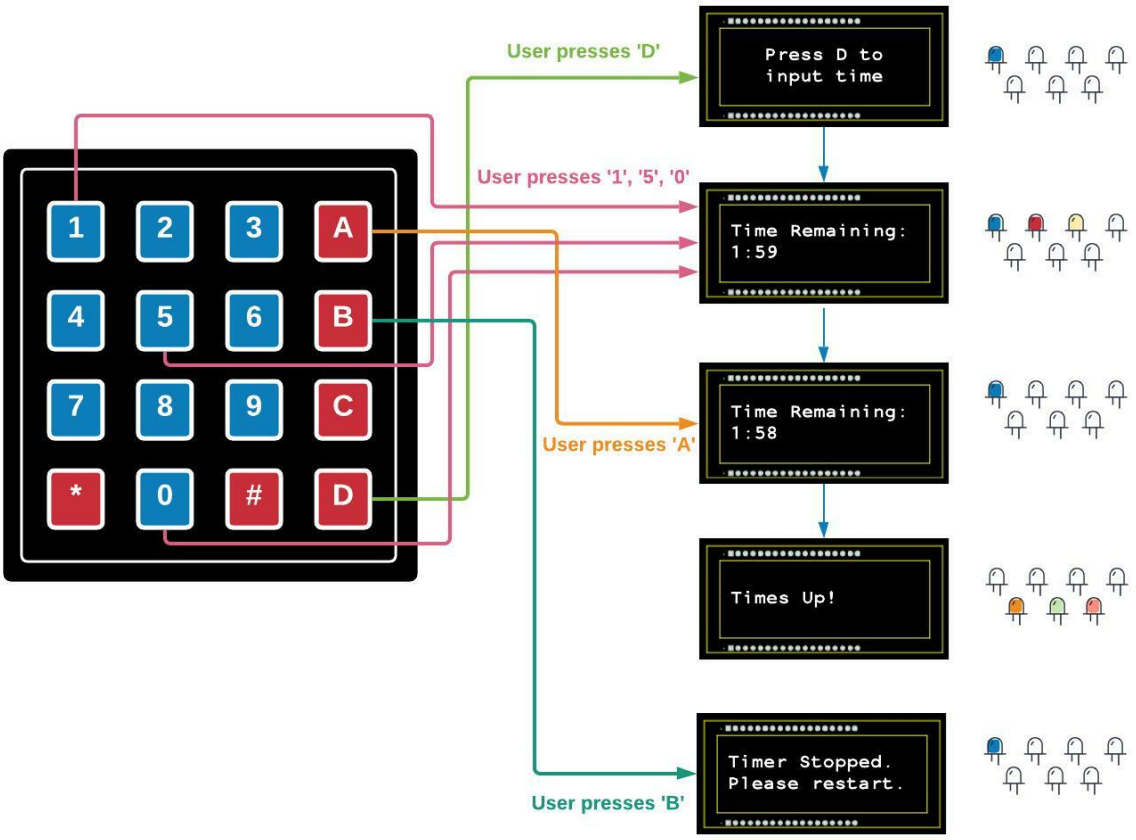

Figure 2 Schematic of Features of Keypad

The figure above displays a schematic representation of the features of the all-in-one count-down timer.

To begin to input values into the timer, the user needs to press 'D'.

Once 'D' is pressed, the user can enter timer values. As each key for the timer value is pressed, an LED will light up correspondingly.

To start the timer, the user needs to press 'A'. Once 'A' is pressed, the count-down timer will begin.

If at any time, the user wants to stop or turn off the timer, 'B' can be pressed. If the user presses 'B' at any point when the timer has been deployed, the MBED Studios program can be run again to restore it back to its beginning state.

Once the timer ends, the LCD will read 'Times Up' and three LEDs will light up consequently.

APPLICATIONS

Since the timer cannot go beyond 9 minutes and 59 seconds, its applications are limited in quick activities.

An egg cooker timer would be an excellent implementation of this timer since an egg is cooked under 10 minutes for a soft-boil.

A high-intensity interval training (HIIT) is a common exercising technique that utilises short time intervals with high-calorie burning exercise until exhaustion. Usual HIIT intervals are for 2 minutes and span for no longer than 6 minutes. This timer would be the perfect product for an HIIT session.

Since the timer uses LED for indications, this implementation could be used for people with hearing disabilities since it depends on the LCD and the lights to dictate keypad presses and timer execution.

BLOCK DIAGRAM

The following figure is a diagrammatic representation of the timer from when it begins execution to when the timer runs out.

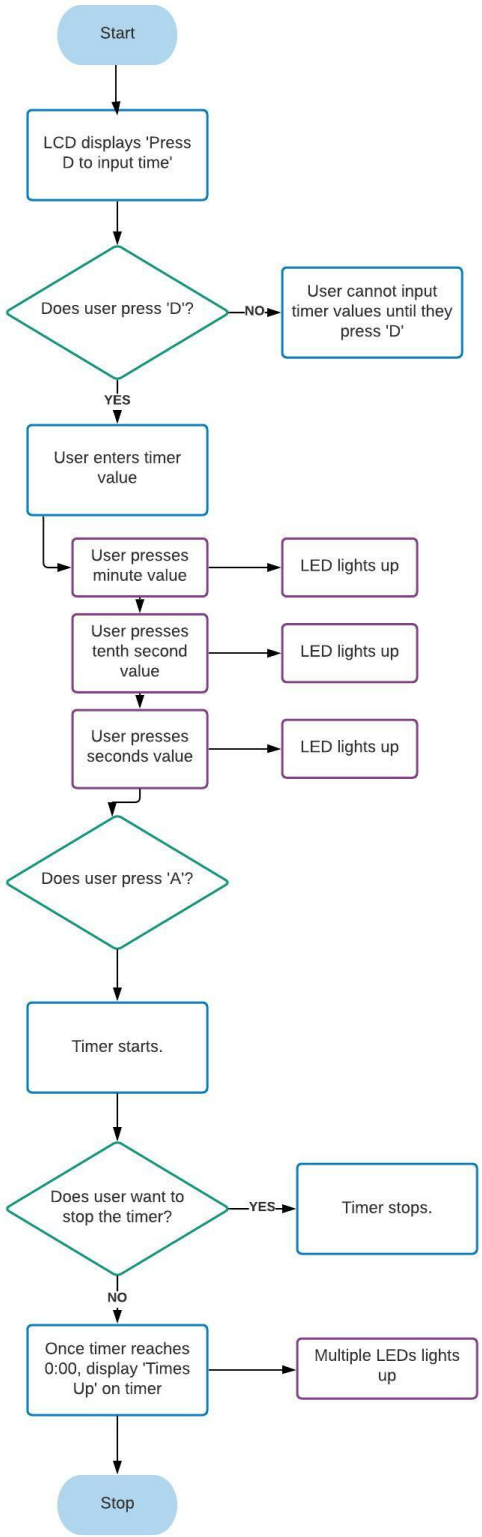

Figure 3 Block Diagram

FUNCTIONALITY DIAGRAMS

The following is the functionality diagram for the program.

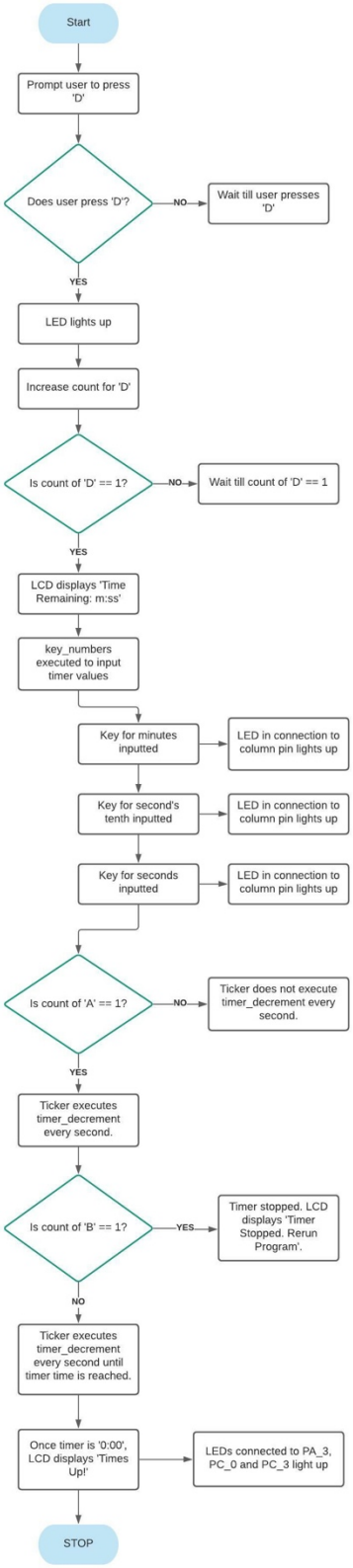

Figure 4 Functionality Diagram

BILL OF MATERIALS

The following list of materials are required to be able to apply this implementation of the all-in-one count-down timer.

Table 1 Bill of Materials

Description	*Manufacturer*	*Where to Find*
STM32 Nucleo-144 board	STMicroelectronics	https://www.st.com/en/evaluation-tools/nucleo-l4r5zi.html#sample-buy
Breadboard kit	REXQualis	https://www.amazon.com/REXQualis-Electronics-tie-Points-Breadboard-Potentiometer/dp/B073ZC68QG
4x4 Matrix Membrane Keypad	SparkFun	https://www.sparkfun.com/products/16038
16x2 I2C LCD Display Module	Inland	https://www.microcenter.com/product/632704/16x2_I2C_LCD_Display_Module

SCHEMATICS

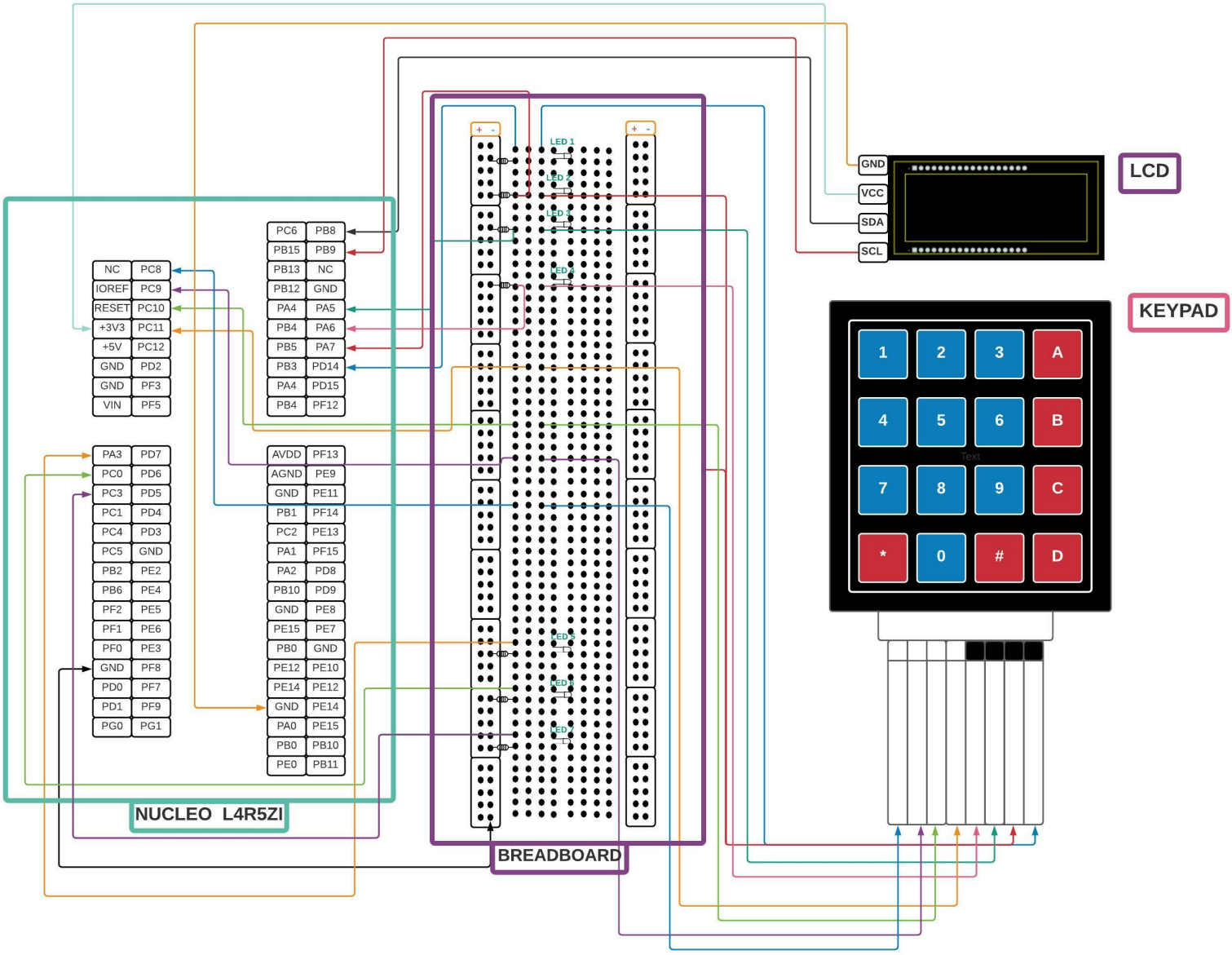

Figure 5 Schematic Diagram

TEST PLAN

The following section of the report displays the test plans that were implemented to adjust for the constraints of this project.

Test Plan #1: User should only be able to input times once 'D' has been pressed.

Once the program is executed, the LCD will display 'Press D to input time' if the counter for key 'D' is equal to one. The keys for the numbers will not output any changes onto the LCD unless the counter for key 'D' increases to equal to one. Once the user presses 'D', the next screen would display 'Time Remaining: m:ss' – which they can then use to input the timer value.

Test Plan #2: User should only be able to input three digits for the timer's value.

The user should only be allowed to enter 3 digits for the timer value. Every time a number key is pressed, variable `digit_i` is incremented by one. Once `digit_i` reaches a value of 3, the function `key_numbers` will stop accepting further values to add to the final character array `fin`.

Test Plan #3: User should not be able to input a seconds value greater than 59 seconds.

The function `key_numbers` will only accept values for the first digit of the seconds value that is lesser than the key '6'.

Test Plan #3: User should not be able to input a seconds value greater than 59 seconds.

The function `key_numbers` will only accept values for the first digit of the seconds value that is lesser than the key '6'. The rest of the values of the timer can accept values between 0 and 9.

Test Plan #4: User should not be able to start the timer until A is pressed.

Once the timer values are inputted in by the user, the LCD will display 'Time remaining: 1:20' until the user presses key 'A' to begin the timer. At this point, the program will not be able to execute further until 'A' is pressed. Once key 'A' is

pressed, the ticker will point to the function that will decrement the character array `count_down` by one every second.

Test Plan #5: User should be able to press 'B' to stop/turn off the timer.

Similar to the implementation of a counter for keys 'A' and 'D', key 'B' has a counter associated with it. Once 'B' is pressed, the counter for key 'B' is incremented by one. In the main function, if 'B' is pressed during execution, it will stop the timer and display 'Timer Stopped. Rerun program.' on the LCD to indicate to the user that the timer has been forcibly stopped.

Test Plan #6: Every time a value is entered, an LED lights up.

To have each key press associated with an LED lighting up, an LED was connected in series to each of the columns of the keypad. Once a key from a corresponding column of the keypad is pressed, the LED connected to that column will light up in response.

Test Plan #7: Once the time is up, multiple LEDs should light up.

Three LEDs are connected to three pins (PA_3, PC_0 and PC_3)on the Nucleo-L4R5ZI. These three pins are configured to output mode so that once the `count_down` reaches a value of '0:00', the three LEDs will light up in response.

RESULTS

Test Plan #1: User should only be able to input times once 'D' has been pressed

Initial LCD Display: **`Press D to input time.`**

Subtest #1: Keypad press of key '1'

Result of subtest #1:

Final LCD Display: **`Press D to input time.`**

Subtest #2: Keypad press of key '3'

Result of subtest #2:

Final LCD Display: **`Press D to input time.`**

Subtest #3: Keypad press of key 'D'

Result of subtest #3:

Final LCD Display: `Time Remaining: m:ss`

Test Plan #2: User should only be able to input three digits for the timer's value

Initial LCD Display: `Time Remaining: 1:30`

Subtest #1: Keypad press of key '1'

Result of subtest #1:

Final LCD Display: `Time Remaining: 1:30`

Subtest #2: Keypad press of key '9'

Result of subtest #2:

Final LCD Display: `Time Remaining: 1:30`

Subtest #3: Keypad press of key '3'

Result of subtest #3:

Final LCD Display: `Time Remaining: 1:30`

Test Plan #3: User should not be able to input a seconds value greater than 59 seconds

Initial LCD Display: `Time Remaining: m:ss`

Subtest #1: Keypad press of key '1', '6' and then '0'

Result of subtest #1:

Final LCD Display: `Time Remaining: 1:ss`

Subtest #2: Keypad press of key '9', '8' and then '9'

Result of subtest #2:

Final LCD Display: **Time Remaining: 9:ss**

Subtest #3: Keypad press of key '3', '6', and then '9'

Result of subtest #3:

Final LCD Display: **Time Remaining: 3:ss**

Test Plan #4: User should not be able to start the timer until A is pressed

Initial LCD Display: **Time Remaining: 1:30**

Subtest #1: Keypad press of key '1'

Result of subtest #1:

Final LCD Display: **Time Remaining: 1:30**

Subtest #2: Keypad press of key '9'

Result of subtest #2:

Final LCD Display: **Time Remaining: 1:30**

Subtest #3: Keypad press of key 'A'

Result of subtest #3:

Final LCD Display 1: **Time Remaining: 1:30**

Final LCD Display 2: **Time Remaining: 1:29**

Final LCD Display 3: **Time Remaining: 1:28**

Test Plan #5: User should be able to press 'B' to stop/turn off the timer

Initial LCD Display: **Time Remaining: 1:30**

Time Remaining: 1:29

Time Remaining: 1:28

Time Remaining: 1:27

.
.
.

Subtest #1: Keypad press of key '4'

Result of subtest #1:

Final LCD Display: **Time Remaining: 1:26**

Subtest #2: Keypad press of key '8'

Result of subtest #2:

Final LCD Display: **Time Remaining: 1:25**

Subtest #3: Keypad press of key 'B'

Result of subtest #3:

Final LCD Display 1: **Timer Stopped.**

Please rerun program.

Test Plan #6: Every time a value is entered, an LED lights up

Initial LED state:

```
LED 1 == off
LED 2 == off
LED 3 == off
LED 4 == off
```

Subtest #1: Keypad press of key '1'

Result of subtest #1:

Final LED state:

```
LED 1 == on
LED 2 == off
LED 3 == off
LED 4 == off
```

Subtest #2: Keypad press of key '4'

Result of subtest #2:

Final LED state:

```
LED 1 == off
LED 2 == on
LED 3 == off
LED 4 == off
```

Subtest #3: Keypad press of key '6'

Result of subtest #3:

Final LED state:

```
LED 1 == off
LED 2 == off
LED 3 == on
LED 4 == off
```

Subtest #4: Keypad press of key 'A'

Result of subtest #4:

Final LED state:

```
LED 1 == off
LED 2 == off
LED 3 == off
LED 4 == on
```

Test Plan #7: Once the time is up, multiple LEDs should light up.

Initial LED state:

```
LED A == off
LED B == off
LED C == off
```

Subtest #1: LCD Displays 'Time Remaining: 1:30'

Result of subtest #1:

Final LED state:

```
LED A == off
LED B == off
LED C == off
LED 4 == on
```

Subtest #2: LCD Displays 'Time Remaining: 0:05'

Result of subtest #2:

Final LED state:

```
LED A == off
LED B == off
LED C == off
```

Subtest #3: LCD Displays 'Times Up!'

Result of subtest #3:

Final LED state:

```
LED A == on
LED B == on
LED C == on
```

RECOMMENDATIONS FOR IMPROVEMENTS

Improvements can be made to implementations to better it for everyday and large-scale usage. The implementation of the timer can be improved in the following ways:

- The timer values restriction can be subdued to take in timer values greater than 9 minutes and 59 seconds.
- The timer should also have an implementation for those who have visual disabilities by adding in a component that would generate noise every time the keys on the keypad is pressed and when the timer begins and ends.
- The LED circuitry on the breadboard could be printed out on a printed circuit board and attached to the keypad and LCD.