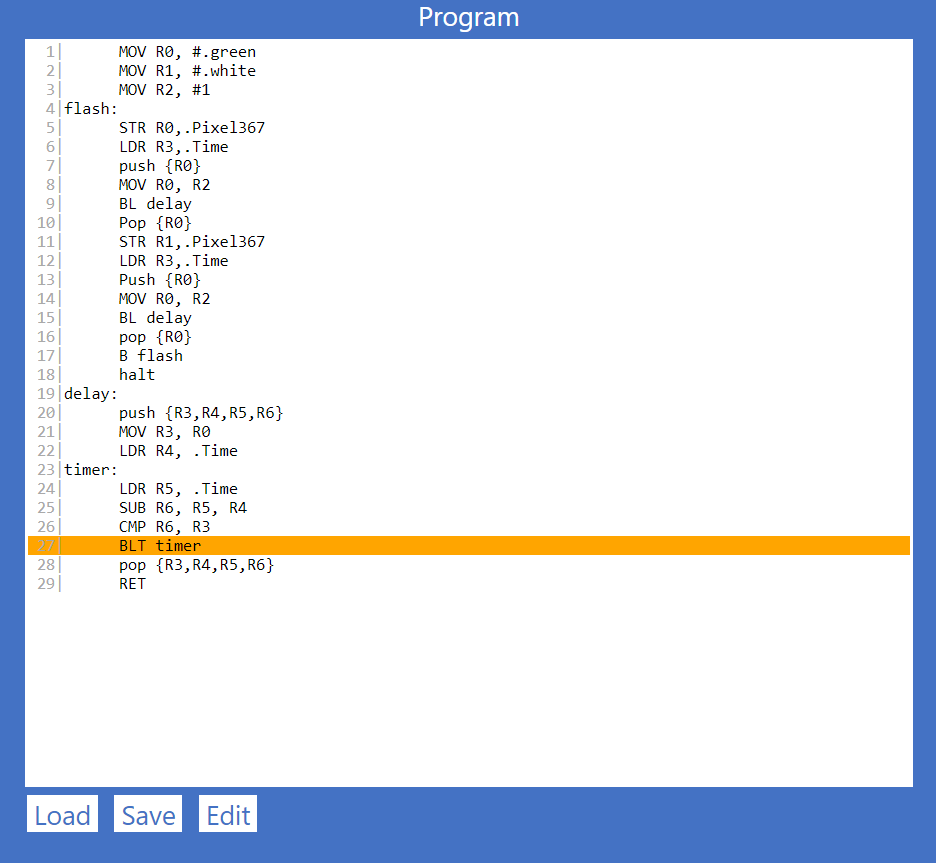
Lab10

Nguyen Nam Tung 103181157

**Exercise 10.1**

Verify your understanding of our modifications by implementing the same changes yourself (*Try not to just copy the solution from lectures - give it a go yourself and use the lecture example as a check point if you need!).*

(a) First write the delay function.  This function should take a single input, the number of seconds to delay for and be called from the main program every time there is a pause required.



MOV R0, #.green

MOV R1, #.white

MOV R2, #1

flash:

STR R0,.Pixel367

LDR R3,.Time

push {R0}

MOV R0, R2

BL delay

Pop {R0}

STR R1,.Pixel367

LDR R3,.Time

Push {R0}

MOV R0, R2

BL delay

pop {R0}

B flash

halt

delay:

push {R3,R4,R5,R6}

MOV R3, R0

LDR R4, .Time

timer:

LDR R5, .Time

SUB R6, R5, R4

CMP R6, R3

BLT timer

pop {R3,R4,R5,R6}

RET

(b) Then write the drawpixel function.  This function should take two inputs: the colour of the pixel to draw, and the time delay between on and off.  This function should also call the delay function to insert the pauses between on and off.

(c) when you implemented drawpixel, what did you have to do with LR to make it work?  Why?

***When your happy with your solution, show your tutor the solution you have written and demonstrate it.   Take a screen shot and include, and provide your answers  in your solution document.***

**Exercise 10.2**

 Modify your program above so that your program flashes the LED rapidly (i.e., 1 second between on and off)  three times, and then pauses for  2 seconds before repeating the rapid pattern.  You should make use of your drawpixel and delay functions to do this.

Ảnh có chứa văn bản

Description automatically generated

1| mov r2, #1 ; 1s delay time

2| mov r7, #2 ;2s delay time

3|loop:

4| mov r8, #0

5|flash:

6| mov r0, #.blue

7| mov r1, r2

8| bl drawpixel

9| mov r0, #.red

10|; mov r1, r2

11| bl drawpixel

12|;end of 1 flash on and off

13| add r8, r8, #1

14| cmp r8, #3 ;3 times on and off

15| blt flash

16|cont:

17| mov r0, #.white ;I use white for the pause time to observe

18| mov r1, r7

19| bl drawpixel

20| add r8, r8, #1

21| cmp r8, #5

22| blt loop

23| halt

24|;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

25|;; drawpixel function

26|;; inputs R0 - colour,

27|;; R1 - time delay in seconds

28|drawpixel:

29| push {R3, R4}

30| mov r3, r0 ;copy pixel color to r3

31| mov r4, r1 ;copy delay time to r4

32| str r3, .Pixel367 ;Draw color to the pixel

33| push {r0, lr}

34| mov r0, r4 ;pass delay time to the function

35| bl delay ;call delay

36| pop {r0, lr}

37| ret

38|;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

39|;; delay function

40|;; inputs R0 - time delay in seconds

41|delay:

42| push {R3, R4, R5, R6}

43| mov r3, r0

44| ldr r4, .Time

45|timer:

46| ldr r5, .Time

47| sub r6, r5, r4

48| cmp r3, r6

49| bne timer

50| pop {R3,R4,R5,R6}

51| RET

***Show your solution to your tutor.   Take a screen shot and include in your solution document.***

**Exercise 10.3**

Modify your program from 10.2 by writing a function called flashpattern: that accepts two inputs:

* the number of "rapid" 1 second flashes before the pause
* the pause time (in seconds) between each set of rapid flashes

You should call this function from your main program and allow it to manage the flashing of your pixel LED.  This function should make use of your previous functions to perform the task.

Ảnh có chứa văn bản

Description automatically generated

1| mov r2, #1 ; Rapid 1s delay time

2| mov r7, #0 ; Wanted pause time

3| mov r9, #0 ; Number of rapid1 second flashes

4| bl flashpattern

5|loop:

6| mov r8, #0

7|flash:

8| mov r0, #.blue

9| mov r1, r2

10| bl drawpixel

11| mov r0, #.red

12| mov r1, r2

13| bl drawpixel

14|;end of 1 flash on and off

15| add r8, r8, #1

16| cmp r8, r9 ;3 times on and off

17| blt flash

18|cont:

19| mov r0, #.white ;I use white for the pause time

20| mov r1, r7

21| bl drawpixel

22| add r8, r8, #1

23| add r9, r9, #2

24| cmp r8, r9

25| blt loop

26| halt

27|;The messages

28|askrapid: .ASCIZ "\n Enter the number of 1-second flashes you want before the pause: "

29|askpausetime: .ASCIZ "\n Enter your desired pause time: "

30|;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

31|;; drawpixel function

32|;; inputs R0 - colour,

33|;; R1 - time delay in seconds

34|drawpixel:

35| push {R3, R4}

36| mov r3, r0 ;copy pixel color to r3

37| mov r4, r1 ;copy delay time to r4

38| str r3, .Pixel367 ;Draw color to the pixel

39| push {r0, lr}

40| mov r0, r4 ;pass delay time to the function

41| bl delay ;call delay

42| pop {r0, lr}

43| ret

44|;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

45|;; delay function

46|;; inputs R0 - time delay in seconds

47|delay:

48| push {R3, R4, R5, R6}

49| mov r3, r0

50| ldr r4, .Time

51|timer:

52| ldr r5, .Time

53| sub r6, r5, r4

54| cmp r3, r6

55| bne timer

56| pop {R3,R4,R5,R6}

57| RET

58|;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

59|;; flashpattern function

60|flashpattern:

61| push {R3, R4, R5, R6, lr}

62|;Take first input

63| mov r3, #askrapid

64| str r3, .WriteString

65| ldr r4, .InputNum

66| str r4, .WriteUnsignedNum

67|;Take second input

68| mov r5, #askpausetime

69| str r5, .WriteString

70| ldr r6, .InputNum

71| str r6, .WriteUnsignedNum

72|;Add input into the right register

73| mov r9, r4

74| mov r7, r6

75| pop {R3,R4,R5,R6, lr}

76| ret

***Show your solution to your tutor.   Take a screen shot and include in your solution document***