MEF ÜNİVERSİTESİ

ÖĞRENMEYE SINIRSIZ ÖZGÜRLÜK

EE 306 - Microprocessors

2020-2021

NUMBER GUESSING GAME SOURCE CODE

Selçuk Coşkun 0418010

Source Code

```
//Selcuk Coskun 041801079 Number Guessing Game
.global _start
_start:
//Declare tools
.equ TIMER, 0xfffec600
.equ LED, 0xff200000
.equ SWITCH, 0xff200040
.equ SEG, 0xff200020
.equ ADDR_KEYBUTTON, 0xff200050
//Declerations
ldr r0, =SEG
ldr r10,=ADDR_KEYBUTTON
ldr r4, =SWITCH
ldr r8, =TIMER
ldr r6, =LED

START:

mov r1, #0
//r2 is the attempt number which is 4
mov r2, #3
ldr r3,[r4]

//Configure LEDs
mov r5, #0
str r5, [r6]
mov r5, #1

//Arranging TIMER
LDR r9, =2000000
STR r9, [r8]
MOV r9, #0b011
STR r9, [r8, #0x8]

Rand:
//Pseudo-Random number generator
MOVS r7,r7,lsr #1
EORCC r7,r7,#1<<14
TST    r7,#1
EORNE r7,r7,#1<<14

/*
//Deprecated Timer-Based pseudo-random generator
//This blocked the games loop so it is deprecated
ldr r7,[r8,#4]
LSR R7,#16
AND R7,#3
STR R7,[R11]
*/
```

```
//Left shift to ensure on the last 2 values are taken so a 2 digit number is obtained
LSL R7, #5

//Number gets checked if higher than 20
NumCheck:
cmp R7, #20
//Subtract 20 if higher than 20
subgt r7, r7, #20
Bgt NumCheck
cmp r7, #0
//add 1 if in negatives
addlt r7, r7, #1
Blt NumCheck
//if both conditions are satisfied move on to user input
B UpdateGuess

//Initialize first attempt

UpdateGuess:
mov r14,#0 //Checking whether or not push buttons are pressed
ldr r13,[r10,#0xC]
cmp r13,#0
eorne r14,r14,#1
strne r13,[r10,#0xC]
cmp r14,#1
bne UpdateGuess
ldr r3,[r4]
//Update LED to indicate attempt number
str r5, [r6]
mov r14,#0
//Once a push button is pressed move on to the checking process to see if the guess
was correct
b Guess


Guess:
//Compares to see if the player has any attempts left
cmp r2,#0
//if equal it'll redirect to the fail subroutine
beq FAIL
//Since r8 is not used anymore it is used to keep the multiplication value to
increase the LEDs
mov r8, #2
cmp r3, r7
//If the guessed number is higher than the random one SEGDown is initiated
bgt SEGDown
//If the guessed number is lower than the random one SEGUp is initiated
blt SEGUp
//If the guessed number is equal to the random one SUCCESS is initiated
beq SUCCESS

SEGDown:
//Subtract 1 attempt
sub r2, r2 ,#1
//Increase the LED attempt counter value for usage on line 70
```

```
mul r5, r8
//Update 7 segment display to display a lower line to indicate "guess lower"
mov r1, #8
str r1, [r0]

B UpdateGuess
SEGUp:
sub r2, r2 ,#1
//Increase the LED attempt counter value for usage on line 70
mul r5, r8
//Update 7 segment display to display a upper line to indicate "guess higher"
mov r1, #1
str r1, [r0]
//Go back to UpdateGuess to get user input
B UpdateGuess

FAIL:
//All LEDs shut down, program ends
mov r5, #0
str r5, [r6]
mov r2, #3
mov r1, #118
str r1, [r0]
B END

SUCCESS:
//All LEDs light up, program ends
LDR r8, =2047
mov r5, r8
str r5, [r6]
mov r2, #3
mov r1, #63
str r1, [r0]
B ABORT

END:
//Reset necessary values and start over if failed
mov r8, #0
mov r13, #0
mov r14, #0
B START

ABORT:

.end
```