Αναφορά 5ης Εργαστηριακής Άσκησης

Στοιχεία Ομάδας:

Δημήτριος Κωστορρίζος, up1054419@upnet.gr

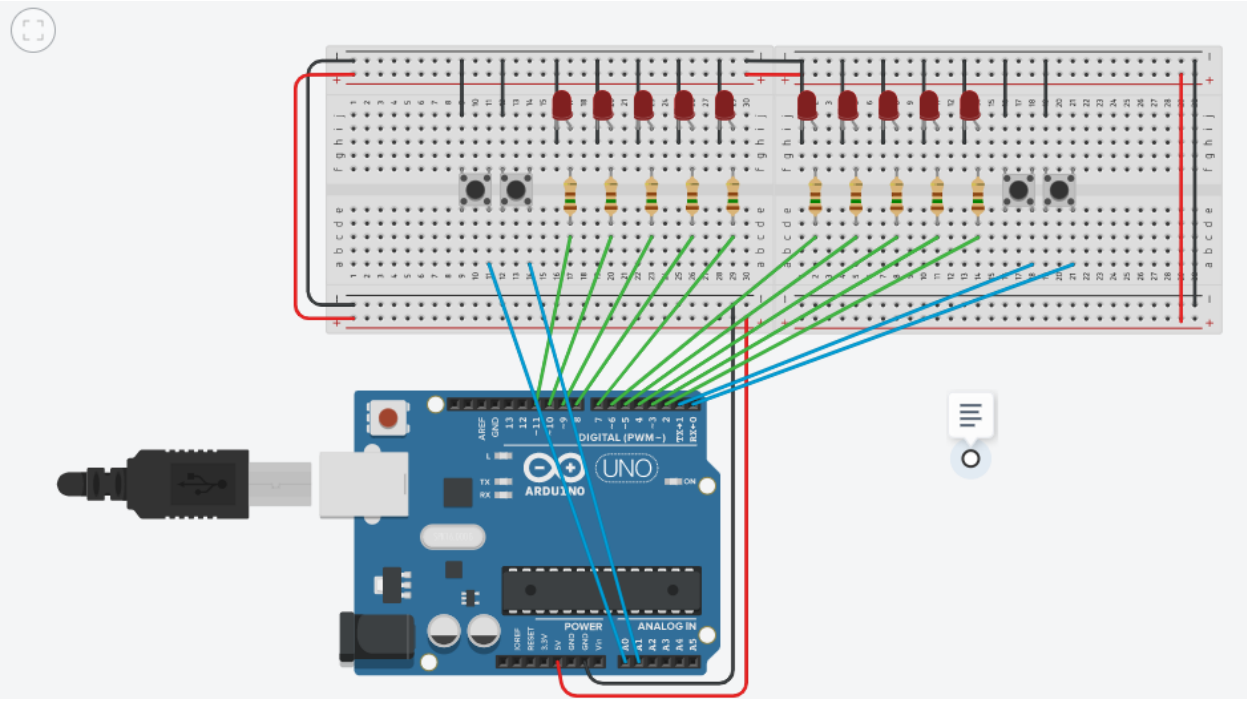
Λάμπρος Παπαδόπουλος, up1054433@upnet.gr

Ομάδα Β

Τμήμα: 1

Tinkercad Link: https://www.tinkercad.com/things/lVrhIDMQsQf-askisi-5/editel

Εικόνα Κυκλώματος:



Περιγραφή υλοποίησης:

Στην υλοποίηση του Arduino , έχουν χρησιμοποιηθεί ως είσοδοι, τα pin Analog In A0 και A1. Οι αντιστάσεις που έχουν χρησιμοποιηθεί στην υλοποίηση είναι ίδιες με την αντίσταση 1K Ohm, που υπήρχε στην αρχική υλοποίηση. Αντί για συστοιχίες LED, έχουν χρησιμοποιηθεί μεμονωμένα LED, τα οποία ρυθμίζονται από τον κώδικα της υλοποίησης. Τα LED του παίχτη 1 αντιστοιχούν στα pin 7-11 του Arduino και τα LED του παίχτη 2 στα pin 2-6. Τα κουμπιά που είναι συνδεδεμένα στις Analog In θύρες αντιστοιχούν στον παίχτη 1 και τα κουμπιά που είναι συνδεδεμένα στα pin 0-1 στον παίχτη 2.

Κώδικες

Arduino

```
void FIQ_handler(void);

//Button constants
#define off 0
#define on 1
#define BUT_IDLE 0
#define BUT_PRESSED 1

//Led_Constants
#define hold 0
#define game 1
#define HIGH 0x1
#define LOW 0x0
#define high 1
#define low 0

//Timer constants
#define continue_count  0x01
#define stop_count  0x02
#define reset_and_start_count  0x05


unsigned int PIOA_int, TC_int;
unsigned long clockPulseCounter;
unsigned int Channel_0_CCR;
```

```
unsigned long Channel_0_RC;


unsigned int playerNumber = 0;
unsigned int led = 0;
unsigned int ledState = hold;
unsigned int player1Score = 0;
unsigned int player2Score = 0;
unsigned int player1Button = BUT_IDLE;
unsigned int currentPlayer1ButtonState = BUT_IDLE;
unsigned int player2Button = BUT_IDLE;
unsigned int currentPlayer2ButtonState = BUT_IDLE;
unsigned int previousPlayer1 = BUT_PRESSED;
unsigned int previousPlayer1ButtonState = BUT_PRESSED;
unsigned int previousPlayer2 = BUT_PRESSED;
unsigned int previousPlayer2ButtonState = BUT_PRESSED;
unsigned int player1State = off;
unsigned int player2State = off;
unsigned int n1_state = off;
unsigned int n2_state = off;
unsigned int player1Counter = 0;
unsigned int player2Counter = 0;
unsigned int n1_counter = 0;
unsigned int n2_counter = 0;
unsigned int winCounter = 0;
unsigned int player1WinFlag = 0;
unsigned int player2WinFlag = 0;

void setup()
{

        pinMode(2, OUTPUT);
        digitalWrite(2, high);

        pinMode(3, OUTPUT);
        digitalWrite(3, high);

        pinMode(4, OUTPUT);
        digitalWrite(4, high);

        pinMode(4, OUTPUT);
        digitalWrite(4, high);

        pinMode(6, OUTPUT);
```

```
        digitalWrite(6, high);

        pinMode(7, OUTPUT);
        digitalWrite(7, high);

        pinMode(8, OUTPUT);
        digitalWrite(8, high);

        pinMode(9, OUTPUT);
        digitalWrite(9, high);

        pinMode(10, OUTPUT);
        digitalWrite(10, high);

        pinMode(11, OUTPUT);
        digitalWrite(11, high);

        pinMode(0, INPUT_PULLUP);
        pinMode(1, INPUT_PULLUP);
        pinMode(A0, INPUT_PULLUP);
        pinMode(A1, INPUT_PULLUP);


  clockPulseCounter=0;
  Channel_0_CCR = stop_count;
  Channel_0_RC = 8192;
        PIOA_int = LOW;
        TC_int = LOW;
}

void loop()
{
        if (Channel_0_CCR == reset_and_start_count)
        {
        clockPulseCounter = 0;
        Channel_0_CCR = continue_count;
        }

        if (Channel_0_CCR == continue_count)
        {
                clockPulseCounter++;
                if (clockPulseCounter == Channel_0_RC)
                {
                clockPulseCounter =0;
```

```
                TC_int = HIGH;
                }
    }

        player1Button = digitalRead(0);
        if (player1Button != previousPlayer1)
        {
                PIOA_int = HIGH;
        }

        currentPlayer1ButtonState = digitalRead(1);
        if (currentPlayer1ButtonState != previousPlayer1ButtonState)
        {
                PIOA_int = HIGH;
        }

        player2Button = digitalRead(A1);
        if (player2Button != previousPlayer2)
        {
                PIOA_int = HIGH;
        }

        currentPlayer2ButtonState = digitalRead(A0);
        if (currentPlayer2ButtonState != previousPlayer2ButtonState)
        {
                PIOA_int = HIGH;
        }

        if (PIOA_int | TC_int)
        {
                FIQ_handler();
        }
}

void FIQ_handler()
{
        if (PIOA_int == HIGH)
        {
                PIOA_int = LOW;

                int player1 = ((player1Button != previousPlayer1) ? 1 : 0);
                previousPlayer1 = ((player1) ? player1Button : previousPlayer1);

                int player2 = ((player2Button != previousPlayer2) ? 1 : 0);
```

```
		previousPlayer2 = ((player2) ? player2Button : previousPlayer2);

		int n1 = ((currentPlayer1ButtonState != previousPlayer1ButtonState) ? 1 : 0);
		previousPlayer1ButtonState = ((n1) ? currentPlayer1ButtonState :
previousPlayer1ButtonState);

		int n2 = ((currentPlayer2ButtonState != previousPlayer2ButtonState) ? 1 : 0);
		previousPlayer2ButtonState = ((n2) ? currentPlayer2ButtonState :
previousPlayer2ButtonState);

		if (winCounter>0)
			return;
		else if(player1)
		{
			if(player1Counter == 0)
			{
				player1Counter = 6;
				player1State = on;
				if(ledState == hold || player2WinFlag == 1)
				{
					ledState = game;
					playerNumber = 2;
					led = 2;
					player2WinFlag = 0;
					digitalWrite(3, low);
					digitalWrite(4, low);
					digitalWrite(5, low);
					digitalWrite(6, low);
					digitalWrite(7, low);
					digitalWrite(8, low);
					digitalWrite(9, low);
					digitalWrite(10, low);
					digitalWrite(11, low);
					digitalWrite(2, high);
					Channel_0_CCR = 0x05;
				}
			}
		}

		else if(n1)
		{
			if(n1_counter == 0)
			{
				n1_counter = 6;
```

```
                n1_state = on;
                if(ledState == hold || player2WinFlag == 1)
                {
                        ledState = game;
                        playerNumber = 2;
                        led = 2;
                        player2WinFlag = 0;
                        Channel_0_RC/=5;
                        digitalWrite(3, low);
                        digitalWrite(4, low);
                        digitalWrite(5, low);
                        digitalWrite(6, low);
                        digitalWrite(7, low);
                        digitalWrite(8, low);
                        digitalWrite(9, low);
                        digitalWrite(10, low);
                        digitalWrite(11, low);
                        digitalWrite(2, high);
                        Channel_0_CCR = 0x05;
                }
        }
}

else if(player2)
{
        if(player2Counter == 0)
        {
                player2Counter = 6;
                player2State = on;
                if(ledState == hold || player1WinFlag == 1)
                {
                        ledState = game;
                        playerNumber = 1;
                        led = 11;
                        player1WinFlag = 0;
                        digitalWrite(2, low);
                        digitalWrite(3, low);
                        digitalWrite(4, low);
                        digitalWrite(5, low);
                        digitalWrite(6, low);
                        digitalWrite(7, low);
                        digitalWrite(8, low);
                        digitalWrite(9, low);
                        digitalWrite(10, low);
```

```
                                digitalWrite(11, high);
                                Channel_0_CCR = 0x05;
                        }
                }
        }

        else if(n2){
                if(n2_counter == 0)
                {
                        n2_counter = 6;
                        n2_state = on;
                        if(ledState == hold || player1WinFlag == 1 )
                        {
                                ledState = game;
                                playerNumber = 1;
                                led = 11;
                                player1WinFlag = 0;
                                Channel_0_RC/=5;
                                digitalWrite(2, low);
                                digitalWrite(3, low);
                                digitalWrite(4, low);
                                digitalWrite(5, low);
                                digitalWrite(6, low);
                                digitalWrite(7, low);
                                digitalWrite(8, low);
                                digitalWrite(9, low);
                                digitalWrite(10, low);
                                digitalWrite(11, high);
                                Channel_0_CCR = 0x05;
                        }
                }
        }
}

if (TC_int == HIGH)
{
        TC_int = LOW;

        if(winCounter>0)
        {
                if(player1WinFlag)
                        digitalWrite(11, ((winCounter%2)?low:high));
                if(player2WinFlag)
                        digitalWrite(2, ((winCounter%2)?low:high));
```

```
        winCounter--;

        if(winCounter)
                return;
        else
                Channel_0_RC = 8192;
}

if(playerNumber == 1)
{
        if(led > 2)
                led--;
        else{
                if(player1State || n1_state)
                {
                        playerNumber = 2;
                        if(n1_state)
                                Channel_0_RC/=5;
                        if(player1State)
                                Channel_0_RC = 8192;
                        player1State = 0;
                        n1_state = 0;
                }
                else
                {
                        player2Score++;
                        player2WinFlag = 1;
                        winCounter = 6;
                        player1State = off;
                        player2State = off;
                        n1_state = off;
                        n2_state = off;
                        player1Counter = 0;
                        player2Counter = 0;
                        n1_counter = 0;
                        n2_counter = 0;
                        playerNumber = 0;
                        Channel_0_RC = 8192/5;
digitalWrite(2, low);
                        digitalWrite(3, low);
                        digitalWrite(4, low);
                        digitalWrite(5, low);
                        digitalWrite(6, low);
                        digitalWrite(7, low);
```

```
                        digitalWrite(8, low);
                        digitalWrite(9, low);
                        digitalWrite(10, low);
                        digitalWrite(11, low);

                        if(player2Score >= 3)
                        {
                                if(player2Score - player1Score >= 2)
                                {
                                        ledState = hold;
                                        winCounter = 0;
                                        player1WinFlag = 0;
                                        player2WinFlag = 0;
                                        led = 0;
                                        player1Score = 0;
                                        player2Score = 0;
digitalWrite(2, high);
                                        digitalWrite(3, high);
                                        digitalWrite(4, high);
                                        digitalWrite(5, high);
                                        digitalWrite(6, high);
                                        digitalWrite(7, high);
                                        digitalWrite(8, high);
                                        digitalWrite(9, high);
                                        digitalWrite(10, high);
                                        digitalWrite(11, high);
                                        Channel_0_RC = 8192;
                                        Channel_0_CCR = 0x02;
                                        return;
                                }
                        }
                }
        }
}
else if(playerNumber == 2)
{
        if(led < 11)
                led++;
        else{
                if(player2State || n2_state)
                {
                        playerNumber = 1;
                        if(n2_state)
                                Channel_0_RC/=5;
```

```
                    if(player2State)
                        Channel_0_RC = 8192;
                    player2State = 0;
                    n2_state = 0;
                }
                else
                {
                    player1Score++;
                    player1WinFlag = 1;
                    winCounter = 6;
                    player1State = off;
                    player2State = off;
                    n1_state = off;
                    n2_state = off;
                    player1Counter = 0;
                    player2Counter = 0;
                    n1_counter = 0;
                    n2_counter = 0;
                    playerNumber = 0;
                    Channel_0_RC = 8192/5;
digitalWrite(2, low);
                    digitalWrite(3, low);
                    digitalWrite(4, low);
                    digitalWrite(5, low);
                    digitalWrite(6, low);
                    digitalWrite(7, low);
                    digitalWrite(8, low);
                    digitalWrite(9, low);
                    digitalWrite(10, low);
                    digitalWrite(11, low);

                    if(player1Score >= 3)
                    {
                        if(player1Score - player2Score >= 2)
                        {
                            ledState = hold;
                            winCounter = 0;
                            player1WinFlag = 0;
                            player2WinFlag = 0;

                            led = 0;
                            player1Score = 0;
                            player2Score = 0;
                            digitalWrite(2, high);
```

```
                                                digitalWrite(3, high);
                                                digitalWrite(4, high);
                                                digitalWrite(5, high);
                                                digitalWrite(6, high);
                                                digitalWrite(7, high);
                                                digitalWrite(8, high);
                                                digitalWrite(9, high);
                                                digitalWrite(10, high);
                                                digitalWrite(11, high);
                                                Channel_0_CCR = 0x02;
                                                Channel_0_RC = 8192/5;
                                                return;
                                        }
                                }
                        }
                }
        }

if(player1Counter>0)
{
        player1Counter--;
        if(player1Counter<4)
                player1State = off;
}

if(n1_counter>0)
{
        n1_counter--;
        if(n1_counter<4)
                n1_state = off;
}

if(player2Counter>0)
{
        player2Counter--;
        if(player2Counter<4)
                player2State = off;
}

if(n2_counter>0)
{
        n2_counter--;
        if(n2_counter<4)
                n2_state = off;
```

```
                        }

                        if(winCounter==0)
                        {
                                digitalWrite(2, low);
                                digitalWrite(3, low);
                                digitalWrite(4, low);
                                digitalWrite(5, low);
                                digitalWrite(6, low);
                                digitalWrite(7, low);
                                digitalWrite(8, low);
                                digitalWrite(9, low);
                                digitalWrite(10, low);
                                digitalWrite(11, low);
                                digitalWrite(led, high);
                        }
                }
}
```

AT91

```
#include <sys/types.h>

#include <sys/stat.h>

#include <fcntl.h>

#include <sys/ioctl.h>

#include <unistd.h>        //At91 libraries

#include <sys/mman.h>

#include <stdio.h>

#include <stdlib.h>

#include <header.h>


#define off 0

#define on 1
```

```
#define BUT_IDLE 0
#define BUT_PRESSED 1          //Button constants

#define hold 0
#define game 1                 //Led_Constants

PIO* pioa = NULL;      //Δημιουργεία PIO για διαχείρηση της παράλληλης
AIC* aic = NULL;       //Δημιουργεία AIC για διαχείρηση της μονάδας διαχείρησης διακοπών
TC* tc = NULL;         //Δημιουργεία TC για διαχείρηση του ρολογιού

unsigned int player = 0;
unsigned int led = 0;
unsigned int led_state = hold;
unsigned int score1 = 0;
unsigned int score2 = 0;
unsigned int player1ButtonState = BUT_IDLE;
unsigned int btn_n1 = BUT_IDLE;
unsigned int btn_p2 = BUT_IDLE;
unsigned int btn_n2 = BUT_IDLE;
unsigned int player1State = off;
unsigned int p2_state = off;
unsigned int n1_state = off;
unsigned int n2_state = off;
unsigned int player1Counter = 0;
unsigned int p2_counter = 0;
unsigned int n1_counter = 0;
unsigned int n2_counter = 0;
unsigned int win = 0;
unsigned int p1_win = 0;
```

```
unsigned int p2_win = 0;

int main( int argc, const char * argv[] )
{
	//SETUP στο arduino
	unsigned int gen;	//Μεταβλήτη για την εκαθάριση των διακοπών

	STARTUP;	//Έναρξη του Συστήματος

	tc->Channel_0.RC = 8192;	//Διακοπή ανά 1 sec
	tc->Channel_0.CMR = 2084;
	tc->Channel_0.IDR = 0xFF;	//Αρχικοποιήση του Ρολόγιου
	tc->Channel_0.IER = 0x10;
	aic->FFER = (1<<PIOA_ID) | (1<<TC0_ID);
	aic->IECR = (1<<PIOA_ID) | (1<<TC0_ID);	//Αρχικοποίηση των διακοπών

	pioa->PUER = 0x3003; //Τα inputs είναι PULLUP
	pioa->ODR = 0x3003;//Τα bits 0,1 και 12,13 είναι input

	pioa->SODR = 0xFFC; //Δυναμικο high στα bits 2-11
	pioa->OER = 0xFFC; //Assign σε output τα bits 2-11

	gen = pioa->ISR;//Interupts cleanup στην παράλληλη
	pioa->PER = 0x3FFF; //Τα bits 0-13 σε καταχωρητές γενικού σκοπού
	gen = tc->Channel_0.SR;//Interupts cleanup στο ρολόϊ
	aic->ICCR = (1<<PIOA_ID)|(1<<TC0_ID);//Interupts cleanup στην μονάδα διαχείρησης διακοπών
	pioa->IER = 0x3003;//Interupt στην παράλληλη 0,1 και 12,13

	tc->Channel_0.CCR = 0x02; //Αρχικοποίηση του timer στην κατάσταση stop_count
```

```
	while((tmp=getchar()) != 'e')
	{ //Όσο δεν τερματίζει το πρόγραμμα με 'e'
		if (led_state == hold)
		{//Σε καθε επανάληψη, αν τα τα led ειναι σε κατάσταση hold
			pioa->SODR = 0x3FFF;//Αναψε τα ολα
		}
		else
		{//Αλλιώς
			pioa->CODR = 0x3FFF; //Καθαρισε τα
			pioa->SODR = hex(pow(2,led));//Και αναψε το ledακι στο οποίο βρίσκετε η μπάλα
		}
	}
}

// Handle Interupts
void FIQ_handler(void)
{
	// Parallel Interupt
	if( fiq & (1<<PIOA_ID) )
	{
		// Reset the interupt flag
		data_in = pioa->ISR;

		//Interupts cleanup στην μονάδα διαχείρησης διακοπών
		aic->ICCR = (1<<PIOA_ID);

		// Read inputs
```

```
data_in = pioa->PDSR;


// Ignore Interupt
if (win>0)
{
	break;
}
else if(!( data_in & 0x01 ))
{
	if(player1Counter == 0)
	{
		if(player1ButtonState == BUT_IDLE)
		{
			player1ButtonState = BUT_PRESSED;
			player1Counter = 6;
			player1State = on;

			if(led_state == hold || p1_win = 1)
			{
				led_state = game;
				player = 2;
				led = 2;
				p1_win = 0
				pioa->CODR = 0x3FFF;
				pioa->SODR = 0x04;
				tc->Channel_0.CCR = 0x05;
			}
		}
	}
```

```
        if(player1ButtonState == BUT_PRESSED)
        {
                player1ButtonState = BUT_IDLE;
        }
}
else if(!( data_in & 0x10 ))
{
        if(n1_counter == 0)
        {
                if(btn_n1 == BUT_IDLE)
                {
                        btn_n1 = BUT_PRESSED;
                        n1_counter = 6;
                        n1_state = on;

                        if(led_state == hold || p1_win = 1)
                        {
                                led_state = game;
                                player = 2;
                                led = 2;
                                p1_win = 0
                                pioa->CODR = 0x3FFF;
                                pioa->SODR = 0x04;
                                tc->Channel_0.CCR = 0x05;
                        }
                }
        }
```

```
            if(btn_n1 == BUT_PRESSED)
            {
                        btn_n1 = BUT_IDLE;
            }
}
else if(!( data_in & 0x1000 ))
{
            if(p2_counter == 0)
            {
                        if(btn_p2 == BUT_IDLE)
                        {
                                    btn_p2 = BUT_PRESSED;
                                    n1_counter = 6;
                                    n1_state = on;

                                    if(led_state == hold || p2_win = 1)
                                    {
                                                led_state = game;
                                                player = 1;
                                                led = 11;
                                                p2_win = 0
                                                pioa->CODR = 0x3FFF;
                                                pioa->SODR = 0x800;
                                                tc->Channel_0.CCR = 0x05;
                                    }
                        }
            }

            if(btn_p2 == BUT_PRESSED)
```

```
		{
			btn_p2 = BUT_IDLE;
		}
}
else if(!( data_in & 0x2000 ))
{
		if(n2_counter == 0)
		{
			if(btn_n2 == BUT_IDLE)
			{
				btn_n2 = BUT_PRESSED;
				n2_counter = 6;
				n2_state = on;

				if(led_state == hold || p2_win = 1)
				{
					led_state = game;
					player = 1;
					led = 11;
					p2_win = 0
					pioa->SODR = 0x800;
					tc->Channel_0.CCR = 0x05;
				}
			}
		}

		if(btn_n2 == BUT_PRESSED)
		{
			btn_n2 = BUT_IDLE;
```

```
                }
        }

}

if( fiq & (1<<TC0_ID) )
{
        data_out = tc->Channel_0.SR;
        aic->ICCR = (1<<TC0_ID);
        data_out = pioa->ODSR;

        if(win>0)
        {
                if(p1_win)
                {
                        pioa->SODR = 0x04;
                        pioa->CODR = 0x04;
                }

                if(p2_win)
                {
                        pioa->SODR = 0x800;
                        pioa->CODR = 0x800;
                }

                win--;

                if(!win)
                {
```

```
                    break;
            }
}

if(player = 1)
{
            if(led > 2)
            {
                    led--;
            }
            else
            {
                    if(player1State || n1_state)
                    {
                            player = 2;
                            if(n1_state)
                            {
                                    tc->Channel_0.RC/=5;
                            }

                            if(player1State)
                            {
                                    tc->Channel_0.RC = 8192;
                            }

                    }
                    else
                    {
                            score2++;
```

```
p2_win = 1;
win = 6;
player1State = off;
p2_state = off;
n1_state = off;
n2_state = off;
player1Counter = 0;
p2_counter = 0;
n1_counter = 0;
n2_counter = 0;
player = 0;
tc->Channel_0.RC = 8192;

if(score2 >= 3)
{
        if(score2 - score1 >= 2)
        {
                led_state = hold;
                win = 0;
                p1_win = 0;
                p2_win = 0;
                led = 0;
                score1 = 0;
                score2 = 0;
                pioa->SODR = 0x3FFF;
                tc->Channel_0.CCR = 0x02;
                break;   //Και τελείωσαε το πρόγραμμα εδω
        }
}
```

```
                }
        }
}
else
{
        if(led < 11)
        {
                led++;
        }
        else
        {
                if(p2_state || n2_state)
                {
                        player = 1;
                        if(n2_state)
                        {
                                tc->Channel_0.RC/=5;
                        }
                        if(p2_state)
                        {
                                tc->Channel_0.RC = 8192;
                        }
                }
                else
                {
                        score1++;
                        p1_win = 1;
                        win = 6;
                        player1State = off;
```

```
                    p2_state = off;
                    n1_state = off;
                    n2_state = off;
                    player1Counter = 0;
                    p2_counter = 0;
                    n1_counter = 0;
                    n2_counter = 0;
                    player = 0;
                    tc->Channel_0.RC = 8192;
                    if(score1 >= 3)
                    {
                        if(score1 - score2 >= 2)
                        {
                            led_state = hold;
                            win = 0;
                            p1_win = 0;
                            p2_win = 0;
                            led = 0;
                            score1 = 0;
                            score2 = 0;
                            pioa->SODR = 0x3FFF;
                            tc->Channel_0.CCR = 0x02;
                            break;
                        }
                    }
                }
            }
        }
```

```
if(player1Counter>0)
{
	player1Counter--;
	if(player1Counter<4)
		player1State = off;
}

if(n1_counter>0)
{
	n1_counter--;
	if(n1_counter<4)
		n1_state = off;
}

if(p2_counter>0)
{
	p2_counter--;
	if(p2_counter<4)
		p2_state = off;
}



if(n2_counter>0)
{
	n2_counter--;
	if(n2_counter<4)
		n2_state = off;
}
```

```
                pioa->CODR = 0x3FFF;
                pioa->SODR = hex(pow(2,led));
        }
}
```