Εργαστήριο Προηγμένοι Μικροεπεξεργαστές

Αναφορά 5ης Εργαστηριακής Άσκησης

# **Ομάδα Β3**

|  |  |
| --- | --- |
| Φοιτητές | ΑΜ |
|  |  |
| Μαγκλάρας Θεμιστοκλής Παναγιώτης | 1047182 |

**Εισαγωγή:**

Σκοπός της 5ης εργαστηριακής άσκησης είναι να φτιάξουμε ένα παιχνίδι τύπου pass-the-ball, στο οποίο συμμετέχουν δύο παίκτες που έχουν στη διάθεση τους ο καθένας δύο κουμπιά. Κάθε κουμπί έχει διαφορετική λειτουργία. Το 1ο κουμπί χρησιμοποιείται για πάσα ή απόκρουση της μπάλας ενώ το 2ο κουμπί χρησιμοποιείται για δυνατό σουτ. Η μπάλα αναπαρίσταται με το άναμμα του εκάστοτε Led. Όταν η διαφορά του σκορ είναι ίση με το 2, τότε κερδίζει ο παίκτης με τα περισσότερα «γκολ».

**Πείραμα σε AT91:**

#define LOW 0

#define HIGH 1

#define OUTPUT 0

#define INPUT 1

#define INPUT\_PULLUP 2

void digitalWrite(int pin, int value)

{

switch (value)

{

case LOW:

pioa->CODR = pioa->CODR | (1 << pin)

break;

case HIGH:

pioa->SODR = pioa->SODR | (1 << pin)

break;

}

}

int digitalRead(int pin)

{

return pioa->PDSR & (1 << pin);

}

int pinMode(int pin, int mode)

{

pioa->PER = pioa->PER | (1 << pin);

switch (mode)

{

case OUTPUT:

pioa->OER = pioa->OER | (1 << pin);

pioa->CODR = pioa->CODR | (1 << pin);

break;

case INPUT:

pioa->ODR = pioa->ODR | (1 << pin);

break;

case INPUT\_PULLUP:

pioa->ODR = pioa->ODR | (1 << pin);

pioa->PUER = pioa->PUER | (1 << pin);

break;

}

}

void setup()

{

for(int j = 11; j>1; --j) pinMode(j, OUTPUT);

pinMode(0, INPUT\_PULLUP);

pinMode(1, INPUT\_PULLUP);

pinMode(12,INPUT\_PULLUP);

pinMode(13,INPUT\_PULLUP);

}

bool button\_press[4] = {0,0,0,0};

bool button\_press\_now[4] = {0,0,0,0};

void polling()

{

static bool button\_press\_before[4] = {0,0,0,0};

button\_press\_before[0] = button\_press[0];

button\_press\_before[1] = button\_press[1];

button\_press\_before[2] = button\_press[2];

button\_press\_before[3] = button\_press[3];

button\_press[0] = !digitalRead(0);

button\_press[1] = !digitalRead(1);

button\_press[2] = !digitalRead(12);

button\_press[3] = digitalRead(13);

button\_press\_now[0] = !button\_press\_before[0] && button\_press[0]? 1 : 0;

button\_press\_now[1] = !button\_press\_before[1] && button\_press[1]? 1 : 0;

button\_press\_now[2] = !button\_press\_before[2] && button\_press[2]? 1 : 0;

button\_press\_now[3] = !button\_press\_before[3] && button\_press[3]? 1 : 0;

}

static short difference\_in\_goals = 0; //Αρχικοποίηση διαφοράς γκολ.

static unsigned short const blink\_speed = 10;

static unsigned short const blink\_speed2 = 2;

static unsigned short ball\_frequency\_basic = 1\*blink\_speed; //Συχνότητα μπάλας.

static bool game\_start = true;

void ball\_drawing (float ball\_position)

{

digitalWrite(round(ball\_position)+2,1);

}

void clear\_board\_game()

{

for(int j = 11; j>1; --j)

{

digitalWrite(j, 0);

}

}

void football\_game()

{

static float ball\_position = 0.0f;

static float ball\_velocity = 0.0f;

static int t1 = 0;

static bool used\_left\_nitro = false; //Αρχικοποιούμε τη χρήση του αριστερού nitro με 0.

static bool used\_right\_nitro = false; //Αρχικοποιούμε τη χρήση του δεξιού nitro με 0.

if (game\_start && 0)

{

if (button\_press\_now[0])

{

ball\_velocity = ball\_frequency\_basic;

game\_start = 0;

}

else if (button\_press\_now[1])

{

ball\_velocity = ball\_frequency\_basic \* 5;

game\_start = 0;

}

}

ball\_position = ball\_position + ball\_velocity\*((millis()-t1)/1000.0f);

t1 = millis();

clear\_board\_game();

if(ball\_position <= 9 && ball\_position >= 0)

{

ball\_drawing(ball\_position);

if(ball\_position <= 4.5)

{

used\_left\_nitro = false;

if(button\_press\_now[0] && !used\_right\_nitro)

{

ball\_velocity = abs(ball\_velocity) >= .4 ? abs(ball\_frequency\_basic\*5\*ball\_velocity) : ball\_frequency\_basic\*5;

used\_right\_nitro = true;

}

if(button\_press\_now[1])

{

ball\_velocity = ball\_frequency\_basic;

}

}

else

{

used\_right\_nitro = false;

if(button\_press\_now[3] && !used\_left\_nitro)

{

ball\_velocity = abs(ball\_velocity) >= .4 ? -abs(ball\_frequency\_basic\*5\*ball\_velocity) : -(float)ball\_frequency\_basic\*5;

used\_left\_nitro = false;

}

if(button\_press\_now[2])

{

ball\_velocity = -(float)ball\_frequency\_basic;

}

}

}

else

{

{

int sc = millis();

int tempTime;

while((tempTime = (millis()-sc))/1000.0f <= 1.2f/blink\_speed2)

{

(ball\_position>9) ? digitalWrite(11,tempTime%(200/blink\_speed2) > 100/blink\_speed2) : digitalWrite(2,tempTime%(200/blink\_speed2)>100/blink\_speed2);

}

}

used\_left\_nitro = used\_right\_nitro = false;

ball\_velocity = 0;

if(ball\_position > 9)

{

ball\_position = 9;

difference\_in\_goals = difference\_in\_goals + 1;

if(difference\_in\_goals == 2)

{

difference\_in\_goals = 0;

int sc = millis();

}

}

else

{

ball\_position = 0;

difference\_in\_goals = difference\_in\_goals - 1;

if(difference\_in\_goals == -2)

{

difference\_in\_goals = 0;

int sc = millis();

}

}

}

}

void loop()

{

polling();

football\_game();

}

void FIQ\_handler()

{

unsigned int data\_in;

if (fiq & (1 << PIOA\_ID))

{

data\_in = pioa->ISR;

aic->ICCR = (1 << PIOA\_ID);

data\_in = pioa->PDSR;

if (data\_in & 0x02)

{

upd\_button();

}

}

}

int main()

{

STARTUP;

setup();

while (getchar() != 'e')

{

loop();

}

aic->IDCR = (1 << PIOA\_ID);

CLEANUP

return 0;

}

***ΣΗΜΕΙΩΣΗ***

Η συνάρτηση *upd\_button()* είναι ίδια με αυτή που υλοποιήσαμε στην εργαστηριακή άσκηση 3.

**Πείραμα σε Tinkercad:**

Παρακάτω παρατίθεται η εικόνα του κυκλώματος που υλοποιήσαμε:

Εικόνα που περιέχει κύκλωμα

Περιγραφή που δημιουργήθηκε αυτόματα

Στη συνέχεια παρατίθεται ο σύνδεσμος του κυκλώματος:

<https://www.tinkercad.com/things/8rqXhQKk6a3-erg-askhsh-5/editel?sharecode=eKkpfzvtok1Df-djyrgvGWOBwwBgdrBgOo71JxXq2Xg>

Σημειώνεται οτι τα δύο ακριανά κουμπιά είναι τα κουμπιά της βολίδας ενώ τα δύο μέσα κουμπιά είναι τα κουμπιά της πάσας και της απόκρουσης για τον κάθε παίκτη αντίστοιχα.

Τέλος, παρατίθεται ο κώδικας σε Tinkercad που υλοποιήσαμε:

void setup()

{

for(int j = 11; j>1; --j) pinMode(j, OUTPUT);

pinMode(0, INPUT\_PULLUP);

pinMode(1, INPUT\_PULLUP);

pinMode(12,INPUT\_PULLUP);

pinMode(13,INPUT\_PULLUP);

}

bool button\_press[4] = {0,0,0,0};

bool button\_press\_now[4] = {0,0,0,0};

void polling()

{

static bool button\_press\_before[4] = {0,0,0,0};

button\_press\_before[0] = button\_press[0];

button\_press\_before[1] = button\_press[1];

button\_press\_before[2] = button\_press[2];

button\_press\_before[3] = button\_press[3];

button\_press[0] = !digitalRead(0);

button\_press[1] = !digitalRead(1);

button\_press[2] = !digitalRead(12);

button\_press[3] = digitalRead(13);

button\_press\_now[0] = !button\_press\_before[0] && button\_press[0]? 1 : 0;

button\_press\_now[1] = !button\_press\_before[1] && button\_press[1]? 1 : 0;

button\_press\_now[2] = !button\_press\_before[2] && button\_press[2]? 1 : 0;

button\_press\_now[3] = !button\_press\_before[3] && button\_press[3]? 1 : 0;

}

static short difference\_in\_goals = 0; //Αρχικοποίηση διαφοράς γκολ.

static unsigned short const blink\_speed = 10;

static unsigned short const blink\_speed2 = 2;

static unsigned short ball\_frequency\_basic = 1\*blink\_speed; //Συχνότητα μπάλας.

void ball\_drawing (float ball\_position) /\*Ανάβει το πιο κοντινό Led στο οποίο βρίσκεται η μπάλα.\*/

{

digitalWrite(round(ball\_position)+2,1);

}

void clear\_board\_game() //Σβήνει όλα τα Led.

{

for(int j = 11; j>1; --j) digitalWrite(j, 0);

}

void football\_game()

{

static float ball\_position = 0.0f;

static float ball\_velocity = 0.0f;

static int t1 = 0;

static bool used\_left\_nitro = false; //Αρχικοποιούμε τη χρήση του αριστερού nitro με 0.

static bool used\_right\_nitro = false; //Αρχικοποιούμε τη χρήση του δεξιού nitro με 0.

ball\_position = ball\_position + ball\_velocity\*((millis()-t1)/1000.0f);

t1 = millis();

clear\_board\_game();

if(ball\_position <= 9 && ball\_position >= 0)

{

ball\_drawing(ball\_position);

if(ball\_position <= 4.5)

{

used\_left\_nitro = false;

if(button\_press\_now[0] && !used\_right\_nitro)

{

ball\_velocity = abs(ball\_velocity) >= .4 ? abs(ball\_frequency\_basic\*5\*ball\_velocity) : ball\_frequency\_basic\*5;

used\_right\_nitro = true;

}

if(button\_press\_now[1])

{

ball\_velocity = ball\_frequency\_basic;

}

}

else

{

used\_right\_nitro = false;

if(button\_press\_now[3] && !used\_left\_nitro)

{

ball\_velocity = abs(ball\_velocity) >= .4 ? -abs(ball\_frequency\_basic\*5\*ball\_velocity) : -(float)ball\_frequency\_basic\*5;

used\_left\_nitro = false;

}

if(button\_press\_now[2])

{

ball\_velocity = -(float)ball\_frequency\_basic;

}

}

}

else

{

{

int sc = millis();

int tempTime;

while((tempTime = (millis()-sc))/1000.0f <= 1.2f/blink\_speed2)

{

(ball\_position>9) ? digitalWrite(11,tempTime%(200/blink\_speed2) > 100/blink\_speed2) : digitalWrite(2,tempTime%(200/blink\_speed2)>100/blink\_speed2);

}

}

used\_left\_nitro = used\_right\_nitro = false;

ball\_velocity = 0;

if(ball\_position > 9)

{

ball\_position = 9;

difference\_in\_goals = difference\_in\_goals + 1;

if(difference\_in\_goals == 2)

{

difference\_in\_goals = 0;

int sc = millis();

}

}

else

{

ball\_position = 0;

difference\_in\_goals = difference\_in\_goals - 1;

if(difference\_in\_goals == -2)

{

difference\_in\_goals = 0;

int sc = millis();

}

}

}

}

void loop()

{

polling();

football\_game();

}

**Διόρθωση:**

Το pin 13 του Arduino στο Tinkercad διαθέτει συνδεσμολογία pull-down αντίστασης, κάτι το οποίο δεν είχαμε αντιληφθεί αρχικά. Αφού το συνειδητοποιήσαμε, το συνδέσαμε σε πηγή τάσης.

**Αποτελέσματα - Συμπεράσματα:**

Παρατηρούμε ότι η εκκίνηση και συνέχιση του παιχνιδιού πραγματοποιείται ομαλά. Οι δύο παίκτες έχοντας τη δυνατότητα της βολίδας ή της πάσας, ανταλλάσσουν μεταξύ τους τη μπάλα η ταχύτητα της οποίας φαίνεται και από την αλλαγή της συχνότητας ανάμματος των Led. Τέλος, χάρις τη συγκεκριμένη άσκηση μαθαίνουμε πως να αλλάζουμε το output των Leds ανάλογα με το αν η μπάλα βρεθεί εντός ή εκτός ορίων (δηλαδή αν έχει μπει γκολ).