

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | |
|  | | 4 ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ | | |  | |
|  |  | | | | |  |
|  | | | **Εργαστήριο Προηγμένοι Μικροεπεξεργαστές** |  | | |
|  | | | ΙΩΑΝΝΗΣ ΧΡΙΣΤΟΔΟΥΛΑΚΟΣ 1062664ΙΩΑΝΝΗΣ ΧΑΡΑΛΑΜΠΟΥΣ1059685 |  | | |

**ΑΣΚΗΣΗ 1**

**1α) Εκτελέστε το πρόγραμμα στον εξομοιωτή και εξηγήστε τί συμβαίνει μετά την εκτέλεση του κώδικα**

#include <avr/io.h>

#include <stdio.h>

#include <stdlib.h>

#include <avr/io.h>

#include <avr/interrupt.h>

int interrWrite = 0; //timi gia to switch5

int interrRead = 0; //timi gia to switch6

int Ram [32] ; //pinakas poy xrhsimopoioyme os mnhmh ram

int counter = 0;

int address = 0 ; //metablhth gia prosorini apothikeusi ths tyxaias timhs

int value =0;

int main(){

//clock for write

PORTD.DIR |= PIN1\_bm; //PIN is output

//prescaler=1024

TCA0.SINGLE.CTRLA= TCA\_SINGLE\_CLKSEL\_DIV1024\_gc;

TCA0.SINGLE.PER = 100; //select the resolution

TCA0.SINGLE.CMP0 = 80; //select the duty cycle palmos falling edge gia write

TCA0.SINGLE.CMP1 = 40; //select the duty cycle palmos falling edge gia read

//select Single\_Slope\_PWM

TCA0.SINGLE.CTRLB |= TCA\_SINGLE\_WGMODE\_SINGLESLOPE\_gc;

//enable interrupt Overflow

TCA0.SINGLE.INTCTRL = TCA\_SINGLE\_OVF\_bm;

//enable interrupt COMP0

TCA0.SINGLE.INTCTRL |= TCA\_SINGLE\_CMP0\_bm;

TCA0.SINGLE.INTCTRL |= TCA\_SINGLE\_CMP1\_bm;

TCA0.SINGLE.CTRLA |= TCA\_SINGLE\_ENABLE\_bm; //EnablTCA\_Se

PORTF.PIN5CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;

PORTF.PIN6CTRL |= PORT\_PULLUPEN\_bm | PORT\_ISC\_BOTHEDGES\_gc;

sei();

while (counter< 32){

address = rand() % 16; //dhmiourgeia tyxaias 4bit timhs

}

}

ISR(TCA0\_OVF\_vect){ //synarthsh pou energopoieitai sto rising edge

//clear the interrupt flag

int intflags = TCA0.SINGLE.INTFLAGS;

TCA0.SINGLE.INTFLAGS = intflags;

PORTD.OUT |= PIN1\_bm; //PIN is off

if(interrWrite == 1){

Ram[counter] = address; //apothikeusi ths tyxaias metablhths

counter++; //aykshsh ths theshs mnhmhs ths ram

}

interrWrite = 0; //apenergopoihsh tou interrupt

}

ISR(TCA0\_CMP0\_vect){ //synarthsh pou energopoieitai sto falling edge ths eggrafhs

//clear the interrupt flag

int intflags = TCA0.SINGLE.INTFLAGS;

TCA0.SINGLE.INTFLAGS = intflags;

PORTD.OUTCLR |= PIN4\_bm; //PIN is off

if(interrWrite == 1){

Ram[counter] = address; //apothikeusi ths tyxaias metablhths

counter++; //aykshsh ths theshs mnhmhs ths ram

}

interrWrite = 0; //apenergopoihsh tou interrupt

}

ISR(TCA0\_CMP1\_vect){ //synarthsh pou energopoieitai sto falling edge ths anagnosis

//clear the interrupt flag

int intflags = TCA0.SINGLE.INTFLAGS;

TCA0.SINGLE.INTFLAGS = intflags;

PORTD.OUTCLR |= PIN2\_bm; //PIN is off

if(interrRead == 1){

value = Ram[counter-1]; //anagnosh ths teleutaias eggrafhs ths mnhmhs

PORTD.OUT = value; //energopoihsh twn pin analoga me thn timh poy exei eggrafei sthn mnhmh

}

interrRead =0; //apenergopoihsh tou interrupt

}

ISR(PORTF\_PORT\_vect){ //synarthsh pou energopoieitai me to interrupt gia ta switches

//clear the interrupt flag

int intflags = PORTF.INTFLAGS;

PORTF.INTFLAGS=intflags;

if(intflags == 0x20 ) //energopoieithike to pin5 kai to interrupt eggrafhs

interrWrite=1;

if(intflags == 0x40 ) //energopoieithike to pin6 kai to interrupt anagnosis

interrRead = 1;

}

Αρχικά χρησιμοπιήσαμε τον κώδικα του παραδείγματος για το ρολόι μας και προσθέσαμε το CMP1 για να έχουμε ένα διαφορετικό παλμό με διαφορετικό falling edge όταν θέλουμε να εκτελέστει το Read.Στη συνέχεια φτιάξαμε ένα πίνακα που θα λειτουργεί ως Ram.Μετά προσθέσαμε τα switches interrupts τα οποία είναι το pin5 του portf για το Write και το pin6 του portf για το Read .Μέσα στην ISR ρουτίνα του έχουμε τον έλεγχο για το ποιο pin ενεργοποιήθηκε και την ενεργοποίηση της αντιστοιχης ISR ρουτίνας του ρολογιού μας .Για το Write χρησιμοποιούμε και το rising edge και το falling edge του ρολογιού ενώ για το Read χρησιμοποιούμε μόνο την ρουτίνα του falling edge με διαφορετικό παλμό ρολογιού.Μέσα στις αντίστοιχες ISR ρουτίνες γίνεται η εγγραφή της τυχαίας λέξης ή η ανάγνωση της τελευταίας αποθηκευμένης λέξης.