**Εργαστήριο Προηγμένοι Μικροεπεξεργαστές**

**Εργαστηριακές Ασκήσεις-Σαριδάκης Γεώργιος 1072478**

1. **Για το πρώτο ερώτημα:**

***Κώδικας:***

#include <avr/io.h>

#include <avr/interrupt.h>

#include <util/delay.h>

#include <stdbool.h>

bool led0\_state = false;

bool led1\_state = false;

int main() {

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

sei(); //enable interrupts

while(1){

// Set up TCB0 for PWM generation with a 2ms period and 40% duty cycle

TCB0.CCMP = F\_CPU \* 0.002; // Set PWM period to 2ms (assuming F\_CPU is the CPU frequency in Hz)

TCB0.CTRLB = TCB\_CNTMODE\_PWM8\_gc | TCB\_CCMPEN\_bm; // Set up TCB0 for PWM generation with 8-bit resolution

TCB0.CCMPH = TCB0.CCMP \* 0.4; // Set duty cycle to 40%

PORTD.DIR = PIN0\_bm; //PIN0 is output

TCB0.INTCTRL = TCB\_CAPT\_bm;

// Set up TCB1 for PWM generation with a 1ms period and 50% duty cycle

TCB1.CCMP = F\_CPU \* 0.001; // Set PWM period to 1ms (assuming F\_CPU is the CPU frequency in Hz)

TCB1.CTRLB = TCB\_CNTMODE\_PWM8\_gc | TCB\_CCMPEN\_bm; // Set up TCB1 for PWM generation with 8-bit resolution

TCB1.CCMPH = TCB1.CCMP / 2; // Set duty cycle to 50%

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

TCB1.INTCTRL = TCB\_CAPT\_bm; // enable interrupt on capture/compare

// Enable global interrupts

sei();

}

}

ISR(TCB0\_INT\_vect) {

// Check the edge type of the interrupt

if (TCB0.INTFLAGS & TCB\_CAPT\_bm) {

// High edge

led0\_state = true;

} else {

// Low edge

led0\_state = false;

}

// Set the state of LED1 based on the current edge

if (led0\_state) {

PORTD.OUTCLR = PIN0\_bm; //PIN0 is on

} else {

PORTD.OUT = PIN0\_bm; //PIN0 is off

}

}

ISR(TCB1\_INT\_vect) {

// Check the edge type of the interrupt

if (TCB1.INTFLAGS & TCB\_CAPT\_bm) {

// High edge

led1\_state = true;

} else {

// Low edge

led1\_state = false;

}

// Set the state of LED2 based on the current edge

if (led1\_state) {

PORTD.OUTCLR = PIN1\_bm; //PIN1 is on

} else {

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

}

}

ISR(PORTF\_PORT\_vect){

//clear the interrupt flag

int y = PORTF.INTFLAGS;

PORTF.INTFLAGS=y;

TCB0.CTRLA = TCB\_ENABLE\_bm; // Enable TCB0

TCB1.CTRLA = TCB\_ENABLE\_bm; // Enable TCB1

}

***Επεξήγηση:*** Για το πρώτο ερώτημα της εργασίας αρχικά στην main αρχικοποιούμε το PIN5 του PORTF για να λειτουργεί σαν διακόπτης και ενεργοποιούμε τις διακοπές. Μόλις πατηθεί ο διακόπτης, δηλαδή πατήσουμε το PIN5 του PORTF (χειροκίνητα) εκτελείται η ISR του PORTF στην οποία ενεργοποιούνται οι δύο διαφορετικοί PWM που θα χρησιμοποιήσουμε TCB0 και TCB1 οι οποίοι έχουν δηλωθεί και αρχικοποιηθεί στην Main με τις περιόδους και duty cycle που ζητούνται, όπου όμως χωρίς τις enable εντολές της ρουτίνας δεν θα ξεκινούσαν την λειτουργία τους. Οι TCB ξεκινούν τους παλμούς και όταν ένας από αυτούς βρίσκεται σε ανερχόμενη κορυφή η ροή μεταφέρεται στην εκάστοτε ISR ρουτίνα όπου η μεταβλητή led0\_state και led1\_state θα τίθεται σε true ή false αντίστοιχα και θα ανάβει ή όχι το αντίστοιχο LED0 ή LED1.

***Διάγραμμα ροής:*** ***Εικόνα που περιέχει κείμενο, διάγραμμα, στιγμιότυπο οθόνης, ορθογώνιο παραλληλόγραμμο

Περιγραφή που δημιουργήθηκε αυτόματα***

**2.Για τα δύο επόμενα ερωτήματα:**

***Κώδικας:***

***Επεξήγηση:*** Όπως και στο προηγούμενο ερώτημα στην main αρχικοποιούμε και ενεργοποιούμε το switch με το PORTF του PIN5, αρχικοποιύμε τους TCB0 και TCB1 ως PWM και ακόμη αρχικοποιούμε τον ADC χωρίς όμως να ενεργοποιείται κανείς από τους τρεις. Αυτό θα γίνει όταν πατηθεί το switch όπου τώρα στην ISR ρουτίνα του PORTF αρχικά θα υπάρχει if συνθήκη που θα ελέγχει αν η διακοπή προήλθε από switch pin και αν είναι η πρώτη, δεύτερη η τρίτη φορά που πατιέται το switch μέσω μεταβλητής switch\_count που αυξάνεται κατά 1 κάθε φορά που εκτελείται η ρουτίνα. Όταν πατηθεί τώρα για πρώτη φορά το switch ενεργοποιούνται οι TCB0,TCB1 και ADC. Οι ρουτίνες των TCB0,TCB1 παραμένουν ίδιες για την ενεργοποίηση των LED και όταν ο ADC μετρήσει τιμή μικρότερη από ένα όριο 10 η ροή μεταβαίνει στην ISR του όπου και για σταματήσει τη λειτουργία του ο ανεμιστήρας σταματάνε να λειτουργούν οι PWMs ,σβήνουν τα LED0,LED1 αν είναι αναμμένα και ανάβει το LED2.

Με το δεύτερο πάτημα του switch μέσα από την ISR του PORTF αρχικά θα σβήσει το LED2,θα αρχικοποιηθεί ο TCB0 με διπλάσιο κύκλο λειτουργίας,θα γίνει ο ίδιος enable, και θα γίνει και ο TCB1 enable με τις προηγούμενες αρχικές τιμές.

Με το τρίτο πάτημα του switch τώρα ο ανεμιστήρας θα απενεργοποιηθεί και οι TCB0,TCB1,ADC θα σταματήσουν την λειτουργία τους στον else if (switch\_count == 3) βρόχο.