

MICROCHIP STUDIO

AVR PROGRAMMING

Κυριακή Τσαντικίδου
e-mail: k DOT tsantikidou AT upatras DOT gr

Διάρθρωση Παρουσίασης

1. Πάλμοευρικός Διαμορφωτής (PWM)
2. Τρίτη Εργαστηριακή Άσκηση
3. Βιβλιογραφία

Παλμοευρικός Διαμορφωτής (PWM)

Ενότητα 1

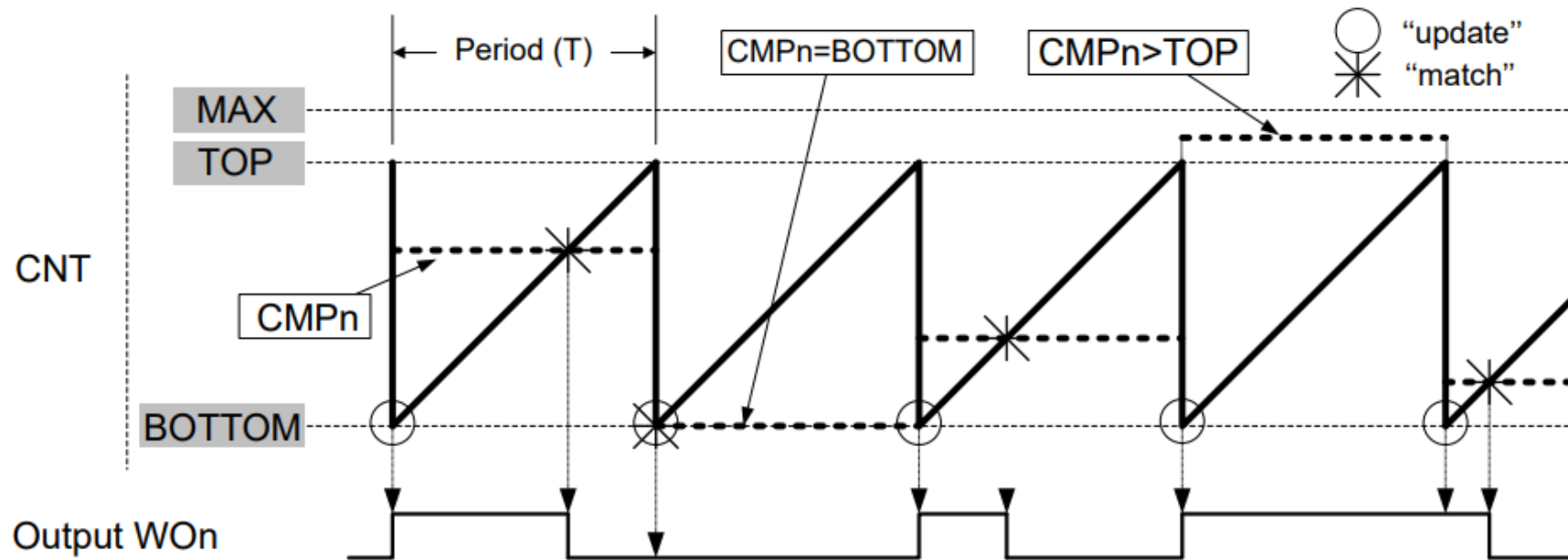
Ορισμός και Τρόποι Λειτουργίας

- ▶ Ο Παλμοευρικός Διαμορφωτής δημιουργεί κυματομορφές (με την βοήθεια των χρονιστών), οι οποίες μπορούν να χρησιμοποιηθούν ως έξοδοι στα PINS των περιφερειακών.
- ▶ Χρησιμοποιείται με πολλά περιφερειακά και εφαρμογές:
 - ▶ Audio
 - ▶ Ρύθμιση έντασης LED
 - ▶ Analog signal generation
 - ▶ Ρύθμιση servos, DC motors, κτλ.

Ενεργοποίηση Single-Slope PWM Generator

- ▶ Ορισμός **Prescaler** Χρονιστή →
`TCA0.SINGLE.CTRLA = TCA_SINGLE_CLKSEL_DIV1024_gc.`
- ▶ Ορισμός **περιόδου** (ορίζεται η **μέγιστη τιμή TOP** μέχρι την οποία θα μετρήσει ο μετρητής) → `TCA0.SINGLE.PER = value.`
- ▶ Ορισμός **κύκλου λειτουργίας** του παλμού (**duty cycle**) →
`TCA0.SINGLE.CMP0 = value.`
- ▶ Επιλογή **δημιουργίας κυματομορφών** (Waveform Generation Mode) →
`TCA0.SINGLE.CTRLB = TCA_SINGLE_WGMODE_SINGLESLOPE_gc.`
- ▶ **Ενεργοποίηση** χρονιστή →
`TCA0.SINGLE.CTRLA |= TCA_SINGLE_ENABLE_bm.`

Δημιουργία Κυματομορφής (Waveform Generation - Single Slope)



Ορισμός Περιόδου:
$$f_{PWM_SS} = \frac{f_{CLK_PER}}{N(PER + 1)}$$

Διακοπές (Interrupts)

- ▶ Όταν ο παλμός που δημιουργείται ανεβαίνει στην υψηλή στάθμη και όταν κατεβαίνει στη χαμηλή στάθμη, πρέπει να ενεργοποιούνται οι κατάλληλες διακοπές (interrupts):
 - ▶ Το **Overflow Interrupt**, δηλαδή εκτελείται διακοπή όταν ο χρονιστής (timer) είναι ίσος με την BOTTOM τιμή (ανεβαίνει στην υψηλή στάθμη) → `TCA0.SINGLE.INTCTRL = TCA_SINGLE_OVF_bm`.
 - ▶ Το **CMPx Interrupt**, δηλαδή εκτελείται διακοπή όταν ο χρονιστής (timer) είναι ίσος με την τιμή του καταχωρητή CMPx (πηγαίνουμε στη χαμηλή στάθμη) → `TCA0.SINGLE.INTCTRL = TCA_SINGLE_CMP0_bm`.

Λειτουργία Παραδείγματος

- ▶ Στο παράδειγμα αυτό ενεργοποιείται ο PWM ώστε να δημιουργηθεί ένας παλμός, που θα λειτουργεί σαν ρολόι.
- ▶ Με βάση τον παλμό αυτόν θα αναβοσβήνει ένα LED, το οποίο θα ενεργοποιείται όταν ο παλμός ανεβαίνει στην υψηλή στάθμη (rising edge) και θα απενεργοποιείται όταν πέφτει στη χαμηλή στάθμη (falling edge).

Υλοποίηση Πέμπτου Παραδείγματος: PWM

```
#include <avr/io.h>
#include <avr/interrupt.h>

int main(){
    PORTD.DIR = PIN1_bm; //PIN is output
    //prescaler=1024
    TCA0.SINGLE.CTRLA=TCA_SINGLE_CLKSEL_DIV1024_gc;
    TCA0.SINGLE.PER = 254; //select the resolution
    TCA0.SINGLE.CMP0 = 127; //select the duty cycle
    //select Single_Slope_PWM
    TCA0.SINGLE.CTRLB = TCA_SINGLE_WGMODE_SINGLESLOPE_gc;
    //enable interrupt Overflow
    TCA0.SINGLE.INTCTRL = TCA_SINGLE_OVF_bm;
    //enable interrupt CMP0
    TCA0.SINGLE.INTCTRL |= TCA_SINGLE_CMP0_bm;
    TCA0.SINGLE.CTRLA |= TCA_SINGLE_ENABLE_bm; //Enable
    sei();
    while (1){
        ;
    }
}
```

```
ISR(TCA0_OVF_vect){
    //clear the interrupt flag
    int intflags = TCA0.SINGLE.INTFLAGS;
    TCA0.SINGLE.INTFLAGS = intflags;
    PORTD.OUT = PIN1_bm; //PIN is off
}
```

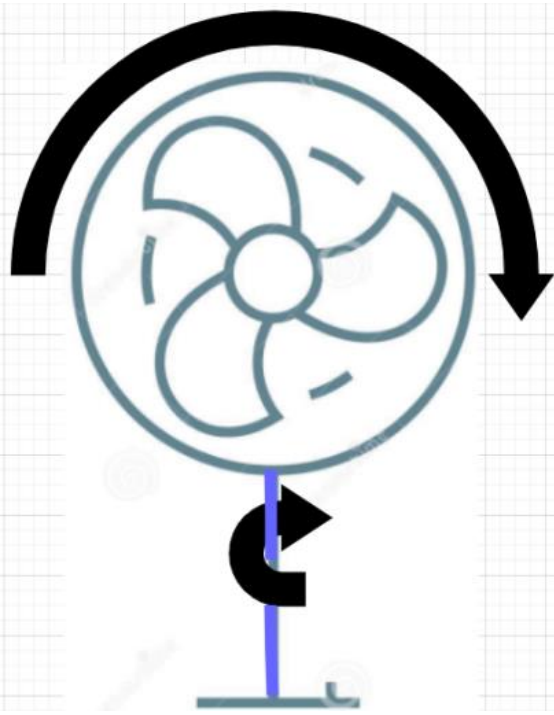
```
ISR(TCA0_CMP0_vect){
    //clear the interrupt flag
    int intflags = TCA0.SINGLE.INTFLAGS;
    TCA0.SINGLE.INTFLAGS = intflags;
    PORTD.OUTCLR = PIN1_bm; //PIN is on
}
```

- ▶ Πάρτε τον κώδικα και κάντε Simulation στο Microchip Studio.
- ▶ Παρατηρήστε πως λειτουργεί ο timer TCA.

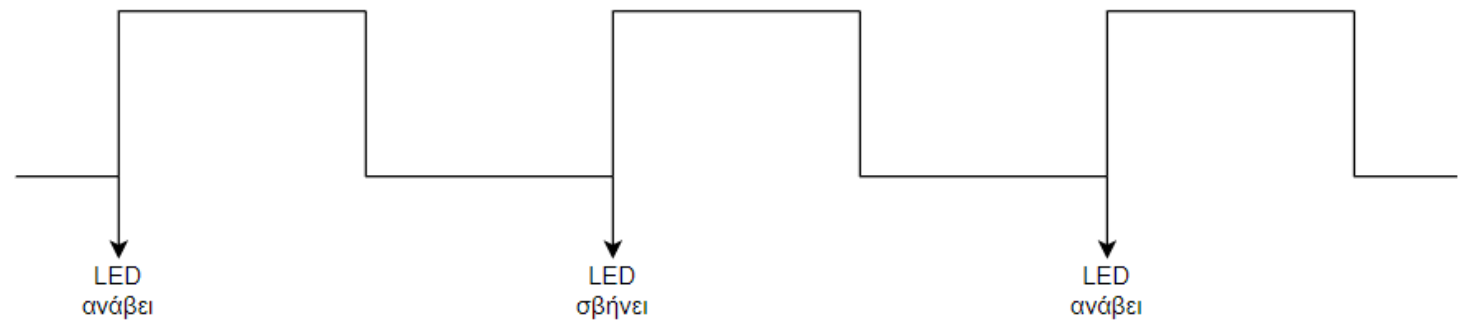
Τρίτη Εργαστηριακή Άσκηση

Ενότητα 2

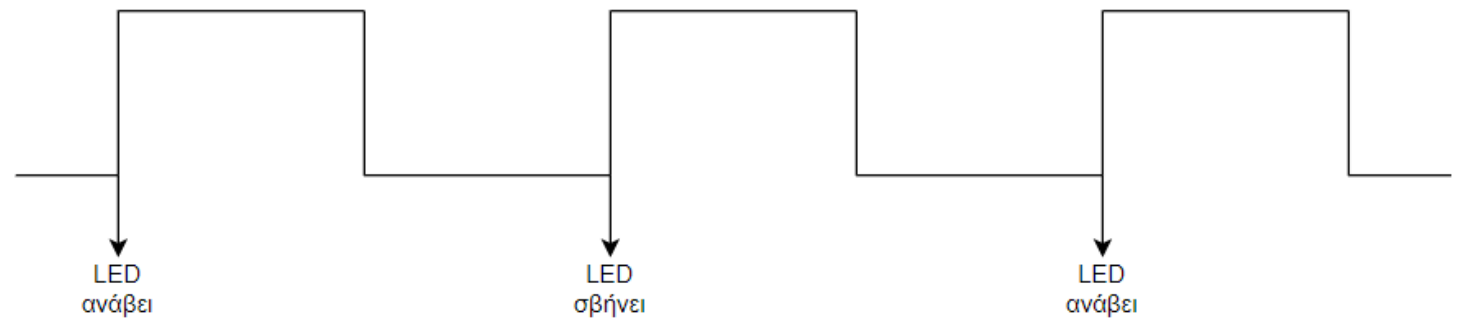
Προσομοίωση κίνησης ανεμιστήρα



Παλμός 1 → Κυκλική κίνηση των Λεπίδων - Ενεργοποίηση του LED0



Παλμός 2 → Κυκλική κίνηση της Βάσης - Ενεργοποίηση του LED1



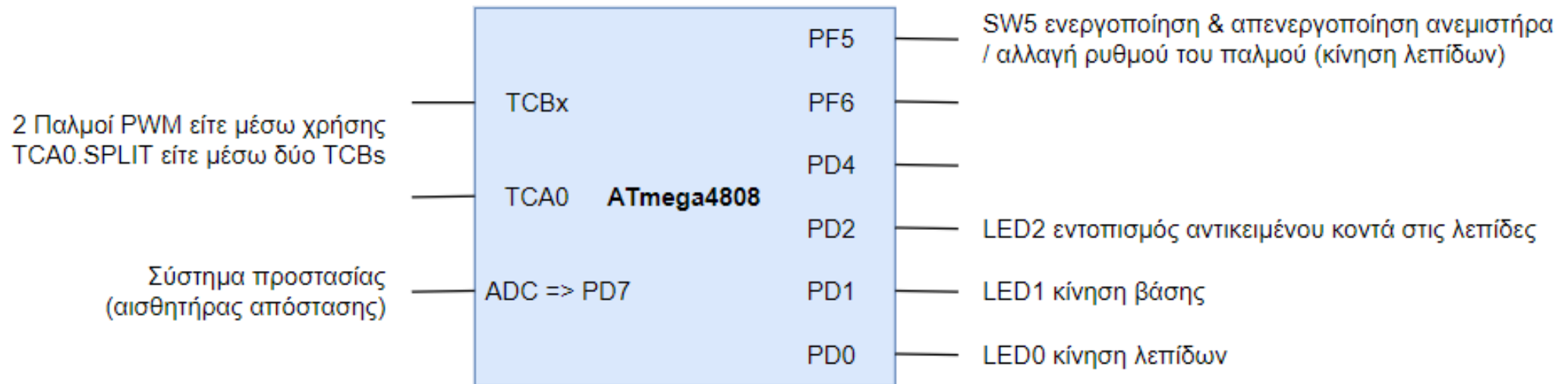
Παραδοχές (1/2)

- ▶ Η ενεργοποίηση του ανεμιστήρα εκτελείται μέσω του διακόπτη (switch 5 - PORTF 5).
 - ▶ Η κυκλική κίνηση της βάσης ενεργοποιείται (παλμός περιόδου $T_b = 2 \text{ ms}$ και κύκλος λειτουργίας $D_b = 40\%$).
 - ▶ Η κυκλική κίνηση των λεπίδων ενεργοποιείται (παλμός περιόδου $T_l = 1 \text{ ms}$ και κύκλος λειτουργίας $D_l = 50\%$).
 - ▶ Ο Μετατροπέας Αναλογικού σε Ψηφιακό (Analog to Digital Converter – ADC) ενεργοποιείται.
- ▶ Αν ο ADC εντοπίσει κάποια τιμή μικρότερη ενός προκαθορισμένου κατωφλίου (threshold), ο ανεμιστήρας απενεργοποιείται άμεσα και ένα τρίτο LED (LED2 του PORTD) ενεργοποιείται.
- ▶ Για να λειτουργήσει ξανά ο ανεμιστήρας πρέπει να ενεργοποιηθεί ο διακόπτης SW5 (το LED2 απενεργοποιείται).

Παραδοχές (2/2)

- ▶ Όταν βρισκόμαστε στην κανονική λειτουργία (ανεμιστήρας ενεργοποιημένος):
 - ▶ Ο διακόπτης SW5 ενεργοποιείται για δεύτερη φορά → Η περίοδος του παλμού της κυκλικής κίνησης των λεπίδων διπλασιάζεται με κύκλο λειτουργίας (duty cycle) $TI = 50\%$.
 - ▶ Ο διακόπτης SW5 ενεργοποιείται για τρίτη φορά → Ο ανεμιστήρας απενεργοποιείται και οι δύο παλμοί μαζί με τον ADC σταματούν τη λειτουργίας τους.

Απεικόνιση ATmega 4808



Ερωτήματα

- ▶ **Ερώτημα 1:** Υλοποιήστε τη λειτουργία ενεργοποίησης του ανεμιστήρα μετά την ενεργοποίηση του διακόπτη (switch 5) δηλαδή, οι δύο παλμοί PWM αρχικοποιούνται και οδηγούν τα δύο LEDs.
- ▶ **Ερώτημα 2:** Προσθέστε τη λειτουργία του ADC (διακοπή της λειτουργίας του ανεμιστήρα και ενεργοποίηση του LED2). Υλοποιήστε τη λειτουργία επανεκκίνησης του ανεμιστήρα μέσω του SW5.
- ▶ **Ερώτημα 3:** Προσθέστε τη δεύτερη ρύθμιση του παλμού της κυκλικής κίνησης των λεπίδων μέσω της ενεργοποίησης του διακόπτη (switch 5). Τέλος, προσθέστε τη λειτουργία απενεργοποίησης του ανεμιστήρα.

Βιβλιογραφία

Βιβλιογραφία

- ▶ Microchip, Επίσημη ιστοσελίδα, <https://www.microchip.com/mplab/microchip-studio>
- ▶ Microchip, “AVR-IoT Wx Hardware User Guide”,
<https://ww1.microchip.com/downloads/en/DeviceDoc/AVR-IoT-Wx-Hardware-User-Guide-DS50002805C.pdf>
- ▶ Microchip, “ATmega4808/4809 Data Sheet”,
<https://ww1.microchip.com/downloads/en/DeviceDoc/ATmega4808-09-DataSheet-DS40002173B.pdf>
- ▶ Microchip, “Advanced Software Framework (ASF)”,
<https://www.microchip.com/mplab/avr-support/advanced-software-framework>
- ▶ Microchip, “MPLAB® XC Compilers”, <https://www.microchip.com/en-us/development-tools-tools-and-software/mplab-xc-compilers>