Εργαστήριο Μικρουπολογιστών

Παπαδόπουλος Κωνσταντίνος el20152 Σκούρτης Παύλος el20052

ΑΣΚΗΣΗ 1

.include "m328PBdef.inc"	
.equ FOSC_MHZ=16 ;MHz	
.equ DEL_mS=100 ;mS	
.equ DEL_NU= FOSC_MHZ*DEL_mS	
.def dc_value=r27	
TABLE:	
.DW Ox0005,0x001A,0x002E,0x0042,0x0057, Ox006B,0x007F,0x0094,0x00A8,0x00B D	Πίνακας με τις προϋπολογισμένες τιμές
.DW 0x00D1,0x00E6,0x00FA	
reset:	
ldi r24,low(RAMEND)	
out SPL,r24	
ldi r24,high(RAMEND)	
out SPH,r24	
ldi r26,0b00111111	
out DDRB, r26	;PORTB=out
clr r26	
out DDRD,r26	;PortD == in
ldi r16,(1 << WGM10) (1 << COM1A1)	;set TMR1Ain fast PWM 8-bit mode with non-inverted output and prescale=8
sts TCCR1A, r16	
ldi r16,(1 << WGM12) (1 << CS10)	
sts TCCR1B, r16	

Idi ZH,HIGH(TABLE*2) Idi ZL,LOW(TABLE*2) adiw ZL,12 Ipm mov dc_value,rO main: Ipm mov dc_value,rO start: sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rimp syn sbrs r26,2 rimp meion rimp start syn: Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) breq start adiw ZL,2 rimp main meion: Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r25, high(DEL_NU)		
adiw ZL,12 pm mov dc_value,r0 main: pm mov dc_value,r0 start: sts OCR1AL,dc_value in r26, PIND Eλεγχος εισόδου sbrs r26,1 Fimp syn sbrs r26,2 Fimp meion rimp start Fimp syn: start Styn: Πατήθηκε αύξηση φωτεινότητας di r24, low(DEL_NU) Idi r25, high(DEL_NU) roall delay_mS Χρησιμοποιείται για debouncing cpi dc_value,OxFA Ελεγχος ορίων breq start adiw ZL,2 rimp main Πατήθηκε μείωση φωτεινότητας ldi r24, low(DEL_NU)	ldi ZH,HIGH(TABLE*2)	
Ipm mov dc_value,rO main: Ipm mov dc_value,rO start: sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, pide value,0xFA Idi r24, low(DEL_NU) Idi r25, rjmp main Idi r24, low(DEL_NU) Idi r25, rjmp main Idi r24, low(DEL_NU) Idi r25, rjmp main Idi r24, low(DEL_NU) Idi r26, rjmp main Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low(DEL_NU) Idi r20, rjmp main Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r26, low(DEL_NU) Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low(DEL_NU) Idi r20, low(DEL_NU) Idi r20, low(DEL_NU) Idi r24, low(DEL_NU)	ldi ZL,LOW(TABLE*2)	;so its the 7th value on the TABLE
mov dc_value,r0 main: Ipm mov dc_value,r0 start: sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,0xFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, high(DEL_NU) Idi r27, high(DEL_NU) Idi r28, high(DEL_NU) Idi r29, high(DEL_NU) Idi r20, low(DEL_NU) Idi r21, low(DEL_NU) Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r24, low(DEL_NU)	adiw ZL,12	
main: Ipm	lpm	
Ipm mov dc_value,rO start: sts OCR1AL,do_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,OxFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r25, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, low DEL_NU Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low DEL_NU Idi r24, low(DEL_NU) Idi r24, low(DEL_NU)	mov dc_value,r0	
Ipm mov dc_value,rO start: sts OCR1AL,do_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,OxFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r25, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, low DEL_NU Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low DEL_NU Idi r24, low(DEL_NU) Idi r24, low(DEL_NU)		
Ipm mov dc_value,rO start: sts OCR1AL,do_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,OxFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r25, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, low DEL_NU Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low DEL_NU Idi r24, low(DEL_NU) Idi r24, low(DEL_NU)		
Ipm mov dc_value,rO start: sts OCR1AL,do_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,OxFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r25, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, low DEL_NU Idi r27, low(DEL_NU) Idi r28, low(DEL_NU) Idi r29, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low(DEL_NU) Idi r25, low DEL_NU Idi r24, low DEL_NU Idi r24, low(DEL_NU) Idi r24, low(DEL_NU)		
mov dc_value,rO start: sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,0xFA breq start adiw ZL,2 rjmp main meion: Idi r24, Iow(DEL_NU) Idi r24, Iow(DEL_NU) Idi r25, high(DEL_NU) Idi r26, high(DEL_NU) Idi r27, Iow(DEL_NU) Idi r28, high(DEL_NU) Idi r29, Iow(DEL_NU) Idi r20, Iow(DEL_NU)	main:	
start: sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) reall delay_mS cpi dc_value,0xFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, high(DEL_NU) Idi r27, low(DEL_NU) Idi r28, high(DEL_NU) Idi r29, high(DEL_NU) Idi r20, low(DEL_NU)	lpm	
sts OCR1AL,dc_value in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,0xFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, value,0xFA Idi r27, value,0xFA Idi r28, value,0xFA Idi r29, low(DEL_NU) Idi r20, value,0xFA Idi r20, value,	mov dc_value,r0	
in r26, PIND sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS cpi dc_value,0xFA breq start adiw ZL,2 rjmp main meion: Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) Idi r26, value,0xFA Idi r27, value,0xFA Idi r28, value,0xFA Idi r28, value,0xFA Idi r29, value,0xFA Idi r24, low(DEL_NU) Idi r24, low(DEL_NU) Idi r25, value,0xFA Idi r26, value,0xFA Idi r27, value,0xFA Idi r28, value,0xFA Idi r28, value,0xFA Idi r29, value,0xFA Idi r2	start:	
Sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn:	sts OCR1AL,dc_value	
sbrs r26,1 rjmp syn sbrs r26,2 rjmp meion rjmp start syn: Πατήθηκε αύξηση φωτεινότητας Idi r24, low(DEL_NU) rcall delay_mS cpi dc_value,OxFA breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	in r26, PIND	Έλενγος εισόδου
sbrs r26,2 rjmp meion rjmp start syn:	sbrs r26,1	Σλογχος οισσοσσ
rjmp meion rjmp start syn: Πατήθηκε αύξηση φωτεινότητας Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS Χρησιμοποιείται για debouncing cpi dc_value,0xFA Έλεγχος ορίων breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	rjmp syn	
rjmp start syn: Πατήθηκε αύξηση φωτεινότητας Idi r24, low(DEL_NU) Idi r25, high(DEL_NU) rcall delay_mS Χρησιμοποιείται για debouncing cpi dc_value,0xFA Έλεγχος ορίων breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	sbrs r26,2	
syn:	rjmp meion	
Idi r24, low(DEL_NU)Idi r25, high(DEL_NU)rcall delay_mSΧρησιμοποιείται για debouncingcpi dc_value,0xFAΈλεγχος ορίωνbreq startadiw ZL,2rjmp mainΠατήθηκε μείωση φωτεινότηταςIdi r24, low(DEL_NU)	rjmp start	
Idi r25, high(DEL_NU)Χρησιμοποιείται για debouncingrcall delay_mSΧρησιμοποιείται για debouncingcpi dc_value,0xFAΈλεγχος ορίωνbreq startαdiw ZL,2rjmp mainΠατήθηκε μείωση φωτεινότηταςIdi r24, low(DEL_NU)	syn:	Πατήθηκε αύξηση φωτεινότητας
rcall delay_mS Cpi dc_value,0xFA Éλεγχος ορίων breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	ldi r24, low(DEL_NU)	
cpi dc_value,0xFA Έλεγχος ορίων breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	ldi r25, high(DEL_NU)	
breq start adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	rcall delay_mS	Χρησιμοποιείται για debouncing
adiw ZL,2 rjmp main meion: Πατήθηκε μείωση φωτεινότητας ldi r24, low(DEL_NU)	cpi dc_value,0xFA	Έλεγχος ορίων
rjmp main meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	breq start	
meion: Πατήθηκε μείωση φωτεινότητας Idi r24, low(DEL_NU)	adiw ZL,2	
ldi r24, low(DEL_NU)	rjmp main	
	meion:	Πατήθηκε μείωση φωτεινότητας
ldi r25, high(DEL_NU)	ldi r24, low(DEL_NU)	
	ldi r25, high(DEL_NU)	

rcall delay_mS		Χρησιμοποιείται για debouncing
cpi dc_value,0x05		Έλεγχος ορίων
breq start		
sbiw ZL,2		
rjmp main		
delay_mS:		
ldi r23, 249		
loop_inn:		
dec r23	; 1 cycle	
nop	; 1 cycle	
brne loop_inn	; 1 or 2 cycles	
sbiw r24 ,1	; 2 cycles	
brne delay_mS	; 1 or 2 cycles	
ret	;4 cycl	

ΑΣΚΗΣΗ 2

#define F_CPU 1600000UL	
#include "avr/io.h"	
#include "util/delay.h"	
void adc_init(){	
return;	
}	
uint8_t table [9]={0, 32, 64, 96. 128, 160, 192, 224,255};	
void open(){	
ADCSRA =(1 << ADSC);	
uint8_t temp;	
while(1){	
temp=ADCSRA;	
temp=temp & 64; //0b01000000; apomonosi ADSC	
if(temp==0)break;	
}	
uint8_t adcvalueh,adcvaluel;	
adcvaluel=ADCL;	
adcvalueh=ADCH;	
if(adcvalueh>0 && adcvalueh<=32) {PORTD=0b00000001;}	Κατηγοροιοποίηση σταθμών τάσης και
if(adcvalueh>32 && adcvalueh<=64) {PORTD=0b00000010;}	αντίστοιχη εξόδου
if(adcvalueh>64 && adcvalueh<=96) {PORTD=0b00000100;}	
if(adcvalueh>96 && adcvalueh<=128) {PORTD=0b00001000;}	

if(adcvalueh>128 & adcvalueh<=160) {		
(PORTD=0b00100000); if(adcvalueh>192 && adcvalueh<=224) (PORTD=0b01000000); if(adcvalueh>224 && adcvalueh<=255) (PORTD=0b10000000); return; return; int main(){ TCCR1A = (1 << WGM10) (1 << COM1A1); (1 << CS11); (1 << CS11);		
{PORTD=0b01000000;} if(adcvalueh>224 && adcvalueh<=255) {PORTD=0b10000000;} return; } int main(){ TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b1000011; uint8_t duty=134; OCR1AL=duty; DDRB =0b0000010; DDRD =0b1111111; while(1){ uint8_t inpout; input = PINB; _delay_ms(100); debouncing AT0µ0νωση button p5-p6 as 0b00110000 = 48		
PORTD=0b10000000;		
int main(){ TCCR1A = (1 << WGM10) (1 << COM1A1); TCCR1B = (1 << WGM12) (1 << CS11); ADMUX = 0b01100001; ADCSRA = 0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010; DDRD =0b1111111; while(1){ uint8_t inpout; input = PINB; _delay_ms(100); input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48		
int main(){ TCCR1A = (1 << WGM10) (1 << COM1A1); TCCR1B = (1 << WGM12) (1 << CS11); ADMUX = 0b01100001; ADCSRA = 0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010; DDRD =0b1111111; while(1){ uint8_t inpout; input = PINB; _delay_ms(100); input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48		
int main(){ TCCR1A = (1 << WGM10) (1 << COM1A1); TCCR1B = (1 << WGM12) (1 << CS11); ADMUX = 0b01100001; ADCSRA = 0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010; DDRD =0b1111111; while(1){ uint8_t inpout; input = PINB; _delay_ms(100); input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48		
int main(){ TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;	return;	
int main(){ TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;		
TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;	}	
TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;		
TCCR1A =(1 << WGM10) (1 << COM1A1); TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;		
TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;	int main(){	
TCCR1B =(1 << WGM12) (1 << CS11); ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;		
ADMUX =0b01100001; ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;	TCCR1A =(1 << WGM10) (1 << COM1A1);	
ADCSRA =0b10000111; uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010;	TCCR1B =(1 << WGM12) (1 << CS11);	
uint8_t duty=134; OCR1AL=duty; DDRB =0b00000010; //B input DDRD =0b11111111; //D output while(1){ uint8_t inpout; input = PINB; debouncing delay_ms(100); debouncing input = input & 48; Aπομόνωση button p5-p6 as Ob00110000 = 48	ADMUX =0b01100001;	
OCR1AL=duty; //B input DDRB =0b00000010; //B input DDRD =0b11111111; //D output while(1){ (aunt8_t inpout; input = PINB; (aebouncing) delay_ms(100); (aebouncing) input = input & 48; (Aπομόνωση button p5-p6) as Ob00110000 = 48	ADCSRA =0b10000111;	
DDRB = Ob00000010;	uint8_t duty=134;	
DDRD =0b11111111; //D output while(1){	OCR1AL=duty;	
while(1){ uint8_t inpout; input = PINB; _delay_ms(100); debouncing input = input & 48; Απομόνωση button p5-p6 as 0b00110000 = 48	DDRB =0b00000010;	//B input
uint8_t inpout; input = PINB; _delay_ms(100); debouncing input = input & 48; Απομόνωση button p5-p6 as 0b00110000 = 48	DDRD =0b11111111;	//D output
input = PINB; debouncing _delay_ms(100); debouncing input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48	while(1){	
_delay_ms(100); debouncing input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48	uint8_t inpout;	
input = input & 48; Aπομόνωση button p5-p6 as 0b00110000 = 48	input = PINB;	
as 0b00110000 = 48	_delay_ms(100);	debouncing
if(input==32) {if(duty<245){duty+=20;OCR1AL=duty;} } Μείωση button pb5	input = input & 48;	
	if(input==32) {if(duty<245){duty+=20;OCR1AL=duty;} }	Μείωση button pb5

else if(input==16) {if(duty>25){duty- =20;OCR1AL=duty;}}	Αύξηση button pb6
open();	
}	
}	

Άσκηση 3

#define F_CPU 1600000UL	
#include "avr/io.h"	
#include "util/delay.h"	
uint8_t duty=134;	
void adc_init(){	
return;	
}	
uint8_t table [9]={0, 32, 64, 96, 128, 160, 192, 224,255};	
void mode1(){	Παρόμοιο με ασκ2
uint8_t input;	
input = PIND;	
_delay_ms(100);	//delay for debouncing
input = input & 6;	//απομόνωση button pd1- pd2 as 0b00000110 = 6
if(input==4) {if(duty<245){duty+=20;}}	//pb1> αύξηση
else if(input==2) {if(duty>25){duty-=20;} } // != exw arnitiki logiki	//pb2> μείωση
OCR1AL=duty;	
}	
void mode2(){	
ADCSRA =(1 << ADSC);	
uint8_t temp;	
while(1){	
temp=ADCSRA;	

temp=temp & 64;	//0b01000000; απομόνωση ADSC
if(temp==0)break;	
}	
uint8_t adcvalueh,adcvaluel, duty2;	
adcvaluel=ADCL;	
adcvalueh=ADCH;	
if(adcvalueh>0 && adcvalueh<=32){duty=250;}	Κατηγοροιοποίηση
if(adcvalueh>32 && adcvalueh<=64){duty=200;}	σταθμών τάσης και αντίστοιχη εξόδου
if(adcvalueh>64 && adcvalueh<=96){duty=170;}	
if(adcvalueh>96 && adcvalueh<=128){duty=140;}	
if(adcvalueh>128 && adcvalueh<=160){duty=110;;}	
if(adcvalueh>160 && adcvalueh<=192){duty=80;}	
if(adcvalueh>192 && adcvalueh<=224){duty=50;}	
if(adcvalueh>224 && adcvalueh<=255){duty=20;}	
OCR1AL=duty;	
return;	
}	
int main(){	
TCCR1A =(1 << WGM10) (1 << COM1A1);	

	1
TCCR1B =(1 << WGM12) (1 << CS11);	
ADMUX =0b01100000;	// adc0 —>ποντεσιόμετρο
ADCSRA =0b10000111;	
DDRD =0b0000000;	//D input
DDRB =0b11111111;	//B output
PORTB=0b00000000;	
int mode=0;	
while(1){	
uint8_t input;	
input = PIND;	
_delay_ms(100);	//delay for debouncing
input = input & 192;	//button p6-p7 as 0b11000000 = 192
if(input==128) {mode=1;}	// PRESSED P7
else if(input==64) {mode=2;}	// PRESSED P6
if (mode==1){	
mode1();	
}	
if(mode==2) {	
mode2();	
}	
}	
}	