**Тема: Аналіз та синтез систем**

* Служба приїжджає на місце спеціальною машиною, відкриває люк і очищує цю посудиною.
* Вивозить це в спеціальна місця.

*Індивідуальне завдання*

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;

; MAIN PROGRAM

;

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

main:

rsp

ld a, #%01111100

ld $04, a

ld $05, a

ld a, #%00110000

ld $03, a ; stop motor 1

ld a, #%00011111

ld $01, a

ld $02, a

ld a, #%00000000

ld $00, a ; stop motor 2

;=======================================================

clr $80

LD A,#%00010000 ; завантажити маску джерела таймеру

LD $0D,A ; джерело таймеру CNTR1 від так тової частоти

LD A,#%00001111 ;1000

LD $10,A ; ATR1 high

LD $1b,A ; DCR2 high (pa4)

LD A,#%00010000 ; завантажити маску вибору ШИМ на PA3

LD $12,A ; вибір ШИМ

LD A,#%10000000

LD $11,A ; ATR1 low (0..255)

LD A,#%10000001 ; значення = 1

LD $1C,A ; DCR2 low (pa4)

LD A,#1 ; завантажити маску дозволу зміни ширини

LD $21,A ; дати дозвіл на зміну ширини імпульсу

;============================================================================

;jp MOTOR2

MOTOR1:

;============================================================================

call Read\_Signal

cp A, #10 ;Еталонне значення

;==============================our kroks============================

ld a, #10

CALL VLIVO1

ld a, #10

CALL VVERH1

ld a, #10

CALL VPRAVO1

ld a, #10

CALL VLIVO1

ld a, #10

CALL VVERH1

ld a, #10

CALL VPRAVO1

ld a, #10

CALL VLIVO1

ld a, #10

CALL VNIZ1

ld a, #10

CALL VPRAVO1

ld a, #10

CALL VLIVO1

ld a, #10

CALL VNIZ1

ld a, #10

CALL VPRAVO1

jp MOTOR1

;=============================================

VLIVO1:

push a

ld a, #%01110100 ;song

ld $03, a

call PAUSE\_S

ld a, #%00100100

ld $03, a

call PAUSE\_S

ld a, #%01100000 ;song

ld $03, a

call PAUSE\_S

ld a, #%00000000

ld $03, a

call PAUSE\_S

ld a, #%00010000

ld $03, a

call PAUSE\_S

ld a, #%00011000

ld $03, a

call PAUSE\_S

ld a, #%00111000

ld $03, a

call PAUSE\_S

ld a, #%00111100

ld $03, a

call PAUSE\_S

pop a

dec a

jrne VLIVO1

ld a, #%00110000

ld $03, a ; stop motor 1

RET

;==============================================

VPRAVO1:

push a

;8

ld a, #%00111100

ld $03, a

call PAUSE\_S

;7

ld a, #%00111000

ld $03, a

call PAUSE\_S

;6

ld a, #%00011000

ld $03, a

call PAUSE\_S

;5

ld a, #%00010000

ld $03, a

call PAUSE\_S

;4

ld a, #%00000000

ld $03, a

call PAUSE\_S

;3

ld a, #%00100000

ld $03, a

call PAUSE\_S

;2

ld a, #%00100100

ld $03, a

call PAUSE\_S

;1

ld a, #%00110100

ld $03, a

call PAUSE\_S

pop a

dec a

jrne VPRAVO1

ld a, #%00110000

ld $03, a ; stop motor 1

RET

;==============================================

VNIZ1:

push A

ld a, #%00000001 ;1

ld $00, a

call PAUSE\_S

ld a, #%00000011 ;1-2

ld $00, a

call PAUSE\_S

ld a, #%00000010 ;2

ld $00, a

call PAUSE\_S

ld a, #%00000110 ;2-3

ld $00, a

call PAUSE\_S

ld a, #%00000100 ;3

ld $00, a

call PAUSE\_S

ld a, #%00001100 ;3-4

ld $00, a

call PAUSE\_S

ld a, #%00001000 ;4

ld $00, a

call PAUSE\_S

ld a, #%00001001 ;4-1

ld $00, a

call PAUSE\_S

pop A

dec A

jrne VNIZ1

ld A,#%00000000

ld $00,A ; stop motor 2

RET

;==============================================

VVERH1:

push A

ld a, #%00001000 ;4

ld $00, a

call PAUSE\_S

ld a, #%00001100 ;3-4

ld $00, a

call PAUSE\_S

ld a, #%00000100 ;3

ld $00, a

call PAUSE\_S

ld a, #%00000110 ;2-3

ld $00, a

call PAUSE\_S

ld a, #%00000010 ;2

ld $00, a

call PAUSE\_S

ld a, #%00000011 ;1-2

ld $00, a

call PAUSE\_S

ld a, #%00000001 ;1

ld $00, a

call PAUSE\_S

ld a, #%00001001 ;4-1

ld $00, a

call PAUSE\_S

pop A

dec A

jrne VVERH1

ld A,#%00000000

ld $00,A ; stop motor 2

RET

;=============================================

PAUSE\_S:

ld a, #6

l\_pause:

push a

ld a, #255

l\_pa:

bset $2e, #6

dec a

jrne l\_pa

pop a

dec a

jrne l\_pause

ret

Read\_Signal:

LD A,#%01100001 ; Завантажити в A маску настройки АЦП для

; PB0

LD $34,A ; Записати її в регістр ADCCSR, що дає дозвіл ан

;алогового вводу

; з ніжки PB0.

M1:

BTJF $34,#7,M1 ; Переписати біт EOC в CF і перейти на M1,

;якщо він нульовий.

LD A,$35 ; Зчитати в регістр A отримане значення.

ret

dummy\_rt:

IRET ; Empty Procedure : Mean return to Main program.