



נספח י' – תוכנת הרובוט "קרפלעד"

התוכנה המופיעה בנספח זה הינה התוכנה בגרסתה האחרונה אותה הרצנו בתחרות העולמית בארצות הברית. חלק מהתוכנה כבר מופיע בפרקים שונים בתוך העבודה אך כאן היא מופיעה בשלמותה.

התוכנה הראשית

```
#include c:\ax12\include\EQUATES.asm
#include c:\ax12\include\INITS.asm
#include c:\ax12\include\HELPFUNC.asm
#include c:\ax12\include\SENSORS.asm
#include c:\ax12\include\motors.asm
#include c:\ax12\include\IR_OVF.asm
#include c:\ax12\include\Tikunim.asm
#include c:\ax12\include\FIRE.asm
#include c:\ax12\include\PRINT.asm

PROGRAM:
INT_RUN:

        CLR    R00M
        CLR    PLZ_TURN
        MOVB   #$21, WALL           ;2A
        MOVB   #$90, TURNWEAK
        MOVW   #9000, E_TURN        ;6
        JSR    X_L
        CLI

SECT1:

        JSR    TKNBYRGT
        TST    PLZ_TURN
        BEQ    SECT1
        JSR    M_BRAKE
        MOVB   #$03, PWEN
        JSR    TURN_LEFT
        JSR    L_BRAKE
        JSR    Q_ENGAGE
        JSR    UV_ON

SECT2:

        JSR    TKNBYRGT
        JSR    US1
        LDX    US_READ
        CPX    #$400
        BPL    SECT2

SECT3:

        JSR    TKNBYRGT
        JSR    US1
        LDX    US_READ
        CPX    #$400
        BMI    SECT3
        MOVB   #$1A, WALL
        CLR    PLZ_TURN

SECT3_1:
```

בעלי אתר הרובוטיקה הישראלי לא יסאו באחריות כלשהי לכל נזק, כספי או אחר שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



	JSR	TKNBYRGT	
	TST	PLZ_TURN	
	BEQ	SECT3_1	
	JSR	UV_OFF	
	CLR	PLZ_TURN	
	MOVB	#\$1F, WALL	
	MOVB	#\$90, TURNWEAK	;A0
	MOVW	#9100, E_TURN	;8500
SECT4:			
	JSR	TKNBYRGT	
	TST	PLZ_TURN	
	BEQ	SECT4	
	JSR	CHECK_UV	
	INC	R00M	
	JSR	TURN_LEFT	
	JSR	L_BRAKE	
	JSR	MOVBYTES	
	JSR	Q_ENGAGE	
SECT5:			
	JSR	TKNBYRGT	
	JSR	US0	
	LDX	US_READ	
	CPX	#\$400	
	BHI	SECT5	
	MOVB	#PWDREG, PWDTY0	
	MOVB	#PWDREG, PWDTY1	
SECT6:			
	JSR	TKNBYLFT	
	JSR	US2	
	LDX	US_READ	
	CPX	#\$400	
	BMI	SECT6	
SECT7:			
	MOVB	#PWDREG, PWDTY0	
	MOVB	#PWDREG, PWDTY1	
	JSR	US2	
	LDX	US_READ	
	CPX	#\$400	
	BPL	SECT7	
	CLR	PLZ_TURN	
	MOVB	#\$20, WALL	
	MOVB	#\$90, TURNWEAK	
	MOVW	#11000, E_TURN	
SECT8:			
	JSR	TKNBYRGT	
	TST	PLZ_TURN	
	BEQ	SECT8	
	JSR	TURN_RIGHT	; Turn Into Room 4
	JSR	L_BRAKE	
OTR_STRT:	MOVB	#\$03, PWEN	
	JSR	MAKOM_L	
	JSR	Q_ENGAGE	
SECT8_1:	JSR	PAS_LAV	; Check For Pas Lavan
	TST	PAS_FLAG	
	BEQ	SECT8_1	

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
LDX    #1200
JSR     L_ENCO
JSR     PLS_CNT
JSR     L_BRAKE
MOVB    #$03, PWEN
JSR     MAKOM_L
JSR     UV_ON_L

SECT8B:
MOVB    #$03, PWEN
MOVB    #$13, PORTPP
MOVB    #$98, PWDY1
MOVB    #$98, PWDY0
LDX     #6700                ; Turn 90 Degrees
JSR     L_ENCO
JSR     PLS_CNT
JSR     BRK_TRNB
JSR     L_BRAKE

MOVB    #$03, PWEN
MOVB    #$43, PORTPP
MOVB    #$98, PWDY1
MOVB    #$98, PWDY0
LDX     #700
JSR     L_ENCO
JSR     PLS_CNT
JSR     BRK_TRNB
JSR     L_BRAKE

JSR     UV_OFF
JSR     CHECK_UV
INC      R00M
MOVB    #$03, PWEN
MOVB    #$13, PORTPP
MOVB    #$C8, PWDY1
MOVB    #$C8, PWDY0
LDX     #3000                ;6      ; Turn 90 Degrees
JSR     L_ENCO
JSR     PLS_CNT
JSR     BRK_TRNB
JSR     L_BRAKE
MOVB    #$03, PWEN
JSR     MAKOM_R
CLR      PLZ_TURN            ; Clear Value
MOVB    #$24, WALL          ; Enter Values For Wall

SECT9:
SECT9_1:
JSR     Q_ENGAGE
JSR     TKNBYRGT
TST     PLZ_TURN
BEQ     SECT9_1              ; Taken By Right Till Turn
JSR     L_BRAKE
MOVW    #8750, E_TURN
JSR     TURN_LEFT_B
JSR     L_BRAKE              ; Do Turn ;M
MOVB    #$03, PWEN
JSR     MAKOM_R              ; Taken Bamakom By Right
JSR     L_BRAKE
```

בעלי אתר הרובוטיקה הישראלי לא יסאו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
JSR    Q_ENGAGE
MOVB   #$B0, PWDTY0
MOVB   #$B0, PWDTY1
SECTA:
JSR    US3
LDX    US_READ
CPX    #$400
BLO    SECTA
JSR    UV_ON
MOVB   #$A0, PWDTY0
MOVB   #$A0, PWDTY1
SECTA2:
JSR    US2
LDX    US_READ
CPX    #$400
BLO    SECTA2
MOVW   #1500, E_TURN
JSR    TURN_RIGHT
JSR    L_BRAKE
MOVB   #$03, PWEN
JSR    MAKOM_L
MOVB   #$03, PWEN
MOVB   #$43, PORTPP
MOVB   #PWDREG, PWDTY1
MOVB   #PWDREG, PWDTY0
LDX    #1200
JSR    L_ENCO
JSR    PLS_CNT
JSR    BRK_TRNB
JSR    L_BRAKE
JSR    UV_OFF
JSR    CHECK_UV
INC    R00M
JSR    Q_ENGAGE

;=====
;=====TO ROOM 2!!!!=====
;=====
SECTE:
JSR    TKNBYLFT                ; Taken By Left to room 2
JSR    US1
LDX    US_READ
CPX    #$450
BMI    SECTE
JSR    L_BRAKE
MOVB   #$03, PWEN
JSR    MAKOM_R                ; Taken Bamakom Before Turn
MOVB   #$03, PWEN
JSR    MAKOM_R                ; Taken Bamakom Before Turn
MOVB   #$03, PWEN
MOVW   #11000, E_TURN
JSR    TURN_LEFT_B            ; Turn To Corridor 2
JSR    L_BRAKE
JSR    Q_ENGAGE
;      JSR      MOVBYTES
SECTF:
JSR    US0                    ; Wait Till In Corridor
LDX    US_READ
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



SECT11:	CPX	#\$360	
	BHI	SECTF	
	CLR	PLZ_TURN	
	MOVB	#\$21, WALL	
SECT12:	JSR	TKNBYLFT	
	TST	PLZ_TURN	
	BEQ	SECT11	
	JSR	L_BRAKE	
	MOVW	#9250, E_TURN	;8150
	JSR	TURN_RIGHT_B	
	JSR	L_BRAKE	
	MOVB	#\$03, PWEN	
	JSR	MAKOM_L	
	JSR	Q_ENGAGE	
SECT13:	JSR	PAS_LAV	
	TST	PAS_FLAG	
	BEQ	SECT12	
	JSR	UV_ON	
	JSR	L_BRAKE	
	JSR	CHECK_UV	
	BRA	SECT13	

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Tikunim.asm

TKNBYRGT :

```
PSHD
PSHX
JSR    US2
LDD    US_READ
JSR    D_INT0_A
STAA   BACK_DIS
JSR    US3
LDD    US_READ
JSR    D_INT0_A
STAA   FRNT_DIS
LDAB   BACK_DIS
SBA
JSR    ABS_A
STAA   DIFFER
CMPA   #$4
BPL    TKN_ANGL
```

TKN_MID :

```
MOVB   #$58, PWD4TKN
LDAA   FRNT_DIS
LDAB   BACK_DIS
ABA
LSRA
CMPA   #$16
BLO    TKN_LFT
MOVB   #$6A, PWD4TKN
CMPA   #$20
BLO    TKN_LFT
MOVB   #$58, PWD4TKN
CMPA   #$31
BHI    TKN_RGT
MOVB   #$74, PWD4TKN
CMPA   #$27
BHI    TKN_RGT
LDAA   DIFFER
CMPA   #$1
BHI    TKN_ANGL
```

TKN_ANGL :

```
LDX    #DUTY4TKN
LDAB   A, X
STAB   PWD4TKN
```

CHS_TKN :

```
TST    WR_2_TKN
BEQ    TKN_LFT
```

TKN_RGT :

```
MOVB   PWD4TKN, PWDTY0
MOVB   #PWDREG, PWDTY1
BRA    END_TKN
```

TKN_LFT :

```
MOVB   PWD4TKN, PWDTY1
MOVB   #PWDREG, PWDTY0
BRA    END_TKN
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
NO_TKN:
    MOVB    #PWDREG, PWDTY0
    MOVB    #PWDREG, PWDTY1

END_TKN:
    PULX
    PULD
    RTS

TKNBYLFT:
    PSHD
    PSHX
    JSR      US0
    LDD      US_READ
    JSR      D_INT0_A
    STAA     BACK_DIS
    JSR      US1
    LDD      US_READ
    JSR      D_INT0_A
    STAA     FRNT_DIS
    LDAB     BACK_DIS
    SBA
    COMA
    JSR      ABS_A
    STAA     DIFFER
    CMPA     #$4
    LBMI     TKN_MID2
    LBRA     TKN_ANGL

TKN_MID2:
    MOVB     #$70, PWD4TKN
    LDAA     FRNT_DIS
    LDAB     BACK_DIS
    ABA
    LSRA
    CMPA     #$1A
    LBLO     TKN_RGT
    MOVB     #$70, PWD4TKN
    CMPA     #$25
    LBHI     TKN_LFT
    MOVB     #$A0, PWD4TKN
    LDAA     DIFFER
    CMPA     #$1
    LBPL     TKN_ANGL
    LBRA     NO_TKN

;=====
;          TAKEN BAMAKOM
;=====

MAKOM_L:
    CLR      MAKM_MONE
    MOVB     #$FF, MAKOM_C

MAKOM_L2:
    INC      MAKM_MONE
    LDAA     MAKM_MONE
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



	CMPA	#MAKOM_OVF
	LBEQ	M_RESCUE
	JSR	US0
	LDD	US_READ
	JSR	D_INT0_A
	STAA	BACK_DIS
	JSR	US1
	LDD	US_READ
	JSR	D_INT0_A
	STAA	FRNT_DIS
	LDAB	BACK_DIS
	SBA	
	JSR	ABS_A
	STAA	DIFFER
	CMPA	#\$2
	BMI	DO_NOT
	TST	WR_2_TKN
	BNE	DO_LEFT
	BRA	DO_RIGHT
MAKOM_R:		
	CLR	MAKM_MONE
	MOVB	#\$00, MAKOM_C
MAKOM_R2:		
	INC	MAKM_MONE
	LDAA	MAKM_MONE
	CMPA	#MAKOM_OVF
	BEQ	M_RESCUE
	JSR	US2
	LDD	US_READ
	JSR	D_INT0_A
	STAA	BACK_DIS
	JSR	US3
	LDD	US_READ
	JSR	D_INT0_A
	STAA	FRNT_DIS
	LDAB	BACK_DIS
	SBA	
	JSR	ABS_A
	STAA	DIFFER
	CMPA	#\$2
	BMI	DO_NOT
	TST	WR_2_TKN
	BNE	DO_RIGHT
	BRA	DO_LEFT
DO_RIGHT:		
	MOVB	#\$88, PWDTY1
	MOVB	#\$88, PWDTY0
	MOVB	#\$13, PORTPP
	TST	MAKOM_C
	BEQ	MAKOM_R2
	LBRA	MAKOM_L2
DO_LEFT:		
	MOVB	#\$88, PWDTY1
	MOVB	#\$88, PWDTY0
	MOVB	#\$43, PORTPP

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
TST    MAKOM_C
BEQ    MAKOM_R2
LBRA   MAKOM_L2

DO_NOT:
      JSR    L_BRAKE
      MOVB   #$00, PWDTY1
      MOVB   #$00, PWDTY0
      MOVB   #$00, PORTPP
      MOVB   #$00, PWEN
      RTS

M_RESCUE:
      MOVB   #$03, PORTPP
      MOVB   #$B8, PWDTY1
      MOVB   #$B8, PWDTY0
      JSR    L_ENCO
      LDX    #200
      JSR    PLS_CNT
      TST    MAKOM_C
      LBEQ   MAKOM_R
      LBRA   MAKOM_L

;=====
;          TAKEN BACK
;=====

TKN_BCK:
      JSR    READ_ATD
      LDAB   ADR4H
      LDAA   ADR5H
      SBA
      BLO    TKN_ST_R
      TSTA
      BHI    TKN_ST_L
      MOVB   #$03, PORTPP
      MOVB   #$A0, PWDTY0
      MOVB   #$A0, PWDTY1
      RTS

TKN_ST_L:
      MOVB   #$A0, PWDTY1
      MOVB   #$A0, PWDTY0
      MOVB   #$13, PORTPP
      BRA    TKN_BCK

TKN_ST_R:
      MOVB   #$A0, PWDTY0
      MOVB   #$A0, PWDTY1
      MOVB   #$43, PORTPP
      BRA    TKN_BCK

DUTY4TKN:
      DB     200
      DB     200
      DB     180
      DB     176
      DB     170
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



DB	163
DB	156
DB	148
DB	140
DB	130
DB	120
DB	110
DB	110
DB	110
DB	110
DB	110
DB	110

Equates.asm

```
RAMSTRT: EQU $0800 ; start of internal ram
MONRAM: EQU $0A00 ; monitor ram start
RAMSIZE: EQU $0400 ; Size of internal ram
REGBS: EQU $0000 ; start of registers
ROMBS: EQU $FD80 ; start of initialization rom
DSTREE: EQU $1000 ; start of eeprom
DENDEE: EQU $1FFF ; end of eeprom
STACK: EQU RAMSTRT+RAMSIZE ; top of stack
LDS $7000

MONSTRT: EQU $2000 ; start of this program if running
under debugger
PRGSTRT: EQU $8000 ; start of this program if running
stand-alone

IBUFSIZ: EQU 35 ; input buffer size
EOT: EQU $04 ; end of text/table character

;*****
; define I/O registers...
;*****
PORTA: EQU REGBS+0 ;port A = Address lines A8 - A15
PORTB: EQU REGBS+1 ;port B = Address lines A0 - A7
DDRA: EQU REGBS+2 ;port A direction register
DRRB: EQU REGBS+3 ;port A direction register
PORTC: EQU REGBS+4 ;port C = Data 7-15 wide,Data 0-
7narrow
PORTD: EQU REGBS+5 ;port D = Data 0-7 wide
DDRC: EQU REGBS+6 ;port C direction register
DDRD: EQU REGBS+7 ;port D direction register
PORTE: EQU REGBS+8 ;port E = mode,IRQandcontrolsignals
DDRE: EQU REGBS+9 ;port E direction register
PEAR: EQU REGBS+$A ;port E assignments
MODE: EQU REGBS+$B ;Mode register
PUCR: EQU REGBS+$C ;port pull-up control register
RDRIV: EQU REGBS+$D ;port reduced drive control register
INITRM: EQU REGBS+$10 ;Ram location register
INITRG: EQU REGBS+$11 ;Register location register
INITEE: EQU REGBS+$12 ;EEprom location register
MISC: EQU REGBS+$13 ;Miscellaneous Mapping control
RTICTL: EQU REGBS+$14 ;Real time clock control
RTIFLG: EQU REGBS+$15 ;Real time clock flag
COPCTL: EQU REGBS+$16 ;Clock operating properly control
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף מוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



COPRST:	EQU	REGBS+\$17	;COP reset register
INTCR:	EQU	REGBS+\$1E	;interrupt control register
HPRIO:	EQU	REGBS+\$1F	;high priority reg
KWIED:	EQU	REGBS+\$20	;Key wake-up port D enable
KWIFD:	EQU	REGBS+\$21	;Key wake-up port D flags
PORTH:	EQU	REGBS+\$24	;port H = keypad port
DDRH:	EQU	REGBS+\$25	;port H direction register
KWIEH:	EQU	REGBS+\$26	;Key wake-up port H enable
KWIFH:	EQU	REGBS+\$27	;Key wake-up port H flags
PORTJ:	EQU	REGBS+\$28	;port J = Keypad / Serial ctl lines
DDRJ:	EQU	REGBS+\$29	;port J direction register
KWIEJ:	EQU	REGBS+\$2A	;Key wake-up port J enable
KWIFJ:	EQU	REGBS+\$2B	;Key wake-up port J flags
KPOLJ:	EQU	REGBS+\$2C	;port J wake-up polarity
PUPSJ:	EQU	REGBS+\$2D	;port J pull-up/down select
PULEJ:	EQU	REGBS+\$2E	;port J Pull-up/down enable
PORTF:	EQU	REGBS+\$30	;port F = Chip selects
PORTG:	EQU	REGBS+\$31	;port G = Address lines A16 - A21
DDRF:	EQU	REGBS+\$32	;port F direction register
DDRG:	EQU	REGBS+\$33	;port G direction register
DPAGE:	EQU	REGBS+\$34	;CSD chip select page register
PPAGE:	EQU	REGBS+\$35	;CSP chip select page register
EPAGE:	EQU	REGBS+\$36	;CS3/Epape chip select page register
WINDEF:	EQU	REGBS+\$37	;memory page window enable register
MXAR:	EQU	REGBS+\$38	;memory expansion enable A16-A21
CSCTL0:	EQU	REGBS+\$3C	;chip select control register
CSCTL1:	EQU	REGBS+\$3D	;chip select control register
CSSTR0:	EQU	REGBS+\$3E	;chip select stretch register
CSSTR1:	EQU	REGBS+\$3F	;chip select stretch register
LDV:	EQU	REGBS+\$40	;PLL loop divider value hi
;LDV	EQU	REGBS+\$41	PLL loop divider value lo
RDV:	EQU	REGBS+\$42	;PLL reference divider value hi
;RDV	EQU	REGBS+\$43	PLL reference divider value lo
CLKCTL:	EQU	REGBS+\$47	;System clock control
PWCLK:	EQU	REGBS+\$40	
PWPOL:	EQU	REGBS+\$41	
PWEN:	EQU	REGBS+\$42	
PWSCAL0:	EQU	REGBS+\$44	
PWPER0:	EQU	REGBS+\$4C	
PWDY0:	EQU	REGBS+\$50	
PWPER1:	EQU	REGBS+\$4D	
PWDY1:	EQU	REGBS+\$51	
PORTPP:	EQU	REGBS+\$56	
DDRP:	EQU	REGBS+\$57	
ATDCTL0:	EQU	REGBS+\$60	;ADC control 0 (reserved)
ATDCTL1:	EQU	REGBS+\$61	;ADC control 1 (reserved)
ATDCTL2:	EQU	REGBS+\$62	;ADC control 2
ATDCTL3:	EQU	REGBS+\$63	;ADC control 3
ATDCTL4:	EQU	REGBS+\$64	;ADC control 4
ATDCTL5:	EQU	REGBS+\$65	;ADC control 5
ATDSTAT:	EQU	REGBS+\$66	;ADC status register hi
;ATDSTAT	EQU	REGBS+\$67	ADC status register lo
ATDTEST:	EQU	REGBS+\$68	;ADC test (reserved)
;ATDTEST	EQU	REGBS+\$69	

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף מוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



PORTAD:	EQU	REGBS+\$6F	;port ADC = input only
ADR0H:	EQU	REGBS+\$70	;ADC result 0 register
ADR0L:	EQU	REGBS+\$71	;ADC result 0 register LOW
ADR1H:	EQU	REGBS+\$72	;ADC result 1 register
ADR2H:	EQU	REGBS+\$74	;ADC result 2 register
ADR3H:	EQU	REGBS+\$76	;ADC result 3 register
ADR4H:	EQU	REGBS+\$78	;ADC result 4 register
ADR5H:	EQU	REGBS+\$7A	;ADC result 5 register
ADR6H:	EQU	REGBS+\$7C	;ADC result 6 register
ADR7H:	EQU	REGBS+\$7E	;ADC result 7 register
TIOS:	EQU	REGBS+\$80	;timer input/output select
CFORC:	EQU	REGBS+\$81	;timer compare force
OC7M:	EQU	REGBS+\$82	;timer output compare 7 mask
OC7D:	EQU	REGBS+\$83	;timer output compare 7 data
TCNT:	EQU	REGBS+\$84	;timer counter register hi
;TCNT	EQU	REGBS+\$85	timer counter register lo
TSCR:	EQU	REGBS+\$86	;timer system control register
TQCR:	EQU	REGBS+\$87	;reserved
TCTL1:	EQU	REGBS+\$88	;timer control register 1
TCTL2:	EQU	REGBS+\$89	;timer control register 2
TCTL3:	EQU	REGBS+\$8A	;timer control register 3
TCTL4:	EQU	REGBS+\$8B	;timer control register 4
TMSK1:	EQU	REGBS+\$8C	;timer interrupt mask 1
TMSK2:	EQU	REGBS+\$8D	;timer interrupt mask 2
TFLG1:	EQU	REGBS+\$8E	;timer flags 1
TFLG2:	EQU	REGBS+\$8F	;timer flags 2
TC0:	EQU	REGBS+\$90	;timer capture/compare register 0
;TC0	EQU	REGBS+\$91	
TC1:	EQU	REGBS+\$92	;timer capture/compare register 1
;TC1	EQU	REGBS+\$93	
TC2:	EQU	REGBS+\$94	;timer capture/compare register 2
;TC2	EQU	REGBS+\$95	
TC3:	EQU	REGBS+\$96	;timer capture/compare register 3
;TC3	EQU	REGBS+\$97	
TC4:	EQU	REGBS+\$98	;timer capture/compare register 4
;TC4	EQU	REGBS+\$99	
TC5:	EQU	REGBS+\$9A	;timer capture/compare register 5
;TC5	EQU	REGBS+\$9B	
TC6:	EQU	REGBS+\$9C	;timer capture/compare register 6
;TC6	EQU	REGBS+\$9D	
TC7:	EQU	REGBS+\$9E	;timer capture/compare register 7
;TC7	EQU	REGBS+\$9F	
PACTL:	EQU	REGBS+\$A0	;pulse accumulator controls
PAFLG:	EQU	REGBS+\$A1	;pulse accumulator flags
PACNT:	EQU	REGBS+\$A2	;pulse accumulator counter
;PACNT	EQU	REGBS+\$A3	
TIMTST:	EQU	REGBS+\$AD	;timer test register
PORTT:	EQU	REGBS+\$AE	;port T = Timer port
DDRT:	EQU	REGBS+\$AF	;port T direction register
SC0BDH:	EQU	REGBS+\$C0	;sci 0 baud reg hi byte
SC0BDL:	EQU	REGBS+\$C1	;sci 0 baud reg lo byte
SC0CR1:	EQU	REGBS+\$C2	;sci 0 control1 reg
SC0CR2:	EQU	REGBS+\$C3	;sci 0 control2 reg
SC0SR1:	EQU	REGBS+\$C4	;sci 0 status reg 1
SC0SR2:	EQU	REGBS+\$C5	;sci 0 status reg 2
SC0DRH:	EQU	REGBS+\$C6	;sci 0 data reg hi
SC0DRL:	EQU	REGBS+\$C7	;sci 0 data reg lo
SC1BDH:	EQU	REGBS+\$C8	;sci 1 baud reg hi byte

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף מוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
SC1BDL: EQU REGBS+$C9 ;sci 1 baud reg lo byte
SC1CR1: EQU REGBS+$CA ;sci 1 control1 reg
SC1CR2: EQU REGBS+$CB ;sci 1 control2 reg
SC1SR1: EQU REGBS+$CC ;sci 1 status reg 1
SC1SR2: EQU REGBS+$CD ;sci 1 status reg 2
SC1DRH: EQU REGBS+$CE ;sci 1 data reg hi
SC1DRL: EQU REGBS+$CF ;sci 1 data reg lo
SP0CR1: EQU REGBS+$D0 ;spi 0 control1 reg
SP0CR2: EQU REGBS+$D1 ;spi 0 control2 reg
SP0BR: EQU REGBS+$D2 ;spi 0 baud reg
SP0SR: EQU REGBS+$D3 ;spi 0 status reg hi
SP0DR: EQU REGBS+$D5 ;spi 0 data reg
PORTS: EQU REGBS+$D6 ;port S = Serial port
DDRS: EQU REGBS+$D7 ;port S direction register
EEMCR: EQU REGBS+$F0 ;EEProm mode control
EEPROT: EQU REGBS+$F1 ;EEProm block protect reg
EETST: EQU REGBS+$F2 ;EEProm test register
EEPORG: EQU REGBS+$F3 ;EEProm program reg
PDLCL: EQU REGBS+$FE ;PortDLC
DDRDLCL: EQU REGBS+$FF ;Port DLC Data Direction Control

INT_CH0: EQU $0B2E
INT_CH1: EQU $0B2C
INT_CH2: EQU $0B2A
INT_CH3: EQU $0B28
INT_OVF: EQU $0B1E

;
;
;
;
;
;*****
;* memory table *
;*****
USFLG: EQU DSTREE+$00
USDNW: EQU DSTREE+$01
USUPW: EQU DSTREE+$02
US_READ: EQU DSTREE+$03 ;A SAVED PLACE 4 SENS READ
;US_READ: EQU DSTREE+$04 ;16BIT
DSIGN: EQU DSTREE+$05 ;FOR ABS THE SIGN THAT D HAD
NTSTRT: EQU DSTREE+$06 ;FOR TIKUN GOINT STRAIGHT OR NOT
TEMP16B2: EQU DSTREE+$07 ;FOR ONE TIME USE
;TEMP16B2: EQU DSTREE+$08
ENC_MONE: EQU DSTREE+$09 ;Encoder Mone
TEMP16B1: EQU DSTREE+$0A ;FOR ONE TIME USE
;TEMP16B1: EQU DSTREE+$0B
USAVG: EQU DSTREE+$0C
;USAVG: EQU DSTREE+$0D
TKN_DUTY: EQU DSTREE+$0E
ENC_PULS: EQU DSTREE+$0F
ENC_FLAG: EQU DSTREE+$10
TEMP8B1: EQU DSTREE+$11
TEMP8B2: EQU DSTREE+$12
DSIGN2: EQU DSTREE+$13 ;ANOTHER SIGN FOR D FOR TIKKUNIM
DIF_TIK: EQU DSTREE+$14
;DIF_TIK: EQU DSTREE+$15
WLL_TIK: EQU DSTREE+$16
;WLL_TIK: EQU DSTREE+$17
US_INIT: EQU DSTREE+$18
AVG1: EQU DSTREE+$19
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



AVG2:	EQU	DSTREE+\$1A
AVG3:	EQU	DSTREE+\$1B
AVG4:	EQU	DSTREE+\$1C
AVG5:	EQU	DSTREE+\$1D
AVG6:	EQU	DSTREE+\$1E
AVG7:	EQU	DSTREE+\$1F
AVG8:	EQU	DSTREE+\$20
AVG_SUM1:	EQU	DSTREE+\$21
AVG_SUM2:	EQU	DSTREE+\$22
L_MONE:	EQU	DSTREE+\$23
L_MONE2:	EQU	DSTREE+\$24
R_MONE:	EQU	DSTREE+\$25
R_MONE2:	EQU	DSTREE+\$26
WHAT_US:	EQU	DSTREE+\$27
;WHAT_US:	EQU	DSTREE+\$28
OVF_MONE:	EQU	DSTREE+\$29
E_TURN:	EQU	DSTREE+\$2A
;E_TURN:	EQU	DSTREE+\$2B
;WHATCTL	was here	
SECTION:	EQU	DSTREE+\$2C
R_SPEED:	EQU	DSTREE+\$2D
L_SPEED:	EQU	DSTREE+\$2E
W_2_TRN:	EQU	DSTREE+\$2F
IR_AVG:	EQU	DSTREE+\$30
SECTIONS:	EQU	DSTREE+\$31
BCK_FWD:	EQU	DSTREE+\$32
LFT_DUTY:	EQU	DSTREE+\$33
;LFT_DUTY:	EQU	DSTREE+\$34
RGT_DUTY:	EQU	DSTREE+\$35
;RGT_DUTY:	EQU	DSTREE+\$36
BF_4_TKN:	EQU	DSTREE+\$37
MM_4_TK1:	EQU	DSTREE+\$38
;MM_4_TK1:	EQU	DSTREE+\$39
MM_4_TK2:	EQU	DSTREE+\$3A
;MM_4_TK2:	EQU	DSTREE+\$3B
MM_4_WTK:	EQU	DSTREE+\$3C
MM_4_DT:	EQU	DSTREE+\$3D
;MM_4_DT:	EQU	DSTREE+\$3E
DID_TURN:	EQU	DSTREE+\$3F
SECT_J:	EQU	DSTREE+\$40
PAS_FLAG:	EQU	DSTREE+\$41
PYRO_FLG:	EQU	DSTREE+\$42
BACK_DIS:	EQU	DSTREE+\$43
FRNT_DIS:	EQU	DSTREE+\$44
DIFFER:	EQU	DSTREE+\$45
WR_2_TKN:	EQU	DSTREE+\$46
PWD4TKN:	EQU	DSTREE+\$47
UV_FLAG:	EQU	DSTREE+\$48
WALL:	EQU	DSTREE+\$49
TURNWEAK:	EQU	DSTREE+\$4A
ERR_NOWL:	EQU	DSTREE+\$4B
ERR_NOWR:	EQU	DSTREE+\$4C
ERR_LSTL:	EQU	DSTREE+\$4D
ERR_LSTR:	EQU	DSTREE+\$4E
D4PID_R:	EQU	DSTREE+\$4F
D4PID_L:	EQU	DSTREE+\$50
MAKOM_C:	EQU	DSTREE+\$51
ERR_SUML:	EQU	DSTREE+\$52

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
ERR_SUMR: EQU DSTREE+$53
SPID4TKN: EQU DSTREE+$54
PLZ_TURN: EQU DSTREE+$55
I4PID_R: EQU DSTREE+$56
I4PID_L: EQU DSTREE+$57
TKN_FLG: EQU DSTREE+$58
TCTLPLCE: EQU DSTREE+$59
CNDL_OFF: EQU DSTREE+$5A
AKOV_SIDE: EQU DSTREE+$5B
ROOM: EQU DSTREE+$5C
MAKM_MONE: EQU DSTREE+$5D
WHO_SENT: EQU DSTREE+$5E
;*****
;* const table *
;*****
DUTY_MAX: EQU $F6
DUTY_MIN: EQU $40
MAX_SPID: EQU $88
STRT_TRN: EQU $600
PWDREG: EQU $E6 ;ROBOT DO 20τ PER SEC
PWDTRNW: EQU $09
PWDTRNS: EQU $E6
ENCTURN: EQU 12800 ;THE ENC READ FOR A TURN (BIG R)
ENCTURNS: EQU 3000 ;THE ENC READ FOR A TURN (BIG R)
ENC180T: EQU 19500
;ENC_PUL0: EQU $02
ENC_PUL0: EQU $F2
ENC_FLG0: EQU $01
;ENC_PUL1 EQU $08
ENC_PUL1 EQU $F8
ENC_FLG1: EQU $02
US_AVG_R: EQU $03 ; Number of reads to memuza in US
TRAKLINM: EQU $210 ; Middle of misdaron
TEN_CM: EQU $5000
TRN_1_RM: EQU $9C0 ;DISTANE FROM WHICH TO TURN TO FIRST ROOM
WALL_AKOV: EQU $32 ; Wall For Akiva ; 30
PWD_AKOV: EQU $B8 ; Duty For Akiva
REG_SPID: EQU 135
MAKOM_OVF: EQU $20 ; Numbers of tikum bamakom to do rescue
```

בעלי אתר הרובוטיקה הישראלי לא יסאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Fire.asm

```
TURN_OFF:      JSR      L_BRAKE
                LDAA     PDLC
                ORAA     #$40
                STAA     PDLC
                LDX      #$FFFF
                LDY      #$15
                JSR      DLOOP
                JSR      L_BRAKE
;TURN_OFF2:    MOVB     #$03,PWEN
;              MOVB     #$A0,PWDTY0
;              MOVB     #$A0,PWDTY1
;              MOVB     #$43,PORTPP
;              LDX      #$8000      ; =\
;              JSR      L_ENCO      ; ==>  CHNG TO UV OFF
;              JSR      PLS_CNT    ; =/
TURN_OFF3:    LDAA     PDLC
                ANDA     #$BF
                STAA     PDLC
                JSR      M_BRAKE
                ; JSR      RTRN_OM
                SWI

;=====
;              PYRO
;=====
PYRO_CHK:      CLR      PYRO_FLG
                JSR      READ_ATD
                LDAA     ADR6H
                CMPA     #$18
                BLO      CNDL_R
                CMPA     #$E0
                BHI      CNDL_R
                LDAA     ADR7H
                CMPA     #$25
                BLO      CNDL_L
                CMPA     #$C0
                BHI      CNDL_L
                TST      WHO_SENT
                BEQ      WAIT2CND
                JMP      END_INT_F

CNDL_L:        MOVB     #$43,PYRO_FLG
                BRA      GO_CNDL

CNDL_R:        MOVB     #$13,PYRO_FLG

GO_CNDL:      JSR      L_BRAKE
                MOVB     #$03,PWEN
                LDAA     PYRO_FLG
                STAA     PORTPP
                MOVB     #$A0,PWDTY0
                MOVB     #$A0,PWDTY1
                CMPA     #$20
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
PYRO_L:   BMI    PYRO_L
           JSR    L_ENCO
           LDX    #3000
           BRA    PYRO_CONT
PYRO_R:   JSR    R_ENCO
           LDX    #7000
PYRO_CONT: JSR    PLS_CNT
           JSR    L_BRAKE
           MOVB   #$03, PWEN
           MOVB   #$03, PORTPP
           MOVB   #$B0, PWDTY0
           MOVB   #$B0, PWDTY1

WAIT2CND: CLR    WHO_SENT
           JSR    SRCH_PAS
           ;      JSR    WC4WAIT
           BRA    PYRO_CHK
           ; ADD ANTI FURNITURE SUBROUTINE HERE
           BRA    WAIT2CND

WC4WAIT:   ; WALL CHECK 4 WAIT2CND
           JSR    READ_ATD
           LDAA   ADR0H
           CMPA   #$4A
           BPL    HALT
           LDAA   ADR1H
           CMPA   #$4A
           BPL    HALT
           RTS

HALT:      JSR    L_BRAKE
           MOVB   #$03, PWEN
           MOVB   #$53, PORTPP
           MOVB   #$C0, PWDTY0
           MOVB   #$C0, PWDTY1
           JSR    L_ENCO
           LDX    #3000
           JSR    PLS_CNT
           JSR    L_BRAKE
           MOVB   #$03, PWEN
           MOVB   #$03, PORTPP
           MOVB   #$90, PWDTY0
           MOVB   #$90, PWDTY1
           INS
           INS
           JMP    END_INT_F

SRCH_PAS:  ; PSHX
           JSR    PAS_LAV
           TST    PAS_FLAG
           LBNE   TURN_OFF
           JSR    PAS_LAV2
           TST    PAS_FLAG
           LBNE   TURN_OFF
           ; PULX
           RTS
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



EXTINGUISHES

PRE_AKOV:

JSR CHK_ROOM

CNT_PRE_A:

LDX #AKOV_INT
STX INT_OVF
MOVB #\$03, PWEN
JSR L_BRAKE
MOVB #\$03, PWEN
JSR MKM_LR
JSR Q_ENGAGE

AKOV:

JSR CHECKWALL ; CHECK FOR TURNS
JSR CHK_SIDE ; Side Cases for us for akiva
TST DSIGN
BNE AKOV_R

AKOV_L:

MOVB #\$00, AKOV_SIDE
JSR US1
LDX US_READ
CPX #\$120
LBMI KAROV
CPX #\$200
LBPL RACHOK
JSR US0
LDX US_READ
JSR US1
LDY US_READ
JSR DECYD
JSR ABSD
CPD #\$30
LBMI DONT_TKN
TST DSIGN
LBEQ TAKN_L
LBRA TAKN_R

AKOV_R:

MOVB #\$FF, AKOV_SIDE
JSR US3
LDX US_READ
CPX #\$120
LBMI RACHOK
CPX #\$200
LBPL KAROV
JSR US2
LDX US_READ
JSR US3
LDY US_READ
JSR DECYD
JSR ABSD
CPD #\$30
LBMI DONT_TKN
TST DSIGN
LBEQ TAKN_R
LBRA TAKN_L

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



DONT_TKN:

```
MOVB  #$03, PORTPP
MOVB  #PWD_AKOV, PWDTY0
MOVB  #PWD_AKOV, PWDTY1
LBRA  AKOV
```

TAKN_L:

```
MOVB  #$80, PWDTY1
MOVB  #PWD_AKOV, PWDTY0
LBRA  AKOV
```

TAKN_R:

```
MOVB  #$80, PWDTY0
MOVB  #PWD_AKOV, PWDTY1
LBRA  AKOV
```

KAROV:

```
MOVB  #$80, PWDTY0
MOVB  #PWD_AKOV, PWDTY1
LBRA  AKOV
```

RACHOK:

```
MOVB  #$80, PWDTY1
MOVB  #PWD_AKOV, PWDTY0
LBRA  AKOV
```

CHECKWALL:

```
JSR    READ_ATD
LDAA   ADR0H
CMPA   #WALL_AKOV
BPL    SEE_WALL
LDAA   ADR1H
CMPA   #WALL_AKOV
BPL    SEE_WALL
RTS
```

SEE_WALL:

```
JSR    L_BRAKE
JSR    CHK_SIDE
TST    DSIGN
BNE    T_LFT_RM
```

left...)

; CHANGE LATER (right and then

T_RGT_RM:

```
MOVB  #$03, PWEN
MOVB  #$13, PORTPP
MOVB  #$A0, PWDTY0
MOVB  #$A0, PWDTY1
JSR    L_ENCO
LDX    #6000
BRA    END_TRNR
```

T_LFT_RM:

```
MOVB  #$03, PWEN
MOVB  #$43, PORTPP
MOVB  #$A0, PWDTY1 ; #PWDTRNS, PWDTY0
MOVB  #$A0, PWDTY0
JSR    R_ENCO
LDX    #5000
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```

END_TRNR:      ; ENCODETS COUNTER WITH WHITE LINE BUILDIN
               MOVB   TCTL4, TCTLPLCE
               MOVB   ENC_PULS, TCTL4
               MOVB   ENC_FLAG, TFLG1
C_P_CNTR:      LDAA   TFLG1
               ANDA   ENC_FLAG
               BEQ    C_P_CNTR
               MOVB   ENC_FLAG, TFLG1
               DEX
               LBNE   C_P_CNTR
               LDAA   PORTPP      ; Brake
               EORA   #$A0        ; Brake
               STAA   PORTPP      ; Brake
               LDX    #$9FFF      ; Brake
               JSR    LOOP        ; Brake
               MOVB   #$03, PORTPP
               MOVB   #PWD_AKOV, PWDTY0
               MOVB   #PWD_AKOV, PWDTY1
               MOVB   TCTLPLCE, TCTL4
MKM_LR:        TST    AKOV_SIDE
               BNE    TKN_MKM_R
TKN_MKM_L:     JSR    MAKOM_L      ; Check for left or right tikun
               BRA    END_TRNR2
TKN_MKM_R:     JSR    MAKOM_R
END_TRNR2:     MOVB   #$03, PWEN
               RTS

CHK_SIDE:      JSR    US2
               LDY    US_READ
               JSR    US0
               LDX    US_READ
               JSR    DECYD
               JSR    ABSD
               RTS

CHK_WLL_BCK:   ; CECHK WALL 4 GOING BACK
               JSR    US2
               CPX    #$170
               BLO    CLR_WLL
               MOVB   #$D0, PWDTY1
               RTS
CLR_WLL:       MOVB   #$60, PWDTY1
               RTS

CHK_ROOM:      TST    R00M
               BEQ    ROOM1
               LDAA   R00M
               CMPA   #$1
               LBEQ   ROOM2
               CMPA   #$2
               LBEQ   ROOM3
               RTS

ROOM1:         MOVB   #$03, PWEN
               JSR    L_BRAKE

```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



	MOVB	#\$03, PWEN	
	JSR	MAKOM_R	
	MOVB	#\$03, PWEN	
	MOVB	#\$53, PORTPP	
	MOVB	#\$D0, PWDY0	
	MOVB	#\$D0, PWDY1	
D00P:	JSR	CHK_WLL_BCK	
	JSR	US1	
	LDX	US_READ	
	CPX	#\$400	
	BPL	D00P	
D00P2:	JSR	CHK_WLL_BCK	
	JSR	US0	
	LDX	US_READ	
	CPX	#\$400	
	BPL	D00P2	
	JSR	L_BRAKE	
	MOVB	#\$03, PWEN	
	JSR	MAKOM_R	
	JSR	Q_ENGAGE	
D00P2B:	JSR	US1	
	LDX	US_READ	
	CPX	#\$400	
	BMI	D00P2B	
	LDX	#1450	
	JSR	L_ENCO	
	JSR	PLS_CNT	
	JSR	L_BRAKE	
	MOVB	#\$03, PWEN	
	JSR	MAKOM_R	
	MOVB	#\$03, PWEN	
	MOVW	#9420, E_TURN	
	JSR	TURN_LEFT_B	
	JSR	L_BRAKE	; Do Turn ;M
	JSR	MOVBYTES	
D00P3:	JSR	PAS_LAV	
	TST	PAS_FLAG	
	BEQ	D00P3	
	LDX	#\$200	
	JSR	PLS_CNT	
	RTS		
ROOM3:			
	MOVB	#\$03, PWEN	
	MOVB	#\$13, PORTPP	
	MOVB	#\$B8, PWDY1	
	MOVB	#\$B8, PWDY0	
	LDX	#4000	; Turn 180 Degrees
	JSR	L_ENCO	
	JSR	PLS_CNT	
	JSR	BRK_TRNB	
	JSR	L_BRAKE	
	MOVB	#\$03, PWEN	
	MOVB	#\$03, PORTPP	
	MOVB	#\$B8, PWDY1	

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
ROOM3_1:  MOVB  #B8, PWDY0
           JSR   PAS_LAV           ; Check For Pas Lavan
           TST   PAS_FLAG
           BEQ   ROOM3_1
           LDX   #$300
           JSR   PLS_CNT
           RTS

ROOM2:     MOVB  #$03, PWEN
           MOVB  #$43, PORTPP
           MOVB  #$A0, PWDY1
           MOVB  #$A0, PWDY0
           LDX   #4000           ; Turn 90 Degrees
           JSR   L_ENCO
           JSR   PLS_CNT
           JSR   BRK_TRNB
           JSR   L_BRAKE
           MOVB  #$03, PWEN
           RTS
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Helpfunc.asm

```
D_INT0_A:
    LSLD      ; - |
    LSLD      ; | =\ PUT 16 BIT D INTO
    LSLD      ; | =/ 8 BIT A
    LSLD      ; - |
    RTS

DECXYD: STY TEMP16B1
        STX TEMP16B2
        LDD TEMP16B2
        SUBD TEMP16B1
        RTS

ABSA:
        TSTA
        BPL ENDABSA
        NEGA
ENDABSA: RTS

ABSD:
        MOVB #$FF, DSIGN
        CPD  #$0
        BPL ENDABSD
        MOVB #$00, DSIGN
        STD  TEMP16B1
        LDD  #$0000
        SUBD TEMP16B1
ENDABSD: RTS

LOOP:   DEX
        BNE LOOP
        RTS

DLOOP:  STX TEMP16B1
ILOOP:  LDX TEMP16B1
        JSR LOOP
        DEY
        BNE ILOOP
        RTS

ABS_A:
        MOVB #$FF, WR_2_TKN
        TSTA
        BPL END_ABSA
        NEGA
        CLR  WR_2_TKN
END_ABSA: RTS

PRIMA:
        MOVB #$03, PORTPP
        MOVB #$13, PORTPP
        MOVB #$B0, PWDY1
        MOVB #$B0, PWDY0
        SWI
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Inits.asm

```
*****
; Program starts here *
*****

*****
;
; *      RAM      *
; *****
;
;      ORG      RAMSTRT

INBUFF      RMB      IBUFSIZ ; input buffer, defined but not used
ENDBUFF      EQU      *
COUNT      RMB      1      ;# characters read, also unused

*****

;
;      ORG      PRGSTRT
;      ORG      MONSTRT
;      LDS      #STACK ; initialize the stack pointer (don't do this
; if under monitor)
START:
;      JSR      COP_RESET      ; Disable watch-dog timer
;      JSR      AD_INIT
;      JSR      TIMER_INI
;      JSR      PWM_INI
;      JSR      PDLC_INI
;      JSR      IC_INIT
;      JSR      INT_INIT
;      JSR      PROP_OFF
;      JSR      UV_INIT
;      JSR      PROGRAM
;      SWI
;      RTS

*****
; End of main loop
*****

;
;      Start Of Initializations
;

COP_RESET:
;      MOVB      #$00,COPCTL ; Turn OFF Cop timer.
;      RTS

PDLC_INI:
;      MOVB      #$DF,DDRDLCL
;      RTS

IC_INIT:
;      MOVB      #$00,TIOS
;      MOVB      #$80,TSCR
;      MOVB      #$05,TCTL4
;      LDAA      TCTL4 ; Pas Lavan Interrupt Init
;      ORAA      #$10 ; Pas Lavan Interrupt Init
;      STAA      TCTL4 ; Pas Lavan Interrupt Init
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
MOV B #FFF, TFLG1
RTS

PWM_INI:
MOV B #FFF, DDRP
MOV B #00, PORTPP
MOV B #08, PWCLK ;20 KHz period
MOV B #0F, PWPOL
MOV B #FF, PWPER0 ; Period - 20 KHz and 5 KHz.
MOV B #PWDREG, PWDTY0 ; Duty - for 20 KHz and 5 KHz.
MOV B #FF, PWPER1 ; Period - Period - 20 KHz and 5 KHz.
MOV B #PWDREG, PWDTY1 ; Duty - Half 20 KHz and 5 KHz.
MOV B #00, PWEN
RTS

AD_INID: LDA A #C8
AD_DELAY: DECA
BNE AD_DELAY
RTS

AD_INIT: MOV B #80, ATDCTL2
JSR AD_INID
MOV B #00, ATDCTL3
MOV B #01, ATDCTL4
RTS

TIMER_INI: MOV B #80, TSCR
RTS

INT_INIT:
MOV B #00, TMSK1
; MOV B #0C, TMSK1
MOV B #80, TMSK2
; LDX #R_ENC
; STX INT_CH0
; LDX #L_ENC
; STX INT_CH1
LDX #UV_IS_GAY
STX INT_CH2
LDX #UV_IS_GAY
STX INT_CH3
LDX #T_OVF
STX INT_OVF
; CLR R_MONE
; CLR R_MONE2
; CLR L_MONE
; CLR L_MONE2
; CLR OVF_MONE
; CLR ERR_LSTL
; CLR ERR_LSTR
; CLR ERR_SUML
; CLR ERR_SUMR
MOV B #0C, TFLG1
MOV B #80, TFLG2
SEI
RTS

PROP_OFF:
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שיגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
LDAA PDLC
ANDAA #$BF
STAA PDLC
RTS
```

UV_INIT:

```
LDAA TCTL4
ORAA #$F0
STAA TCTL4
MOVB #$0C, TFLG1
CLR UV_FLAG
RTS
```

```
;;
;; End Of Initializations
;;
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Ir_ovf.asm

```
T_OVF:
    PSHD
    PSHX
    PSHY
    PSHC

SIUM:
CHK_WALL:                                ; Check for front wall
    CLRA
    CLRB
    JSR    READ_ATD    ;MEMOTZA 4 READS FOR RIGHT FRONT

SENSOR
    LDAB    ADR0H
    LDAA    ADR1H
    JSR    READ_ATD
    ADDB    ADR0H
    ADDA    ADR1H
    LSRB
    LSRB
    CMPA    WALL
    BMI     SIOM
    CMPB    WALL
    BMI     SIOM

NEARWALL:
    MOVB    #$FF,PLZ_TURN    ; Make the flag and jump to end
SIOM:
    MOVB    #$80,TFLG2
    PULC
    PULY
    PULX
    PULD
    RTI

UV_IS_GAY:
    MOVB    #$FF,UV_FLAG
    MOVB    #$0C,TFLG1
    RTI

AKOV_INT:
    PSHX
    PSHD
    PSHC
    MOVB    #$DD,WHO_SENT
    CLR     PAS_FLAG
    LDAA    PDLC
    ORAA    #$DF
    INCA
    LBNE    TURN_OFF
    LDAA    PDLC
    ORAA    #$EF
    INCA
    LBNE    TURN_OFF
    JMP     PYRO_CHK
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



END_INT_F:

```
PULC
PULD
PULX
MOVB    #$80,TFLG2
RTI
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Motors.asm

```
Q_ENGAGE:
    MOVB    #$03, PWEN
    MOVB    #$03, PORTPP
    MOVB    #PWDREG, PWDTY1
    MOVB    #PWDREG, PWDTY0
    RTS

ENGAGE:
    JSR     Q_ENGAGE
    LDX     #$FFFF
    LDY     #$7
    JSR     DLOOP
    RTS

ENG_TRY:
    MOVB    #$03, PWEN
PRENG_F:
    MOVB    #$81, PORTPP
    MOVB    #PWDREG, PWDTY1
    MOVB    #PWDREG, PWDTY0
    LDX     #$A2FF
    JSR     LOOP
    JSR     ENGAGE
    RTS

W_BRAKE:
    MOVB    #$60, PWDTY1
    MOVB    #$60, PWDTY0
    LDAA    PORTPP
    EORA    #$50
    STAA    PORTPP
    LDX     #$8FFF
    LDY     #$2
    JSR     DLOOP
    MOVB    #$00, PORTPP
    MOVB    #$00, PWEN
    RTS

M_BRAKE:
    MOVB    #$90, PWDTY1
    MOVB    #$90, PWDTY0
    LDAA    PORTPP
    EORA    #$50
    STAA    PORTPP
    LDX     #$FFFF
    JSR     LOOP
    MOVB    #$00, PORTPP
    MOVB    #$00, PWEN
    RTS

VW_BRAKE:
    MOVB    #$00, PWDTY1
    MOVB    #$00, PWDTY0
    MOVB    #$00, PORTPP
    MOVB    #$00, PWEN
    LDX     #$FFFF
    JSR     LOOP
    RTS
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
BRAKE:
    MOVB #$A0, PORTPP
    LDX  #$4
    LDY  #$FFF
    JSR  DLOOP      ; Delay for brk
    MOVB #$00, PWEN
    RTS

L_BRAKE:
    MOVB #$A0, PORTPP
    LDX  #$A
    LDY  #$FFF
    JSR  DLOOP      ; Delay for brk
    MOVB #$00, PWEN
    RTS

L_ENCO:
    PSHA
    LDAA #ENC_PUL1
    STAA ENC_PULS
    LDAA #ENC_FLG1
    STAA ENC_FLAG
    PULA
    RTS

R_ENCO:
    PSHA
    LDAA #ENC_PUL0
    STAA ENC_PULS
    LDAA #ENC_FLG0
    STAA ENC_FLAG
    PULA
    RTS

TRN_R_F:
    MOVB #$13, PORTPP
    MOVB #PWDTRNS, PWDY1
    MOVB #PWDTRNW, PWDY0
    JSR  L_ENCO
    LDX  E_TURN      ; Load encoders for turn
    BRA  TRN_BEGN

TRN_L_F:
    MOVB #$43, PORTPP
    MOVB #PWDTRNS, PWDY0
    MOVB #PWDTRNW, PWDY1
    JSR  R_ENCO      ; Inits for Right Encoder
    LDX  E_TURN      ; Load encoders for turn
    BRA  TRN_BEGN

TRN_L_B:
    MOVB #$43, PORTPP
    MOVB #PWDTRNS, PWDY0
    MOVB #PWDTRNW, PWDY1
    JSR  L_ENCO
    LDX  E_TURN      ; Load encoders for turn
    BRA  TRN_BEGN

TRN_R_B:
    MOVB #$13, PORTPP
    MOVB #PWDTRNS, PWDY1
    MOVB #PWDTRNW, PWDY0
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```

        JSR    R_ENCO
        LDX    E_TURN          ; Load encoders for turn
        BRA    TRN_BEGN

TRN_BEGN:
        JSR    PLS_CNT

END_TURN:
        JSR    BRK_TURN
        RTS

TURN_180:
        MOVB   #$43, PORTPP
        MOVB   #$05, PWDTY0
        MOVB   #PWDTRNW, PWDTY1
        JSR    L_ENCO
        LDX    #ENC180T
        JSR    PLS_CNT
        BRA    END_TURN

PLS_CNT:
        MOVB   TCTL4, TCTLPLCE
        MOVB   ENC_PULS, TCTL4
        MOVB   ENC_FLAG, TFLG1

C_P_CNT:
        LDAA   TFLG1
        ANDA   ENC_FLAG
        BEQ    C_P_CNT
        MOVB   ENC_FLAG, TFLG1
        DEX
        BNE    C_P_CNT
        MOVB   TCTLPLCE, TCTL4
        RTS

TRN_R_FS:
        MOVB   #$13, PORTPP
        MOVB   #PWDTRNW, PWDTY0
        MOVB   #PWDTRNW, PWDTY1
        JSR    L_ENCO
        LDX    #ENCTURNS
        JSR    PLS_CNT
        MOVB   #$13, PORTPP
        MOVB   #90, PWDTY0
        MOVB   #90, PWDTY1
        LDX    #$AEFF
        LDY    #$2
        JSR    DLOOP
        JSR    MOVBYTES
        RTS

;U_TURN:
;        MOVB   #$43, PORTPP
;        JSR    L_ENCO
;        LDX    #$2200
;        JSR    PLS_CNT
;        MOVB   #$E0, PWDTY0
;        MOVB   #$30, PWDTY1
;        MOVB   #$13, PORTPP
;        LDX    #$AEFF

```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
;          LDY      #$2
;          JSR      DL00P
;          MOVB     #$03, PORTPP
;          MOVB     #$E0, PWDY1
;          MOVB     #$E0, PWDY0
;          RTS
;
;O_TURN:
;          MOVB     #$43, PORTPP
;          JSR      L_ENCO
;          LDX      #$4000
;          JSR      PLS_CNT
;          RTS

X_L:      MOVB     #$03, PWEN
          MOVB     #$03, PORTPP
          MOVB     #$68, PWDY1
          MOVB     #$68, PWDY0
ACC_2:    INC      PWDY0
          INC      PWDY1
          LDX      #$1200
          JSR      LOOP
          LDAA     PWDY1
          CMPA     #$F0
          BLO      ACC_2
          LDX      $FFFF
          LDY      #$4
          JSR      DL00P
          MOVB     #$E0, PWDY1
          MOVB     #$E0, PWDY0
          RTS

TURN_RIGHT_B:
          PSHC
          PSHX
          SEI
          MOVB     #$03, PWEN
          MOVB     #$23, PORTPP
          MOVB     #PWDREG, PWDY1
          MOVB     #$00, PWDY0
          JSR      L_ENCO
          LDX      E_TURN
          JSR      PLS_CNT
          JSR      BRK_TRNB
          CLR      PLZ_TURN
          PULX
          PULC
          RTS

TURN_LEFT_B:
          PSHC
          PSHX
          SEI
          MOVB     #$03, PWEN
          MOVB     #$83, PORTPP
          MOVB     #PWDREG, PWDY0
          MOVB     #$00, PWDY1
          JSR      R_ENCO
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
LDX E_TURN
JSR PLS_CNT
JSR BRK_TRNB
CLR PLZ_TURN
PULX
PULC
RTS

TURN_LEFT:
PSHC
PSHX
SEI
JSR R_ENCO
MOVB #$43, PORTPP
MOVB #$F0, PWDY0
MOVB TURNWEAK, PWDY1
LDX E_TURN ; Load encoders for turn
JSR PLS_CNT
JSR BRK_TURN
CLR PLZ_TURN
PULX
PULC
RTS

TURN_RIGHT:
PSHC
PSHX
SEI
JSR L_ENCO
MOVB #$13, PORTPP
MOVB #$F0, PWDY1
MOVB TURNWEAK, PWDY0
LDX E_TURN ; Load encoders for turn
JSR PLS_CNT
JSR BRK_TURN
CLR PLZ_TURN
PULX
PULC
RTS

BRK_TURN:
PSHA
LDAA PWDY0
LDAB PWDY1
STAB PWDY0
STAA PWDY1
LDAA PORTPP ; Brake
EORA #$50 ; Brake
STAA PORTPP ; Brake
LDY #$2
LDX $AFFF ; Brake WAS FFFF
JSR DLOOP ; Brake
JSR MOVBYTES
PULA
RTS

MOVBYTES:
MOVB #$03, PWEN
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
MOVW    #$03, PORTPP
MOVW    #PWDREG, PWDY1
MOVW    #PWDREG, PWDY0
RTS

BRK_TRNB:
PSHA
LDAA    PWDY0
LDAB    PWDY1
STAB    PWDY0
STAA    PWDY1
LDAA    PORTPP      ; Brake
EORA    #$A0        ; Brake
STAA    PORTPP      ; Brake
LDY     #$2
LDX     #$AFFF      ; Brake WAS FFFF
JSR     DLOOP       ; Brake
JSR     MOVBYTES
PULA
RTS
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



Sensors.asm

```
US0:    MOVB #$40,USUPW
        MOVB #$80,USFLG
        MOVB #$80,USDNW
        MOVB #$08,US_INIT
        JMP  USENG

US1:    MOVB #$10,USUPW
        MOVB #$40,USFLG
        MOVB #$20,USDNW
        MOVB #$04,US_INIT
        JMP  USENG

US2:    MOVB #$04,USUPW
        MOVB #$20,USFLG
        MOVB #$08,USDNW
        MOVB #$08,US_INIT
        JMP  USENG

US3:    MOVB #$01,USUPW
        MOVB #$10,USFLG
        MOVB #$02,USDNW
        MOVB #$04,US_INIT
        JMP  USENG

USENG:
        PSHC
        PSHD
        PSHX
        SEI
PREREAD: LDX  #$0400    ;10 mili seconds delay
PREDAL:  LDAA #$20
PRERD:   DECA
        BNE PRERD
        DEX
        BNE PREDAL

        LDAA PDLC
        ORAA US_INIT
        STAA PDLC
        LDAA #$20
USSHEV:  DECA
        BNE USSHEV

        LDAA PDLC
        LDAB US_INIT
        COMB
        STAB US_INIT
        ANDA US_INIT
        STAA PDLC
        LDX  #$0000

USREAD:
        LDAA TCTL3
```

בעלי אתר הרובוטיקה הישראלי לא ישראו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
ORAA    USUPW
STAA    TCTL3
MOVB    USFLG, TFLG1
SHUV:   LDAA    TFLG1
        ANDA    USFLG
        BEQ     SHUV
        LDAA    TCTL3
        EORA    USUPW
        ORAA    USDNW
        STAA    TCTL3
        MOVB    USFLG, TFLG1
        LDX     #$0000

MONE:   INX
        LDAA    TFLG1
        ANDA    USFLG
        BEQ     MONE
        STX     US_READ
        PULX
        PULD
        PULC
        RTS

;=====
;                               A2D
;=====

READ_ATD:  MOVB  #$50, ATDCTL5
AD_PROC:   BRCLR ATDSTAT $80 AD_PROC;
          RTS

IR_AVG_F:
        PSHD
        JSR     READ_ATD
        LDAA    ADR0H
        LDAB    ADR1H
        JSR     READ_ATD
        ADDA    ADR0H
        ADDB    ADR1H
        ABA
        LSRA
        LSRA
        STAA    IR_AVG
        PULD
        RTS

IR_AVG_B:
        PSHD
        JSR     READ_ATD
        LDAA    ADR4H
        LDAB    ADR5H
        JSR     READ_ATD
        ADDA    ADR4H
        ADDB    ADR5H
        ABA
        LSRA
        LSRA
        STAA    IR_AVG
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



PULD
RTS

PAS LAVAN

PAS_LAV:

PSHD
CLR PAS_FLAG
LDAA PDLC
ORAA #\$DF
INCA

PAS_0: BNE PAS_1
MOVB #\$00, PAS_FLAG

PULD
RTS

PAS_1: MOVB #\$FF, PAS_FLAG
PULD
RTS

PAS_LAV2:

PSHD
CLR PAS_FLAG
LDAA PDLC
ORAA #\$EF
INCA

PAS2_0: BNE PAS2_1
MOVB #\$00, PAS_FLAG

PULD
RTS

PAS2_1: MOVB #\$FF, PAS_FLAG
PULD
RTS

ULTRA VIOLET

UV_ON:

MOVB #\$0C, TFLG1
LDAA TMSK1
ORAA #\$0C
STAA TMSK1
RTS

UV_ON_L:

MOVB #\$08, TFLG1
LDAA TMSK1
ORAA #\$08
STAA TMSK1
RTS

UV_OFF:

MOVB #\$0C, TFLG1
LDAA TMSK1
ANDAA #\$F3
STAA TMSK1

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



RTS

CHECK_UV:

TST UV_FLAG
LBNE PRE_AKOV
RTS

;CHK_UV:

BRCLR TFLG1 \$04 CHK_UV2
BRA YES_UV

;CHK_UV2:

BRCLR TFLG1 \$08 NO_UV
;YES_UV: MOVB #\$FF,UV_FLAG
MOVB #\$0C,TFLG1
RTS

;NO_UV:

MOVB #\$00,UV_FLAG
RTS

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



[Print.asm](#)

```
SPILL:   STAA      $0850

        LDAA      $0850
        RORA
        RORA
        RORA
        RORA
        ANDA      #$0F
        CMPA      #$09
        BPL       ADD1
MORE1:   ADDA      #$30
        STAA      $0851
        LDAA      $0850
        ANDA      #$0F
        CMPA      #$09
        BPL       ADD2
MORE2:   ADDA      #$30
        STAA      $0852
        LDAA      #E0T
        STAA      $0853

        LDX       #$0851
        JSR       OUTSTRG ; send it out serial port
        JSR       OUTCRLF ; output carriage-return

WAITL:   PSHX
        PSHB
        LDAB      #$06
SB0:     LDX       #$FFFF
SB1:     DBNE     X, SB1
        DBNE     B, SB0
        PULB
        PULX

        RTS

ADD1:    ADDA      #$07
        BRA      MORE1

ADD2:    ADDA      #$07
        BRA      MORE2

ENDLESS  BRA      ENDLESS
        RTS

ONSCI:   ldaa      #$34 ; get baud rate constant
        staa      SC0BDL ; store low byte
        clr       SC0BDH ; clear high byte
        ldaa      #$00 ; configure SCI0 control registers
        staa      SC0CR1
        ldaa      #$0C ; enable transmit and receive
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר



```
        staa    SC0CR2
        RTS

; Output string of ASCII bytes starting at x until end of text ($04).
OUTSTRG:
        JSR     OUTCRLF        ; output carriage-return
OUTSTRG0:
        PSHA                    ; save a
OUTSTRG1:
        LDAA    0,X             ; read char into a
        CMPA    #EOT           ; is this end of text?
        BEQ     OUTSTRG3       ; jump if yes
        JSR     OUTPUT         ; output character
        INX                     ; increment pointer
        BRA     OUTSTRG1       ; loop
OUTSTRG3:
        PULA                    ; restore a
        RTS

; Output a Carriage return and a line feed. Returns a = cr.
OUTCRLF:
        LDAA    #$0A           ; get LF
        JSR     OUTPUT         ; send it
        LDAA    #$0D           ; get CR
        JSR     OUTPUT         ; send it
        LDAA    #$00
        JSR     OUTPUT         ; output padding
        LDAA    #$0D
        RTS

; Output A to SCIO
OUTPUT:
OUTSCI2:
        LDAB    SC0SR1        ; read status
        BITB    #$80          ; test Transmit Data Register Empty bit
        BEQ     OUTSCI2       ; loop if TDRE=1
        ANDA    #$7F         ; mask parity
        STAA    SC0DRL        ; send character
        RTS

;*****
; TEXT TABLES *
;*****
MSGIL    FCC    'IN LINE'
        FCB     EOT
MSGTL    FCC    'METAKEN LEFT'
        FCB     EOT
MSGTR    FCC    'METAKEN RIGHT'
        FCB     EOT
MSGFX    FCC    'GOING STRT'
        FCB     EOT
MSGCL    FCC    'CHECK LEFT'
        FCB     EOT
```

בעלי אתר הרובוטיקה הישראלי לא ישאו באחריות כלשהי לכל נזק, כספי או אחר
שייגרם במישורין או בעקיפין משימוש במידע המצוי באתר זה

© כל הזכויות שמורות לאסף פוניס ולגיא יונה
אין להעתיק תכנים מאתר זה ללא רשות בכתב ממנהלי האתר