


```

Dez 05, 14 14:15      C:\Users\Public\ProjectoTRON.as      Page 5/20

;
;      =====
;      EscStrLCD: Rotina para escrita de string no LCD
;      =====

EscStrLCD:      PUSH    R1
;
;      PUSH    R2
;      PUSH    R3
;      MOV     R2, M[SP+6]      ; Apontador para inicio da "string"
;      MOV     R3, M[SP+5]      ; Localizacao do primeiro caracter
CicloStrLCD:    MOV     M[LCD_CURSOR], R3
;      MOV     R1, M[R2]
;      CMP     R1, FIM_TEXTO
;      BR.Z    FimStrLCD
;      MOV     R1, M[R2]
;      MOV     M[LCD_WRITE], R1
;      INC     R2
;      INC     R3
;      BR      CicloStrLCD
FimStrLCD:     POP      R3
;      POP     R2
;      POP     R1
;      RETN    2              ; Actualiza STACK

;
;      =====
;      EscLCD: Rotina para escrita de inteiros no LCD
;      =====

EscLCD:        PUSH    R1
;      PUSH    R2
;      MOV     R1, M[SP+4]
;      MOV     R2, M[SP+5]
;      MOV     M[LCD_CURSOR], R1
;      ADD     R2, 0030h      ;Converte valor para ASCII
;      MOV     M[LCD_WRITE], R2
;      POP     R2
;      POP     R1
;      RETN    2

;
;      =====
;      EscreveLCD: Rotina Principal do EscreveLCD
;      =====

EscreveLCD:    PUSH    TextoLCD
;      PUSH    8000h          ;Primeira linha do LCD
;      CALL    EscStrLCD

;      PUSH    TextoLCD_J
;      PUSH    8010h          ;Segunda linha do LCD
;      CALL    EscStrLCD

;      RET

;=====
;
;
;      MenuDeJogo: Rotina que escreve o Menu de Jogo
;

```

```

Dez 05, 14 14:15      C:\Users\Public\ProjectoTRON.as      Page 6/20

;=====
Flag1:         INC     M[FLAG]      ;Rotina Ligada a Interrupcao 1
;      RTI

;
;      =====
;      ConverteCoordenadas: Recebe uma coordenada e converte
;      numa coordenada para a Tabela de
;      Colisao
;      =====

ConverteCoordenadas:  PUSH    R2
;      PUSH    R3

;      MOV     R1, M[SP+4]
;      SUB     R1, 0110h      ;Localizacao do canto colocado na po
;      sicao(0,0)
;      MOV     R2, 100h
;      DIV     R1, R2

;      MOV     R3, 50d        ;Maior numero de linhas possivel
;      MUL     R1, R3
;      ENI
;      ADD     R3, R2
;      MOV     R1, R3        ;R1 vai ser aproveitado fora da rotina

;      POP     R3
;      POP     R2
;      RETN    1

;
;      =====
;      EscString_Tabela: Semelhante ao EscString, acrescentando
;      o facto deste escrever na Tabela de
;      Colisao
;      =====

EscString_Tabela:  PUSH    R1
;      PUSH    R2
;      PUSH    R3
;      MOV     R2, M[SP+6]      ; Apontador para inicio da "string"
;      MOV     R3, M[SP+5]      ; Localizacao do primeiro caracter
Ciclo_Tabela:    MOV     M[IO_CURSOR], R3
;      PUSH    R3
;      CALL    ConverteCoordenadas
;      INC     M[R1+TabelaJogo] ;Escrita na Tabela de Colisao
;      MOV     R1, M[R2]
;      CMP     R1, FIM_TEXTO
;      BR.Z    FimEsc_Tabela
;      CALL    EscCar          ;Escrita na Janela de Texto
;      INC     R2
;      INC     R3
;      BR      Ciclo_Tabela
FimEsc_Tabela:   POP      R3
;      POP     R2
;      POP     R1
;      RETN    2              ; Actualiza STACK

;
;      =====
;      MenuDeJogo: Rotina que escreve o menu inicial de jogo
;

```

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 7/20
; Com 48 colunas por 20 linhas		
; =====		
MenuDeJogo:	CALL	LimpaJanela
	PUSH	MolduraLimitel ;Desenha Cimo da Moldura
	PUSH	0110h
	CALL	EscString_Tabela
	PUSH	MolduraLimitel ;Desenha Baixo da Moldura
	PUSH	1610h
	CALL	EscString_Tabela
DesenharMoldura:	PUSH	0210h ;Desenha Lado Esquerdo da Moldura
	PUSH	R1
	PUSH	R2
	PUSH	R3
	MOV	R2, MolduraLateral ; Inicio da "string"
	MOV	R3, M[SP+4] ; Localizacao do simbolo
Ciclo_1:	MOV	M[IO_CURSOR], R3
	PUSH	R3
	CALL	ConverteCoordenadas
	INC	M[R1+TabelaJogo]
	MOV	R1, M[R2]
	CMP	R1, FIM_TEXTO
	BR.Z	FimEsc_1
	CALL	EscCar
	ADD	R3, 0100h
	CMP	R3, 1610h
	BR.NZ	Ciclo_1
FimEsc_1:	POP	R3
	POP	R2
	POP	R1
	POP	R0
	PUSH	0241h ;Desenha Lado Direito da Moldura
	PUSH	R1
	PUSH	R2
	PUSH	R3
	MOV	R2, MolduraLateral ; Inicio da "string"
	MOV	R3, M[SP+4] ; Localizacao do simbolo
Ciclo_2:	MOV	M[IO_CURSOR], R3
	PUSH	R3
	CALL	ConverteCoordenadas
	INC	M[R1+TabelaJogo]
	MOV	R1, M[R2]
	CMP	R1, FIM_TEXTO
	BR.Z	FimEsc_2
	CALL	EscCar
	ADD	R3, 0100h
	CMP	R3, 1641h
	BR.NZ	Ciclo_2
FimEsc_2:	POP	R3
	POP	R2
	POP	R1
	POP	R0
	RET	; Actualiza STACK
; =====		
; MenuDeJogo2: Rotina que escreve o menu inicial de jogo		
; Com 48 colunas por 15 linhas		
;		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 8/20
; =====		
MenuDeJogo2:	CALL	LimpaJanela
	PUSH	MolduraLimitel ;Desenha Cimo da Moldura
	PUSH	0310h
	CALL	EscString_Tabela
	PUSH	MolduraLimitel ;Desenha Baixo da Moldura
	PUSH	1310h
	CALL	EscString_Tabela
DesenharMoldura2:	PUSH	0410h ;Desenha Lado Esquerdo da Moldura
	PUSH	R1
	PUSH	R2
	PUSH	R3
	MOV	R2, MolduraLateral ; Inicio da "string"
	MOV	R3, M[SP+4] ; Localizacao do simbolo
Ciclo_1_2:	MOV	M[IO_CURSOR], R3
	PUSH	R3
	CALL	ConverteCoordenadas
	INC	M[R1+TabelaJogo]
	MOV	R1, M[R2]
	CMP	R1, FIM_TEXTO
	BR.Z	FimEsc_1_2
	CALL	EscCar
	ADD	R3, 0100h
	CMP	R3, 1310h
	BR.NZ	Ciclo_1_2
FimEsc_1_2:	POP	R3
	POP	R2
	POP	R1
	POP	R0
	PUSH	0441h ;Desenha Lado Direito da Moldura
	PUSH	R1
	PUSH	R2
	PUSH	R3
	MOV	R2, MolduraLateral ; Inicio da "string"
	MOV	R3, M[SP+4] ; Localizacao do simbolo
Ciclo_2_2:	MOV	M[IO_CURSOR], R3
	PUSH	R3
	CALL	ConverteCoordenadas
	INC	M[R1+TabelaJogo]
	MOV	R1, M[R2]
	CMP	R1, FIM_TEXTO
	BR.Z	FimEsc_2_2
	CALL	EscCar
	ADD	R3, 0100h
	CMP	R3, 1341h
	BR.NZ	Ciclo_2_2
FimEsc_2_2:	POP	R3
	POP	R2
	POP	R1
	POP	R0
	RET	; Actualiza STACK
; =====		
; MenuDeJogo3: Rotina que escreve o menu inicial de jogo		
; Com 32 colunas por 20 linhas		
;		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 9/20
MenuDeJogo3:	CALL LimpaJanela PUSH MolduraLimite2 ;Desenha Cimo da Moldura PUSH 0118h CALL EscString_Tabela PUSH MolduraLimite2 ;Desenha Baixo da Moldura PUSH 1618h CALL EscString_Tabela	
DesenharMoldura3:	PUSH 0218h ;Desenha Lado Esquerdo da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_1_3 CALL EscCar ADD R3, 0100h CMP R3, 1618h BR.NZ Ciclo_1_3 POP R3 POP R2 POP R1 POP R0	
FimEsc_1_3:	PUSH 0239h ;Desenha Lado Direito da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_2_3 CALL EscCar ADD R3, 0100h CMP R3, 1639h BR.NZ Ciclo_2_3 POP R3 POP R2 POP R1 POP R0	
Ciclo_2_3:	PUSH 0239h ;Desenha Lado Direito da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_2_3 CALL EscCar ADD R3, 0100h CMP R3, 1639h BR.NZ Ciclo_2_3 POP R3 POP R2 POP R1 POP R0	
FimEsc_2_3:	RET ; Actualiza STACK	
; ===== ; MenuDeJogo4: Rotina que escreve o menu inicial de jogo ; Com 32 colunas por 15 linhas ; =====		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 10/20
MenuDeJogo4:	CALL LimpaJanela PUSH MolduraLimite2 ;Desenha Cimo da Moldura PUSH 0318h CALL EscString_Tabela PUSH MolduraLimite2 ;Desenha Baixo da Moldura PUSH 1318h CALL EscString_Tabela	
DesenharMoldura4:	PUSH 0418h ;Desenha Lado Esquerdo da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_1_4 CALL EscCar ADD R3, 0100h CMP R3, 1318h BR.NZ Ciclo_1_4 POP R3 POP R2 POP R1 POP R0	
FimEsc_1_4:	PUSH 0439h ;Desenha Lado Direito da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_2_4 CALL EscCar ADD R3, 0100h CMP R3, 1339h BR.NZ Ciclo_2_4 POP R3 POP R2 POP R1 POP R0	
Ciclo_2_4:	PUSH 0439h ;Desenha Lado Direito da Moldura PUSH R1 PUSH R2 PUSH R3 MOV R2, MolduraLateral ; Inicio da "string" MOV R3, M[SP+4] ; Localizacao do simbolo MOV M[IO_CURSOR], R3 PUSH R3 CALL ConverteCoordenadas INC M[R1+TabelaJogo] MOV R1, M[R2] CMP R1, FIM_TEXTO BR.Z FimEsc_2_4 CALL EscCar ADD R3, 0100h CMP R3, 1339h BR.NZ Ciclo_2_4 POP R3 POP R2 POP R1 POP R0	
FimEsc_2_4:	RET ; Actualiza STACK	
; ===== ; IniciaJogo: Rotina onde decorre a accao do Jogo ; =====		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 11/20
<pre> ; ===== ; Rotinas de interrupcoes ; ===== Relogio: MOV R7, 1d MOV M[TIMER_VALUE], R7 MOV M[TIME_CONTROL], R7 INC M[Contador_Nivel] INC M[Contador_Vel] RTI ViraJ1Esquerda: PUSH R1 ;Rotinas que alteram a direccao das par ticulas MOV R1, MOV11 CMP M[MOVI_J1], R1 BR.Z Circula_E1_1 MOV R1, MOV12 CMP M[MOVI_J1], R1 BR.Z Circula_E1_2 MOV R1, MOV13 CMP M[MOVI_J1], R1 BR.Z Circula_E1_3 MOV R1, MOV14 CMP M[MOVI_J1], R1 BR Circula_E1_4 Circula_E1_1: MOV R1, MOV14 BR SaiInt_E1 Circula_E1_2: MOV R1, MOV11 BR SaiInt_E1 Circula_E1_3: MOV R1, MOV12 BR SaiInt_E1 Circula_E1_4: MOV R1, MOV13 SaiInt_E1: MOV M[MOVI_J1], R1 POP R1 RTI ViraJ2Esquerda: PUSH R1 MOV R1, MOV11 CMP M[MOVI_J2], R1 BR.Z Circula_E2_1 MOV R1, MOV12 CMP M[MOVI_J2], R1 BR.Z Circula_E2_2 MOV R1, MOV13 CMP M[MOVI_J2], R1 BR.Z Circula_E2_3 MOV R1, MOV14 CMP M[MOVI_J2], R1 BR Circula_E2_4 Circula_E2_1: MOV R1, MOV14 BR SaiInt_E2 Circula_E2_2: MOV R1, MOV11 BR SaiInt_E2 Circula_E2_3: MOV R1, MOV12 BR SaiInt_E2 Circula_E2_4: MOV R1, MOV13 SaiInt_E2: MOV M[MOVI_J2], R1 POP R1 RTI </pre>		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 12/20
<pre> ViraJ1Direita: PUSH R1 MOV R1, MOV11 CMP M[MOVI_J1], R1 BR.Z Circula_D1_1 MOV R1, MOV12 CMP M[MOVI_J1], R1 BR.Z Circula_D1_2 MOV R1, MOV13 CMP M[MOVI_J1], R1 BR.Z Circula_D1_3 MOV R1, MOV14 CMP M[MOVI_J1], R1 BR Circula_D1_4 Circula_D1_1: MOV R1, MOV12 BR SaiInt_D1 Circula_D1_2: MOV R1, MOV13 BR SaiInt_D1 Circula_D1_3: MOV R1, MOV14 BR SaiInt_D1 Circula_D1_4: MOV R1, MOV11 SaiInt_D1: MOV M[MOVI_J1], R1 POP R1 RTI ViraJ2Direita: PUSH R1 MOV R1, MOV11 CMP M[MOVI_J2], R1 BR.Z Circula_D2_1 MOV R1, MOV12 CMP M[MOVI_J2], R1 BR.Z Circula_D2_2 MOV R1, MOV13 CMP M[MOVI_J2], R1 BR.Z Circula_D2_3 MOV R1, MOV14 CMP M[MOVI_J2], R1 BR Circula_D2_4 Circula_D2_1: MOV R1, MOV12 BR SaiInt_D2 Circula_D2_2: MOV R1, MOV13 BR SaiInt_D2 Circula_D2_3: MOV R1, MOV14 BR SaiInt_D2 Circula_D2_4: MOV R1, MOV11 SaiInt_D2: MOV M[MOVI_J2], R1 POP R1 RTI ; ===== ; ContHex: Rotina que incrementa o contador ; nos quatro display de 7 segmentos. ; ===== ContHex: PUSH R1 PUSH R2 PUSH R3 PUSH R4 PUSH R5 MOV R1, M[Contador_Nivel] </pre>		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 13/20
	<pre> MOV R2, 10d MOV R3, R2 MOV R4, R2 MOV R5, R2 DIV R1, R5 ;As decimas de segundo acabam a ser ignoradas DIV R1, R4 DIV R1, R3 DIV R1, R2 CMP R1, 10h BR.NZ Salto MOV R1, R0 MOV M[Contador_Nivel], R0 Salto: CALL EscCont POP R5 POP R4 POP R3 POP R2 POP R1 RET ; ===== ; EscCont: Rotina que efectua a escrita do contador ; ===== EscCont: PUSH R6 MOV R6, DISPLAY7SEG_1 MOV M[R6], R4 ;FFF0h Unidades de Segundo MOV M[R6+1], R3 ;FFF1h Dezenas de Segundo MOV M[R6+2], R2 ;FFF2h Centenas de Segundo MOV M[R6+3], R1 ;FFF3h Milhares de Segundo POP R6 RET ; ===== ; Confere_Nivel: Rotina que verifica em que nivel de ; jogo nos encontramos e actualiza ; as caracteristicas do jogo dependentes ; do mesmo ; ===== Confere_Nivel: PUSH R2 PUSH R3 PUSH R4 MOV R4, M[Contador_Nivel] Lvl_1_Vel: CMP R4, TIME_NIVEL_2 BR.NN Lvl_2_Vel JMP AcabaConferir ;Caracteristicas ja estavam atribuidas i nicialmente Lvl_2_Vel: CMP R4, TIME_NIVEL_3 BR.NN Lvl_3_Vel MOV R3, NIVEL_2_VEL MOV M[Vel_Actual], R3 MOV R2, LED_2 MOV M[IO_LEDS], R2 JMP AcabaConferir Lvl_3_Vel: CMP R4, TIME_NIVEL_4 </pre>	

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 14/20
	<pre> BR.NN Lvl_4_Vel MOV R3, NIVEL_3_VEL MOV M[Vel_Actual], R3 MOV R2, LED_3 MOV M[IO_LEDS], R2 JMP AcabaConferir Lvl_4_Vel: CMP R4, TIME_NIVEL_5 BR.NN Lvl_5_Vel MOV R3, NIVEL_4_VEL MOV M[Vel_Actual], R3 MOV R2, LED_4 MOV M[IO_LEDS], R2 JMP AcabaConferir Lvl_5_Vel: MOV R3, NIVEL_5_VEL MOV M[Vel_Actual], R3 MOV R2, LED_5 MOV M[IO_LEDS], R2 BR AcabaConferir AcabaConferir: POP R4 POP R3 POP R2 RET ; ===== ; Movimento: Rotina que desencadeia o movimento das particulas ; ===== Movimento: PUSH R2 PUSH R3 PUSH R4 PUSH R5 MOV R4, M[Vel_Actual] MOV R3, M[Contador_Vel] CMP R3, R4 JMP.NZ Sair_Mov ;Caso ainda nao seja para se mover, a con dicao tem de permitir circular dentro do CicloCont MOV M[Contador_Vel], R0 MOV R2, M[MOVI_J1] ;Determinacao das posicoes a ocupar MOV R5, M[MOVI_J2] ADD M[XY_I_J1], R2 ADD M[XY_I_J2], R5 MOV R7, M[XY_I_J1] ;Verifica se vai haver um choque frontal CMP M[XY_I_J2], R7 JMP.Z Empate PUSH M[XY_I_J1] ;Verificacao se as posicoes ja estao ocu padas CALL ConverteCoordenadas CMP M[R1+TabelaJogo], R0 JMP.NZ J1Perde </pre>	

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 15/20
	INC M[R1+TabelaJogo] PUSH M[XY_I_J2] CALL ConverteCoordenadas CMP M[R1+TabelaJogo], R0 JMP.NZ J2Perde INC M[R1+TabelaJogo] PUSH SimboloJ1 ;Escrita dos caracteres nas respectivas po PUSH M[XY_I_J1] CALL EscString PUSH SimboloJ2 PUSH M[XY_I_J2] CALL EscString MOV M[FLAG_Vencedor], R0 BR Sair_Mov J1Perde: PUSH R2 MOV R2, 3 ;Inicialmente e dado como empate MOV M[FLAG_Vencedor], R2 POP R2 ;Verificar se o jogador 2 nao colidiu PUSH M[XY_I_J2] CALL ConverteCoordenadas CMP M[R1+TabelaJogo], R0 BR.NZ Sair_Mov PUSH R2 MOV R2, 2 ;Deixa de ser empate e passa a ser uma vitori a do jogador 2 MOV M[FLAG_Vencedor], R2 POP R2 BR Sair_Mov J2Perde: INC M[FLAG_Vencedor] BR Sair_Mov Empate: PUSH R2 MOV R2, 3 MOV M[FLAG_Vencedor], R2 POP R2 Sair_Mov: POP R5 POP R4 POP R3 POP R2 RET ; ===== ; IniciaJogo: Rotina Principal do Rotina Jogo ; ===== IniciaJogo: MOV R7, INT_MASK_Jogo ;Activa IO, IB, I7, I9 e Interrupca o 15 MOV M[INT_MASK_ADDR], R7 MOV R7, 1d MOV M[TIMER_VALUE], R7 MOV M[TIME_CONTROL], R7 MOV M[FLAG_Vencedor], R0	

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 16/20
	MOV R3, NIVEL_1_VEL ;Caracteristicas do Nivel1 MOV M[Vel_Actual], R3 MOV R2, LED_1 MOV M[IO_LEDS], R2 CicloCont: CALL ContHex CALL Confere_Nivel CALL Movimento PUSH R2 ;Confirmacao da interrupcao de Pause MOV R2, M[IO_SWITCH] AND R2, 1 CMP R2, R0 BR.Z NaoPara Parado: MOV M[TIME_CONTROL], R0 MOV R2, M[IO_SWITCH] AND R2, 1 CMP R2, R0 BR.Z SaiDePause SaiDePause: BR Parado MOV R7, 1d ;Retoma do Jogo MOV M[TIMER_VALUE], R7 MOV M[TIME_CONTROL], R7 NaoPara: POP R2 CMP M[FLAG_Vencedor], R0 BR.NZ Colisao JMP CicloCont Colisao: RET ;===== ; ; MenuTerminado: Rotina que escreve que o jogo terminou e ; pergunta se queremos jogar novamente ; ;===== MenuTerminado: MOV M[TIME_CONTROL], R0 ;Para o contador MOV R7, M[Contador_Nivel] ;Actualiza o tempo maximo CMP M[TEMPO_MAX], R7 BR.NN NaoRegista MOV M[TEMPO_MAX], R7 NaoRegista: MOV R7, INT_MASK_Inicio ;Apenas deixa I1 activo MOV M[INT_MASK_ADDR], R7 PUSH TextoFinal1 ;Escreve as Mensagens PUSH XY_MT_1 CALL EscString PUSH TextoFinal2 PUSH XY_MT_2 CALL EscString RET	

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 17/20
<pre> ;===== ; ; Reinicia: Rotina que limpa tudo o que e necess�rio para se jogar novamente ; ;===== Reinicia: MOV R2, 8000h ;Limpamos a Tabela de Colisao LimparTabela: CMP R2, 844Bh BR.Z TabelaLimpa MOV M[R2], R0 INC R2 BR LimparTabela TabelaLimpa: MOV M[Contador_Nivel], R0 ;Limpamos Variaveis MOV M[Contador_Vel], R0 MOV M[Nivel_Actual], R0 MOV M[Vel_Actual], R0 MOV R2, 1h MOV M[MOVI_J1], R2 MOV R2, -1h MOV M[MOVI_J2], R2 MOV R2, 0B18h MOV M[XY_I_J1], R2 MOV R2, 0B38h MOV M[XY_I_J2], R2 MOV M[IO_LEDS], R0 ;Limpamos Perifiericos MOV M[DISPLAY7SEG_1], R0 MOV M[FFF1h], R0 MOV M[FFF2h], R0 MOV M[FFF3h], R0 RET ;===== ; ; ActualizaLCD: Rotina que actualiza as informacoes do LCD ; ;===== ; ; ===== ; ContHexLCD: Rotina semelhante ao ContHex ; mas que escreve no LCD em vez de no Display ; ===== ContHexLCD: PUSH R1 PUSH R2 PUSH R3 PUSH R4 PUSH R5 MOV R1, M[TEMPO_MAX] MOV R2, 10d MOV R3, R2 MOV R4, R2 MOV R5, R2 DIV R1, R5 DIV R1, R4 </pre>		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 18/20
<pre> DIV R1, R3 DIV R1, R2 CMP R1, 10h BR.NZ SaltoLCD MOV R1, R0 MOV M[TEMPO_MAX], R0 SaltoLCD: CALL EscContLCD POP R5 POP R4 POP R3 POP R2 POP R1 RET ; ===== ; EscContLCD: Rotina que efectua a escrita do TEMPO_MAX no LCD ; ===== EscContLCD: PUSH R4 ;Unidades de Segundo PUSH 800Eh CALL EscLCD PUSH R3 ;Dezenas de Segundo PUSH 800Dh CALL EscLCD PUSH R2 ;Centenas de Segundo PUSH 800Ch CALL EscLCD PUSH R1 ;Milhares de Segundo PUSH 800Bh CALL EscLCD RET ; ===== ; QuemGanhou: Rotina que aumenta a pontuacao de quem ganhou ; ===== QuemGanhou: PUSH R2 PUSH R3 MOV R2, 2 MOV R3, 1 CMP M[FLAG_Vencedor], R2 JMP.Z GanhouJ2 CMP M[FLAG_Vencedor], R3 JMP.Z GanhouJ1 BR JaSeiQuem ;Flag estara a 3 logo nenhuma pontuacao e in crementada GanhouJ2: INC M[P_J2] BR JaSeiQuem GanhouJ1: INC M[P_J1] JaSeiQuem: POP R3 POP R2 RET </pre>		

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 19/20
<pre> ; ; ===== ; ActualizaLCD: Rotina Principal do ActualizaLCD ; ===== </pre>		
ActualizaLCD:	CALL	QuemGanhou
	CALL	ContHexLCD
	PUSH	R1
	PUSH	R2
	PUSH	R3
	PUSH	R4
	MOV	R1, 10d
	MOV	R2, 10d
	MOV	R3, M[P_J1]
	MOV	R4, M[P_J2]
	DIV	R3, R1 ;Separacao de unidades e dezenas
	DIV	R4, R2
	PUSH	R3 ;Escrita das dezenas da pontuacao do jogador 1
	PUSH	8014h
	CALL	EscLCD
	PUSH	R1 ;Escrita das unidades da pontuacao do jogador 1
	PUSH	8015h
	CALL	EscLCD
	PUSH	R4 ;Escrita das dezenas da pontuacao do jogador 2
	PUSH	801Dh
	CALL	EscLCD
	PUSH	R2 ;Escrita das unidades da pontuacao do jogador 2
	PUSH	801Eh
	CALL	EscLCD
	POP	R4
	POP	R3
	POP	R2
	POP	R1
	RET	
<pre> ;===== ; ; Programa Principal ; ;===== </pre>		
Inicio:	MOV	R1, SP_INICIAL
	MOV	SP, R1
	CALL	Reinicia ;Usada para quando nao pretendemos reinic
iar o simulador do p3	CALL	MenuInicial
	CALL	EscreveLCD
	ENI	
ConfereFlag:	CMP	M[FLAG], R0 ;Apenas avanca quando I1 for carregado
	BR.Z	ConfereFlag ;Condicao Comecar a Jogar
	MOV	M[FLAG], R0

Dez 05, 14 14:15	C:\Users\Public\ProjectoTRON.as	Page 20/20
<pre> Jogo: PUSH R2 ;Seleccao de tamanho na Moldura de acord o com os interruptores ligados </pre>		
	MOV	R2, M[IO_SWITCH]
	AND	R2, 110b
	CMP	R2, R0
	BR.Z	Normal
	CMP	R2, 110b
	BR.Z	Col_Lin_Red
	CMP	R2, 10b
	BR.Z	Col_Red
	BR	Lin_Red
Col_Red:	CALL	MenuDeJogo3
	JMP	MolduraDef
Lin_Red:	CALL	MenuDeJogo2
	JMP	MolduraDef
Col_Lin_Red:	CALL	MenuDeJogo4
	JMP	MolduraDef
Normal:	CALL	MenuDeJogo
		;Moldura de Jogo escrita
MolduraDef:	CALL	IniciaJogo
	CALL	MenuTerminado
	CALL	ActualizaLCD
ConfereFlag1:	CMP	M[FLAG], R0 ;Apenas avanca quando I1 for carregado, c
aso contrario o jogo termina aqui	BR.Z	ConfereFlag1 ;Condicao Jogar Novamente
	MOV	M[FLAG], R0
	CALL	Reinicia
	JMP	Jogo
	DSI	;Este final do codigo nunca e alcancado,
apenas servindo como referencia		
Fim:	BR	Fim