

Dec 05, 13 21:27	final_coment1.as	Page 1/27
;Grupo: 38 ; ;Turno: Quinta-Feira 8:00; ; ;Gonçalo Fialho nº79112 ; ;Gonçalo Mendes nº79222 ; ;Luís Borges nº78349 ;		
;_____ ;Temporizador; ;_____ ATIVA_TEMP EQU FFF7h DURACAO EQU FFF6h		
;_____ ; LEDS ; ;_____ LEDS EQU FFF8h		
;_____ ; LCD ; ;_____ LCD_CONTROLO EQU FFF4h LCD_ESCRITA EQU FFF5h LCD_MASK EQU 1000000000000000b		
;Mascaras; SP_INICIAL EQU FDFFh INT_MASK_ADDR EQU FFFAh ;Mascara de interrupções, permite definir os ve tores de interrupção habilitados, um por cada bit da mascara ;  INT_MASK_INICIAL EQU 0000000000000010b INT_MASK EQU 1000110000000101b ;Interrupções activadas;  Mascara_Alea EQU 1000000000010110b		
;_____ ;Dispositivos Janela Texto; ;_____ IO_READ EQU FFFFh IO_WRITE EQU FFFEh IO_CONTROL EQU FFFCh POS_INI EQU 0000h FIM_TEXTO EQU '\$'		
;_____ ; Display ; ;_____ IO_DISPLAY1 EQU FFF0h IO_DISPLAY2 EQU FFF1h IO_DISPLAY3 EQU FFF2h IO_DISPLAY4 EQU FFF3h		
BemVindoPos EQU 0918h ;Posicao da mensagem Bem-Vindo INICIO EQU 001Ah ;Posicao da Primeira Barreira NUM_LINHAS EQU 0018h ;24 linhas		
;MEMORIAS PARA INTERRUPTCOES;		

Dec 05, 13 21:27	final_coment1.as	Page 2/27
Comeca_Jogo EQU 0001h		
Random EQU 0000h ;Constante Random		
;Variaveis;		
BemVindo ORIG 8000h		
BemVindo1 STR 'Bem-Vindo a Corrida de Bicicleta! ',FIM_TEXTO		
Apaga_Bem STR 'Prima o interruptor Il para comecar.',FIM_TEXTO		
XTO ;Este texto apaga as mensagens e a Janela de jogo		
ASTERISCOS STR '***',FIM_TEXTO		
Lim1 STR '+  ',FIM_TEXTO		
Lim2 STR ' +',FIM_TEXTO		
Bicla STR 'o o ',FIM_TEXTO		
ESPACO STR ' ',FIM_TEXTO		
FimdoJogo1 STR 'Fim do Jogo',FIM_TEXTO		
FimdoJogo2 STR 'Prima o interruptor Il para recomecar',FIM_TEXTO		
LCD1 STR 'Distancia:0000m',FIM_TEXTO		
LCD2 STR 'Maximo: m',FIM_TEXTO		
LCDPAUSA1 STR ' PAUSA ',FIM_TEXTO		
LCDPAUSA2 STR ' ',FIM_TEXTO		
Apaga_FimJogo WORD 001Ah ;Posicao da primeira Barreira		
Valor_ASCII_LCD_1 WORD 0030h ; Corresponde ao numero 0		
Valor_ASCII_LCD_2 WORD 0030h		
Valor_ASCII_LCD_3 WORD 0030h		
Valor_ASCII_LCD_4 WORD 0030h		
Ultima_Pontuacao_1 WORD 0030h		
Ultima_Pontuacao_2 WORD 0030h		
Ultima_Pontuacao_3 WORD 0030h		
Ultima_Pontuacao_4 WORD 0030h		
Distancia_Percorrida WORD 0000h		
Distancia_Percorrida_MAX WORD 0000h		
;_____ ; Constantes ; ;_____ Soma2Linhas WORD 0200h Numero1 WORD 0001h Numero_0100h WORD 0100h Numero_002Fh WORD 002Fh Numero_0019h WORD 0019h Numero_1527h WORD 1527h Numero_0018h WORD 0018h Numero_001Ch WORD 001Ch Numero_181Ch WORD 181Ch Numero_0300h WORD 0300h Numero_0004h WORD 0004h Numero_1833h WORD 1833h Numero_9000h WORD 9000h Numero_0030h WORD 0030h Numero_F000h WORD F000h Numero_FF00h WORD FF00h Numero_0008h WORD 0008h Numero_FFF0h WORD FFF0h Numero_0006h WORD 0006h		

Dec 05, 13 21:27	final_coment1.as	Page 3/27
Numero_000Ch	WORD	000Ch
Numero_FFFFh	WORD	FFFFh
Numero_003Ah	WORD	003Ah
Numero_800Dh	WORD	800Dh
Numero_0002h	WORD	0002h
Numero_0003h	WORD	0003h
Numero_0010h	WORD	0010h
Numero_801Dh	WORD	801Dh
Numero_0014h	WORD	0014h
Numero_0012h	WORD	0012h
Numero_0924h	WORD	0924h
Numero_0016h	WORD	0016h
Numero_0005h	WORD	0005h
;_____;		
;Posicoes das Barreiras;		
;_____;		
Inicia_Barreir	WORD	0012h ;Utilizado para verificar quantas barreiras existe
m na janela;		
Primeira_Posic	WORD	0000h
Barreira_1	WORD	0000h
Barreira_2	WORD	0000h
Barreira_3	WORD	0000h
Barreira_4	WORD	0000h
;_____;		
;Displays constantes ;		
;_____;		
Barreiras_Ultr	WORD	0000h
Barreiras_90	WORD	0000h
Barreiras_900	WORD	0000h
Barreiras_9000	WORD	0000h
;_____;		
; Colisoes ,		
;_____;		
PosicColis	WORD	0000h
HouveColisao	WORD	0001h
;_____;		
;Nível de Jogo;		
;_____;		
ValorNivel	WORD	0001h
ValorTemp	WORD	0005h
ValorLeds	WORD	0000h
;_____;		
;Memorias para as interrup;		
;_____;		
Temporizador_Mem	WORD	0001h
Esquerda_Mem	WORD	0001h
Direita_Mem	WORD	0001h
Pausa_Mem	WORD	0001h

Dec 05, 13 21:27	final_coment1.as	Page 4/27
Turbo_Mem	WORD	0001h
;_____;		
;Tabela Interrupções ;		
;_____;		
INT0	ORIG	FE00h
	WORD	Esquerda_INT
INTB	ORIG	FE0Bh
	WORD	Direita_INT
INT1	ORIG	FE01h
	WORD	Comeca_Jogo_INT
INT3	ORIG	FE0Fh
	WORD	Temporizador_INT
INTA	ORIG	FE0Ah
	WORD	Pausa_INT
INT2	ORIG	FE02h
	WORD	Turbo_INT
	ORIG	0000h
	JMP	Inicio
ApagaJanela:	PUSH	R1 ;apaga o campo de jogo
	PUSH	R2
	PUSH	R3
Salto:	MOV	R3,M[Apaga_FimJogo]
	MOV	M[IO_CONTROL],R3
ApagaJanelal:	MOV	R1,Apaga_Bem
	MOV	R2,M[R1]
	CMP	R2,FIM_TEXTO
	BR.Z	AcabouApagar
	MOV	M[IO_WRITE],R2
	INC	R3
	MOV	M[IO_CONTROL],R3
	INC	R1
	BR	ApagaJanelal
AcabouApagar:	ADD	R3,M[Numero_0100h]
	SUB	R3,M[Numero_002Fh]
	MOV	R1,R3
	SHR	R1,8
	CMP	R1,M[Numero_0019h]
	BR.Z	TerminouApag
	JMP	Salto
TerminouApag:	POP	R3
	POP	R2
	POP	R1

Dec 05, 13 21:27	final_coment1.as	Page 5/27
	<b>RET</b>	
Turbo_INT:	<b>CMP</b> M[Turbo_Mem],R0 ;Ativa Turbo <b>BR.NZ</b> AtivaTurbo <b>PUSH</b> R1 <b>MOV</b> R1,M[Numero1] <b>MOV</b> M[Turbo_Mem],R1 <b>POP</b> R1 <b>RTI</b>	
AtivaTurbo:	<b>MOV</b> M[Turbo_Mem],R0 <b>RTI</b>	
Pausa_INT:	<b>CMP</b> M[Pausa_Mem],R0 ; Ativa/Desativa Pausa <b>BR.Z</b> Pausa_a_1  <b>MOV</b> M[Pausa_Mem],R0 <b>BR</b> TerminaPausa	
Pausa_a_1:	<b>INC</b> M[Pausa_Mem]	
TerminaPausa:	<b>RTI</b>	
Comeca_Jogo_INT:	<b>MOV</b> M[Comeca_Jogo],R0 ;Começa jogo <b>RTI</b>	
Temporizador_INT:	<b>MOV</b> M[Temporizador_Mem],R0 ;Temporizador Ativo <b>RTI</b>	
Esquerda_INT:	<b>MOV</b> M[Esquerda_Mem],R0 ; IO Ativo <b>RTI</b>	
Direita_INT:	<b>MOV</b> M[Direita_Mem],R0 ; IB Ativo <b>RTI</b>	
BemVind:	<b>PUSH</b> R1 <b>PUSH</b> R2	
BemVind1:	<b>MOV</b> R1,BemVindo <b>MOV</b> R2,M[R1] ;Escreve 1º Mensagem-Bemvindo <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> MudaLinha <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R6 <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> BemVind1	
MudaLinha:	<b>MOV</b> R6,BemVindoPos ;Espaceamento de 1 linha <b>ADD</b> R6,M[Soma2Linhas] <b>SUB</b> R6,M[Numero1]	

Dec 05, 13 21:27	final_coment1.as	Page 6/27
BemVind2:	<b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,BemVindo1 <b>MOV</b> R2,M[R1] ;Escreve 2º Mensagem-Bemvindo <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> AcabaBemVindo <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R6 <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> BemVind2	
AcabaBemVindo:	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Escreve_Campo:	<b>DSI</b> ;Escreve o Campo de Jogo <b>PUSH</b> R1 <b>PUSH</b> R2 <b>PUSH</b> R3 <b>MOV</b> R2,NUM_LINHAS	
Escreve_Linha:	<b>CALL</b> Escr_Limites ;Escreve uma linha de jogo <b>DEC</b> R2 <b>BR.NZ</b> Escreve_Linha ;Enquanto o campo nao estiver concluido co ntinua a escrever <b>BR</b> Acaba	
Acaba:	<b>POP</b> R3 <b>POP</b> R2 <b>POP</b> R1 <b>CALL</b> EscreveBicla ;Escreve Bicicleta <b>ENI</b> <b>RET</b>	
EscreveBicla:	<b>MOV</b> R6,M[Número_1527h] ;Posição central da bicicleta <b>MOV</b> M[IO_CONTROL],R6  <b>PUSH</b> R1 <b>PUSH</b> R2 <b>MOV</b> R1,Bicla <b>MOV</b> R2,M[R1] <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> EscreveuBicla <b>MOV</b> M[IO_WRITE],R2 <b>ADD</b> R6,M[Número_0100h] ;Proxima Linha <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> EscreveBicla1	
EscreveuBicla:	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Escr_Limites:	<b>MOV</b> R1,Lim1 ;Esta rotina so uma linha os Limites <b>MOV</b> R3,M[R1] ;Escreve Limite Esquerdo	

Dec 05, 13 21:27	final_coment1.as	Page 7/27
	<b>CMP</b> R3,FIM_TEXTO <b>BR.Z</b> Escr_Lim2 <b>MOV</b> M[IO_WRITE],R3 <b>INC</b> R6 <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> Escr_Limites1	
Escr_Lim2:	<b>ADD</b> R6,M[Numero_0018h] ;Tamanho do campo (colunas) <b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,Lim2 <b>MOV</b> R3,M[R1] ;Escreve Limite Esquerdo <b>CMP</b> R3,FIM_TEXTO <b>BR.Z</b> Retorna <b>MOV</b> M[IO_WRITE],R3 <b>INC</b> R6 <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> Escr_Limites2	
Retorna:	<b>SUB</b> R6,M[Numero_001Ch] ;Coloca em R6 a proxima linha qu e deve ser escrita <b>ADD</b> R6,M[Numero_0100h] <b>MOV</b> M[IO_CONTROL],R6 <b>RET</b>	
Esquerda:	<b>PUSH</b> R1 ;Escreve a bicicleta uma posicao para a esquerd a <b>PUSH</b> R2  <b>MOV</b> R1,M[Numero1] <b>MOV</b> M[Esquerda_Mem],R1  <b>CMP</b> R6,M[Numero_181Ch] <b>JMP.Z</b> EscreveuEsquer	
ApagaEsquerda:	 <b>SUB</b> R6,M[Numero_0300h] <b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,ESPACO <b>MOV</b> R2,M[R1] ;Apaga posicao anterior da bicicleta <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> ApagouEsquerda <b>MOV</b> M[IO_WRITE],R2 <b>ADD</b> R6,M[Numero_0100h] <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> ApagaEsquerda	
ApagouEsquerda:	<b>SUB</b> R6,M[Numero_0300h] ; Escreve a nova posicao da bici cleta  <b>SUB</b> R6,1 <b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,Bicla <b>MOV</b> R2,M[R1] <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> EscreveuEsquer	
Esquerda1:	 <b>CALL</b> MovBiclaColis	

Dec 05, 13 21:27	final_coment1.as	Page 8/27
	<b>MOV</b> M[IO_WRITE],R2 <b>ADD</b> R6,M[Numero_0100h] <b>CALL</b> MovBiclaColis ;verifica se existe colisao <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> Esquerda1	
EscreveuEsquer:	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Direita:	 <b>PUSH</b> R1 ;Escreve a bicicleta uma posicao para a direita  <b>PUSH</b> R2  <b>MOV</b> R1,M[Numero1] <b>MOV</b> M[Direita_Mem],R1  <b>CMP</b> R6,M[Numero_1833h] <b>JMP.Z</b> EscreveuDir	
ApagaDir:	 <b>SUB</b> R6,M[Numero_0300h] <b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,ESPACO <b>MOV</b> R2,M[R1] ;Apaga posicao anterior da bicicleta <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> ApagouDir <b>MOV</b> M[IO_WRITE],R2 <b>ADD</b> R6,M[Numero_0100h] <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> ApagaDir	
ApagouDir:	 <b>SUB</b> R6,M[Numero_0300h] <b>ADD</b> R6,1 <b>MOV</b> M[IO_CONTROL],R6 <b>MOV</b> R1,Bicla <b>MOV</b> R2,M[R1] ; Escreve a nova posicao da bicicleta <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> EscreveuDir <b>MOV</b> M[IO_WRITE],R2 <b>ADD</b> R6,M[Numero_0100h] <b>CALL</b> MovBiclaColis ;verifica se existe colisao <b>MOV</b> M[IO_CONTROL],R6 <b>INC</b> R1 <b>BR</b> Direita1	
Direita1:		
EscreveuDir:	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	

Dec 05, 13 21:27	final_coment1.as	Page 9/27
MovBiclaColis:	<b>PUSH</b> R1 ;Verifica todos os locais possíveis de o	
correr colisão	<b>PUSH</b> R2	
	<b>PUSH</b> R3	
;1;		
	<b>MOV</b> R3,M[Barreira_1]	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
	<b>ADD</b> R3,2	
	<b>CMP</b> R3,R2	
	<b>CALL.Z</b> GAMEOVER	
;2;		
	<b>MOV</b> R3,M[Barreira_2]	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
	<b>ADD</b> R3,2	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
;4;		
	<b>MOV</b> R3,M[Barreira_3]	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
	<b>ADD</b> R3,2	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
;4;		
	<b>MOV</b> R3,M[Barreira_3]	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
	<b>ADD</b> R3,2	
	<b>CMP</b> R3,R6	
	<b>CALL.Z</b> GAMEOVER	
	<b>POP</b> R3	
	<b>POP</b> R2	
	<b>POP</b> R1	
	<b>RET</b>	
Apaga_Mensagem:	<b>PUSH</b> R1 ;Apaga as mensagens de BemVindo e Termin	

Dec 05, 13 21:27	final_coment1.as	Page 10/27
a Jogo.	<b>PUSH</b> R2	
	<b>PUSH</b> R3	
	<b>MOV</b> R3,M[Número1]	
	<b>MOV</b> R6,BemVindoPos	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>MOV</b> R1,Apaga_Bem	
Apaga_BemVind:	<b>MOV</b> R2,M[R1] ;Escreve espaços por cima da 1ºm	
ensagem	<b>CMP</b> R2,FIM_TEXTO	
	<b>BR.Z</b> Apaga_MudaLinha	
	<b>MOV</b> M[IO_WRITE],R2	
	<b>INC</b> R6	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>INC</b> R1	
	<b>BR</b> Apaga_BemVind	
Apaga_MudaLinha:	<b>MOV</b> R6,BemVindoPos ;Soma +1 linha	
	<b>ADD</b> R6,M[Soma2Linhas]	
	<b>SUB</b> R6,M[Número1]	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>MOV</b> R1,Apaga_Bem	
	<b>CMP</b> R3,R0	
Apaga_BemVind1:	<b>MOV</b> R2,M[R1] ;Escreve espaços por cima da 2ºm	
ensagem	<b>CMP</b> R2,FIM_TEXTO	
	<b>BR.Z</b> ApagaBemVindo	
	<b>MOV</b> M[IO_WRITE],R2	
	<b>INC</b> R6	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>INC</b> R1	
	<b>BR</b> Apaga_BemVind1	
ApagaBemVindo:	<b>POP</b> R3	
	<b>POP</b> R2	
	<b>POP</b> R1	
	<b>RET</b>	
Soma_Display:	<b>PUSH</b> R1 ; Incrementa o Display de 7 segmentos	
	<b>PUSH</b> R2	
	<b>MOV</b> R2,M[Barreiras_Ultr]	
	<b>INC</b> R2	
	<b>CMP</b> M[Barreiras_90],R0 ;verifica se chegou a 9	
	<b>BR.NZ</b> Naoaumentalvl	
	<b>CMP</b> R2,M[Número_0004h] ;Se passou 4 obs. muda para nive	
12	<b>CALL.Z</b> Nivel2	
ara nivel3	<b>CMP</b> R2,M[Número_0008h] ;Se passou 8 obs. muda p	
	<b>CALL.Z</b> Nivel3	

Dec 05, 13 21:27	final_coment1.as	Page 11/27
Naoaumentalvl:	<b>MOV</b> R2,M[Barreiras_Ultr] <b>SHL</b> R2,12 <b>CMP</b> R2,M[Numero_9000h] ;Verifica se chegou a 9 <b>BR.Z</b> Soma_Display1  <b>MOV</b> R1,IO_DISPLAY1 ; Incrementa Display1 <b>MOV</b> R2,M[Barreiras_Ultr] <b>INC</b> R2 <b>MOV</b> M[Barreiras_Ultr],R2 <b>MOV</b> M[R1],R2  <b>JMP</b> Termina_Display	
Soma_Display1:	<b>MOV</b> R2,M[Barreiras_90] <b>SHL</b> R2,12 <b>CMP</b> R2,M[Numero_9000h] <b>BR.Z</b> Soma_Display2  <b>MOV</b> R1,IO_DISPLAY2 <b>MOV</b> R2,M[Barreiras_90] <b>INC</b> R2 <b>MOV</b> M[Barreiras_90],R2 ;Incrementa Display2 <b>MOV</b> M[R1],R2 <b>MOV</b> R1,IO_DISPLAY1 <b>MOV</b> M[Barreiras_Ultr],R0 ;Coloca Display1 a 0 <b>MOV</b> M[R1],R0 <b>JMP</b> Termina_Display	
Soma_Display2:	<b>MOV</b> R2,M[Barreiras_900] <b>SHL</b> R2,12 <b>CMP</b> R2,M[Numero_9000h] <b>BR.Z</b> Soma_Display3  <b>MOV</b> R1,IO_DISPLAY3 <b>MOV</b> R2,M[Barreiras_900] ;Incrementa Display3 <b>INC</b> R2 <b>MOV</b> M[Barreiras_900],R2 <b>MOV</b> M[R1],R2 <b>MOV</b> R1,IO_DISPLAY2 <b>MOV</b> M[Barreiras_90],R0 ;Coloca Display2 a 0 <b>MOV</b> M[R1],R0 <b>MOV</b> R1,IO_DISPLAY1 <b>MOV</b> M[Barreiras_Ultr],R0 ;Coloca Display1 a 0 <b>MOV</b> M[R1],R0 <b>JMP</b> Termina_Display	
Soma_Display3:	<b>MOV</b> R1,IO_DISPLAY4 <b>MOV</b> R2,M[Barreiras_9000] ;Incrementa Display4 <b>INC</b> R2 <b>MOV</b> M[Barreiras_9000],R2 <b>MOV</b> M[R1],R2  <b>MOV</b> R1,IO_DISPLAY3 <b>MOV</b> M[Barreiras_900],R0 ;Coloca Display3 a 0 <b>MOV</b> M[R1],R0  <b>MOV</b> R1,IO_DISPLAY2 <b>MOV</b> M[Barreiras_90],R0 ;Coloca Display2 a 0 <b>MOV</b> M[R1],R0 <b>MOV</b> R1,IO_DISPLAY1 <b>MOV</b> M[Barreiras_Ultr],R0 ;Coloca Display1 a 0	

Dec 05, 13 21:27	final_coment1.as	Page 12/27
	<b>MOV</b> M[R1],R0	
Termina_Display:	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Nivel2:	<b>PUSH</b> R1 <b>MOV</b> R1,M[Numero_FF00h] ;Coloca nos Leds FF00h correspon	
dente ao lvl2	<b>MOV</b> M[LEDS],R1 <b>MOV</b> M[ValorLeds],R1 <b>INC</b> M[ValorNivel] ;Coloca numa Memoria o Valor do Nivel(2) <b>POP</b> R1 <b>RET</b>	
Nivel3:	<b>PUSH</b> R1 <b>MOV</b> R1,M[Numero_FFF0h];Coloca nos Leds FFF0h correspondente	
ao lvl3	<b>MOV</b> M[LEDS],R1 <b>MOV</b> M[ValorLeds],R1 <b>INC</b> M[ValorNivel];Coloca numa Memoria o Valor do Nivel(3) <b>POP</b> R1 <b>RET</b>	
Alea_Barr1:	<b>PUSH</b> R1 ;Gera um valor aleatorio para a Barreira_1 e guarda o respectivo valor <b>PUSH</b> R2	
	<b>MOV</b> R2,M[Barreira_1] <b>MOV</b> R5,R2 <b>SUB</b> R5,M[Numero_0100h] <b>MOV</b> M[IO_CONTROL],R5	
Apaga_Aleal:	<b>MOV</b> R1,ESPACO <b>MOV</b> R2,M[R1] ;Apaga a Barreira que se encontra na últ	
ima linha do campo	<b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> Apagou_Aleal	
	<b>MOV</b> M[PosicColis],R5 <b>CALL</b> Ver_Colisao <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R5 <b>MOV</b> M[IO_CONTROL],R5 <b>INC</b> R1 <b>BR</b> Apaga_Aleal	
Apagou_Aleal:	<b>CALL</b> Soma_Display ;Soma o Display de 7 segmentos <b>CALL</b> Aleatoriolst ;Gera Aleatório para a Barreira_1 <b>MOV</b> R1,M[Random] <b>ADD</b> R1,M[Numero_001Ch] ;Para ficar na zona da pista som	
a-se 001Ch	<b>MOV</b> M[Barreira_1],R1  <b>POP</b> R2 <b>POP</b> R1	

Dec 05, 13 21:27	final_coment1.as	Page 13/27
	<pre> RET Alea_Barr2:  PUSH  R1      ;Gera um valor aleatorio para a Barreira_2 e guar da o respetivo valor               PUSH  R2                MOV   R2,M[Barreira_2]               MOV   R5,R2               SUB   R5,M[Numero_0100h]               MOV   M[IO_CONTROL],R5                MOV   R1,ESPACO               MOV   R2,M[R1]      ;Apaga a Barreira que se encontra na últ ima linha do campo               CMP   R2,FIM_TEXTO               BR.Z  Apagou_Alea2                MOV   M[PosicColis],R5               CALL  Ver_Colisao               MOV   M[IO_WRITE],R2               INC   R5               MOV   M[IO_CONTROL],R5               INC   R1               BR    Apaga_Alea2  Apagou_Alea2: CALL  Soma_Display      ;Soma o Display de 7 segmentos               CALL  Aleatoriolst      ;Gera Aleatório para a Barreira_2               MOV   R1,M[Random]               ADD   R1,M[Numero_001Ch]               MOV   M[Barreira_2],R1               POP   R2               POP   R1               RET  Alea_Barr3:  PUSH  R1      ;Gera um valor aleatorio para a Barreira_3 e guarda o respetivo valor               PUSH  R2                MOV   R2,M[Barreira_3]               MOV   R5,R2               SUB   R5,M[Numero_0100h]               MOV   M[IO_CONTROL],R5                MOV   R1,ESPACO               MOV   R2,M[R1]      ;Apaga a Barreira que se encontra na últ ima linha do campo               CMP   R2,FIM_TEXTO               BR.Z  Apagou_Alea3                MOV   M[PosicColis],R5               CALL  Ver_Colisao      ;Verifica Colisao               MOV   M[IO_WRITE],R2               INC   R5               MOV   M[IO_CONTROL],R5               INC   R1               BR    Apaga_Alea3  Apagou_Alea3: CALL  Soma_Display      ;Soma o Display de 7 segmentos </pre>	

Dec 05, 13 21:27	final_coment1.as	Page 14/27
	<pre> CALL  Aleatoriolst      ;Gera Aleatório para a Barreira_3 MOV   R1,M[Random] ADD   R1,M[Numero_001Ch] ;Para ficar na zona da pista soma-se  001Ch  MOV   M[Barreira_3],R1 POP   R2 POP   R1 RET  Alea_Barr4:  PUSH  R1      ;Gera um valor aleatorio para a Barreira_4 e guar da o respetivo valor               PUSH  R2                MOV   R2,M[Barreira_4]               MOV   R5,R2               SUB   R5,M[Numero_0100h]               MOV   M[IO_CONTROL],R5                MOV   R1,ESPACO               MOV   R2,M[R1]      ;Apaga a Barreira que se encontra na últ ima linha do campo               CMP   R2,FIM_TEXTO               BR.Z  Apagou_Alea4                MOV   M[PosicColis],R5               CALL  Ver_Colisao               MOV   M[IO_WRITE],R2               INC   R5               MOV   M[IO_CONTROL],R5               INC   R1               BR    Apaga_Alea4  Apagou_Alea4: CALL  Soma_Display      ;Soma o Display de 7 segmentos               CALL  Aleatoriolst      ;Gera Aleatório para a Barreira_4               MOV   R1,M[Random]               ADD   R1,M[Numero_001Ch]               MOV   M[Barreira_4],R1               POP   R2               POP   R1               RET  Temporizador: PUSH  R1               PUSH  R2               PUSH  R5  ira          CMP   M[Inicia_Barreir],R0      ;Verifica se pode criar 4º Barre               BR.Z  Barreira4  3º Barreira  MOV   R1,M[Numero_0006h]          ;Verifica se pode criar                CMP   M[Inicia_Barreir],R1               BR.Z  Barreira3  2º Barreira  MOV   R1,M[Numero_000Ch]          ;Verifica se pode criar                CMP   M[Inicia_Barreir],R1               BR.Z  Barreira2               JMP   EscriBarreiral </pre>	

Dec 05, 13 21:27	final_coment1.as	Page 15/27
Barreira4:	<b>CALL</b> Aleatoriolst ;Só é realizada 1vez <b>MOV</b> R1,M[Random] <b>ADD</b> R1,M[Numero_001Ch] <b>MOV</b> M[Barreira_4],R1 <b>BR</b> EscritBarreiral	
Barreira3:	<b>CALL</b> Aleatoriolst ;Só é realizada 1vez <b>MOV</b> R1,M[Random] <b>ADD</b> R1,M[Numero_001Ch] <b>MOV</b> M[Barreira_3],R1 <b>BR</b> EscritBarreiral	
Barreira2:	<b>CALL</b> Aleatoriolst ;Só é realizada 1vez <b>MOV</b> R1,M[Random] <b>ADD</b> R1,M[Numero_001Ch] <b>MOV</b> M[Barreira_2],R1	
;1 barreira;		
EscritBarreiral:	<b>MOV</b> R5,M[Barreira_1] ;Realiza testes em Barreira_1 e escreve a sua nova posição	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,M[Numero_0018h] ;Se a 1º barreira estiver na ultima linha cria novo valor para a mesma	
	<b>CALL.Z</b> Alea_Barr1	
	<b>MOV</b> R5,M[Barreira_1]	
	<b>MOV</b> M[Primeira_Posic],R5	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,R0	
	<b>CALL.Z</b> EscBarreiral ;Se for 1º posição só escreve e não apaga pos. anterior	
	<b>MOV</b> R1,M[Numero1]	
	<b>MOV</b> M[ATIVA_TEMP],R1	
	<b>CALL</b> ApagaEscreve	
	<b>MOV</b> R5,M[Primeira_Posic]	
	<b>MOV</b> M[Barreira_1],R5	
	<b>CMP</b> M[Barreira_2],R0	
	<b>JMP.Z</b> TerminaTemp	
;2 barreira;		
	<b>MOV</b> R5,M[Barreira_2] ;Realiza testes em Barreira_2 e escreve a sua nova posição	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,M[Numero_0018h] ;Se a 2º barreira estiver na ultima linha cria novo valor para a mesma	
	<b>CALL.Z</b> Alea_Barr2	
	<b>MOV</b> R5,M[Barreira_2]	
	<b>MOV</b> M[Primeira_Posic],R5	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,R0	
	<b>CALL.Z</b> EscBarreiral ;Se for 1º posição só escreve e não apaga pos. anterior	

Dec 05, 13 21:27	final_coment1.as	Page 16/27
não apaga pos. anterior	<b>CALL</b> ApagaEscreve <b>MOV</b> R5,M[Primeira_Posic] <b>MOV</b> M[Barreira_2],R5	
	<b>CMP</b> M[Barreira_3],R0	
	<b>JMP.Z</b> TerminaTemp	
;3 barreira;		
	<b>MOV</b> R5,M[Barreira_3] ;Realiza testes em Barreira_3 e escreve a sua nova posição	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,M[Numero_0018h] ;Se a 3º barreira estiver na ultima linha cria novo valor para a mesma	
	<b>CALL.Z</b> Alea_Barr3	
	<b>MOV</b> R5,M[Barreira_3]	
	<b>MOV</b> M[Primeira_Posic],R5	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,R0	
	<b>CALL.Z</b> EscBarreiral ;Se for 1º posição só escreve e não apaga pos. anterior	
	<b>CALL</b> ApagaEscreve	
	<b>MOV</b> R5,M[Primeira_Posic]	
	<b>MOV</b> M[Barreira_3],R5	
	<b>CMP</b> M[Barreira_4],R0	
	<b>JMP.Z</b> TerminaTemp	
;4 barreira;		
	<b>MOV</b> R5,M[Barreira_4] ;Realiza testes em Barreira_4 e escreve a sua nova posição	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,M[Numero_0018h] ;Se a 4º barreira estiver na ultima linha cria novo valor para a mesma	
	<b>CALL.Z</b> Alea_Barr4	
	<b>MOV</b> R5,M[Barreira_4]	
	<b>MOV</b> M[Primeira_Posic],R5	
	<b>SHR</b> R5,8	
	<b>CMP</b> R5,R0	
	<b>CALL.Z</b> EscBarreiral ;Se for 1º posição só escreve e não apaga pos. anterior	
	<b>CALL</b> ApagaEscreve	
	<b>MOV</b> R5,M[Primeira_Posic]	
	<b>MOV</b> M[Barreira_4],R5	
TerminaTemp:	<b>JMP</b> AcabaTemp	
ApagaEscreve:	<b>PUSH</b> R1 ;Esta rotina Apaga a pos. Anterior da Bicicleta e escreve a nova	
	<b>PUSH</b> R2	
	<b>PUSH</b> R5	
	<b>MOV</b> R5,M[Primeira_Posic]	
	<b>SUB</b> R5,M[Numero_0100h]	
	<b>MOV</b> M[IO_CONTROL],R5	
	<b>MOV</b> R1,ESPACO	
ApagaBarr:	<b>MOV</b> R2,M[R1] ;Apaga a Barreira	



Dec 05, 13 21:27	final_coment1.as	Page 17/27
	<b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> ApagouBarr  <b>MOV</b> M[PosicColis],R5 <b>CALL</b> Ver_Colisao  <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R5 <b>MOV</b> M[IO_CONTROL],R5 <b>INC</b> R1 <b>BR</b> ApagaBarr  ApagouBarr: <b>ADD</b> R5,M[Numero_0100h] <b>SUB</b> R5,3 <b>MOV</b> M[Primeira_Posic],R5  <b>SHR</b> R5,8 <b>CMP</b> R5,M[Numero_0019h] <b>CALL.NZ</b> EscBarreiral ;Escreve a nova Barreira  <b>POP</b> R5 <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>  EscBarreiral: <b>PUSH</b> R1 ; Escreve Barreira; <b>PUSH</b> R2 <b>PUSH</b> R5 <b>MOV</b> R5,M[Primeira_Posic] <b>MOV</b> M[IO_CONTROL],R5 <b>MOV</b> R1,ASTERISCOS  Barreiral: <b>MOV</b> R2,M[R1] ;Escreve Barreira <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> AcabouBarr1  <b>MOV</b> M[PosicColis],R5 <b>CALL</b> Ver_Colisao_Esc  <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R5 <b>MOV</b> M[IO_CONTROL],R5 <b>INC</b> R1 <b>BR</b> Barreiral  AcabouBarr1: <b>ADD</b> R5,M[Numero_0100h] ;Terminou de Escrever a Barreira <b>SUB</b> R5,3 <b>MOV</b> M[Primeira_Posic],R5 <b>POP</b> R5 <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>  Ver_Colisao_Esc: <b>PUSH</b> R1 ;Realiza CMP's para verificar se existiram colisões <b>PUSH</b> R2  <b>MOV</b> R1,M[PosicColis]	

Dec 05, 13 21:27	final_coment1.as	Page 18/27
	<b>MOV</b> R2,R6  <b>SUB</b> R2,M[Numero_0300h] <b>CMP</b> R1,R2 <b>CALL.Z</b> GAMEOVER  <b>ADD</b> R2,M[Numero_0100h] <b>CMP</b> R1,R2 <b>CALL.Z</b> GAMEOVER  <b>ADD</b> R2,M[Numero_0100h] <b>CMP</b> R1,R2 <b>CALL.Z</b> GAMEOVER  <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>  AcabaTemp: <b>CALL</b> VerificaNivel ;Atualiza o nivel de jogo <b>MOV</b> R1,M[ValorTemp]  <b>MOV</b> M[DURACAO],R1 <b>MOV</b> R1,M[Numero1] <b>MOV</b> M[ATIVA_TEMP],R1  <b>CALL</b> IncrementaLCD1 ;Incrementa a Distancia Percorrida  <b>MOV</b> R1,M[Numero1] ;Este processo verifica quantas barreiras foram inicializadas <b>SUB</b> M[Inicia_Barreir],R1 <b>MOV</b> M[Temporizador_Mem],R1  <b>POP</b> R5 <b>POP</b> R2 <b>POP</b> R1  <b>RET</b>  IncrementaLCD1: <b>PUSH</b> R1 ;Incrementa LCD <b>PUSH</b> R2 <b>PUSH</b> R3  <b>INC</b> M[Distancia_Percorrida] ;Incrementa a distancia percorrida <b>MOV</b> R3,M[Numero_0030h]  <b>INC</b> M[Valor_ASCII_LCD_1] <b>MOV</b> R1,M[Valor_ASCII_LCD_1] <b>CMP</b> R1,M[Numero_003Ah] ;Se valor não for um numero (:) coloca ascii a 0030 = 0 <b>BR.Z</b> ColocaAscii0 <b>JMP</b> ContinuaASCII_1  ColocaAscii0: <b>MOV</b> M[Valor_ASCII_LCD_1],R3 <b>INC</b> M[Valor_ASCII_LCD_2]	

Dec 05, 13 21:27	final_coment1.as	Page 19/27
	<b>MOV</b> R2,M[Valor_ASCII_LCD_2] <b>CMP</b> R2,M[Número_003Ah] <b>BR.Z</b> ColocaAscii1  <b>JMP</b> ContinuaASCII_1	
ColocaAscii1:	<b>INC</b> M[Valor_ASCII_LCD_3] <b>MOV</b> R2,M[Valor_ASCII_LCD_3] <b>CMP</b> R2,M[Número_003Ah] <b>BR.Z</b> ColocaAscii2  <b>MOV</b> M[Valor_ASCII_LCD_1],R3 <b>MOV</b> M[Valor_ASCII_LCD_2],R3 <b>JMP</b> ContinuaASCII_1	
ColocaAscii2:	<b>MOV</b> R2,M[Valor_ASCII_LCD_4]  <b>MOV</b> M[Valor_ASCII_LCD_1],R3 <b>MOV</b> M[Valor_ASCII_LCD_2],R3 <b>MOV</b> M[Valor_ASCII_LCD_3],R3 <b>INC</b> M[Valor_ASCII_LCD_4]	
ContinuaASCII_1:	<b>MOV</b> R2,M[Número_800Dh] <b>MOV</b> M[LCD_CONTROLO],R2 <b>MOV</b> R1,M[Valor_ASCII_LCD_1] <b>MOV</b> M[LCD_ESCRITA],R1  <b>DEC</b> R2 <b>MOV</b> M[LCD_CONTROLO],R2 <b>MOV</b> R1,M[Valor_ASCII_LCD_2] <b>MOV</b> M[LCD_ESCRITA],R1  <b>DEC</b> R2 <b>MOV</b> M[LCD_CONTROLO],R2 <b>MOV</b> R1,M[Valor_ASCII_LCD_3] <b>MOV</b> M[LCD_ESCRITA],R1  <b>DEC</b> R2 <b>MOV</b> M[LCD_CONTROLO],R2 <b>MOV</b> R1,M[Valor_ASCII_LCD_4] <b>MOV</b> M[LCD_ESCRITA],R1  <b>POP</b> R3 <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
VerificaNivel:	<b>PUSH</b> R1 ;Verifica o Nível Atual e refresca a variável da duração <b>PUSH</b> R2	

Dec 05, 13 21:27	final_coment1.as	Page 20/27
	<b>PUSH</b> R3  <b>CMP</b> M[Turbo_Mem],R0 <b>JMP.Z</b> MODOTURBO  <b>MOV</b> R3,M[ValorLeds] <b>MOV</b> M[LEDS],R3  <b>MOV</b> R1,M[ValorNivel] <b>MOV</b> R2,M[Número1] <b>CMP</b> R1,R2 <b>BR.Z</b> Nivel_1  <b>MOV</b> R2,M[Número_0002h] <b>CMP</b> R1,R2 <b>BR.Z</b> Nivel_2  <b>MOV</b> R2,M[Número_0003h] <b>CMP</b> R1,R2 <b>BR.Z</b> Nivel_3	
Nivel_1:	<b>CMP</b> R3,M[Número_F000h] <b>BR.Z</b> Nivel1TEMP	
Nivel_2:	<b>CMP</b> R3,M[Número_FF00h] <b>BR.Z</b> Nivel2TEMP	
Nivel_3:	<b>CMP</b> R3,M[Número_FFF0h] <b>BR.Z</b> Nivel3TEMP	
Nivel1TEMP:acao	<b>MOV</b> R1,M[Número_0005h] ; Coloca o nivel 1 na variavel da duracao <b>MOV</b> M[ValorTemp],R1 <b>JMP</b> TerminaVerifica	
Nivel2TEMP:cao	<b>MOV</b> R1,M[Número_0004h]; Coloca o nivel 2 na variavel da duracao <b>MOV</b> M[ValorTemp],R1 <b>JMP</b> TerminaVerifica	
Nivel3TEMP:cao	<b>MOV</b> R1,M[Número_0003h]; Coloca o nivel 3 na variavel da duracao <b>MOV</b> M[ValorTemp],R1 <b>JMP</b> TerminaVerifica	
MODOTURBO:	<b>MOV</b> R1,M[Número_FFFFh]; Coloca o turbo na variavel da duracao <b>MOV</b> M[LEDS],R1 <b>MOV</b> R1,M[Número_0002h] <b>MOV</b> M[ValorTemp],R1	
TerminaVerifica:	<b>POP</b> R3 <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Ver_Colisao:	<b>PUSH</b> R2 ;Verifica as barreiras a descer e se houve colisão	

Dec 05, 13 21:27	final_coment1.as	Page 21/27
	<pre>PUSH    R3  MOV     R3,M[PosicColis] ADD     R3,M[Número_0100h] MOV     R2,R6  SUB     R2,M[Soma2Linhas] CMP     R3,R2 CALL.Z  GAMEOVER  ADD     R2,M[Número_0100h] CMP     R3,R2 CALL.Z  GAMEOVER  ADD     R2,M[Número_0100h] CMP     R3,R2 CALL.Z  GAMEOVER  POP     R3 POP     R2 RET  GAMEOVER:  PUSH    R1                ;Coloca Distancia a e se Distancia Perc &gt; Distancia Max coloca Distancia Perc. em distancia Max.             PUSH    R2             PUSH    R3             MOV     M[HouveColisao],R0              MOV     M[Barreiras_Ultr],R0             MOV     M[Barreiras_90],R0             MOV     M[Barreiras_900],R0              MOV     R2,M[Distancia_Percorrida]             MOV     R3,M[Distancia_Percorrida_MAX]              CMP     R2,R3             BR.NP   AcabaGameOver              MOV     R1,M[Valor_ASCII_LCD_4]             MOV     M[Ultima_Pontuacao_4],R1              MOV     R1,M[Valor_ASCII_LCD_3]             MOV     M[Ultima_Pontuacao_3],R1              MOV     R1,M[Valor_ASCII_LCD_2]             MOV     M[Ultima_Pontuacao_2],R1              MOV     R1,M[Valor_ASCII_LCD_1]             MOV     M[Ultima_Pontuacao_1],R1              MOV     M[Distancia_Percorrida_MAX],R2  AcabaGameOver:  MOV     M[Distancia_Percorrida],R0                 POP     R3                 POP     R2</pre>	

Dec 05, 13 21:27	final_coment1.as		Page 22/27
	POP	R1	
	RET		
Inicio:	MOV	R7,SP_INICIAL	;Inicializa a pilha
	MOV	SP,R7	
	MOV	R7, INT_MASK_INICIAL	;Ativa Interrupcao I1
	MOV	M[INT_MASK_ADDR], R7	
Texto	MOV	R6,IO_READ	;Porto de controlo da Janela de
	MOV	M[IO_CONTROL],R6	
	MOV	R6,BemVindoPos	
	MOV	M[IO_CONTROL],R6	
	CALL	BemVind	
	ENI		
	JMP	Infinito	
Escreve_LCD:	PUSH	R1	
	PUSH	R2	
	PUSH	R3	
	MOV	R1,LCD_MASK	
	MOV	M[LCD_CONTROLO],R1	; Porto de controlo do LCD
	MOV	R3,LCD1	
OutraVez:	MOV	R2,M[R3]	;Escreve "Distancia: m" no LC
D	CMP	R2,FIM_TEXTO	
	BR.Z	MudalinhaLCD	
	MOV	M[LCD_ESCRITA],R2	
	INC	R3	
	INC	R1	
	MOV	M[LCD_CONTROLO],R1	
	BR	OutraVez	
MudalinhaLCD:	MOV	R1,LCD_MASK	;Atualiza Porto Controlo LCD
	ADD	R1,M[Número_0010h]	
	MOV	M[LCD_CONTROLO],R1	
	MOV	R3,LCD2	
OutraVez1:	MOV	R2,M[R3]	;Escreve "Maximo: m" no LCD
	CMP	R2,FIM_TEXTO	
	BR.Z	Ajuda	
	MOV	M[LCD_ESCRITA],R2	
	INC	R3	
	INC	R1	
	MOV	M[LCD_CONTROLO],R1	
	BR	OutraVez1	

Dec 05, 13 21:27	final_coment1.as	Page 23/27
Ajuda: no LCD	<b>MOV</b> R2,M[Numero_801Dh] ;Coloca Pontuação Maxima	
Ultima_Pontuacao_X	<b>MOV</b> M[LCD_CONTROLO],R2 ;Baseia-se em Variaveis chamadas	
	<b>MOV</b> R1,M[Ultima_Pontuacao_1]	
	<b>MOV</b> M[LCD_ESCRITA],R1	
	<b>DEC</b> R2	
	<b>MOV</b> M[LCD_CONTROLO],R2	
	<b>MOV</b> R1,M[Ultima_Pontuacao_2]	
	<b>MOV</b> M[LCD_ESCRITA],R1	
	<b>DEC</b> R2	
	<b>MOV</b> M[LCD_CONTROLO],R2	
	<b>MOV</b> R1,M[Ultima_Pontuacao_3]	
	<b>MOV</b> M[LCD_ESCRITA],R1	
	<b>DEC</b> R2	
	<b>MOV</b> M[LCD_CONTROLO],R2	
	<b>MOV</b> R1,M[Ultima_Pontuacao_4]	
	<b>MOV</b> M[LCD_ESCRITA],R1	
AcabaLCD:	<b>POP</b> R3	
	<b>POP</b> R2	
	<b>POP</b> R1	
	<b>RET</b>	
IniciaJogo:	<b>CALL</b> Apaga_Mensagem ;Inicializa o Jogo	
	<b>MOV</b> R6,IO_READ	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>MOV</b> R6,INICIO	
	<b>MOV</b> M[IO_CONTROL],R6	
	<b>CALL</b> Escreve_Campo	
	<b>CALL</b> Escreve_LCD	
as	<b>MOV</b> R7, INT_MASK ;Novas Interrupc. a serem ativad	
	<b>MOV</b> M[INT_MASK_ADDR], R7	
	<b>MOV</b> R1,M[Numero_F000h] ;Corresponde ao lvl1	
	<b>MOV</b> M[LEDS],R1	
	<b>MOV</b> M[ValorLeds],R1	
ia perc. 0	<b>PUSH</b> R1 ;Coloca nas variaveis de distanc	
	<b>MOV</b> R1,M[Numero_0030h]	
	<b>MOV</b> M[Valor_ASCII_LCD_4],R1	
	<b>MOV</b> M[Valor_ASCII_LCD_3],R1	
	<b>MOV</b> M[Valor_ASCII_LCD_2],R1	
	<b>MOV</b> M[Valor_ASCII_LCD_1],R1	
	<b>MOV</b> R1,M[Numero1]	
_jogo nao ser chamado	<b>MOV</b> M[Comeca_Jogo],R1 ;Para no ciclo Infinito o Inicia	
		;vai manter-se inalterado até ex

Dec 05, 13 21:27	final_coment1.as	Page 24/27
istir o Fim do Jogo	<b>MOV</b> R1,R0 ;Coloca Display de 7 seg. a 0	
	<b>MOV</b> M[IO_DISPLAY1],R1	
	<b>MOV</b> M[IO_DISPLAY2],R1	
	<b>MOV</b> M[IO_DISPLAY3],R1	
	<b>MOV</b> M[IO_DISPLAY4],R1	
eira	<b>CALL</b> Aleatorio1st ;Gera a posição da Primeira Barr	
	<b>PUSH</b> R5	
	<b>MOV</b> R5,M[Primeira_Posic]	
	<b>ADD</b> R5,M[Numero_001Ch]	
	<b>MOV</b> M[Barreira_1],R5	
	<b>MOV</b> M[IO_CONTROL],R5	
	<b>MOV</b> R1,M[Numero_0014h] ;Ativa Temporizador	
	<b>MOV</b> M[DURACAO],R1	
	<b>MOV</b> R1,M[Numero1]	
	<b>MOV</b> M[ATIVA_TEMP],R1	
	<b>POP</b> R5	
	<b>POP</b> R1	
	<b>RET</b>	
TerminaJogo:	<b>PUSH</b> R1 ;Termina o Jogo	
	<b>PUSH</b> R2	
terrupções menos a int.	<b>MOV</b> R7, INT_MASK_INICIAL ; Desabilita todas as in	
	<b>MOV</b> M[INT_MASK_ADDR], R7	
	<b>MOV</b> R1,R0	
	<b>MOV</b> M[ATIVA_TEMP],R1 ;Desativa Temporizador	
	<b>MOV</b> R1,M[Numero_0012h] ;Coloca o valor inicial (0012h)	
	<b>MOV</b> M[Inicia_Barreir],R1	
s é colocado a 0	<b>MOV</b> M[Barreira_1],R0 ;O valor da posicao das barreira	
	<b>MOV</b> M[Barreira_2],R0	
	<b>MOV</b> M[Barreira_3],R0	
	<b>MOV</b> M[Barreira_4],R0	
	<b>MOV</b> R1,M[Numero1]	
rbo	<b>MOV</b> M[HouveColisao],R1 ;Reset nas memorias Colisao e Tu	
	<b>MOV</b> M[Turbo_Mem],R1	
	<b>CALL</b> ApagaJanela	
	<b>CALL</b> TerminouFrase	

Dec 05, 13 21:27	final_coment1.as	Page 25/27
	<b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
TerminouFrase:	<b>PUSH</b> R1 ; Escreve a Frase do GameOver  <b>MOV</b> R1,M[Numero_0924h] <b>MOV</b> M[IO_CONTROL],R1  <b>MOV</b> R6,R0 <b>CALL</b> AcabouFrase  <b>POP</b> R1 <b>RET</b>	
AcabouFrase:	<b>PUSH</b> R1 <b>PUSH</b> R2 <b>PUSH</b> R3 <b>MOV</b> R3,M[Numero_0924h]	
TerminaFrase:	<b>MOV</b> R1,FimdoJogo1 <b>MOV</b> R2,M[R1] <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> AcabaMudaLinha <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R3 <b>MOV</b> M[IO_CONTROL],R3 <b>INC</b> R1 <b>BR</b> TerminaFrase	
AcabaMudaLinha:	<b>ADD</b> R3,M[Soma2Linhas] <b>SUB</b> R3,M[Numero_0018h] <b>MOV</b> M[IO_CONTROL],R3	
TerminaFrase1:	<b>MOV</b> R1,FimdoJogo2 <b>MOV</b> R2,M[R1] <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> AcabaTermina <b>MOV</b> M[IO_WRITE],R2 <b>INC</b> R3 <b>MOV</b> M[IO_CONTROL],R3 <b>INC</b> R1 <b>BR</b> TerminaFrase1	
AcabaTermina:	<b>POP</b> R3 <b>POP</b> R2 <b>POP</b> R1 <b>RET</b>	
Aleatoriolst:	<b>PUSH</b> R5 ;Gera Valores Aleatorios <b>MOV</b> R5,M[Random]	
Aleatorio:	<b>PUSH</b> R1 <b>PUSH</b> R2	

Dec 05, 13 21:27	final_coment1.as	Page 26/27
	<b>PUSH</b> R6 <b>CMP</b> R5,R0 <b>BR.NZ</b> Alea1 <b>ROR</b> R5,1	
	<b>MOV</b> R1,M[Numero_0016h] <b>MOV</b> R2,R5 <b>DIV</b> R2,R1  <b>MOV</b> R5,R1 <b>MOV</b> M[Random],R5	
	<b>POP</b> R6 <b>POP</b> R2 <b>POP</b> R1 <b>POP</b> R5 <b>RET</b>	
	;DIVIDIR POR 0015; Alea1:	
	<b>XOR</b> R5,M[Mascara_Alea] <b>ROR</b> R5,1 <b>MOV</b> R6,R5 <b>SHL</b> R6,8 <b>SHR</b> R6,8 <b>MOV</b> R6,R5	
	<b>MOV</b> R1,M[Numero_0016h] <b>MOV</b> R2,R5 <b>DIV</b> R2,R1 <b>MOV</b> R5,R1 <b>MOV</b> M[Primeira_Posic],R5 <b>MOV</b> M[Random],R5	
	<b>POP</b> R6 <b>POP</b> R2 <b>POP</b> R1 <b>POP</b> R5  <b>RET</b>	
Pausa:	<b>CALL</b> EscrevePausaLCD ;Se ativada a Pausa Faz um ciclo infinito em Pausal até I1 ser clicado novamente	
Pausal:	<b>CMP</b> M[Pausa_Mem],R0 <b>BR.Z</b> Pausal <b>CALL</b> Escreve_LCD <b>RET</b>	
EscrevePausaLCD:	<b>PUSH</b> R1 ;Escreve no LCD pausa <b>PUSH</b> R2 <b>PUSH</b> R3 <b>MOV</b> R1,LCD_MASK <b>MOV</b> M[LCD_CONTROLO],R1 <b>MOV</b> R3,LCDPAUSA1	
EscPausal:	<b>MOV</b> R2,M[R3] <b>CMP</b> R2,FIM_TEXTO <b>BR.Z</b> MudalinhaPausa	

Dec 05, 13 21:27	final_coment1.as	Page 27/27
	<div><div>MOV</div><div>M[LCD_ESCRITA],R2</div></div> <div><div>INC</div><div>R3</div></div> <div><div>INC</div><div>R1</div></div> <div><div>MOV</div><div>M[LCD_CONTROLO],R1</div></div> <div><div>BR</div><div>EscPausa1</div></div> <div>MudalinhaPausa:</div> <div><div>MOV</div><div>R1,LCD_MASK</div></div> <div><div>ADD</div><div>R1,M[Numero_0010h]</div></div> <div><div>MOV</div><div>M[LCD_CONTROLO],R1</div></div> <div><div>MOV</div><div>R3,LCDPAUSA2</div></div> <div>EscPausa2:</div> <div><div>MOV</div><div>R2,M[R3]</div></div> <div><div>CMP</div><div>R2,FIM_TEXTO</div></div> <div><div>BR.Z</div><div>AcabaEscPausa</div></div> <div><div>MOV</div><div>M[LCD_ESCRITA],R2</div></div> <div><div>INC</div><div>R3</div></div> <div><div>INC</div><div>R1</div></div> <div><div>MOV</div><div>M[LCD_CONTROLO],R1</div></div> <div><div>BR</div><div>EscPausa2</div></div> <div>AcabaEscPausa:</div> <div><div>POP</div><div>R3</div></div> <div><div>POP</div><div>R2</div></div> <div><div>POP</div><div>R1</div></div> <div><div>RET</div><div></div></div> <div>Infinito:</div> <div><div>CMP</div><div>M[Comeca_Jogo],R0</div><div></div><div>;Verifica se o jogo já começou</div></div> <div><div>CALL.Z</div><div>IniciaJogo</div><div></div><div></div></div> <div><div>INC</div><div>M[Random]</div><div></div><div></div></div> <div>u interrupcao</div> <div><div>CMP</div><div>M[Temporizador_Mem],R0</div><div></div><div>;Verifica se o Temporizador gero</div></div> <div><div>CALL.Z</div><div>Temporizador</div><div></div><div></div></div> <div><div>CMP</div><div>M[Esquerda_Mem],R0</div><div></div><div>;Verifica se I0 foi ativado</div></div> <div><div>CALL.Z</div><div>Esquerda</div><div></div><div></div></div> <div><div>CMP</div><div>M[Direita_Mem],R0</div><div></div><div>;Verifica se IB foi ativado</div></div> <div><div>CALL.Z</div><div>Direita</div><div></div><div></div></div> <div><div>CMP</div><div>M[HouveColisao],R0</div><div></div><div>;Verifica se ocorreu colisão</div></div> <div><div>CALL.Z</div><div>TerminaJogo</div><div></div><div></div></div> <div><div>CMP</div><div>M[Pausa_Mem],R0</div><div></div><div>;Verifica se ocorreu uma pausa</div></div> <div><div>CALL.Z</div><div>Pausa</div><div></div><div></div></div> <div><div>MOV</div><div>R4,M[ValorNivel]</div><div></div><div></div></div> <div><div>BR</div><div>Infinito</div><div></div><div></div></div>	