

Dec 02, 16 22:50		ProjetoFinal.as		Page 1/31
<pre>; Programa projeto.as ; Zona I: Definição das constantes WindowControl EQU 8000h WindowWrite EQU FFFCh Temporizador_I EQU FFF7h Temporizador_V EQU FFF6h SP_INIT EQU FDFh INT_MASK_ADDR EQU FFFAh INT_MASK EQU 1100000000011111b RandomMask EQU 100000000010110b LOWER_EDGE EQU 1700h RIGHT_EDGE EQU 78 END_Vetor EQU '@' Max_Lasers EQU 50 Asteroide EQU '*' Black EQU 'O' Laser EQU '-' L_VarTextol EQU 10 L_VarTexto2 EQU 16 L_VarTexto3 EQU 11 L_VarTexto4 EQU 11 LCDControl EQU FFF4h LCDWrite EQU FFF5h Display EQU FFF0h ; ZONA II: definicao de variaveis ; Posicao da nave POS WORD 0503h ; Posicao dos asteroides POS_Ast TAB 14 ; Posicao dos buracos negros POS_Black TAB 7 ;Posicao dos lasers Num_Lasers WORD 0 POS_L TAB 77 T_Cria T_Cria_Black WORD 1 T_Ast T_Ast WORD 31 Velocidade F_Baixo WORD 0001h F_Cima WORD 0001h F_Esquerda WORD 0001h F_Direita WORD 0001h F_Dispara WORD 0001h F_Tempo WORD 0001h F_Asteroide WORD 0001h F_Laser_C WORD 0001h F_Inicio WORD 0001h F_Over WORD 0001h Ni_Random WORD 000Ch Feed_Random WORD 0000h</pre>				

Dec 02, 16 22:50		ProjetoFinal.as		Page 2/31
Res	TAB	4		
Pont	WORD	0000h		
Ordem	WORD	0000h		
VarTextol1	STR			'Prepare se', END_Vetor
VarTextol2	STR			'Prima o botao IE', END_Vetor
VarTextol3	STR			'Fim do jogo', END_Vetor
VarTextol4	STR			'Pontuacao: ', END_Vetor
; Atribuicao das Rotinas de tratamento				
; das interrupcoes as respetivas				
INT0	WORD	ORIG	FE00h	
INT1	WORD	INT_Baixo		
INT2	WORD	INT_Cima		
INT3	WORD	INT_Esquerda		
INT4	WORD	INT_Direita		
INT5	WORD	INT_Dispara		
INT6	WORD	INT_OVER		
INT7	WORD	INT_OVER		
INT8	WORD	INT_OVER		
INT9	WORD	INT_OVER		
INT10	WORD	INT_OVER		
INT11	WORD	INT_OVER		
INT12	WORD	INT_OVER		
INT13	WORD	INT_OVER		
INT_E	WORD	INT_Inicio		
TEMP	WORD	INT_Tempo		
; ZONA III: codigo				
; ZONA III - 1: Rotinas de desenho				
; Desenha o ecrã de jogo				
Entradas: ---				
Saidas: ----				
Efeito: ----				
Descricao: Desenha a nave e desenha as bordas				
DrawGame:	CALL	DrawShip		
	CALL	DrawEdges		
	RET			
; DrawCharacter: Rotina que efectua a escrita de um caracter para o ecrã.				
; O caracter pode ser visualizado na janela de texto.				
Entradas: pilha - caracter a escrever e posicao				
Saidas: ---				
Efeitos: alteracao da posicao de memoria M[WindowControl] e M[WindowWrite]				
DrawCharacter:	PUSH	R1		
	MOV	R1, M[SP + 3]		
	MOV	M[WindowControl], R1		
	MOV	R1, M[SP + 4]		

Dec 02, 16 22:50	ProjetoFinal.as	Page 3/31
<pre> ; Desenha a nave ; ; ; ; ; OS ate POS + 3 ; ; res da nave) ; ; espetivas DrawShip: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[POS] MOV R3, 0100h MOV M[WindowControl], R1 desenharm o caractere '>' MOV R2, 003Eh MOV M[WindowWrite], R2 Desenha o caractere '>' DEC R1 MOV M[WindowControl], R1 desenharm o caractere '\,' MOV R2, 0029h MOV M[WindowWrite], R2 Desenha o caractere '\,' ADD R1, R3 MOV M[WindowControl], R1 desenharm o caractere '/' MOV R2, 002Fh MOV M[WindowWrite], R2 Desenha o caractere '/' SUB R1, R3 SUB R1, R3 MOV M[WindowControl], R1 desenharm o caractere '\,' MOV R2, 005Ch MOV M[WindowWrite], R2 Desenha o caractere '\,' POP R3 POP R2 POP R1 RET Entradas: --- Saidas: ---- Efeito: ----- Descricao: Coloca o caractere '#' na primeira e n a ultima linha. ; ; pois na linha 23, depois aumenta</pre>	<pre> MOV M[WindowWrite], R1 POP R1 RETN 2 Entradas: --- Saidas: ---- Efeito: ----- Descricao: Percorre os enderecos de memoria de P (onde estao as coordenadas dos caract e desenha os caracteres nas r PUSH R1 PUSH R2 PUSH R3 MOV R1, M[POS] MOV R3, 0100h MOV M[WindowControl], R1 desenharm o caractere '>' MOV R2, 003Eh MOV M[WindowWrite], R2 Desenha DEC R1 MOV M[WindowControl], R1 desenharm o caractere '\,' MOV R2, 0029h MOV M[WindowWrite], R2 Desenha o caractere '\,' ADD R1, R3 MOV M[WindowControl], R1 desenharm o caractere '/' MOV R2, 002Fh MOV M[WindowWrite], R2 Desenha o caractere '/' SUB R1, R3 SUB R1, R3 MOV M[WindowControl], R1 desenharm o caractere '\,' MOV R2, 005Ch MOV M[WindowWrite], R2 Desenha o caractere '\,' POP R3 POP R2 POP R1 RET Entradas: --- Saidas: ---- Efeito: ----- Descricao: Coloca o caractere '#' na primeira e n a ultima linha. ; ; pois na linha 23, depois aumenta</pre>	<pre> ; Desenha as bordas ; ; ; ; a ultima linha. ; ; pois na linha 23, depois aumenta</pre>

Dec 02, 16 22:50	ProjetoFinal.as	Page 4/31
<pre> ; ; sta dentro dos limites da janela de ; ; DrawEdges: PUSH R1 PUSH R2 PUSH R3 MOV R1, LOWER_EDGE MOV R3, R0 MOV R2, 0023h DrawCicle: MOV M[WindowControl], R3 na primeira posicao(0,0) da janela de texto, primeira linha e primeira colun MOV M[WindowWrite], R2 MOV M[WindowControl], R1 o cursor na posicao (24,0) da janela de texto, ultima linha e primeira coluna MOV M[WindowWrite], R2 INC R1 ;Incrementa a coluna MVL R3, R1 ;Copia os dois bits menos significativos de R1, que correspondem a coluna CMP R3, RIGHT_EDGE ;Testa se ja chegou a ultima coluna BR.NZ DrawCicle ;Se nao e a ultima coluna, repete POP R3 POP R2 POP R1 RET ; Desenha caracteres nas posicoes de uma tabela Entradas: Stack Saidas: ---- Efeito: ----- Descricao: Percorre os todos enderecos de memori a da tabela passada ; ; esenha o caracte tambem passado pelo stack ; ; ue se encontra nesse endereco de memoria Draw_ALL: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[SP + 5] MOV R2, M[SP + 6] MOV R3, END_Vetor CMP M[R1], R3 BR.NZ Draw_ALL_Cic POP R3 POP R2 POP R1 RETN 2 Draw_ALL_Cic: PUSH R2 PUSH M[R1] CALL DrawCharacter</pre>	<pre> a coluna, e repete enquanto e texto (80 colunas). PUSH R1 PUSH R2 PUSH R3 MOV R1, LOWER_EDGE MOV R3, R0 MOV R2, 0023h MOV M[WindowControl], R3 na primeira posicao(0,0) da janela de texto, primeira linha e primeira colun MOV M[WindowWrite], R2 MOV M[WindowControl], R1 o cursor na posicao (24,0) da janela de texto, ultima linha e primeira coluna MOV M[WindowWrite], R2 INC R1 ;Incrementa a coluna MVL R3, R1 ;Copia os dois bits menos significativos de R1, que correspondem a coluna CMP R3, RIGHT_EDGE ;Testa se ja chegou a ultima coluna BR.NZ DrawCicle ;Se nao e a ultima coluna, repete POP R3 POP R2 POP R1 RET ; Desenha caracteres nas posicoes de uma tabela Entradas: Stack Saidas: ---- Efeito: ----- Descricao: Percorre os todos enderecos de memori a da tabela passada ; ; esenha o caracte tambem passado pelo stack ; ; ue se encontra nesse endereco de memoria Draw_ALL: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[SP + 5] MOV R2, M[SP + 6] MOV R3, END_Vetor CMP M[R1], R3 BR.NZ Draw_ALL_Cic POP R3 POP R2 POP R1 RETN 2 Draw_ALL_Cic: PUSH R2 PUSH M[R1] CALL DrawCharacter</pre>	<pre> ; ; sta dentro dos limites da janela de ; ; DrawEdges: PUSH R1 PUSH R2 PUSH R3 MOV R1, LOWER_EDGE MOV R3, R0 MOV R2, 0023h DrawCicle: MOV M[WindowControl], R3 na primeira posicao(0,0) da janela de texto, primeira linha e primeira colun MOV M[WindowWrite], R2 MOV M[WindowControl], R1 o cursor na posicao (24,0) da janela de texto, ultima linha e primeira coluna MOV M[WindowWrite], R2 INC R1 ;Incrementa a coluna MVL R3, R1 ;Copia os dois bits menos significativos de R1, que correspondem a coluna CMP R3, RIGHT_EDGE ;Testa se ja chegou a ultima coluna BR.NZ DrawCicle ;Se nao e a ultima coluna, repete POP R3 POP R2 POP R1 RET ; Desenha caracteres nas posicoes de uma tabela Entradas: Stack Saidas: ---- Efeito: ----- Descricao: Percorre os todos enderecos de memori a da tabela passada ; ; esenha o caracte tambem passado pelo stack ; ; ue se encontra nesse endereco de memoria Draw_ALL: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[SP + 5] MOV R2, M[SP + 6] MOV R3, END_Vetor CMP M[R1], R3 BR.NZ Draw_ALL_Cic POP R3 POP R2 POP R1 RETN 2 Draw_ALL_Cic: PUSH R2 PUSH M[R1] CALL DrawCharacter</pre>

Dec 02, 16 22:50	ProjetoFinal.as	Page 5/31
<pre>INC R1 CMP M[R1], R3 BR.NZ Draw_ALL_Cic POP R3 POP R2 POP R1 RETN 2 ; ZONA III - 2: Rotinas de apagar ; Apaga a nave ; ; Entradas: --- ; Saidas: ---- ; Efeito: ----- ; Descricao: Substitui os caracteres da nave por ' ; ClearShip: PUSH R1 PUSH R2 PUSH R3 MOV R3, 0100h MOV R1, M[POS] MOV M[WindowControl], R1 ;Substit MOV R2, 0020h MOV M[WindowWrite], R2 ;Desenha DEC R1 MOV M[WindowControl], R1 ;Substit MOV M[WindowWrite], R2 ;Desenha ADD R1, R3 MOV M[WindowControl], R1 ;Substit MOV M[WindowWrite], R2 ;Desenha SUB R1, R3 SUB R1, R3 MOV M[WindowControl], R1 ;Substit MOV M[WindowWrite], R2 ;Desenha POP R3 POP R2 POP R1 RET Entradas: --- Saidas: ---- Efeito: ----- Descricao: Substitui os caracteres da nave por ' ; Apaga o laser ; ; ; ; ; ClearObjeto: PUSH R1</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 6/31
<pre>PUSH R2 MOV R1, M[SP + 4] MOV M[WindowControl], R1 ;Substit ui o caractere '-' por um ' , MOV R2, 0020h MOV M[WindowWrite], R2 POP R2 POP R1 RETN 1 ; Clear_All: Apaga todos os elementos de um vetor (da janela de texto) ; ; Entradas: Stack ; Saidas: ---- ; Efeito: ---- ; Descricao: Substitui os caracteres nas posicoes ; descritas na tabela, ; , , ; Clear_All: PUSH R1 PUSH R2 MOV R1, M[SP + 4] MOV R2, END_Vetor CMP M[R1], R2 BR.NZ Clear_Ciclo POP R2 POP R1 RETN 1 Clear_Ciclo: PUSH M[R1] CALL ClearObjeto INC R1 CMP M[R1], R2 BR.NZ Clear_Ciclo POP R2 POP R1 RETN 1 ; ClearEdges: Apaga as fronteiras de jogo ; ; Entradas: Stack ; Saidas: ---- ; Efeito: ---- ; Descricao: Substitui todos os # nas fronteiras d o jogo por ' , ClearEdges: PUSH R1 PUSH R2 PUSH R3 MOV R1, LOWER_EDGE MOV R3, R0 MOV R2, ' , , ClearECicle: MOV M[WindowControl], R3 ;Coloca o cursor na prim eira posicao(0,0) da janela de texto, primeira linha e primeira colun</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 7/31
<pre>MOV M[WindowWrite], R2 MOV M[WindowControl], R1 ;Coloca MOV M[WindowWrite], R2 o cursor na posicao (24,0) da janela de texto, ultima linha e primeira coluna INC R1 ;Incrementa a coluna MVL R3, R1 ;Copia os dois bits menos significativos de R1, que correspondem a coluna CMP R3, RIGHT_EDGE ;Testa se ja chegou a ultima coluna BR.NZ ClearECicle ;Se nao e a ultima coluna, repete POP R3 POP R2 POP R1 RET ; ApagaObjeto: Apaga um objeto (asteroide, laser ou buraco negro) ; Entradas: Stack ; Saidas: ----- ; Efeito: Move os valores guardados nos enderecos de memoria a seguir ao parametro ; por cima dele, ate chegar ao END_Vetor ('@') ; fornecido (stack), para tras, escrevendo</pre>		
ApagaObjeto:	PUSH R1	
	PUSH R2	
	PUSH R3	
	MOV R1, M[SP + 5]	
	MOV R3, END_Vetor	
ApagaCiclo:	MOV R2, M[R1 + 1]	
	MOV M[R1], R2	
	INC R1	
	CMP M[R1], R3	
	BR.NZ ApagaCiclo	
	MOV M[R1], R0	
	POP R3	
	POP R2	
	POP R1	
	RETN 1	
<pre>; ZONA III - 2: Rotinas de Strings ; EscString: Rotina que efectua a escrita de uma cadeia de caracter, terminada ; pelo caracter END_Vetor ; Entradas: Stack ; Saidas: --- ; Efeitos: ---</pre>		
EscString:	PUSH R1	
	PUSH R2	
	PUSH R3	

Dec 02, 16 22:50	ProjetoFinal.as	Page 8/31
	PUSH R4	
	MOV R3, RIGHT_EDGE	
	MOV R4, M[SP + 7]	
	MOV R2, M[SP + 6]	
	SHR R3, 1	
	SHR R4, 1	
	SUB R3, R4	
	ADD R3, M[SP + 8]	
CicloA:	MOV R1, M[R2]	
	CMP R1, END_Vetor	
	BR.Z FimEsc	
	PUSH R1	
	PUSH R3	
	CALL DrawCharacter	
	INC R3	
	INC R2	
	BR CicloA	
FimEsc:	POP R4	
	POP R3	
	POP R2	
	RETN 3	
<pre>; ApagaString: Rotina que apaga de uma cadeia de caracteres, terminada ; pelo caracter END_Vetor ; Entradas: Stack ; Saidas: --- ; Efeitos: --- ; fornecida como parametro (stack) ; no centro do ecrã</pre>		
ApagaString:	PUSH R1	
	PUSH R2	
	PUSH R3	
	PUSH R4	
	MOV R3, RIGHT_EDGE	
	MOV R4, M[SP + 7]	
	MOV R2, M[SP + 6]	
	SHR R3, 1	
	SHR R4, 1	
	SUB R3, R4	
	ADD R3, M[SP + 8]	
CicloApaga:	MOV R1, M[R2]	
	CMP R1, END_Vetor	
	BR.Z FimApaga	
	PUSH ' ,	
	PUSH R3	
	CALL DrawCharacter	
	INC R3	
	INC R2	

Dec 02, 16 22:50	ProjetoFinal.as				Page 9/31
	BR	CicloApaga			
FimApaga:	POP	R4	POP	R3	
			POP	R2	
	POP	R1			
	RETN	3			
; EscPontos: Rotina que escreve o valor dos pontos (Guardado em M[Pont])					
	Entradas: Stack				
	Saidas: ---				
	Efeitos: ---				
	; ve os na posicao fornecida				
	;				
EscPontos:	PUSH	R1	PUSH	R2	
			PUSH	R3	
			PUSH	R4	
	MOV R1, M[Pont]				
	MOV R3, Res				
EscPontosCiclo: MOV R2, 10	DIV R1, R2				
	MOV M[R3], R2				
	INC R3				
	INC M[Ordem]				
	CMP R1, R0				
	BR.NZ EscPontosCiclo				
	MOV R1, Res				
	DEC R1				
	MOV R2, M[SP + 6]				
PontosCiclo: MOV R4, M[R3]	ADD R4, '0'				
	PUSH R4				
	PUSH R2				
	CALL DrawCharacter				
	INC R2				
	DEC R3				
	INC M[Ordem]				
	CMP R3, R1				
	BR.NZ PontosCiclo				
	POP R4				
	POP R3				
	POP R2				
	POP R1				
	RETN 1				

Dec 02, 16 22:50	ProjetoFinal.as		Page 10/31
<pre>; ApagaPontos: Rotina que com o proposito de apagar o escrito pela rotina EscPon tos ; Entradas: Stack ; Saidas: --- ; Efeitos: --- ; Descricao: Escreve ' 's na posicao fornecida ate ; percorrer as posicoes ; no ecrã correspondentes ao nu ; mero guardado em M[Ordem] ; Começa o processo na posicao ; fornecida como parametro (stack)</pre>			
ApagaPontos:	PUSH R1	PUSH R2	
		MOV R1, M[SP + 4]	
		MOV R2, M[Ordem]	
ApagaPontosCic:	PUSH R1	CALL ClearObjeto	
		INC R1	
		DEC R2	
		CMP R2, R0	
		BR.NZ ApagaPontosCic	
		POP R2	
		POP R1	
		RETN 1	
; Zona IV - Rotinas de movimentacao			
; Trata da movimentacao da nave para baixo			
; Entradas: Stack			
; Saidas: ----			
; Efeito: Diminuicao (- 0001h) dos valores guardad			
os na tabela especificada pelo parametro (stack)			
; Descricao: Atualiza as coordenadas (- 0001h) de			
cada um dos elementos presentes			
; arametro (stack). Faz os testes de colisao com a nave			
Move_All:	PUSH R1	PUSH R2	na tabela especificada como p
		PUSH R3	
		MOV R1, M[SP + 5]	
		MOV R2, END_Vetor	
		MOV R3, R0	
		CMP M[R1], R2	
		BR.NZ Move_Ciclo	

Dec 02, 16 22:50	ProjetoFinal.as	Page 11/31
	<pre>POP R3 POP R2 POP R1 RETN 1 Move_Ciclo: PUSH M[R1] CALL ColisoesNave DEC M[R1] PUSH M[R1] CALL ColisoesNave MVBL R3, M[R1] CMP R3, R0 BR.Z Apaga_ Iter_Move: CMP M[R1], R2 BR.NZ Continua_ POP R3 POP R2 POP R1 RETN 1 Continua_: INC R1 CMP M[R1], R2 BR.NZ Move_Ciclo POP R3 POP R2 POP R1 RETN 1 Apaga_: PUSH R1 CALL ApagaObjeto JMP Iter_Move Entradas: --- Saídas: ---- Efeito: Aumento das coordenadas dos elementos da tabela POS_L ; POS_L e aumenta incrementa os ; Move_All_L: PUSH R1 PUSH R2 PUSH R3 PUSH R4 MOV R1, POS_L MOV R2, END_Vetor MOV R3, RIGHT_EDGE MOV R4, R0 CMP M[R1], R2 BR.NZ Move_L_Ciclo POP R4 POP R3 POP R2</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 12/31
Move_L_Ciclo:	<pre>INC M[R1] POP R1 RET MVBL R4, M[R1] CMP R4, R3 BR.NN Apaga CMP M[R1], R2 BR.NZ Continua POP R4 POP R3 POP R2 POP R1 RET INC R1 CMP M[R1], R2 BR.NZ Move_L_Ciclo POP R4 POP R3 POP R2 POP R1 RET PUSH R1 CALL ApagaObjeto DEC M[Num_Lasers] JMP Iter_Move ; Trata da movimentacao da nave para baixo ; Entradas: --- ; Saídas: ----- ; Efeito: Atualiza o que esta em POS -> POS + 3 ; Descricao: Apaga a Nave, atualiza as suas coorde ; nadas (+ 1 linha), testa ; jogo, se nao estiver, nao a desenha TrataBaixo: DSI PUSH R1 PUSH R2 PUSH R3 MOV R1, POS MOV R2, R0 MOV R3, M[POS] ADD R3, 0100h MVBH R2, R3 CMP R2, LOWER_EDGE BR.NZ NextBaixo ;Verifica se a posicao a baixo e o limite de jogo, se for nao modifica as coorde nadas da nave MOV R3, 0100h SUB M[R1], R3</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 13/31
NextBaixo:	POP R3	
	POP R2	
	POP R1	
	ENI	
	RET	
	MOV R2, 0100h	
	SUB M[R1], R2	
	CALL ClearShip	
	ADD M[R1], R2	
	;Modifica as coordenadas da nave colocando todos os caracteres uma linha a baixo	
	PUSH M[R1]	
	PUSH POS_Ast	
	CALL Colisoos_MOVE	
	PUSH M[R1]	
	PUSH POS_Black	
	CALL Colisoos_MOVE	
	CALL DrawShip	
	POP R3	
	POP R2	
	POP R1	
	ENI	
	RET	
	; Trata da movimentacao da nave para cima	
	Entradas: ---	
	Saidas: ----	
	Efeito: Atualiza o que esta em POS -> POS + 3	
	Descricao: Apaga a Nave, atualiza as suas coorde	
	nadas (- 1 linha), testa	
	; jogo, se nao estiver, nao a desenha	
	se a nave está nos limites de	
TrataCima:	DSI	
	PUSH R1	
	PUSH R2	
	PUSH R3	
	MOV R1, POS	
	MOV R2, R0	
	MOV R3, M[R1]	
	SUB R3, 0100h	
	MVBH R2, R3	
	CMP R2, R0	
	BR.NZ NextCima	
	;Verifica se a posicao a cima e o limite de jogo, se for nao modifica as coordenadas da nave	
	MOV R3, 0100h	
	ADD M[R1], R3	
	POP R3	
	POP R2	
	POP R1	
	ENI	

Dec 02, 16 22:50	ProjetoFinal.as	Page 14/31
NextCima:	RET	
	MOV R3, 0100h	
	ADD M[R1], R3	
	CALL ClearShip	
	SUB M[R1], R3	
	;Modifica as coordenadas da nave colocando todos os caracteres uma linha a cima	
	PUSH M[R1]	
	PUSH POS_Ast	
	CALL Colisoos_MOVE	
	PUSH M[R1]	
	PUSH POS_Black	
	CALL Colisoos_MOVE	
	CALL DrawShip	
	POP R3	
	POP R2	
	POP R1	
	ENI	
	RET	
	; Trata da movimentacao da nave para a esquerda	
	Entradas: ---	
	Saidas: ----	
	Efeito: Atualiza o que esta em POS -> POS + 3	
	Descricao: Apaga a Nave, atualiza as suas coorde	
	nadas (- 1 coluna), testa	
	; jogo, se nao estiver, nao a desenha	
	se a nave está nos limites de	
	TrataEsquerda: DSI	
	PUSH R1	
	PUSH R2	
	MOV R1, POS	
	MOV R2, R0	
	MVBH R2, M[R1]	
	DEC R2	
	CMP R2, R0	
	BR.NN NextEsquerda	
	;Verifica se a posicao a cima e o limite de jogo, se for nao modifica as coordenadas da nave	
	INC M[R1]	
	POP R2	
	POP R1	
	ENI	
NextEsquerda:	RET	
	INC M[R1]	
	CALL ClearShip	
	DEC M[R1]	
	PUSH M[R1]	
	PUSH POS_Ast	
	CALL Colisoos_MOVE	

Dec 02, 16 22:50	ProjetoFinal.as	Page 15/31
<pre> PUSH M[R1] PUSH POS_Black CALL Colisoos_MOVE CALL DrawShip POP R2 POP R1 ENI RET ; Trata da movimentacao da nave para a direita ; Entradas: --- ; Saidas: ---- ; Efeito: Atualiza o que esta em POS -> POS + 3 ; Descricao: Apaga a Nave, atualiza as suas coorde ; nadas (+ 1 coluna), testa ; jogo, se nao estiver, nao a desenha TrataDireita: DSI PUSH R1 PUSH R2 PUSH R3 MOV R1, POS MOV R2, RIGHT_EDGE MOV R3, R0 MVEL R3, M[R1] CMP R2, R3 BR.P NextDireita MOV R2, 0001h SUB M[R1], R2 ;Modifica as coordenadas da nave colocando todos os caracteres uma coluna para a direita POP R3 POP R2 POP R1 ENI RET NextDireita: DEC M[R1] ;Modific a as coordenadas da nave colocando todos os caracteres uma coluna para a direita CALL ClearShip INC M[R1] ;Modifica as coordenadas da nave colocando todos os caracteres uma coluna para a direita PUSH M[R1] PUSH POS_Ast CALL Colisoos_MOVE PUSH M[R1] PUSH POS_Black CALL Colisoos_MOVE </pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 16/31
<pre> CALL DrawShip POP R3 POP R2 POP R1 ENI RET ; Trata do disparo do laser ; Entradas: --- ; Saidas: ---- ; Efeito: Atualiza o que esta guardad em POS_I_L ; Descricao: Desenha o Laser, permite a interrupca o pelo temporizador e inicializa o TrataDispara: DSI PUSH R1 PUSH R2 PUSH R3 MOV R2, Max_Lasers INC M[Num_Lasers] CMP M[Num_Lasers], R2 BR.NZ Shoot PUSH M[Velocidade] CALL InitTempo POP R3 POP R2 POP R1 RET Shoot: MOV R1, POS MOV R3, M[R1] INC R3 PUSH R3 PUSH POS_L CALL Cria_Objeto PUSH Laser PUSH R3 CALL DrawCharacter PUSH M[Velocidade] CALL InitTempo POP R3 POP R2 POP R1 ENI RET ; TrataTempo: Rotina que trata dos acontecimentos aquando da interurpcao do temp orizador ; Entradas: --- </pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 17/31
<pre>; ; es dos objetos no ecrã de jogo ; os negros) es/buracos negros, se for, move-os ; la este acontecimento) e testa se e tempo de criar ; e procede para movimentar os lasers, fazendo tambem os ; nao for tempo de criar um buraco negro, teste se o e para os ; um asteroide e procede para desenhar e testar as colisoes dos lasers</pre>	<pre>TrataTempo: PUSH R1 INC M[Feed_Random] DEC M[T_Cria] DEC M[T_Cria_Black] DEC M[T_Ast] CMP M[T_Ast], R0 JMP.NZ CicloTempo PUSH POS_Ast CALL Clear_All PUSH POS_Ast CALL Move_All PUSH Asteroide PUSH POS_Ast CALL Draw_ALL PUSH POS_Black CALL Clear_All PUSH POS_Black CALL Move_All PUSH Black PUSH POS_Black CALL Draw_ALL MOV R1, 2 MOV M[T_Ast], R1 CMP M[T_Cria_Black], R0 JMP.NZ Testa_Ast CALL RandomGen MOV R1, M[Ni_Random] ADD R1, RIGHT_EDGE PUSH R1 PUSH POS_Black CALL Cria_Objeto</pre>	<pre>Saídas: ---- Efeito: Atualiza Flags, Movimenta e testa coliso (nave, lasers, asteroides, burac (repoe a flag que contro um buraco negro, se for, testes das colisoes. Se nao for tempo de criar os asteroides. Se for, cria colisoes dos lasers</pre>

Dec 02, 16 22:50	ProjetoFinal.as	Page 18/31
<pre>Testa_Ast: CicloTempo: INT_Baixo:</pre>	<pre>MOV R1, 40 MOV M[T_Cria_Black], R1 MOV R1, 10 MOV M[T_Cria], R1 JMP CicloTempo CMP M[T_Cria], R0 BR.NZ CicloTempo CALL RandomGen MOV R1, M[Ni_Random] ADD R1, RIGHT_EDGE PUSH R1 PUSH POS_Ast CALL Cria_Objeto MOV R1, 10 MOV M[T_Cria], R1 PUSH POS_L CALL Clear_All CALL Colisao_Laser CALL Colisao_LaserB CALL Move_All_L CALL Colisao_Laser CALL Colisao_LaserB PUSH Laser PUSH POS_L CALL Draw_ALL PUSH M[Velocidade] CALL InitTempo POP R1 RET ZONA V - Tratamento das interrupcoes ; Tratamento da interrupcao 0 ; Entradas: --- ; Saídas: ---- ; Efeito: Modifica M[F_Baixo] para 0 ; Descricao: PUSH R1 MOV M[F_Baixo], R0 MOV R1, 0100h ADD M[POS], R1 POP R1 RTI</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 19/31
<pre>; Tratamento da interrupcao 1 ; ; Entradas: --- ; Saidas: ---- ; Efeito: Modifica M[F_Cima] para 0 ; Descricao: INT_Cima: PUSH R1 MOV M[F_Cima], R0 MOV R1, 0100h SUB M[POS], R1 POP R1 RTI ; Tratamento da interrupcao 2 ; ; Entradas: --- ; Saidas: ---- ; Efeito: Modifica M[F_Esquerda] para 0 ; Descricao: INT_Esquerda: MOV M[F_Esquerda], R0 DEC M[POS] RTI ; Tratamento da interrupcao 3 ; ; Entradas: --- ; Saidas: ---- ; Efeito: Modifica M[F_Direita] para 0 ; Descricao: INT_Direita: MOV M[F_Direita], R0 INC M[POS] RTI ; Tratamento da interrupcao 4 ; ; Entradas: --- ; Saidas: ---- ; Efeito: Modifica M[F_Dispara] para 0 ; Descricao: INT_Dispara: MOV M[F_Dispara], R0 RTI ; Tratamento da interrupcao testada na rotina GameOver ; ; Entradas: --- ; Saidas: ---- ; Efeito: Altera a flag F_Over para 0 ; Descricao: INT_OVER: MOV M[F_Over], R0 RTI ; Tratamento da interrupcao IE ; ; Entradas: ---</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 20/31
<pre>; ; Saidas: ---- ; Efeito: Altera a flag F_Inicio para 0 ; Descricao: (Re)Inicia o jogo INT_Inicio: MOV M[F_Inicio], R0 RTI ; Tratamento da interrupcao do Temporizador ; ; Entradas: --- ; Saidas: ---- ; Efeito: Altera a flag F_Tempo para 0 ; Descricao: INT_Tempo: MOV M[F_Tempo], R0 RTI ; ; Zona VI - Utilidades ; ; Gera um numero aleatorio ; Entradas: ---- ; Saidas: ---- ; Efeito: Coloca um numero aleatorio entre 0 e 23 na varia ; vel Ni_Random ; Descricao: Utiliza o program counter para alimentar um a ; lgoritmo de geracao ; aleatoria de numeros RandomGen: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[Feed_Random] MOV R3, Ni_Random MOV R2, M[Ni_Random] SHR R1, 1 BR.NC e_zero XOR R2, RandomMask ROR R2, 1 MOV R4, 22 DIV R2, R4 INC R4 SHL R4, 8 MOV M[R3], R4 POP R3 POP R2 POP R1 RET e_zero: ROR R2, 1 MOV R4, 22 DIV R2, R4 INC R4 SHL R4, 8 MOV M[R3], R4 POP R3 POP R2</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 21/31
<pre>POP R1 RET ; Gera um numero aleatorio ; Entradas: ---- ; Saidas: ----- ; Efeito: Coloca um numero aleatorio entre 0 e 23 na varia vel Ni_Random ; ; Igoritmo de geracao ; ; GameOver: PUSH POS_Ast CALL Clear_All PUSH POS_Black CALL Clear_All PUSH POS_L CALL Clear_All CALL ClearShip CALL ClearEdges ;Reposicao das Flags INC M[F_Over] INC M[F_Baixo] INC M[F_Cima] INC M[F_Esquerda] INC M[F_Direita] INC M[F_Dispara] INC M[F_Inicio] ;Reposicao dos Registos alterados e nao repostos MOV R1, R0 MOV R2, R0 MOV R3, R0 MOV R4, R0 PUSH 01111111111111b CALL InitMask PUSH 0C00h PUSH L_VarTexto3 PUSH VarTexto3 CALL EscString PUSH 0E00h PUSH L_VarTexto4 PUSH VarTexto4 CALL EscString PUSH 0E2Dh CALL EscPontos ENI CMP M[F_Baixo], R0 JMP.Z TrataOver CMP M[F_Cima], R0 JMP.Z TrataOver CMP M[F_Esquerda], R0 JMP.Z TrataOver CMP M[F_Direita], R0</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 22/31
<pre>JMP.Z TrataOver CMP M[F_Dispara], R0 JMP.Z TrataOver CMP M[F_Over], R0 JMP.Z TrataOver CMP M[F_Inicio], R0 JMP.Z TrataOver JMP CicloOver TrataOver: DSI INC M[F_Baixo] INC M[F_Cima] INC M[F_Esquerda] INC M[F_Direita] INC M[F_Inicio] INC M[F_Over] PUSH 0C00h PUSH L_VarTexto3 PUSH VarTexto3 CALL ApagaString PUSH 0E00h PUSH L_VarTexto4 PUSH VarTexto4 CALL ApagaString PUSH 0E2Ch CALL ApagaPontos JMP Inicia ; Cria um objeto (asteroide, laser ou buraco negro) ; Entradas: Stack ; Saidas: ----- ; Efeito: Acrescenta ao parametro de entrada (TAB) uma pos icao ; Descricao: Move para o fim do (TAB) uma posicao, shiftan do o caracter de fim de tabela ; etira a tabela e a posicao a nela inserir Cria_Objeto: PUSH R1 PUSH R2 PUSH R3 MOV R1, M[SP + 5] MOV R2, M[R1] CMP R2, END_Vetor BR.NZ Cria_Cicle MOV R3, M[SP + 6] MOV M[R1], R3 MOV R2, END_Vetor MOV M[R1 + 1], R2</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 23/31
<pre>POP R3 POP R2 POP R1 RETN 2 Cria_Cicle: INC R1 ;Incrementa a coluna MOV R2, M[R1] CMP R2, END_Vetor BR.NZ Cria_Cicle MOV R3, M[SP + 6] MOV M[R1], R3 MOV R3, END_Vetor MOV M[R1 + 1], R3 POP R3 POP R2 POP R1 RETN 2 ; Escreve no Display de 7 segmentos ; Entradas: ----- ; Saídas: ----- ; Efeito: Escreve a Pontuacao(em decimal) no display de 7 ; segmentos ; nha-o no display de 7 segmentos digito ; a digito (maximo 4 digitos) EscreveDisplay: PUSH R1 MOV R4, Display MOV R1, M[Pont] MOV R2, 10 DIV R1,R2 MOV M[R4],R2 MOV R3, 10 DIV R1,R3 MOV M[R4 + 1],R3 MOV R3, 10 DIV R1,R3 MOV M[R4 + 2],R3 MOV R3, 10 DIV R1,R3 MOV M[R4 + 3],R3 POP R4 POP R3 POP R2 POP R1 RETN ; Escreve no LCD</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 24/31
<pre>; Entradas: ----- ; Saídas: ----- ; Efeito: Escreve a Posicao da nave (linha coluna) no LCD ; Descricao: Converte para decimal os 2 bytes corresponden ; tes a linha e a coluna e ; escreve os no LCD, separados de um es ; paco em branco EscreveLCD: PUSH R1 PUSH R2 PUSH R3 PUSH R4 MOV R1, M[POS] MOV R2, R0 MVL R2, R1 MVBH R3, R1 SHR R3, 8 MOV R1, 10 DIV R2, R1 MOV R4, 1000000000000000b MOV M[LCDControl], R4 ADD R2, '0' MOV M[LCDWrite], R2 INC R4 MOV M[LCDControl], R4 ADD R1, '0' MOV M[LCDWrite], R1 MOV R1, 10 DIV R3, R1 ADD R4, 2 ; aumenta a posicao duas colunas, de modo a dar um espaco MOV M[LCDControl], R4 ADD R3, '0' MOV M[LCDWrite], R3 INC R4 MOV M[LCDControl], R4 ADD R1, '0' MOV M[LCDWrite], R1 POP R4 POP R3 POP R2 POP R1 RET ; Inicializa - Repõe os valores iniciais de todas as variaveis ; necessarias para o funcionamento do jog</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 25/31
<pre>o ; ado quando este ; la interrupções respetiva (IE) Inicializa: DSI</pre>	<pre> PUSH R1 MOV R1, 0503h MOV M[POS], R1 MOV M[Pont], R0 MOV R1, END_Vetor MOV M[POS_Ast], R1 MOV M[POS_Black], R1 MOV M[POS_L], R1 MOV M[Pont], R0 MOV M[Ordem], R0 MOV R1, 1 MOV M[T_Crial, R1 MOV M[T_Ast], R1 MOV R1, 31 MOV M[T_Cria_Black], R1 MOV R1, Res MOV M[R1], R0 MOV M[R1 + 1], R0 MOV M[R1 + 2], R0 MOV M[R1 + 3], R0 INC M[F_Over] INC M[F_Baixo] INC M[F_Cima] INC M[F_Esquerda] INC M[F_Direita] INC M[F_Dispara] INC M[F_Inicio] POP R1 ENI RET</pre>	
	<pre> ; Atribuiçao da mascara de interrupçoes ; ; ; InitMask: PUSH R7</pre>	
	<pre> MOV R7, M[SP + 3] MOV M[INT_MASK_ADDR], R7 ;Define as interrupçoes que vamos perimitr</pre>	
	<pre> POP R7 RETN 1 Entradas: Stack Saídas: ---- Efeito: Inicia o temporizador Descrição: Inicializa o temporizador para o temp</pre>	
	<pre>o dado pelo parametro (que se encontra em SP + 3) InitTempo: PUSH R7</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 26/31
	<pre> ;Velocidade de jogo MOV R7, M[SP + 3] MOV M[Temporizador_V], R7 MOV R7, 1 MOV M[Temporizador_I], R7 POP R7 RETN 1 ZONA VII - Colisoes ; Trata das colisoes dos lasers com os buracos negros ; Entradas: ---- ; Saídas: ----- ; Efeito: Apaga os Lasers que se encontram na mesma posica o que os buracos negros, caso existam ; Descrição: Testa a posicao de todos os laser, para cada buraco negro, acerca da sua posicao e elimina ; os lasers que se encontram na mesma p osicao que os buracos negros</pre>	
	<pre>Colisao_LaserB: PUSH R1 PUSH R2 PUSH R3 PUSH R4 MOV R1, POS_Black MOV R2, POS_L MOV R4, END_Vetor Teste_FIMB: CMP M[R1], R4 JMP.Z AcabaColisaoB Colision_CicloB: MOV R3, M[R1] CMP R3, M[R2] BR.Z ApagaB INC R2 CMP M[R2], R4 BR.Z Update_Black JMP Colision_CicloB AcabaColisaoB: POP R4 POP R3 POP R2 POP R1 RET ApagaB: PUSH R2 CALL ApagaObjeto MOV R2, POS_L INC R1 JMP Teste_FIMB Update_Black: INC R1</pre>	

[illegible]

Dec 02, 16 22:50	ProjetoFinal.as	Page 27/31
<pre>; Trata das colisoes dos lasers com os asteroides ; ; Entradas: ---- ; Saídas: ----- ; Efeito: Remove da respetiva tabela os lasers e os astero ; ides que colidem ; ; Descricao: Testa a posicao de todos os asteroides, para ; cada laser, acerca da sua posicao e elimina ; os lasers e os asteroides que se enco ; ntram na mesma posicao Colisao_Laser: PUSH R1 MOV R2, POS_L JMP Teste_FIMB MOV R1, POS_L MOV R2, POS_Ast MOV R4, END_Vetor CMP M[R1], R4 JMP.Z AcabaColisao MOV R3, M[R2] BR.Z ApagaBoth INC R2 CMP M[R2], R4 BR.Z Update_Laser JMP Colision_Ciclo AcabaColisao: POP R4 ApagaBoth: INC M[Pont] CALL EscreveDisplay PUSH R1 CALL ApagaObjeto PUSH M[R2] CALL ClearObjeto PUSH R2 CALL ApagaObjeto MOV R3, FFFFh MOV M[FFF8h], R3 ;LED ACENDE MOV R2, POS_Ast JMP Teste_FIM Update_Laser: INC R1</pre>		

Dec 02, 16 22:50	ProjetoFinal.as	Page 30/31
	<pre>PUSH L_VarTexto2 PUSH VarTexto2 CALL ApagaString PUSH 0C00h PUSH L_VarTextol PUSH VarTextol CALL ApagaString INC M[F_Over] INC M[F_Baixo] INC M[F_Cima] INC M[F_Esquerda] INC M[F_Direita] INC M[F_Dispara] INC M[F_Inicio] CALL DrawGame CALL EscreveLCD CALL EscreveDisplay ENI PUSH M[Velocidade] CALL InitTempo ; Ciclo Principal de Jogo Ciclo: MOV M[FFFF8h], R0 ;LED APAGA CMP M[F_Baixo], R0 JMP .Z Baixo CMP M[F_Cima], R0 JMP .Z Cima CMP M[F_Esquerda], R0 JMP .Z Esquerda CMP M[F_Direita], R0 JMP .Z Direita CMP M[F_Dispara], R0 JMP .Z Dispara INC M[Feed_Random] CMP M[F_Tempo], R0 JMP .Z Tempo CMP M[F_Inicio],R0 JMP .Z Reinicia JMP Ciclo Baixo: DEC M[Feed_Random] DSI CALL TrataBaixo CALL EscreveLCD INC M[F_Baixo] ENI JMP Ciclo Cima: INC M[Feed_Random]</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 29/31
	<pre>ADD R1, 0100h CMP R1, M[R2] JMP .Z GameOver SUB R1, 0200h CMP R1, M[R2] JMP .Z GameOver INC R2 CMP M[R2], R3 BR.NZ Col_Ciclo ReturnCol: POP R3 POP R2 POP R1 RETN 2 ; ZONA VIII - Programa Principal Inicio: MOV R1, FFFFh MOV M[WindowControl], R1 ;Inicial iza o porto de controlo para a janela de texto ; MOV R1, R0 Inicializa todas as variaveis necessarias para o funcionamento d o jogo. Serve como reposicao de valores quando o jogo e reiniciado Inicia: MOV R7, SP_INIT MOV SP, R7 Inicializacao do Stack Pointer PUSH 0100000000000000b CALL InitMask CALL Inicializa PUSH 0C00h PUSH L_VarTextol PUSH VarTextol CALL EscString PUSH 0E00h PUSH L_VarTexto2 PUSH VarTexto2 CALL EscString Ciclo_Inicia: CMP M[F_Inicio],R0 BR.NZ Ciclo_Inicia PUSH INT_MASK CALL InitMask DSI PUSH 0E00h</pre>	

Dec 02, 16 22:50	ProjetoFinal.as	Page 31/31
	<div><div>DSI</div><div>TrataCima</div><div>CALL</div><div>EscreveLCD</div><div>INC</div><div>M[F_Cima]</div><div>ENI</div><div>JMP</div><div>Ciclo</div></div>	
Esquerda:	<div><div>INC</div><div>M[Feed_Random]</div><div>DSI</div><div>TrataEsquerda</div><div>CALL</div><div>EscreveLCD</div><div>INC</div><div>M[F_Esquerda]</div><div>ENI</div><div>JMP</div><div>Ciclo</div></div>	
Direita:	<div><div>DEC</div><div>M[Feed_Random]</div><div>DSI</div><div>TrataDireita</div><div>CALL</div><div>EscreveLCD</div><div>INC</div><div>M[F_Direita]</div><div>ENI</div><div>JMP</div><div>Ciclo</div></div>	
Dispara:	<div><div>INC</div><div>M[Feed_Random]</div><div>DSI</div><div>TrataDispara</div><div>CALL</div><div>M[F_Dispara]</div><div>ENI</div><div>JMP</div><div>Ciclo</div></div>	
Tempo:	<div><div>DSI</div><div>TrataTempo</div><div>INC</div><div>M[F_Tempo]</div><div>ENI</div><div>JMP</div><div>Ciclo</div></div>	
Reinicia:	<div><div>DSI</div><div>INC</div><div>M[F_Inicio]</div><div>PUSH</div><div>POS_Ast</div><div>CALL</div><div>Clear_All</div><div>PUSH</div><div>POS_Black</div><div>CALL</div><div>Clear_All</div><div>PUSH</div><div>POS_L</div><div>CALL</div><div>Clear_All</div><div>CALL</div><div>ClearShip</div><div>CALL</div><div>ClearEdges</div><div>ENI</div><div>JMP</div><div>Inicia</div></div>	