

HOT-ASM

HOTBIT

INTRODUÇÃO

O HOT-ASM é um programa destinado ao usuário que possuam algum conhecimento de Montadores (“Assembler”) e da linguagem do Z80.

Para maiores detalhes sobre o Z-80 deve ser consultado literatura específica.

Recomendamos a leitura atenta deste manual, para um entendimento perfeito do mesmo.

SOBRE OS EQUIPAMENTOS

Aparelhos necessários:

- Computador Pessoal HOT-BIT
- TV

CARREGANDO O SISTEMA

A introdução e/ou retirada do cartucho deverá ser feita apenas com os aparelhos desligados (o computador pessoal HOT-BIT e a TV).

O cartucho deve ser introduzido no slot (abertura no painel do teclado) com a etiqueta voltada para o usuário.

CARACTERÍSTICAS

1. - Acesso ao HOT-ASM

Quando o computador é ligado entra-se no HOT-BASIC. O acesso ao HOT-ASM é feito através do comando CALL START. Este comando inicia toda a área de memória de utilização pelo HOT-ASM.

Quando de retorno ao HOT-BASIC desejar voltar para o HOT-ASM, sem perder o status, utilizar o comando CALL EDT.

Um programa em BASIC pode alterar as áreas de trabalho e texto utilizadas pelo HOT-ASM. Portanto após a execução de um programa em BASIC aconselha-se utilizar o comando CALL START para voltar ao HOT-ASM, permitindo assim que toda área de utilização por este, seja reiniciada.

2. - Utilização do HOT-ASM

Através do EDITOR, se edita um programa de mnemônicos da linguagem de montagem (assembly language) do Z80. Este programa é denominado PROGRAMA FONTE.

A partir do programa fonte e através do MONTADOR ("ASSEMBLER"), gera-se o PROGRAMA OBJETO. Este é o que realmente o microprocessador entende. O Montador também indica todos erros de sintaxe encontrados no programa fonte. Se algum erro deste for encontrado deve-se retornar ao EDITOR. O MONITOR executa o programa objeto e monitora a execução deste, permitindo alterar posições de memória

e registradores do microprocessador. Com o monitor pode-se encontrar possíveis erros de lógica. Se isto acontecer deve-se voltar novamente para o EDITOR a fim de corrigir estes erros.

O programa fonte e o programa objeto gerados podem ser armazenados em fita cassete através de comandos dentro do HOT-ASM ou fora dele, no HOT-BASIC. Aconselha-se deixar os programas OBJETO em fita diferente dos FONTES.

3. - Memória utilizada pelo HOT-ASM

Distribuição da memória utilizada pelo HB8000 com o cartucho HOT-ASM:

HOT-BASIC	0000H
Área compartilhada entre HOT-BASIC e HOT-ASM.	4000H
RAM	8000H
RAM	C000H

Área utilizada pelo HOT-ASM

ÁREA DE TEXTO (Área variável)	8100H
TABELA DE SÍMBOLOS	Logo depois da área de texto
ÁREA LIVRE	E000H
ÁREA DE TRABALHO (uso do HOT-ASM)	EC00H

Área de Texto: Área utilizada para edição dos PROGRAMAS FONTES.

Tabela de Símbolos: Registro de todos os símbolos (LABELS) utilizados no programa fonte.

Área Livre: Área Livre para receber e executar os programas editados e montados pelo HOT-ASM.

4. - Editor

A entrada das instruções devem ser feitas em letras maiúsculas. O EDITOR é semelhante ao do HOT-BASIC.

Cada linha de comando deve ter um número. Este número pode variar de 1 a 65535. Uma linha permite até 255 caracteres. A correção das linhas, como no HOT-BASIC pode ser feita a nível de tela (FULL SCREEN EDITOR), após pressionar **RETURN** a linha é registrada. As teclas **INS** , **DEL** , **<<** (BS) e demais caracteres de controle do HOT-BASIC também funcionam.

INS → Uma vez teclada inicia modo inserção. Cada caracter digitado é inserido sob o cursor. Outra vez pressionada sai de modo inserção.

DEL → Apaga o caracter sob o cursor e a linha se move um espaço para a esquerda.

<< → Apaga o caracter antes do cursor e move a linha um espaço para a esquerda.

CTRL + **R** → O mesmo que **INS**.

CTRL + **I** → O mesmo que **TAB**.

CTRL + **E** → Apaga toda linha à direita do cursor.

CTRL + **U** → Apaga toda a linha.

CTRL + **F** → Move o cursor para a palavra seguinte.

CTRL + **B** → Move o cursor para a palavra anterior.

CTRL + **N** → Move o cursor para o fim da linha.

4.1. - Comandos do HOT-ASM dentro do Editor.

NEW: Efetua a limpeza da área de texto

AUTO

AUTO n1

AUTO n1,n2 Gera automaticamente os números das linhas.
“n1” é a linha inicial, “n2” o incremento das linhas. Quando “n1” e/ou
“n2” é omitido, assume-se o valor 10.

LIST

LIST n1

LIST n1,

LIST n1,n2 Visualiza as linhas do programa.

“n1” é a linha inicial, “n2” é a linha final. Se “n1” e “n2” ausentes, lista
todo o programa. Somente “n1”, lista somente a linha “n1”.

“n1”, (vírgula) lista a partir da linha “n1”.

LLIST

LLIST n1

LLIST n1,

LLIST n1,n2 Lista na impressora as linhas do programa. O
Uso é semelhante ao LIST.

RENUM

RENUM n1

RENUM n1,n2

RENUM n1,n2,n3 Renumeras as linhas do programa. “n1” é o novo
número da linha, “n2” o antigo e “n3” o espaçamento entre as linhas.

“n1” e/ou “n3” ausentes subtemde-se 10. “n2” ausente considera-se
desde a primeira linha.

n1

DELETE n1

DELETE n1-n2 Apaga as linhas do programa. “n1” apaga so-
mente a linha de número “n1”. “n1-n2” apaga da linha “n1” até a linha
“n2”, inclusive. Pelo menos “n1” deve estar presente.

SEARCH s1 Procura a “string” “s1” dentro do texto e mostra
na tela. **CTRL** + **STOP** ou **SPACE** pára de procurar temporaria-
mente.

LSEARCH s1 Procura a “string” “s1” dentro do texto e imprime
na impressora. A utilização é semelhante ao SEARCH.

CHANGE \$s1\$s2 Efetua a substituição da “string” “s1” pela “s2”.
“\$” é o delimitador; pode ser qualquer caractere deste que não esteja
incluso em “s1”.

SAVE “nome” Grava o programa texto em fita cassete, “nome”
pode conter até 16 caracteres.

MERGE “nome”

MERGE “” Carrega o programa da fita cassete e interliga com o já existente na área de texto. Serve para carregar vários programas como subrotinas. Para não haver interligação utilize o comando NEW para limpar a área de texto. No caso de uma linha a ser carregada já existir na área de texto, esta será sobreposta pela linha sendo carregada. A ausência do “nome” faz com que seja carregado o primeiro programa encontrado na fita cassete.

BASIC

BA Volta para o interpretador HOT-BASIC.

MAP Mostra qual área de memória está ocupada pela área de texto.

A Executa a montagem do programa fonte (vide MONTADOR).

Se introduzir um comando ou fator erroneamente, aparecerão indicações de erros como:

OVERFLOW Quando o número da linha ultrapassar o limite de 65535 no RENUM ou AUTO.

OUT OF MEMORY Quando não houver mais espaço livre na memória.

? Quando for introduzido algum comando errado.

5. - Montador (“Assembler”)

O MONTADOR é executado pelo EDITOR através do comando “A”. A gramática é baseada no microprocessador Z80 da ZILOG. O comando “A” pode vir acompanhado de parâmetros que modificam o modo da montagem.

U (Unlist) Não lista o programa sendo montado, somente as linhas com erro. Sua ausência implica em listar o programa sendo montado.

P (Print) Efetua a saída das listagens na impressora. Funciona com “U” e “N” (ou a ausência destes).

N (No number) É feita a listagem da montagem sem os números das linhas.

D (order) Efetua a saída dos “labels” em ordem de registro.

S (Sort) Efetua a saída dos “labels” classificados em ordem crescente de acordo com o padrão ASCII.

R (Reference) Efetua a saída da listagem de referências cruzadas (onde os “labels” são definidos e onde são referenciados).

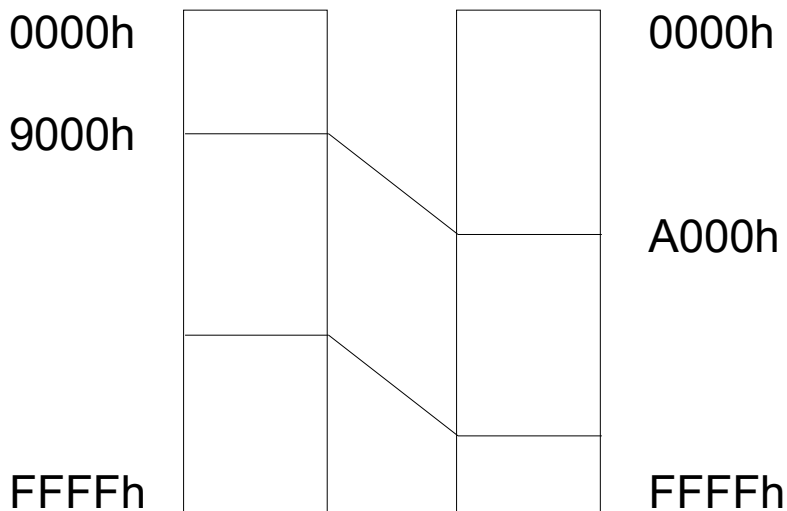
H (Hardcopy) Efetua a saída do conteúdo gerado por “D”, “S” e “R” na impressora.

I (Output) Efetua a saída do programa objeto na fita cassete. Este programa poderá ser lido através do comando “R” do monitor.

O (Object) O programa é montado com endereço inicial igual ao apresentado pela pseudo-instrução ORG no programa fonte e é montado a partir deste.

O/xxxx (Offset) O valor (4 bytes hexadecimal) é acrescentado ao valor apresentado pelo ORG no programa fonte e é montado a partir deste.

Exemplo AO/1000 supondo ORG 9000H



Endereço absoluto = Valor do ORG + Valor do OFFSET.

Formato das Listagens

Listagem da montagem:

colunas

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	...
-	-	-	1	0	-	U	Q	-	9	0	0	0					2	1	0	0	0
número da linha					indicações de erros				endereços					código objeto das instruções							

Exemplo:

10					ORG 1000H
20				;	
30		1000	AF	START:	XOR A
40	U	1001	320000		LD (ANS),A
50		1004			END

Listagem da referência cruzada:

"label"	endereço onde foi definido	endereço onde foi referenc.	endereço onde foi referenc.
---------	-------------------------------	--------------------------------	--------------------------------

Exemplo:

LUM	9010	9017		
MARIA	9001	9007		
PTR	7816			
QU	AB00	9002	9012	9022
QUIT	9038	902D	9034	

Alteração do número de linhas por página na impressora:

O endereço EC00h tem, inicialmente, o valor 3Dh (61 decimal) o que indica 61 linhas por página. O conteúdo deste endereço pode ser alterado livremente através do monitor.

Erros de montagem:

OUT OF MEMORY Ocorre quando o programa objeto vai além da área livre para utilização. Neste caso, a montagem vai ser interrompida.

LABEL FULL Aparece quando o excesso de “labels” definidos no programa fonte não cabem na área da tabela de símbolos. Cada “label” ocupa 10 bytes nesta área; aconselha-se a diminuir estes “labels”. Pode também surgir com a utilização do parâmetro R (referência cruzada) na montagem.

Códigos de Erros:

F	(Format error)	Erro de sintaxe (fora da gramática).
O	(Operand error)	Operando inadequado para a instrução.
Q	(Operation error)	Uso de uma instrução inexistente.
D	(Displacement error)	Ultrapassou a área para desvios relativos
U	(Undefined label)	“Label” indefinido.
M	(Multidefined label)	Está definido o “label” em múltiplos lugares.

6.- Tipos de Programas Gravados em Fita Cassete

Tipo de programa	Como salvar em fita	Como recuperar da fita
PROGRAMA FONTE “Assembler”	SAVE (dentro do EDITOR)	MERGE (dentro do EDITOR)
PROGRAMA OBJETO	Parâmetro I do MONTADOR	Comando R do MONITOR
PROGRAMA BASIC	CSAVE, SAVE (dentro do HOT-BASIC)	CLOAD, LOAD (dentro do HOT-BASIC)
PROGRAMA OBJETO (direto em memória)	BSAVE (dentro do HOT-BASIC)	BLOAD (dentro do HOT-BASIC)

7. - Monitor

O MONITOR possibilita verificar o programa objeto, direto na memória, alterando conteúdos de posições desta, valores dos registradores do microprocessador, executando o programa em partes, etc. Os comandos se formam através de uma letra seguido ou não de valores hexadecimais. Todos os valores apresentados são hexadecimais.

Comandos do MONITOR

Dei

Dei , ef Visualiza o conteúdo da memória começando no endereço “ei” até o endereço “ef”, inclusive. Para interromper a visualização utiliza-se **CTRL** + **STOP** e, para parar temporariamente, utiliza-se **.SPACE**. Se “ef” ausente, serão visualizados 16 bytes começando em “ei”.

Pei

Pei , ef Imprime o conteúdo da memória na impressora. A utilização é a mesma do comando D, com exceção que, cada linha impressa contém 16 bytes, ao contrário do comando D que visualiza 4 bytes por linha.

M

Mei Visualiza e altera o valor na memória. O conteúdo do endereço é visualizado seguido de um hífen. Se não quiser alterar o dado utilizar **SPACE**, senão entrar com dois dígitos hexadecimais. A próxima posição de memória aparece automaticamente. Para encerrar utilize a tecla **RETURN**. A ausência de “ei” faz com que seja considerado o último endereço utilizado por um comando S ou mesmo M. Endereço zero será assumido se nenhum destes comandos tiver sido utilizado anteriormente.

S

Sei Serve para visualizar e introduzir na memória, dados em hexadecimal da mesma forma que o comando M, porém certas teclas são posicionadas para simularem um teclado numérico hexadecimal. O teclado ficaria então:

& 7	* 8	(9	) 0	
U	I	O	P	
J	K	L	Ç	
M	; ,	: .	? /	

que seria equivalente a:

7	8	9	A	
4	5	6	B	
1	2	3	C	
0	F	E	D	

Fei, ef, d Preenche a memória a partir do endereço “ei” até o endereço “ef” com o dado “d”

Tei, ef, et Transfere os dados a partir do endereço “ei” até o endereço “ef”, inclusive, a partir do endereço “et”.

X Visualiza os conteúdos dos registradores do microprocessador e os flags simbolizados.

A	F	B	C	D	E	H	L	
88	FF	67	87	98	90	A0	DE	SZVC

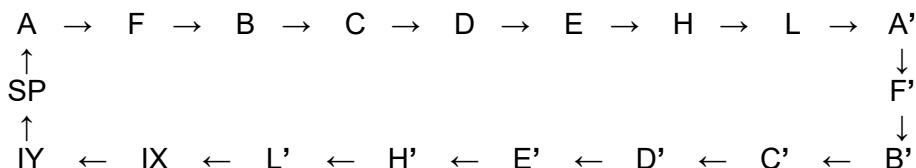
A'	F'	B'	C'	D'	E'	H'	L'	I
00	00	00	00	00	00	00	00	00

IX	IY	SP	PC
0000	0000	FD89	F000

Quando os flags estão ligados aparece a letra indicativa deste flag, caso contrário aparece um espaço. Os flags são:

S	=	flag de sinal
Z	=	flag de zero
V	=	flag de paridade ou “overflow”
C	=	flag de “carry”

Xreg Visualiza e altera o registrador especificado em “reg”. Entra-se com dois dígitos hexadecimais ou 4 dígitos quando forem os registradores IX, IY e SP. X, Y e S representam os registradores IX, IY e SP respectivamente enquanto A, F, B, C, D, E, H, L representam os demais. Após a alteração de cada registrador automaticamente passa-se ao seguinte na cadeia abaixo:



Os registradores auxiliares A', B', C', D', E', H', L' não podem ser representados; devem ser alterados seguindo a cadeia acima. Os registradores I e P não são alteráveis.

Exemplo:

```
LD    SP , $ + 5000H - 0110B
```

```
JR    $ + TESTE
```

```
LD    HL , 'HOT-ASM'
```

```
LD    HL , 'SM'
```

Os dois últimos exemplos têm o mesmo efeito. O excesso é ignorado.

Gei

Gei , b1

Gei , b1 , b2 Inicia a execução do programa no endereço “ei” e interrompe quando o PC atingir o endereço “b1” ou “b2”. Se o PC não atingir “b1” ou “b2” o programa não pára de executar. A ausência de “b1” e “b2” faz com que o programa execute permanentemente. Entre “b1” e “b2” deve haver uma distância superior a 3 bytes.

S ou mesmo M. Endereço zero será assumido se nenhum destes comandos tiver sido utilizado anteriormente.

R

Roff Carrega o programa objeto da fita a partir do endereço indicado pelo ORG no programa fonte adicionado ao valor “off” de offset. A ausência de “off” faz assumir o valor zero para offset. Para que o programa possa ser lido na forma BLOAD dentro do HOT-BASIC é necessário, após carregá-lo com o comando R, sair do HOT-ASM (comando BA) e salvá-lo com o comando BSAVE do HOT-BASIC.

8. - Gramática do “Assembler” do HOT-ASM

Usa-se as letras alfabéticas maiúsculas de “A” a “Z” (41h-5Ah da tabela ASCII) e algarismos de “0” a “9” (30h-39h).

RÓTULOS:

Para os rótulos (“labels”), até 6 caracteres, podendo iniciar com qualquer um entre “?” e “x” (3Fh-5Fh na tabela ASCII) e podendo seguir, além deles, os caracteres entre “0” e “9” (30h-39h).

Exemplo:

HOTASM

J1234

?X

OPERADORES:

Como operadores utiliza-se os caracteres “+” e “-”.

CARACTER ESPECIAL

O carácter “\$” é utilizado para indicar o valor do PC, ou seja, tem como valor o endereço da instrução a executar.

CONSTANTES

As constantes podem ser valores numéricos binários, decimais e hexadecimais e caracteres.

Os valores binários são constituídos por algarismos “0” e “1” seguidos da letra “B”

Exemplo: 10111000B

Os decimais são constituídos por algarismos “0” a “9”

Exemplo: 125

E os hexadecimais dos algarismos “0” a “9” e caracteres “A” a “F” seguidos da letra “H”. Constantes não podem começar com caracteres (“A” a “F”), portanto deve-se acrescentar o dígito “0” antes da constante quando necessitar.

Exemplo: 4A34H
 0FA34H

As constantes caracteres possuem como valor o “string” entre apóstrofes (‘ ’). Se necessitar usar o caractere apóstrofo dentro do “string” é só repeti-lo.

Exemplo: ‘HOT-ASM’
 “GOTA D”ÁGUA’

INSTRUÇÕES

Uma instrução é uma reunião de caracteres que forma a unidade de constituição básica do programa fonte.

Deve ser escrita de forma que caiba em uma linha e é constituída de uma combinação, em sequência das seguintes divisões:

Número da linha

Campo do rótulo (“label”).

Campo do mnemônico,

Campo do comando e

Campo do comentário.

[rótulo:] [mnemônico [operando]] [;comentário]

Os que estão entre colchetes podem ser suprimidos.

[rótulo:]

É um nome para referenciar a posição da instrução. Depois de um rótulo deve-se acrescentar dois pontos (" : "). Não pode ser rótulo palavras reservadas para o montador como END, ORG, registradores do Z-80, etc.

[mnemônico [operando]]

Mnemônicos são as pseudo-instruções do montador ou instruções do Z80 e o operando, e sua complementação. A especificação das instruções é baseada na especificação da ZILOG. O operando pode conter expressões com os operadores "+" e "-".

[;comentário]

Comentários devem ser sempre precedidos de " ; ". Tudo o que vier após o ponto e vírgula até o fim da linha será considerado comentário e será ignorado pelo montador.

PSEUDO-INSTRUÇÕES

Alem das instruções do Z-80 existem aquelas que atribuem informações ao montador. São pseudo-instruções.

ORG n (ORGin)

Determina "n" como o endereço inicial onde começar a carregar o programa. Na ausência da pseudo-instrução assume-se 0 (zero) como valor inicial.

r : EQU n (EQUate)

Define o valor "n" ao rótulo "r"

DEFW d,...,d

DW d,...,d (DEFine Word)

Atribui valores de 16 bits (“d”) da memória no modo invertido (8 bits significativos seguidos dos 8 bits mais significativos).

DEFB d,...,d

DB d,...,dr (DEFine Byte)

Atribui valores de 8 bits (“d”) da memória.

DEFS d

DS d (DEFine Storage)

Reserva espaço na memória com o número de bytes indicados por “d”.

DEFM ‘c’

DM ‘c’ (DEFine Memory)

Atribui os caracteres “c” para a memória. “c” deve estar entre apóstrofes.. Para incluir o apóstrofo em “c” basta repeti-lo.

END (END)

Indica final do programa para o montador.

Exemplos:

PSEUDO	FORMATO	EXEMPLO
ORG	[rótulo:] ORG endereço	ORG 5470H
EQU	rótulo: EQU endereço/dado	LABEL: EQU 1000H
DEFW	[rótulo:] DEFW dados	DEFW 1234H, 124, 'AB'
DEFB	[rótulo:] DEFB dados	DEVB 'A', 34, 23H
DEFS	[rótulo:] DEFS dado	DEFS 100
DEFM	[rótulo:] DEFM caracteres	DEFM 'HOT-ASM'
END	[rótulo:] END	END

Entre colchetes significa opcional

Os três exemplos abaixo têm o mesmo efeito:

DEFM 'ABCD'

DEFB 'A' , 'B' , 'C' , 'D'

DEFW 'BA' , 'DC'

9. - Como entrar com um programa no HOT-ASM

- Com o HB desligado, insira o cartucho do HOT-ASM;
- Ligue o microcomputador e ele entrará no HOT-BASIC;
- Acesse o HOT-ASM através do comando CALL START;
- Uma vez dentro do HOT-ASM introduza o programa como segue:

1000 ;

1010 ; Um programa simples

1020 ;

1030 CHPUT: EQU 00A2H ; sub-rotina que faz sair 1 caracter

```

1040 ASM:    EQU    4010H    ; end que volta controle para
                        HOT-ASM
1050 ;
1060          ORG    0E000H    ; onde começar o programa na
                        memória
1070          LD     HL, DATA ; posiciona endereço da mens.
                        em HL
1080 LOOP:   LD     A,(HL)    ; traz caractere da mensagem
                        para A
1090          CP     0         ; é indicação de fim de mensa-
                        gem?
1100          JR     Z,SAI     ; Desvia para SAI se for fim
1110          CALL   CHPUT     ; mostra caractere no vídeo
1120          INC    HL        ; aponta próximo caractere na
                        mensagem
1130          JR     LOOP      ; reprocessa
1140 ;
1150 SAI:     JP     ASM       ; volta para o controle do
                        HOT-ASM
1160 ;
1170 DATA:   DEFM   'HOT-ASM'; mensagem
1180          DEFB   0         ; indica fim da mensagem para o
                        programa
1190 ;
1200          END             ; indica fim do programa para o
                        MONTADOR

```

- No término da introdução experimente listar o programa com o comando LIST. Se encontrar algum erro é melhor corrigir aqui;
- Utilizando-se o montador através do comando AU (A para o “Assembler” e U para listar somente os erros) localiza-se os erros gramaticais. Se houver erro corrija e monte novamente.

- Se não houver erro grava-se o programa fonte em fita cassete com o comando SAVE:

SAVE "TESTE"

- Gere o programa objeto com o comando AO (Assembler + Objeto). Após este comando ele será carregado na memória;

- Para verificar o programa na memória, use o comando D:

DE000 , E01B

A listagem tem que ser:

E000	21	11	E0	7E	! . . .
E004	FE	00	28	06	.. (.
E008	CD	A2	00	23	▼ó.#
E00C	18	F5	C3	10	..■.
E010	40	48	4F	54	@HOT
E014	20	41	53	4D	ASM
E018	00				.

- Para listar o programa montado na impressora use o comando AP;

- Para executar somente o primeiro comando (LD HL, DATA) digite:

GE000 , E003

- Através do comando X verifique se os registradores ficaram assim:

A	F	B	C	D	E	H	L	
00	00	00	00	00	00	E0	11	
A'	F'	B'	C'	D'	E'	H'	L'	I
00	00	00	00	00	00	00	00	00
IX	IY	SP	PC					
0000	0000	FD89	E003					
dsf								

ANOTAÇÕES

ANOTAÇÕES

HOTBIT