



República Federativa do Brasil
Ministério do Desenvolvimento, Indústria
e Comércio Exterior
Instituto Nacional da Propriedade Industrial

(21) PI 0809585-0 A2



* B R P I 0 8 0 9 5 8 5 A 2 *

(22) Data de Depósito: 14/04/2008
(43) Data da Publicação: 23/09/2014
(RPI 2281)

(51) Int.Cl.:
H04N 7/26

(54) Título: DECODIFICADOR DE REFERÊNCIA
HIPOTÉTICA PARA CODIFICAÇÃO DE VÍDEO
MULTIVISUALIZAÇÃO

(57) Resumo:

(30) Prioridade Unionista: 17/04/2007 US 60/923.800

(73) Titular(es): Thomson Licensing

(72) Inventor(es): Jiancong Luo, Lihua Zhu, Peng Yin, Purvin
Bibhas Pandit

(74) Procurador(es): Nellie Anne Daniel-Shores

(86) Pedido Internacional: PCT US2008004821 de
14/04/2008

(87) Publicação Internacional: WO 2008/130528de
30/10/2008

“DECODIFICADOR DE REFERÊNCIA HIPOTÉTICA PARA CODIFICAÇÃO DE VÍDEO MULTIVISUALIZAÇÃO”

Referência Cruzada a Pedidos Relacionados

5 Este pedido reivindica o benefício do Pedido Provisório US 60/923.800, depositado em 17 de abril de 2007, que é aqui incorporado pela referência em sua íntegra.

Campo Técnico

Este pedido diz respeito a um decodificador de referência hipotética para Codificação de Vídeo Multivisualização em sistemas de vídeo comprimido.

Antecedentes da Invenção

10 Um decodificador de referência hipotética (HRD) é valioso em sistemas de vídeo comprimido em virtude de ele servir para validar um fluxo contínuo de bits codificado para conformidade com um padrão. Em um padrão de codificação, tal como H.264/AVC, há inúmeros pontos de interoperabilidade no recurso da Codificação de Vídeo Multivisualização (MVC) do padrão. MVC permite que o decodificador decodifique uma ou mais visualizações
15 simultaneamente, e a decodificação de uma visualização pode precisar de informação proveniente de outras visualizações. O padrão H.264/AVC tem regras (também referidas como exigências, restrições ou especificações operacionais) que definem um HRD. A conformidade do HRD é uma parte normativa do padrão H.264/MPEG-4 AVC. Todos os fluxos contínuos de bits AVC precisam estar em conformidade com o HRD construído de acordo com as
20 regras. Atualmente, não há regra definida para um HRD para MVC. HRDs anteriores não têm regras suficientes para permitir que eles validem fluxos contínuos de bits como aqueles da MVC no AVC.

Em muitos casos, as variações de taxa de bits do fluxo contínuo de bits comprimido precisam ser estabilizadas usando mecanismos de armazenamento temporário no codificador e no decodificador. Os tamanhos dos armazenamentos temporários físicos são finitos e,
25 portanto, o codificador restringe as variações de taxa de bits para se adequar às limitações do armazenamento temporário. Padrões de codificação de vídeo não obrigam mecanismos de armazenamento temporário específicos para codificador ou decodificador, mas eles exigem que codificadores controlem as flutuações na taxa de bits, de forma que um decodificador de referência hipotética (HRD) (ou Verificador de Armazenamento Temporário Virtual (VBV)) de um dado tamanho de armazenamento temporário decodifique o fluxo contínuo de
30 bits de vídeo sem sofrer sobrefluxo ou subfluxo do armazenamento temporário.

O HRD é baseado em um modelo de decodificador idealizado. O propósito de um HRD é colocar restrições básicas de armazenamento temporário nas variações na taxa de
35 bits durante o tempo em um fluxo contínuo codificado. Por sua vez, estas restrições habilitam camadas superiores a multiplexar o fluxo contínuo, e decodificadores econômicos a decodificá-lo em tempo real. Na seguinte discussão, H.264/AVC será usado como um e-

xemplo. Entretanto, a invenção pode ser estendida a outros modelos ou padrões de codificação de vídeo.

Sumário da Invenção

Esta invenção propõe diversos métodos para definir um Decodificador de Referência Hipotética (HRD) para Codificação de Vídeo Multivisualização (MVC).

Na implementação com base em H.264/MPEG-4 AVC atual da MVC, o software de referência consegue predição de multivisualização pela codificação de cada visualização com um único codificador, e levando em consideração as referências de visualização cruzada. Cada visualização é codificada como um fluxo contínuo de bits separado pelo codificador em sua resolução original e, posteriormente, todos os fluxos contínuos de bits são combinados para formar um único fluxo contínuo de bits. No decodificador, um usuário pode decodificar uma ou mais visualizações com base na sua aplicação. Na especificação atual, nenhuma regra do HRD é definida para MVC.

Na MVC, considere que há N visualizações codificadas no total. O decodificador pode ter a flexibilidade de decodificar todas as M ($1 \leq M \leq N$) visualizações simultaneamente. Cada combinação de M visualizações denota um ponto de interoperabilidade (IOP). Para permitir a melhor flexibilidade, nesta categoria, a presente invenção propõe que o HRD para MVC defina restrições do HRD para cada IOP. O problema com esta solução é que há muitas combinações, até $(2^N - 1)$. Mesmo se uma limitação for adicionada para M , em que $M \leq 32$, o número total de combinações ainda é grande. A presente invenção descreve métodos para fornecer regras para um Decodificador de Referência Hipotética para Codificação de Vídeo Multivisualização, em que as regras são especificadas para cada ponto interoperável.

Descrição Resumida dos Desenhos

A figura 1 mostra parâmetros do HRD de combinações de visualização que são derivadas pelos parâmetros do HRD de visualizações únicas.

A figura 2 mostra um codificador de Codificação de Vídeo Multivisualização.

A figura 3 mostra um decodificador de Codificação de Vídeo Multivisualização.

A figura 4 mostra Informação de Usabilidade de Vídeo para processo de codificação MVC da categoria 1.

A figura 5 mostra Informação de Usabilidade de Vídeo para processo de decodificação MVC da categoria 1.

A figura 6 mostra o processo de codificação SEI com aninhamento Multivisualização da categoria 1.

A figura 7 mostra o processo de decodificação SEI com aninhamento Multivisualização da categoria 1.

A figura 8 mostra Informação de Usabilidade de Vídeo para o processo de codificação MVC da categoria 2.

A figura 9 mostra Informação de Usabilidade de Vídeo para o processo de decodificação MVC da categoria 2.

A figura 10 mostra o processo de codificação SEI com aninhamento Multivisualização das categorias 2 & 3.

5 A figura 11 mostra o processo de decodificação SEI com aninhamento Multivisualização das categorias 2 & 3.

A figura 12 mostra Informação de Usabilidade de Vídeo para o processo de codificação MVC da categoria 3.

10 A figura 13 mostra Informação de Usabilidade de Vídeo para o processo de decodificação MVC da categoria 3.

Descrição Detalhada da Modalidade Preferida

Nas implementações de referência H.264/MPEG-4 AVC atuais, os parâmetros relacionados ao HRD são colocados em mensagens de *ajuste de parâmetro de sequência* (SPS) e SEI.

15 SPS contém os *parâmetros VUI* da estrutura de sintaxe, da forma ilustrada na Tabela 1.

Tabela 1: Ajuste de Parâmetro de Sequência

seq_parameter_set_rbsp() {	C	Descritor
profile_idc	0	u(8)
...		
sequence_parameter_set_id	0	ue(v)
...		
vui_parameters_present_flag	0	u(1)
se(vui_parameters_present_flag)		
vui_parameters()	0	
rbp_trailing_bits()	0	
}		

20 os elementos de sintaxe num_units_in_tick, time_scale, fixed_frame_rate_flag e low_delay_hrd_flag e os *parâmetros* HRD da estrutura de sintaxe são colocados nos *parâmetros VUI* da estrutura de sintaxe, da forma mostrada na Tabela 2.

Tabela 2: Parâmetros VUI

...		
timing_info_present_flag	0	u(1)
se(timing_info_present_flag) {		
num_units_in_tick	0	u(32)
time_scale	0	u(32)
fixed_frame_rate_flag	0	u(1)

}		
nal_hrd_parameters_present_flag	0	u(1)
se(nal_hrd_parameters_present_flag)		
hrd_parameters()		
vcl_hrd_parameters_present_flag	0	u(1)
se(vcl_hrd_parameters_present_flag)		
hrd_parameters()		
se(nal_hrd_parameters_present_flag vcl_hrd_parameters_present_flag)		
low_delay_hrd_flag	0	u(1)
...		

O HRD_parameters da estrutura de sintaxe contém sintaxe para especificar o tamanho do armazenamento temporário e a taxa de bits, etc.

A SEI do período do armazenamento temporário e a SEI do sincronismo da imagem especificaram o atraso da remoção do armazenamento temporário da imagem codificada inicial antes da decodificação de uma imagem, e os parâmetros para derivar o sincronismo de cada imagem.

Com base na implementação AVC atual, os parâmetros relacionados ao HRD colocados no SPS, em conjunto com os parâmetros colocados na SEI do período do armazenamento temporário e na SEI do tempo da imagem, um conjunto de restrições do HRD é definido para um fluxo contínuo de bits H.264/MPEG-4 AVC.

Entretanto, a implementação AVC atual somente suporta um conjunto de parâmetros do HRD, que não se adequa aos múltiplos IOPs contidos no fluxo contínuo de bits da MVC.

Na implementação com base em H.264/MPEG-4 AVC atual da MVC, o software de referência alcança predição de multivisualização pela codificação de cada visualização com um único codificador e levando em consideração as referências de visualização cruzada. Cada visualização é codificada como um fluxo contínuo de bits separado pelo codificador na sua resolução original e, posteriormente, todos os fluxos contínuos de bits são combinados para formar um único fluxo contínuo de bits. No decodificador, o usuário pode decodificar uma ou mais visualizações com base na sua aplicação. Os diagramas do codificador e do decodificador da MVC são mostrados na figura 2 e na figura 3, respectivamente.

Na especificação atual, nenhuma regra do HRD é definida para MVC.

Em muitas das modalidades descritas, usa-se a estrutura da implementação com base no H.264/MPEG-4 AVC da MVC. A ideia similar pode ser aplicada em outros padrões de codificação de vídeo. Por exemplo, o recurso de especificação de parâmetros em particular para um ou mais pontos operacionais (tal como um ponto de interoperabilidade) pode ser aplicado a padrões diferentes do padrão H.264/MPEG-4 AVC.

Três classes separadas de implementações são descritas.

A. Especificar regras do HRD para cada ponto interoperável na MVC

Na MVC, considerando que há N visualizações codificadas no total, o decodificador pode ter a flexibilidade de decodificar todas as M ($1 \leq M \leq N$) visualizações simultaneamente. Cada combinação de M visualizações denota um ponto de interoperabilidade (IOP). Para permitir a melhor flexibilidade, nesta categoria, a presente invenção propõe que um HRD para MVC define restrições do HRD para cada IOP. O problema com esta solução é que há muitas combinações, até $(2^N - 1)$. Mesmo se uma limitação for adicionada para M , em que $M \leq 32$, o número total de combinações ainda é grande.

Uma modalidade desta solução é apresentada a seguir.

Todas as combinações das visualizações são combinadas nas seguintes modificações para as regras do AVC-HRD. A Tabela 3 é tomada do padrão AVC, e diz respeito ao AVC-HRD. As adições nas Tabelas do padrão AVC são mostradas usando fonte *itálica*. Uma nova mensagem SEI, SEI com aninhamento multivisualização, é introduzida na Tabela 4. Os fluxogramas dos processos de codificação e de decodificação VUI são mostrados na figura 4 e na figura 5, respectivamente. E os fluxogramas dos processos de codificação e de decodificação SEI com aninhamento multivisualização são mostrados na figura 6 e na figura 7, respectivamente.

1. Parâmetros VUI: Como pode-se ver, a Tabela 3 mostra que a sintaxe do parâmetro VUI no padrão AVC foi modificada pela introdução de uma declaração "se - então" que testa a variável "profile_idc". Se "profile_idc" for igual a "MVC", então, um laço é realizado uma ou mais vezes para testar um ou mais pontos (isto é, IOPs). Se "profile_idc" não for igual a "MVC", então, presume-se que "AVC" seja o padrão relevante, e um "senão" é executado para testar um ponto para conformidade AVC (usando regras do AVC-HRD existentes). As linhas 30 - 60 são executadas se o "profile_idc" for MVC, caso contrário, as linhas 63 - 88 são executadas. A variável "num_view_combinations_minus1" (linha 30) fornece o número de IOPs no laço. No laço principal, um sub-laço para cada IOP (linhas 33 - 34) fornece os view_ids associados com o IOP. Para cada sub-laço, "num_views_minus1[i]" (linha 32) fornece o número de visualizações associadas com o i -ésimo IOP, e "view_id[i][j]" (linha 34) fornece o view_id da j -ésima visualização do i -ésimo IOP. As linhas 35 - 59 fornecem sintaxes para o i -ésimo IOP. Pode haver até $(2^N - 1)$ IOPs para um fluxo contínuo de bits da MVC. Isto se compara a um único ponto de verificação para um fluxo contínuo de bits AVC.

2. Uma nova mensagem SEI, SEI com aninhamento multivisualização, é definida na Tabela 4. Uma mensagem SEI com aninhamento multivisualização diz respeito a uma unidade de acesso. Uma mensagem SEI com aninhamento multivisualização contém uma, e somente uma, mensagem SEI, que é referida como a mensagem SEI aninhada. O escopo ao qual a mensagem SEI aninhada se aplica é indicado pelas sintaxes das linhas 2 - 10.

"num_view_combinations_minus1", "num_views_minus1[i]" e "view_id[i][j]" compartilham as mesmas semânticas daquelas nos parâmetros VUI.

3. A mensagem SEI do período do armazenamento temporário e a mensagem SEI do sincronismo da imagem (que são partes do AVC) podem ser implementadas como uma carga útil da mensagem SEI embutida na mensagem SEI com aninhamento multivisualização. A sintaxe da SEI do período do armazenamento temporário e da SEI do sincronismo da imagem permanece a mesma do AVC. A mensagem SEI do período do armazenamento temporário e as mensagens SEI do sincronismo da imagem para a visualização compatível com AVC não devem ser aninhadas na SEI com aninhamento multivisualização. Isto permite a compatibilidade com AVC.

Tabela 3: Sintaxe dos parâmetros VUI

1	vui_parameters() {	C	Descritor
2	aspect_ratio_info_present_flag	0	u(1)
3	se(aspect_ratio_info_present_flag) {		
4	aspect_ratio_idc	0	u(8)
5	se(aspect_ratio_idc == Extended_SAR) (		
6	sar_width	0	u(16)
7	sar_height	0	u(16)
8	}		
9	}		
10	overscan_info_present_flag	0	u(1)
11	se(overscan_info_present_flag)		
12	overscan_appropriate_flag	0	u(1)
13	video_signal_type_present_flag	0	u(1)
14	se(video_signal_type_present_flag) {		
15	video_format	0	u(3)
16	video_full_range_flag	0	u(1)
17	colour_description_present_flag	0	u(1)
18	se(colour_description_present_flag) {		
19	colour_primaries	0	u(8)
20	transfer_characteristics	0	u(8)
21	matrix_coefficients	0	u(8)
22	}		
23	}		
24	chroma_loc_info_present_flag	0	u(1)
25	se(chroma_loc_info_present_flag) {		

26	<i>chroma_sample_loc_type_top_field</i>	0	ue(v)
27	<i>chroma_sample_loc_type_bottom_field</i>	0	ue(v)
28	}		
29	<i>se(profile_idc == 'MVC') {</i>		
30	<i>num_view_combinations_minus1</i>	0	ue(v)
31	<i>para(i = 0; i <= num_view_combinations_minus1; i++) {</i>		
32	<i>num_views_minus1[i]</i>	0	u(5)
33	<i>para(j = 0; j <= num_views_minus1[i]; j++)</i>		
34	<i>view_id[i][j]</i>	0	u(5)
35	<i>timing_info_present_flag[i]</i>	0	u(1)
36	<i>se(timing_info_present_flag[i]) {</i>		
37	<i>num_units_in_tick[i]</i>	0	u(32)
38	<i>time_scale[i]</i>	0	u(32)
39	<i>fixed_frame_rate_flag[i]</i>	0	u(1)
40	}		
41	<i>nal_hrd_parameters_present_flag[i]</i>	0	u(1)
42	<i>se(nal_hrd_parameters_present_flag[i])</i>		
43	<i>hrd_parameters()</i>		
44	<i>vcl_hrd_parameters_present_flag[i]</i>	0	u(1)
45	<i>se(vcl_hrd_parameters_present_flag[i])</i>		
46	<i>hrd_parameters()</i>		
47	<i>se(nal_hrd_parameters_present_flag[i] v- cl_hrd_parameters_present_flag[i])</i>		
48	<i>low_delay_hrd_flag[i]</i>	0	u(1)
49	<i>pic_struct_present_flag[i]</i>	0	u(1)
50	<i>bitstream_restriction_flag</i>	0	u(1)
51	<i>se(bitstream_restriction_flag) {</i>		
52	<i>motion_vectors_over_pic_boundaries_flag</i>	0	u(1)
53	<i>max_bytes_per_pic_denom</i>	0	ue(v)
54	<i>max_bits_per_mb_denom</i>	0	ue(v)
55	<i>log2_max_mv_length_horizontal</i>	0	ue(v)
56	<i>log2_max_mv_length_vertical</i>	0	ue(v)
57	<i>num_reorder_frames</i>	0	ue(v)
58	<i>max_dec_frame_buffering</i>	0	ue(v)
59	}		
60	}		

61	}		
62	senão{		
63	timing_info_present_flag	0	u(1)
64	se(timing_info_present_flag) (		
65	num_units_in_tick	0	u(32)
66	time_scale	0	u(32)
67	fixed_frame_rate_flag	0	u(1)
68	}		
69	nal_hrd_parameters_present_flag	0	u(1)
70	se(nal_hrd_parameters_present_flag)		
71	hrd_parameters()		
72	vcl_hrd_parameters_present_flag	0	u(1)
73	se(vcl_hrd_parameters_present_flag)		
74	hrd_parameters()		
75	se(nal_hrd_parameters_present_flag v- cl_hrd_parameters_present_flag)		
76	low_delay_hrd_flag	0	u(1)
77	pic_struct_present_flag	0	u(1)
78	bitstream_restriction_flag	0	u(1)
79	se(bitstream_restriction_flag) {		
80	motion_vectors_over_pic_boundaries_flag	0	u(1)
81	max_bytes_per_pic_denom	0	ue(v)
82	max_bits_per_mb_denom	0	ue(v)
83	log2_max_mv_length_horizontal	0	ue(v)
84	log2_max_mv_length_vertical	0	ue(v)
85	num_reorder_frames	0	ue(v)
86	max_dec_frame_buffering	0	ue(v)
87	}		
88	}		
89	}		

num_view_combinations_minus1 mais 1 indica o número de combinações de visualização suportadas pelo fluxo contínuo de bits em relação ao mesmo seq_parameter_set_id no SPS que contém estes hrd_parameters().

num_views_minus1[i] indica o número de visualizações associadas com o *i*-ésimo

5 IOP.

view_id[i][j] indica o view_id da *j*-ésima visualização do *i*-ésimo IOP.

timing_info_present_flag[i] especifica o valor timing_info_present_flag do *i*-ésimo

IOP.

num_units_in_tick[i] especifica o valor num_units_in_tick do *i-ésimo* IOP.

time_scale[i] especifica o valor time_scale do *i-ésimo* IOP.

fixed_frame_rate_flag[i] especifica o valor fixed_frame_rate_flag do *i-ésimo* IOP.

5 nal_hrd_parameters_present_flag[i] especifica o valor
nal_hrd_parameters_present_flag do *i-ésimo* IOP.

 vcl_hrd_parameters_present_flag[i] especifica o valor v-
cl_hrd_parameters_present_flag do *i-ésimo* IOP.

 low_delay_hrd_flag[i] especifica o valor low_delay_hrd_flag do *i-ésimo* IOP.

10 pic_struct_present_flag[i] especifica o valor pic_struct_present_flag do *i-ésimo* IOP.
 timing_info_present_flag, num_units_in_tick, time_scale,
nal_hrd_parameters_present_flag, vcl_hrd_parameters_present_flag, low_delay_hrd_flag e
pic_struct_present_flag têm a mesma semântica do AVC.

Tabela 4: Sintaxe da mensagem SEI com aninhamento multivisualização

1	multiview_nesting(payloadSize) {	C	Descritor
2	all_view_combinations_in_au_flag	5	u(1)
3	se(all_view_combinations_in_au_flag == 0) {		
4	num_view_combinations_minus1	5	ue(v)
5	para(i = 0; i <= num_view_combinations_minus1; i++) {		
6	num_views_minus1 [i]	0	u(5)
7	para(j = 0; j <= num_views_minus1 [i]; j++)		
8	view_id[i][j]	0	u(5)
9	}		
10	}		
11	enquanto(!byte_aligned())		
12	sei_nesting_zero_bit /* igual a 0 */	5	f(1)
13	sei_message()	5	
14	}		

15 all_view_combinations_in_au_flag igual a 1 indica que a mensagem SEI aninhada
se aplica a todas as combinações de visualização da unidade de acesso.

 all_view_combinations_in_au_flag igual a 0 indica que o escopo aplicável da men-
sagem SEI aninhada é sinalizado pelos elementos de sintaxe
num_view_combinations_minus1, num_views_minus1[i] e view_id[i][j].

20 num_view_combinations_minus1 mais 1 indica o número de combinações de visua-
lização suportadas pelo fluxo contínuo de bits.

 num_views_minus1[i] indica o número de visualizações associadas com o *i-ésimo*
IOP.

`view_id[i][j]` indica o `view_id` da j -ésima visualização do i -ésimo IOP.

B. Especificar regras do HRD somente para um conjunto de IOPs

Nesta categoria, propõe-se definir HRD para MVC para um conjunto de IOPs, e derivar outros conjuntos de IOPs a partir deste conjunto. Em uma modalidade, é uma proposta que o HRD da MVC somente defina restrições do HRD para a decodificação de cada uma das possíveis únicas visualizações, e os parâmetros HRD para decodificação de mais de uma visualização são derivados a partir dos parâmetros de visualização única relevantes. Esta abordagem permite que transmita-se menos parâmetros do que a primeira abordagem, mas ela adiciona mais complexidade para a derivação.

Uma modalidade desta solução é apresentada a seguir:

Um HRD é definido para cada visualização. O número máximo de conjuntos dos parâmetros do HRD transmitindo é M , em que $M=1\dots N$. Então, se houver N visualizações, esta modalidade transmitirá N conjuntos de parâmetros HRD. Os fluxogramas dos processos de codificação e de decodificação VUI são mostrados na figura 8 e na figura 9, respectivamente. E os fluxogramas dos processos de codificação e de decodificação SEI com aninhamento multivisualização são mostrados na figura 10 e na figura 11, respectivamente.

1. Parâmetros VUI: Como pode-se ver, mostra-se que a sintaxe do parâmetro VUI no padrão AVC foi modificada pela introdução de uma declaração "se – então" que testa a variável "profile_idc". Se "profile_idc" for igual a "MVC", então, um laço é realizado uma ou mais vezes para testar um ou mais pontos. Se "profile_idc" não for igual a "MVC", então, presume-se que "AVC" seja o padrão relevante, e um "senão" é executado para testar um ponto para conformidade AVC (usando regras do AVC-HRD existentes). As linhas 30 - 58 são executadas se o "profile_idc" for MVC, caso contrário, as linhas 61 – 85 são executadas. A variável "num_view_minus1" (linha 30) fornece o número de visualizações no laço. "view_id[i]" (linha 32) fornece o `view_id` da i -ésima visualização. As linhas 33 - 58 fornecem sintaxes para a i -ésima visualização. Pode haver até M visualizações ($M=1\dots N$) em um fluxo contínuo de bits da MVC. Isto se compara a um único ponto de verificação para um fluxo contínuo de bits AVC.

2. Uma nova mensagem SEI, SEI com aninhamento multivisualização, é definida na Tabela 6. Uma mensagem SEI com aninhamento multivisualização diz respeito a uma unidade de acesso. Uma mensagem SEI com aninhamento multivisualização contém uma, e somente uma, mensagem SEI, que é referida como a mensagem SEI aninhada. O escopo ao qual a mensagem SEI aninhada se aplica é indicado pelas sintaxes das linhas 2 - 7. "num_views_minus1" e "view_id[i]" compartilham as mesmas semânticas daquelas nos parâmetros VUI.

3. A mensagem SEI do período do armazenamento temporário e a mensagem SEI do sincronismo da imagem podem ser como uma carga útil da mensagem SEI embutida na

mensagem SEI com aninhamento multivisualização. A sintaxe da SEI do período do armazenamento temporário e da SEI do sincronismo da imagem permanece a mesma do AVC. A mensagem SEI do período do armazenamento temporário e as mensagens SEI do sincronismo da imagem para a visualização compatível com AVC não devem ser aninhadas na

5 SEI com aninhamento multivisualização.

Os parâmetros HRD de cada IOP (combinação de visualização) são derivados pelos parâmetros HRD das visualizações associadas com o IOP. Um exemplo é dado na figura 1.

Tabela 5: Sintaxe dos parâmetros VUI

1	vui_parameters() {	C	Descriptor
2	aspect_ratio_info_present_flag	0	u(1)
3	se(aspect_ratio_info_present_flag) {		
4	aspect_ratio_idc	0	u(8)
5	se(aspect_ratio_idc == Extended_SAR) (		
6	sar_width	0	u(16)
7	sar_height	0	u(16)
8	}		
9	}		
10	overscan_info_present_flag	0	u(1)
11	se(overscan_info_present_flag)		
12	overscan_appropriate_flag	0	u(1)
13	video_signal_type_present_flag	0	u(1)
14	se(video_signal_type_present_flag) {		
15	video_format	0	u(3)
16	video_full_range_flag	0	u(1)
17	colour_description_present_flag	0	u(1)
18	se(colour_description_present_flag) {		
19	colour_primaries	0	u(8)
20	transfer_characteristics	0	u(8)
21	matrix_coefficients	0	u(8)
22	}		
23	}		
24	chroma_loc_info_present_flag	0	u(1)
25	se(chroma_loc_info_present_flag) {		
26	chroma_sample_loc_type_top_field	0	ue(v)
27	chroma_sample_loc_type_bottom_field	0	ue(v)
28	}		
29	se(profile_idc == 'MVC') {		
30	num_views_minus1[i]	0	ue(v)
31	para(i = 0; i <= num_views_minus1; i++) (		
32	view_id[i][j]	0	u(5)
33	timing_info_present_flag[i]	0	u(1)
34	se(timing_info_present_flag[i]) {		
35	num_units_in_tick[i]	0	u(32)
36	time_scale[i]	0	u(32)
37	fixed_frame_rate_flag[i]	0	u(1)
38	}		
39	nal_hrd_parameters_present_flag[i]	0	u(1)
40	se(nal_hrd_parameters_present_flag[i])		
41	hrd_parameters()		

42	<i>vcl_hrd_parameters_present_flag[i]</i>	0	u(1)
43	<i>se(vcl_hrd_parameters_present_flag[i])</i>		
44	<i>hrd_parameters()</i>		
45	<i>se(nal_hrd_parameters_present_flag[i] vcl_hrd_parameters_present_flag[i])</i>		
46	<i>low_delay_hrd_flag[i]</i>	0	u(1)
47	<i>pic_struct_present_flag[i]</i>	0	u(1)
48	<i>bitstream_restriction_flag</i>	0	u(1)
49	<i>se(bitstream_restriction_flag) {</i>		
50	<i>motion_vectors_over_pic_boundaries_flag</i>	0	u(1)
51	<i>max_bytes_per_pic_denom</i>	0	ue(v)
52	<i>max_bits_per_mb_denom</i>	0	ue(v)
53	<i>log2_max_mv_length_horizontal</i>	0	ue(v)
54	<i>log2_max_mv_length_vertical</i>	0	ue(v)
55	<i>num_reorder_frames</i>	0	ue(v)
56	<i>max_dec_frame_buffering</i>	0	ue(v)
57	<i>}</i>		
58	<i>}</i>		
59	<i>}</i>		
60	<i>senão{</i>		
61	<i>timing_info_present_flag</i>	0	u(1)
62	<i>se(timing_info_present_flag) {</i>		
63	<i>num_units_in_tick</i>	0	u(32)
64	<i>time_scale</i>	0	u(32)
65	<i>fixed_frame_rate_flag</i>	0	u(1)
66	<i>}</i>		
67	<i>nal_hrd_parameters_present_flag</i>	0	u(1)
68	<i>se(nal_hrd_parameters_present_flag)</i>		
69	<i>hrd_parameters()</i>		
70	<i>vcl_hrd_parameters_present_flag</i>	0	u(1)
71	<i>se(vcl_hrd_parameters_present_flag)</i>		
72	<i>hrd_parameters()</i>		
73	<i>se(nal_hrd_parameters_present_flag vcl_hrd_parameters_present_flag)</i>		
74	<i>low_delay_hrd_flag</i>	0	u(1)
75	<i>pic_struct_present_flag</i>	0	u(1)
76	<i>bitstream_restriction_flag</i>	0	u(1)
77	<i>se(bitstream_restriction_flag) {</i>		
78	<i>motion_vectors_over_pic_boundaries_flag</i>	0	u(1)
79	<i>max_bytes_per_pic_denom</i>	0	ue(v)
80	<i>max_bits_per_mb_denom</i>	0	ue(v)
81	<i>log2_max_mv_length_horizontal</i>	0	ue(v)
82	<i>log2_max_mv_length_vertical</i>	0	ue(v)
83	<i>num_reorder_frames</i>	0	ue(v)
84	<i>max_dec_frame_buffering</i>	0	ue(v)
85	<i>}</i>		
86	<i>}</i>		
87	<i>}</i>		

num_views_minus1 indica o número total de visualizações no fluxo contínuo de bits.

view_id[i] indica o *view_id* da *i*-ésima visualização.

timing_info_present_flag[i] especifica o valor *timing_info_present_flag* da *i*-ésima visualização.

5 *num_units_in_tick[i]* especifica o valor *num_units_in_tick* da *i*-ésima visualização.

time_scale[i] especifica o valor *time_scale* do *i*-ésimo IOP.

fixed_frame_rate_flag[i] especifica o valor fixed_frame_rate_flag da *i-ésima* visualização.

nal_hrd_parameters_present_flag[i] especifica o valor nal_hrd_parameters_present_flag da *i-ésima* visualização.

5 vcl_hrd_parameters_present_flag[i] especifica o valor vcl_hrd_parameters_present_flag da *i-ésima* visualização.

low_delay_hrd_flag[i] especifica o valor low_delay_hrd_flag da *i-ésima* visualização.

pic_struct_present_flag[i] especifica o valor pic_struct_present_flag da *i-ésima* visualização.

10 timing_info_present_flag, num_units_in_tick, time_scale, nal_hrd_parameters_present_flag, vcl_hrd_parameters_present_flag, low_delay_hrd_flag e pic_struct_present_flag têm a mesma semântica do AVC.

Tabela 6: Sintaxe da mensagem SEI com aninhamento multivisualização

1	multiview_nesting(payloadSize) {	C	Descriptor
2	all_views_in_au_flag	5	u(1)
3	se(all_views_in_au_flag == 0) {		
4	num_views_minus1	5	ue(v)
5	para(i = 0; i <= num_views_minus1; i++)		
6	view_id[i]	0	u(5)
7	}		
8	enquanto(!byte_aligned())		
9	sei_nesting_zero_bit /* igual a 0 */	5	f(1)
10	sei_message()	5	
11	}		

15 all_views_in_au_flag igual a 1 indica que a mensagem SEI aninhada se aplica a todas as visualizações da unidade de acesso. all_views_in_au_flag igual a 0 indica que o escopo aplicável da mensagem SEI aninhada é sinalizado pelos elementos de sintaxe num_views_minus1 e view_id[i].

num_views_minus1 mais 1 indica o número de visualizações suportadas pelo fluxo contínuo de bits.

20 view_id[i] indica o view_id da *i-ésima* visualização.

C. Especificar regras do HRD para o pior caso

Nesta categoria, propõe-se que o HRD da MVC somente defina restrições do HRD para o pior caso da decodificação de M (M=1, N) visualizações. O pior caso pode ser definido como, por exemplo, o caso que exige o maior tamanho de armazenamento temporário, o maior atraso, a mais alta taxa de bits, o maior consumo de bits da transmissão de todas as 25 M visualizações, etc. Então, pode-se definir somente 1 conjunto de parâmetros do HRD para

a decodificação de cada uma das M visualizações. Isto é, define-se um conjunto de parâmetros do HRD para o pior caso de única visualização, e um conjunto de parâmetros do HRD para o pior caso de combinação de 2 visualizações, etc., até um conjunto de parâmetros do HRD para o pior caso de combinação de $N-1$ visualizações, e um conjunto de parâmetros do HRD para a combinação (pior caso) de N visualizações. No total, somente define-se N conjuntos de parâmetros do HRD, que são usados para testar a conformidade do HRD para 1, 2, e até N visualizações. A desvantagem desta abordagem é que ela exige transmitir restrições mais altas do que um caso em particular pode exigir. Mas, pode-se economizar os bits para transmitir todos os parâmetros do HRD e não deriva-se nenhum parâmetro.

Uma modalidade desta solução é apresentada a seguir.

O número máximo de conjuntos dos parâmetros HRD transmitindo é M , em que $M=1$ a N . Os fluxogramas dos processos de codificação e de decodificação VUI são mostrados na figura 12 e na figura 13, respectivamente.

1. Parâmetros VUI: Como pode-se ver, a Tabela 7 mostra que a sintaxe do parâmetro VUI no padrão AVC foi modificada pela introdução de uma declaração "se - então" que testa a variável "profile_idc". Se "profile_idc" for igual a "MVC", então, um laço é realizado uma ou mais vezes para testar um ou mais pontos. Se "profile_idc" não for igual a "MVC", então, presume-se que "AVC" seja o padrão relevante, e um "senão" é executado para testar um ponto para conformidade AVC (usando regras do AVC-HRD existentes). As linhas 30 - 57 são executadas se o "profile_idc" for MVC, caso contrário, as linhas 60 - 84 são executadas. A variável "num_view_minus1" (linha 30) fornece o número de visualizações no fluxo contínuo de bits. As linhas 32 - 56 fornecem sintaxes para os parâmetros relacionados com HRD associados com as restrições do HRD mais altas da decodificação de todas as i visualizações das $(\text{num_views_minus1}+1)$ visualizações.

Note que a Tabela 7 difere da Tabela 5. Veja, por exemplo, a linha 32 da Tabela 5, citando "view_id[i]".

2. A mensagem SEI do período do armazenamento temporário e a mensagem SEI do sincronismo da imagem podem ser como uma carga útil da mensagem SEI embutida na mensagem SEI com aninhamento multivisualização. A sintaxe da mensagem SEI com aninhamento multivisualização pode compartilhar aquela definida na Tabela 6. A mensagem SEI do período do armazenamento temporário e do sincronismo da imagem são associadas com as mais altas restrições do HRD da decodificação de todas $(\text{num_views_minus1}+1)$ visualizações do número total de visualizações transmitindo no fluxo contínuo de bits. A sintaxe da SEI do período do armazenamento temporário e da SEI do sincronismo da imagem permanece a mesma do AVC. A mensagem SEI do período do armazenamento temporário e as mensagens SEI do sincronismo da imagem para a visualização compatível com AVC não devem ser aninhadas na SEI com aninhamento multivisualização.

Tabela 7: Sintaxe de parâmetros VUI

1	vui_parameters() {	C	Descritor
2	aspect_ratio_info_present_flag	0	u(1)
3	se(aspect_ratio_info_present_flag) {		
4	aspect_ratio_idc	0	u(8)
5	se(aspect_ratio_idc == Extended_SAR) (		
6	sar_width	0	u(16)
7	sar_height	0	u(16)
8	}		
9	}		
10	overscan_info_present_flag	0	u(1)
11	se(overscan_info_present_flag)		
12	overscan_appropriate_flag	0	u(1)
13	video_signal_type_present_flag	0	u(1)
14	se(video_signal_type_present_flag) {		
15	video_format	0	u(3)
16	video_full_range_flag	0	u(1)
17	colour_description_present_flag	0	u(1)
18	se(colour_description_present_flag) {		
19	colour_primaries	0	u(8)
20	transfer_characteristics	0	u(8)
21	matrix_coefficients	0	u(8)
22	}		
23	}		
24	chroma_loc_info_present_flag	0	u(1)
25	se(chroma_loc_info_present_flag) {		
26	chroma_sample_loc_type_top_field	0	ue(v)
27	chroma_sample_loc_type_bottom_field	0	ue(v)
28	}		
29	se(profile_idc == 'MVC') {		
30	num_views_minus1[i]	0	ue(v)
31	para(i = 0; i <= num_views_minus1; i++) (		
32	timing_info_present_flag[i]	0	u(1)
33	se(timing_info_present_flag[i]) {		
34	num_units_in_tick[i]	0	u(32)
35	time_scale[i]	0	u(32)

36	<i>fixed_frame_rate_flag[i]</i>	0	u(1)
37	}		
38	<i>nal_hrd_parameters_present_flag[i]</i>	0	u(1)
39	<i>se(nal_hrd_parameters_present_flag[i])</i>		
40	<i>hrd_parameters()</i>		
41	<i>vcl_hrd_parameters_present_flag[i]</i>	0	u(1)
42	<i>se(vcl_hrd_parameters_present_flag[i])</i>		
43	<i>hrd_parameters()</i>		
44	<i>se(nal_hrd_parameters_present_flag[i] vcl_hrd_parameters_present_flag[i])</i>		
45	<i>low_delay_hrd_flag[i]</i>	0	u(1)
46	<i>pic_struct_present_flag[i]</i>	0	u(1)
47	<i>bitstream_restriction_flag</i>	0	u(1)
48	<i>se(bitstream_restriction_flag) {</i>		
49	<i>motion_vectors_over_pic_boundaries_flag</i>	0	u(1)
50	<i>max_bytes_per_pic_denom</i>	0	ue(v)
51	<i>max_bits_per_mb_denom</i>	0	ue(v)
52	<i>log2_max_mv_length_horizontal</i>	0	ue(v)
53	<i>log2_max_mv_length_vertical</i>	0	ue(v)
54	<i>num_reorder_frames</i>	0	ue(v)
55	<i>max_dec_frame_buffering</i>	0	ue(v)
56	}		
57	}		
58	}		
59	<i>senão{</i>		
60	<i>timing_info_present_flag</i>	0	u(1)
61	<i>se(timing_info_present_flag) {</i>		
62	<i>num_units_in_tick</i>	0	u(32)
63	<i>time_scale</i>	0	u(32)
64	<i>fixed_frame_rate_flag</i>	0	u(1)
65	}		
66	<i>nal_hrd_parameters_present_flag</i>	0	u(1)
67	<i>se(nal_hrd_parameters_present_flag)</i>		
68	<i>hrd_parameters()</i>		
69	<i>vcl_hrd_parameters_present_flag</i>	0	u(1)
70	<i>se(vcl_hrd_parameters_present_flag)</i>		
71	<i>hrd_parameters()</i>		

72	se(nal_hrd_parameters_present_flag vcl_hrd_parameters_present_flag)		
73	low_delay_hrd_flag	0	u(1)
74	pic_struct_present_flag	0	u(1)
75	bitstream_restriction_flag	0	u(1)
76	se(bitstream_restriction_flag) {		
77	motion_vectors_over_pic_boundaries_flag	0	u(1)
78	max_bytes_per_pic_denom	0	ue(v)
79	max_bits_per_mb_denom	0	ue(v)
80	log2_max_mv_length_horizontal	0	ue(v)
81	log2_max_mv_length_vertical	0	ue(v)
82	num_reorder_frames	0	ue(v)
83	max_dec_frame_buffering	0	ue(v)
84	}		
85	}		
86	}		

num_views_minus1 indica o número total de visualizações no fluxo contínuo de bits.

timing_info_present_flag[i] especifica o valor timing_info_present_flag para a decodificação de *i* visualizações.

num_units_in_tick[i] especifica o valor num_units_in_tick para a decodificação de *i* visualizações.

time_scale[i] especifica o valor time_scale para a decodificação de *i* visualizações.

fixed_frame_rate_flag[i] especifica o valor fixed_frame_rate_flag para a decodificação de *i* visualizações.

nal_hrd_parameters_present_flag[i] especifica o valor nal_hrd_parameters_present_flag para a decodificação de *i* visualizações.

vcl_hrd_parameters_present_flag[i] especifica o valor vcl_hrd_parameters_present_flag para a decodificação de *i* visualizações.

low_delay_hrd_flag[i] especifica o valor low_delay_hrd_flag para a decodificação de *i* visualizações.

pic_struct_present_flag[i] especifica o valor pic_struct_present_flag para a decodificação de *i* visualizações.

Os parâmetros timing_info_present_flag, num_units_in_tick, time_scale, nal_hrd_parameters_present_flag, vcl_hrd_parameters_present_flag, low_delay_hrd_flag e pic_struct_present_flag têm a mesma semântica do AVC.

As figuras 4 até 13 mostra vários fluxogramas para a gravação ou leitura da sintaxe em particular que é identificada.

São aqui descritas diversas implementações que, por exemplo, fornecem regras pa-

ra a gravação e/ou leitura de uma descrição de um HRD para MVC. Isto permite que um HRD seja definido e usado na MVC.

Entretanto, recursos e aspectos das implementações descritas também podem ser adaptados para outras implementações. Por exemplo, da forma supraindicada, um HRD pode ser fornecido usando os recursos expostos para outros padrões. Adicionalmente, um HRD pode ser fornecido para MVC usando variações das modalidades supradescritas. Tais variações podem incluir, por exemplo, implementações que usam outra sintaxe de alto nível, que usam sintaxe que não é de alto nível, que fornecem parâmetros do HRD para outros IOPs. Dessa maneira, embora implementações aqui descritas possam ser descritas no contexto da sintaxe de alto nível para H.264, implementando de uma das três implementações principais aqui descritas, tais descrições não devem, de maneira nenhuma, ser tomadas como limitantes dos recursos e conceitos em tais implementações ou contextos.

As implementações aqui descritas podem ser implementadas, por exemplo, em um método ou processo, um aparelho ou um programa de software. Mesmo se somente discutidos no contexto de uma única forma de implementação (por exemplo, discutidos somente como um método), a implementação ou recursos discutidos também podem ser implementados em outras formas (por exemplo, um aparelho ou programa). Um aparelho pode ser implementado, por exemplo, em hardware, software e software embarcado apropriados. Os métodos podem ser implementados, por exemplo, em um aparelho, tais como, por exemplo, um computador ou outro dispositivo de processamento. Adicionalmente, os métodos podem ser implementados por instruções que são realizadas por um dispositivo de processamento ou outro aparelho, e tais instruções podem ser armazenadas em uma mídia legível por computador, tais como, por exemplo, um CD ou outro dispositivo de armazenamento legível por computador, ou um circuito integrado.

Como deve ficar evidente aos versados na técnica, implementações também podem produzir um sinal formatado para portar informação que pode ser, por exemplo, armazenado e transmitido. A informação pode incluir, por exemplo, instruções para realizar um método ou dados produzidos por uma das implementações descritas. Por exemplo, um sinal pode ser formatado para portar como dados as regras para gravação ou leitura da sintaxe de uma modalidade descrita, ou para portar como dados a sintaxe real gravada por uma modalidade descrita.

Adicionalmente, muitas implementações podem ser implementadas tanto em um codificador quanto em um decodificador, ou em cada um deles.

Adicionalmente, outras implementações são contempladas por esta invenção. Por exemplo, implementações adicionais podem ser criadas pela combinação, deleção, modificação ou complementação dos vários recursos das implementações divulgadas.

A descrição exposta fornece algumas das várias implementações. Não pretende-se

que seja completa, mas que forneça meramente uma descrição resumida de um pequeno número das muitas implementações possíveis.

REIVINDICAÇÕES

1. Decodificador de referência hipotética para codificação de vídeo multivisualização, **CARACTERIZADO** pelo fato de que regras relacionadas a um ou mais parâmetros e informação de restrição de fluxo contínuo de bits são especificadas para cada ponto interoperável na MVC.

2. Parâmetros e informação de restrição do fluxo contínuo de bits relacionados ao decodificador de referência hipotética para codificação de vídeo multivisualização, **CARACTERIZADOS** pelo fato de que regras são especificadas somente para um conjunto de pontos interoperáveis, e outros conjuntos são derivados a partir do conjunto especificado.

3. Parâmetros e informação de restrição do fluxo contínuo de bits relacionados ao decodificador de referência hipotética para codificação de vídeo multivisualização, **CARACTERIZADOS** pelo fato de que regras são especificadas somente para o pior caso da decodificação de qualquer número de visualizações.

4. Método de codificação de vídeo multivisualização em um decodificador de referência hipotética, **CARACTERIZADO** pelo fato de que uma mensagem SEI aninhada é definida para especificar informação para cada visualização ou para a combinação de diferentes visualizações.

5. Decodificador de referência hipotética, de acordo com as reivindicações 1, 2 ou 3, **CARACTERIZADO** pelo fato de que todos os parâmetros e informação de restrição do fluxo contínuo de bits relacionados ao HRD são definidos na sintaxe de alto nível.

6. Decodificador de referência hipotética, de acordo com as reivindicações 1, 2 ou 3, **CARACTERIZADO** pelo fato de que a sintaxe de alto nível é usada para descrever as regras, e em que a sintaxe de alto nível pode ser composta por pelo menos um do conjunto de parâmetro de sequência (SPS), da informação de usabilidade de vídeo (VUI), do conjunto de parâmetro da imagem (PPS), do cabeçalho da fatia, do cabeçalho do nal_unit ou da mensagem de informação de melhoria complementar (SEI).

7. Decodificador de referência hipotética, de acordo com as reivindicações 1, 2 ou 3, **CARACTERIZADO** pelo fato de que o parâmetro profile_idc é usado para diferenciar o caso de codificação de vídeo multivisualização ou o caso compatível com AVC.

8. Decodificador de referência hipotética, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que a implementação faz laço através de todas as visualizações e suas combinações.

9. Decodificador de referência hipotética, de acordo com a reivindicação 2, relacionado aos parâmetros e informação de restrição do fluxo contínuo de bits relacionados ao HRD para codificação de vídeo multivisualização, **CARACTERIZADO** pelo fato de que as regras são especificadas somente para decodificação de uma única visualização, e os parâmetros e a informação de restrição do fluxo contínuo de bits relacionados ao decodificador

de referência hipotética para decodificação de mais de uma visualização são derivados daqueles da única visualização.

10. Decodificador de referência hipotética, de acordo com a reivindicação 9, **CHARACTERIZADO** pelo fato de que o decodificador faz laço através de cada visualização.

5 11. Decodificador de referência hipotética, de acordo com a reivindicação 3, **CHARACTERIZADO** pelo fato de que também compreende fazer laço através de cada uma das visualizações no fluxo contínuo de bits.

10 12. Método para especificar regras para criar um decodificador de referência hipotética para codificação de vídeo multivisualização, **CHARACTERIZADO** pelo fato de que as regras podem ser em sintaxe de alto nível.

13. Aparelho, **CHARACTERIZADO** pelo fato de que é para gravar sintaxe apropriada para descrever parâmetros para um decodificador de referência hipotética para codificação de vídeo multivisualização.

15 14. Aparelho, **CHARACTERIZADO** pelo fato de que é para ler sintaxe apropriada que descreve parâmetros para um decodificador de referência hipotética para codificação de vídeo multivisualização.

20 15. Mídia legível por computador que armazena regras para especificar um decodificador de referência hipotética para codificação de vídeo multivisualização, **CHARACTERIZADA** pelo fato de que tais regras armazenadas podem ser acessadas por um codificador para descrever parâmetros para o HRD.

25 16. Mídia legível por computador que armazena regras para interpretar informação recebida, **CHARACTERIZADA** pelo fato de que a informação descreve parâmetros de um decodificador de referência hipotética para codificação de vídeo multivisualização, e em que tais regras armazenadas podem ser acessadas por um decodificador para interpretar a sintaxe recebida de um codificador.

17. Método de codificação de um fluxo contínuo de bits, **CHARACTERIZADO** pelo fato de que o fluxo contínuo de bits é formatado para incluir informação que descreve parâmetros de um decodificador de referência hipotética para codificação de vídeo multivisualização.

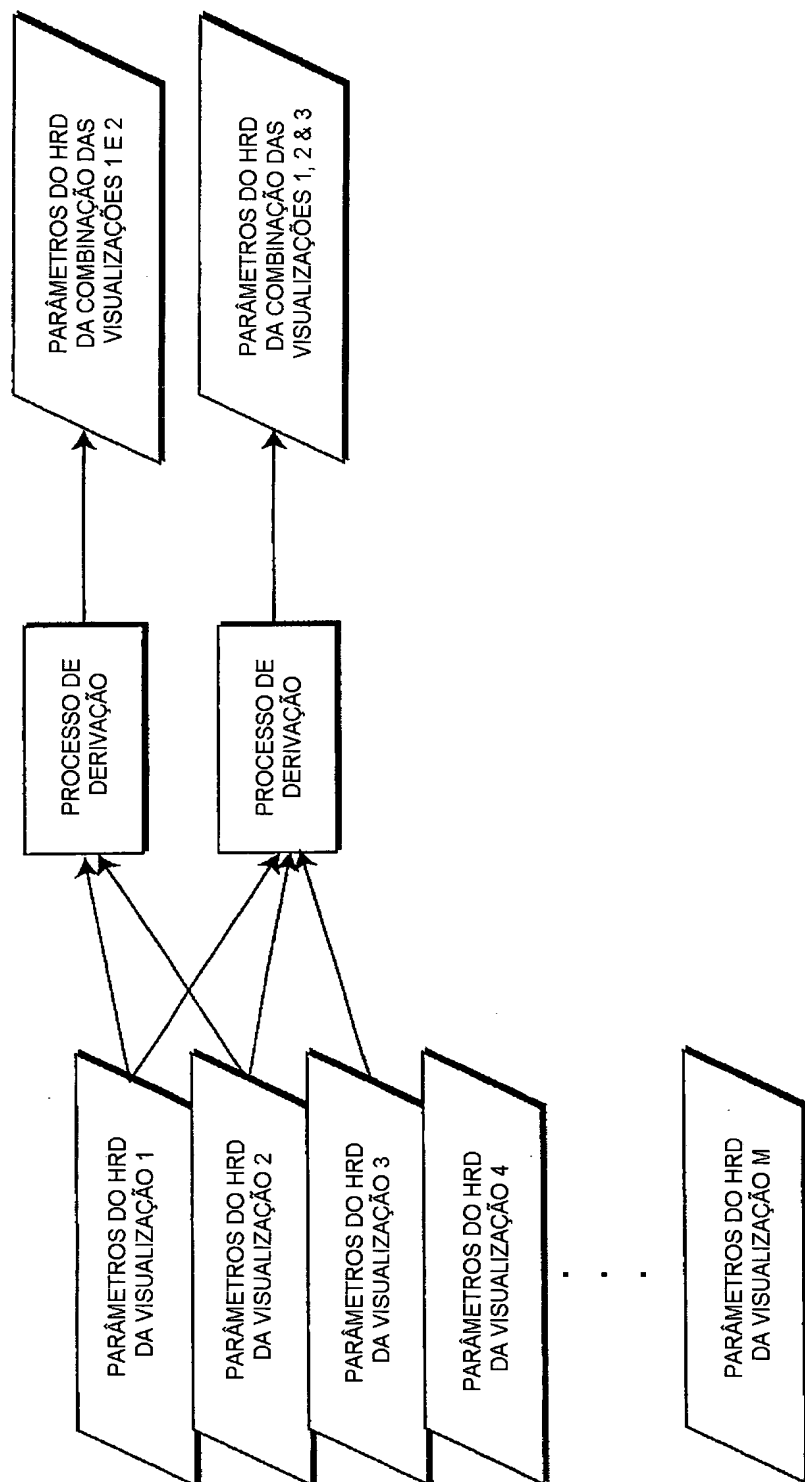


FIG. 1

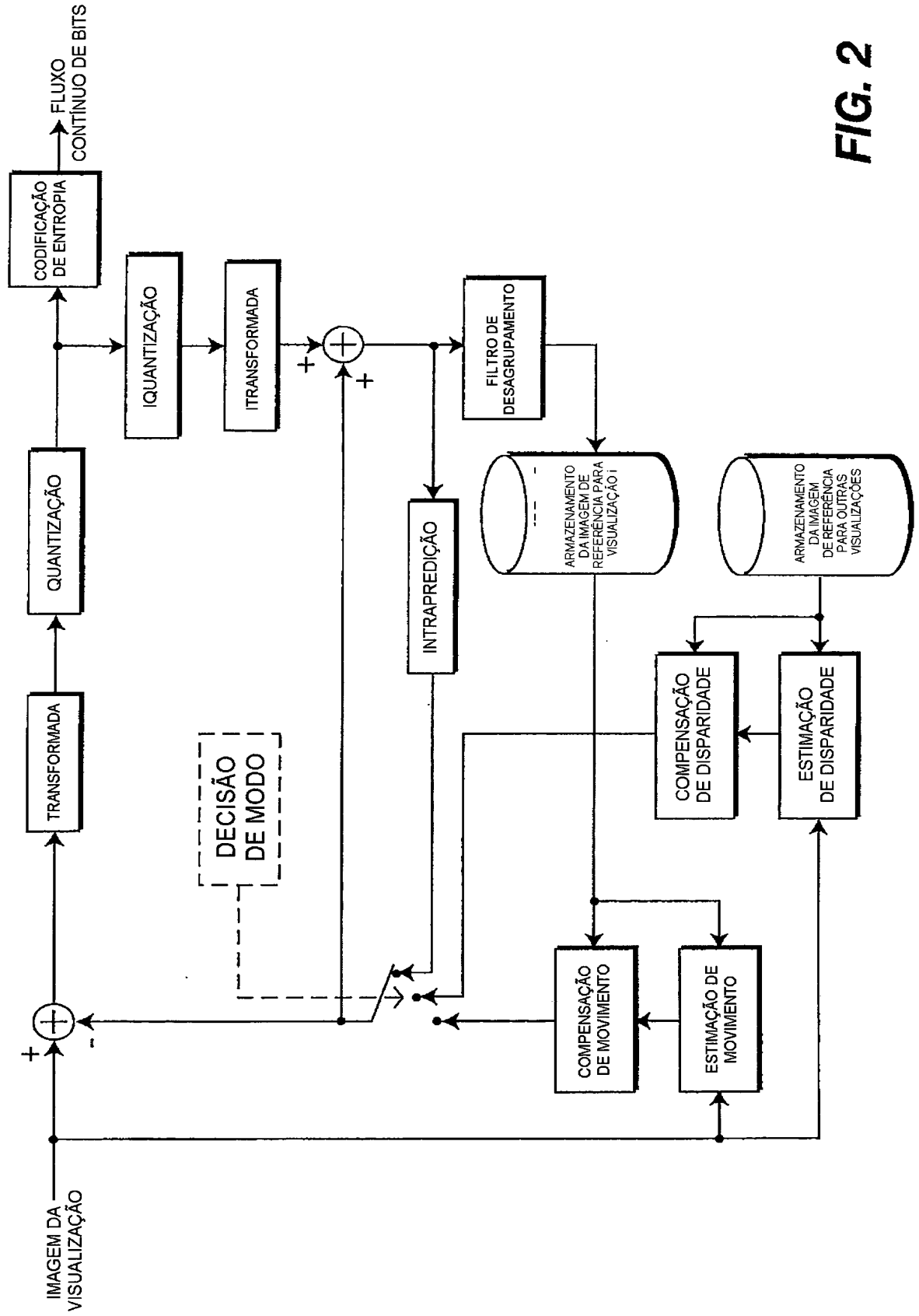


FIG. 2

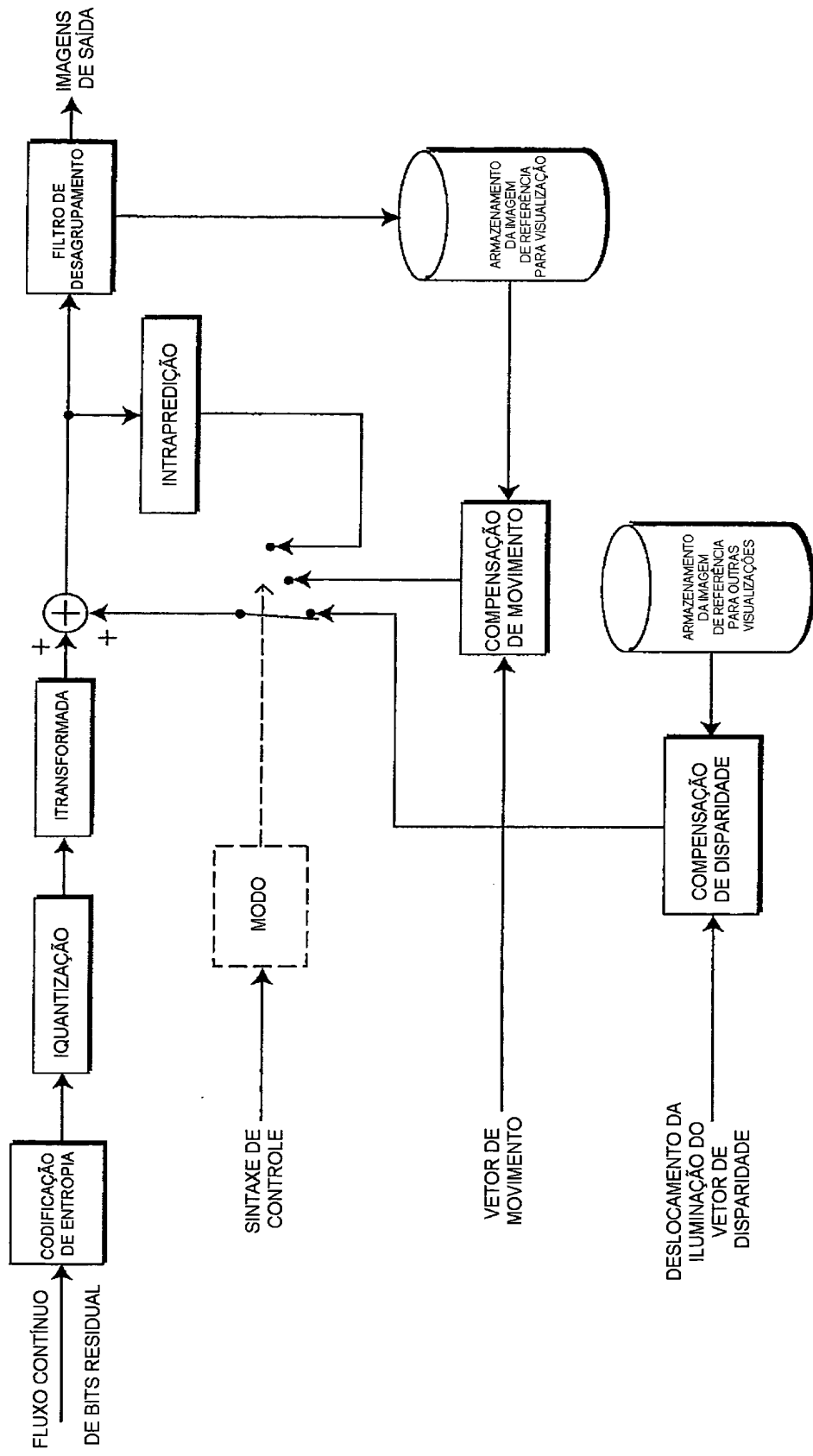


FIG. 3

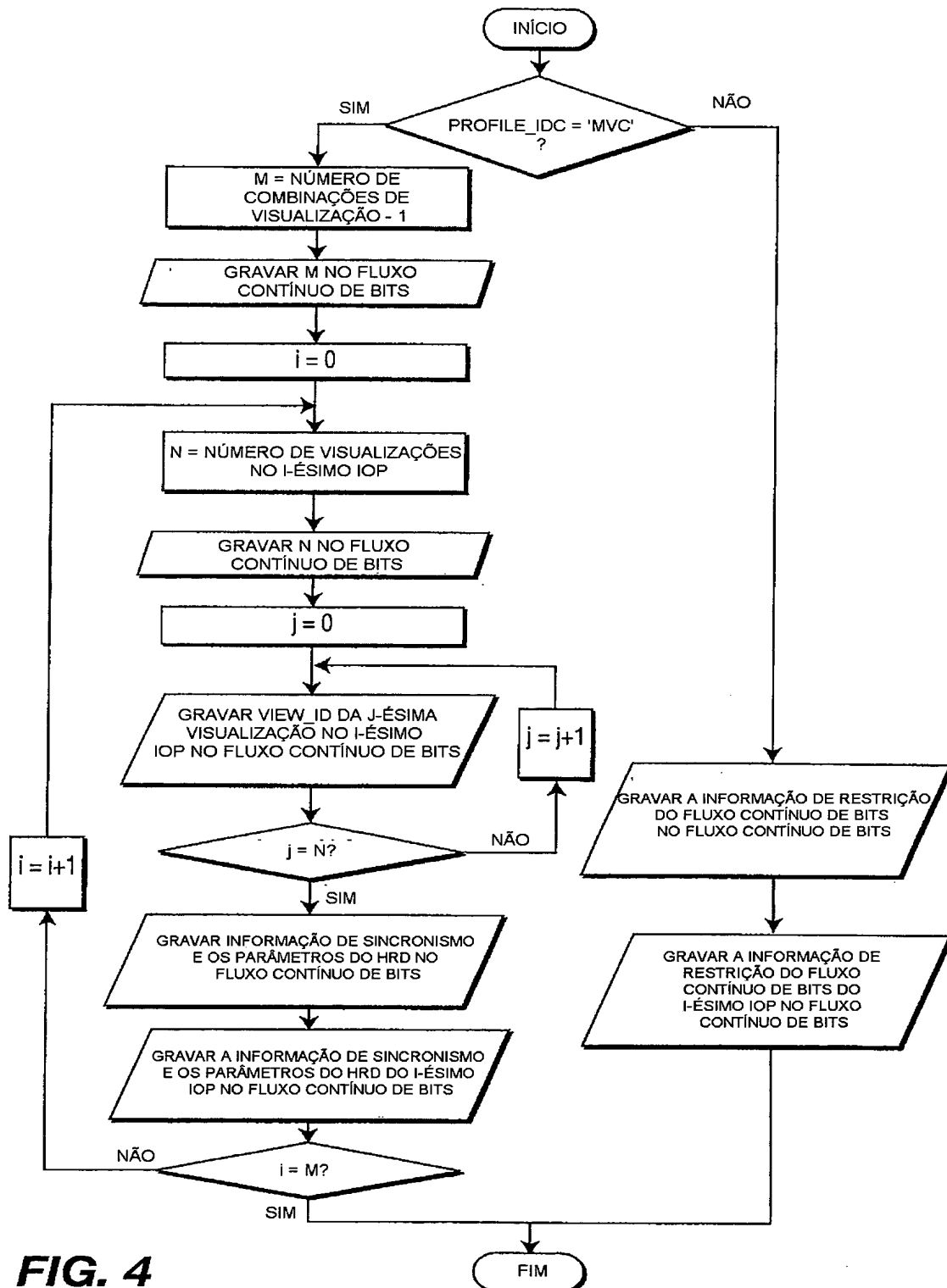


FIG. 4

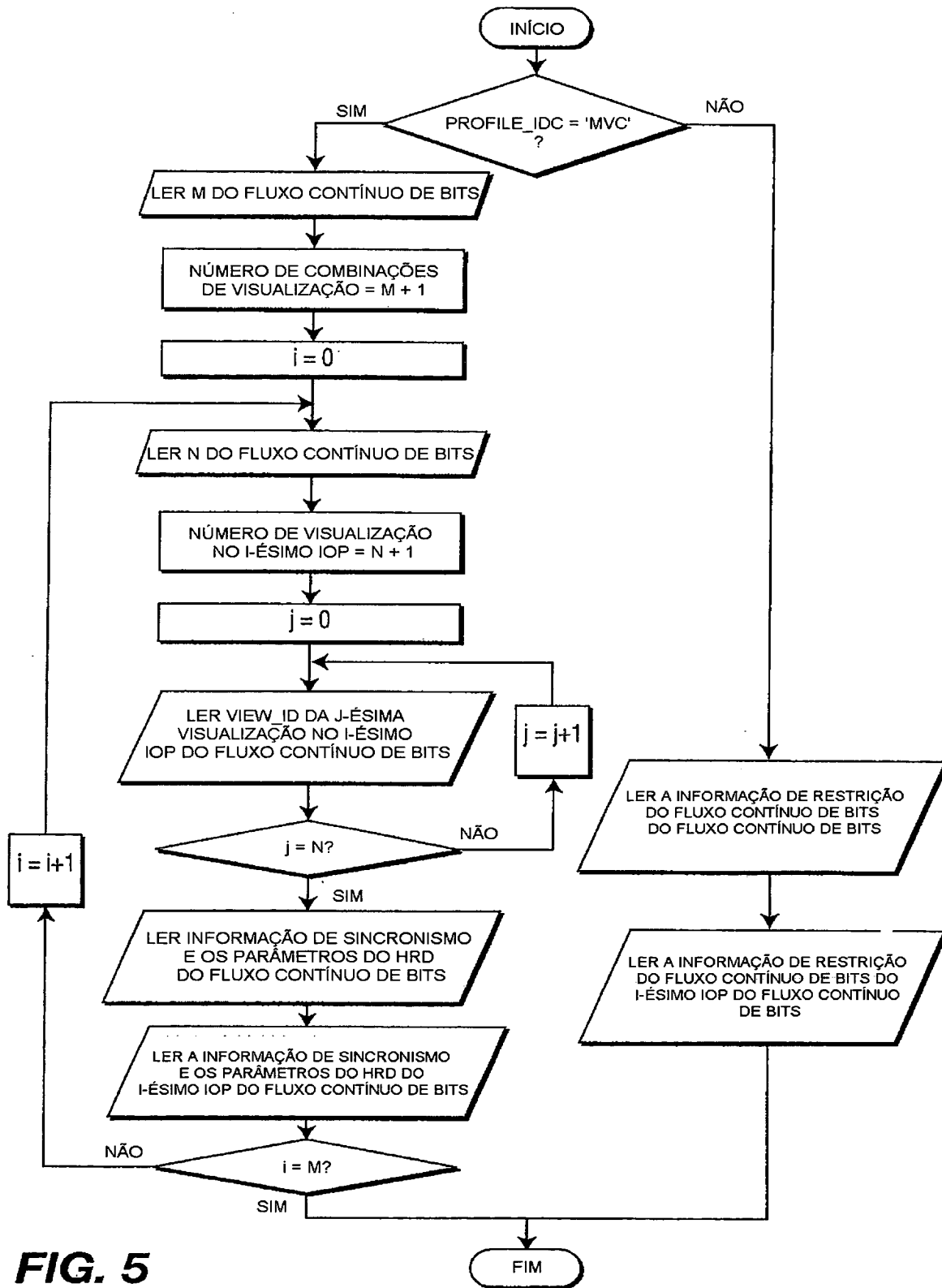
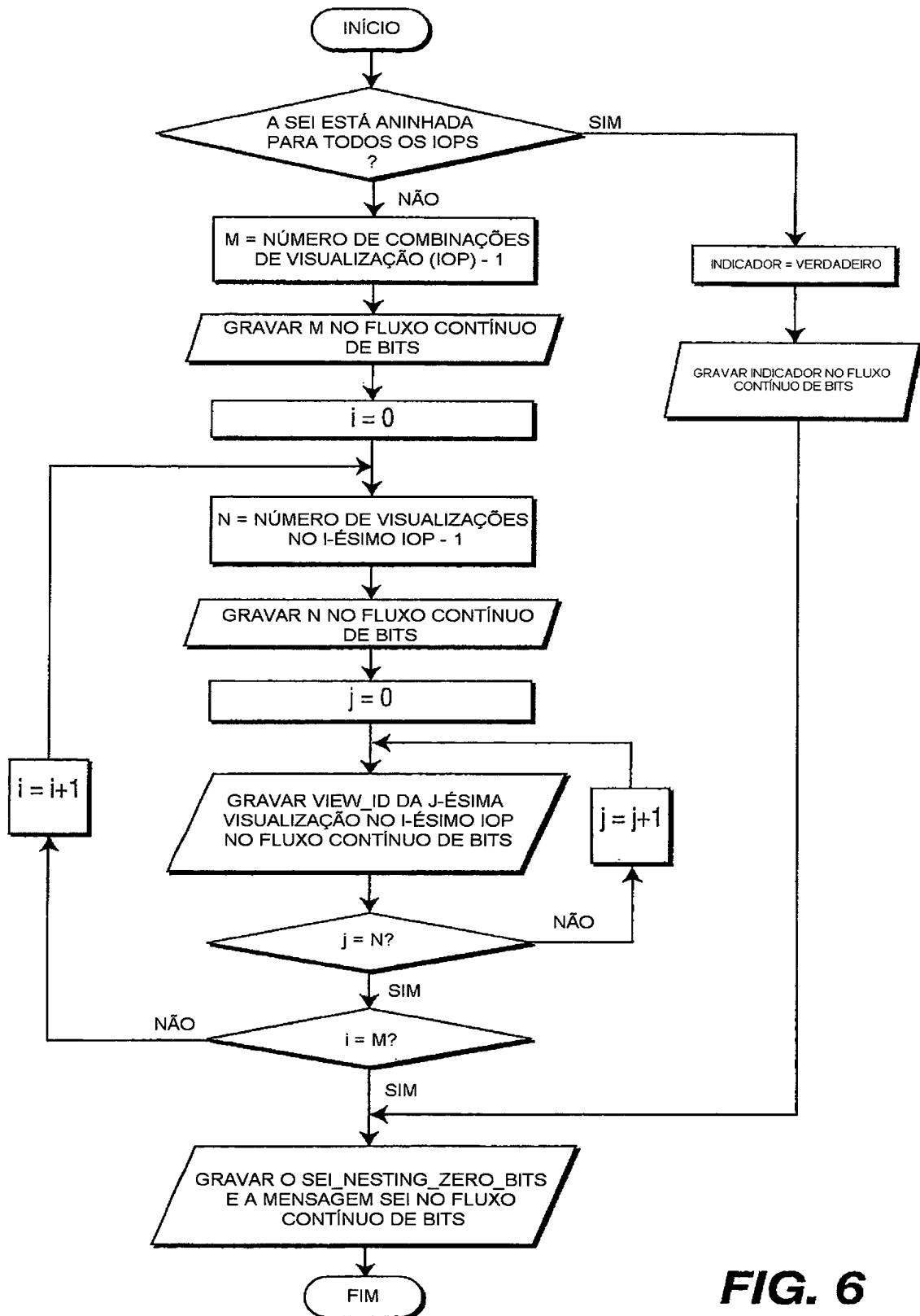


FIG. 5

**FIG. 6**

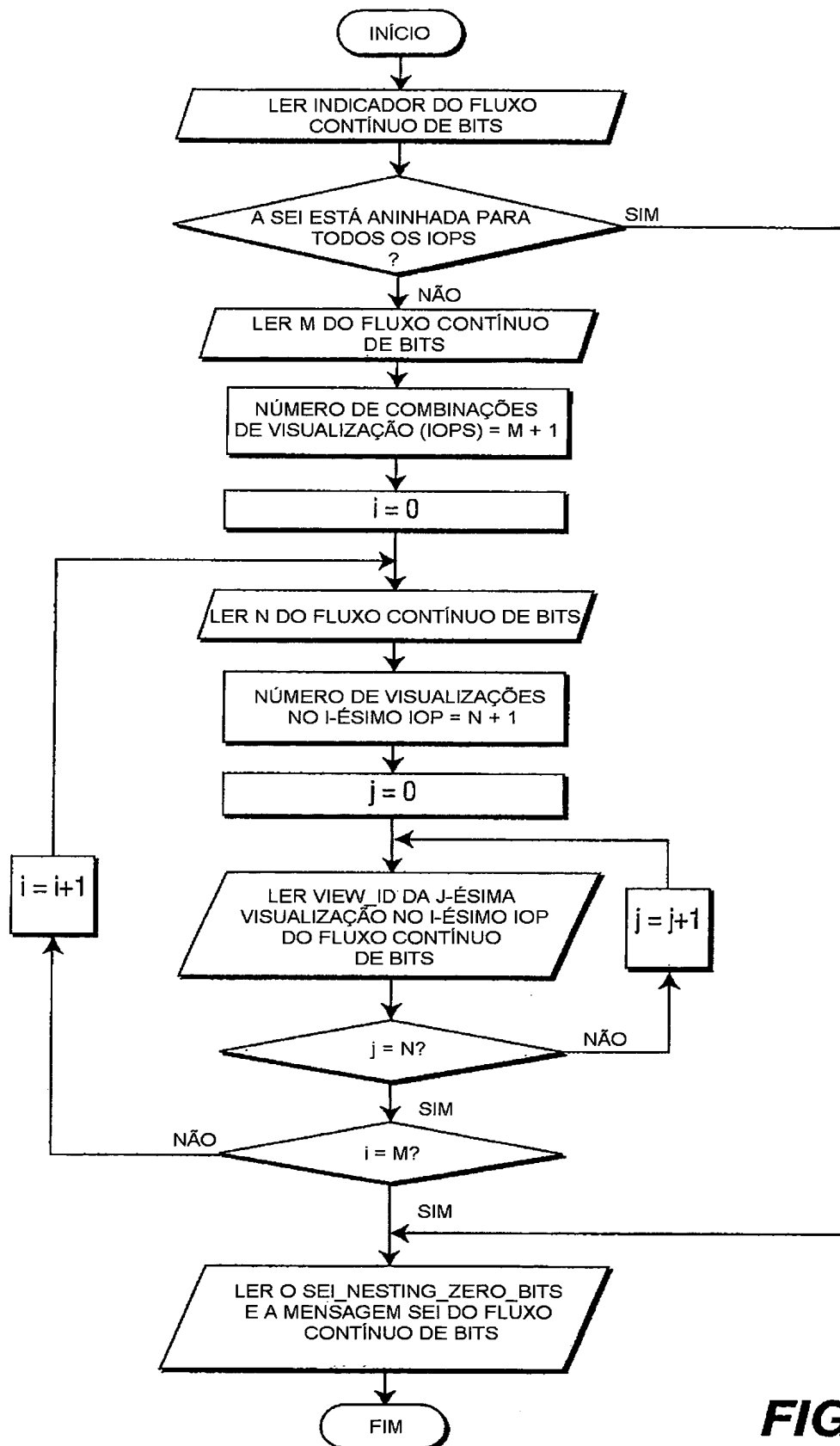
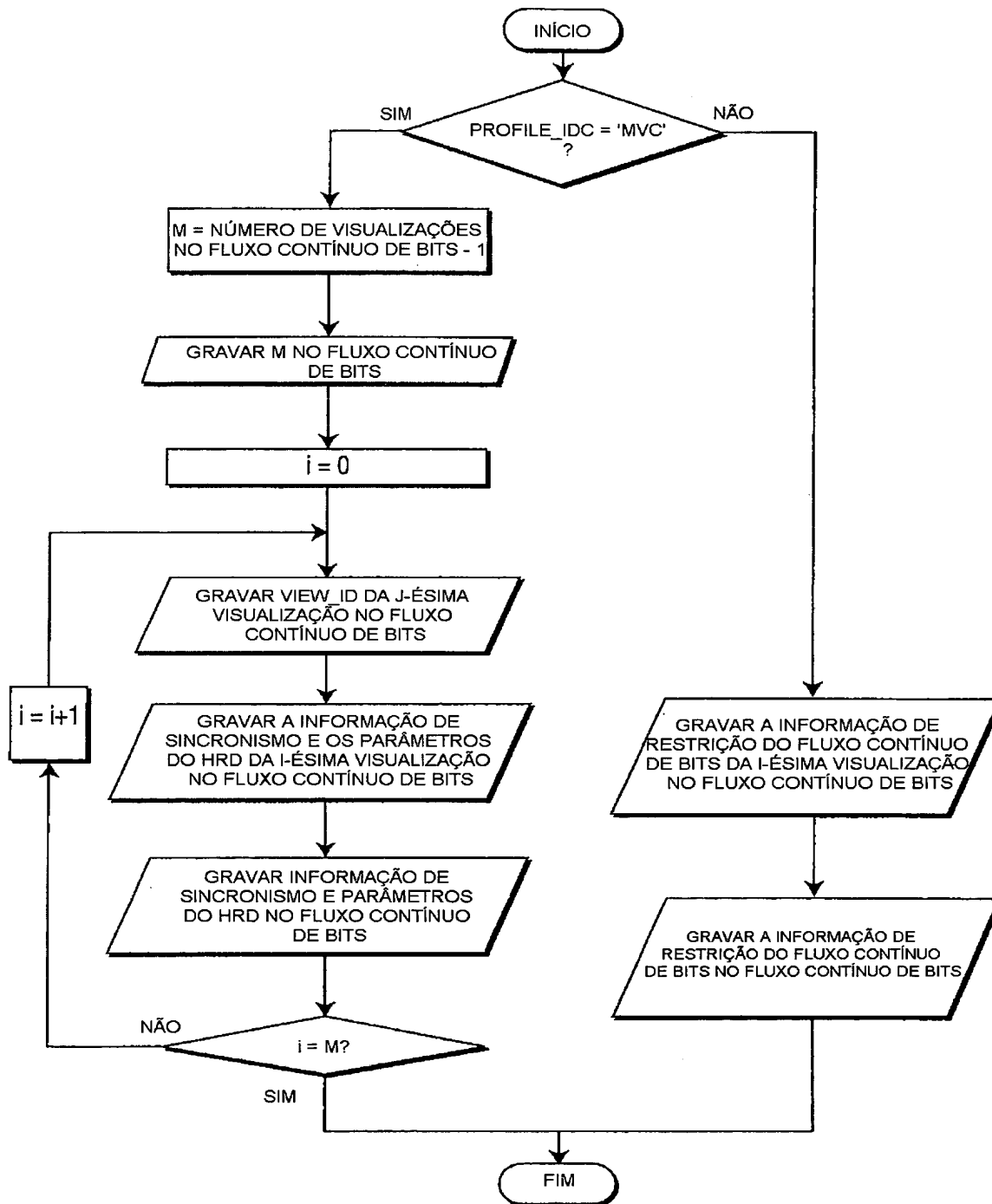
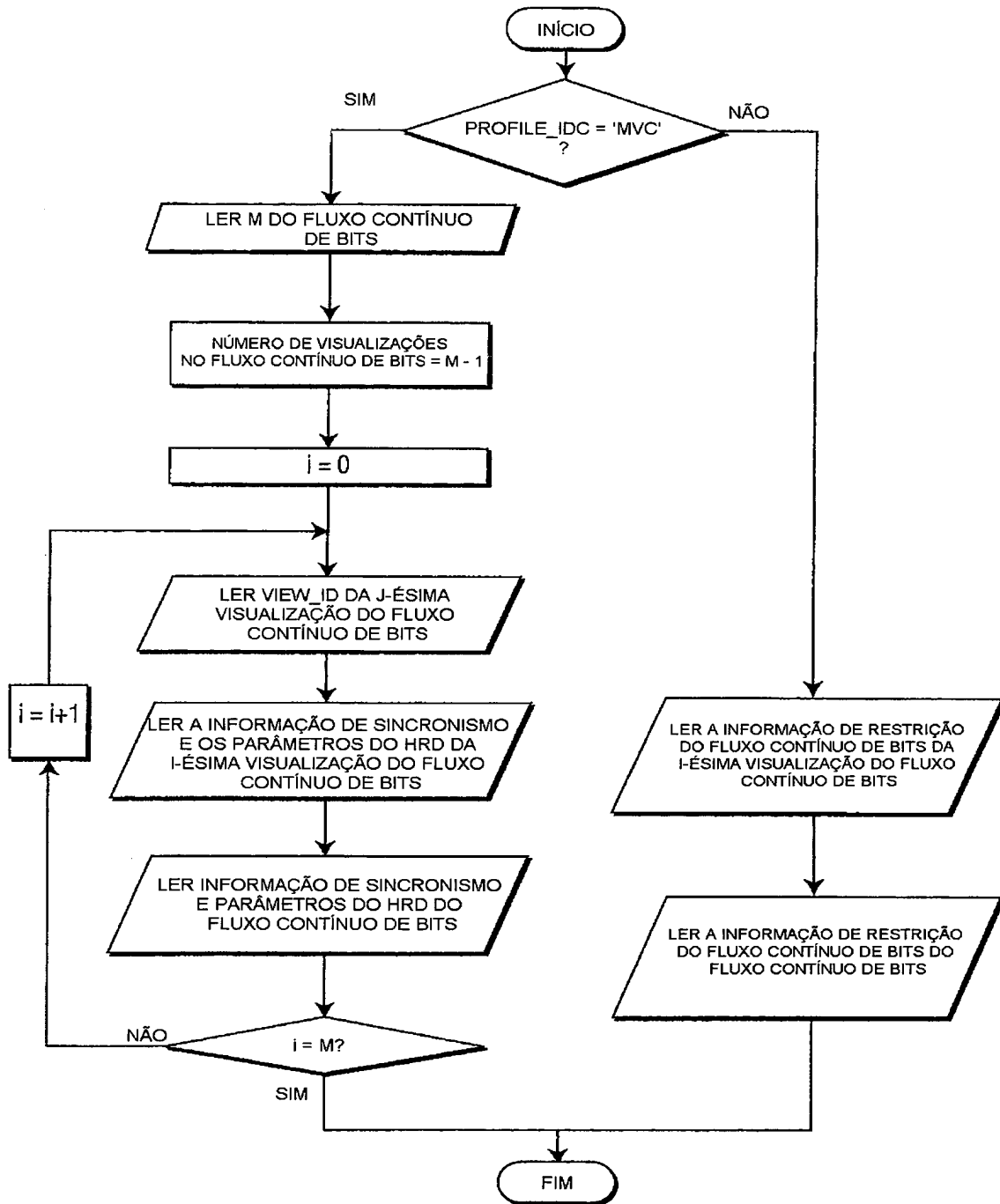
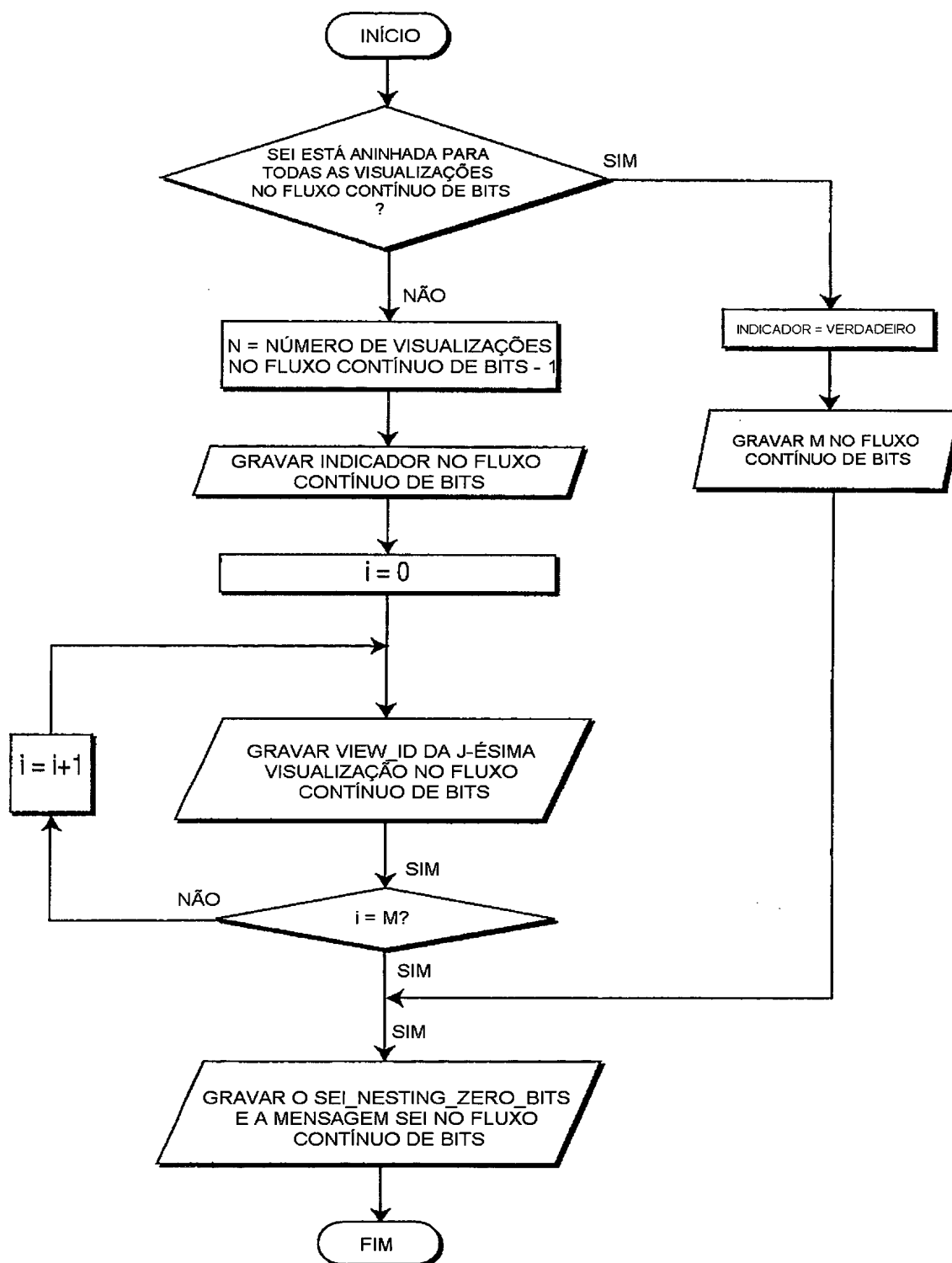
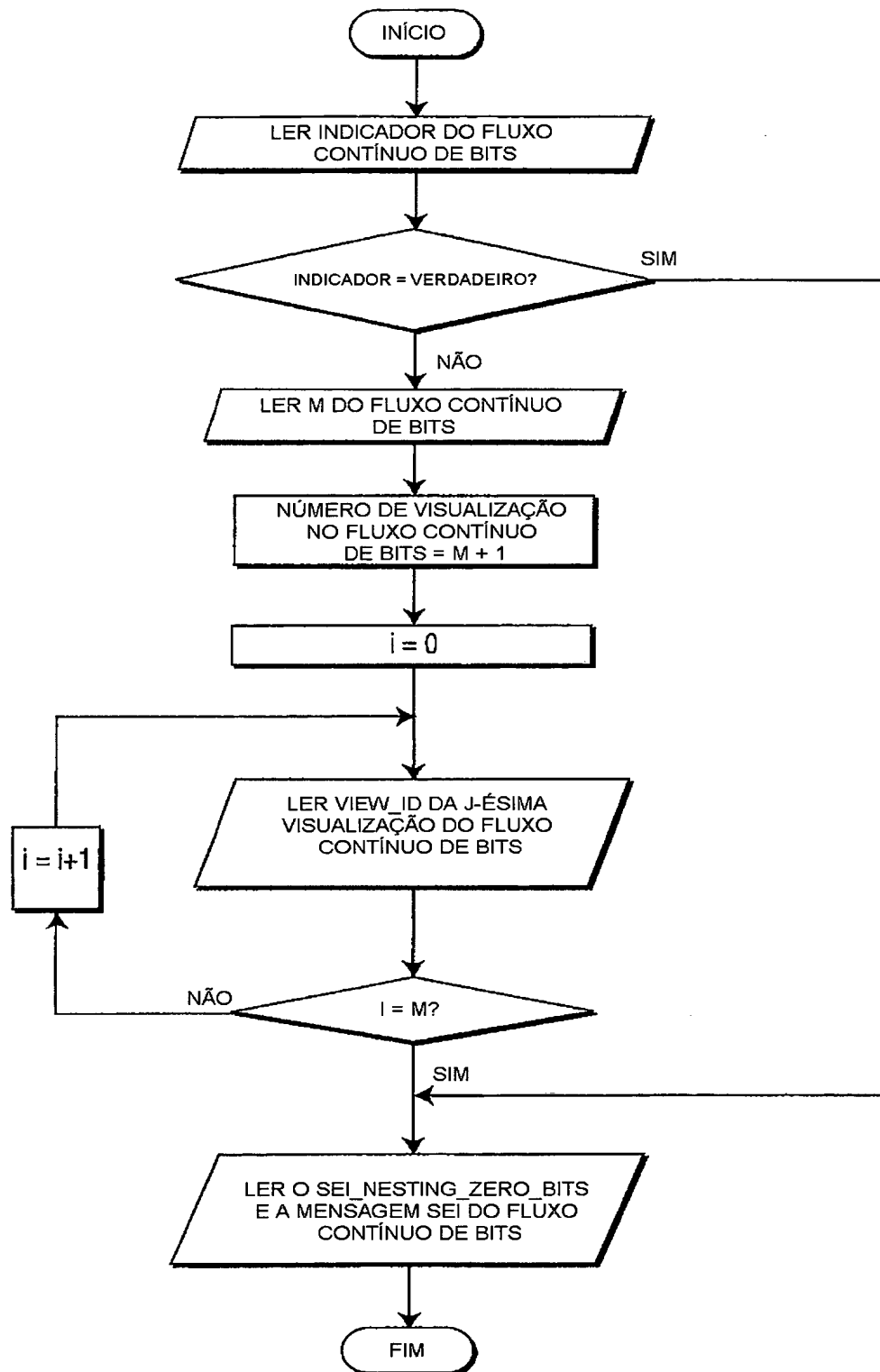


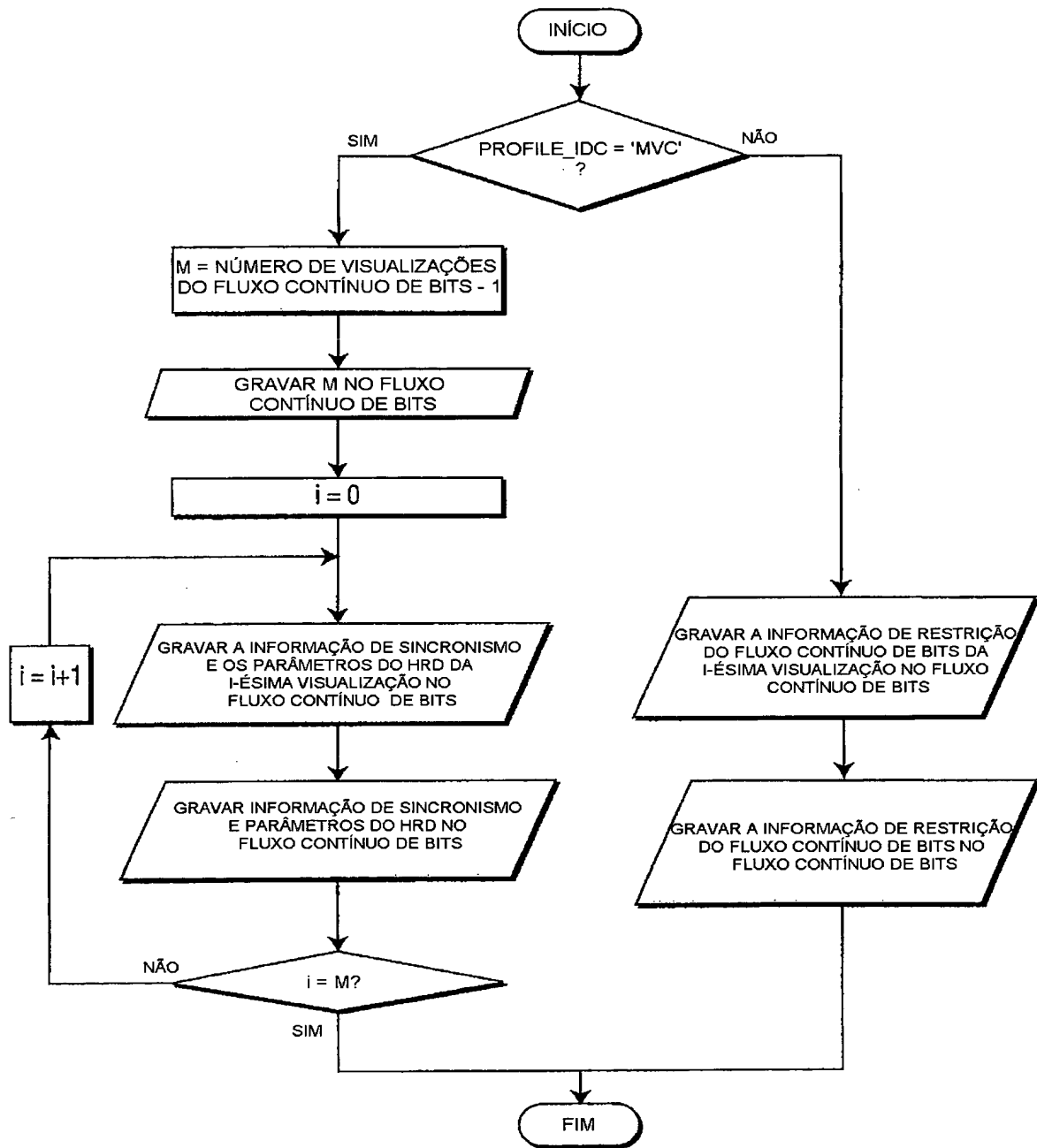
FIG. 7

**FIG. 8**

**FIG. 9**

**FIG. 10**

**FIG. 11**

**FIG. 12**

RESUMO

“DECODIFICADOR DE REFERÊNCIA HIPOTÉTICA PARA CODIFICAÇÃO DE VÍDEO MULTIVISUALIZAÇÃO”

São descritos um método e um aparelho para definir um Decodificador de Referência Hipotética (HRD) para Codificação de Vídeo Multivisualização (MVC) que definem restrições para cada ponto de interoperabilidade possível. Cada combinação de M visualizações denota um ponto de interoperabilidade. Os princípios da presente invenção também permitem que restrições sejam definidas para um conjunto de pontos interoperáveis e que outros conjuntos sejam derivados do conjunto definido. O método também permite que as regras dos pontos de interoperabilidade sejam derivadas da informação na sintaxe do fluxo contínuo de bits.