

FIG. 1

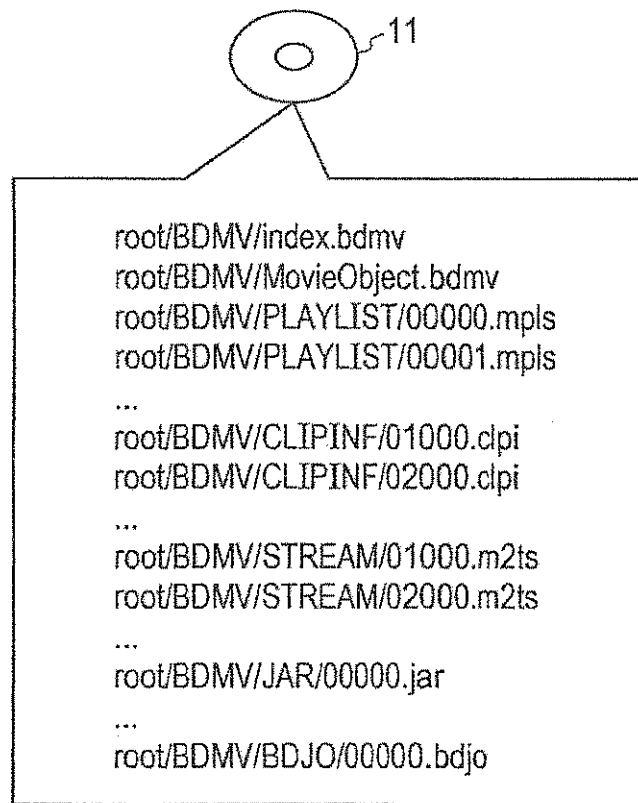


FIG. 2

Title NO.	TYPE AND NO. OF OBJECT TO EXECUTE
First Play	MovieObject#1
Top menu	MovieObject#2
Title#1	BD-J Object#1
***	***
Title#N	MovieObject#M

FIG. 3

POINT-IN-TIME OF PLAYING	OFFSET DIRECTION	OFFSET VALUE	PLANE
00:00:14	NEAR SIDE	32	PG
00:00:16	NEAR SIDE	24	IG
...			

FIG. 4

```

xxxxx.mpIs{
...
AppInfoPlaylist()
Playlist()
PlaylistMark()
ExtensionData()
...
}

ExtensionData() {
...
number_of_ext_data_entries
for()
{
    ext_data_entry()
}
...
data_block()
}

ext_data_entry() {
...
ID1 = 0x0001
ID2 = 0x0003
...
}

data_block() {
    offset_metadata() {
        length
        number_of_metadata_block_entries
        for(int i = 0; i < number_of_metadata_block_entries
            i++) {
            ref_to_PlayItem_id[i]
            stream_type[i]
            ref_to_stream_id[i]
            reserved_for_future_use
            number_of_offset_metadata[i]
            for(int j = 0; j < number_of_offset_metadata;
                j++) {
                offset_pts[j]
                offset_flag[j]
                offset_value[j]
                reserved_for_future_use
            }
        }
    }
}

```

FIG. 5

```

data_block()
{
    offset_metadata() {
        length
        number_of_metadata_block_entries
        for (int i = 0; i < number_of_metadata_block_entries
            i++) {
            ref_to_PlayItem_id[i]
            plane_identifier[i]
            reserved_for_word_align
            ref_to_stream_id[i]
            reserved_for_future_use
            number_of_offset_metadata[i]
            for (int j = 0; j < number_of_offset_metadata;
                j++) {
                offset_pts[j]
                offset_flag[j]
                offset_value[j]
                reserved_for_word_align
                reserved_for_future_use
            }
        }
        reserved_for_future_use
    }
}

```

32bits length of this offset_metadata()
16bits

16bits playitem_id which is referred by this metadata
1bit 0:IG plane 1:PG plane
7bits
8bits stream_id (if BDJ == 0xFF)
16bits
16bits

32bits time_stamp
1bit offset direction
7bits actual value
8bits
16bits

16bits

FIG. 6

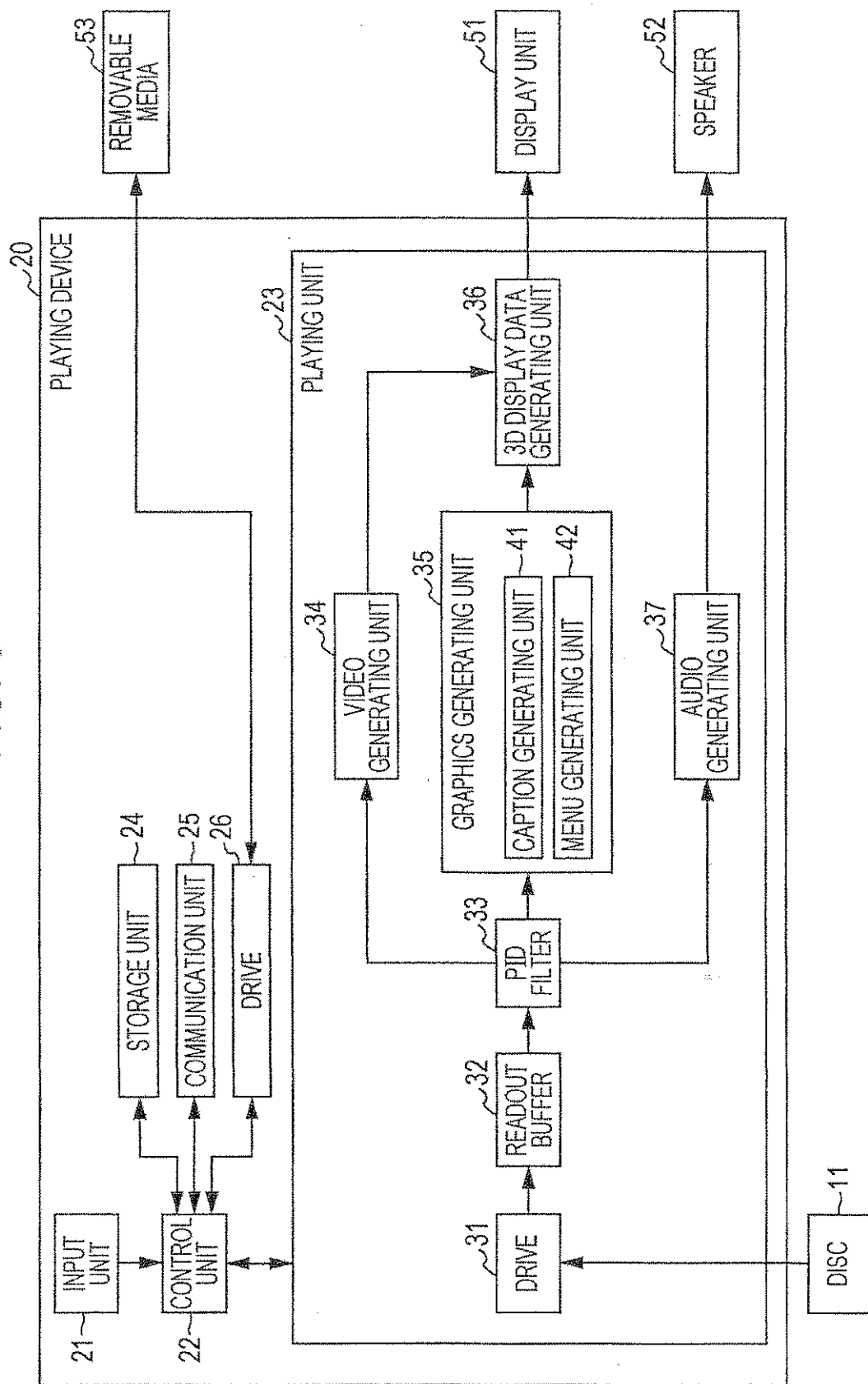


FIG. 7

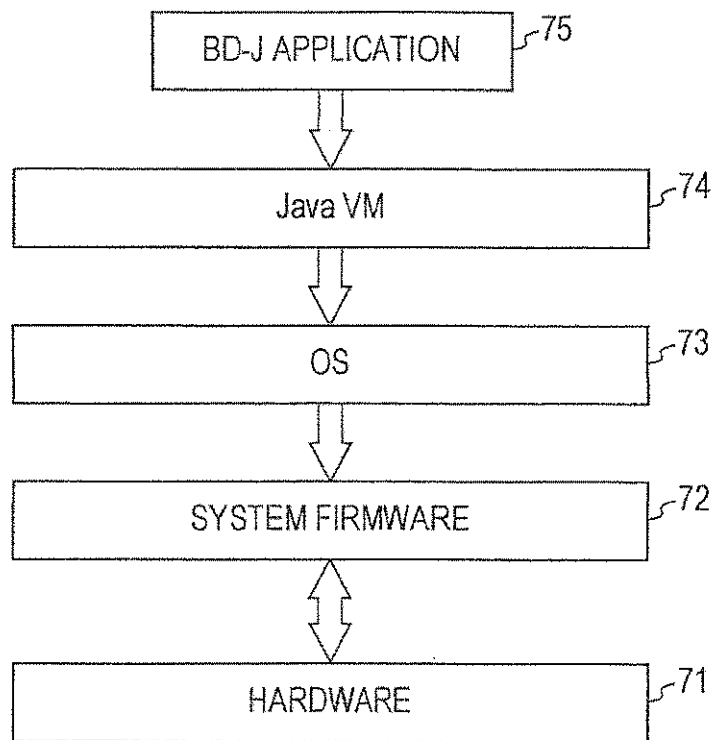


FIG. 8

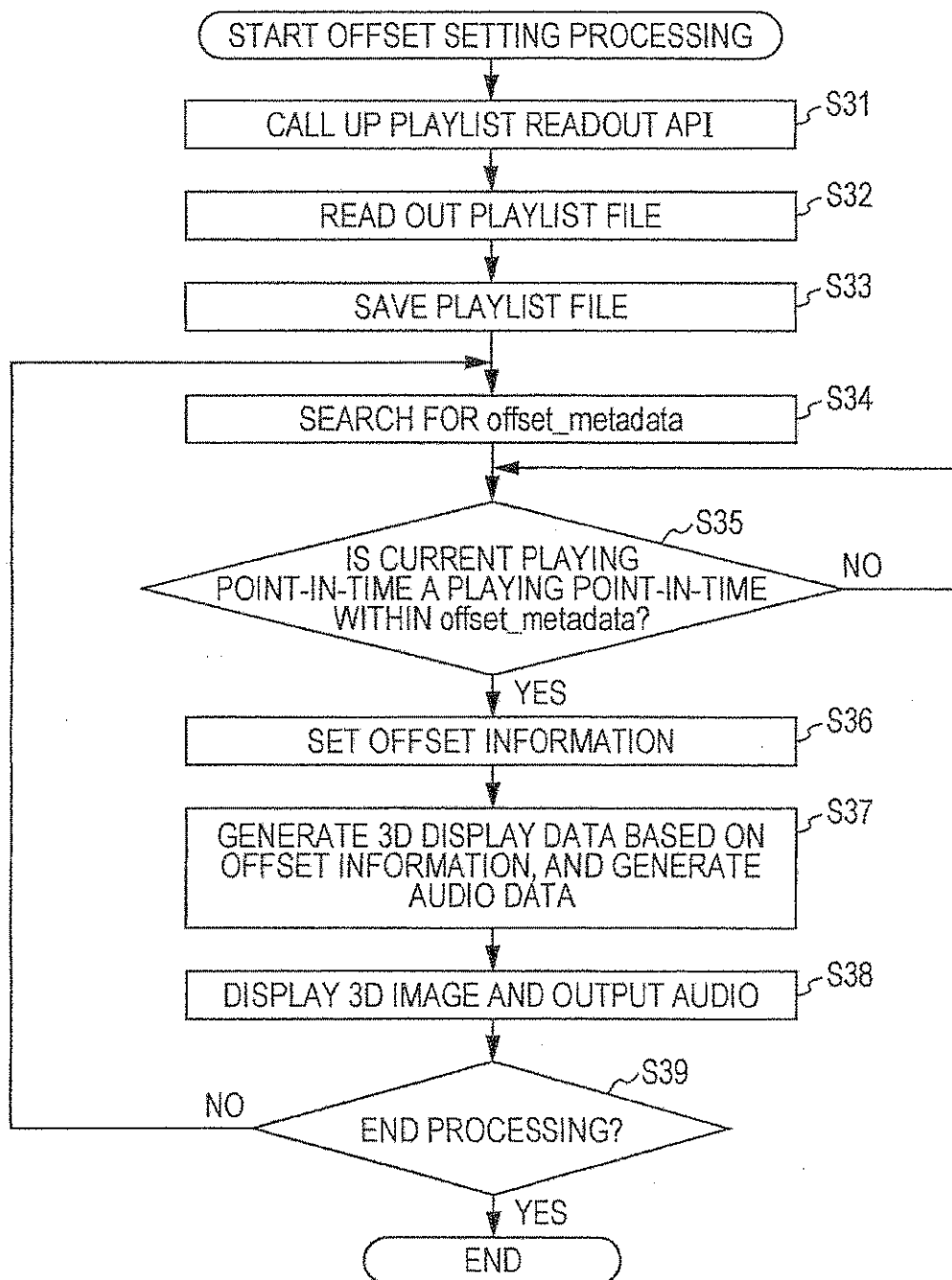


FIG. 9

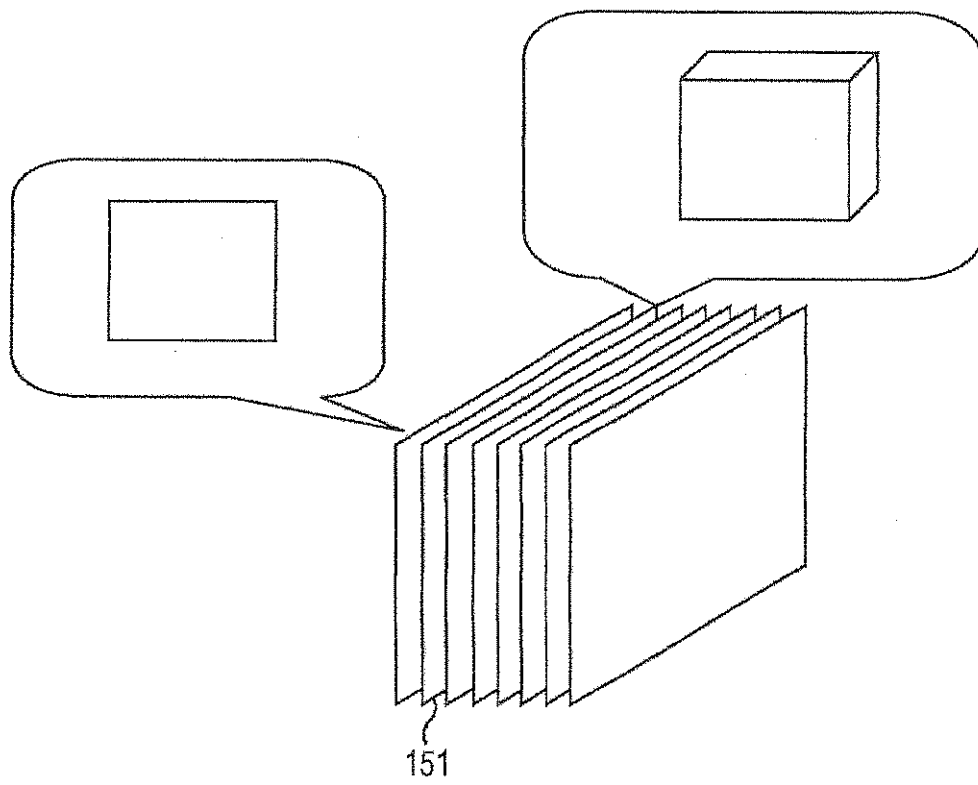


FIG. 10

Syntax	No. of bits	Mnemonic
offset_metadata() {		
length	32	uimsbf
for (i=0; i<number_of_plane_items; i++) {		
number_of_plane_offsets[i]	16	uimsbf
for (j=0; j<number_of_plane_offsets[i]; j++) {		
plane_offset_pts[j]	32	uimsbf
is_IQ_Offsets	1	bslbf
if (is_IQ_Offsets==1) {		
IQ_Plane_offset_direction[j]	1	bslbf
IQ_Plane_offset_value[j]	6	uimsbf
IQ_Plane_sub_offset_rate[j]	5	uimsbf
number_of_IQ_Plane_sub_offsets[j]	5	uimsbf
for (k=0; k<number_of_IQ_Plane_sub_offsets[j]; k++) {		
IQ_Plane_sub_offset_direction[k]	1	bslbf
IQ_Plane_sub_value[k]	6	uimsbf
}		
}		
}		
is_PG_Offsets	1	bslbf
if (is_PG_Offsets==1) {		
PG_Plane_offset_direction[j]	1	bslbf
PG_Plane_offset_value[j]	6	uimsbf
PG_Plane_sub_offset_rate[j]	5	uimsbf
number_of_PG_Plane_sub_offsets[j]	5	uimsbf
for (k=0; k<number_of_PG_Plane_sub_offsets[j]; k++) {		
PG_Plane_sub_offset_direction[k]	1	bslbf
PG_Plane_sub_value[k]	6	uimsbf
}		
}		
}		

FIG. 11

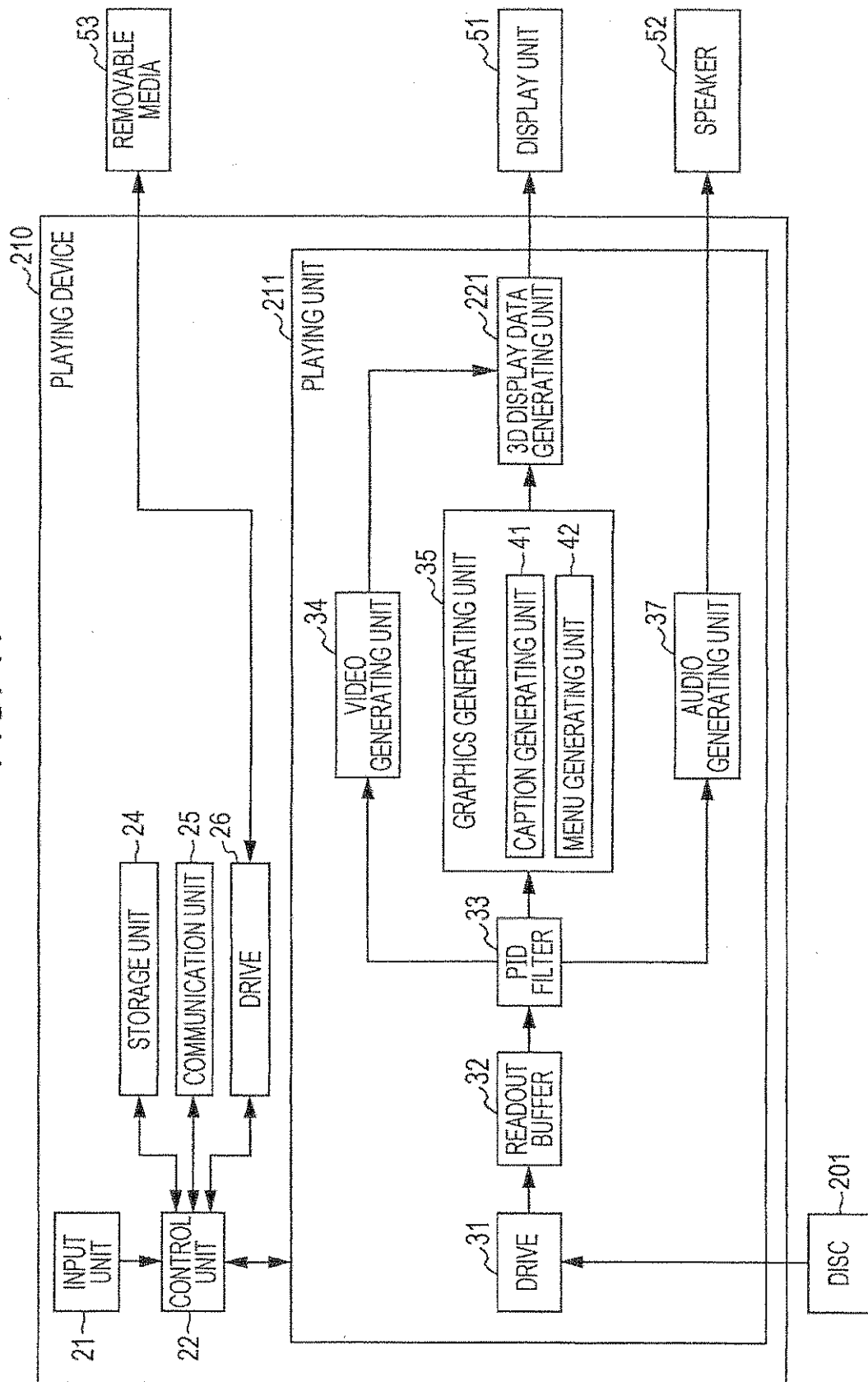


FIG. 12

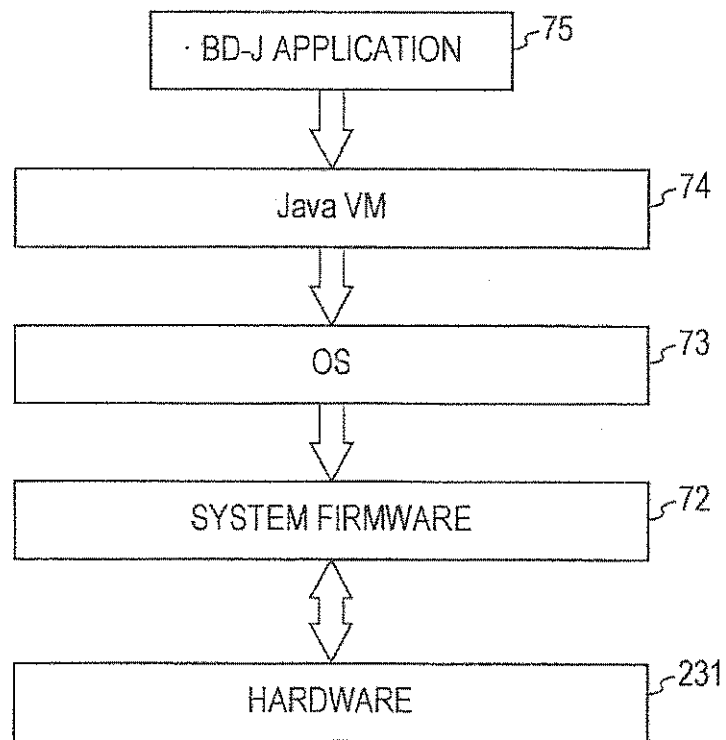


FIG. 13

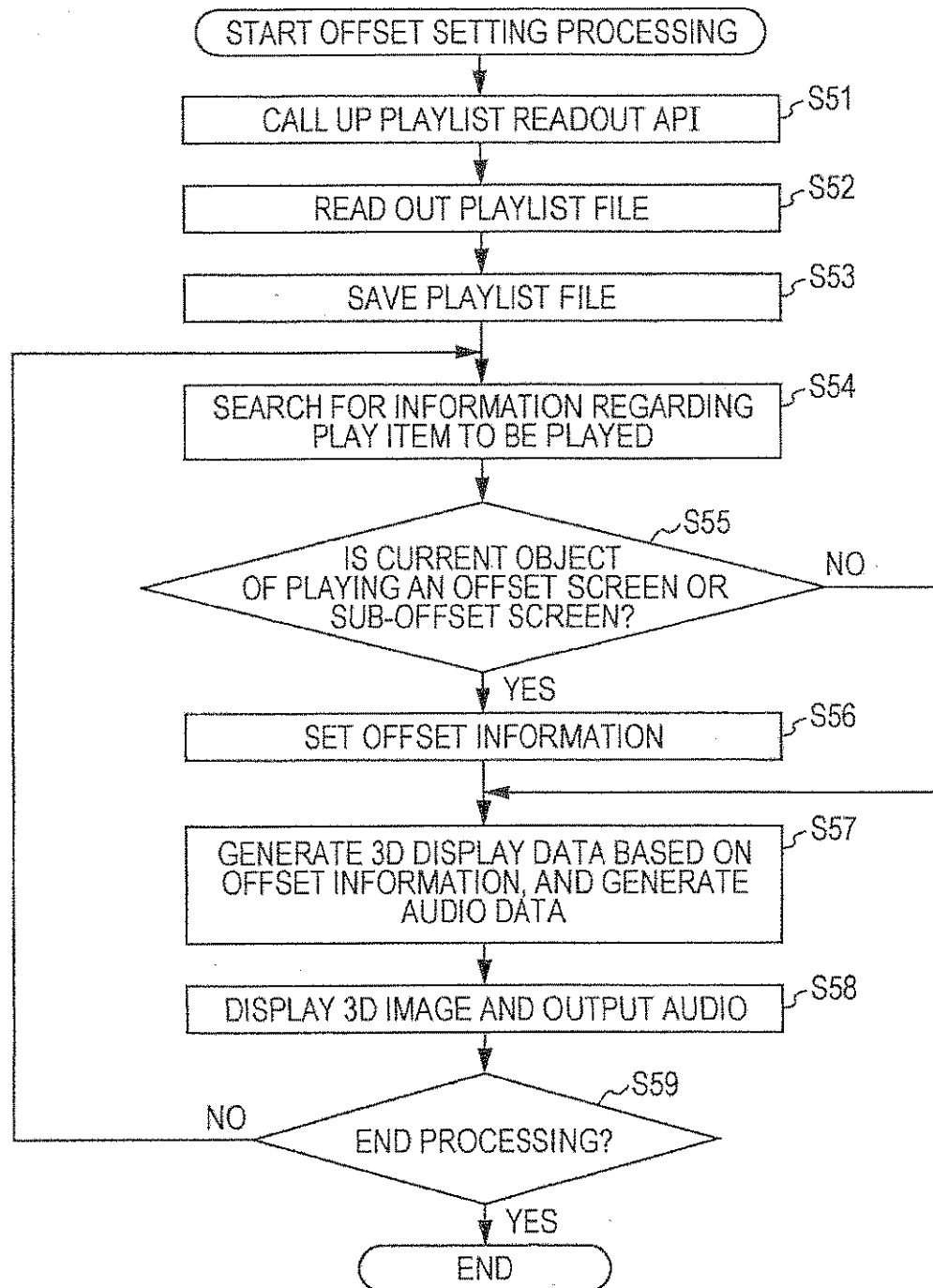


FIG. 14

Syntax	No. of bits	Mnemonic
offset_metadata0 {		
length	32	uimsbf
number_of_offsets_id	4	uimsbf
reserved_for_future_use	12	bslbf
for (offset_id=1; offset_id <= number_of_offsets_id; i++) {		
for (i=0; i < number_of_playlistitem; i++) {		
number_of_offsets[i]	16	uimsbf
for (j=0; j < number_of_offsets[i]; j++) {		
offset_pts[j]	32	uimsbf
offset_direction[j]	1	bslbf
offset_value[j]	6	uimsbf
sub_offset_rate[j]	5	uimsbf
number_of_sub_offsets[j]	5	uimsbf
for (k=0; k < number_of_sub_offsets[j]; k++) {		
sub_offset_direction[k]	1	bslbf
sub_offset_value[k]	6	uimsbf
}		
}		
}		
}		
padding_word	n1	bslbf
}		

FIG. 15

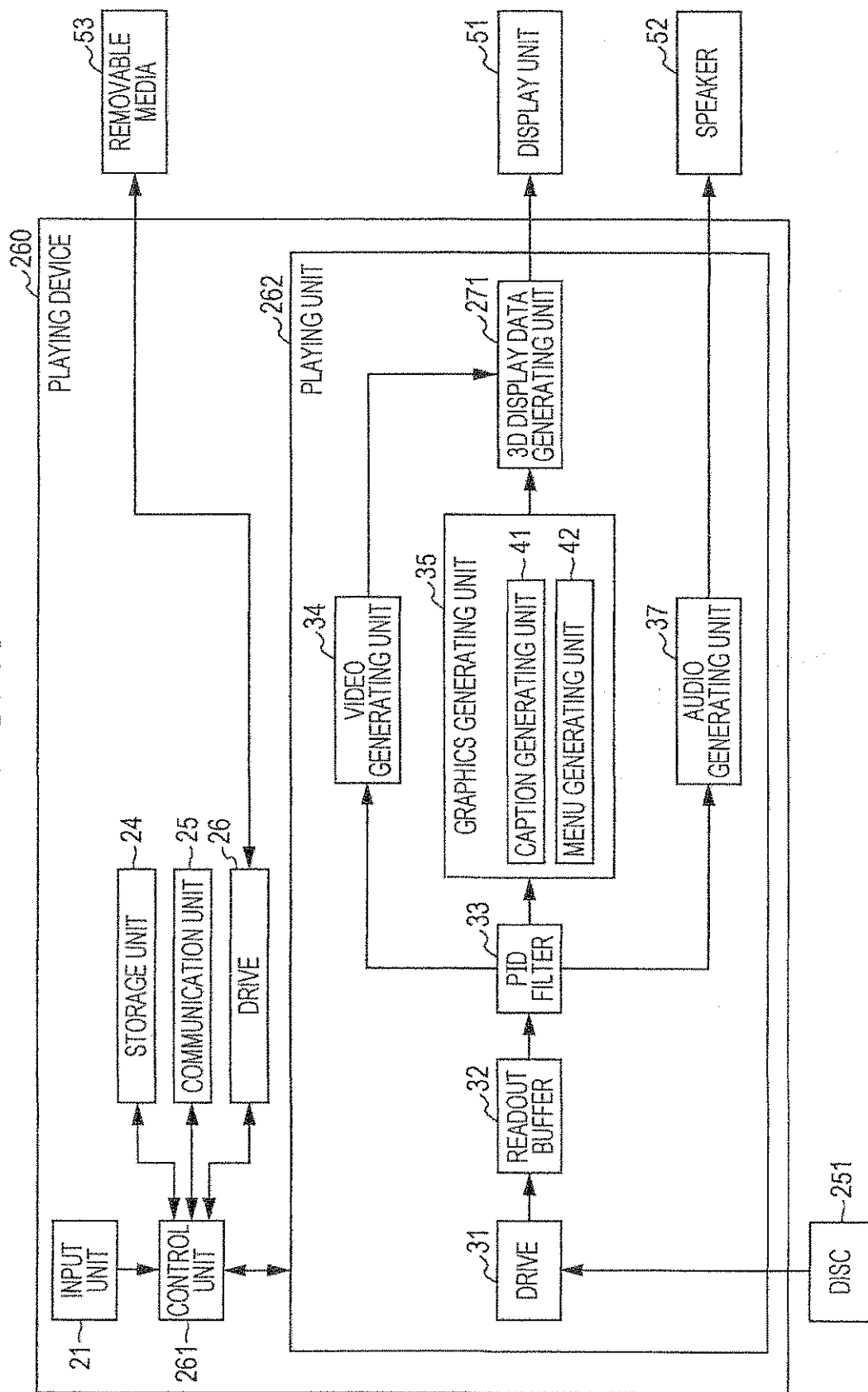


FIG. 16

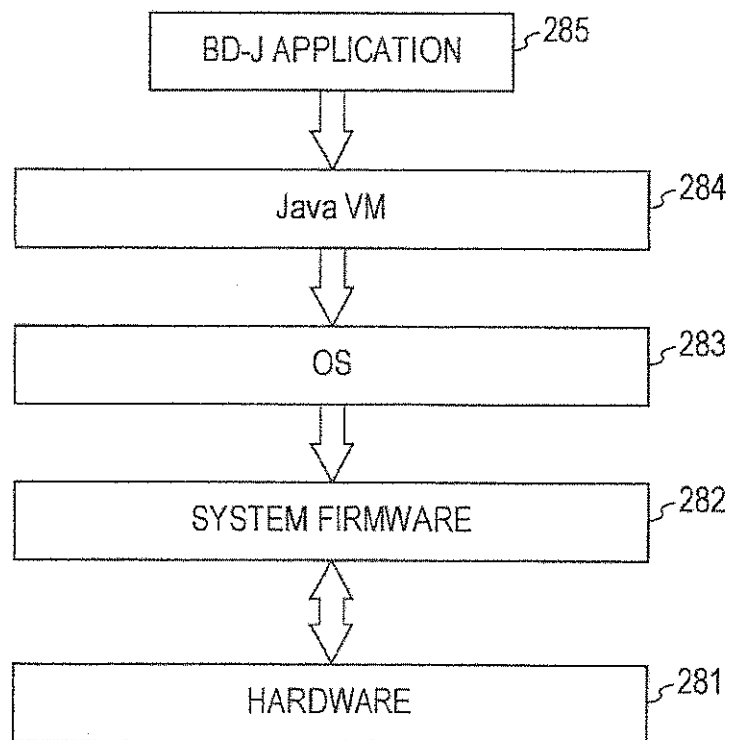


FIG. 17

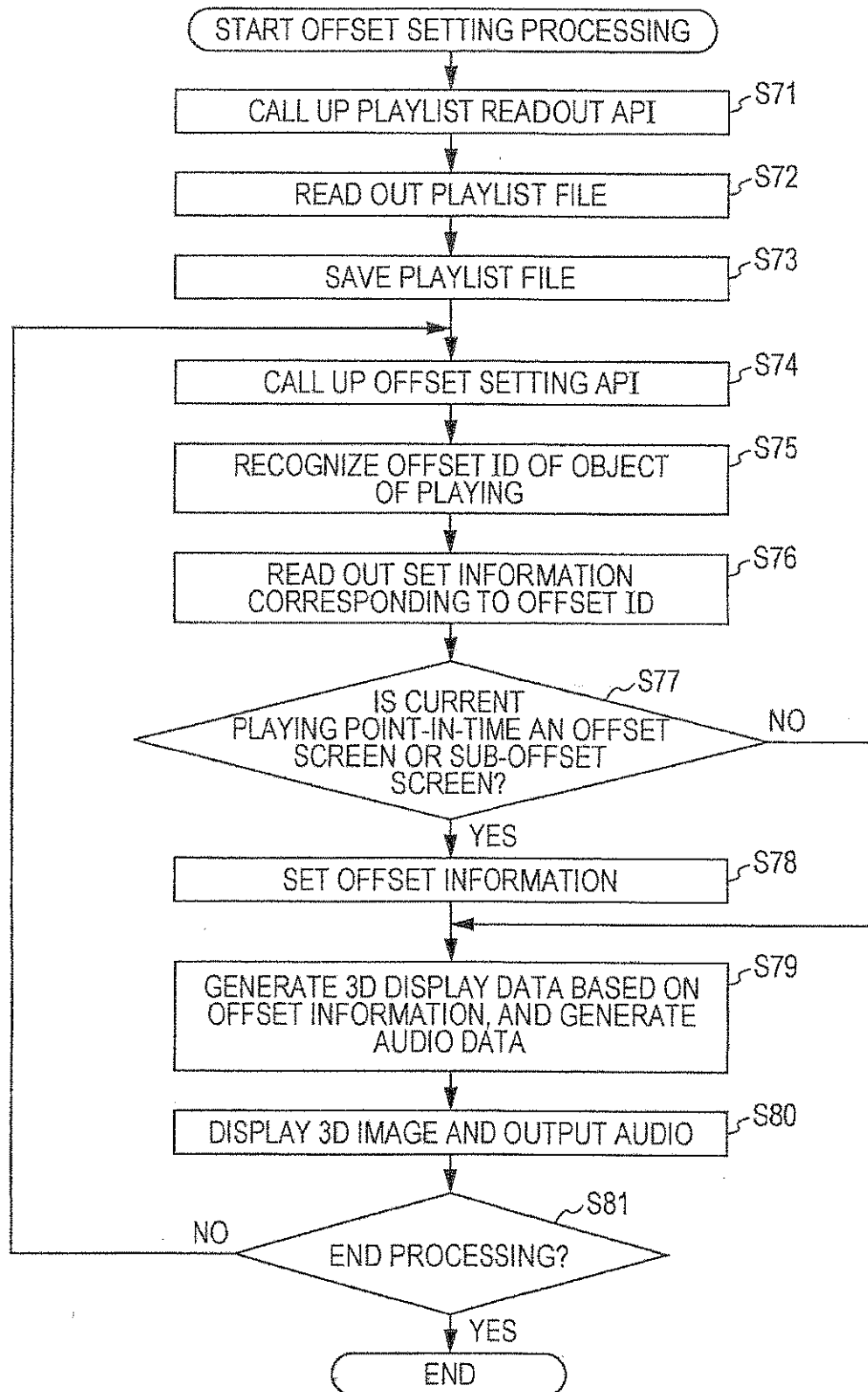


FIG. 18

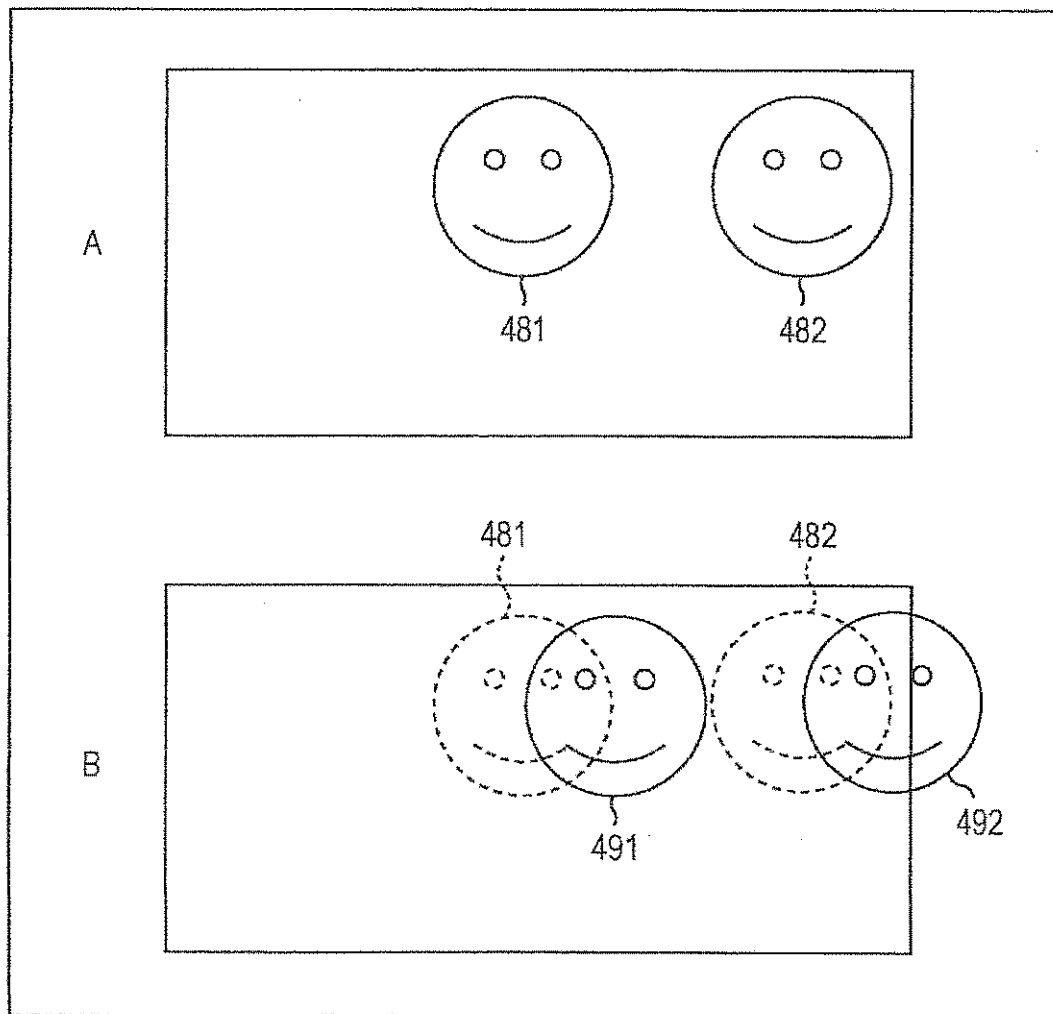


FIG. 19

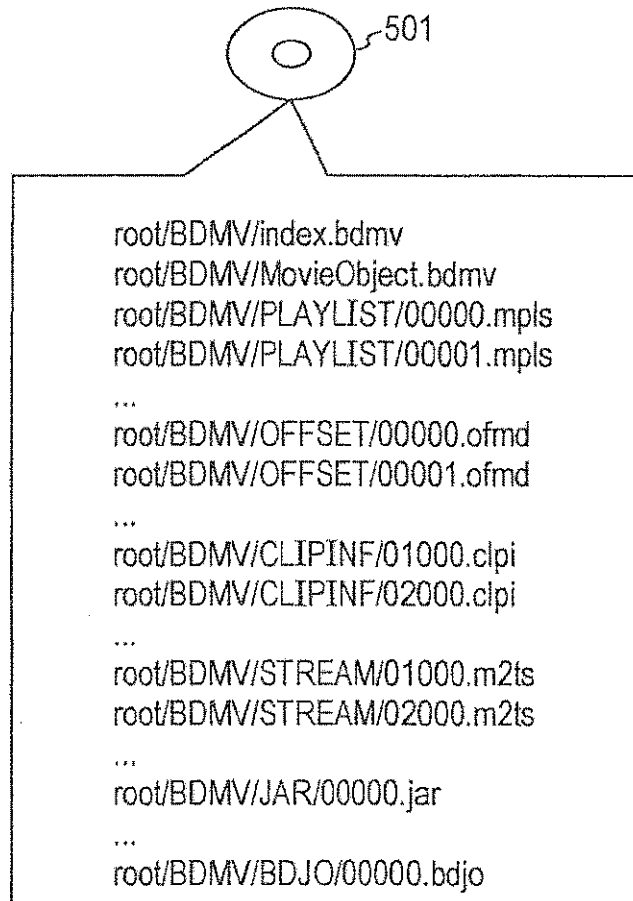


FIG. 20

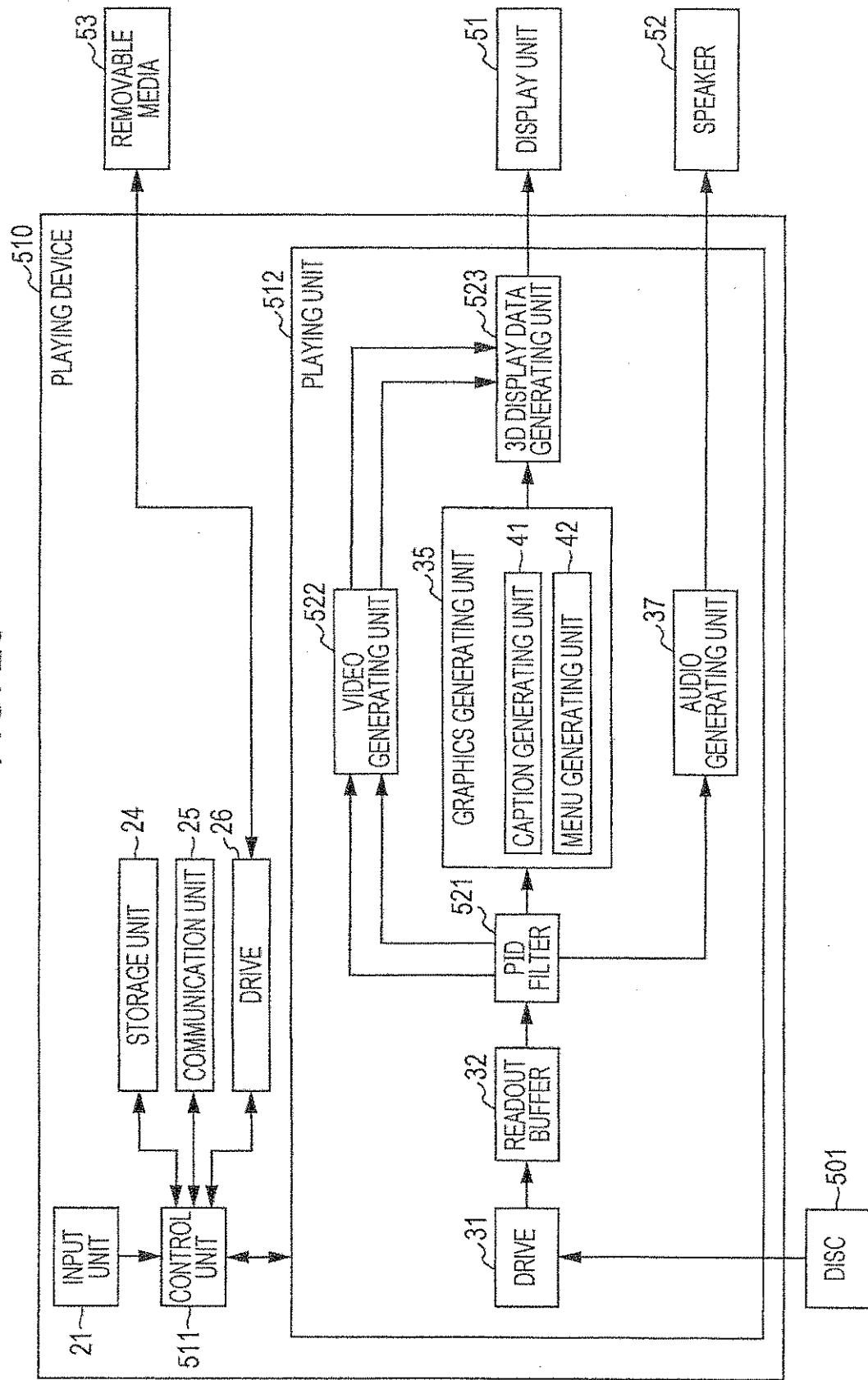


FIG. 21

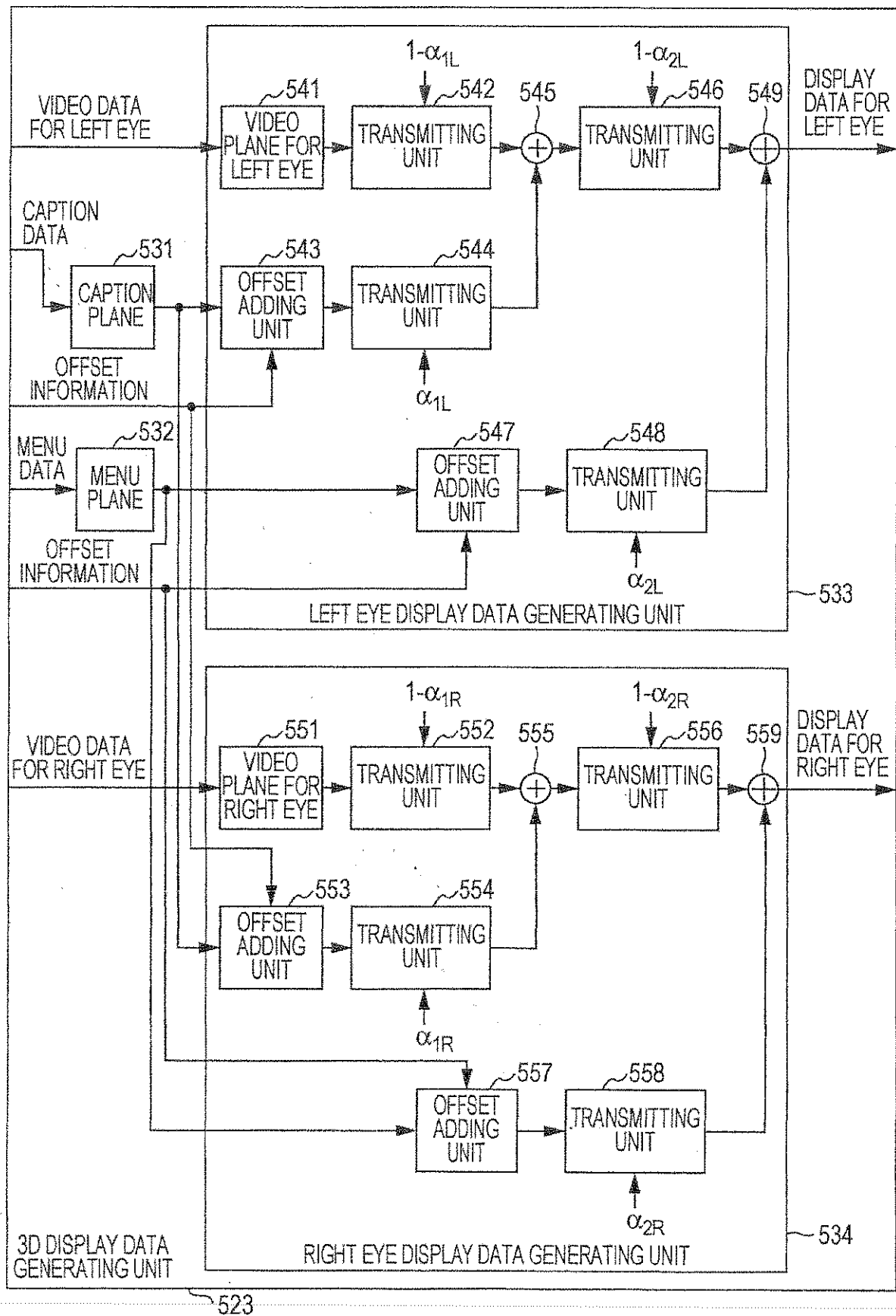


FIG. 22

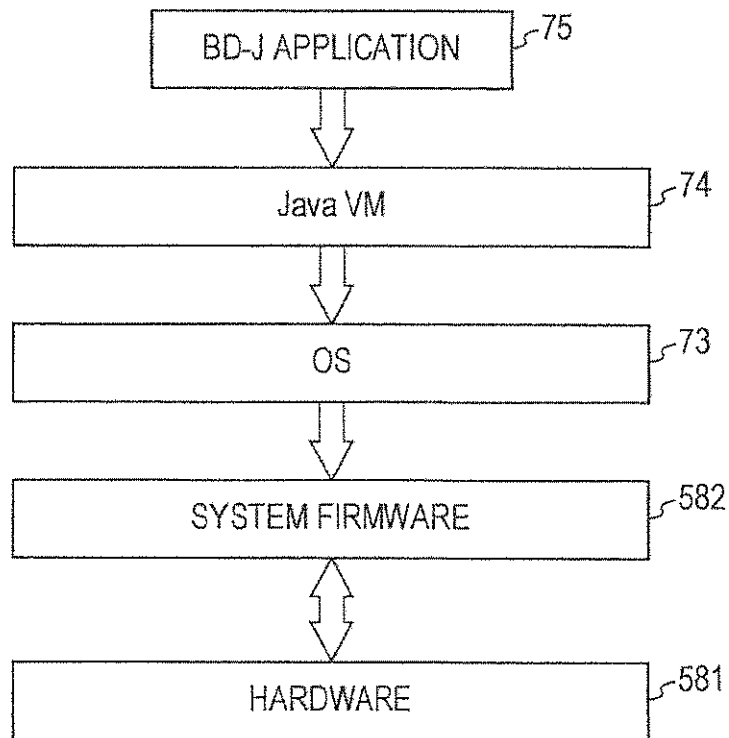


FIG. 23

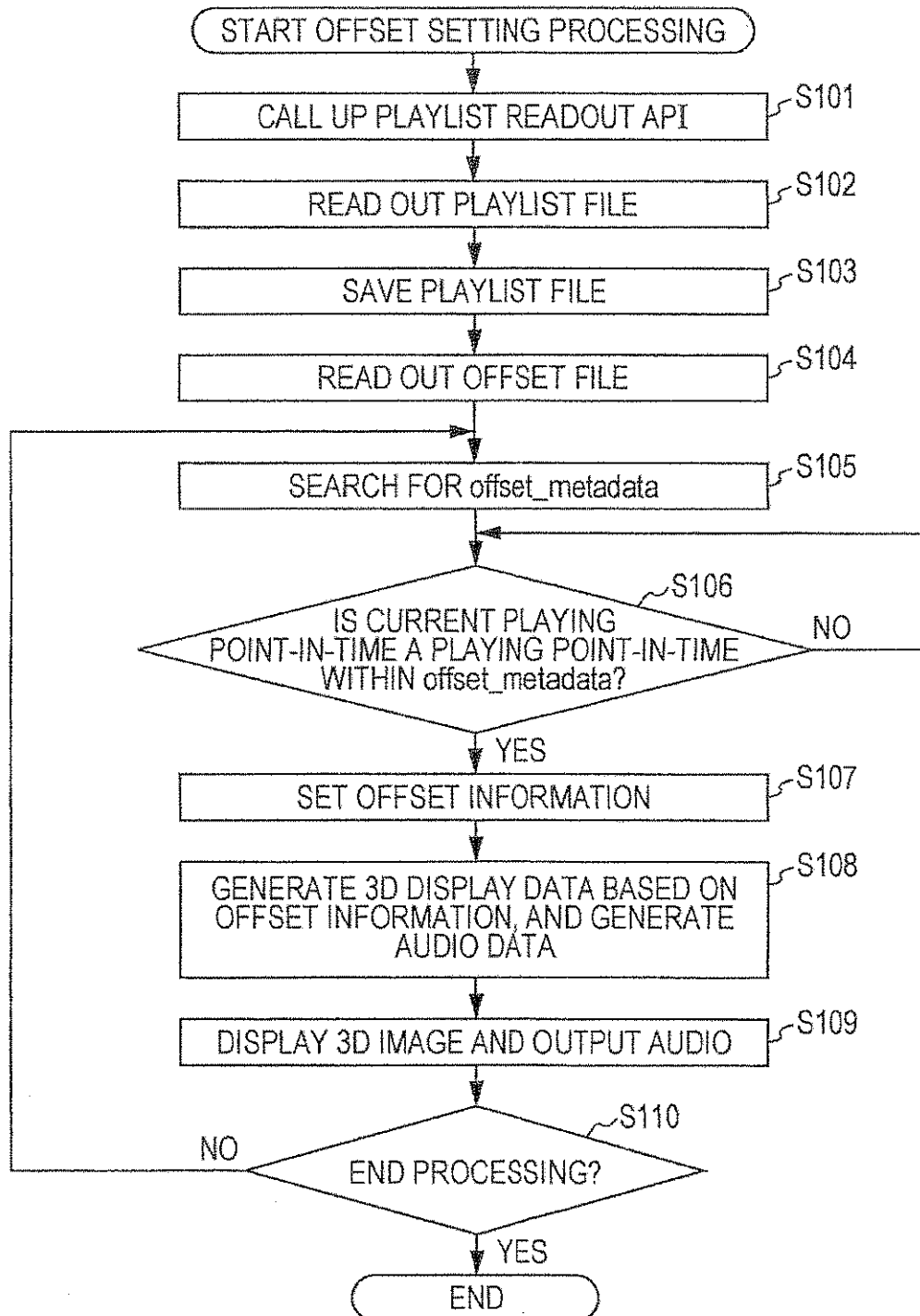


FIG. 24

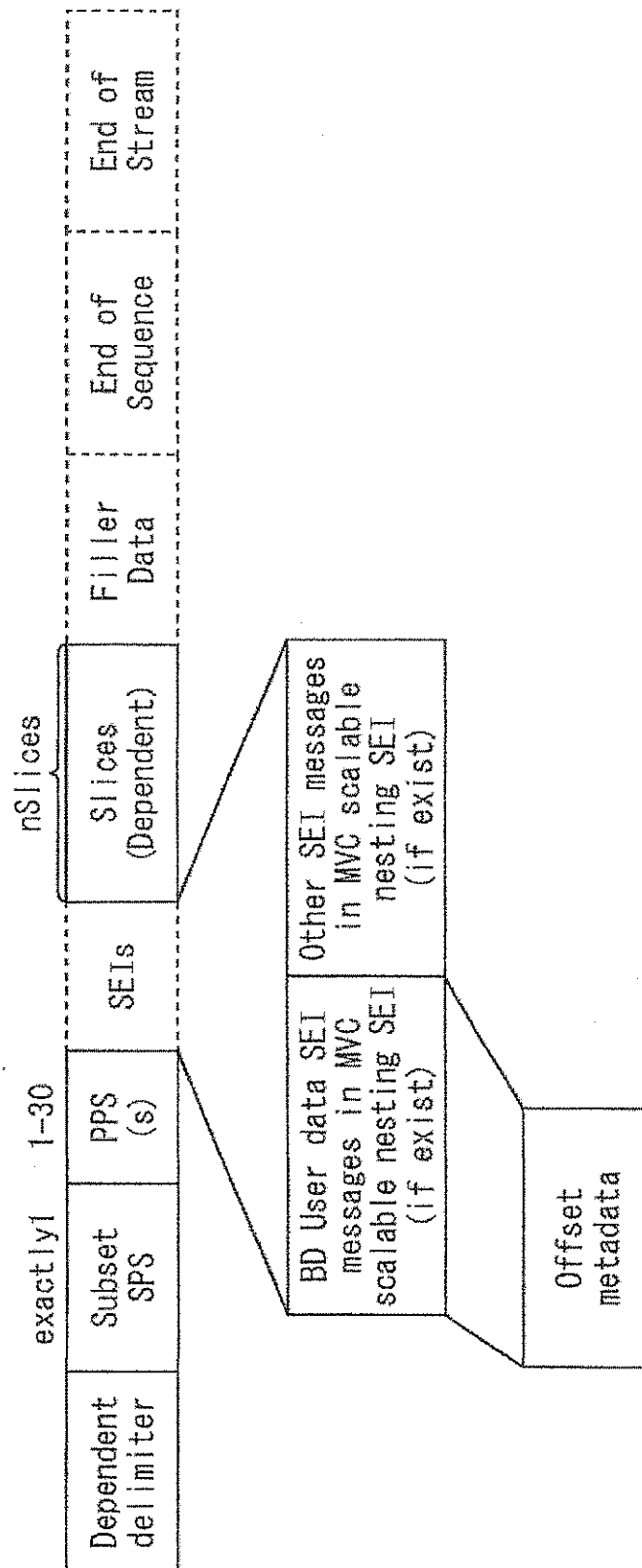


FIG. 25

Syntax offset_metadata() { frame_rate offset_start_PTS number_of_frames number_of_PG_offset_sequences number_of_IG_offset_sequences for (int i=0; i < number_of_PG_offset_sequences; i++) { PG_offset_sequence() } for (int i=0; i < number_of_IG_offset_sequences; i++) { IG_offset_sequence() } }	Syntax PG_Offset_sequence() { for (i=0; i < number_of_frames; i++) { offset_direction_flag offset_value } }
A	B
	Syntax IG_Offset_sequence() { for (i=0; i < number_of_frames; i++) { offset_direction_flag offset_value } }
	C

FIG. 26

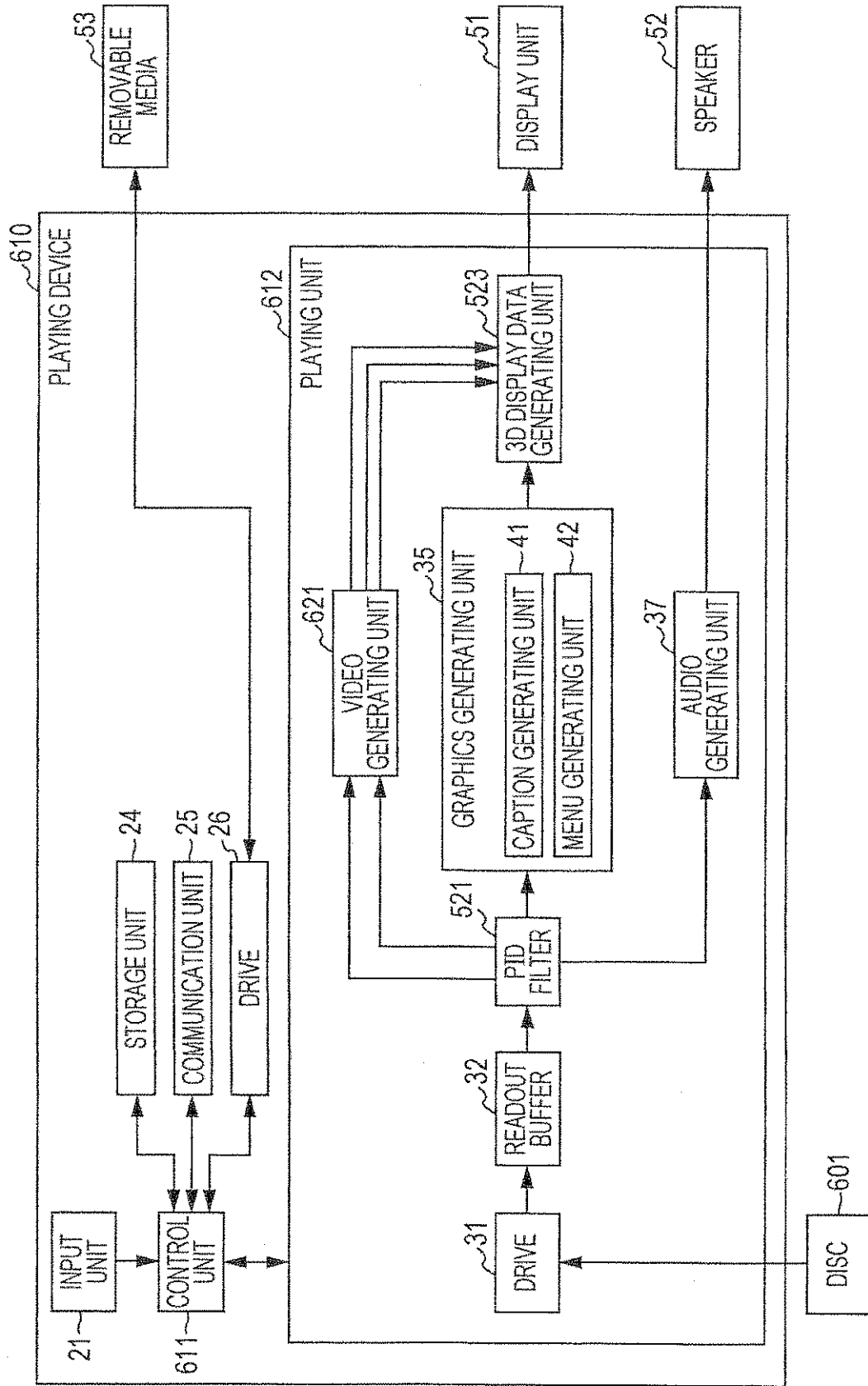


FIG. 27

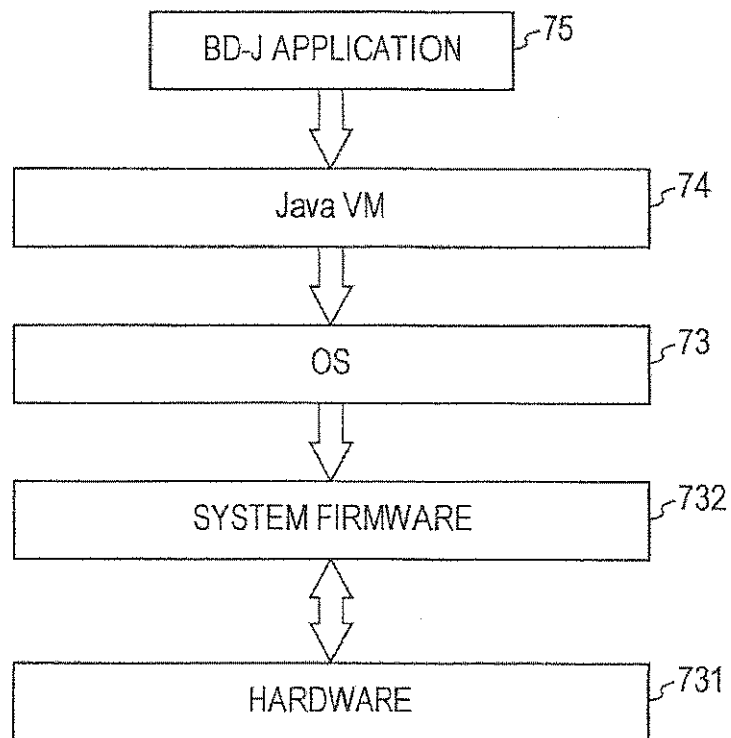


FIG. 28

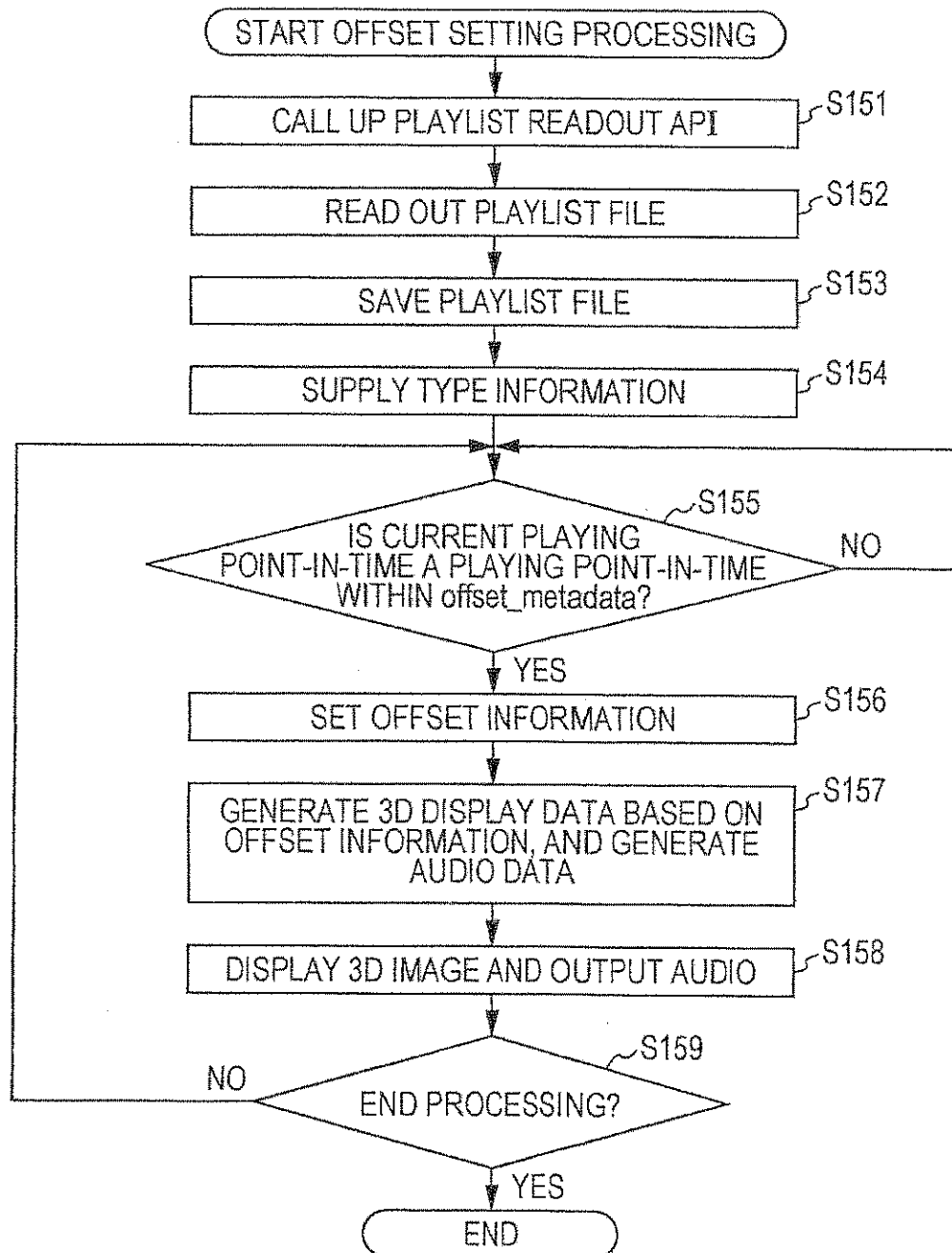


FIG. 29

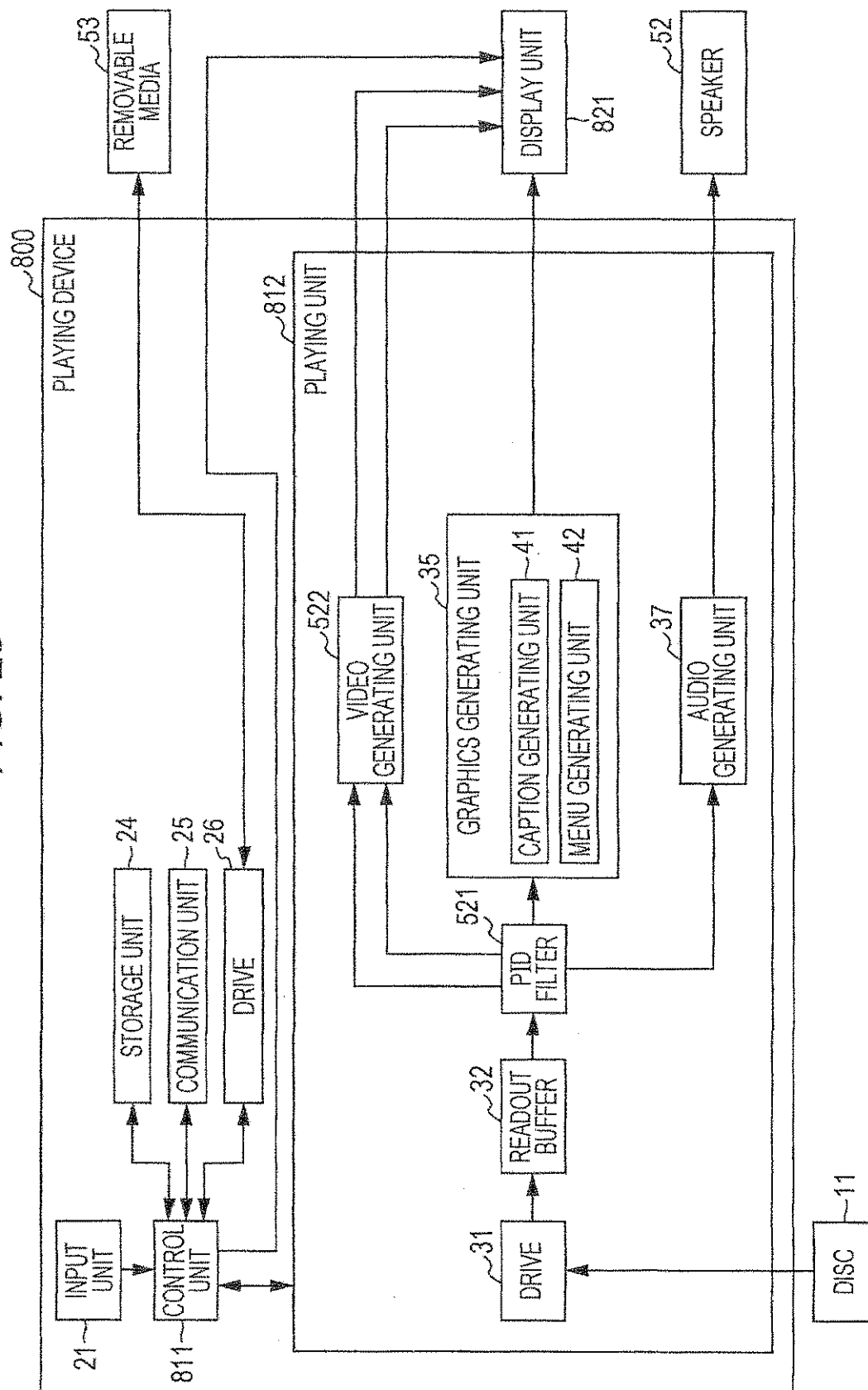


FIG. 30

