OPIC
OFFICE DE LA PROPRIÉTÉ INTELLECTUELLE DU CANADA

CIPO
CANADIAN INTELLECTUAL PROPERTY OFFICE

(12) (19) (CA) **Brevet-Patent**

(11) (21) (C) **2,195,942**

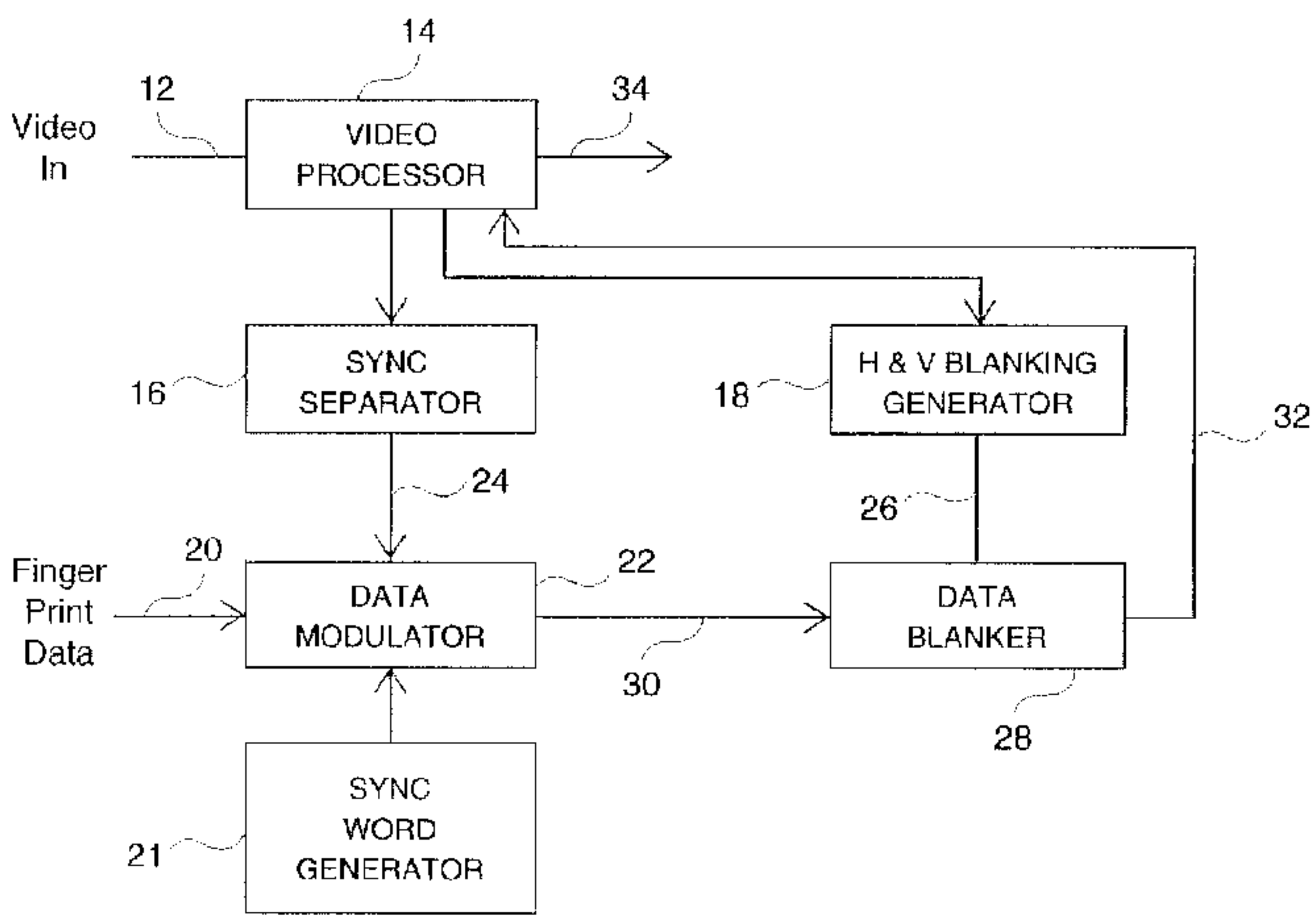
(86) 1995/08/22
(87) 1996/02/29
(45) 2001/04/10

(72) Copeland, Gregory C., US
(73) MACROVISION CORPORATION, US

(51) Int.Cl.^{6} H04N 7/08
(30) 1994/08/24 (294,983) US

(54) PROCEDE ET DISPOSITIF D'INSERTION DE DONNEES D'IDENTIFICATION ORIGINE DANS UN SIGNAL VIDEO

(54) METHOD AND APPARATUS FOR INSERTING SOURCE IDENTIFICATION DATA INTO A VIDEO SIGNAL



(57) Les données d'identification origine (empreintes digitales) sont injectées dans la zone d'image active d'un signal vidéo sans interférer avec la visualisation obtenue au moyen du signal vidéo. Les données sont récupérées ensuite par un lecteur de données ou "Lecteur d'Empreintes Digitales". Le procédé d'injection des données ou de "reproduction d'empreintes digitales" consiste à décaler dynamiquement le palier de suppression vidéo de façon à véhiculer l'information qui peut être restituée en lecture depuis n'importe quelle bande vidéo réalisée à partir de la sortie de l'unité d'injection de données. En l'occurrence, l'empreinte digitale comporte le numéro d'identification de l'unité utilisée et la date du jour. Le décalage dure une zone entière et présente une amplitude d'environ 0,5 unités I.R.E., c'est à dire qu'une zone spécifique présentera des caractéristiques constitutives nominales ou des valeurs de caractéristiques constitutives s'écartant de 0,5 unités I.R.E. des valeurs nominales. Les données se répètent toutes les 128 zones de façon à fournir au lecteur un vaste échantillonnage et à afficher les données d'identification origine.

(57) The source identification data (Finger Print) is injected into the active picture area of a video signal without disturbing the viewing of the video signal and the data is retrieved by a data reader, called a Fingerprint Reader. The data injection or "fingerprinting" process consists of dynamically offsetting the video pedestal to carry information which can then be read back from any videotape made from the output of the data-injecting unit. In particular, the fingerprint carries the ID number of the unit used and the current date. The offset lasts for one entire field and has an amplitude of approxmately 0.5 IRE that is, a given field either has the nominal setup or a setup value differing from nominal by 0.5 IRE. The data is repeated every 128 fields in order to provide ample samples for the reader to detect and display the source identification data.

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04N 7/08

A1

(11) International Publication Number: WO 96/06503

(43) International Publication Date: 29 February 1996 (29.02.96)

2195942

(21) International Application Number: PCT/US95/10665

(22) International Filing Date: 22 August 1995 (22.08.95)

(30) Priority Data:
08/294,983 24 August 1994 (24.08.94) US

(71) Applicant: MACROVISION CORPORATION [US/US]; 1341 Orleans Drive, Sunnyvale, CA 94089 (US).

(72) Inventor: COPELAND, Gregory, C.; 1479 Lapaz Court, San Jose, CA 94089 (US).

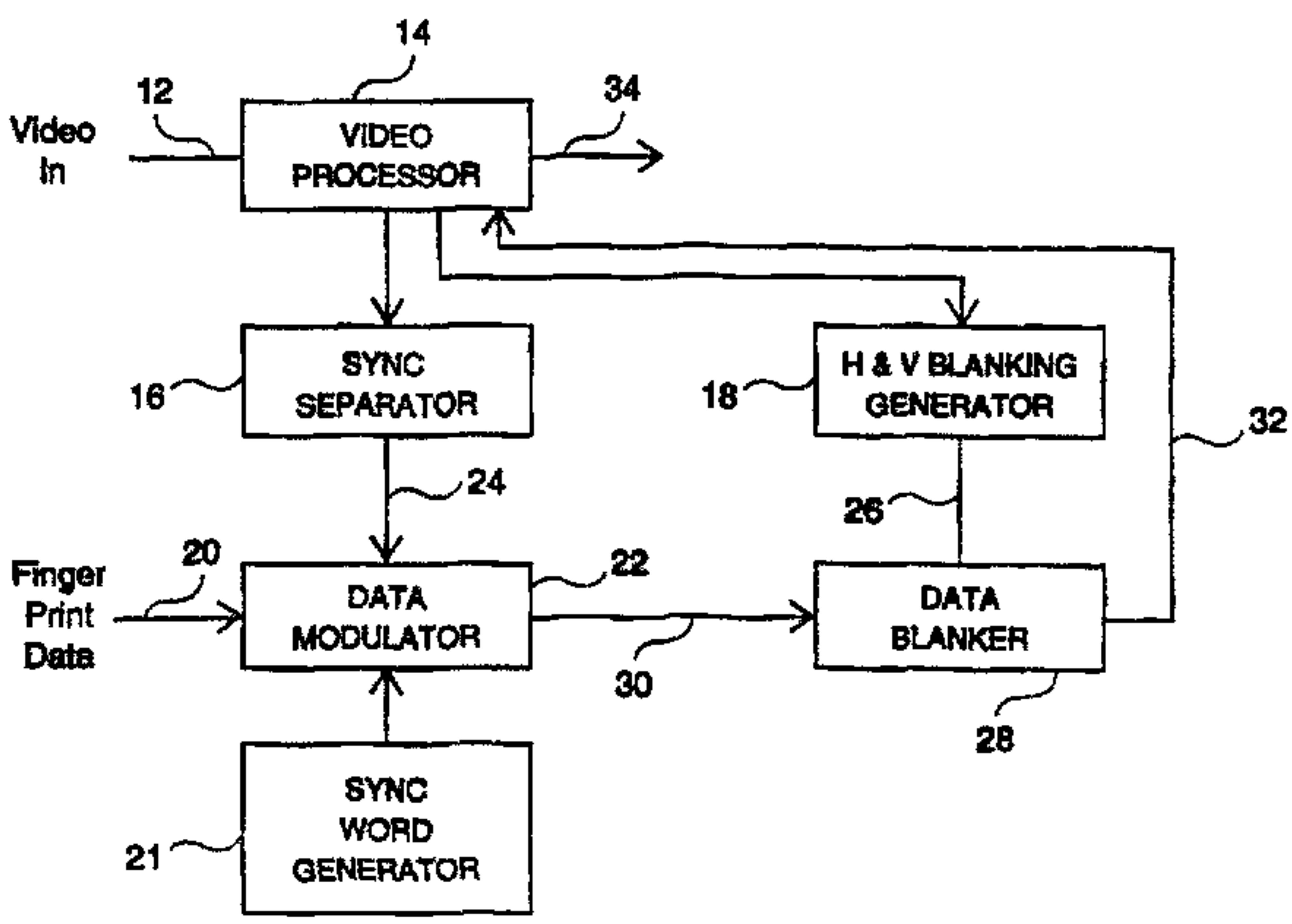
(74) Agent: BRILL, Gerow, D.; Macrovision Corporation, 1341 Orleans Drive, Sunnyvale, CA 94089 (US).

(81) Designated States: AM, AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TT, UA, UG, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, MW, SD, SZ, UG).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

2195942

(54) Title: METHOD AND APPARATUS FOR INSERTING SOURCE IDENTIFICATION DATA INTO A VIDEO SIGNAL



(57) Abstract

The source identification data (Finger Print) is injected into the active picture area of a video signal without disturbing the viewing of the video signal and the data is retrieved by a data reader, called a Fingerprint Reader. The data injection or "fingerprinting" process consists of dynamically offsetting the video pedestal to carry information which can then be read back from any videotape made from the output of the data-injecting unit. In particular, the fingerprint carries the ID number of the unit used and the current date. The offset lasts for one entire field and has an amplitude of approxmately 0.5 IRE that is, a given field either has the nominal setup or a setup value differing from nominal by 0.5 IRE. The data is repeated every 128 fields in order to provide ample samples for the reader to detect and display the source identification data.

METHOD AND APPARATUS FOR INSERTING SOURCE IDENTIFICATION DATA INTO A VIDEO SIGNAL

BACKGROUND

With video piracy becoming more rampant by the day, it is becoming more desirable to have a method for identifying whether a video recording or video transmission is originating from an authorized source. Source or tape identification processes using the data transmission capability of the vertical interval have been known to those skilled in the art. However such system suffer from the ease of eliminating the source identification data by blanking and reinsertion techniques. The source identification or "fingerprint" systems known do not transmit the data during the active time of the video signal.

One form of video piracy has been to use a video camera to record the picture and sound off the screen and speakers in a theater displaying a movie. Admittedly this method produces a very inferior copy. However, in certain parts of the world, generally outside of the United States, such a copy is acceptable. The use of video movie projection systems in theaters is becoming more popular. Generally, these systems incorporate a form of video scrambling to protect the electronic video signals prior to projection. However, vertical interval source identification and video scrambling do not protect the projected image once the signal has been descrambled.

A method of source identification of movie and other material is needed to provide a source code to reduce the likelihood of illegal copying and if such copying is done, identify the theater or source of the duplication.

SUMMARY

In accordance with one aspect of the present invention there is provided a method for incorporating source identification data into the active picture of a video signal formed of successive video fields while maintaining the source identification data imperceptible to a viewer, the method comprising the steps of: generating source identification data, separating vertical sync pulses at vertical rate from the video signal, synchronizing the source identification data to the video signal at a vertical rate in response to the vertical sync pulses, blanking the synchronized source identification data during horizontal and vertical blanking intervals of the video signal, and adding the synchronized source identification data to the video signal.

In accordance with another aspect of the present invention there is provided apparatus for incorporating source identification data into the active picture of a video signal formed of successive video fields while maintaining the source identification data imperceptible to a viewer, said apparatus comprising: means for generating source identification data, processing means for adding said synchronized source identification data to the video signal, means for providing sync signals synchronized to the video signal, said means for providing having a vertical sync separator for separating vertical sync pulses at vertical rate from the video signal, means for synchronizing the source identification data to the video signal by synchronizing the source identification data to the vertical rate and to the video signal in response to the vertical sync pulses, and means for blanking said synchronized source identification data during the horizontal and vertical blanking intervals of the video signal and before the synchronized source identification data is added to said video signal.

In accordance with yet another aspect of the present invention there is provided an apparatus for incorporating source identification data not visible or disturbing to a viewer into an active picture area of a video signal formed of successive video fields, comprising: means for receiving and processing the video signal, means coupled to the processing means for generating a vertical sync pulse from the video signal, means responsive to the vertical sync pulse for providing a sync word and source identification information synchronized to vertical rate, means coupled to the providing means for blanking said synchronized sync word and source identification information during video blanking intervals, and wherein said receiving and processing means is coupled to said blanking means for dynamically offsetting the video pedestal of the video signal for at least an entire video field to therein supply said sync word and source identification information at a very low signal level to said active picture area of the video signal.

In accordance with still yet another aspect of the present invention there is provided a method for incorporating source identification data into the active picture of a video signal, the method comprising the steps of: generating source data including a synchronizing word for incorporation into the active picture of a video signal, wherein said source data is source identification data, synchronizing said source data to the video signal, and adding said synchronized source data to said video signal, wherein, to render the source data imperceptible to a viewer, said synchronized source data is blanked during the horizontal and vertical blanking intervals of said video signal, a video pedestal of the video signal to carry the source identification data is dynamically offset for at least one entire field, and wherein said synchronized source identification data is blanked during the horizontal and vertical blanking intervals before said synchronized source identification data is added to said video signal.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of the data insertion apparatus; and

Fig. 2a - 2e is a series of wave forms representing sync word; and

Fig. 3 is a block diagram of the Fingerprint Reader.

DETAILED DESCRIPTION OF THE INVENTION

The data injection or "fingerprinting" process consists of dynamically offsetting the video pedestal to carry information which can then be read back from any videotape made from the output of the data-injecting unit or from a videotape of a screen display of a signal containing the data. In particular, the fingerprint carries the ID number of the unit used and the current date. The offset lasts for at least one entire field and has an amplitude of approximately 0.5 IRE unit. A given field either has the nominal setup or a setup value differing from nominal by 0.5 IRE.

Video Signal 12 to be fingerprinted is inputted to Video Processor 14. Video Processor 14 provides Video Signal 12 to Sync Separator 16 and H & V Blanker 18. Fingerprint Data 20 is inputted to Data Modulator 22. Fingerprint Data 20 may be in serial or parallel form.

The fingerprint data format is a 64-bit block. The first 16 bits are a (data) frame synchronization word generated by Sync Word Generator 21; the next 16 bits are the source ID number; and the final 32 bits are the date code. A typical sync word is shown in Fig. 2a. The block is repeated indefinitely. The signal format is binary Manchester: a "0" is represented by a 0-1 transition, a "1" by a 1-0 transition, with the phase reference supplied by the sync word as shown in Fig. 2b. Each data bit therefore occupies two fields. It will be apparent to one skilled in the art that other data formats can be used, with the recovery process adjusted accordingly.

Data Modulator 22 receives the Fingerprint Data 20 in either serial or parallel form and Sync Word Data Generator 21 and generates Formatted Fingerprint Data as described above consisting of 16 bits as synchronization word followed by 16 bits for a source ID and 32 bits for a date code. Since the data is to be inputted to a video signal, it is necessary to synchronize the data to the video signal. An output of Sync Separator 16 consists of Vertical (field) Synchronizing Pulse 24. Sync Separator 16 uses techniques known to one skilled in the art of television engineering to separate Vertical Synchronizing Pulse 24 from the video signal. In addition to putting the data in a format usable by the system, Data Modulator 22 synchronizes the beginning of any Formatted Fingerprint Data to the field rate using Vertical Synchronizing Pulse 24.

Since the individual bits of data 30 output from Data Modulator 22 one or more field, it is necessary to blank out the data during the horizontal and vertical blanking periods of the inputted video signal. H & V Blanking Generator 18 receives the video signal from Video Processor 14 and generates a combined H & V Blanking Signal 26 that is coupled to Data Blanker 28. Formatted Fingerprint Data 30 is coupled to a data input of Data Blanker 28. Data Blanker 28 uses H & V Blanking Signal 26 to blank out the data during the horizontal and vertical blanking intervals of the input video signal. The output of Data Blanker 28 is Blanked Formatted Fingerprint Data 32 which is coupled back into Video Processor for adding to the video signal thus producing Fingerprinted Video Signal 34.

When Fingerprinted Video Signal 34 is projected on a screen or displayed on a video monitor, the variation in video level due to the insertion is imperceptible to a viewer, but is detected by any video recorder recording the signal directly or via a television camera doing an off-screen recording of the projected image.

The recovery or "reading" process operation is shown in Fig. 3. A Fingerprint Reader 40 is used to detect and analyze any Formatted Fingerprint Data 30 (Fig. 1) present in a Fingerprinted Video Signal 42 (similar to the signal 34 of Fig. 1) is coupled to Low Pass Filter 44 with a cut-off frequency of about 1 Khz. The output of Low Pass Filter 44 is coupled via an Analog to Digital Converter 46 to 30 Hz Notch Filter 82. 30 Hz Notch Filter 82 is used to remove a 30 Hz component that can be observed in the data as a frame to frame variation in tape output level due to differences in playback video heads. The 30 Hz Notch Filter is coupled to a Matched Filter 52 but can be placed either before or after Matched Filter 52.

Analog to Digital Converter 46 also receives Clock Signal 48 from Clock Generator 50. The clock frequency is approximately 4 Khz. However clock frequencies from 1 Khz to 15 Khz are equally usable. The clock frequency also may be locked to the incoming video. The output of 30 Hz Notch Filter 82 is coupled to Matched Filter 52, which to a first approximation, doubles the data amplitude and largely cancels the video content. The output of Matched Filter 52 is represented by Fig. 2c.

The output of Matched Filter 52 may not always provide a sufficiently clean signal for further processing due to time base errors in the playback signal. To improve the usability of the output signal of Matched Filter 52 is coupled to Squarer 84 to Phase Lock Loop 86 to generate a clock signal at the bit of the data. This clock signal 88 is coupled to Storage Register 56 to eliminate variations in the data due to time base errors.

The output of Matched Filter 52 that contains data with canceled video is then correlated to the sync word to establish the data framing. Once the data is properly framed, it is digitally integrated to further improve the Signal to Noise Ratio (SNR). This process consists of writing the digitized values of 128 consecutive fields into Storage Register 56 that contains 128 individual registers that have been clocked by Clock Signal 88. Shift Register 56, Correlator 58, Sync Word Generator 60, Peak Detector 62 and Address Control 64 are in a loop that is used to synchronize the data to make it readable by the user. The output of Storage Register 56 is coupled to Correlator 58. In addition a preprogrammed Sync Word Generator 60 couples a unique Sync Word 66, as shown in Fig. 2d, to Correlator 58 to correlate the sync word information in the Fingerprint Data 32 with Sync Word 66. Correlation Data 68, as shown in Fig. 2e is coupled to Peak Detector 62. If there is a match between Sync Word 66 and the sync word information in the Fingerprint Data 32, the digitized values of the next 128 fields are added to the first field, and the process is continued as required. As the accumulation proceeds, the data value in each register will be directly multiplied by the number of passes while the video and noise will tend to average out. After a suitable number of passes, generally in excess of twenty, the recovered sequence of 128 high or low offset fields is coupled to Adder 72 and 64 Word Register 76 where the data is decoded to the original 64 bits, and the pertinent data is read out to Output Terminal 80.

The apparatus described above may use a hardware implementation or a combination of hardware and software. Attached is the source code information using the Matlab language for a software implementation of part of the fingerprint reader.

The data that is read out can be connected to any display device capable of reading a 64 bit data stream. Such a device could be an alphanumeric display, a computer screen or incorporated back into the video signal for an on screen display. The display device displays the ID number and date code.

During an experiment using the elements described above, data has been found to be recoverable down to the vicinity of 1/2 millivolt on a 1 volt video signal, less than 0.1 IRE.

It is important to note, that unless the sync word in the insertion device and the reader are identical, no output will result or misleading results will be obtained. In order to make the reading device have a more universal use, the sync word generator in the reader has an ability to be preprogrammed either by the manufacturer for the user or by the user.

In principle, while using the system described above, a person attempting to defeat the system could simply blank or otherwise distort one or two fields out of every 128 fields, a distortion which might be brief enough not to significantly affect the utility of the pirated signal. Since the data is periodic, this would distort one of the characters. Which character in particular would not be known, but with the present data format, there is be a 25% probability of hitting one of the ID number characters, thereby disguising the identity of the pirated unit. Alternatively a pirate may add a signal that will swamp or override the original fingerprint so that the fingerprint reader produces inconsistent or incorrect results.

An alternative implementation has some advantages. In a technique similar to that used for spread-spectrum transmission, the basic data rate may be slowed, and Exclusive-Ors with a known pseudo-random sequence at the original frame rate. Normally the pseudo random sequence will be much longer than the data sequence. This has the advantage of keeping the setup variation rate high, to minimize visibility of the data, while insuring that any attempt to mask a data character would blank an unacceptably long portion of the signal. A second alternative would simply randomly interchange the position of data bits within the block in a time-variant manner, such that no periodic distortion would suffice to distort any particular data character. Data recovery for this second method is clearly more complex, but well within the state of the art.

A third alternative may be to encrypt the data prior to it being added to the signal and using complementary decryption techniques for detection.

The apparatus and method described above describes a system and method of adding a fingerprint signal to a video signal. One of the principle uses of this method and apparatus is to prevent piracy or identify the source of the pirated video material. Another embodiment of the concept is to fingerprint an original film that is ultimately recorded into a video format. This can be accomplished by one of two methods. The film as it is being duplicated in film has a bias light source with a short turn on/off time to provide a small increase of illumination in the film printer. This bias light would be sufficiently low to create a very small shift in the brightness, but not be visible to a viewer of the film. A second method is to provide such a bias during the projection of the film. The rate of the bias light takes into account the various projection media, direct film projection at 24 frames per second, 25 frames per second when used in a 50 Hz television system or the 3/2 mechanism when used in a 60 Hz television system.

The programming code for the fingerprint detection system is outlined following this Detailed Description of the Invention.

The above description is illustrative and not limiting. Further modifications will be apparent to one of ordinary skill in the art in light of this disclosure.

```
function [zc,y,loc] = finger2(file_name,sim)
% This function computes the fingerprint values given the input
% filename. It also plots intermediate results for debugging and
% visualation. This function assumes that the input is in voc
% type 1 format digitized at 4 KHz. The data is assumed to be
% frame of 64 bits manchester encoded with a sync word of 16 bits
% followed by 16 bits of decoder id, 3 bytes of date data (in bcd)
% and finally 1 byte of spare presently set to 0's. Data is
% assumed to non-inverted, although it is believed that the decoder
% now ouputs inverted fingerprint. The sound blaster used did not
% invert the data on input, but the sound blaster inverted the data
% on output. The data was collected using the program named
% fingrprt.exe and executed from the DOS prompt.
%       Greg Copeland 8/16/94

%
% Get the data and decimate by 8 using a 256 tap
% low pass filter. Each bit before decimation is approximately
% 133 samples long (4000 Hz sample rate / 30 Hz bit rate). After
% decimation each symbol period then becomes
% approximately 16.7 (4000Hz/30Hz/8) samples long.
%

if sim==1                          % 1 for data, 0 simulation
 fid = fopen(file_name);% open the input file
 x = fread(fid,100,'uint8');       % eliminate the header info
 i=0; x0 = zeros(8,1);   % clear filter state
 x1 = x0; x2 = x0; x3 = x0;        % create lpf for decimation
 lpf = firls(123,[0 .08 .125 1],[1.0 1.0 0.0 0.0]);
 m = [-ones(1,66) 0 ones(1,66) ];% and conv with matched filt
 lpf = conv(lpf,m);                % for combined filtering
 plot(20*log10(abs(fft([lpf zeros(1,8*size(lpf))])+.00001)));
 pause(1);
 dec = 8;
 flag = 1;
 x0 = zeros(256,1);
 x = zeros(dec,1);
 while flag                        % loop for each 8 sample blk
  x0(1:256-dec)=x0(dec+1:256);% until no more data avail
  [x,count] = fread(fid,dec,'uint8');      % read input file
  flag = count == dec;  % full block to use ?
  if(flag)
   x0(256-dec+1:256)=x(1:dec);% shift state info
   i = i+1;
   z(i) = x0' * lpf';        % find decimation sample
  end
 end
 y = z;

 clear lpf                         % free memory for later use
 clear x0                          % these vars are no longer
 clear x                  % needed
 clear z

else                               % if here generate test data
 pad = rand(1,48) > .5; % generate some random data
```

```
  datain = [1 0 1 0 1 1 0 0 1 0 0 0 1 1 1 0 pad]; % with sync
  y = 2*datain - ones(1,64);
  y = kron(y,[ones(1,8) 0 -ones(1,8)]);     % symbol waveform
  y = kron(ones(1,3),y);  % gen 3 frames
  y = 0.0625*conv(y,[-ones(1,8) ones(1,8)]);  % matched filtering
  y = y + 0.25*randn(y);  % additive noise
end

z = y;                              % this is a simple agc
[m,n]=size(z);
y(1:16) = y(1:16)/std(z(1:31));     % normalize data by the
for i=17:16:n-32             % max in the block and
  s = 1.0/std(z(i-16:i+31));        % adjacent blocks
  y(i:i+15) = s*z(i:i+15);
end
i=i+15;
y(i:n)=zeros(1,n-i+1);              % clear tail
clear z;                     % free some more memory

plot(y);                     % plot agc'ed data
pause(1);

%
% next we aquire symbol clock and decimate again by this clock
% This phase consists of taking the matched filter output,
% differentiating, multipling by the matched filter output and
% filtering around twice the symbol clock frequency. The filter
% is complex, so that the phase of the filtered data may be found
% for later use in the Kalman filter symbol synchronizer
%

dif=0.5*conv([1 0 -1],y); % differentate data
y = [0 y 0];                        % pad input to same length
err = dif .* y;                     % product of each
[a,b]=size(err);             % build bpf around clock
bitfilt = [sin((-63:63)*(pi/4.15))-j*cos((-63:63)*(pi/4.15))];
bitfilt = bitfilt' .* hanning(127);% freqency
err = conv(err,bitfilt);     % band pass the data
[c,d]=size(err);
err(1:63)=[]; err(b+1:d)= [];
phase = -atan2(imag(err),real(err))/pi; % find the phase of bpf out
err = real(err)/max(real(err));
m= 400; n=450; t=m:n;               % plot some debug stuff
plot(t,y(m:n),'w',t,err(m:n),'b',t,dif(m:n),'r',t,phase(m:n),'g');
pause(1);
clear dif                    % free memory

%
%       This is the Kalman filter for tracking the symbol clock
%
%       Kalman filter parameters
%
%       X(k+1) = A * X(k) + U(k) model
%       Z(k) = C * X(k) + W(k)            observation
%
x=[10 8.35 0]';  % initial state var (phase,period,aux1,aux2)
```

```
A = [1 1 0                          % state transition matrix
    0 1 0                           % phase,period,aux1,aux2
    0 0 .8];                        % aux to decorrelate err measurements
C = [1 0 1];                        % observation matrix
Rw = 1;                             % observation noise covariance
Rx = 20*eye(3);                     % initial state covariance
Ru = [0 0    0                      % model driving covariance
    0 .00001 0
    0  0 .00001];
%
%      This is an interpolation filter for finding the
%      interpolated matched filter output. This is required
%      becuase the symbol clock may not land exactly on a
%      sample.
%

interp = firls(64,[0 .12 .13 1],[1 1 0 0]);   % simple ls fir
interp(65) = 0;
interp = interp/sum(interp(33));% normalize interp filter

j=1; k = 1;                         % init some loop vars
z = zeros(1,128);
[n,m] = size(y);
pass = 1;

while x(1) < m-10                   % x(1) is the sample #
 i = floor(x(1)); frac = x(1) - i; % i is the integer sample #
 e = -(phase(i) + 0.255*frac);% find the phase err

 Rv = C*Rx*C' + Rw;                 % update the error variance
 Rvi = inv(Rv);              % and its inverse
 G = A*Rx*C'*Rvi;                   % calculate Kalman gain
 x = A*x + 2*G*e;                   % compute prediction state
 Rx = Rx - Rx*C'*Rvi*C*Rx;          % find prediction covariance
 Rx = A*Rx*A' + Ru;                 % find state est covariance

 iindex = 9-floor(8*frac);          % interpolate sample
 yt = y(i-3:1:i+4); it = interp(iindex:8:56+iindex);
 yi = yt * it';              % this is the interp result
 if(pass > 1)                       % allow 1 frame pass to
  ze = yi - z(j);            % establish good clock
  if(ze > 1) ze = .5; end% hard limit for noise
  if(ze < -1) ze = -.5; end         % spikes
  z(j) = z(j) + ze/(pass-1);        % accumulate data each pass
 end
 loc(k,:) = [ x(2)-8 x(3) e yi];% diagnostic vector
 k=k+1; j=j+1;                      % update loop counters
 if(j > 128)                        % mod 128 for accumulation
  j = 1; pass = pass+1;% (128 1/2 symbols/frame)
 end
end

plot(z);                     % data accumulated at symbol
pause(1);                           % intervals
plot(loc);                          % plot vco freq, aux var,
pause(1);                           % phase error, and sample value
```

```
%
%       find the sync word, using correlation and display results
%
m=[1 0];% A          C          8          E
sync= [m -m  m -m  m m -m -m m -m -m -m     m m m -m zeros(1,96)];
zc = real(ifft(fft(z).*conj(fft(sync))));
zc = zc/max(abs(zc));               % normalize cross correlation data
plot(zc);                           % plot corr for debug
pause(1);                           % Oh, you want to see the results?
index = find(zc==max(zc))           % find the location of the corr peak
zs = zeros(1,128);                  % and rotate the data to the normal
zs(index) = 1;                      % orientation
zc = real(ifft(fft(z).*conj(fft(zs))));
zc = zc/std(abs(zc));
zc = zc-mean(zc(1:2:32));           % offset compensation
plot(zc(1:2:128))                   % due to flutter @ 30Hz
dataout = zc(1:2:128)>0;            % decode bits
if sim == 0                         % test for errors
 'bit errors = ',sum(xor(datain,dataout))
else
 'bit errors = ',sum(xor([1 0 1 0 1 1 0 0 1 0 0 0 1 1 1 0],dataout(1:16)))
end
byte_w = [ 8 4 2 1];                % convert bits to hex out
bytes = zeros(1,16);
hexstr=['0' '1' '2' '3' '4' '5' '6' '7' '8' '9' 'A' 'B' 'C' 'D' 'E' 'F'];
for i=0:15                          % loop for each hex digit
 t = byte_w*dataout(1+4*i:4+4*i)' + 1;
 bytes(i+1) = hexstr(t);
end
'sync  dec id   date  spare'        % display decoded data
bytes                               % here
```

CLAIMS:

1. A method for incorporating source identification data into the active picture of a video signal formed of successive video fields while maintaining the source identification data imperceptible to a viewer, the method comprising the steps of:

generating source identification data,

separating vertical sync pulses at vertical rate from the video signal,

synchronizing the source identification data to the video signal at a vertical rate in response to the vertical sync pulses,

blanking the synchronized source identification data during horizontal and vertical blanking intervals of the video signal, and

adding the synchronized source identification data to the video signal.

2. A method as claimed in claim 1 wherein the source identification data is generated by dynamically offsetting positively or negatively the video pedestal of the video signal, wherein each offset is constant for one or more fields of the successive video fields.

3. A method as claimed in claim 1 or 2 wherein the source identification data is encoded.

4. A method as claimed in any of claims 1, 2 or 3 wherein said offset has an amplitude of plus or minus 0.1 to 0.5 IRE unit.

5. Apparatus for incorporating source identification data into the active picture of a video signal formed of successive video fields while maintaining the source identification data imperceptible to a viewer, said apparatus comprising:

means for generating source identification data,

processing means for adding said synchronized source identification data to the video signal,

means for providing sync signals synchronized to the video signal, said means for providing having a vertical sync separator for separating vertical sync pulses at vertical rate from the video signal,

means for synchronizing the source identification data to the video signal by synchronizing the source identification data to the vertical rate and to the video signal in response to the vertical sync pulses, and

means for blanking said synchronized source identification data during the horizontal and vertical blanking intervals of the video signal and before the synchronized source identification data is added to said video signal.

6. Apparatus as claimed in claim 5 wherein said means for generating source identification data comprise means for

generating a synchronizing word, and wherein the means generate encoded data.

7. Apparatus as claimed in claim 5 or 6 wherein said processing means dynamically offset the video pedestal of the video signal for at least an entire field.

8. Apparatus as claimed in claim 7 wherein the video pedestals of entire fields in the successive video fields are selectively offset positively or negatively an amplitude of 0.1 to 0.5 IRE unit.

9. Apparatus as claimed in claim 5 wherein said processing means comprise a video processor receiving said video signal and outputting the video signal to said sync separator and to said blanking means, and wherein the output of said blanking means is applied to said video processor whereby the blanked and synchronized source identification data is added to the video signal.

10. Apparatus as claimed in claim 9 wherein said blanking means comprise a blanking generator receiving the video signal from the video processor and generating a combined horizontal and vertical blanking signal, and a data blanker receiving the synchronized source identification data and the combined blanking signal and outputting the blanked and synchronized source identification data to the video signal.

11. Apparatus as claimed in any one of claims 5, 6, 7, 8, 9 or 10 wherein said source identification data comprises

data identifying a source of video material, and a date code.

12. Apparatus as claimed in claim 11 wherein said source of video material comprises a specified theatre or specified video cassette recorder.

13. Apparatus as claimed in any one of claims 5, 6, 7, 8, 9, 10, 11 or 12 wherein said synchronizing word is generated by a user controlled keyboard or keypad.

14. An apparatus for incorporating source identification data not visible or disturbing to a viewer into an active picture area of a video signal formed of successive video fields, comprising:

means for receiving and processing the video signal,

means coupled to the processing means for generating a vertical sync pulse from the video signal,

means responsive to the vertical sync pulse for providing a sync word and source identification information synchronized to vertical rate,

means coupled to the providing means for blanking said synchronized sync word and source identification information during video blanking intervals, and

wherein said receiving and processing means is coupled to said blanking means for dynamically offsetting the video pedestal of the video signal for at least an entire video field to therein supply said sync word and source identification information at a very low signal level to said active picture area of the video signal.

15. A method for incorporating source identification data into the active picture of a video signal, the method comprising the steps of:

generating source data including a synchronizing word for incorporation into the active picture of a video signal, wherein said source data is source identification data,

synchronizing said source data to the video signal, and adding said synchronized source data to said video signal, wherein, to render the source data imperceptible to a viewer, said synchronized source data is blanked during the horizontal and vertical blanking intervals of said video signal,

a video pedestal of the video signal to carry the source identification data is dynamically offset for at least one entire field, and

wherein said synchronized source identification data is blanked during the horizontal and vertical blanking intervals before said synchronized source identification data is added to said video signal.

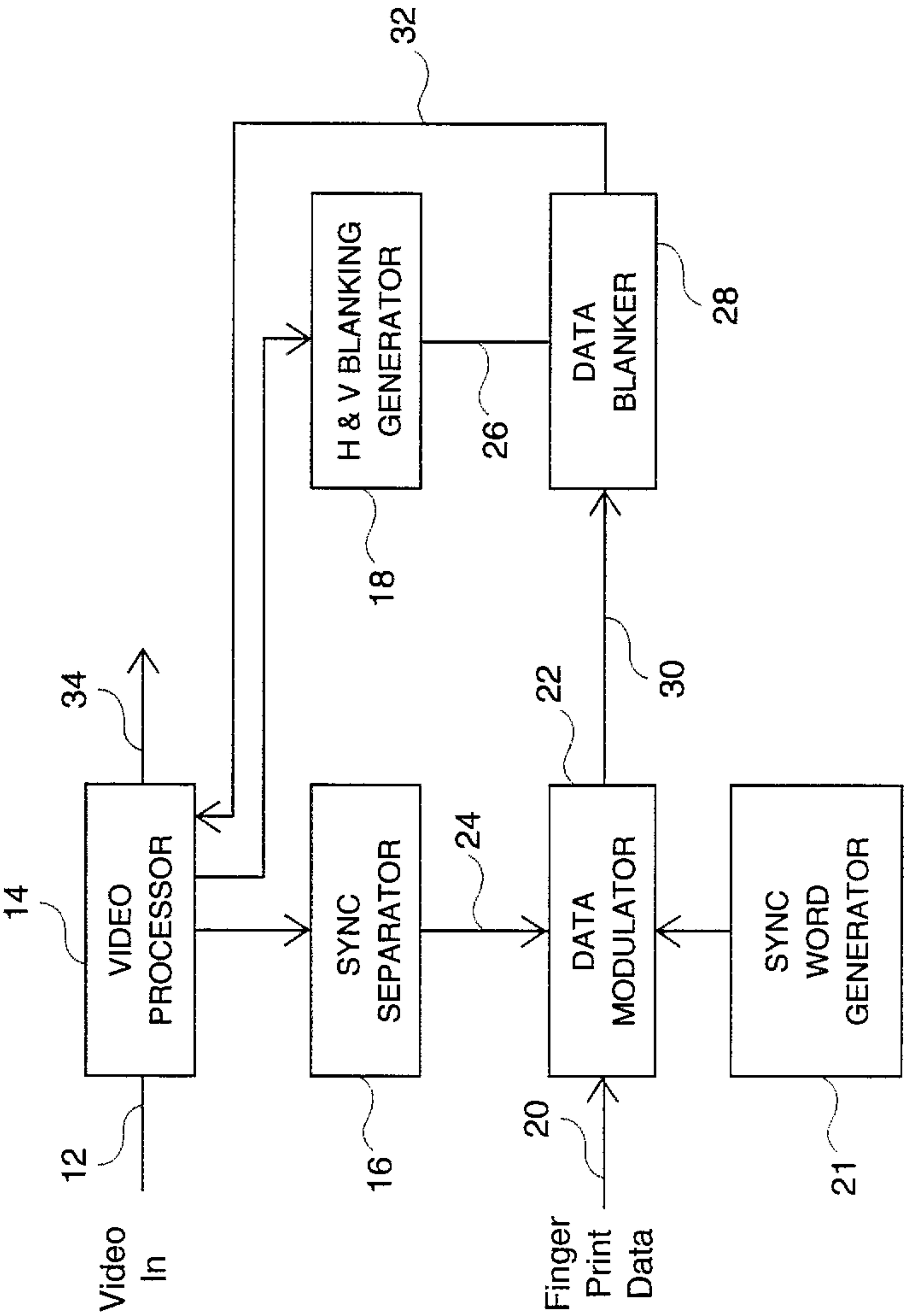
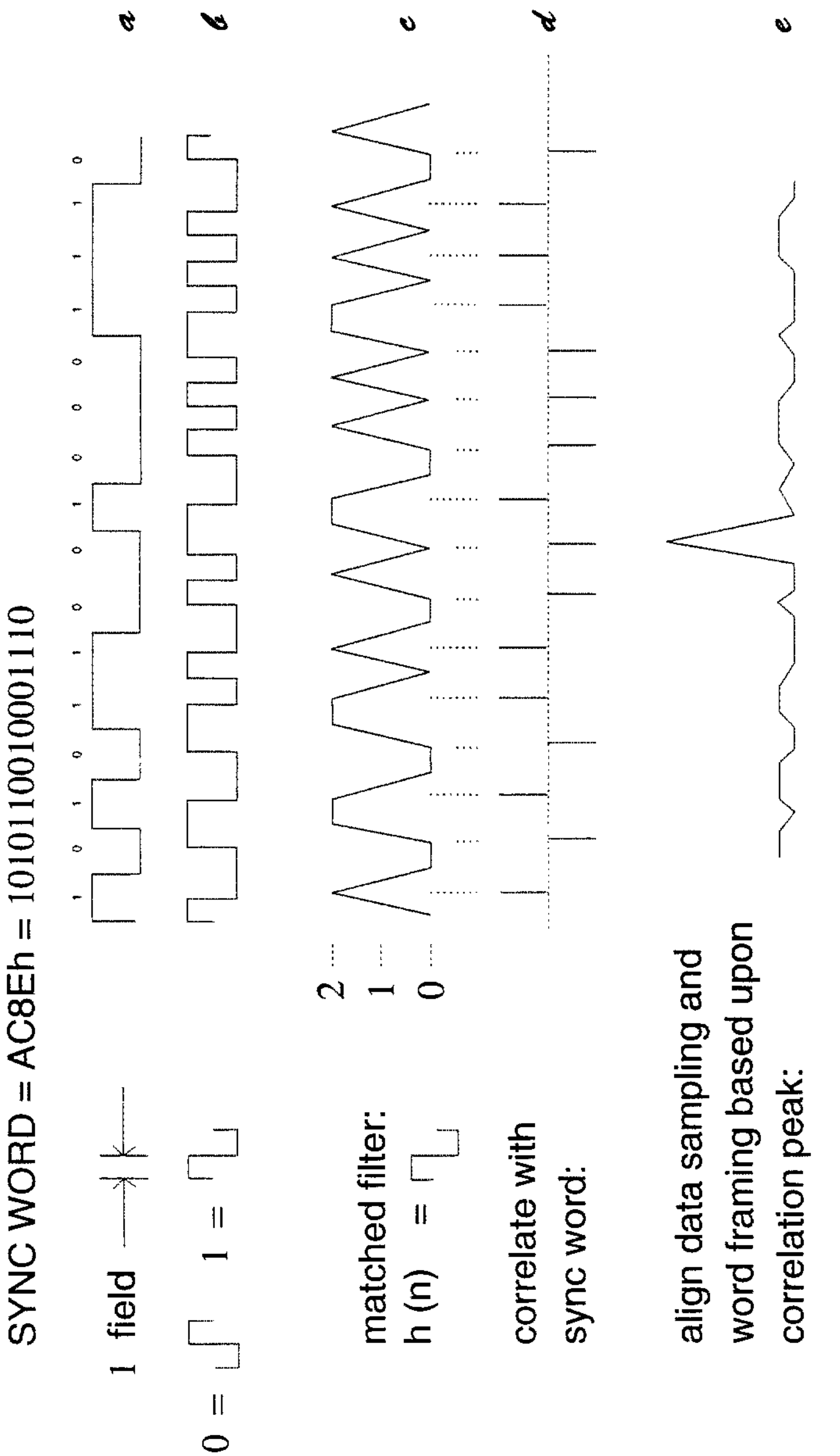
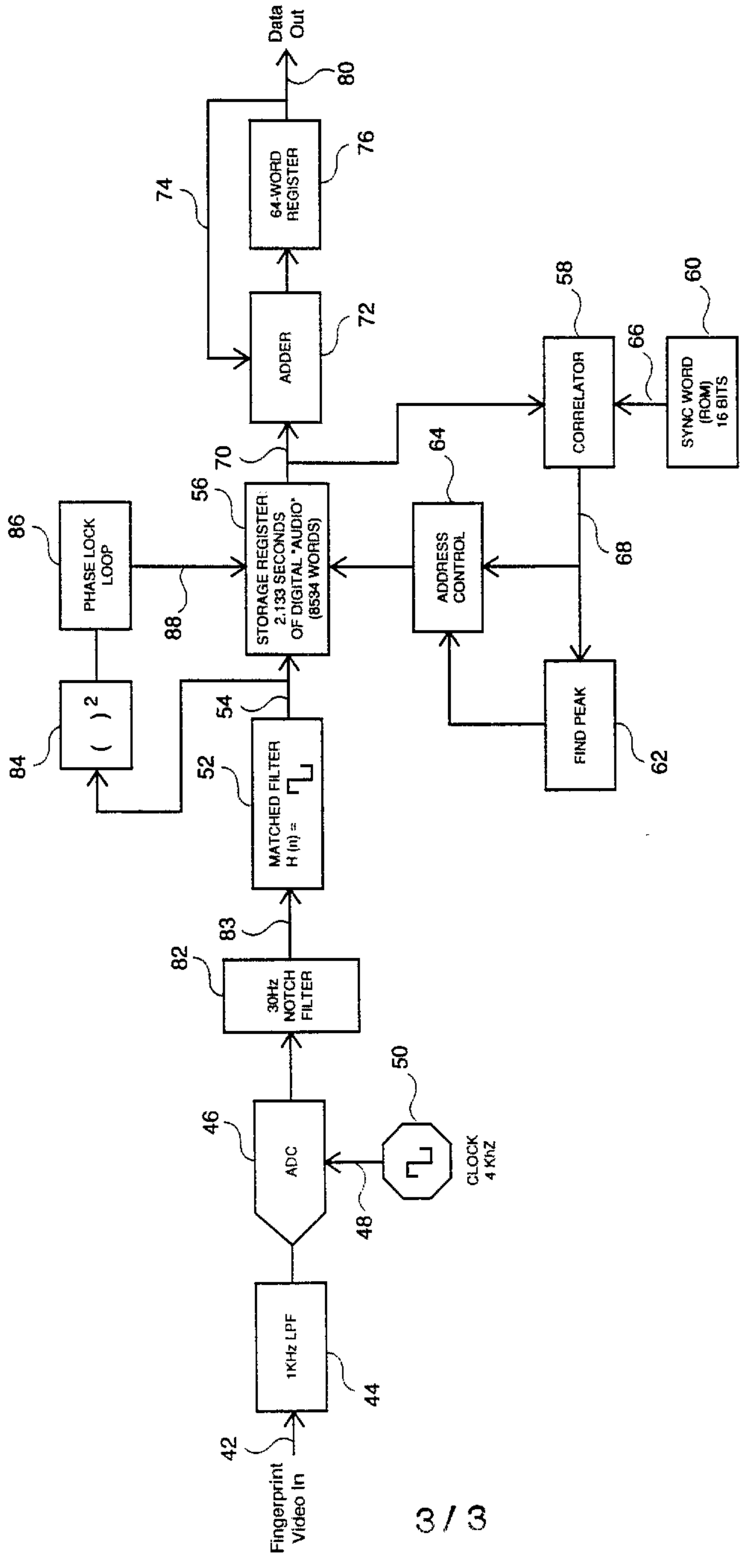
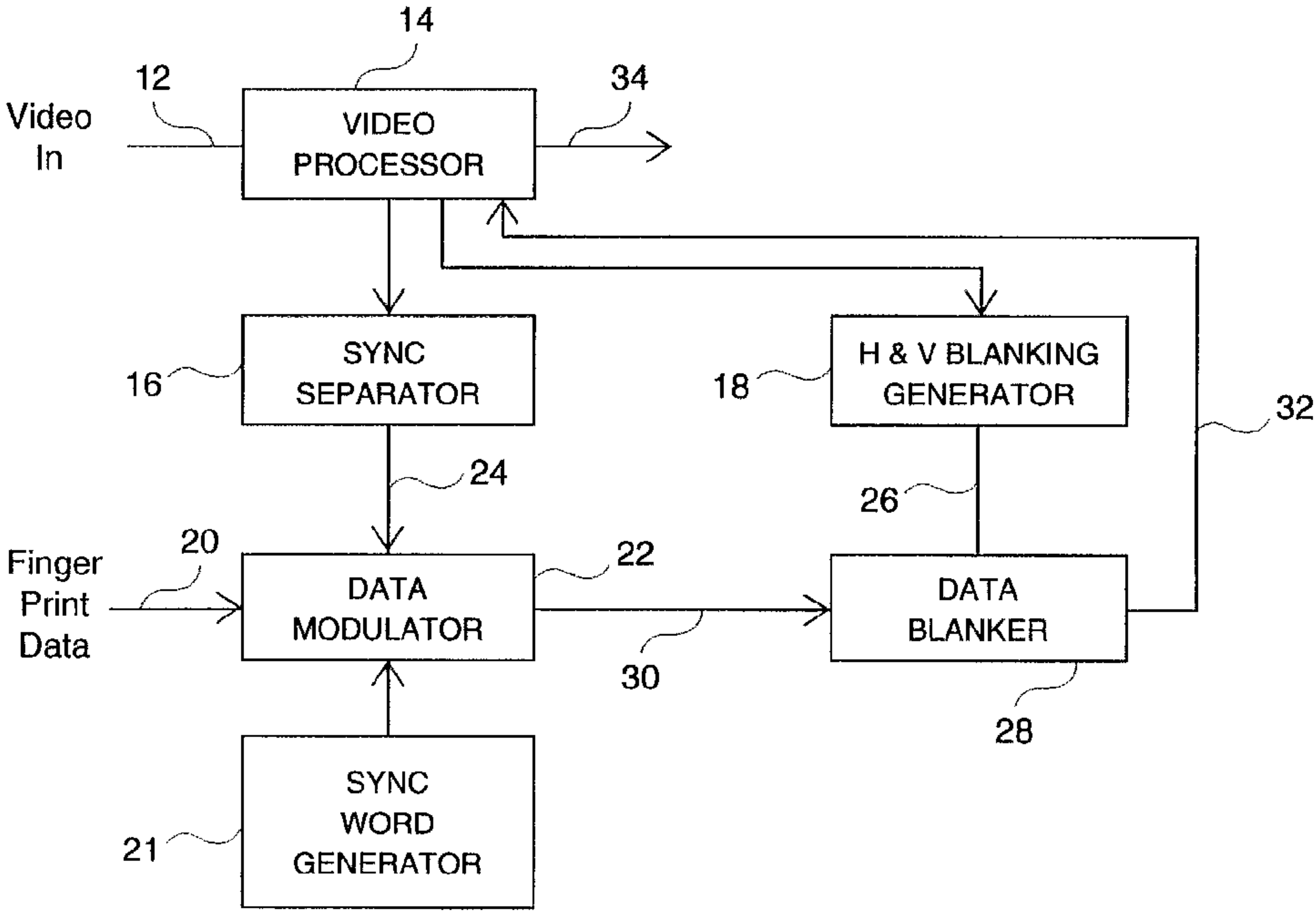

Video In
12
VIDEO PROCESSOR
14
34
32
SYNC SEPARATOR
16
24
H & V BLANKING GENERATOR
18
26
DATA BLANKER
28
30
22
DATA MODULATOR
20
Finger Print Data
SYNC WORD GENERATOR
21

Fig 1


SYNC WORD = AC8Eh = 1010110010001110
1 field
0 =
1 =
matched filter:
h (n) =
2
1
0
correlate with
sync word:
align data sampling and
word framing based upon
correlation peak:
a
b
c
d
e


Fingerprint Video In
42
1KHz LPF
44
ADC
46
48
CLOCK 4 KHz
50
30Hz NOTCH FILTER
82
83
MATCHED FILTER H (n) =
52
54
STORAGE REGISTER: 2.133 SECONDS OF DIGITAL "AUDIO" (8534 WORDS)
56
( )2
84
PHASE LOCK LOOP
86
88
70
ADDER
72
74
64-WORD REGISTER
76
80
Data Out
ADDRESS CONTROL
64
CORRELATOR
58
SYNC WORD (ROM) 16 BITS
60
66
68
FIND PEAK
62



14
12
Video In
VIDEO PROCESSOR
34
SYNC SEPARATOR
16
H & V BLANKING GENERATOR
18
32
24
26
20
Finger Print Data
DATA MODULATOR
22
DATA BLANKER
30
28
SYNC WORD GENERATOR
21