A data processing system (FIG. 9) comprises a software-oriented implementation for extracting partial response coded data (FIG. 21) from a vertical blanking interval of a television signal. The system synchronizes to a synchronization signal, such as a color burst, and extracts the multi-level coded data (FIG. 4A-4B). The system may include multiple modes having different bit transfer rates (FIG. 22) to accommodate various transmission environments and transmission rates.
FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

<table>
<thead>
<tr>
<th>AL</th>
<th>Albania</th>
<th>ES</th>
<th>Spain</th>
<th>LS</th>
<th>Lesotho</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM</td>
<td>Armenia</td>
<td>FI</td>
<td>Finland</td>
<td>LT</td>
<td>Lithuania</td>
</tr>
<tr>
<td>AT</td>
<td>Austria</td>
<td>FR</td>
<td>France</td>
<td>LU</td>
<td>Luxembourg</td>
</tr>
<tr>
<td>AU</td>
<td>Australia</td>
<td>GA</td>
<td>Gabon</td>
<td>LV</td>
<td>Latvia</td>
</tr>
<tr>
<td>AZ</td>
<td>Azerbaijan</td>
<td>GB</td>
<td>United Kingdom</td>
<td>MC</td>
<td>Monaco</td>
</tr>
<tr>
<td>BA</td>
<td>Bosnia and Herzegovina</td>
<td>GE</td>
<td>Georgia</td>
<td>MD</td>
<td>Republic of Moldova</td>
</tr>
<tr>
<td>BB</td>
<td>Barbados</td>
<td>GH</td>
<td>Ghana</td>
<td>MG</td>
<td>Madagascar</td>
</tr>
<tr>
<td>BE</td>
<td>Belgium</td>
<td>GN</td>
<td>Guinea</td>
<td>MK</td>
<td>The former Yugoslav Republic of Macedonia</td>
</tr>
<tr>
<td>BF</td>
<td>Burkina Faso</td>
<td>GR</td>
<td>Greece</td>
<td>ML</td>
<td>Mali</td>
</tr>
<tr>
<td>BG</td>
<td>Bulgaria</td>
<td>HU</td>
<td>Hungary</td>
<td>MN</td>
<td>Mongolia</td>
</tr>
<tr>
<td>BJ</td>
<td>Benin</td>
<td>IE</td>
<td>Ireland</td>
<td>MR</td>
<td>Mauritania</td>
</tr>
<tr>
<td>BR</td>
<td>Brazil</td>
<td>IL</td>
<td>Israel</td>
<td>MW</td>
<td>Malawi</td>
</tr>
<tr>
<td>BY</td>
<td>Belarus</td>
<td>IS</td>
<td>Iceland</td>
<td>MX</td>
<td>Mexico</td>
</tr>
<tr>
<td>CA</td>
<td>Canada</td>
<td>IT</td>
<td>Italy</td>
<td>NE</td>
<td>Niger</td>
</tr>
<tr>
<td>CF</td>
<td>Central African Republic</td>
<td>JP</td>
<td>Japan</td>
<td>NL</td>
<td>Netherlands</td>
</tr>
<tr>
<td>CG</td>
<td>Congo</td>
<td>KE</td>
<td>Kenya</td>
<td>NO</td>
<td>Norway</td>
</tr>
<tr>
<td>CH</td>
<td>Switzerland</td>
<td>KG</td>
<td>Kyrgyzstan</td>
<td>NZ</td>
<td>New Zealand</td>
</tr>
<tr>
<td>CI</td>
<td>Côte d'Ivoire</td>
<td>KP</td>
<td>Democratic People's Republic of Korea</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CM</td>
<td>Cameroon</td>
<td>KR</td>
<td>Republic of Korea</td>
<td>PL</td>
<td>Poland</td>
</tr>
<tr>
<td>CN</td>
<td>China</td>
<td>KZ</td>
<td>Kazakhstan</td>
<td>PT</td>
<td>Portugal</td>
</tr>
<tr>
<td>CU</td>
<td>Cuba</td>
<td>LC</td>
<td>Saint Lucia</td>
<td>RO</td>
<td>Romania</td>
</tr>
<tr>
<td>CZ</td>
<td>Czech Republic</td>
<td>LI</td>
<td>Liechtenstein</td>
<td>RU</td>
<td>Russian Federation</td>
</tr>
<tr>
<td>DE</td>
<td>Germany</td>
<td>LK</td>
<td>Sri Lanka</td>
<td>SD</td>
<td>Sudan</td>
</tr>
<tr>
<td>DK</td>
<td>Denmark</td>
<td>LR</td>
<td>Liberia</td>
<td>SE</td>
<td>Sweden</td>
</tr>
<tr>
<td>EE</td>
<td>Estonia</td>
<td></td>
<td></td>
<td>SG</td>
<td>Singapore</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SI</td>
<td>Slovenia</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SK</td>
<td>Slovakia</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SN</td>
<td>Senegal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SZ</td>
<td>Swaziland</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TD</td>
<td>Chad</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TG</td>
<td>Togo</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TJ</td>
<td>Tajikistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TM</td>
<td>Turkmenistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TR</td>
<td>Turkey</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TT</td>
<td>Trinidad and Tobago</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UA</td>
<td>Ukraine</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UG</td>
<td>Uganda</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>US</td>
<td>United States of America</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>UZ</td>
<td>Uzbekistan</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>VN</td>
<td>Viet Nam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>YU</td>
<td>Yugoslavia</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ZW</td>
<td>Zimbabwe</td>
</tr>
</tbody>
</table>
APPLICATION FOR PATENT

Title: METHOD AND SYSTEM FOR TRANSMITTING AND DECODING DATA IN A SIGNAL

CROSS-REFERENCES TO RELATED APPLICATIONS

This application is a continuation-in-part of U.S. Patent Application Number 08/841,794, filed May 5, 1997, and is a nonprovisional application based on U.S. Provisional Application No. 60/052,890, filed July 17, 1997.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates to decoding signals, and more particularly, to methods and apparatus for decoding teletext and/or closed caption data embedded within signals such as video signals.

2. Description of the Related Art

Although watching television has been the national pastime for people of the United States and other countries for many years, the computer revolution has begun to expand the usefulness of television beyond its traditional role. Technological developments in recent decades have led to the advent of closed caption television, which integrates a secondary signal into the main television signal. The closed caption data may be decoded and displayed separately on the display system, for example to provide subtitles to hearing-impaired viewers.
An even more recent development is the introduction of teletext, which provides a method of broadcasting data by integrating it into the television signal. Like closed caption data, the teletext information is integrated into the television signal without disrupting the television image. As information technology like the Internet develops, teletext is likely to assume a greater role in information systems.

Decoder systems for extracting the teletext and closed caption information from the television signal have been developed and made publicly available. These decoder systems are typically hardware systems integrated into the television circuitry. As the television signal is received by the television, the decoder system continuously monitors the incoming signal for codes indicating that teletext or closed caption data follows. The decoder system then decodes the data “on the fly” and provides it to the user.

Although conventional decoder systems effectively decode and present the data, they suffer several drawbacks. One of the most obvious problems is the cost. The circuitry required to perform the processing in real time is typically dedicated solely to detecting and decoding the teletext and closed caption data. Designing and manufacturing such dedicated circuitry is typically costly. In addition, such systems suffer limited flexibility, as alterations to hardware designs are often costly and difficult to implement, especially after the decoder system has been installed. Further, such systems may be subject to interference that, while of limited effect on the television image, may significantly affect the detection and decoding of the data.

SUMMARY OF THE INVENTION

A teletext and closed caption data decoding system according to various aspects of the present invention provides a software-oriented implementation of the decoding process. The software-oriented implementation tends to reduce the cost of the implementation, as conventional circuitry may be used to perform the decoding and related functions. In addition, the software-oriented decoding system provides for improved efficiency with respect to
processing resources, which improves its usefulness in a computer environment. Further, the software may be relatively easily redesigned, tested, and implemented.

In particular, a decoder system according to one embodiment includes a processing unit, such as a microprocessor or a microcontroller, which implements both a closed caption module and a teletext module. The system also facilitates high speed data transmission. In particular, the data transmission rate is enhanced using partial response coding. The system may also facilitate the transfer of data at different rates. Lower rates may be used for robustness, whereas higher rates may be employed when the signal quality is sufficient. In addition, the data distribution may be clocked to the color burst signal, providing high data clock speeds than conventional run-in clocks and greater portions of the vertical blanking available for data injection.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

The subject matter of the invention is particularly pointed out and distinctly claimed in the concluding portion of the specification. The invention, however, both as to organization and method of operation, may best be understood by reference to the following description taken in conjunction with the claims and the accompanying drawing, in which like parts may be referred to by like numerals:

Figure 1 is a block diagram of a general television system;

Figures 2A-B are diagrams of portions of an NABTS video signal and a WST video signal, respectively;

Figure 3 is a diagram of a portion of a closed caption signal;

Figures 4A-B are tables illustrating the position and amplitude of multi-level signals in mode 1 and mode 2;

Figure 5 is a block diagram of a television system including a secondary transmitter;

Figure 6 is a block diagram of an encoder;

Figure 7 is a functional diagram of the functions performed by an encoder;

Figure 8 is a diagram of the data organization of a data bundle provided by an encoder;
Figure 9 is a block diagram of a receiver according to various aspects of the present invention;
Figure 10 is a block diagram of a system for analyzing closed caption data;
Figure 11 is a flow diagram of a process for tracking and extracting closed caption data from a digitized television signal;
Figures 12A-E are flow diagrams of a process for recovering the clock synchronization signal in a closed caption signal;
Figure 13 is a block diagram of a system for analyzing teletext data;
Figure 14 is a flow diagram of a process for tracking and extracting teletext data from a digitized television signal;
Figures 15A-E are flow diagrams of a process for recovering the clock synchronization signal in a teletext signal;
Figure 16 is a block diagram of a system for data interpolation;
Figures 17A-C are block diagrams of systems for equalization;
Figure 18 is a table listing of exemplary acceptable framing codes exhibiting a one-bit error;
Figure 19 is a flow diagram of a process for extracting closed caption and teletext data from a television signal;
Figure 20 is a block diagram of a system for analyzing partial code response encoded data;
Figure 21 is a flow diagram of a process for tracking and extracting partial code response encoded data from a digitized television signal;
Figure 22 is a table of characteristics for partial response code operating modes 1 and 2;
Figure 23 is a graph of the relative amplitude of the partial response code signal as a function of frequency;
Figure 24 is a graph of the impulse response of the spectrum of Figure 23;
Figure 25 is an eye diagram for a system using a binary partial response scheme; and
Figure 26 is an eye diagram for a system using a 4-ary partial response scheme.
DETAILED DESCRIPTION OF PREFERRED EXEMPLARY EMBODIMENTS

Referring now to Figure 1, a generalized signal transmission and reception system 100 comprises a transmitter 102, a communications medium 104, and a receiver 106. The transmitter 102 suitably comprises a source of television or other video signals to be transmitted to the receiver via the communications medium 104. The communications medium 104 transmits the signals generated by the transmitter 102 to the receiver 106. The receiver 106 decodes the signals and provides the teletext, closed caption, and other data, and suitably video information, to a user.

The 525-line 60-fields-per-second NTSC television signal typically includes certain lines in the vertical blanking interval to allow the receiver to synchronize and perform vertical retrace before the active video picture begins. The vertical blanking interval includes lines 1 through 21 which may contain vertical synchronization pulses (lines 1 to 9), vertical interval test signals (VITS), vertical interval reference (VIR) signals, and digital data signals. Vertical blanking interval data transmission is commonly defined as any digitally coded information inserted between lines 10 through 21 (fields 1 and 2) of the analog television signal.

Full field data transmission uses the active part of the video signal as well as the VBI for data insertion. Lines 10 through 262 in field 1 and the corresponding lines in field 2 may be used for data transmission.

Teletext information is suitably included in the VBI according to a conventional specification, such as North American Broadcast Teletext System (NABTS) or Word System Teletext (WST). Similarly, the closed caption data may be integrated into the signal generated by the transmitter 102 in accordance with conventional standards for closed caption transmissions, for example EIA-608 standards. It should be noted, however, that various aspects of a decoding system according to the present invention are not so limited, and various aspects of the present decoding system and methods may be used in any analogous signal receiver for decoding data in a signal.

Referring now to Figures 2A and 2B, the teletext data is suitably integrated into the VBI portion of the video signal. In the present embodiment, the video signal is a standard
NTSC video signal, and the teletext information is suitably provided within a selected portion of the video signal, for example between lines 10 and 20 of both the odd and even fields of the video signal. Alternatively, teletext information may be transmitted in any or all video lines in place of an actual video line, for example a full field teletext transmission. To facilitate tracking of the signal and retrieval of the teletext signal, the video signal in the VBI may include information, for example in the form of pulses or waveforms. The teletext portion of the signal suitably comprises: a video synchronization pulse; a color burst; a clock synchronization signal; a byte synchronization signal; a set of prefix values; and a data block. The video synchronization pulse is suitably provided in accordance with conventional television specifications to indicate the beginning of a line. Following the video synchronization pulse, a color burst in the chroma carrier is transmitted to lock the phase of the receiver’s 106 chroma oscillator for demodulation of the transmitted color signals. It should be noted that the color burst may not be included in some types of signals, for example, black and white NTSC video signals.

After the horizontal synchronization signal and the color burst (if present), the transmitter 102 may transmit the clock synchronization signal, suitably comprising a sine waveform, to facilitate synchronization of the receiver’s 106 data reading components to the transmitted signal. The clock synchronization signal suitably serves to mark the optimal bit position for the remaining portion of the signal. The duration of the clock synchronization signal may vary according to the specifications for the signal. For NABTS and WST signals, however, the clock synchronization signal comprises eight cycles. By tracking the phase of the clock synchronization signal, the receiver 106 can synchronize to track the following data. As described below, however, other signals, such as the color burst, may be used for synchronization purposes, allowing the clock synchronization signal to be ignored or omitted.

After the clock synchronization signal is transmitted to facilitate clock synchronization, the transmitter 102 transmits the byte synchronization signal. The byte synchronization signal suitably includes a framing code, for example an eight-bit code, to forecast the transmission of the prefix bytes and the data block. Thus, when this code is identified, the receiver 106 may determine when the prefix bytes and the data block are about to be received. Accordingly, the
transmitter 102 transmits the prefix bytes and the data block after the framing code. The prefix bytes suitably include Hamming encoded bytes. For example, an NABTS signal includes a five-byte packet, and a WST signal includes a two-byte packet, though the prefix’s size is variable depending on the values in the first two bytes. Packet structure information, packet address, and continuity index information are suitably included in the prefix bytes.

Finally, the transmitter 102 transmits the data block, which optionally includes an appropriate suffix. For an NABTS transmission, the data and optional suffix block comprise 28 bytes, while the corresponding block in a WST transmission is typically 29 bytes for an NTSC television system. The data block suitably contains the actual data associated with the transmission and is to be analyzed by the receiver 106.

The closed caption signal structure is suitably similar to that of the teletext signal. In the present embodiment and in conjunction with conventional closed caption signals, however, the signal frequency is lower and the data block includes only two bytes with no prefix bytes. Unlike the teletext signal, the closed caption information is conventionally integrated into line 21 of one or both (odd and even) fields. The closed caption information may, however, be integrated into any VBI lines, or even any video line.

Referring now to Figure 3, the video signal includes the closed caption signal following the byte synchronization signal. The closed caption signal suitably comprises: a horizontal synchronization pulse; a color burst; a clock synchronization signal; a byte synchronization signal; and a data block. The horizontal video synchronization pulse is suitably provided in accordance with conventional television specifications to indicate the beginning of the line. Following the video synchronization pulse, the color burst in the chroma carrier is transmitted for demodulation of the transmitted color signals as previously described.

After the color burst, the transmitter 102 transmits the run-in clock synchronization signal to facilitate synchronization of the receiver 106 with the video signal. The duration of the closed caption clock synchronization signal is typically seven cycles long. After the clock synchronization signal, the transmitter 102 transmits the byte synchronization signal. For closed caption data, the byte synchronization signal suitably comprises a three-bit framing code. Thus, the arrival of the three-bit framing code indicates that the data block is about to
follow. Accordingly, the transmitter 102 transmits the data. For a closed caption transmission, the data block comprises only two bytes containing the actual data associated with the closed caption transmission.

An alternative embodiment of a data transmission and reception system according to various aspects of the present invention can deliver higher net data rates than conventional systems. In an exemplary embodiment, net data rates may exceed 192 kbps in the VBI, or 12 Mbps in full field mode. The major differences between the present high-speed system and other systems are filtering, multi-level coding, synchronization with the color burst, and multiline packetizing.

Although the signal in the vertical blanking interval may include a run-in clock, the data frequency may be any suitable frequency. In the present embodiment, the data frequency is associated with a signal integrated in to the television signal, such as the color burst, so that the run-in clock may be omitted from the signal. The present system suitably includes a clock doubler associated with the color burst so that the actual data clock signal oscillates at 7.15909 MHz. The system further may also operate in conjunction with various operating modes facilitating the transfer of data at different rates. For example, the present system operates in two operating modes: mode 1 has one bit encoded per symbol and can be used for improved robustness, and mode 2 has two bits per symbol and may be used for higher transfer rates. In mode 1, the capacity of each line is suitably 360 bits for a throughput of 21.6 kbps per line. In mode 2, the capacity of each line is suitably 720 bits for a throughput of 43.2 kbps per line, with a bit transmission rate of 14.31818 Mbps. Mode 2 may require better signal quality than mode 1 or conventional NABTS signals. Further, it should be noted that the partial response coding used allows the detection of certain errors at the data link level. In the present embodiment, however, the forward error correction method is defined at the network level.

The symbol clock rate may be any suitable frequency. As described above, in a color television system in accordance with the present embodiment, the symbol rate may be taken as double the color burst frequency, conventionally 3.579545 MHz, to generate a symbol clock rate of 7.159090 MHz +/- 25 Hz. Thus, in mode 1, each symbol contains one bit of information and is transmitted every symbol clock. The instantaneous bit rate obtained is
7.159090 Mbits/s +/- 25 bits/sec. In mode 2, each symbol contains 2 bits of information. The instantaneous bit rate obtained is 14.31818 Mbits/s +/- 50 bits/sec. The half-amplitude point of the first transition from level 0 to the highest level (in both modes) of the clock synchronization sequence is 9.778 microseconds +/- 0.014 microseconds. Characteristics for both system operating modes are further set forth in Figure 22.

Referring now to Figure 23, the partial response coding modulation with class 1 signaling has a cosine-shaped bell spectrum for relative amplitude as a function of frequency. All of the energy distribution of that spectrum falls under half the symbol frequency, that is 3.58 MHz. The relative amplitude as a function of frequency as shown in Figure 23 is functions according to the equation:

\[ A_r(t) = 2\cos[2\pi f(T/2)] \]

where T is the period of the symbol clock rate, or (1/7.15909).

The corresponding impulse response of that spectrum is shown in Figure 24, wherein each vertical line represents a sample point. The impulse response in the present embodiment may be expressed in accordance with the following equation:

\[ h_i = \frac{4/\pi}{(\cos(\pi t/T)/(1 - 4t^2/T^2))} \]

The present system uses class 1 partial response coding, for example as classified by E.R. Kretzmer, "Generalization of a Technique for Binary Data Communication," IEEE Trans. Comm. Tech., vol. COM-14, February 1966, pp. 67-68. The partial response coding, also called correlative coding, introduces a correlation between the amplitude of successive pulses to form a multilevel signal. This technique increases the bandwidth efficiency, and may be implemented without requiring an excessively high-quality low-pass filter. For example, in the present system, a class 1 correlative digital filter is relatively simple to implement and its
spectral energy distribution is suitable for implementation on television signals. In the present embodiment, partial response coding is suitably performed by adding two successive impulses:

\[ Y_k = A_k + A_{k-1} \]

This coding, followed by appropriate filtering, results in a cosine-shaped energy spectrum such that all of the energy distribution tends to fall under half the symbol frequency, which is approximately 3.58 Mhz in the present embodiment. For example, this coding can be realized with a cosine-shaped low-pass filter.

Because correct decoding of each symbol depends on the correct decoding of the previous symbol, the above modulation tends to propagate errors. To avoid this problem, the data may be pre-coded prior to the partial response coding. The pre-coding for operating mode 2 is implemented as:

\[ B_k = \text{Mod}_4 (A_k - B_{k-1}) \]

The resulting sequence is then applied to the digital coding of the partial response:

\[ Y_k = B_k + B_{k-1} \]
\[ = (\text{Mod}_4 (A_k - B_{k-1})) + B_{k-1} \]

This equation indicates that the value of \( Y_k \) can be decoded according to the simple rule:

Decoded \( A_k = \text{Mod}_4 (Y_k) \)

Each symbol is then dependent only on the knowledge of the current sample, which eliminates or reduces error propagation. Similarly, the same rules can be applied to operating mode 1 by replacing modulo 4 by modulo 2.
For data transmission in accordance with various aspects of the present embodiment, the nominal data levels are 70 +/- 2 IRE units and 0 +/- 2 IRE units for the highest and the lowest levels respectively. For mode 2, the seven levels of the multi-level modulation used are suitably equally spaced between these two levels, which result in approximately 11.7 IRE units between each level for mode 2. Similarly, for mode 1, the three levels are equally spaced between these maximum and minimum levels so that approximately 35 IRE units separate the levels for mode 1.

The logical values resulting from the use of the pre-coding and the three-level partial response signaling of mode 1 are shown in Figure 4B. The logical values resulting from the use of the pre-coding and the seven-level partial response signaling of mode 2 are shown in Figure 4A. The data waveform may contain minimal overshoot due to modulation filtering; consequently, the peak-to-peak data amplitude may exceed the nominal data amplitude. Figures 25 and 26 illustrate eye diagrams for a system using a binary and 4-ary partial response scheme clocked at 5.727272MHz.

As described above, in the present embodiment, all of the energy in the frequency spectrum falls under 4.2 MHz in the video baseband, so there are no out-of-band emissions. Thus, audible artifacts or adjacent channel interference are minimized because no out-of-band emissions are induced. Further, visual degradation of the television signal is minimized because the data are transmitted during the vertical blanking interval.

Data is suitably extracted from the video signal by the receiver 106 and made available immediately. Implementation issues may induce delays between the moment that data is acquired by the receiver 106 and the moment that it is made available to the application. For instance, some receiver 106 implementations may signal the application upon each vertical synchronization pulse to indicate that data has accumulated, thus inducing an artificial delay of up to 28 frames, about a half-second.

The structure of the data line suitably consists of a string of 368 symbols. The first seven symbols constitute the synchronization sequence, the next 360 symbols are the data and the last symbol is the "forced zero" symbol. Symbols 0 through 7 constitute the framing code and may be selected to distinguish mode 1 from mode 2. In an exemplary embodiment, the
receiver is initiated in one mode, such as mode 2, and remains in its current mode until a framing code designating another mode is received. Also, the framing code is suitably used as a reference for byte synchronization. In the present embodiment, the synchronization sequence designating mode 1 is 0011012. For mode 2, the synchronization sequence is 3035552. The forced zero symbol brings the signal back to a blank level. Two clock pulses are necessary to do so, which is the purpose of the forced zero bit. The forced zero is suitably inserted after the pre-coding.

In a system according to various aspects of the present invention for transmitting and receiving the television with the embedded data signal, the transmitter 102 suitably comprises any source of signals, such as video signals including a television signal or the like, including a television broadcast company, a cable service provider, a television camera, or a videocassette system. In the present embodiment, the transmitter 102 suitably generates a conventional video signal, for example according to NTSC standards. In addition, the signals generated by the transmitter 102 suitably include an integrated secondary signal, such as teletext and/or closed caption data in selected blanking intervals. In particular, the transmitter suitably generates signals having baseband teletext and/or closed caption information transmitted in a vertical blanking interval (VBI) of the video signal.

Referring to Figure 5, an alternative exemplary signal transmission and reception system 550 comprises includes a primary transmitter 552, the communication medium 104, and a secondary transmitter 554. The primary transmitter 552 suitably transmits a main television signal, such as a national broadcast, with an embedded data signal. The signal is modulated by a modulator 556 and transmitted via the communications medium 104 to the secondary transmitter 554. The secondary transmitter 554 receives the signal, demodulates it, and retrieves the data embedded in the television signal. The secondary transmitter 554 then suitably combines the received television signal with a secondary television signal, such as a local television signal, and encodes the digital data into the secondary television signal. The secondary television signal with the embedded data is then transmitted to the receiver 106. This embodiment accommodates inclusion of the original data signal in local television signals.
The primary transmitter 552 may comprise any suitable system for embedding a data signal in the vertical blanking interval of a television signal. In the present embodiment, the primary transmitter 552 comprises a communications server 560 and an encoder 562. The server 560 suitably performs several tasks, including receiving data from one or more data sources, storing the data, and providing the data to the encoder 562. The server 560 may also perform various other tasks according to the particular needs or desires of a service provider. In the present embodiment, the server 560 comprises a suitable processing system, such as an INTEL PENTIUM-based computer, executing server software designed to perform selected tasks. For example, the server 560 may comprise software running in a multi-tasking environment including a collection of modules that can be combined to assemble different packages according to a service provider's needs.

The server 560 may be configured to accomplish most of its tasks associated with the higher levels of a layered architecture compliant with the seven-layered reference model of the International Organization for Standardization (OSI), including data acquisition, user administration, addressing/encryption, file management, transmission, scheduling, and stream support (real-time or pass-through data transmission). The server 560 may also include other modules, such as network monitoring and management, data compression, billing, hybrid network support, and optimization. Further, the server 560 preferably includes support for multiple data broadcasting protocols, such as NABTS.

The encoder 562 suitably comprises a high-speed encoder configured to accept data from the server 560, for example via an RS-422 port at speeds of up to 384kbps, encode them, and insert them at the baseband level on the lines of the television signal reserved for data broadcasting. The net throughput of the encoder 562 depends on the operating mode and the number of lines allocated to data transmission. The encoder 562 suitably performs the operations generally associated with the two lower layers of the OSI reference model, physical and data link. Preferably, the encoder 562 is capable of handling, on the output side, the complete VBI bandwidth.

Referring now to Figure 6, in the present embodiment, the encoder 562 comprises a signal generator 550. The signal generator 550 receives the main television signal and the data
and embeds the data in the vertical blanking interval. The signal generator 550 may include any suitable component for inserting the data into the vertical blanking interval, such as a Tektronix VITS 200 NTSC VITS Inserter. The signal generator 550 suitably comprises: a multiplexer 552; a digital-to-analog converter (DAC) 554; a pattern generator 556; and a codec 558. In the present embodiment, the multiplexer 552, the DAC 554, and the pattern generator 556 are integrated into the VITS inserter, and the codec 558 is included on a piggyback board connected to the VITS inserter. The multiplexer 552 selects an input signal from multiple input signals and transmits it to a single output, and may comprise any device for performing this function. In the present embodiment, the multiplexer 552 receives the main television signal at a first input and the digital data signal at a second input. The multiplexer 552 transmits either the main television signal or the digital data according to a control signal. The control signal selects the digital data for transmission by the multiplexer 552 at selected times, for example during the vertical blanking interval and when digital data is available to be transmitted, and otherwise suitably transmits the main television signal.

The pattern generator 556 suitably generates a conventional vertical interval test signal (VITS) or other appropriate test signal for insertion at selected points in the television signal. The pattern generator 556 may comprise any suitable mechanism for generating any desired pattern or signal. Signals from the pattern generator 556 and the codec 558 are provided to the DAC 554, which converts the digital data to a corresponding analog signal. The output of the DAC 554 is provided to the multiplexer 552.

Codec 558 receives data from a data source and formats it for transmission. In the present embodiment, the codec 558 includes a serial controller 570 and a data generator 572. The serial controller 570 receives the data and processes the data according to selected criteria. For example, the serial controller 570 suitably comprises a universal asynchronous receiver/transmitter (UART) which receives the serial data, transmits the data to the data generator 572, and performs various functions, for example identifying start bits. The data generator 572 receives the serial data and creates a corresponding waveform for conversion to analog format and insertion into the television signal. For example, the data generator 572 may convert the serial data into multi-bit words and separate the data into lines for
transmission. The resulting multi-bit digital signal is then provided to the DAC 554 via a bus 564.

The encoder 562 suitably performs several functions to facilitate transmission of the data in the VBI. For example, referring now to Figures 7 and 8, the encoder 562 accumulates data into blocks (step 750) and randomizes the data (step 752), which improves transmission characteristics of the resulting signal. A correction code is then suitably added to the end of each line of data, such as a nine-byte Reed-Solomon code to facilitate forward error correction (step 754). A final line of forward error correction codes, such as Reed-Solomon codes, is also suitably appended to the end of the block. In the present embodiment, each block includes 27 lines of data, wherein each line includes 83 bytes. The error correction codes are suitably based on data organization. In this configuration, the block of data is independent of the video timing.

The encoder further suitably adds a block header, such as a low frequency five-byte block header, and a known training sequence for training an equalizer associated with the receiver 106 is also suitably added to the beginning of each block of data (step 756). The header may include any desired information. In the present embodiment, the header includes indicia to identify which lines of the signal include data to form multi-line data packets. As a result, the receiver 106 can determine which lines contain data by checking the information in the header, and then only processing those lines. Lines that do not contain data are suitably ignored, thus reducing processor requirements.

The encoder 562 further converts the serial data bits into symbols for transmission (step 758). Data transmitted in accordance with various aspects of the present invention may be coded using partial response coding (PRC). PRC signals suitably include multiple magnitude levels, such as three or seven, in contrast to conventional binary signals. The encoder 562 further suitably pre-codes the signals for error detection and correction (step 760)(as described in greater detail below and includes a filter system, including, for example, a root-square cosine-shaped filter (step 762). The encoder 562 may further include pre-equalization filtering (step 764), for example to improve the signal strength and/or clarity. In the present embodiment, the bit-to-symbol conversion and the pre-coding are performed using
suitable hardware. Further, multiple functions, such as the bit-to-symbol conversion, the precoding, and the filtering may be performed using a lookup table associated with the encoder 562.

The resulting data is then combined with the television signal in the VBI. The television signal is provided to a color clock phase lock loop (step 766) which generates a signal based on the color burst frequency. This signal is provided to a clock doubler (step 768), which suitably generates a signal at double the frequency of the color burst. The resulting signal is provided to a VBI injector (step 770), such as the multiplexer 552, which injects the data signal into the television signal during the VBI in conjunction with the doubled color burst signal as a clock signal for the data insertion.

Referring again to Figure 5, the secondary transmitter 554 receives the combined television and data signal, demodulates it, and provides for the addition or substitution of a local television signal. The secondary transmitter 554 suitably includes a demodulator 580; a signal selector 582; and a signal bridge system 584. The demodulator 580 receives the signal received from the primary transmitter 552, demodulates it, and provides the demodulated combined television and data signal to the signal selector 582 and the signal bridge system 584. In the present embodiment, the signal selector 582 replaces the received television signal with a local television signal, for example at the option of the station operator. The resulting signal is then provided to the signal bridge system 584, which embeds the data signal into the VBI of the signal transmitted by the signal selector 582.

The signal bridge system 584 decodes the data from the received television signal and encodes the same information into the television signal generated by the signal selector 582. To effect these tasks, the signal bridge system 584 includes a data decoder 586, like the decoder associated with the receiver 106 described in greater detail below. Further, the signal bridge system 584 includes an encoder 588, such as the encoder 562 described above with respect to the primary transmitter 552. The data decoded from the primary transmitter 552
signal is then provided to the data generator for re-encoding and injection into the VBI of the local television signal.

Referring again to Figure 1, the transmitter 102 transmits the video signal including the teletext and/or the closed caption information to the receiver 106 via the communications medium 104. The communications medium 104 suitably comprises any type of medium, system and/or technique for transmitting signals. For example, the communications medium 104 suitably comprises a broadcast system, cable, wireless communications, fiber optics, cellular systems, microwave transmissions, or any other type of communication system.

The signal generated by the transmitter 102 is transmitted via the communications medium 104 to the receiver 106. The receiver 106 is preferably configured to extract the teletext and/or the closed caption data from the video signal. The receiver 106 suitably comprises a decoder and, if desired, various peripherals, such as a printer, display, and the like. The decoder suitably comprises software running on a processor, such as an INTEL PENTIUM microprocessor. The receiver 106 suitably includes a front end card to receive the television signal, demodulate it, digitize it, and store the results in memory.

In particular, referring now to Figure 9, a receiver 106 according to various aspects of the present invention suitably comprises: a sampling unit 404 for sampling the incoming signal; a memory 406; a processing unit 408, such as a microprocessor; and a set of output buffers 410A-B. The receiver 106 may optionally include a conventional video unit 402 for generating images on a display based on the video signal. The video unit 402 is suitably a conventional video system for receiving the incoming signal and generating an image on a display. It should be noted, however, that the video unit 402 is not a necessary component of a decoding system according to the present invention, and is included in the present embodiment solely for purposes of illustrating a possible configuration of a decoding system according to various aspects of the present invention. In many configurations, the video unit is entirely omitted. In a preferred embodiment, the receiver comprises a conventional, general purpose, non-dedicated computer. For the purposes of this specification, a “general purpose” or “non-dedicated” computer comprises any processor-based system that is not configured primarily to decode the signal from the transmitter 102, such as a standalone desktop or laptop
personal computer, a client-server network terminal, or a processor-based or “smart” television system.

The video signal is suitably provided to the sampling unit 404 via a conventional receiving apparatus, such as a broadcast receiver, a cable receiver, or a satellite receiver. The sampling unit 404 samples the video signal at periodic intervals and stores the samples in the main memory 406. As data is collected in the main memory 406, the processing unit 408 periodically retrieves data from the main memory 406 and processes some or all of the information. The processing unit 408 suitably comprises any component capable of performing the present functions, including a microprocessor, a digital signal processor, or a microcontroller. In the present embodiment, the processing unit 408 comprises a microprocessor. Specifically, the processing unit 408 may be configured to perform other functions, such as execution of conventional computer functions, as the sampling unit 404 stores video signal information in the main memory 406. The processing unit 408 suitably periodically reads the information from the main memory 406, processes the information, and returns to the execution of conventional computer functions.

The processing unit 408 may be programmed to extract the teletext and closed caption data from the incoming signal and provide the resulting data to the output buffers 410A-B. The data in the output buffers 410A-B may be used in any manner like any digital data signals, for example, to receive news, download files or programs, accumulate data, and the like. More particularly, the sampling unit 404 suitably collects data from the incoming signal, for example by sampling, converts the analog signal to a digital representation, and transfers the digital data to the main memory 406. In the present embodiment, the sampling unit 404 comprises a video grabber unit dedicated to acquiring raw data. In one embodiment, the video grabber is a component of another system as well, such as a television or computer video system. The video grabber samples the incoming signal and performs a transfer of the data to the main memory 406, for example at a rate of 24.5454 million samples per second during the VBI. The sampling unit 404, however, may suitably comprise any system for providing digital data to the processing unit 408 for analysis. The main memory 406 suitably comprises
any sort of memory, but preferably provides high speed access and storage, such as a conventional random access memory (RAM).

The processing unit 408 suitably comprises a conventional microprocessor for analyzing the digitized signal and extracting the closed caption and teletext data. For example, the microprocessor may be a PENTIUM® processor from INTEL® Corporation. The extracted data is written to the buffers 410A-B, which suitably comprise memory blocks. The size of the memory blocks may be configured, suitably dynamically, according to the number of lines containing data in the incoming signal. The data is now ready to be used by the application 412.

The processing unit 408 suitably performs multiple functions in the present system. For example, the processing unit 408 suitably comprises the central processing unit for a conventional computer system. In the present embodiment, the processing unit 408 also extracts the closed caption and teletext data from the video signal. Accordingly, the processing unit 408 includes two software modules to extract the relevant data from the signal.

It should be noted that the extraction process described is implemented in software and executed by the processing unit 408. Although discrete components are disclosed, the present embodiment implements the components as software modules executed by the processing unit 408. Nonetheless, alternative embodiments may include dedicated hardware components or a programmable device for performing various individual functions that are executed by the processing unit 408 in the present embodiment.

In particular, referring now to Figures 10 and 11, to extract the closed caption data, the processing unit 408 suitably includes a closed caption module 500 for identifying portions of the video signal including closed caption data and extracting the data. The closed caption module 500 suitably reads the digital representation of the signal from the main memory 406, recovers the clock synchronization signal, identifies the framing code, and extracts the data from the incoming signal.

In one embodiment, the closed caption module 500 initially executes a clock recovery process to identify the clock synchronization portion of the video signal (step 602). If the clock synchronization signal is not recovered, for example, if the clock synchronization signal
is not detected in the sampled data by the closed caption module 500, the analysis terminates (step 604). It should be noted that this automatic signal detection feature may be configured to be selectively disabled. If the clock synchronization signal is detected in the incoming signal (step 605), the closed caption module 500 uses the clock synchronization signal to extract the closed caption data from the signal via a bit slicing and shifting process (step 606).

The closed caption module 500 suitably monitors the extracted data to identify a synchronization pattern from the extracted bits (step 607). In particular, the closed caption module 500 of the present embodiment may analyze the binary data for the three-bit framing code preceding the data block. The closed caption module 500 suitably analyzes the binary data for a preselected period or number of bits. If the framing code is not detected within the preselected period, the analysis is discontinued. If the framing code is detected, the closed caption module 500 continues to perform the bit slicing process to extract the incoming closed caption data, for example the conventional 16 closed caption binary bits, from the incoming signal. The extracted information is then written to the output buffer 410A (step 609).

The closed caption data retrieval process is suitably implemented by a set of software modules executed by the processing unit 408. For example, referring now to Figure 10, the closed caption module 500 suitably includes a clock recovery module 502; a data extraction module 506; a framing code detection module 504; and a serial-to-parallel module 508. The clock recovery module 502 is suitably configured to determine whether the clock synchronization signal is present, determine phase characteristics of the incoming signal relative to the samples, and to provide information to the data extraction module 506 to synchronize the bit slicing function. In addition, the clock recovery module 502 suitably measures the amplitude of the incoming signal and determines the DC signal offset. Further, the clock recovery module 502 is suitably configured to suspend the analysis if the input signal does not include closed caption data.

In a preferred embodiment, the clock recovery module 502 reads samples of the digitized video signal from the main memory 406 and processes them according to a suitable algorithm to recover the clock synchronization signal. A conventional closed caption signal
has a bit rate of 0.5035 MHz. The sampling unit 404 suitably samples the incoming signal at a sufficiently high rate to track and recover the clock synchronization signal and the binary data in the video signal. In the present embodiment, the sampling unit 404 acquires samples at 24.5454 MHz. Thus, the sampling unit 404 acquires approximately 48.75 samples per bit of the closed caption data. Preferably, the sampling unit 404 acquires data at a rate at least four times higher than the data rate of the incoming binary signal. The clock synchronization signal operates at the same frequency as the bit rate, so the sampling unit 404 acquires 24.375 samples per half-cycle of the clock synchronization signal. Consequently, the clock recovery module 500 may analyze the closed caption signal to a phase error of less than eight degrees, which is sufficiently precise to effectively provide bit synchronization information to the data extraction module 506. If greater precision is required or desired, however, the sampling rate may be increased and/or the clock recovery module may provide enhanced signal analysis, such as through an interpolation analysis of the sampled data.

In the present embodiment, the clock recovery module 502 analyzes only a portion of the signal to ensure analysis of valid data. For example, in the present embodiment, the clock recovery module 502 suitably ignores the samples acquired during the first 9.4 microseconds after the falling edge of the horizontal synchronization pulse, to account for the minimum delay between the synchronization pulse to the rising edge of the first cycle of the clock synchronization signal.

In addition, because the initial cycles of the clock signal may not be stable, for example due to attenuation caused by a superimposed echo, the clock recovery module 502 may discard some of the initial clock synchronization signal cycles prior to recovering the clock synchronization signal, for example the first two cycles, and reserve analysis for the remaining cycles. The clock recovery module 502 then suitably analyzes the sample data to identify certain characteristics of the incoming signal to facilitate bit synchronization or bit decoding.

In particular, referring now to Figures 12A-B, the clock recovery module 502 suitably first seeks the rising edge of the first cycle (step 702), for example by searching for the first sample exceeding a threshold value (Figure 12B). The threshold is preferably selected to reduce the likelihood that the preceding color burst is not mistaken for the rising edge of the
clock synchronization signal, for example a value signifying 0.44 volt (based on a one volt peak-to-peak video signal). In addition, because the sampling unit 404 acquires over 48 samples per cycle of the closed caption clock synchronization signal, certain samples may be selected for analysis and the remainder ignored to improve the efficiency of the analysis. In the present embodiment, the clock recovery module 502 initially reads a first sample from main memory 406 following the initial 9.4 microsecond delay and advances to the next sample to be analyzed.

If the amplitude of the first sample does not exceed the selected threshold, the clock recovery module 502 advances to the next sample to be analyzed, in this example skips four samples, and again compares the value of the sample to the threshold. Thus, the present clock recovery module 502 analyzes only one of every four samples in this example for the clock synchronization signal. The process repeats until a sample value exceeds the threshold. Upon identifying a sample having a value exceeding the selected threshold, the clock recovery module 502 designates the sample as the rising edge of the cycle.

After detection of the rising edge, the clock recovery module 502 then seeks a maximum point in the clock synchronization cycle (step 704). For example, referring now to Figure 12C, the clock recovery module 502 reads the values of the current sample and a sample that preceded the current sample by a selected number of samples. Preferably, the number of preceding samples is selected to enhance the extracted data, for example to provide optimal gain. If the sampling unit takes signals at a much higher frequency than the data signal or clock signal frequency, for example about an order of magnitude higher, the number of preceding samples may be selected to correspond to a 180-degree phase shift. In the present embodiment, the sample 24 samples prior to the current sample is read, which corresponds to a 180-degree shift at a sampling rate of 24.375 samples per half-cycle of the clock synchronization signal. Again, four samples are skipped. If the value of the current sample is less than that of the older sample, then the maximum has not been identified. If the value of the current sample is greater than or equal to the value of the older sample, the current value is designated as the maximum. The clock recovery module 502 thus operates as a comb filter where the center of the first lobe of the frequency response corresponds to the closed caption
clock synchronization signal's frequency. This improves the selectivity of the clock for the peak detector.

Upon detection of the maximum, the clock recovery module 502 advances to the maximum of the next cycle, for example by skipping the next 48 samples (for a 24.5454 MHz sampling rate) following the detected maximum (step 705). The clock recovery module 502 then seeks the falling edge of the present cycle (step 706). This process suitably amounts to the converse of the process for identifying the rising edge of the clock synchronization signal. For example, referring now to Figure 12D, the clock recovery module 502 reads the current sample and skips four samples. If the amplitude of the read sample exceeds the selected threshold, the clock recovery module 502 continues searching for the falling edge at the next sample, and again skips four samples. This process suitably repeats until a sample value does not exceed the threshold. If the present sample is below the threshold, however, the falling edge, or at least an approximation of the falling edge, has been identified.

Upon identification of the falling edge, the clock recovery module 502 suitably seeks the minimum portion of the cycle (step 708). Referring now to Figure 12E, this process suitably comprises the converse of the process to find the maximum. The clock recovery module 502 reads the value of a current sample and a sample that preceded the current sample by 24 samples, and then skips four samples. If the value of the current sample is greater than that of the preceding sample, then the minimum has not been identified. The clock recovery module 502 consequently repeats the process. If the value of the current sample is less than or equal to the value of the older sample, the current value is designated as the minimum.

Thus, the clock recovery module 502 identifies the rising edge and the maximum of the first cycle to exceed the threshold and the falling edge and minimum of the second. At this point, the detection of the clock synchronization signal is considered verified. The first two cycles have been processed for initial estimates of the positions of the maximum and minimum and the rising and falling edges. In addition, the data is assumed to be stable after the first two cycles to exceed the threshold defining the rising edge. It should be noted that if the initial cycles of the clock synchronization signals are so attenuated as to never exceed the threshold, the attenuated cycles are ignored.
Referring again to Figure 12A, the clock recovery module 502 then suitably proceeds to establish the optimal clock position by tracking the maximum and minimum portions of the signal. For example, a counter is suitably set according to the number of cycles in the closed caption clock synchronization signal to be analyzed, such as four (step 710). Next, the next rising edge and the maximum of the next cycle are identified, for example as described above (steps 712 and 714).

As the rising edge and the maximum are identified, however, the clock recovery module 502 now suitably stores a value corresponding to the duration between the previous minimum and the most recently identified maximum. For example, with each repetition of the searches for the rising edge and the maximum, the position value is incremented by four. Upon detection of the maximum, the position value is compared to the ideal duration between the previous minimum and the current maximum, which in this case is 24+3/8 samples. The clock recovery module 502 then adds the results of the comparison to an adjustment value, which is ultimately used to determine the optimal clock position or phase position adjustment (step 716). In addition, the clock recovery module 502 suitably adds the value of each sample to a data average value stored, for example, in memory 406. The accumulated data average value may be ultimately used to determine an average sample value and a DC offset.

Upon identification of the maximum, the value of the sample at the maximum is stored. The difference between the value of the maximum and the preceding minimum is suitably compared to a selected threshold amplitude to ensure the analysis of valid data. If no value for a preceding minimum has been stored, then a default minimum, such as 0.28 volt (for a peak-to-peak video signal of 1.0 volt) may be provided. If the difference is less than the threshold amplitude, such as 0.14 volt, then an error message is returned and the data discarded. If the difference is greater than or equal to the threshold amplitude, however, the analysis continues.

The clock recovery module 502 then suitably proceeds to find the falling edge (step 718) and the minimum (step 720). Like the rising edge and maximum analyses, the falling edge and minimum analyses continue to update the data average value and the position value with each repetition. The clock recovery module 502 then suitably checks the amplitude of
the difference between the preceding maximum and the current minimum to ensure that it is at least equal to 0.14 volt. Similarly, the clock recovery module 502 then suitably computes the optimal position of the minimum and adds it to the optimal clock position adjustment value (step 722). The repetition counter is then decremented (step 723).

This process repeats four times, suitably once for each of the remaining clock synchronization cycles. Upon completion of the four repetitions, the clock recovery module 502 suitably checks the frequency of the tracked signal to ensure the detection of valid data. For example, the clock recovery module 502 suitably determines the duration between the sample at which the search for the first rising edge began (following the discarded initial cycles) and the sample at which the final minimum was detected. If the duration is out of range compared to that required for four cycles of the clock synchronization signal, then an error message is returned and the data is discarded.

If the duration is within an acceptable range, then the adjustment value is averaged, for example by dividing the adjustment value by eight, which corresponds to the number of adjustments added to the adjustment value. The averaged adjustment value corresponds to an approximate phase error from the maximum or minimum of the incoming signal. The averaged adjustment value may then be used to determine an optimal clock position (step 724), which may then be used to synchronize the extraction of the binary data bits from the byte synchronization and data block portions of the signal.

In addition, the clock recovery module 502 suitably determines the average data value for the samples analyzed. The data value average may be compared to an expected amplitude average, and the difference stored as a DC offset value. The DC offset value may be used by the data slicing process to establish the appropriate thresholds for determining logical values of the binary data and to remove DC offset prior to signal processing.

Referring again to Figures 10 and 11, if the clock recovery process successfully recovers the clock synchronization signal (step 605), the closed caption module 500 initiates the data extraction module 506 using the recovered clock synchronization signal information to synchronize the extraction of the binary data. For example, the data extraction module 506 suitably implements a bit slicing process which operates in conjunction with the clock
synchronization signal information to reconstruct the binary data embedded in the video signal (step 606). The data extraction module 506 suitably uses the data generated by the clock recovery module 502 to identify relevant samples and recover binary data in the signal. The bit slicing process is then applied to the identified relevant samples to assign appropriate binary values to the corresponding bits.

In the present embodiment, the data extraction module 506 selects particular samples for analysis based on the adjusted clock synchronization signal information. For example, the samples most likely to be closest to the rising and falling edges of the data bits may be identified based on the clock synchronization information. Similarly, the samples closest to the projected center of each bit signal, and hence those with the highest integrity, may be identified. Thus, the data extraction module 506 selects a suitable number of samples corresponding to each bit, for example one, for analysis in the bit slicing process.

The bit slicing process suitably assigns a binary value to each relevant sample. Alternatively, in systems using partial response coding, the bit slicing process assigns a value corresponding to one of multiple possible values. In the present embodiment, the data extraction module 506 compares each relevant sample of the incoming signal to a selected threshold, for example a binary data threshold corresponding to 0.52 volt (based on a peak-to-peak voltage of 1.0 volt) to establish a logical value for the signal. If the sample exceeds the threshold, it is designated as a logical “one”, and if not, the signal is designated as a logical “zero”. If more than one sample is analyzed for a particular bit, the data extraction module suitably assigns the binary value according to a suitable algorithm, for example a majority value with respect to the threshold. The data extraction module 506 then suitably provides the serial binary data to the framing code detection and serial-to-parallel modules.

As the bit slicing process executed by the data extraction module 506 proceeds, the framing code detection module 504 searches the resulting binary data for a valid framing code indicating that closed caption data is to follow (step 607). The framing code provides byte synchronization information to align the incoming bit stream on a byte boundary. It is also used to verify that a closed caption line has been identified. For conventional closed caption data, the framing code comprises the three bit code 1-0-0.
In the present embodiment, the framing code detection module 504 reads the bits in
the register as they are stored and compares them to the desired framing code. If the desired
code is received, the serial-to-parallel module 508 begins converting the following 16 serially
received bits into two bytes of parallel data (step 609 in Figure 6). If the framing code is not
detected, the framing code detection module 504 suitably waits for the next bit to be received
in the register, and repeats the analysis. The framing code detection module 504 suitably
repeats the analysis for a selected number of bits or time period. If the framing code is not
detected within the selected period or number of bits, the analysis process terminates.

When the framing code is detected following the clock synchronization signal, the
subsequent data, for example 16 bits of data, is provided to an 8-bit portion of the output
buffer 410A such that the first bit of the data signal is aligned with the first bit of the buffer
410A by the serial-to-parallel module 508. The result is a serial-to-parallel conversion of the
incoming closed caption binary data, which may then be provided to an application, a device,
or otherwise used.

Referring again to Figure 9, the processing unit 408 also suitably includes an analogous
system for teletext data extraction. In particular, referring to Figures 13 and 14, the processing
unit 408 suitably includes a teletext module 800 for identifying portions of the video signal
including teletext data and extracting the data. The teletext module 800 suitably reads the
signal from the main memory 406, recovers the clock synchronization signal from the
incoming signal, identifies the framing code in the byte synchronization signal, and extracts
the data. In addition, the teletext module 800 suitably performs several other signal processing
steps, such as data interpolation, equalization, echo cancellation, and error detection.

For example, a suitable teletext module 800 comprises: a clock recovery module 802;
a data interpolation module 804; an equalizer module 806; an echo cancellation module 808;
a framing code detection module 810; a data extraction module 812; a Hamming decoder
module 814; and a serial-to-parallel module 816. The clock recovery module 812 suitably
identifies the clock synchronization signal to provide bit synchronization for the teletext
module 800 (step 902). If the clock synchronization signal is not present, the clock recovery
module suspends further processing of the current line of the incoming signal (step 904). It
should be noted, however, that this automatic data detection function is suitably configured to be selectively disabled.

If the clock recovery module 802 identifies the clock synchronization signal, it provides clock synchronization signal information to the data interpolation module 804, which suitably refines the synchronization of the bit slicing function with the embedded binary data. The data interpolation module 804 suitably interpolates the data from the incoming samples to facilitate optimal synchronization of the sample analysis with the teletext data position and to convert the data bit rate to the original frequency (step 906). The equalizer module 806 and the echo cancellation module 808 further perform filtering and amplitude normalization to correct errors induced by echo signals (steps 908 and 910). The data extraction module 812 then suitably performs a bit slicing process on the amplitude normalized data sample (step 912).

Following the bit slicing, the framing code detection module 810 suitably surveys the incoming bit stream to identify the framing code in the signal (step 913). When (and if) the framing code is identified, the framing code detection module 810 generates byte synchronization information, which may be used by the serial-to-parallel module 816. The serial-to-parallel module 816 suitably accumulates data (step 914) as the data is extracted from the data extraction module 812 and aligns bits on the byte boundaries with the signal byte synchronization information from the framing code module 810. The Hamming decode module 814 suitably extracts the data information from the Hamming encoded byte (step 916). The Hamming decode module 814 and the serial-to-parallel module 816 store data bytes in the output buffer 410B (step 918).

The clock recovery module 802 and data interpolation module 804 are suitably configured to determine the optimal position for identifying the binary data embedded in the signal and synchronize the bit slicing function accordingly. In addition, the clock recovery module 802 may further measure information such as a DC signal offset and signal amplitude, as well as detect an invalid input signal and, if so, terminate further analysis of the signal.

For example, in a preferred embodiment, the clock recovery module 802 reads the digitized samples of the video signal from a buffer in the main memory 406 and processes the
samples according to a suitable algorithm to recover the clock synchronization signal. According to the NABTS specification, a teletext signal has a bit rate of 5.7272 MHz, and in the present embodiment, the sampling unit 404 acquires samples at 24.5454 MHz. Thus, the sampling unit acquires approximately 4.2857 samples per bit of the teletext data.

The samples acquired during the first 9.4 microseconds after the falling edge of the horizontal synchronization pulse are suitably ignored to account for the minimum delay between the synchronization pulse to the rising edge of the first cycle of the clock synchronization signal. Like the closed caption clock recovery module 502, the teletext recovery module 802 suitably analyzes only a portion of the signal to ensure analysis of valid data. In a similar manner, the first two cycles of the clock synchronization signal are suitably subjected to minimal processing and the remaining cycles are analyzed to generate the clock synchronization information.

Referring now to Figure 15A, the clock recovery module 802 first seeks the rising edge of the first cycle (step 1002), for example by searching for the first sample exceeding a threshold value (Figure 15B). The amplitude of the teletext clock synchronization signal is typically higher than that of the closed caption clock synchronization signal, so the threshold is accordingly higher, for example approximately 0.52 volt (based on a one volt peak-to-peak video signal). If the amplitude of the first sample does not exceed the selected threshold, the clock recovery module 802 advances to and reads the next sample in the buffer. Because the teletext clock synchronization signal is a much higher frequency than that of the closed caption clock synchronization signal, all of the samples in the teletext clock synchronization signal may be analyzed.

When the rising edge is detected, the clock recovery module 802 then seeks a maximum point in the clock synchronization cycle (step 1004). For example, referring now to Figure 15C, the clock recovery module 802 compares the value of a current sample and a preceding sample. If the value of the current sample is greater than or equal to that of the preceding sample, then the cycle value is still rising and the maximum has not yet been identified. The clock recovery module 802 advances to the next sample, and repeats the
process. If the value of the current sample is less than the value of the preceding sample, the preceding value is designated as the maximum.

Upon detection of the maximum, the clock recovery module 802 advances to look for the next maximum of the next cycle, for example by skipping the next eight samples following the first identified maximum (step 1005). The clock recovery module 802 then seeks the falling edge of the present cycle (step 1006). This process suitably amounts to the converse of the process for identifying the rising edge. For example, referring now to Figure 15D, the clock recovery module 802 reads the current sample and compares it to the selected threshold. If the amplitude of the sample exceeds the selected threshold, the data is ignored and the clock recovery module begins searching for the falling edge again with the next sample. If the present sample is below the threshold, however, the falling edge has been identified.

Upon identification of the falling edge, the clock recovery module 802 seeks the minimum portion of the cycle (step 1008). Referring now to Figure 15E, this process is suitably the converse of the process to find the maximum. The clock recovery module 802 reads the value of a current sample and a preceding sample. If the value of the current sample is less than or equal to that of the preceding sample, then the minimum has not been identified. The clock recovery module 802 advances to the next sample, and repeats the process. If the value of the current sample is greater than the value of the preceding sample, the preceding value is designated as the minimum.

Thus, the clock recovery module 802 has identified the maximum of the first cycle to exceed the threshold and the minimum of the second cycle to exceed the threshold. At this point, the assertion of the teletext clock synchronization signal has been verified. The first two cycles to exceed the rising edge threshold have been processed and the data is assumed to be stable. Referring again to Figure 15A, the clock recovery module 802 then attempts to establish the optimal clock position by tracking the maximum and minimum portions of the signal. For example, a counter is suitably set to four, which is equal to the number of cycles to be analyzed in the teletext clock synchronization signal (step 1010).

Next, the subsequent rising edge and the maximum of the next cycle are identified (steps 1012 and 1014). The clock recovery module 802 then computes the optimal position
of the maximum and adds it to the optimal clock position adjustment (step 1016). Further, the
clock recovery module suitably accumulates data to facilitate validation of the amplitude and
frequency of the signal and to compute the DC offset of the signal. Moreover, the clock
recovery module 802 collects data to facilitate the data interpolation process.

For example, the clock recovery module 802 suitably searches for the rising edge of
the present cycle as described above. After each sample is read and compared to the threshold,
the value of the sample is added to an accumulator to compute, at a later point in the analysis,
a data average value. The accumulated data average value is suitably ultimately used to
determine an average sample value and a DC offset. In addition, for each repetition of the
search for the rising edge, a position value is incremented to reflect the addition of one sample.

Upon detection of the rising edge, the clock recovery module 802 searches for the
maximum of the cycle. Like the search for the rising edge, after each sample is read and
compared to the preceding sample value, the value of the sample is added to a data average
value. Likewise, for each repetition of the search for the maximum, the position value is
incremented to reflect the addition of one sample. In addition, as each sample is compared to
the preceding sample, the value of the sample preceding the current sample by two samples
is stored. Thus, the value of the current sample and the two preceding samples are stored in
a buffer.

The position value is then compared to the ideal duration between the previous
minimum and the current maximum, which in this case is 4+2/7 samples. The clock recovery
module 802 then provides the values corresponding to the current sample and the preceding
two samples to the data interpolation module 804. The data interpolation module 806
calculates a fractional adjustment of the position to approximate the value of the cycle at its
center using a suitable algorithm. For example, in the present embodiment, if:

\[
\begin{align*}
\text{DATA} &= \text{value of the current sample} \\
\text{DATA}_A &= \text{value of the sample preceding DATA} \\
\text{DATA}_B &= \text{value of the sample preceding DATA}_A
\end{align*}
\]
then the fractional adjustment is calculated according to the following equation:

\[ \text{ADJ} = K \times \frac{(\text{DATA} - \text{DATA}_B)}{(2\times\text{DATA}_A - \text{DATA} - \text{DATA}_B)} \]

where K is a constant, suitably related to the sampling rate relative to the data rate. In the present embodiment, K is suitably set at 3.5. For efficiency, the calculation of the ADJ value may be performed using a lookup table.

This equation generates a fractional adjustment that is proportional to the phase offset between the actual center of the cycle and the sample nearest the center point. If the sample is very near the actual centerpoint, then \((\text{DATA} - \text{DATA}_B)\) is very small, generating a very small fractional adjustment. If the phase offset is large, the fractional adjustment value is large. The denominator comprises a normalizing factor to cancel variations in signal amplitude. The fractional adjustment value and the results of the comparison of the position value to the ideal position are then added to an adjustment value (step 1016).

Upon identification of the maximum, the difference between the value of the maximum and the preceding minimum is suitably compared to a threshold amplitude to ensure the analysis of valid data. If no value for a preceding minimum has been stored, then a default minimum, such as 0.28 volt (for a peak-to-peak video signal of 1.0 volt) may be provided. If the difference is less than the threshold amplitude, such as 0.14 volt, then an error message is returned and the data discarded. If the difference is greater than or equal to the threshold amplitude, however, the analysis continues.

The clock recovery module 802 then suitably proceeds to find the falling edge (step 1018) and the minimum (step 1020). Like the rising edge and maximum analyses, the falling edge and minimum analyses continue to update the data average value and the position value with each repetition. The clock recovery module 802 then suitably checks the amplitude of the difference between the preceding maximum and the current minimum to ensure that it is at least equal to 0.14 volt. Similarly, the clock recovery module 802 then suitably computes the optimal position of the minimum and adds it to the optimal clock position adjustment value (step 1022). The repetition counter is then decremented (step 1023).
This process repeats four times, once for each of the remaining clock synchronization cycles. Upon completion of the four repetitions, the clock recovery module 802 suitably checks the frequency of the tracked signal to ensure the detection of valid data. For example, the clock recovery module 802 suitably determines the duration between the sample at which the search for the first rising edge began (following the discarded initial cycles) and the sample at which the final minimum was detected. If the duration is out of range compared to that required for four cycles of the clock synchronization signal, then an error message is returned and the decoding process for the current video line terminates.

If the duration is within an acceptable range, then the adjustment value is averaged, for example by dividing the adjustment value by the number of values added to the accumulator, such as eight in the present example. The averaged adjustment value may then be used to determine an optimal clock position (step 724), which may then be used to synchronize the extraction of the binary data bits for the byte synchronization and data block portions of the signal. In addition, the clock recovery module 802 suitably determines the average data value for the samples analyzed. The data average may be compared to an ideal average, and the difference stored as a DC offset value. The DC offset value may be used by the data slicing process to establish the appropriate thresholds for determining logical values of the binary data and to remove DC offset prior to signal processing.

The data interpolation module 804 performs an interpolation function, such as described above, to identify the optimal clock position of the signal and more closely approximate the value of the signal at the exact center position of the data symbols. The value at the center position provides for maximum effectiveness of the equalizer module 806, echo cancellation module 808, and the data extraction module 812.

In addition, the data interpolation module 804 suitably performs a conversion of the sampling rate to the original data rate. The conversion process is suitably implemented using a digital interpolator filter, which is suitably a function performed by the processing unit 408. Referring now to Figure 16, the interpolator filter 1102 is suitably comprised of a sampling rate converter 1104, a low pass filter 1106, and a decimation filter 1108. The interpolator filter preferably uses zero padding to provide a higher rate that is an exact multiple of the data rate.
The sampling rate converter 1104 receives the input signal at the sampling rate (\textit{i.e.} 24.5454 MHz), and generates an output signal at 28.6363 MHz, which is five times the data rate. The output of the sampling rate converter 1104 is provided to the low pass filter 1106, suitably comprising a finite-impulse response (FIR) filter, which suitably corrects the amplitude of every sample. The FIR filter 1106 provides an output signal at 28.6363 MHz to the decimation filter 1108, which generates a corresponding signal at an output rate corresponding to that of the teletext data rate, \textit{i.e.} 5.7272 MHz.

The equalizer module 806 receives the interpolated data from the data interpolation module 804 and transfers processed data to the echo cancellation module 808. The equalizer module 806 compensates for amplitude and phase distortion caused by intersymbol interference. In the present embodiment, the equalizer module 806 is suitably configured as a digital adaptive equalizer, and is preferably adaptable to any amount of attenuation or DC offset of the incoming signal. Referring now to Figure 17A, in the present embodiment, the equalizer module 806 comprises a modified zero-forcing equalizer including a transversal filter (or a finite impulse response filter) 1204, a decision device 1206, a coefficients compute unit 1208, and a coefficients damping unit 1210. Further, the equalizer module 806 suitably includes various features, such as a variable number of taps associated with the transversal filter 1204 and an error clipping component 1212.

The transversal filter 1204 suitably comprises a delay line tapped at data rate (or symbol rate) intervals. Each tap along the delay line is connected through an amplifier to a summing device that provides the output. The transversal filter 1204 is suitably programmable in three different lengths or numbers of taps. When the incoming data signal is clean, the transversal filter 1204 is configured to operate using zero taps. Using zero taps provides the fastest decoding, which facilitates the reduction of the load on the CPU. For more indistinct signals, the transversal filter 1204 may be configured using a greater number of taps, for example five taps (Figure 17B). The greater number of taps provides better filtering but requires greater computing resources. If the incoming signal is very indistinct, a third mode configures the FIR filter with eleven taps (Figure 17C). Initially, the number of taps may be set according to a preselected default parameter. The adjustment of the number of taps is
suitably performed automatically based on the quality of the incoming signal. The quality of
the incoming signal may be assessed according to any suitable technique, for example,
according to the number of detected Hamming code errors, number of framing code errors,
and/or the magnitude of the tap coefficients.

The values of the samples at the taps are multiplied by coefficients stored in the main
memory 406. The amplification coefficients at the taps are controlled by the coefficient
compute unit 1208 to subtract the effects of interference from symbols that are adjacent in time
to the desired symbol. This process is repeated at regular intervals, for example, at every 15
symbols. The coefficients damping unit 1210 multiplies all coefficients by pre-determined
damping values at regular intervals, for example every 1/60 of a second. This allows the
equalizer module 806 to gradually reset the amplifier coefficients when no data signal is being
received or the teletext module is unable to detect the received teletext data. This is useful to
avoid divergence in the coefficients and helps to re-establish the proper coefficients when the
signal reception changes, for example in the event that a TV antenna is rotated.

The summed output of the transversal filter 1204 is sampled at the symbol rate and
then provided to the decision device 1206. The decision device 1206 computes the difference
between the projected amplitude of a symbol and the data sample output of the transversal
filter 1204. The resulting difference is used to calculate a new set of coefficients. The new
set of coefficients is stored in a table in the main memory 406 for use by the coefficient
compute unit 1208.

The equalizer module 806 may further include an error clipping component 1212
which compares the magnitude of the calculated error to a maximum threshold. If the
magnitude exceeds the maximum threshold, the amount of correction applied to the incoming
signal is reduced, for example limited to the value of the threshold. If the magnitude is below
the maximum threshold, however, the correction is applied without modification. In a similar
manner, the error clipping component 1212 suitably compares the error magnitude to a
minimum threshold. If the error magnitude is below the minimum threshold, no correction
is applied to the incoming signal. If the magnitude exceeds the minimum threshold, however,
the correction is applied without modification.
Alternatively, other types of adaptive filters can be used to perform similar functions, such as adaptive equalizers like an infinite impulse response (IIR), reverse adaptive equalizer, or any other suitable component. Further, in an alternative embodiment, instead of using a direct zero-forcing equalizer, a non-linear error table may be used to update the coefficients to avoid divergence when the transversal filter compensation is small.

The echo cancellation module 808 suitably receives the signal generated by the equalizer module and processes the signal to reduce the "ghost" effect caused by microreflection. In the present embodiment, the echo cancellation module 808 suitably comprises a module for implementing a method for ghost cancellation as described in U.S. Patent Application Serial No. 08/631,346, filed April 12, 1996. The echo cancellation module 808 suitably uses a last ten bits pattern in conjunction with a lookup table stored in the main memory 406 to provide the relevant correction. Preferably, the echo cancellation module 808 is called following the equalizer module 806, as the equalizer module 806 typically performs amplitude correction which facilitates the proper operation of the echo cancellation module 808.

It should be noted that many of the present components perform similar functions and may be used separately or in combination. The particular configuration may be adjusted according to the efficiency constraints of the system and processing resources. If the available processing resources are limited, the equalizer module 806, which typically consumes significant processing unit resources, especially when using a large number of taps, may be omitted or limited to a relatively low number of taps. The echo cancellation module 808, on the other hand, is typically relatively efficient, as it employs a lookup table to provide the signal processing function, but may not provide sufficient correction for certain applications and signals.

It should be further noted that the data interpolation module 804 performs a type of noise filtering function, and may be used without either of the equalizer module 806 or the echo cancellation module 808, with either individually, or with both. In the present embodiment, the equalizer module 806 uses data from the data interpolation module 804 to identify the optimal clock position; thus, if the data interpolation module 804 is omitted, the
equalizer module 806 may be similarly omitted or reconfigured. In addition, any suitable interpolation algorithm may be implemented by the data interpolation module 804 to generate the appropriate signal. Furthermore, other components may be used to perform the various functions of the data interpolation module 804, equalizer module 806, and/or echo cancellation module 808, including a matching filter, a linear interpolation filter, or other suitable filter.

The signal provided by the echo cancellation module 808 is suitably provided to the data extraction module 812, which slices the symbol data to extract the value of the symbol. The data provided by the data extraction module 812 is suitably provided to the serial-to-parallel module and the framing code detection module 810. The framing code detection module 810 searches for a valid framing code to provide for byte synchronization. The byte synchronization facilitates the alignment of the incoming bit stream on a byte boundary in the output buffer 410B. For conventional teletext data, the framing code comprises an eight bit code. For NABTS data, the framing code is 1-1-1-0-0-1-1-1 (E7h). Similarly, the framing code for WST is 1-1-1-0-0-1-0-0 (E4h). Two other framing codes are reserved for future use in NABTS as well: 1-0-0-0-0-1-0-0 (84h) and 0-0-1-0-1-1-0-1 (4Dh). When the appropriate code is detected by the framing code detection module, the data extraction module 812 transfers the following data to the serial-to-parallel module. The serial-to-parallel module converts the serial data stream into 8-bit (one byte) parallel data. If the data byte is Hamming encoded, the Hamming decode module decodes the data. The data is then provided to the output buffer 410B such that the first bit of the data signal is aligned with the first bit of the buffer 410B.

It should be further noted that the framing code detection module 810 suitably operates in conjunction with a lookup table of values which indicate receipt of a framing code. For example, the framing code detection module 810 suitably collects the incoming serial bits in an 8-bit buffer. Each set of bits collected in the buffer may be looked up in the lookup table to detect whether the framing code has been received. This lookup table may be also used to correct errors in the framing code byte. In the current lookup table, an algorithm to minimize the length error between bytes is used to generate correspondence between a decoded framing code byte versus a known acceptable framing code byte.
In particular, each time a new bit is shifted into the 8-bit register, the current eight bits are suitably looked up in the lookup table to detect a valid framing code. If the lookup table indicates that the current 8-bit value matches a framing code, such as one of the four conventional teletext framing codes described above, a result is returned indicating that the framing code has been received. If not, the framing code detection module exits and waits until a new bit is shifted into the register.

In some cases, it is possible that an error may occur and change the value of one or more bits. Certain errors, however, do not cause any synchronization problems because the system may be programmed to proceed with the serial-to-parallel conversion despite the error. For example, referring now to Figure 18, a one-bit error in any of the four teletext framing codes produces eight potential erroneous codes. Those that might be confused with another signal, such as a clock synchronization signal or the framing code of a different sort of teletext, are ignored. However, those codes having a one-bit error that do not bear similarity to other relevant codes, such as the clock synchronization signal or the framing code of a different sort of teletext, are suitably considered valid codes. If a line having an erroneous-but-acceptable code is processed, an error counter is suitably incremented to track the number of Hamming code errors. The list of codes corresponding to valid codes and erroneous-but-acceptable codes may be stored in a lookup table for efficient access. In the present embodiment, the framing code error detection function may be enabled or disabled, for example according to the desires of the user.

Before being written to the output buffer 410B, however, the data is subject to error checking, for example by the Hamming decode module 814, if the Hamming decode module 814 is enabled. In the present embodiment, the Hamming decode module 814 may be selectively enabled or disabled, or particular functions, such as error correction, may be selectively enabled or disabled. An NABTS packet includes five Hamming encoded bytes, whereas a WST packet includes two Hamming encoded bytes. The Hamming decode module 814 suitably operates in conjunction with a lookup table to detect errors. If a one-bit error occurs, the Hamming code decoder module 814 suitably corrects the error prior to writing the data to the output buffer 410B. If a two-bit error is detected, the Hamming decode module 814
analyzes the data according to a logical algorithm to decide whether to reject or retain the data packet. For example, if one of the first three Hamming bytes in a NABTS packet have two-bit errors, the packet is rejected. Alternatively, the Hamming decode module suitably counts the number of Hamming bits in error. If the total number of errors exceeds, for example, four errors, the packet is rejected. A Hamming decode scheme can detect two-bit errors and correct one-bit errors.

For NABTS, the first three bytes of the five Hamming encoded bytes represent a packet address value. Once they are decoded, the result is 12 bits to represent the packet address which is used to filter the service channel. Each of the 4096 service channels may be enabled or disabled by the software interface. The only information required is whether to accept or reject a specific packet based on its address value, which can be established a lookup table in the main memory 406.

Prior to operation of the receiver 106, the processing unit 408 and other relevant components of the receiver 106 are suitably initialized to prepare for operation. For example, various parameters such as a framing code table and a Hamming code table may be loaded into main memory 406. In addition, the various other internal operating parameters, such as a multiplication lookup table, the filter coefficients for the equalization module, and the echo cancellation module lookup table, may also be loaded. The receiver 106 then begins processing data to process each line of raw video data. To optimize efficiency, the decoder functions may not be called for each line of data; instead, the data may be stored in a buffer that contains, for example, 20 lines. When the buffer is full, the processing unit 408 may then initiate the decoding functions to decode all of the information in the buffer and store the decoded information in the output buffer. If some lines of data are rejected, the rejected lines may also be noted.

For example, referring now to Figure 19, the receiver 106 initially executes an initialization routine to perform various initialization functions, such as those described above (step 1310). A data processing module is then suitably initiated to process each line of raw video data that may contain VBI information. The receiver 106 commences collecting data and storing it in a buffer, for example in the main memory 406. When the buffer is full, an
input pointer is set at the beginning of the input buffer in the main memory 406 and an output pointer is set at the beginning of the output buffer, such as output buffer 410B (step 1312).

Because conventional teletext data is provided on lines 10 through 20 of the video signal and closed caption data is provided on line 21, the processing unit 408 initially determines whether the current line in the input buffer corresponds to line 21 of the video signal (step 1314). If not, the processing unit 408 calls the teletext module 800. The teletext module 800 searches for a valid clock synchronization signal (step 1316). If none is identified, the processing unit 408 proceeds to the next line in the buffer (step 1318). If the clock synchronization signal is detected, the line of data is decoded and provided to the output buffer 410B (step 1320). The processing unit 408 then determines whether the current line of data is the last line in the buffer. If so, the process terminates until the input buffer fills with data, and the processing unit 408 is released to perform other duties (step 1322). If not, the pointers are set to the next entries, and the analysis process is renewed.

If the current line of data in the buffer is line 21, the processing unit 408 calls the closed caption module 500. The closed caption module 500 searches for a valid closed caption clock synchronization signal (step 1324). If none is identified, the processing unit 408 proceeds to the next line in the input buffer (step 1326). If the clock synchronization signal is detected, the line of data is decoded and provided to the output buffer 410A (step 1328). The processing unit 408 then proceeds with determining whether the current line of data is the last line in the buffer and processing the data accordingly as described above.

It should be noted that the present decoding system may be programmed to decode any line with any type of data, including closed caption and/or teletext data. For example, if the decoder system is programmed to decode teletext data on line 21, the system attempts to decode the data as teletext before trying to decode the data as closed caption data. Similarly, if the system is programmed to decode closed caption on line 50, it attempts to decode the data as closed caption data on this line. In addition, the decoder of the present system may be adapted to support other television standards, for example PAL, SECAM, and other types of data transmission systems associated with a video signal.
In a preferred embodiment, the decoder may be programmed to collect information relating to the performance of the system, such as the quality of the incoming signal, whether lines of data are rejected, and processing time. The collected information may be processed to generate statistics, stored to be downloaded or processed later, or otherwise manipulated.

An alternative embodiment of the receiver 106 according to various aspects of the present invention is configured to receive and decode multi-level PRC data. The receiver 106 suitably performs the tasks associated with the first two layers of the OSI reference model. The data is then passed on to a computer application for further processing, including forward error correction.

The receiver 106 may also include a more sophisticated decoder system to relieve the computer of certain processing tasks or support other applications. For example, the receiver 106 may perform any required data processing, possibly up to the presentation layer of the OSI model. The decoder system may also operate in conjunction with any suitable interface, such as serial RS-232, parallel, Ethernet, or any other appropriate interface, depending on the application.

Referring now to Figure 20, the processing unit 408 also suitably includes a system analogous to the teletext data extraction system. In particular, the processing unit 408 suitably includes a buffer 2010 for receiving the data; a filter 2012; a data interpolation module 804; an equalizer module 806; a data extraction module 812; an error correction module 814; and a framing code detection module 810. The data interpolation module 804, the equalizer module 806, the data extraction module 812, and the framing code detection module 810 are suitably identical or analogous to and perform similar functions as the corresponding elements described above in conjunction with the binary closed caption and teletext data extraction.

Referring now to Figures 20 and 21, the analog combined television and data signal is provided to a VBI extractor 2110, such as a frame grabber, to temporarily store samples of the signal. The combined signal is also provided to a color clock phase lock loop 2112 to generate a clock signal synchronized with the color burst frequency. The color burst signal is then doubled by a clock doubler 2114, and the resulting signal is used to synchronize other
signal processing tasks, such as the operations of the equalizer module 806 and the data extraction module 812.

Signal samples from the vertical blanking interval are sampled and stored in the buffer 2010. The data samples are then filtered, suitably comprising a cosine-shaped or root square filter 2012, for filtering out-of-band noise from the incoming signal. Further, the samples may also be filtered by a pre-equalizing filter 2116 for improving the signal quality.

The data interpolation module 804, the equalizer module 806, the data extraction module 812, and the framing code detection module 810 process the incoming data samples to extract the digital data from the samples. Like the data extraction module 812 previously described, the present data extraction module 812 suitably performs a bit slicing process to assign particular values to each relevant sample. In mode 1, the data extraction module 812 assigns a three-level value to the sample, and in mode 2, it assigns a seven-level value to the sample. The data extraction module 812 then suitably provides the serial binary data to the framing code detection module 810. The symbols are then decoded and subjected to error detection and correction by the error correction module 814. Finally, the data is derandomized 2120 to counter the effect of the initial randomization at the primary transmitter 552.

The present embodiment suitably further includes forward error correction processing. For example, forward error correction may be based on Reed-Solomon codes (28, 26) for horizontal coding and (18, 16) for vertical coding. It defines an intermediate network layer, comprising 16 data packets and 2 check packets. Decoding this layer allows error correction such that any error in a single byte can be corrected, any pair of two-bit errors may be corrected, and one or two missing or uncorrectable packets can be replaced.

Thus, a receiver according to various aspects of the present invention provides a software-implemented system for decoding partial code response data transmitted with a video signal. The system is especially suited for time critical applications, such as in a computer environment. The system is also particularly suited to a system in which the sampling of the incoming signal is not synchronized to the incoming signal.

Because the processing unit is not required to process the incoming data continuously, processor resources may be dedicated to other tasks in the system. In addition, conventional
circuitry may be used to provide the decode functions. Further, because the present system is capable of detecting whether the incoming video line contains teletext or closed caption data, the system may more quickly relinquish the processing resources to other tasks if no data is included in the current video line.

Moreover, due to the software nature of the implementation, the system is easily and inexpensively upgraded and maintained. For example, the decoder system may be easily reconfigured to adapt to a particular environment by simply adjusting portions of the code and operating parameters. These changes may be provided in any manner. In addition, upgrades to the operation of the decoder system may be downloaded to the system using the decoder system itself. Further, any sort of data may be received via the decoder system, including computer files, applications, and the like.

In addition, the use of multi-level signal enhances the rate at which data may be transmitted. The availability of multiple modes having different bit transmission rates provides selectable robustness according to the signal and noise characteristics of the system.

While the principles of the invention have now been made clear in illustrative embodiments, there will be immediately obvious to those skilled in the art many modifications of structure, arrangements, proportions, the elements, materials and components, used in the practice of the invention which are particularly adapted for a specific environment and operating requirements without departing from those principles.
CLAIMS:

1. A computer-implemented decoding system for decoding a secondary signal embedded in a primary signal, comprising:
   a clock recovery module for identifying a clock synchronization signal embedded in the primary signal;
   a data extraction module responsive to the identified clock synchronization signal for extracting the secondary signal from the primary signal based on the identified clock synchronization signal, wherein the secondary signal comprises partial response coded data.
### FIG. 4A

<table>
<thead>
<tr>
<th>Level position</th>
<th>Level amplitude (in IRE units)</th>
<th>Logical value</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>70.0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>35.0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
</tbody>
</table>

### FIG. 4B

<table>
<thead>
<tr>
<th>Level position</th>
<th>Level amplitude (in IRE units)</th>
<th>Logical value</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>70.0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>58.3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>46.7</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>35.0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>23.3</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>11.7</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
</tbody>
</table>
FIG. 5
FIG. 7

FIG. 8

SUBSTITUTE SHEET (RULE 26)
FIG. 10
CLOCK RECOVERY 602

CLOCK OK? NO → EXIT 604

YES → BIT SLICING 606

FC FOUND? NO → LIMIT? 608

YES → TEST FRAMING CODE TEST ++

Write result 1 byte per 8 bits 609

BITCOUNT++

BITCOUNT>16

YES → EXIT

FIG. 11
CLOCK RECOVER

702 Find rising edge
704 Find Maximum
705 Skip data to next Maximum
706 Find falling edge
708 Find Minimum
710 Count=4

712 Find rising edge
714 Find Maximum
716 Calculate position & clock adjustment
718 Find falling edge yes
720 Find Minimum
722 Calculate position & clock adjustment
723 decrement Count
724 Count<>0
    no
Calculate optimal clock position

FIG. 12A

SUBSTITUTE SHEET (RULE 26)
FIG. 14

CLOCK RECOVERY

CLOCK OK?

INTERPOLATE

ABBE

ECHO CANCELER

BIT SLICING

FC FOUND?

TEST > LIMIT

TEST FRAMING CODE
TEST ++

SERIAL TO PARALLEL

BITCOUNT<40

HAMMING DECODE

Write result

BITCOUNT++

BITCOUNT>273

EXIT

Yes

No

Yes

No
FIG. 15A
FIG. 17A

OUTPUT SIGNAL

DECISION DEVICE
1206

ERROR CLIPPING
1212

COEFFICIENTS COMPUTE
1208

TRANSVERSAL FILTER
1204

MAIN MEMORY
405

COORDICIENT DAMPENING
1210

INCOMING SIGNAL
<table>
<thead>
<tr>
<th>Binary code</th>
<th>Hex</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 1 1 0 0 1 1 1 1</td>
<td>E7</td>
<td>FC NABTS</td>
</tr>
<tr>
<td>0 1 1 0 0 1 1 1 1</td>
<td>67</td>
<td>used</td>
</tr>
<tr>
<td>1 0 1 0 0 1 1 1 1</td>
<td>A7</td>
<td>Not used 1010 look like run-in clock</td>
</tr>
<tr>
<td>1 1 0 0 0 1 1 1 1</td>
<td>C7</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 0 0 1 1 1 1</td>
<td>F7</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 0 1 1 1 1 1</td>
<td>EF</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 0 0 0 1 1 1</td>
<td>E3</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 0 0 1 0 1 0</td>
<td>E5</td>
<td>Not used conflict with FC WST</td>
</tr>
<tr>
<td>1 1 1 0 0 1 1 1 0</td>
<td>E6</td>
<td>Not used conflict with FC WST</td>
</tr>
<tr>
<td>1 0 0 0 0 1 0 0 0</td>
<td>84</td>
<td>FC NABTS Alt 1</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0</td>
<td>04</td>
<td>used</td>
</tr>
<tr>
<td>1 1 0 0 0 1 0 0 0</td>
<td>C4</td>
<td>Not used conflict with FC WST</td>
</tr>
<tr>
<td>1 0 1 0 0 1 0 0 0</td>
<td>A4</td>
<td>Not used conflict with FC WST</td>
</tr>
<tr>
<td>1 0 0 1 0 1 0 0 0</td>
<td>94</td>
<td>used</td>
</tr>
<tr>
<td>1 0 0 0 1 1 0 0 0</td>
<td>8C</td>
<td>used</td>
</tr>
<tr>
<td>1 0 0 0 0 0 0 0 0</td>
<td>80</td>
<td>used</td>
</tr>
<tr>
<td>1 0 0 0 0 1 1 0 0</td>
<td>86</td>
<td>used</td>
</tr>
<tr>
<td>1 0 0 0 0 1 0 1 0</td>
<td>85</td>
<td>used</td>
</tr>
<tr>
<td>0 1 0 0 1 1 0 0 1</td>
<td>4D</td>
<td>FC NABTS Alt 2</td>
</tr>
<tr>
<td>1 1 0 0 1 1 0 0 1</td>
<td>CD</td>
<td>used</td>
</tr>
<tr>
<td>0 0 0 0 1 1 0 0 1</td>
<td>0D</td>
<td>used</td>
</tr>
<tr>
<td>0 1 1 0 1 1 0 0 1</td>
<td>60</td>
<td>used</td>
</tr>
<tr>
<td>0 1 0 0 1 1 0 0 1</td>
<td>50</td>
<td>Not used 0101 look like run-in clock</td>
</tr>
<tr>
<td>0 1 1 0 0 1 0 1 0</td>
<td>45</td>
<td>Not used 0101 look like run-in clock</td>
</tr>
<tr>
<td>0 1 0 0 1 0 0 1 0</td>
<td>49</td>
<td>used</td>
</tr>
<tr>
<td>0 1 0 0 1 1 1 0 0</td>
<td>4F</td>
<td>used</td>
</tr>
<tr>
<td>0 1 1 0 0 1 0 0 0</td>
<td>4C</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 0 0 1 0 0 0</td>
<td>E4</td>
<td>FC WST</td>
</tr>
<tr>
<td>0 1 1 0 0 1 0 0 0</td>
<td>64</td>
<td>used</td>
</tr>
<tr>
<td>1 0 1 0 0 1 0 0 0</td>
<td>A4</td>
<td>Not used conflict with FC NABTS</td>
</tr>
<tr>
<td>1 1 0 0 0 1 0 0 0</td>
<td>C4</td>
<td>Not used conflict with FC NABTS</td>
</tr>
<tr>
<td>1 1 1 1 0 1 0 0 0</td>
<td>F4</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 1 0 1 0 0 0</td>
<td>EC</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 1 0 0 0 0 0</td>
<td>E0</td>
<td>used</td>
</tr>
<tr>
<td>1 1 1 1 0 1 1 0 0</td>
<td>E6</td>
<td>Not used conflict with FC NABTS</td>
</tr>
<tr>
<td>1 1 1 1 0 1 0 1 0</td>
<td>E5</td>
<td>Not used conflict with FC NABTS</td>
</tr>
</tbody>
</table>

**FIG. 18**
FIG. 19

1310 Init param

1312 Set input pointer & output pointer

1314 Is line 21

1316 Recover NABTS clock

1318 Clock OK?

1320 Decode NABTS Line

1322 Last line in buffer

1324 Recover CC clock

1326 Clock OK?

1328 Decode CC Line

1328 No
FIG. 20
FIG. 21

<table>
<thead>
<tr>
<th>System specification</th>
<th>MODE1—PRC3</th>
<th>MODE2—PRC7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock rate</td>
<td>7.16 Mbits/sec</td>
<td>7.16 Mbits/sec</td>
</tr>
<tr>
<td>Transmitted bit rate</td>
<td>7.16 Mbits/sec</td>
<td>14.32 Mbits/sec</td>
</tr>
<tr>
<td>Net bit rate</td>
<td>21.6 Kbits/sec/line</td>
<td>43.2 Kbits/sec/line</td>
</tr>
<tr>
<td>Modulation</td>
<td>Binary partial response</td>
<td>4-any partial response</td>
</tr>
<tr>
<td>Frequency spectrum</td>
<td>Cos-shaped&lt;3.6MHz</td>
<td>Cos-shaped&lt;3.6MHz</td>
</tr>
<tr>
<td>Signal amplitude</td>
<td>70 IRE</td>
<td>70 IRE</td>
</tr>
<tr>
<td>Number of levels</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>Framing symbols</td>
<td>8 Bsync + 360 data</td>
<td>8 Bsync + 720 data</td>
</tr>
<tr>
<td>Application</td>
<td>Robustness</td>
<td>High speed</td>
</tr>
</tbody>
</table>

FIG. 22

SUBSTITUTE SHEET (RULE 26)
\[ T = \frac{1}{71.5909} \]

\[ A_r = 2 \cdot \cos \left( 2 \cdot \pi \cdot f \cdot \frac{T}{2} \right) \]

**FIG. 23**

\[ h_i = \frac{4}{\pi} \cdot \frac{\cos \left( \pi \cdot t_i \cdot \frac{T}{T} \right)}{\left[ 1 - 4 \cdot \left( \frac{t_i}{T} \right)^2 \right]} \]

**FIG. 24**

(symbol period = 8)
<table>
<thead>
<tr>
<th></th>
<th>Main</th>
<th>Delay/Pos</th>
<th>Reference</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timebase</td>
<td>100 ns/div</td>
<td>21.9950 us</td>
<td></td>
<td>Repetitive</td>
</tr>
<tr>
<td>Sensitivity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Channel 2</td>
<td>82.0 mV/div</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offset</td>
<td>335.688 mV</td>
<td>1.000 : 1</td>
<td></td>
<td>dc (1M ohm)</td>
</tr>
</tbody>
</table>

Trigger mode: Edge
On Negative Edge Of Chan2
Trigger Level
Chan2 = -150.000 mV (noise reject OFF)
Holdoff = 49.0000 us

**FIG. 25**
Eye Diagram of 7-level partial response signal
**INTERNATIONAL SEARCH REPORT**

**A. CLASSIFICATION OF SUBJECT MATTER**

IPC(6) : Please See Extra Sheet.
US CL : Please See Extra Sheet.
According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y,P</td>
<td>US 5,708,475 A (HAYASHI ET AL.) 13 January 1998, 19th and 24th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y,P</td>
<td>US 5,699,124, A (NUBER ET AL.) 16 December 1997, paragraph 62 of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 5,596,372 A (BERMAN ET AL.) 21 January 1997, 10th and 12th paragraph of background and summary; 44th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 5,521,712 A (OGURO) 28 May 1996, 23rd and 46th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 5,530,655 A (LOKHOFF ET AL.) 25 June 1996, 46th paragraph of detailed description</td>
<td>1</td>
</tr>
</tbody>
</table>

[X] Further documents are listed in the continuation of Box C.  

See patent family annex.

* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
*A* document defining the general state of the art which is not considered to be of particular relevance
'E' earlier document published on or after the international filing date
'L' document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
'O' document referring to an oral disclosure, use, exhibition or other means
'P' document published prior to the international filing date but later than the priority date claimed

Date of the actual completion of the international search  
05 OCTOBER 1998

Date of mailing of the international search report  
08 JAN 1999

Name and mailing address of the ISA/US  
Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231  
Facsimile No. (703) 305-3230  

Authorized officer  
PETER CHU  
Telephone No. (703) 305-3900

Form PCT/ISA/210 (second sheet)(July 1992)*
<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>US 5,519,780 A (WOO ET AL.) 21 May 1996, 5th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 5,486,865 A (JAMES) 23 January 1996, 10th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 4,742,543 A (FREDERIKSEN) 03 May 1988, 77th and 80th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 4,694,489 A (FREDERIKSEN) 15 September 1987, 78th and 81st paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 4,682,360 A (FREDERIKSEN) 21 July 1987, 77th and 80th paragraph of detailed description</td>
<td>1</td>
</tr>
<tr>
<td>Y</td>
<td>US 4,605,961 A (FREDERIKSEN) 12 August 1986, 78th and 81st paragraph of detailed description</td>
<td>1</td>
</tr>
</tbody>
</table>

Form PCT/ISA/210 (continuation of second sheet) (July 1992)*
A. CLASSIFICATION OF SUBJECT MATTER:
IPC (6):
H04N 7/087, 7/08, 5/76, 7/00, 7/025, 7/167; H04B 1/66; H04L 9/00

A. CLASSIFICATION OF SUBJECT MATTER:
US CL. :
348/468, 564, 465, 467, 478, 845.2, 537, 464, 536, 545, 405, 398, 1, 5.5; 386/95; 381/22, 23; 380/49, 10, 9, 19, 46, 20, 14, 15; 375/243;

B. FIELDS SEARCHED
Minimum documentation searched
Classification System: U.S.
348/468, 564, 465, 467, 478, 845.2, 537, 464, 536, 545, 405, 398, 1, 5.5; 386/95; 381/22, 23; 380/49, 10, 9, 19, 46, 20, 14, 15; 375/243;