VideoMan 2.0 Requirements
Basic Model

Draft 1.5
5/1/95

1.0 Scope

This document describes the requirements for the digital video camera to be developed and manufactured by Logitech.

2.0 Reference Documents

3.0 Product Requirements

3.1 Electrical Requirements

3.1.1 Sensors Supported

512 x 582 PAL color 1/3" CCDs from Sharp and Panasonic

512 x 492 NTSC color 1/3" CCDs from Sharp and Panasonic

752 x 582 PAL color 1/3" CCDs from Sharp and Sony

3.1.2 Frame Rate

Full-motion, selectable 12.5, 15, 25 or 30 frames per second

3.1.3 Shutter Speed

Selectable from 1/field rate to 1/10,000 sec

3.1.4 Gamma Correction

Selectable Gamma correction factor of 1.0 or 2.3. Others values may be loaded via I2C.

3.1.5 Automatic Gain Control (AGC)

AGC will automatically adjust camera to provide correct exposure/gain for ambient lighting. Small bright spots will not skew AGC performance. AGC may be switched off for manual operation via I2C commands.

3.1.6 Automatic White Balance

The automatic white balance will set the camera color balance to acceptable levels for incandescent, florescent, and daylight conditions upon initialization and without user input. Host may override AWB and color may be set manually via I2C commands.

3.1.7 Ambient Lighting Conditions

The camera must operate in normal office lighting including incandescent, florescent, and window daylight conditions. Camera must accommodate both 50 Hz and 60 Hz lighting.

3.1.8 Control/Snapshot Button

The camera will have a button that can be pressed by the user to initiate a still image capture or other functions. The button will cause a signal to be sent to the host indicating that it has been pressed.

3.1.9 Sleep Mode

The camera may be put into a sleep mode via I2C command from the host. In sleep mode, power consumption will be reduced, the clock will be off and there will be no data output. I2C control shall remain enabled and shall be used to wake the camera.

3.1.10 LED Indicator

The Camera shall have an LED indicator that can be switched by the host computer

3.1.11 Microphone

The camera will be designed to allow for an optional built-in microphone with the same performance requirements as the mic used in VM1.0. Provision will also be made for a snap-on external microphone.

3.1.15 Camera Electrical Interface

3.1.15.1 Video Output

The camera ASIC shall support the following interfaces which may be implemented on different camera versions

a) VM20 shall be backward compatible with VM1.0 DVCI

b) An expanded version of the Digital Video Computer Interface (DVCI-2) employed in VideoMan 1.0 (and similar to the current IndyCam video communication format) shall be provided. DVCI-2 digital video output will conform to DVCI except that it will be expanded to accommodate higher resolution images, and allowances will be made for alternate scanning frequencies to accommodate PAL and other format CCDs.

c) Standard PAL and NTSC outputs shall be supported

3.1.15.2 Electrical Connections

The DVCI-2 interface shall transmit video data via an 8-bit parallel data bus, provide a 2 wire I2C control interface, provide a 2 wire analog audio line, and transmit DC power from the host to the camera. A small, high density 36-Pin HDChamp connector shall be used.

An I2C line shall be provided in conjunction with analog versions. The camera will operate without an I2C input, but there will be no way to adjust the camera settings

3.1.15.3 Host Control

The camera shall be controlled by the host computer via I2C interface.

3.1.16 Power

The camera will require the following DC supply voltages from the host computer with DVCI.

12 vdc	TBD mA
-12 vdc	TBD mA
5vdc	TBD mA

Total power shall not exceed 5 watts.

Analog versions will require only 12vdc.

3.2 Optical Requirements

3.2.1 Standard Lens

Lens type: f1.4, 8mm with M11 or M12 screw mount and built-in IR filter. Focus shall be controlled from 3" to infinite by manual adjustment. The lens must be user replaceable.

3.2.2 Alternate Lens Options

Other lens options to be available for wide angle, close-up and stereo.

3.3 Mechanical and Industrial Design Features

3.3.1 Industrial Design Concept

VideoMan 2.0 and MooseCam will share the same industrial design. The ID will incorporate the mechanical requirements of section 3.3 and maintain a "family resemblance" to both the VideoMan and the relevant SGI product lines.

3.3.2 Privacy Cover

The camera will include sliding panel cover for the lens

3.3.3 Microphone Mount

MooseCam will have a mounting point for a snap-on external microphone.

DVCI-2

Digital Video Camera Interface

draft 7 - for review and comment - October 20, 1995
feedback to: Kwok (510) 713-4502 voice, (510) 796-8058 fax,
kwok_yan@logitech.com

Table of contents Page

1 Scope

This standard defines an interface for connection of a digital video source, such as a digital video camera, to a receiving device such as a personal computer system. It is based on SMPTE-125M and CCIR Recommendation 601. The standard is suitable for use in consumer electronics and computer peripherals, over distances of 3 m (10 ft). The characteristics of the interface are summarized in this section.

1.1 This interface incorporates: An 8-bit digital video channel from source to receiver, based on ANSI/SMPTE 125M; A monaural differential analog audio channel from source to receiver; A bidirectional serial bus controlled by the receiver; And limited DC power provided for the source's use by the receiver at +12V, -12V, and +5V.

1.1 The video signal is transmitted in the form of one luminance (Y) and two color-difference components (scaled version of R-Y and B-Y).

1.2 The video signal is transmitted at the 4:2:2 family level of CCIR 601. Provision is made to convey signals of 8-bit precision. The nominal luminance sampling frequency is listed in a table in section 1.9.

1.3 The bits of the digital video code words that describe the video signal are transmitted in a parallel arrangement using eight conductor pairs. Each pair carries a multiplexed stream of bits (of the same significance) of each of the component signals. A ninth conductor pair carries a clock signal, the clock frequency is listed in a table in section 1.9.

1.4 The signals on the interface are transmitted using single-ended conductor pairs for a distance up to 3 m (10 ft) without equalization.

1.5 The interface consists of one transmitter and one receiver in a point-to-point connection.

1.6 Parameters of the video signal format are chosen to facilitate conversion to and from a serial digital interface format.

1.7 The interface allows the transmission of appropriate ancillary signals that may be multiplexed into the data stream during video blanking intervals.

1.8 Where hexadecimal values are used, they are indicated by a subscript h, such as FFh; other values are decimal.

1.9 The following video data formats are supported. More details follows.

	Field Rate	**Sampling Frequency**	**Clock Frequency**
CCIR 656-NTSC	59.94 Hz	13.50 MHz	27.00 MHz
CCIR 656-PAL	50.00 Hz	13.50 MHz	27.00 MHz
Modified CCIR 656-PAL @ 30 Hz	29.97 Hz	13.50 MHz	27.00 MHz
Modified CCIR 656-PAL @ 25 Hz	25.00 Hz	13.50 MHz	27.00 MHz
768x576 Square Pixel @ 25 Hz	25.00 Hz	13.50 MHz	27.00 MHz
768x576 Square Pixel @ 30 Hz	29.97 Hz	13.50 MHz	27.00 MHz
640x480 Square Pixel @ 60 Hz	59.94 Hz	13.50 MHz	27.00 MHz

2 General Considerations

2.1 Digital signal convention

The signaling sense of the voltage appearing across the digital signal conductors follows standard TTL levels as follows (refer to figure):

2.1.1 The A terminal of the transmitter shall be < 0.4V with respect to the G terminal for a binary 0 (LOW or L or OFF) state.

2.1.2 The A terminal of the transmitter shall be > 2.4V with respect to the G terminal for a binary 1 (HIGH or H or ON) state.

2.1.3 The signals of the serial (I^2C) bus are TTL levels in accordance with 2.1.1 and 2.1.2 above.

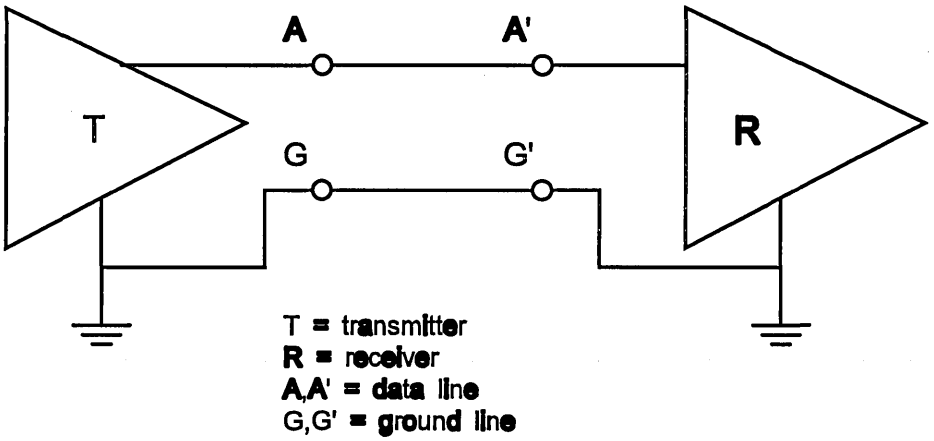


Figure 1 - Digital video signal convention

2.2 Signal names

The video data lines are designated DATA 0 through DATA 7. The group of eight signals is identified by placing parentheses around the range of subscripts included, as DATA (0-7). DATA 7 is always the most significant bit. The video clock is designated DCLK.

The signals of the serial (I^2C) bus are designated I2CD for data, I2CCLK for clock.

2.3 SinX/X considerations

The characteristics of the data word at the interface are based on the assumption that the location of any required sinX/X correction is at the point where the digital signal is converted to an analog format.

2.4 Blanking interval considerations

This standard does not require the device feeding the interface to transmit video data during the entire blanking interval. Therefore, ancillary information [see 3.6, 3.7] may be inserted into the horizontal blanking interval by the source within the constraints specified in 3.4 and 3.5.

The vertical blanking duration is a minimum of nine lines. Ancillary information may be inserted into this nine-line interval by the source within the constraints specified in 3.4 and 3.5.

2.5 Signal specifications

All digital signal time intervals are specified at the half-amplitude points. All transitions are specified between the 20% and 80% amplitude points.

2.6 Electromagnetic interference considerations

Digital apparatus can radiate a significant amount of energy at harmonics of the clock frequency. In the case of 13.5 MHz clock harmonics lie at 121.5MHz and 243 MHz, both of which are aeronautical distress frequencies. Equipment and system designers must, therefore, pay particular attention to the provision of adequate screening.

3 Digital video interface format

3.1 General description

The interface consists of an unidirectional nine-pair interconnection between a transmitting equipment and a receiving equipment. Video data, timing reference information, and ancillary signals are time multiplexed and transferred on eight data pairs in NRZ form. A ninth pair provides a synchronous clock.

3.2 Encoding parameters

Table 1a to 1f summarizes the encoding parameter values.

3.3 Interface characteristics

Table 2 specifies the interface characteristics.

Table 1a - Video encoding parameters

Coded signals: These values are obtained from the gamma precorrected signals.	Y = 0.299R + 0.587G + 0.114B C_R = 0.713 (R-Y) = 0.500R - 0.419G - 0.081B C_B = 0.564 (B-Y) = 0.500B - 0.169R - 0.331G
Form of encoding:	Uniformly quantized, PCM, 8 bits per sample, for the luminance signal and each color-difference signal.
levels and quantization levels: - luminance signal (Y) - each color difference signal (C_R, C_B)	220 quantization levels with the black level corresponding to level 16 and the peak white level corresponding to level 235 225 quantization levels symetrically distributed about level 128, corresponding to the zero signal
Sampling structure:	Orthogonal: line, field, and frame repetitive; C_R and C_B samples are cosited with odd (1st, 3rd, 5th) Y samples in each line
Sampling frequency: - luminance (Y) - each color-difference signal (C_R, C_B)	 13.50 MHz nominal 6.75 MHz nominal

Table 1b - Video encoding parameters, CCIR656 NTSC (60 Hz) and Modified 30Hz CCIR656-PAL

Number of samples per line:	Total	Active
- luminance (Y)	858	720
- each color-difference signal (C_R, C_B)	429	360
- total number of samples	1716	1440

Table 1c - Video encoding parameters, CCIR656 PAL (50 Hz) and Modified 25Hz CCIR656-PAL

Number of samples per line:	Total	Active
- luminance (Y)	864	720
- each color-difference signal (C_R, C_B)	432	360
- total number of samples	1728	1440

Table 1d - Video encoding parameters, 768x576 Square Pixel @ 25 Hz

Number of samples per line:	Total	Active
- luminance (Y)	1000	768
- each color-difference signal (C_R, C_B)	500	384
- total number of samples	2000	1536

Table 1e - Video encoding parameters, 768x576 Square Pixel @ 30 Hz

Number of samples per line:	Total	Active
- luminance (Y)	990	768
- each color-difference signal (C_R, C_B)	495	384
- total number of samples	1980	1536

Table 1f - Video encoding parameters, 640x480-Square Pixel

Number of samples per line:	Total	Active
- luminance (Y)	858	640
- each color-difference signal (C_R, C_B)	429	320
- total number of samples	1716	1280

Table 2 - Digital video interface characteristics

Digital format	Parallel: nine single-ended signal pairs carrying clock and eight data bits
interface clock	27.00 MHz nominal
Voltage levels	Standard TTL
Driver impedance	nominal 50 ohms
Receiver impedance	110 ohms nominal, to ground

3.4 Digital blanking relationship

3.4.1 Horizontal sync relationship

3.4.1.1 Horizontal sync relationship : CCIR656-NTSC and Modified 30Hz CCIR656-PAL Formats

Figure 2a shows the relationship between video signals in the digital and analog domain for 525-line systems. Figure 2b shows the multiplex structure.

Transmitted during each active line are 1440 multiplexed luminance and chrominance values (720 luminance, 360 chrominance CR, and 360 chrominance C_B values).

Eight of the remaining 276 interface clock intervals are used to transmit synchronizing information; the other 268 interface clock intervals may be used to carry ancillary information, including digital audio.

The first of these 1716 interface clock intervals is designated line word 0 for the purpose of reference only. The 1716 sample words per total line are therefore numbered 0 through 1715. Intervals 0 through 1439, inclusive, contain video data. The interface clock intervals occurring during digital blanking are designated 1440 through 1715.

Intervals 1440 through 1443 are reserved for the end-of-active-video (EAV) timing reference described in 3.5.3. Intervals 1712 through 1715 are reserved for the start-of-active-video (SAV) timing reference described in 3.5.3.

The half-amplitude point of the leading (falling) edge of the analog horizontal sync signal shall be coincident with a sample point which would be conveyed by word 1473 if carried across the interface.

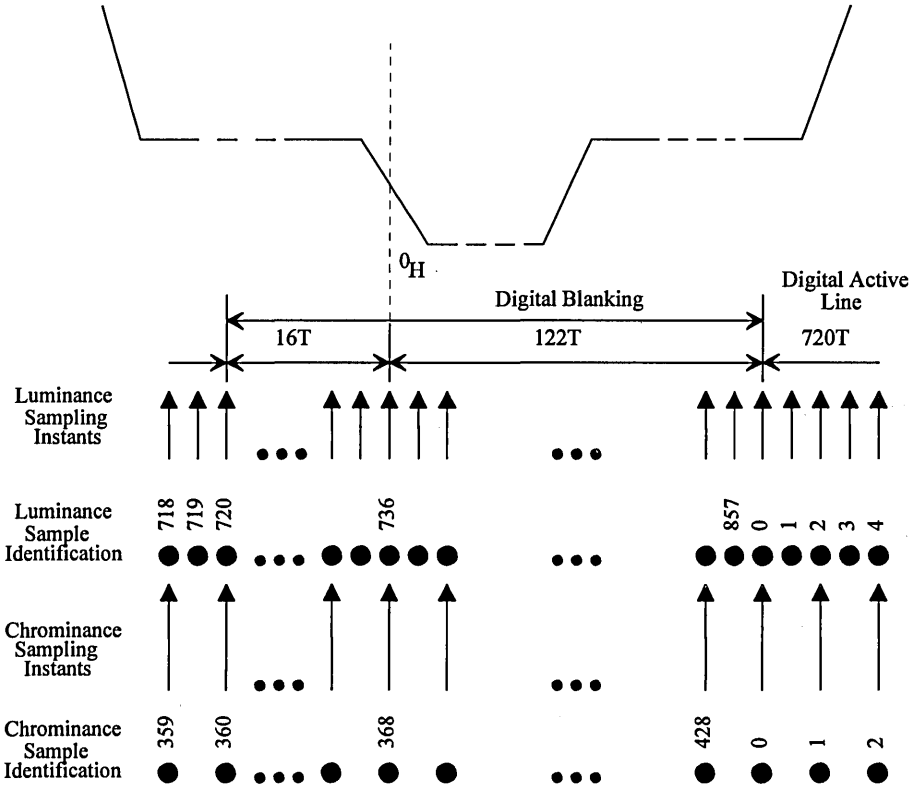


Figure 2a - Horizontal sync relationship, CCIR656-NTSC (60Hz) and Modified 30Hz PAL

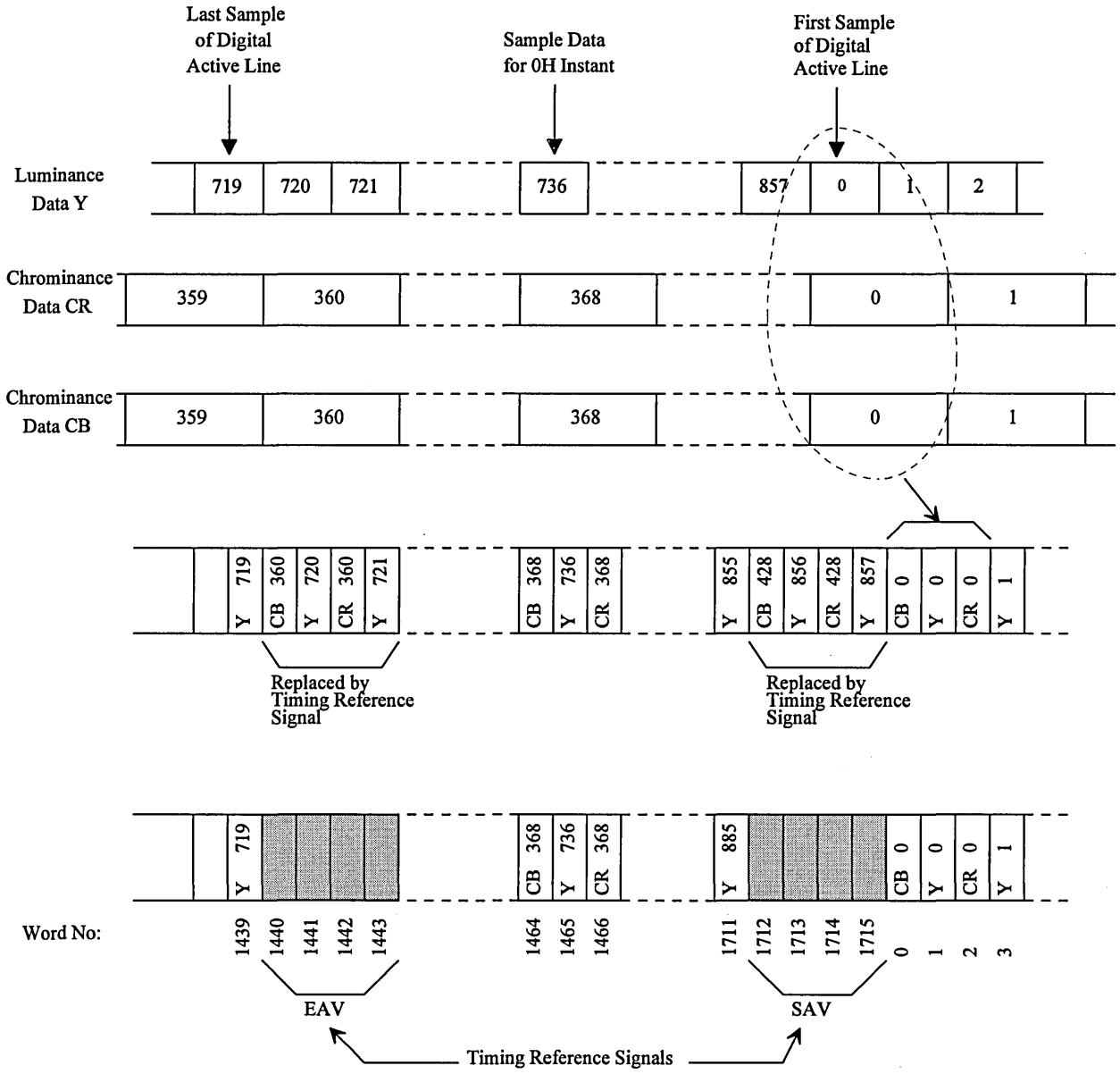


Figure 2b - Multiplex structure, CCIR656-NTSC (60Hz) and Modified 30Hz PAL

3.4.1.2 Horizontal sync relationship - CCIR656-PAL (50Hz) and Modified PAL (25Hz) Formats

Figure 2c shows the relationship between video signals in the digital and analog domain for 625-line systems. Figure 2d shows the multiplex structure.

Transmitted during each active line are 1440 multiplexed luminance and chrominance values (720 luminance, 360 chrominance CR, and 360 chrominance C_B values).

Eight of the remaining 288 interface clock intervals are used to transmit synchronizing information; the other 280 interface clock intervals may be used to carry ancillary information, including digital audio.

The first of these 1728 interface clock intervals is designated line word 0 for the purpose of reference only. The 1728 sample words per total line are therefore numbered 0 through 1727. Intervals 0 through 1439, inclusive, contain video data. The interface clock intervals occurring during digital blanking are designated 1440 through 1727.

Intervals 1440 through 1443 are reserved for the end-of-active-video (EAV) timing reference described in 3.5.3. Intervals 1724 through 1727 are reserved for the start-of-active-video (SAV) timing reference described in 3.5.3.

The half-amplitude point of the leading (falling) edge of the analog horizontal sync signal shall be coincident with a sample point which would be conveyed by word 1465 if carried across the interface.

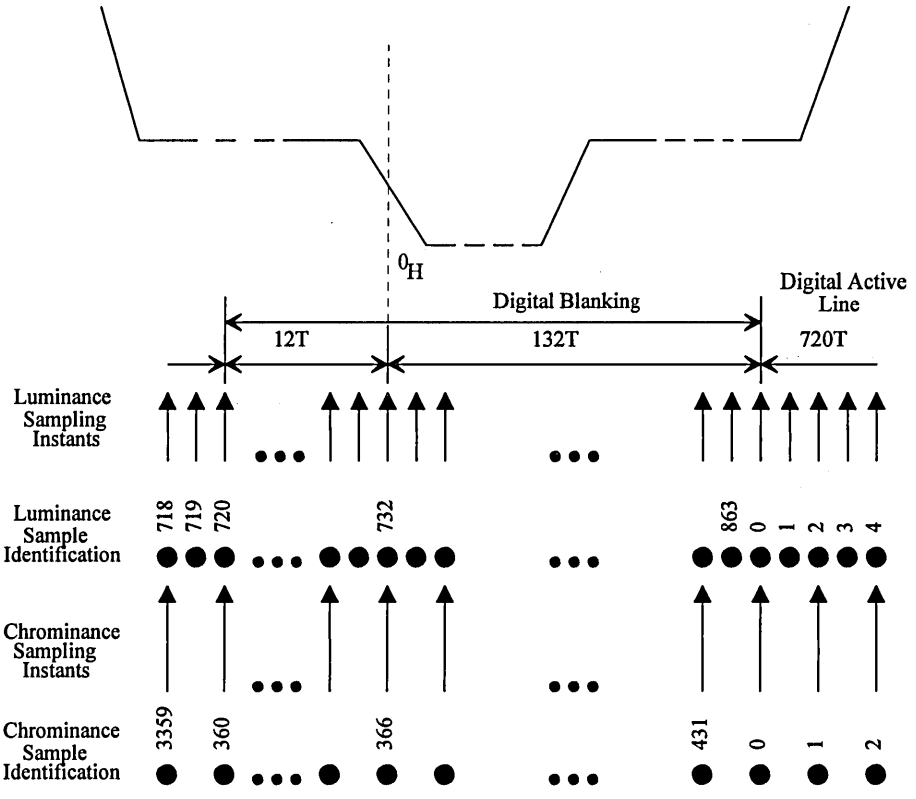


Figure 2c - Horizontal sync relationship, CCIR656-PAL (50Hz) and Modified 25 Hz CCIR656-PAL

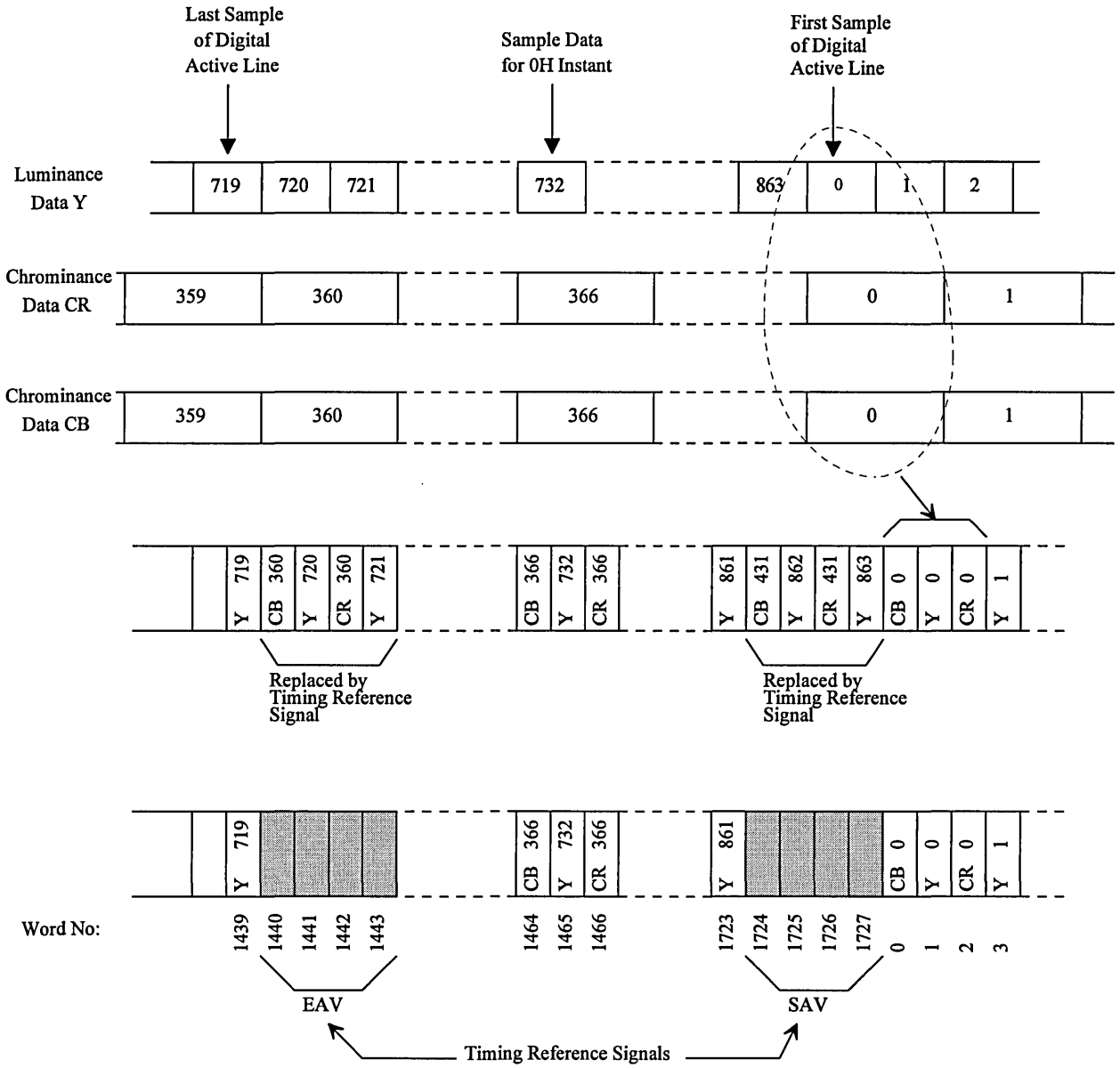


Figure 2d - Multiplex structure, CCIR656-PAL (50Hz) and Modified 25Hz CCIR656-PAL

3.4.1.3 Horizontal sync relationship : 768x576 Square Pixel @ 25Hz

Transmitted during each active line are 1536 multiplexed luminance and chrominance values (768 luminance, 384 chrominance CR, and 384 chrominance C_B values).

Eight of the remaining 464 interface clock intervals are used to transmit synchronizing information; the other 456 interface clock intervals may be used to carry ancillary information, including digital audio.

The first of these 2000 interface clock intervals is designated line word 0 for the purpose of reference only. The 2000 sample words per total line are therefore numbered 0 through 1999. Intervals 0 through 1535, inclusive, contain video data. The interface clock intervals occurring during digital blanking are designated 1536 through 1999.

Intervals 1536 through 1539 are reserved for the end-of-active-video (EAV) timing reference described in 3.5.3. Intervals 1996 through 1999 are reserved for the start-of-active-video (SAV) timing reference described in 3.5.3.

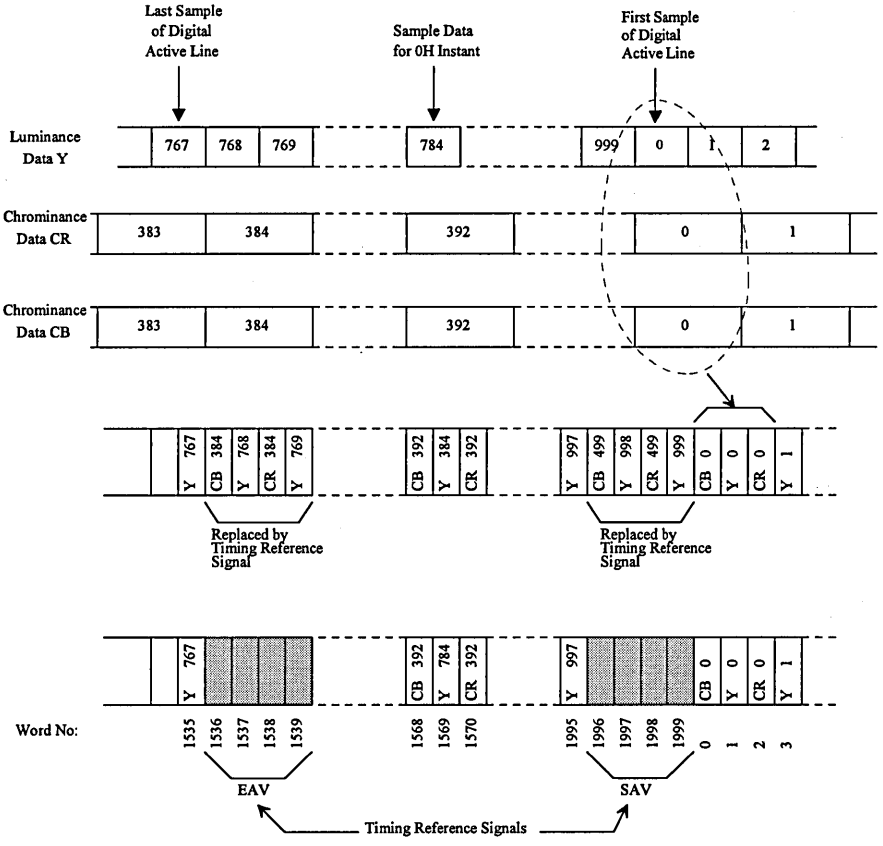


Figure 2e - Multiplex structure, 768x576 Square Pixel

3.4.1.4 Horizontal sync relationship : 768x576 Square Pixel @ 30Hz

Transmitted during each active line are 1536 multiplexed luminance and chrominance values (768 luminance, 384 chrominance CR, and 384 chrominance C_B values).

Eight of the remaining 444 interface clock intervals are used to transmit synchronizing information; the other 436 interface clock intervals may be used to carry ancillary information, including digital audio.

The first of these 1980 interface clock intervals is designated line word 0 for the purpose of reference only. The 1980 sample words per total line are therefore numbered 0 through 1999. Intervals 0 through 1535, inclusive, contain video data. The interface clock intervals occurring during digital blanking are designated 1536 through 1979.

Intervals 1536 through 1539 are reserved for the end-of-active-video (EAV) timing reference described in 3.5.3. Intervals 1976 through 1979 are reserved for the start-of-active-video (SAV) timing reference described in 3.5.3.

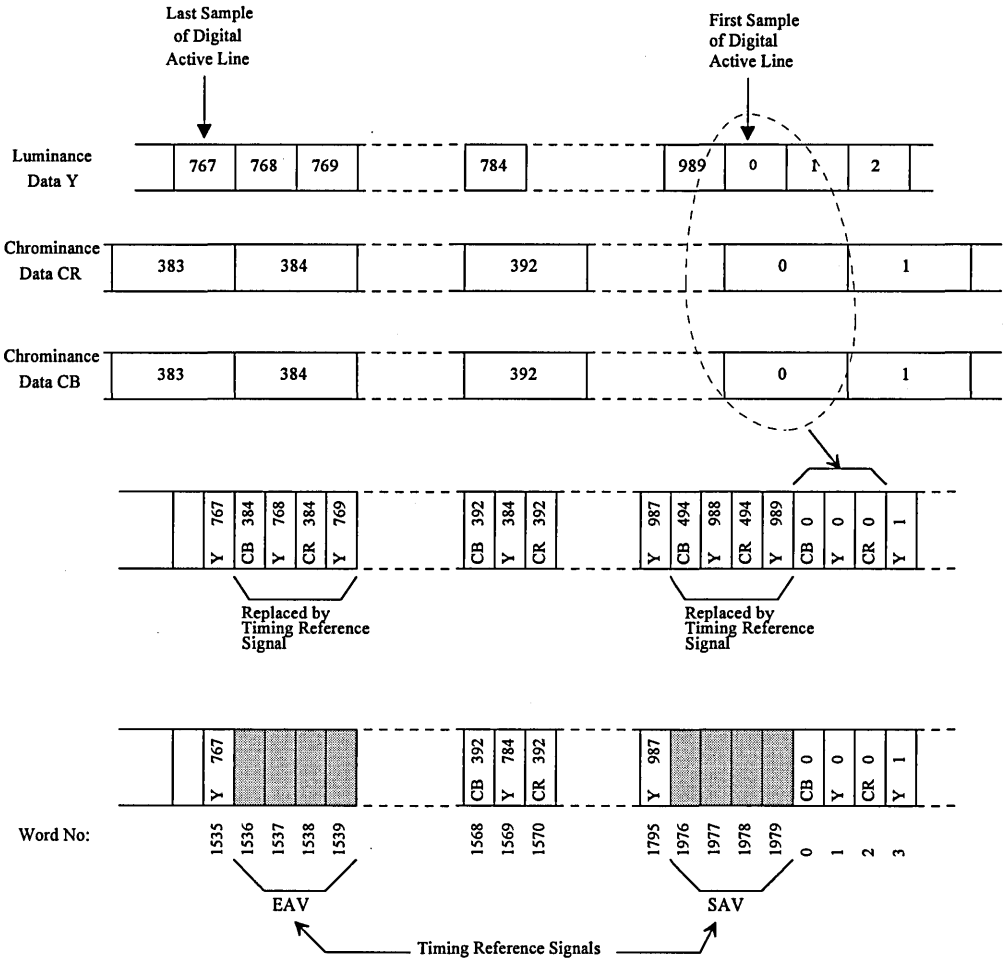


Figure 2f - Multiplex structure, 768x576 Square Pixel

3.4.1.5 Horizontal sync relationship : 640x480 Square Pixel

Transmitted during each active line are 1280 multiplexed luminance and chrominance values (640 luminance, 320 chrominance CR, and 320 chrominance C_B values).

Eight of the remaining 436 interface clock intervals are used to transmit synchronizing information; the other 428 interface clock intervals may be used to carry ancillary information, including digital audio.

The first of these 1716 interface clock intervals is designated line word 0 for the purpose of reference only. The 1716 sample words per total line are therefore numbered 0 through 1715. Intervals 0 through 1279, inclusive, contain video data. The interface clock intervals occurring during digital blanking are designated 1280 through 1715.

Intervals 1280 through 1283 are reserved for the end-of-active-video (EAV) timing reference described in 3.5.3. Intervals 1712 through 1715 are reserved for the start-of-active-video (SAV) timing reference described in 3.5.3.

The half-amplitude point of the leading (falling) edge of the analog horizontal sync signal shall be coincident with a sample point which would be conveyed by word 1473 if carried across the interface.

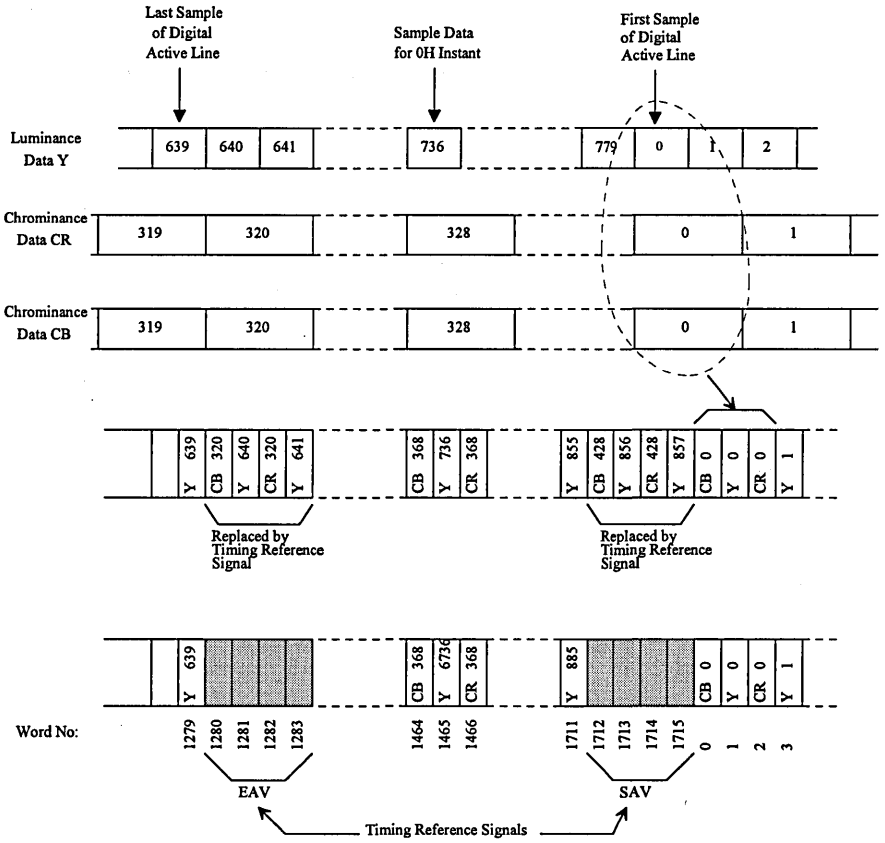


Figure 2g - Multiplex structure : 640x480 Square Pixel

3.4.2 Vertical sync relationship

3.4.2.1 Vertical sync relationship : 525-line systems

Figure 3a shows the relationship between video signals in the digital and analog domains for 525-line systems.

NTSC vertical signals such as VITC, close captioning, and so on, are permitted in lines 1-19 and 264-281, but image data is not permitted in these lines.

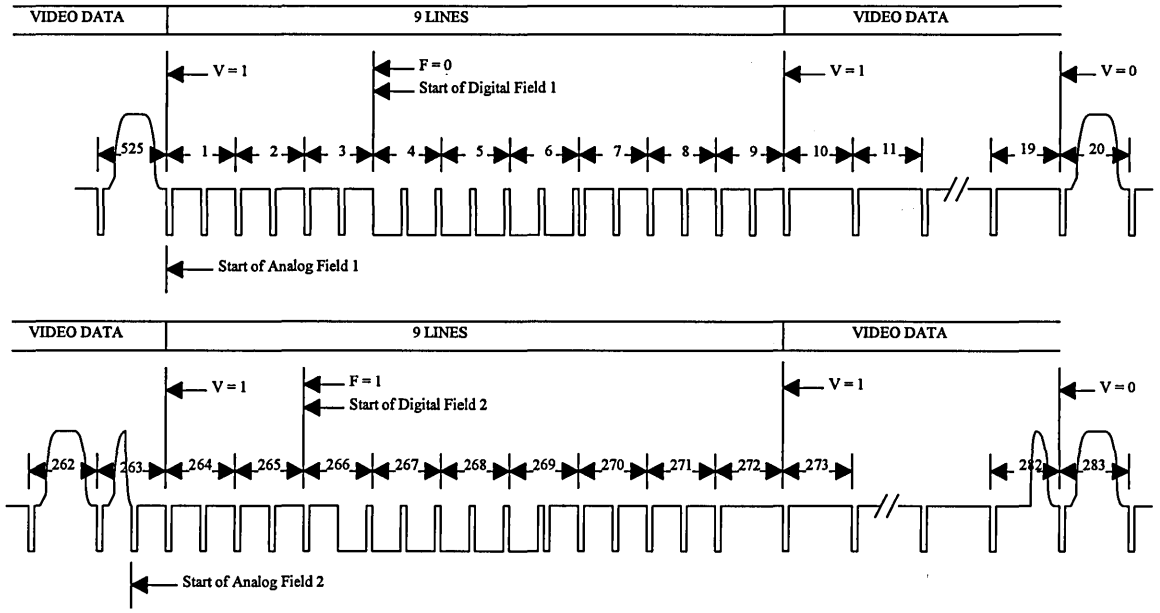


Figure 3a - Relationship of video data and vertical sync, 525-line systems Used by CCIR656-NTSC and 640x480 Square Pixel

3.4.2.2 Vertical sync relationship - 625-line Systems

Figure 3b shows the relationship between video signals in the digital and analog domains for 625-line systems.

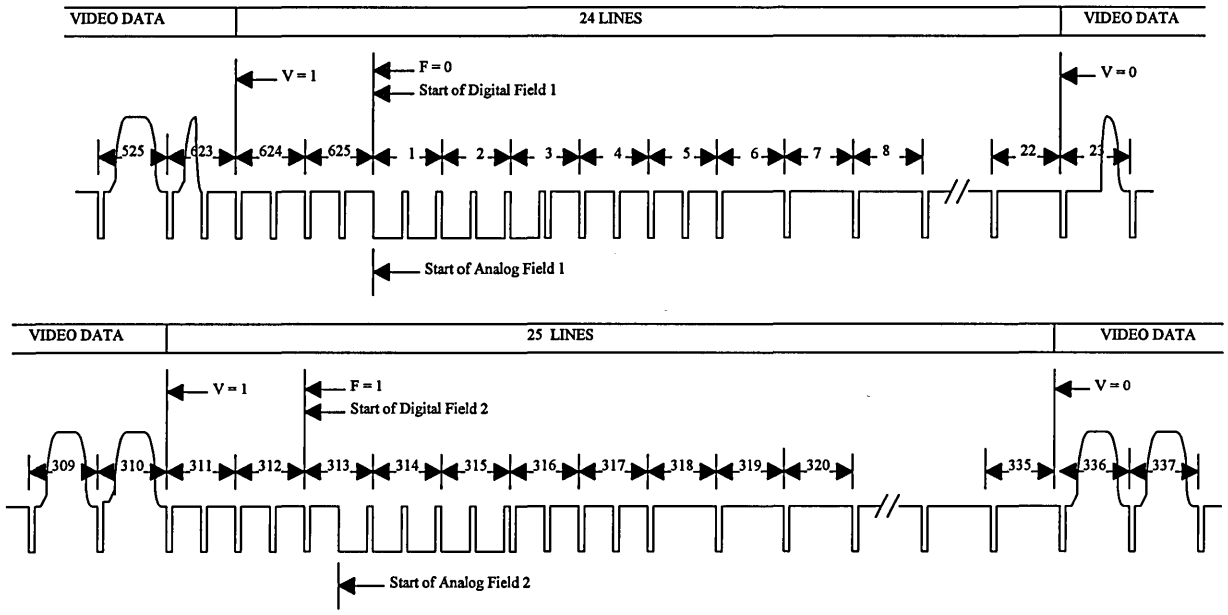


Figure 3b - Relationship of video data and vertical sync, 625-line Systems Used by CCIR656-PAL and modified CCIR656-PAL format

3.5 Video data signal format

3.5.1 Data signal format

Data is transmitted across the interface on eight data pairs: DATA 0 through DATA 7. DATA 7 is the most significant bit (MSB). 254 levels of the 8-bit word (levels 1 through 254 or 01h through FEh) are used to express quantized values.

Data levels 0 and 255 (00h and FFh) are reserved to indicate timing references.

3.5.2 Multiplex structure

The video data words shall be conveyed as multiplex in the following order:

C_B Y C_R [Y] C_B ...

where the three words C_B Y C_R refer to cosited samples, the following word [Y] being an isolated luminance only sample. The C_B and C_R samples are cosited with the first and subsequent alternate Y samples (0, 2, 4 ...) on each line. (See figure 2B.) The first video data word in each active line period shall be C_B.

3.5.3 Timing reference signals - Video

Figure 2a shows the position of the timing reference signals with respect to horizontal blanking in the multiplexed data stream. It is implicit that the timing reference signals are contiguous with the video data, when present, and continue through the vertical blanking interval.

Each timing reference signal consists of a four-word sequence in the following format:

FFh 00h 00h xy

The first three words are a fixed preamble. The fourth word shall contain information defining:

- even field (field 2) identification
- state of vertical blanking
- state of horizontal blanking

Assignment of bits within the fourth word is shown in table 3.

P0, P1, P2 and P3 have states dependent on states of bits F, V, and H according to table 4.

Vertical blanking in the digital interface is in full-line increments.

EAV and SAV are the digital horizontal synchronization signals and occur on every line.

The interval starting at EAV and ending with SAV is the digital horizontal blanking period as shown in figure 2b.

Table 3 - Timing reference signals

Bit	TRS Word #1	TRS Word #2	TRS Word #3	TRS Word #4	
7	1	0	0	1	fixed
6	1	0	0	F	F = 0 during field 1 F = 1 during field 2
5	1	0	0	V	V = 0 during active video V = 1 during vertical blanking
4	1	0	0	H	H = 0 for SAV H = 1 for EAV
3	1	0	0	P3	- see table 4
2	1	0	0	P2	
1	1	0	0	P1	
0	1	0	0	P0	

Notes

1 This is simply the upper 8 bits of the 10-bit SMPTE 125M-1992 Table 3.
2 The H, V, and F bits provide all the necessary state information. Bits 0-3 provide error detection and correction information.
3 Each 525-line digital video frame is divided into two fields. Field 1 contains 262 complete horizontal lines; field 2 contains 263 complete horizontal lines.
4 Each 625-line digital video frame is divided into two fields. Field 1 contains 312 completed horizontal lines, field 2 contains 313 completed horizontal lines.
5 The protection bits allow correction of all single-bit errors and detection of two-bit errors.

Table 4 - Protection bit states

Bit	7	6 F	5 V	4 H	3 P3	2 P2	1 P1	0 P0
	1	0	0	0	0	0	0	0
	1	0	0	1	1	1	0	1
	1	0	1	0	1	0	1	1
	1	0	1	1	0	1	1	0
	1	1	0	0	0	1	1	1
	1	1	0	1	1	0	1	0
	1	1	1	0	1	1	0	0
	1	1	1	1	0	0	0	1

3.5.3.1 Timing reference signals - Video, 525-line systems

Figure 4a is a spatial representation of the timing reference signals during a television frame.

Lines are numbered from 1 through 525 as shown in figure 3a.

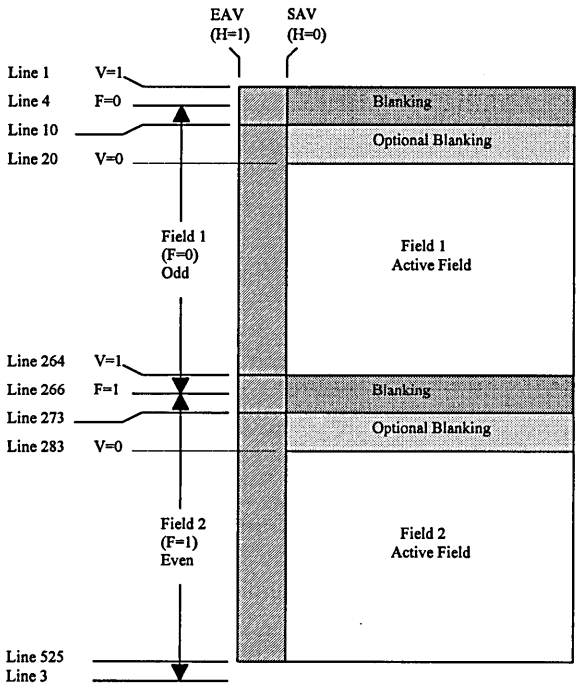


Figure 4a - Video timing reference signal locations, 525-line systems

Small blocks of data, less than 268 words in total length, including the HANC sequence (as described in 3.6.1), can be transmitted within the horizontal blanking period on every line.

Large blocks of data, up to 1440 words in total length, including the VANC sequence, can be transmitted within the interval starting with the end of SAV and terminating with the beginning of EAV on lines 1 through 19 and 264 through 282 only.

Video data will not be present on lines 1-9 and 264-272. Video data, other than image data, may optionally be present on lines 10-19 and 273-282. Ancillary data could be optionally transmitted in the active portion of these lines.

The words during:

- horizontal blanking period on every line;
- the active portion of lines 1-9 and 264-272;
- the active portion of lines 10-19 and 273-282 (when non-image video data is not present);

not used to transmit ancillary data must have the following values:

- the words corresponding to Y samples must have the value 10h
- the words corresponding to C_B and C_R samples must have the value 80h

3.5.3.2 Timing reference signals - Video, 625-line systems

Figure 4b is a spatial representation of the timing reference signals during a television frame.

Lines are numbered from 1 through 625 as shown in figure 3b.

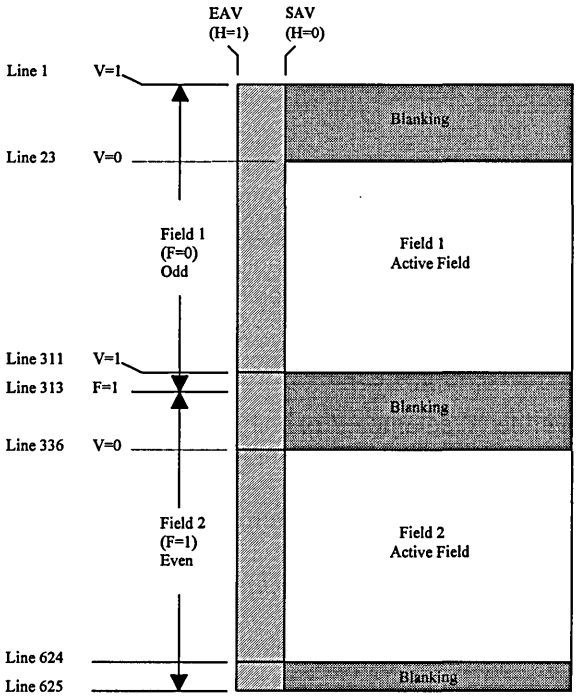


Figure 4b - Video timing reference signal locations, 625-line systems

Small blocks of data, less than 280 words in total length, including the HANC sequence (as described in 3.6.1), can be transmitted within the horizontal blanking period on every line.

Large blocks of data, up to 1440 words in total length, including the VANC sequence, can be transmitted within the interval starting with the end of SAV and terminating with the beginning of EAV on lines 1 through 22 and 311 through 336 and 624 through 625 only.

Video data will not be present on lines 1-22 and 311-336 and 624-625. Ancillary data could be optionally transmitted in the active portion of these lines.

The words during:

- horizontal blanking period on every line;
- the active portion of lines 1-22 and 311-336 and 624-625;

not used to transmit ancillary data must have the following values:

- the words corresponding to Y samples must have the value 10h
- the words corresponding to C_B and C_R samples must have the value 80h

3.5.4 Output formats

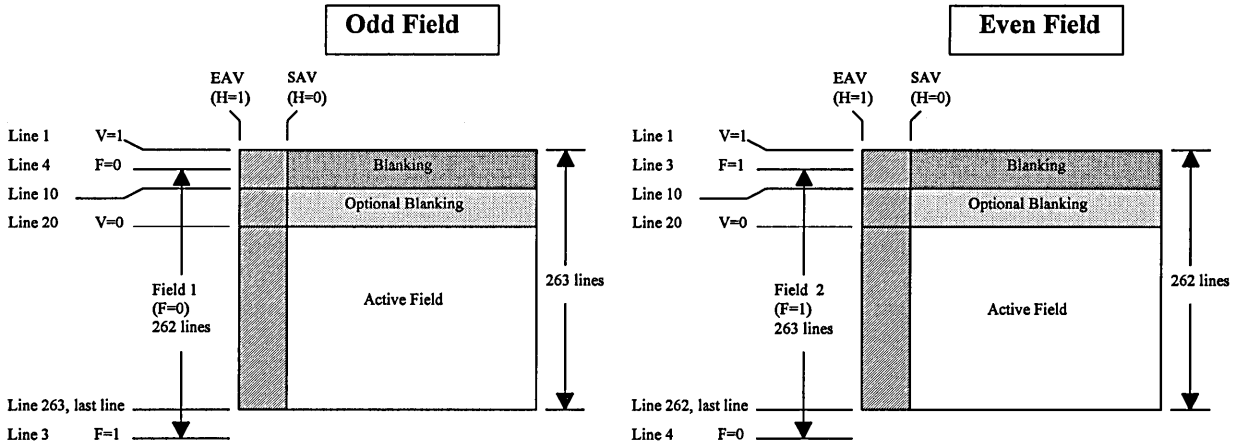


Figure 4c - CCIR656-NTSC and 640x480 Square Pixel @60Hz (Full Frame Rate)

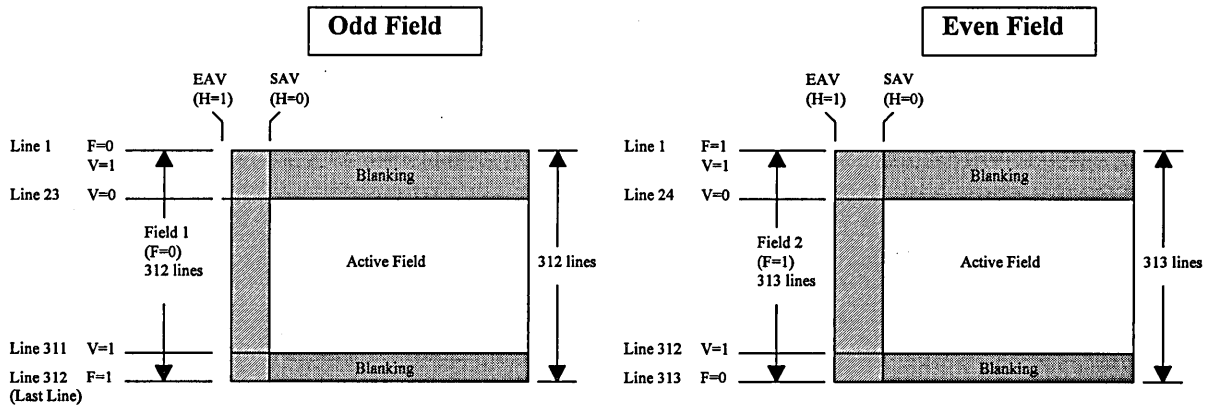


Figure 4d - CCIR656-PAL @50Hz (Full Frame Rate)

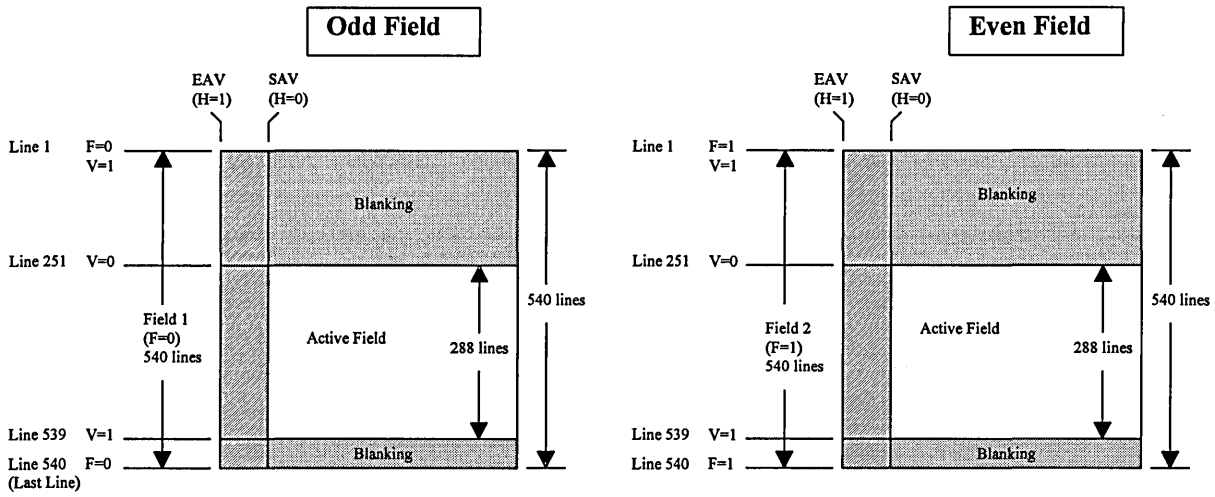


Figure 4e - 768x576 Square Pixel @ 25Hz (Half Frame Rate)

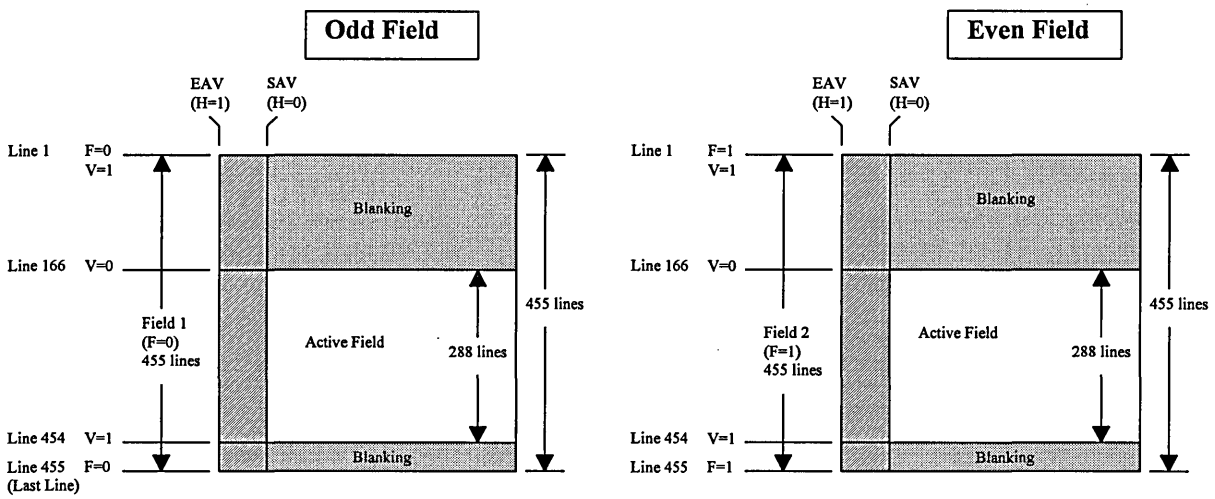


Figure 4f - 768x576 Square Pixel @30Hz (Half Frame Rate)

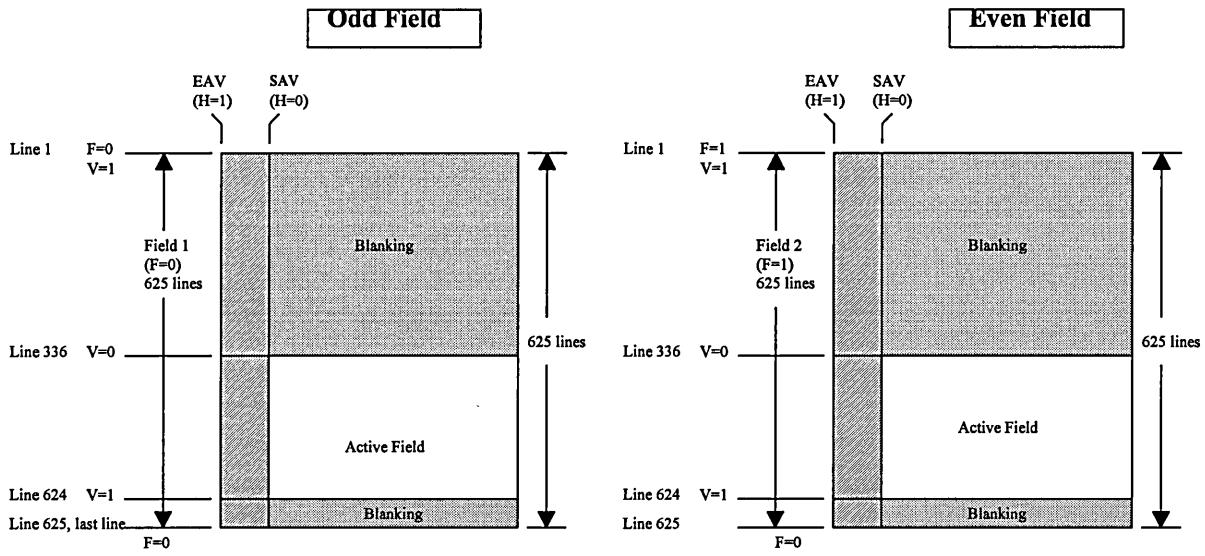


Figure 4g - CCIR656-PAL @ 25Hz (Half Frames Rate)

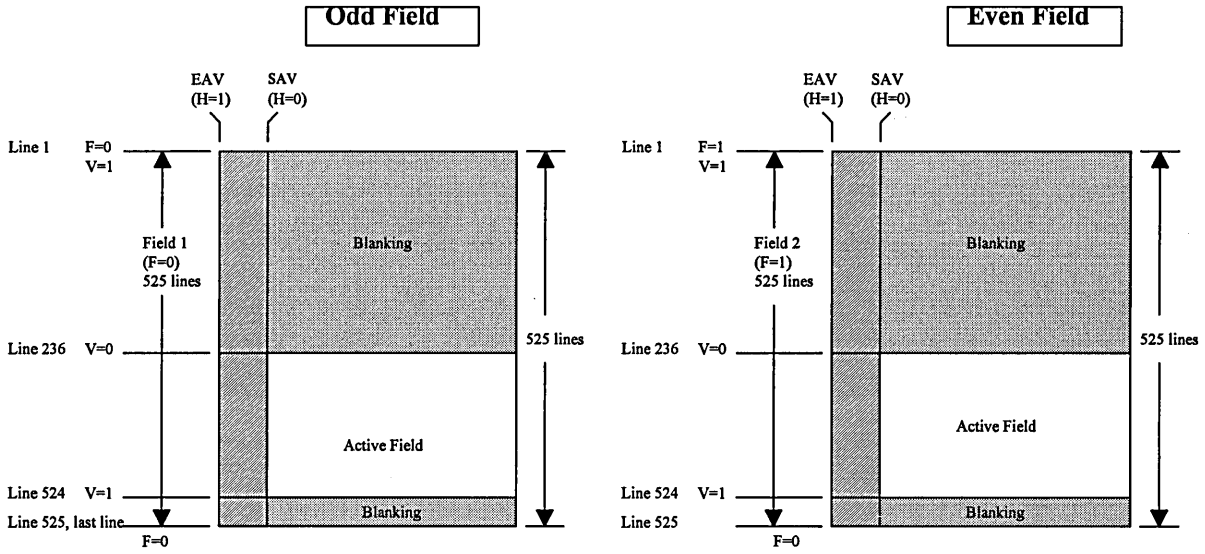


Figure 4h - Modified CCIR656-PAL @ 30Hz (Half Frames Rate)

3.6 Ancillary data signal format

Ancillary data may be inserted in any portion of the data stream not occupied by timing reference signals or video data (see 3.4.1 and 3.4.2). Two categories of ancillary data, horizontal ancillary (HANC) and vertical ancillary (VANC), are defined for different portions of the data stream. Note that the three-word header used to identify ancillary data is the same for HANC and VANC.

3.6.1 Horizontal Ancillary (HANC) data

HANC data are permitted in all horizontal intervals, but not in the active portion of lines. Each block of HANC data is preceded by a three-word ancillary data header

00h FFh FFh

The ancillary data header may occur multiple times during each horizontal blanking period if different blocks of data are to be transmitted.

All permitted data identification words and data formats will protect the values 00h and FFh.

3.6.2 Vertical Ancillary (VANC) data

For 525-line system, VANC data are permitted only in the active portion of lines 1-13, 15-19, 264-276, and 278-282. (Lines 14 and 277 are reserved for digital vertical interval time code (DVITC) and video index.) For 625-line system, VANC data are permitted only in the active portion of line 1-19, 21-22, 311-332, 334-336, an d 624-625. Each block of VANC data is preceded by the three-word ancillary data header

00h FFh FFh

The ancillary data header may occur multiple times during each line period if different blocks of data are transmitted

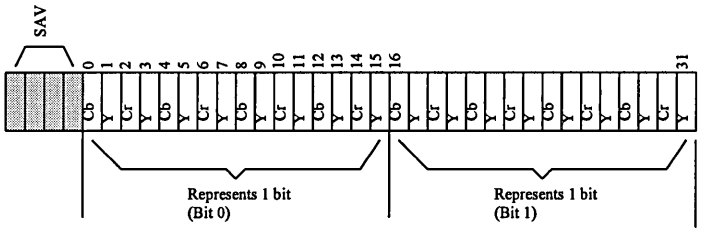
All permitted data identification words and data formats will protect the values 00h and FFh.

3.7 Embedded Camera status information

This signal, if present, is carried by the video data in the active portion of line 18 and 281 for the 525-line systems and line 21 and 331 for the 625-line system. A total of 80 bit is represented serially by the video data samples of the active portion of the line. 8 pixel samples are used to represent 1 bit.

The first pixel the active portion of the line (word 0 and 1 of the multiplexed signal, normally a C_B and Y sample) represents the least significant bit (bit 0) of status bit stream. The next 7 pixels are duplicate of the first pixel. The ninth pixel represents bit 1, etc. The last 8 pixels of the active portion of the line (word 1272-1279 of the multiplexed signal) represents the most significant bit (bit 79) of the status bit stream.

For all samples, a value of white(Y=EBh, Cb=80h, Cr=80h) represents a binary "one" for the status bit, and a value of black(Y=10h, Cb=80h, Cr=80h) represents a binary "zero" for the appropriate status bit.



Bit	Signal Name	One	Zero
0	Push button	pressed	open
1-79	un-defined		

3.7.1 525-line systems

The active portion of line 18 and 281 contains the camera's status information.

3.7.2 625-line systems

The active portion of line 21 and 331 contains the camera's status information.

3.8 Reserved vertical lines

3.8.1 525-line systems, Digital vertical interval time code and video index

Digital vertical interval time code (DVITC) and video index, if present, are carried by the data in the active portion of lines 14 and 277.

3.8.1.1 DVITC

This signal, if present, is carried by the luminance data in the active portion of lines 14 and 277.

3.8.1.2 Video index

This signal, if present, is carried by the color-difference data in the active portion of lines 14 and 277. A total of 80 8-bit data words is represented serially by DATA 0 of the 640 color-difference samples of the active portion of the line.

The first color-difference word of the active portion of the line (word 0 of the multiplexed signal, normally a C_B sample) represents the least significant bit (bit 0) of video index word 0. The second color-difference word represents bit 1 of the same word, etc. The last color-difference word of the active portion of the line (word 1278 of the multiplexed signal, normally a C_R sample) represents the most significant bit (bit 7) of video index word 79.

For all samples, a value of 81h represents a binary "one" for the appropriate video index bit, and a value of 80h represents a binary "zero" for the appropriate video index bit.

This transmission method ensures that, after digital to analog conversion, the video signal may be sent to an NTSC encoder without any requirement for special blanking. DVITC will be preserved through the encoder without interference from any video index information which may be present.

3.8.2 625-line systems, Equipment self-checking

This signal, if present, is carried by the luminance and chrominance data in the active portion of line 20 and 333.

3.9 Clock signal

3.9.1 Clock signal description (at transmitter)

The clock signal is a 27.00 MHz square wave as shown in figure 5.

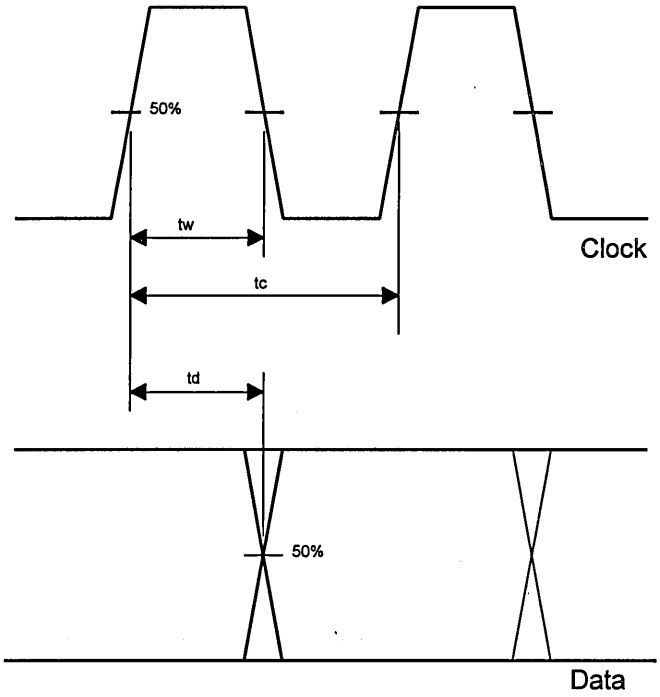
The clock pulse width(tw) is 18.5ns +/1 3.0ns.

3.9.2 Clock jitter

The peak-to-peak jitter between rising edges shall be within 2 ns of the average time of the rising edge computed over at least one field.

3.9.3 Clock data timing relationship

The positive transition of the clock signal nominally occurs midway between data transitions (figure 5).



tw = 18.50 ns +/- 3.0ns

tc = 37.00 ns (nomial)

td = 18.50 ns +/- 3.0ns

Figure 5 - Video clock to data timing (at transmitter)

4 Serial control bus

The serial control bus is the I^2C-bus developed by Philips Semiconductor.
The serial control bus protocol is an 8-bit byte protocol controlled by the receiver. To the receiver, the control bus looks like an 8-bit bidirectional channel, down which short packets are sent, and sometimes a data is read back. Each source device appears to the receiver as a set of 8-bit registers, addressable through a pair of *station addresses*. Packets are used to write and read the contents of these device registers.

There are three formats of packets: Write format, Address Set format, and Read format. Each packet is addressed to a device by its station address. The LSB of the station address is the data direction bit. This bit is set to 0 in Write and Address Set packets, and to 1 for Read packets.

4.1 Packet formats

4.1.1 Write format

The write format consists of a three or more byte packet. The first byte is the station address with the data direction bit set to '0' which indicates a write. The second byte is the device register address (0..255). The source device then sends one or more bytes on the bus to the register addressed. The number of bytes send by a Write format depends on the register being written, and can be up to 258 bytes long.

4.1.2 Address Set format

The address set format consists of a two byte packet. The first byte is the station address with the data direction bit set to '0' which indicates a write. The second byte is the register address. The address set format is the same as the write format but the register data is not sent. It is used to set the address for the read format.

4.1.3 Read format

The read format consists of a one byte packet. The first byte is the station address with the data direction bit set to '1' which indicates a read. The source device then sends one or more bytes back on the bus, from the register addressed by the last write format or address set format packet. The number of bytes returned by a Read format depends on the device and register being read, and can be up to 256 bytes.

4.1.4 Nonexistent registers

It is valid for the receiver to read or write registers that do not exist within the addressed device. Writing to such a register shall have no effect. Reading from such a register shall return bytes of value FFh, as many such bytes as the receiver chooses to read.

4.2 Digital video camera

The station address for a digital video camera is 56h for writes and 57h for reads.

The registers are not all defined for this revision of the specification, but are reserved. The definition and the address of the newly created registers will be publish as soon the information are available.

5 Analog audio channel

5.1 General

The interface of the analog audio signals consists of a unidirectional 3-conductor interconnection from the source to the receiver. Voltage-driven mono differential audio signals can be transmitted.

5.2 Interface Characteristics

The source and receiver shall have characteristics in accord with table 8 below.

Table 8 - Analog audio interface characteristics

Singal format	mono differential voltage driven
Voltage levels	2 Vp-p typical differential
Frequency range	20Hz to 20kHz maximum
Driver impedance	600 ohms maximum (at f=1kHz, Vo = 1Vp-p)
Receiver impedance	600 ohms minimum
Grounding (ARTN)	shall be grounded at driver

6 Electrical characteristics

6.1 General

The nine digital video signals shall be transmitted via single-ended signal pairs.

Although the use of TTL technology is not specified, the line driver and receiver must be TTL-compatible to permit the use of standard TTL parts for either or both ends in applications where such TTL parts are deemed adequate.

Standard TTL parameters are provided in Annex A.

6.2 Digital transmitter characteristics

6.2.1 Output impedance

The transmitter digital drivers shall have a single-ended output with a nominal impedance of 50 ohms.

6.2.2 Signal amplitude

Shall be in accordance with 2.1.

6.2.3 Rise and fall times

Rise and fall times shall be no longer than 5 ns and shall differ by not more than 2 ns, as measured between the 20% and 80% amplitude points across a 132-ohm resistor connected to the output terminals without any transmission line.

6.3 Digital receiver characteristics

6.3.1 Terminating impedance

The cable shall be terminated by 110 ohms +/- 10ohms to ground.

6.3.2 Maximum input signal

The line receiver must sense properly the binary data when connected directly to a line driver operating at the extreme voltage limits permitted by 6.2.2.

6.3.3 Input sensitivity

<to be determined>

6.4 Power characteristics

The receiver shall provide DC power on its connector in accord with table 9.

Table 9 - Power characteristics

Pin	Nominal voltage	Voltage tolerance	Max ripple (mVp-p)	Max current (mA)
20	+12V	+/- 10%	150	
22	-12V	+/- 10%	150	
23	+5V	+/- 5%	100	

7 Mechanical characteristics

7.1 General

This clause defines the mechanical specifications for the interface of digital video systems used in environments where the physical distance between devices is very limited, in indoor or 'office' conditions.

7.2 Interconnecting cable characteristics

7.2.1 Cable length

Applications of this interface are expected to use cable lengths between 1 m (3 ft) and the maximum specified length of 3 m (10 ft). For these lengths, cables of reasonable uniformity will generally give satisfactory results. It is not expected that applications of this interface will require equalization, however for cable lengths greater than 3 m, the cable and termination characteristics become more critical, and equalization may be required.

7.2.2 Cable construction

The cable shall be constructed to minimize the effects of crosstalk between signal lines, the susceptibility of the signal lines to external noise, and the transmission of interface signals to the external environment. Recommended twisted pairs for data and ground lines are provided in table 10.

The cable shall contain an overall shield to minimize radiation, carried through the cable assembly and connector via the connector body at the receiver end and connected source device shield.

7.3 Connector characteristics

7.3.1 Mechanical considerations

The mechanical characteristics of the connector conforming to the industry standard 26-contact high-density 3-row D subminiature connector and the 50 pin ALT connector described below. These connectors are commonly available in the market.

Note: Most applications of this interface require that the connectors be inserted many times. The materials used in the connector should be appropriate to the application.

7.3.2 Connector contact assignments

The connector contact assignments shall be in accord with table 10.

Table 10a - 26 Pin Connector contact assignments

Pin	Signal	Pairing	Pin	Signal	Pairing
1	DCLK+	Pair 1	15	GND	Pair 6
2	DATA 0	Pair 2	16	GND	Pair 7
3	DATA 1	Pair 3	17	GND	Pair 8
4	DATA 2	Pair 4	18	GND	Pair 9
5	DATA 3	Pair 5	19	I2CD	Pair 10
6	DATA 4	Pair 6	20	+12V	Pair 10
7	DATA 5	Pair 7	21	I2CCLK	Pair 11
8	DATA 6	Pair 8	22	-12V	Pair 11
9	DATA 7	Pair 9	23	+5V	
10	GND	Pair 1	24	Aud-	
11	GND	Pair 2	25	AudGnd	
12	GND	Pair 3	26	Aud+	
13	GND	Pair 4	shall	GND	
14	GND	Pair 5			

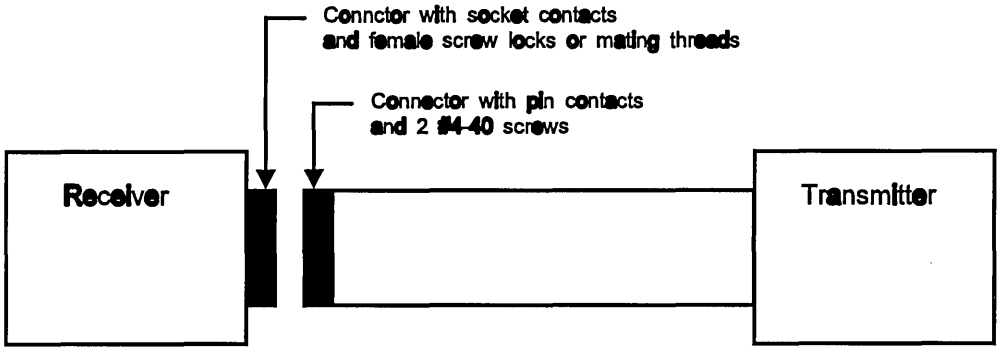
Note: Pairings are recommendations only.

Table 10b - 50 Pin ALT Connector contact assignments

Pin	Signal	Pairing	Pin	Signal	Pairing
1	IDCLK+	Pair 1	2	GND	Pair 1
3	IDATA 0	Pair 2	4	GND	Pair 2
5	IDATA 1	Pair 3	6	GND	Pair 3
7	IDATA 2	Pair 4	8	GND	Pair 4
9	IDATA 3	Pair 5	10	GND	Pair 5
11	IDATA 4	Pair 6	12	GND	Pair 6
13	IDATA 5	Pair 7	14	GND	Pair 7
15	IDATA 6	Pair 8	16	GND	Pair 8
17	IDATA 7	Pair 9	18	GND	Pair 9
19	GPIBIN	Pair 10	20	GND	Pair 10
21	WRDCL	Pair 11	22	GND	Pair 11
23	-12V	Pair 12	24	IICCLK	Pair 12
25	audioP	Pair 13	50	audioN	Pair 13
48	+12V	Pair 14	49	IICDATA	Pair 14
27	GPIBO	Pair 15	26	GND	Pair 15
29	+5V	Pair 16	28	GND	Pair 16
31	ODATA0	Pair 17	30	GND	Pair 17
33	ODATA1	Pair 18	32	GND	Pair 18
35	ODATA2	Pair 19	34	GND	Pair 19
37	ODATA3	Pair 20	36	GND	Pair 20
39	ODATA4	Pair 21	38	GND	Pair 21
41	ODATA5	Pair 22	40	GND	Pair 22
43	ODATA6	Pair 23	42	GND	Pair 23
45	ODATA7	Pair 24	44	GND	Pair 24
47	ODCLK+	Pair 25	46	GND	Pair 25

7.3.3 Cable connector assembly

Cable connectors employ pin (male) contacts and equipment connectors employ socket (female) contacts (see figure 6).



7.3.4 Connector retaining mechanism

The cable connectors shall be provided with #4-40 mounting screws and the equipment connectors shall be provided with female screw locks or with mating threads as shown in annex B.

7.3.5 Connector shielding

The connector shell shall be electrically conductive and provide electrical connection between cable shield and receiver chassis ground.

Annex A - TTL parameters (normative)

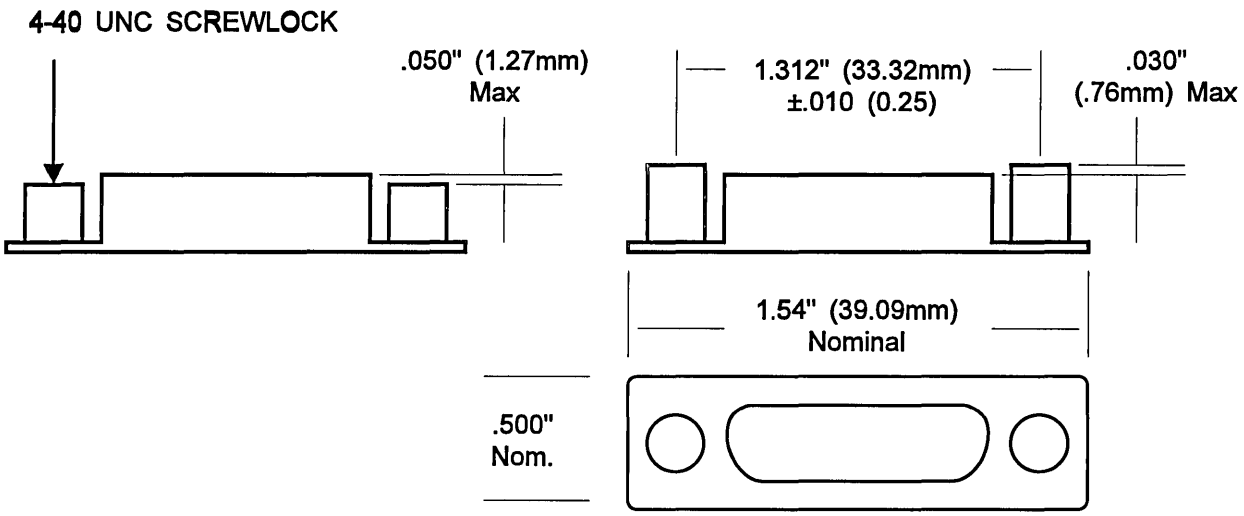
See section 2.1.

Annex B - Connector characteristics (normative)

The interface employs the 26-contact high-density D subminiature connector, with the connectors on the receiver using socket (female) contacts and the connectors on the cable using pin (male) contacts. Connectors are locked together by two #4-40 screws on the cable connectors, which go in female screw locks mounted on the equipment connector.

The relative position of the connector and female screw lock is defined in figure B.1.

It is recommended that the cable connectors employ a conductive backshell to maintain shielding of the signal conductors. Care must be taken to select designs that are appropriate for use with the screw-latching method specified.



Annex C - Connector orientation (informative)

Vertical or horizontal mounting: Pin 1 uppermost.

Annex D - Monochrome operations (informative)

Monochrome operation can be achieved by setting the color-difference signals (C_R, C_B) to zero (80h).

Annex E - Error detection and correction in video timing signal (informative)

[dentical to ANSI/SMPTE 125M-1992, Annex F.

Annex F - Comparison to ANSI/SMPTE 125M-1992 (informative)

References in [ ] are to sections or tables in this (DVCI) document.

1. No 10-bit wide video format is specified. [1.1, 1.3, 3.1, etc.]

2. The frame parameters are changed to accomodate an image of 640 x 480 pixels. [Table 1]

3. The connector is a 26-pin subminiature D, with corresponding cable changes. [7.3.1, Annex B]

4. The digital signal convention is single-ended TTL. [2.1, 6]

5. Analog audio channels are added. [5]

6. Several levels of DC power are provided by the receiver for the transmitter's use. [6.4]

7. An I2C-bus is provided for communication between transmitter and receiver. [4]

8. A protocol is defined for communicating with a video camera via the I^2C-bus. [4.2]

Annex G - Bibliography (informative)

CCIR Recommendation 601-1, Encoding Parameters of Digital Television for Studios

CCIR Report 962-1, The Filtering, Sampling and Multiplexing for Digital Encoding of Colour Teleision Signals

ANSI/SMPTE 125M-1992, Component Video Signal 4:2:2 - Bit Parallel Digital Interface

The I2C-bus and How to Use It (document number 98-8080-575)

Philips Electronics North America Corporation
<contact info tbd>

Documents are in preparation to cover the embedded digital audio signal, but are not yet available.

Cesar Audi/Video Capture Board

Hardware Specification

Revision A

April 19, 1994

Logtiech Confidential

LOGITECH

General Description

Movieman is an integrated audio/video capture board for the IBM compatible computer. It is targeted at the emerging desktop video markets. The board contains many functions that are useful for the desktop video.

The primary function of the Movieman is to captures and digitizes both the video and audio signals and store them to the computer's memory.

Movieman does not provides video playback. It utilizes the host to play back the motion image on the computer graphic terminal. However, it does provides stereo audio playback. It is capable of driving 6 watt per channel.

Movieman provides an control-L connector to remotely control the video terminal device(VTR) sure as camcorder and VCR.

SCSI interface connnector is provided for connection to the CD ROM drive.

Revision History

April 19, 94	Kwok Yan	Rev A - Initial Release

References

741663-00	Movieman Board Inteface Specification
741664-00	Cesar Expansion Bus Interface Specification
620819-00	Sony Control-L Data Sheet
360310-00	Data sheet for VAD2150
360279-00	Data sheet for VSP 2860
360281-00	Data sheet for ASCO 2300
360363-00	Data sheet for the WAVIA ASIC

Features

Video Captures

Analog video inputs

Number of inputs	2
Standard supported	NTSC, PAL
Color format	YUV 4-1-1 and YUV 4-2-2

Digital video input

Number of inputs	1
Satandard supported	Modified SMPTE 601 YCbCr YUV 4-2-2
Data width	8-bit
Interface	TTL

On all of the video inputs

Frame rate	30 frame/sec to hard disk @160x120 pixel 30 frame/sec to system @320x240 pixel 15 frame/sec to hard disk @320x240 pixel Still image @ 640x240 pixel
Horizontal scaling	1/64 to 64/64
Horiz. decimation	4 pel average
Vert. decimation	2 line average

Audio captures

Input sources	Five inputs Digital camera input at line level input TUNER input at line level AUDIO IN at line level or mic level CD analog audio at line level
Sample frequency	16-bit mono or stereo @ 44kHz 16-bit mono or stereo @22KHz 8-bit mono or stereo @ 11kHz
Resolution	16 bits

Audio Playback

Power rating	6 watts per channel @ 8 Ohm
Mixing sources	Sampled Digitized data CD Analog audio Standard OPL3 FM synthesizer

Control-L

Supports Sony serial data communication protocol.

Host Interface

Base address supported	150 hex (default), 160, 170 110, 120, 130
I/O Data path width	16-bits
Interrupt supported	2, 5, 7, 10, 11, 12 and 15
DMA supported	None

Expansion Connectors

The expansion connectors allows additional devices to be connected to Cesar. It supports following devices

- Memory
- Video
- Audio

Physical Specifications

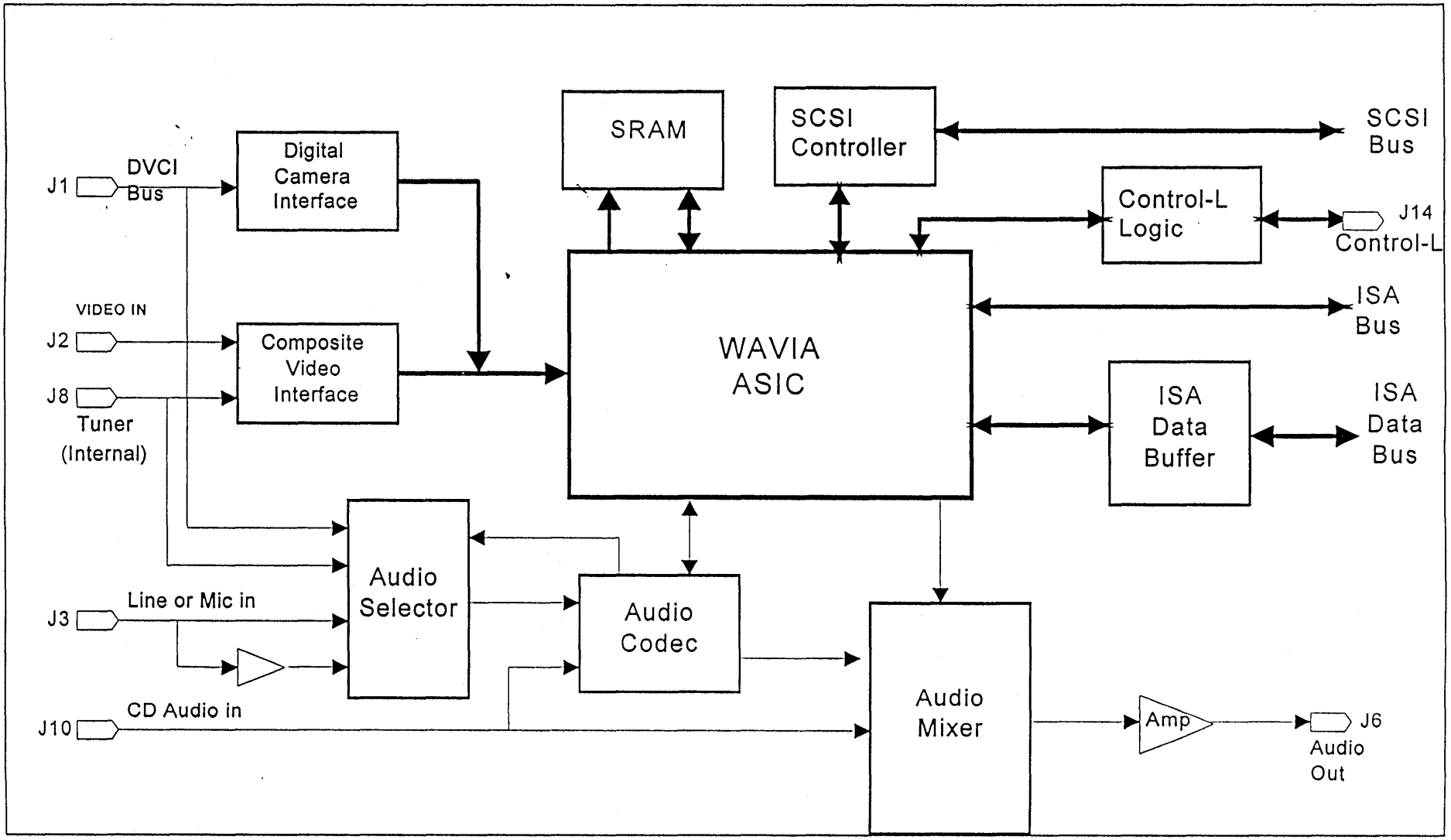
Board dimension

Height	4.8 inches
Length	10.0 inches

Connectors

Digital camera	J1	DB26 High Density, 3 rows, female
Video	J2 J8	RCA female jack 10 pins header
Audio	J3, J6 J10	3.5mm stereo phono jacks 5 pin header
Control-L	J14	2.5mm phone jack
SCSI	J18	50 pins

Block Diagram



Block Diagram Description

Composite Video Interface Section

Video Source Selection

There are two composite video sources to select from: VIDEO IN and TUNER. VIDEO IN is expected to have video signal of 1 volt peak to peak, while the TUNER input is expected to have signal of 2 volts peak to peak. TUNER input is intended to be used for internal video tuner that can generate 2 volt peak to peak video signal. VIDEO IN is the primary composite video input.

These 2 inputs go to the VAD2150 chip. The WAVIA ASIC generates the select signals: VAD_SEL and VID_TYPE.

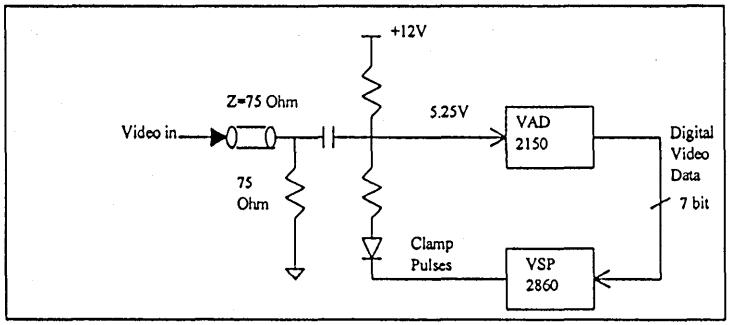
VID_TYPE	VAD_SEL	Video Source Selected
0	x	Digital Camera
1	0	TUNER
1	1	VIDE IN

Video Standard Selection

The VAD21500 chip supports PAL and NTSC video signal format. The format are selected by programming the VSP registers, established through software

Clamping Operation

The dc level of the video signal would be blocked due to ac input coupling capacitors and other board effects. The average dc level of the incoming video also varies according ot the "whiteness" or "blackness" of the picture it carries. These effects cause the dc level to vary, and a method of restoring the dc level is needed. The time period before and after the h-sync represents the black level and is used as a reference. This reference needs to be re-established in order to properly reproduce the original video. Clamping/DC restoration provides this. The figure below illustrates the loss of dc level due to capacitive coupling.



Bias setting is done by balancing of the combination of the pullup and pulldown resistors and the digitally produced low going clamping pulses. These clamping pulses (low signals) are generated by the VSP 2860 chip that forward bias the diode effectively grounding the bottom bias resistor.

Clamping pulses may be also generated depending on the amount of drift of the black level. The amount of drift is evaluated digitally, after conversion of the analog voltage to 128 different digitaal levels form the VAD. Figure below shows the schematic diagram that is used for clamping

AA Filter

a π-filter is used (3rd order) with a cutoff frequency of 6MHz. The filter helps in eliminating aliasing effectes.

IM-BUS Control

The IM-BUS (intermetall Bus) is a 3 wire on-board serial bus supported by ITT. This facilitate communication between the WAVIA ASIC and the VSP chip. Please refer to the VSP 2860 data sheet for further information

Digital Camera Interface Section

Clock Synchronizer

The clock synchroinzer cirucuit prevents gitches on the M_CLK and V_CLK line when switching video sources. It times the C_EN~ signal to the C_CLK and VSP_OE~ signal to the VSP_CLK.

Watchdog Circuitry

When the digital camera is selected, the M_CLK is generated by the clock line from the digital camera. If the camera is not connected to the Movieman, the board can hang during system access. The watchdog is design to eliminate the hang. When no clock is detected on the clock line from the digital camera, the watchdog circuitry prevent C_EN~ from asserting. This causes VSP_OE~ to be asserted.

Power Switches

The Movieman provides the power neccessary to drive the digital camera. The board provides 3 voltages: +5Volt, +12Volt and -5Volt. The on/off switch can be programmed through a bit in the register of the WAVIA chip.

Video Expansion Support

Additional video source can be connected to the Movieman via the expansion bus. The expansion connectors provides the daugther boards an access to the VIDEO bus. These signals, XVID_RQ~, XVID_ACK~ and XVID_OE~, are used by the expansion board to access the VIDEO bus. XVID_RQ~ initiates the request to the arbitration logic. The arbitrator grant the expansion board the access to the VIDEO bus by asserting XVID_ACK~. The expansion board take control of the VIDEO bus by asserting XVID_OE~. See Video Srouce Arbitration Logic for more details.

It is the responsibility of the expansion board to provide the V_CLK and M_CLK signals. Movieman will hang if M_CLK is not provided. M_CLK is limited to 25MHz. V_CLK is limited to 17.5MHz.

Video Source Arbitration Logic

The Video Source arbitration logic arbitrate the VIDEO bus between the 3 video devices: digital camera, VSP 2860 and the expansion video. The digital camera has the highest priority, the expansion video is next and VSP has the lowest.

The following list the truth table for the arbitrator. ZEye_En is the enable signal for the digital camera. It is asserted only when VID_TYPE is low and there are clock pulses on the clock line from the digital camera, otherwise it is low.

ZEye_ En	XVID_ OE~	XVID_ RQ~		TP140	TP141	XVID_ ACK~
1	1	X		0	1	1
0	0	X		1	1	1
0	1	1		1	0	1
0	1	0		1	1	1
0	0	1		1	1	1
0	0	0		1	1	0

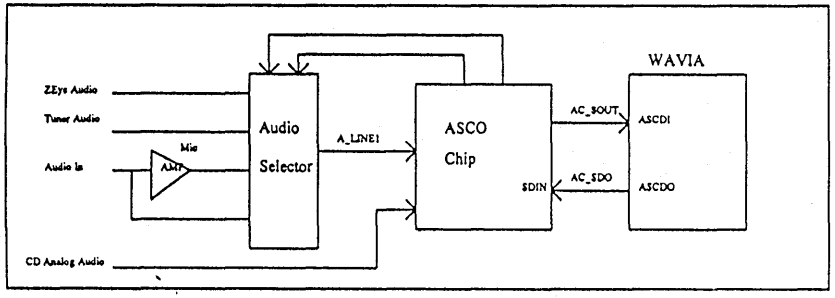
Audio section

Audio Sources

There are five audio input sources that Movieman can capture. Four of the sources are connected to the Audio Selectors. It output goes to the ASCO. The fifth audio source goes to the ASCO chip directly. The ASCO accepts 2 audio inputs.

Audio Selector

The Audio Selector select one of the 4 audio inputs to be capture by the ASCO codec chip. The output of the selector goes to the ASCO chips. The multiplexer control is generated by the ASCO chip. The select is programmed in the WAVIA ASIC and it is send to the ASCO chip.



Audio CODEC

The main audio smaple frequencies suppported are 44KHz, 22KHz and 11KHz(Stereo and mono). The ASCO chip also contains sophisticated AA filters. In addition, other frequencies are supported through the ASIC chip. Please refer to the WAVIA ASIC specification docuement.

Digital Format to WAVIA

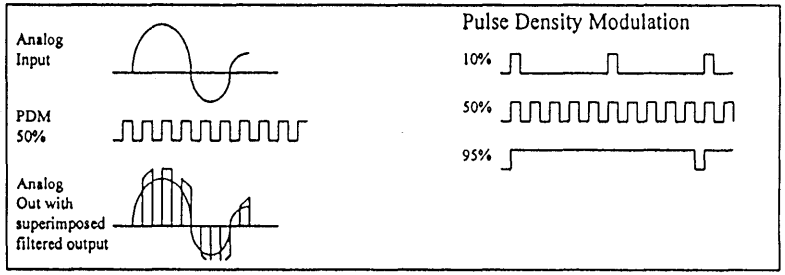
A 256 bit format is used for the digital interface between the WAVIA ASIC chip and the ASCO chip. There are 4 subframtes (64 bits each). Each subframe is divided into 16 bits of left and right channel audio each and 32 bits of control.

Audio Playback

ASCO chip converts the digital data send by the WAVIA to analog signal by mean of D-to-A converter. The analog audio output of the ASCO goes to the low pass RC filter to remove the high frequency components. The amplitude of the audio signal is reduced by half. The output then goes to the Audio mixer.

Audio Mixer Section

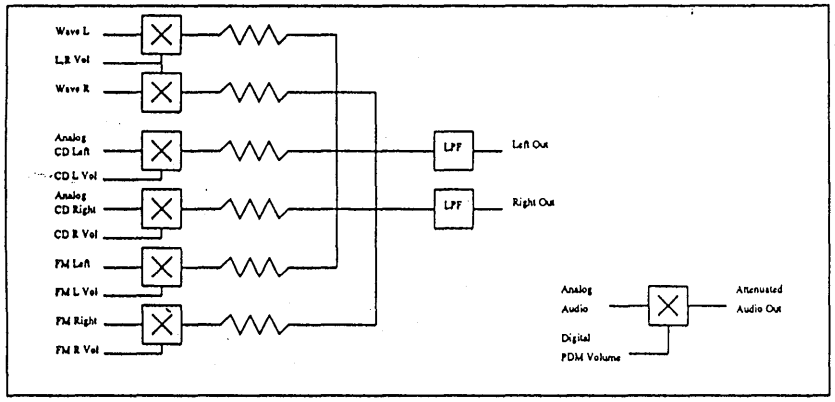
Analog inputs from the CD, FM synthesizer and the WAVE outputs (obtained as described above) are mixed together. A volume control step is provided before mixing for each of the audio sources. The figure below shows the analog audio input (WAVE, CD or FM synthesize signal) Pulse Density Modulation (PDM) technique is employed. Depending on the volume of the audio input, the PDM waveform will contain more high pulses or continuous high periods reflecting higher volume. Figure below show the principle of PDM volume control Also note that the operates on a linear scale. Internally, 32 logarithmic levels are used, converted from 256 linear levels. When the volume levels are changes on the various sources using software, the change amounts to a logarithmic level control from each of the panels. The WAVIA ASIC chip ontrols the PDM operation and outputs 5 PDM signals to the audio mixer section.



The modulated digital waveform is then multiplied with the audio input signal. The process is ismilar to chipping, and results in an attenuated analog audio signal, as shown above (left). For the WAVE audio, the left and right volume information is carried on a signal line, two lines are used for the other two types of audio signals. The resulting signal is mixed with other audio signals.

The ASCO has volume controls for both audio left and right, at a resolution of 4 bits each. This is insufficient, to represent final volume control but is sufficient for balance control. So, then the output volume control in the ASCO chip is used to output the balance information (which is th edifference of the audio left and right signals). Also contained in theASCO balance information are average and maximum levels. Only a signle output formt he WAVIA to the mixer section is necessary to control the final output levels for the WAVIA outputs.

Figure below illustrates the PDM volume input multiplied with the analog audio signal, mixed together with other audio sources. Note that the mixing implied in this section is a summation of the various multipled audio outputs. The left and right analog audio outputs then go through an amplifier, which then can drive speakers directly.



Audio Ouput Amplifier

The Movieman board uses the Signetics TDA1517 stereo power amplifier for audio amplification. TDA1517 can provides up to 6 watts of power for each channel.

Control-L Port

Using the Ctrl-L serial interface, an edit controller can send instructions to the video equipment sure as camcorders and VCRs to fast-forward, rewind, play using software control. Mose camcorers capable of remove operation are Control-L based, and can be used for video editing and other applications. In turn, the edit controller can potentially obtain information such as tape position form the video device.

The Control-L port is bi-directional, using a single wire for communication. The device designated as the master initiates the start signal, followed by the slaves reading the this signal. Bi-directional data communication begins immediately after. Please refer to the Sonly Ctrl-L specificaiton for further information. In addition, the host CPU can take contorl of the control-L interface by asserting the line state which can open other communication capabilities.

Movieman board supplies +5V dc onto pin 4 of the Control-l port. The switch to the +5V dc is turned off when this pin is shorted to ground. This is to protect the devices on Movieman board.

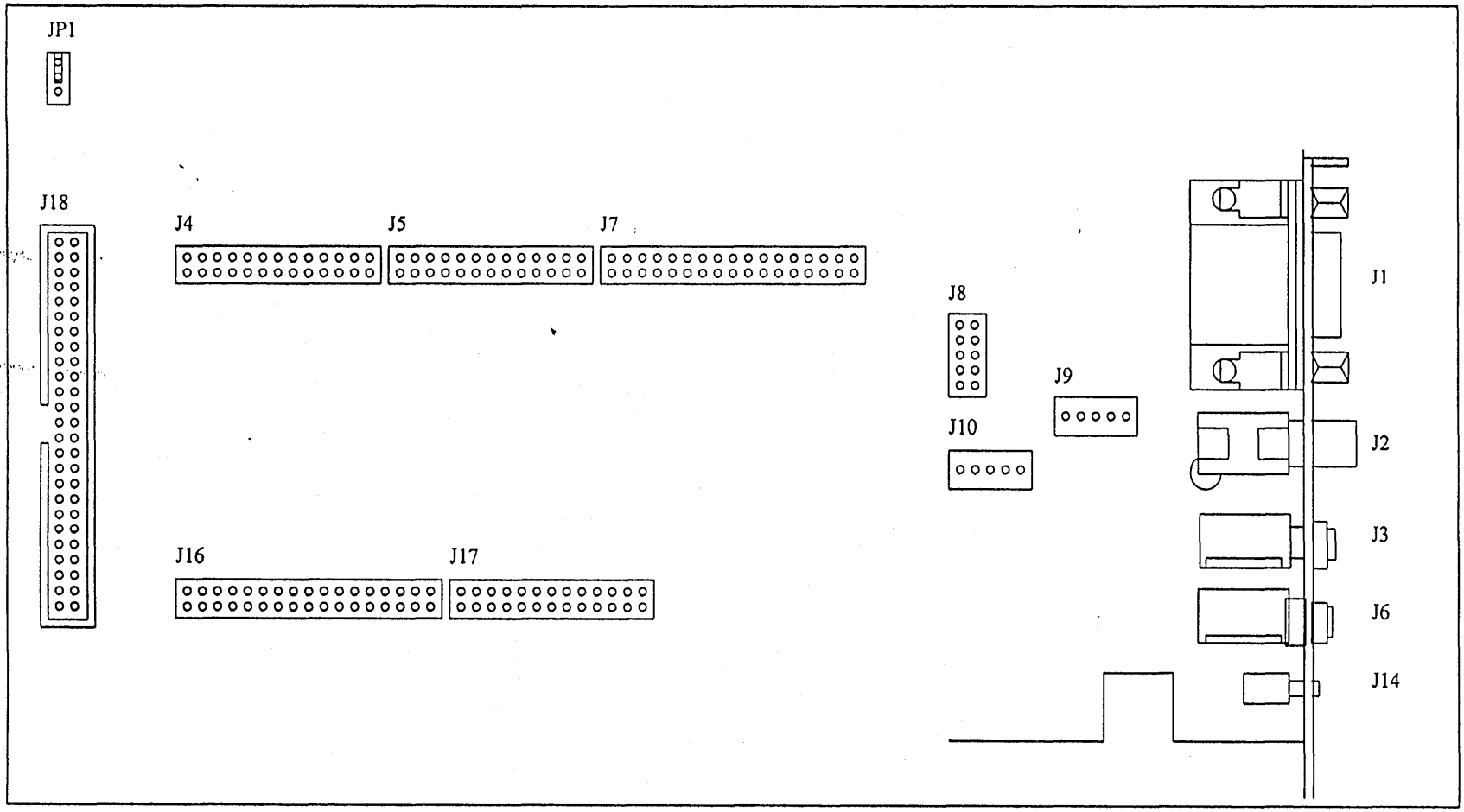
SCSI Controller

The SCSI controller is provides to interface with the CD ROM drive. The Movieman uses the Zilog 5380 to generate the timing for the SCSI interface.

The upper byte is to be ignored when accessing the 5380 chip. The Zilog 5380 chip is an 8-bit device while Movieman only accepts 16-bit I/O operation.

The HBUFLEN~ and W_IORD~ signals are added to fix a bug with the WinNov WAVIA chip. The WAVIA chip did not properly generate the enable signal for the low byte buffer when reading the 5380 chip. The HBUFLEN~ is asserted when the SCSI chip is read. At the same time W_IORD~ is negated so the WAVIA chip will not read the data on the data bus.

Connectors



Connectors Pin Definitions

Address Select Jumper : JP1

Pin	Definition
1-2	Primary Card, base address @ 150 Hex
2-3	Secondary, base address @110Hex

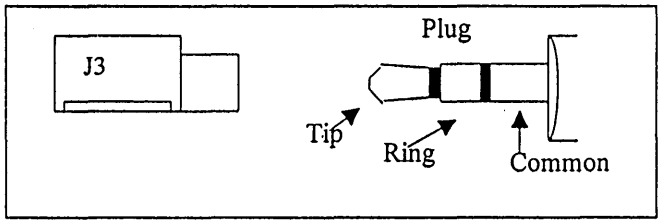
DVCI Connector : J1

Pin	Signals	I/O	Definition
1	DCLK	In	Data Clock
2	D0	In	Data bit 0 (LSB)
3	D1	In	Data bit 1
4	D2	In	Data bit 2
5	D3	In	Data bit 3
6	D4	In	Data bit 4
7	D5	In	Data bit 5
8	D6	In	Data bit 6
9	D7	In	Data bit 7
10	GND	Out	Data Clock return
11	GND	Out	Data bit 0 return
12	GND	Out	Data bit 1 return
13	GND	Out	Data bit 2 return
14	GND	Out	Data bit 3 return
15	GND	Out	Data bit 4 return
16	GND	Out	Data bit 5 return
17	GND	Out	Data bit 6 return
18	GND	Out	Data bit 7 return
19	I2CD	Bi	I2C control data
20	+12V dc	Out	+12V DC power supply
21	I2CCLK	Bi	I2C Control Clock
22	-12V dc	Out	-12 V DC power supply
23	+5V dc	Out	+5V DC power supply
24	MICR	In	Analog audio right channel input
25	MIC RTN	Out	Analog audio ground return
26	MICL	In	Analog audio left channel input

External Analog video input : J2

Pin	Definition
Tip	Composite Video
Shield	Ground

External Audio Input : J3



Pin	Definition
Tip	Left Signal
Ring	Right Signal
Common	Ground

Audio Out, Speaker Connector : J6

Pin	Definition
Tip	Left Signal
Ring	Right Signal
Common	Ground

Control-L Connector : J14

Pin	Definition
Tip	Control-L Serial Data
Ring	Power Supply (+5Vdc)
Common	Ground

Internal CD Audio In : J10

Pin	Definition
1	Ground
2	Analog audio left channel input
3	Ground
4	Analog audio right channel input
5	Ground

Tuner Connector : J8

Pin	Signals	I/O	Definition
1	Vid In	In	Composite video input
2	GND	Out	Ground
3	Reserved		
4	GND	Out	Ground
5	Reserved		
6	Reserved		
7	AudL	In	Analog audio left channel input
8	GND	Out	Ground
9	AudR	In	Analog audio right channel input
10	GND	Out	Ground

SCSI Interface : J18

Pin	Signals	I/O	Definition
Odd Pins, starting @1, except 25		Out	Ground
2	D0	Bi	Data bit 0 (LSB)
4	D1	Bi	Data bit 1
6	D2	Bi	Data bit 2
8	D3	Bi	Data bit 3
10	D4	Bi	Data bit 4
12	D5	Bi	Data bit 5
14	D6	Bi	Data bit 6
16	D7	Bi	Data bit 7
18	DBP~	Bi	Data parity bit
20	GND	Out	Ground
22	GND	Out	Ground
24	GND	Out	Ground
25	NC		No Connect, reserved
26	NC		No Connect, reserved
28	GND	Out	Ground
30	GND	Out	Ground
32	ATN-	Bi	Attention
34	GND	Out	Ground
36	BSY-	Bi	Busy
38	ACK-	Bi	Acknowledge
40	RST-	Bi	Reset
42	MSG-	Bi	Message
44	SEL-	Bi	Select
46	C/D-	Bi	Control/Data
48	REQ-	Bi	Request
50	IO-	Bi	Input/Output

Expansion Connectors

Connector Location : J4

Pin	Signals	I/O	Definition
1	GND	Out	Ground for the expansion board
2	XB_IRQ	In	Expansion board Interrupt request
3	M_CS2~	Out	Memory Sel 2 enable, @address=0x6xxx
4	M_CS3~	Out	Memory Sel 3 eanble, @address=0x7xxx
5	M_CLK	Bi	Memory clock for the WAVIA ASIC
6	GND	Out	Ground for the expansion board
7	XM_RD	In	Expansion Memory Read command
8	XM_RQ~	In	Expansion Memory Request
9	XM_ACK~	Out	Expansion Memory Acknowledge
10	MB_RQ~	In	Memory Bus Request
11	MB_ACK~	Out	Memory Bus Acknowledge
12	+5V dc	Out	+5Volt for the expansion board
13	MA18	Bi	Memory Address Bit 18
14	GND	Out	Ground for the expansion board

15	MA15	Bi	Memory Address Bit 15
16	MA16	Bi	Memory Address Bit 16
17	MA17	Bi	Memory Address Bit 17
18	MA14	Bi	Memory Address Bit 14
19	M_WE~	Bi	Memory Write Control
20	MA12	Bi	Memory Address Bit 12
21	MA13	Bi	Memory Address Bit 13
22	MA19	Bi	Memory Address Bit 19 (MSB)
23	MA7	Bi	Memory Address Bit 7
24	MA8	Bi	Memory Address Bit 8
25	MA6	Bi	Memory Address Bit 6
26	MA9	Bi	Memory Address Bit 9

Connector Location : J5

Pin	Signals	I/O	Definition
1	MA5	Bi	Memory Address Bit 5
2	MA11	Bi	Memory Address Bit 11
3	MA4	Bi	Memory Address Bit 4
4	M_OE~	Bi	Memory Output Enable control
5	MA3	Bi	Memory Address Bit 3
6	MA10	Bi	Memory Address Bit 10
7	MA2	Bi	Memory Address Bit 2
8	MA1	Bi	Memory Address Bit 1
9	MD15	Bi	Memory Data Bit 15 (MSB)
10	MD7	Bi	Memory Data Bit 7
11	MD14	Bi	Memory Data Bit 14
12	MD6	Bi	Memory Data Bit 6
13	MD8	Bi	Memory Data Bit 8
14	MD0	Bi	Memory Data Bit 0 (LSB)
15	MD13	Bi	Memory Data Bit 13
16	MD5	Bi	Memory Data Bit 5
17	MD9	Bi	Memory Data Bit 9
18	MD1	Bi	Memory Data Bit 1
19	MD12	Bi	Memory Data Bit 12
20	MD4	Bi	Memory Data Bit 4
21	MD10	Bi	Memory Data Bit 10
22	MD2	Bi	Memory Data Bit 2
23	MD11	Bi	Memory Data Bit 11
24	MD3	Bi	Memory Data Bit 3
25	XV_SWT~	In	Expansion Video Sampler wait
26	XV_STS~	Out	Expansion Video Sampler Status

Connector Location : J7

Pin	Signals	I/O	Definition
1	VD_U7	Bi	Video Chroma data bit 7 (MSB)
2	VD_U6	Bi	Video Chroma data bit 6
3	VD_U5	Bi	Video Chroma data bit 5
4	VD_U4	Bi	Video Chroma data bit 4
5	VD_U3	Bi	Video Chroma data bit 3
6	VD_U2	Bi	Video Chroma data bit 2
7	VD_U1	Bi	Video Chroma data bit 1
8	VD_U0	Bi	Video Chroma data bit 0 (LSB)
9	VC_CHX	In	Video Converter Chroma extend
10	VC_HB~	Bi	Video Converter Skew Data/Horizontal Blank
11	VC_VS~	Bi	Video Converter Vertical Blank
12	+5V dc	Out	+5Volt for Expansion board
13	VD_Y7	Bi	Vdeo Luma data bit 7 (MSB)
14	VD_Y6	Bi	Vdeo Luma data bit 6
15	VD_Y5	Bi	Vdeo Luma data bit 5
16	VD_Y4	Bi	Vdeo Luma data bit 4
17	V_CLK	Bi	Clock for Video logic in WAVIA ASIC
18	GND	Out	Ground for Expansion board
19	VD_Y3	Bi	Vdeo Luma data bit 3
20	VD_Y2	Bi	Vdeo Luma data bit 2
21	VD_Y1	Bi	Vdeo Luma data bit 1
22	VD_Y0	Bi	Vdeo Luma data bit 0 (LSB)
23	VAD_SEL	Out	Video Source Select #1
24	XVID_OE~	In	Expansion Video Output enable
25	IM_SD~	Bi	IM-Bus Serial Data (for ITT chips)
26	IM_CLK~	Out	IM-Bus Clock
27	IM_INDENT~	Out	IM-Bus Identifier
28	I2_CLK~	Bi	I2C-Bus Clock
29	VSP_CLK	Out	Clock from Video logic on the board

30	M_CLK	Bi	Clock for memroy logic in WAVIA ASIC
31	XVID_ACK~	Out	Expansion Video bus Acknowledge
32	XVID_RQ~	In	Expansion Video bus request
33	H_-12v	Out	-12Volt for expansion board
34	H_+12V	Out	+12Volt for expansion baord

Connector Location : J16

Pin	Signals	I/O	Definition
1	LD15	Bi	Host Data Bit 15 (MSB)
2	LD14	Bi	Host Data Bit 14
3	LD13	Bi	Host Data Bit 13
4	LD12	Bi	Host Data Bit 12
5	LD11	Bi	Host Data Bit 11
6	LD10	Bi	Host Data Bit 10
7	LD9	Bi	Host Data Bit 9
8	LD8	Bi	Host Data Bit 8
9	W_IO16~	In	16-bit I/O inserts 1 wait I/O cycle
10	H_BHE~	In1	Host Bus High Enable
11	SA0	Out	Host Address Bit 0 (LSB)
12	SA1	Out	Host Address Bit 1
13	SA2	Out	Host Address Bit 2
14	SA3	Out	Host Address Bit 3
15	SA4	Out	Host Address Bit 4
16	SA5	Out	Host Address Bit 5
17	SA6	Out	Host Address Bit 6
18	SA7	Out	Host Address Bit 7
19	SA8	Out	Host Address Bit 8
20	SA9	Out	Host Address Bit 9
21	SA10	Out	Host Address Bit 10
22	SA11	Out	Host Address Bit 11
23	SA12	Out	Host Address Bit 12
24	SA13	Out	Host Address Bit 13
25	SA14	Out	Host Address Bit 14
26	SA15	Out	Host Address Bit 15 (MSB)
27	LD0	Bi	Host Data Bit 0 (LSB)
28	LD1	Bi	Host Data Bit 1
29	LD2	Bi	Host Data Bit 2
30	LD3	Bi	Host Data Bit 3
31	LD4	Bi	Host Data Bit 4
32	LD5	Bi	Host Data Bit 5
33	LD6	Bi	Host Data Bit 6
34	LD7	Bi	Host Data Bit 7

Connector Location : J17

Pin	Signals	I/O	Definition
1	H_IORD~	Out	IO Read command from the Host
2	H_IOWR~	Out	IO Write command from the host
3	H_NOWS~	In	No wait state
4	H_AEN	Out	Host Address Enable
5	W_CHRDY	OC	Host I/O Channel Ready
6	BD_RST~	Out	Board System Reset
7	SDLEN~	OC	Host Data Low byte buffer enable control
8	SDHEN~	OC	Host Data High byte buffer enable control
9	FM_IRQ	IN	Interrupt Request from FM Synthesizer
10	FM_CS~	Out	FM Synthesizer enable control
11	AC_SOUT	Out	Audio Codec Serial Data Out
12	H_OSC14	Out	14MHz Clock from Host
13	+5V dc	Out	+5Volt for the expansion board
14	AC_SCLK	Out	Audio Codec Serial Clock
15	GND	Out	Ground for the expansion board
16	AC_SYNC	Out	Audio Codec Sync signal
17	AC_SDO	Out	Audio Codec Serial Data In
18	ASCO_SDI	Tri	Audio Codec Serial Data In
19	H_IRQ5	In	Host Interrupt Levle #5
20	H_DRQ1	In	Host DMA request #1
21	H_IRQ7	In	Host Interrupt Level #7
22	H_DACK1~	Out	Host DMA acknowlege #1
23	BIAS_DC	Out	The DC Bias Voltage from Audio Codec
24	H_IRQ2	In	Host Interrupt Level #2
25	FM_OUTL	In	Analog audio Left-input thru FM line
26	FM_OUTR	In	Analog audio Right-input thru FM line

Electrical Specifications

DC Power Input Requirement

Without Digital Camera

+5V dc	1.10A typ
+12V dc	140mA typ
-12V dc	25mA typ

With Digital Camera

+5V dc	1.35A typ
+12V dc	300mA typ
-12V dc	65mA typ

Video Section

Logitech Digital Camera Interface

Vih	2.0Volt Min
Vil	0.8 Volt Max
Current for +5V dc	500ma Max
Current for +12V dc	500ma Max
Current for -12V dc	200ma Max

Video 1 Input

Input Impedance	75 Ohm
Frequency response	0Hz to 4.5M Hz
Nomial Input Level	1 volt peak to Peak

Tuner VideoVideo Tuner Input

Input Impedance	75 Ohm
Frequency response	0Hz to 4.5 MHz
Nomial Input Level	2 volt peak to Peak

Audio Section

External Audio input - Line Level

Input Impedance	12K Ohm
Frequency Responses	20 Hz to 18Hz
Dynamic Range	84 db
Nomial Input Level	1 volt peak to Peak
Maximum input level	9.0 Volt peak to peak

External Audio Input - Mic Level

Input Impedance	12K Ohm
Frequency response	20 Hz to 18K Hz
Nomial Input Level	50 mv peak to Peak
Maximum input level	9.0 volt peak to peak

Tuner Input

Input Impedance	30K Ohm
Frequency response	20 Hz to 18K Hz
Dynamic Range	84 db
Nomial Input Level	1 volt peak to Peak
Maximum input level	9.0 Volt peak to peak

CD analog audio input

Input Impedance	30K Ohm

Frequency response	20 Hz to 18K Hz
Dynamic Range	84 db
Nomial Input Level	1 volt peak to Peak
Maximum input level	9.0 Volt peak to peak

Audio Output

Output Power	5Watt @THD=0.5% 6 Watt @THD=10%

Control-L

Output	TTL Output levels
Current for +5V dc	100ma Max

Video ASIC Functions

Tom Noggle
Original: Jan. 18, 1995
Updated: Feb. 27, 1995

I. Introduction

This document is an attempt to describe the various functional blocks of the Video ASIC used for VideoMan 2.0 Basic and MooseCam.

The various functional blocks are shown in **Figure 1.** Descriptions of the operation of the functional blocks follows in **Section 2.**

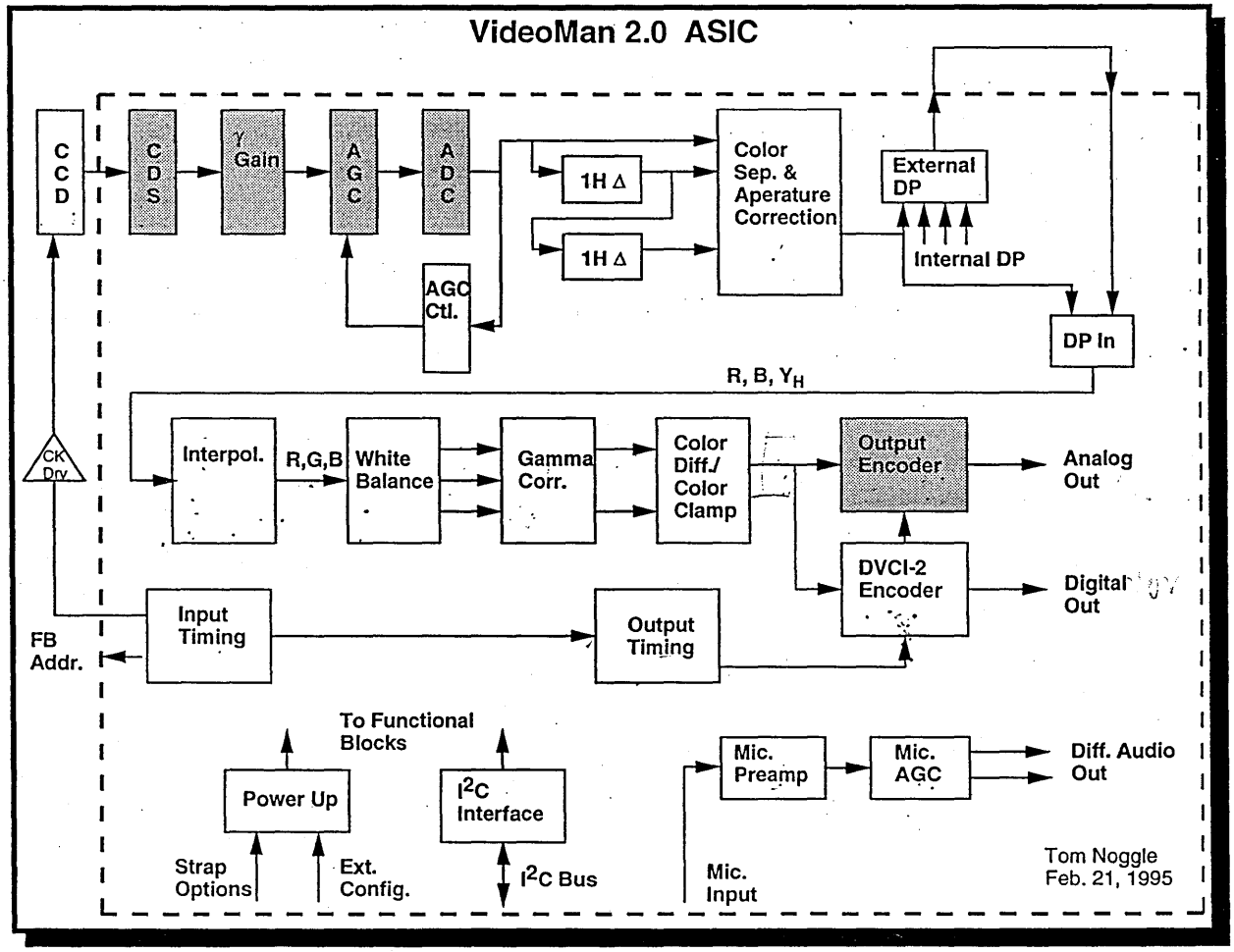


Figure 1
VASIC Functional Block Diagram

II. Functional Blocks

A. Correllated Double Sampling (CDS)

This functional block provides the correllated double sampling of pixels from the CCD. It provides an output analog signal which is the buffered output of signals from the CCD. The CDS block operates by clamping during the inactive portion of the CCD signal (after the Reset clock) on an amplifier and then sampling the active portion of the CCD signal onto a sample/hold to generate the output signal.

B. Non-Linear (Gamma) Gain

A functional block to provide increased dynamic range of the image. It gets its name from the transfer characteristic: the gain function looks like a gamma function ($out = in^{\gamma}$). The specific transfer function of this block and its operation is currently being defined.

C. Automatic Gain Control (AGC)

This analog functional block provides up to 18 dB (x8) gain of the output signal received from the CDS functional block. Gain control is set by digital values into a DAC integral to this functional block. Note that there is an additional fixed 20 DB (x10) gain inside this block to boost the CCD signal of nominally 250 mV to the ADC-required level of 2.5V. (The exact gain required is vendor-specific to the ADC recommended input range).

It should be noted that first order control of gain is by control of the "electronic shutter". The electronic shutter is increased or decreased in increments of one line time to adjust the maximum white level. Only when the electronic shutter time has been increased to the maximum does the analog portion of the gain get involved to raise the overall gain.

D. Analog to Digital Converter (ADC)

The ADC section provides the digitizing element of VASIC. The ADC must convert at rates equal to the maximum pixel rate of 17.025 MHz. The ADC has its zero level clamped to the "Output Black" level of the CCD (photosites that are covered by opaque material) and the white or maximum level is adjusted by the AGC circuitry. A pipelined or "half-flash" conversion of the ADC element is acceptable given the pipelined nature of further video signal processing.

E. AGC Control

The AGC control consists of two functional elements. One implements a servo loop that tries to establish the "Output Black" level from the CCD at a digital value of 10 (0Ah). The other element implements a servo loop that attempts to set a "proper" white level. The algorithms used by these functional elements are currently rather ill-defined. The heuristics are described below but the specific algorithm may well change.

The servo loop that sets the "Output Black" level operates on an eight frame averaging basis (to prevent changing black levels too rapidly). It detects an average minimum luminance level over the eight frames and attempts to set this level to a digital value of 10 (0Ah). It changes the ADC offset DAC to drive the output luminance level to a value of 10 but only attempts to drive the

level up or down when the average minimum luminance level exceeds 10 by +/- 2 counts. In other words, the servo loop exhibits some hysteresis before changing the black level once it has found the proper operating level. This hysteresis is necessary to prevent changing the overall image black level on small shifts in the black level (due to thermal drift or other slow changes), which could severely impact codec performance on compressed video data. This servo loop could be considered to operate in two major states: one state is "black seeking" where it attempts to establish a proper black level, the other is "idle", where the black level has been properly established. During "black seeking", the servo loop changes the offset level more rapidly the further away it is from the correct level.

The servo loop that sets the output white level (ADC gain) is more complicated. A "Region of Interest" (ROI) is established by setting pixel and line values of opposite corners of a rectangular window under host control. (On power up, this ROI is established as the lower half of the full video window). Within this ROI, it builds a limited histogram of pixel values averaged over every eight frames. The histogram is "limited" in the sense that the pixel values are classified into 16 uniform levels and the resulting histogram is examined by a small state machine after being averaged over eight frames. This state machine attempts to satisfy two criteria: first, that a certain percentage (default of 88%) of the pixel values lie below the maximum white level of 235 (0EBh) and that the mean pixel value be above a certain threshold (default 33%). The rationale for these two conditions is that some amount of white pixels should be allowed before the AGC reduces gain, as in the case of a backlit subject. Secondly, the overall image should be biased into the "lighter" area of exposure, rather than darker.

As is the case for the black-seeking AGC control, this algorithm exhibits some hysteresis once a proper gain for the white levels has been reached. In particular, the state machine will only seek for a new white gain level when the conditions described above are not met and exceed +/- 4 counts of the maximum luminance.

External control of this state machine is provided. Under host control, the first and second criteria levels (percentage of pixels below "white" and mean pixel threshold) can be adjusted and the conditions that the state machine try to satisfy these conditions can be switched on and off. Finally, the overall operation of the AGC control for both black-seeking and white-seeking can be enabled or disabled by the host. The host can change the ROI window for the white-seeking AGC function by writing new pixel and line values for the ROI window. When the AGC control operation is disabled by the host, the host can still control the black and white levels by writing to the appropriate AGC control registers directly.

F. Color Separation/Line Delays

This functional block separates the four-color complementary matrix from the CCD into a three color system. The three colors generated are: Red (R), Blue (B), and high frequency luminance (Y_H). In addition, a low frequency luminance (Y_L) is generated for use in gamma correction.

The CCD complementary color matrix provides red and blue information on alternate lines. The horizontal line delays provide delay so that red and blue information can be aligned (cosited) with the luminance information. However, the red and blue information has one-half the spatial frequency of the luminance signal. The line delays also allow for an internal 2D

edge-enhancement operation (the "aperature correction") to compensate for lens effects. The resulting, higher frequency luminance component is Y_H and the luminance component that has the same spatial frequency as the chrominance components is Y_L.

Red and blue are derived from the complementary color matrix signals from the CCD. Red and blue are derived as:

$$R = (<Ye + Mg> - \alpha<Cy + G>)/1.15$$

$$B = (<Cy + Mg> - \beta<Ye + G>)/1.27$$

where $\alpha = 0.71$ and $\beta = 0.66$. These coefficients are adjustable to compensate for the presence of an IR filter and the use of different manufacturer's CCDs with slightly different color filters characteristics over the photodiodes.

After passing through the color separator, the luminance (Y_H) and chrominance components are aligned both spatially and in frequency. The aligned or cosited luminance and chrominance components are used in further processing stages and conversion to a 4:2:2 relationship takes place only at the output.

G. External Data Path Interface

The digital data path is brought out to external pins after the Color Separation operation. As mentioned in the previous paragraphs, the R, B, Y_H, and Y_L signals are available on these pins. If an external frame buffer is used, external logic would be connected to this data path to take video data from the VASIC and place in the frame buffer.

This port has a second purpose. It has a multiplexer attached to various points of the internal data path for improved chip production testing. The specific connection points are still TBD.

H. Data Path In

These inputs connect to further video signal processing logic. The data path is internal if no external frame buffer memory is used, or connects to an external frame buffer if one is used (as would be the case if a pan/zoom engine is connected to the external frame buffer).

I. Interpolator

The interpolator block is used to adjust the number of pixels between the CCD and the output. It implements a 1D polynomial interpolation of the pixels using a cubic spline interpolation.

J. White Balance

This block provides for white balance adjustment (color temperature adjustment) of the image by changing the relative percentages of Red/Blue. The Red/Blue ratio can be changed under control of the host computer. The white balance algorithm implemented is:

$$R' = 2*\alpha R$$

$$B' = 2*(1-\alpha)B$$

for $0.25 < \alpha < 0.75$. α is the control parameter supplied by the host computer.

Automatic white balance is also provided by a small state machine that analyzes the R/B ratio of a ROI of the image and adjusts the R/B ratio. This control function can be disabled by an external host computer and the relative R/B ratios selected can be controlled by an external host. Control over Red and Blue saturation is also provided within this block as a gain function after the white balance operation.

K. Gamma Correction

Gamma correction of Red, Blue, and Y_H is implemented in this functional block. Gamma correction (at present) is implemented via a single, time-multiplexed look-up table. Using a single table for gamma correction provides the most functionality for a given gate count but limits the color components to a common gamma correction value. This limitation appears to be reasonable, given the tradeoff in gate count (and chip cost).

Also being considered is a six-segment piecewise linear approximation for the gamma function. Enough work has been carried out to prove that the errors are quite small. However, a lookup table would be the most direct solution.

L. Color Differencing/Color Clamping

This functional block provides.the translation from red, blue, and luminance components to luminance and chrominance components. In particular, we form:

$U = Y_L - R$ and
$V = Y_L - B$.

For values of luminance approaching "black" or "white", a pixel should contain no chrominance components. Accordingly, the U and V values are decreased as Y_L goes to white or black and are clamped to 0 for Y_L values at or beyond black or white. The color clamping circuitry has the following characteristics:

$0 \le Y_L \le 0Ah$	U',V' = 0
$0Bh \le Y_L \le 01Ah$	U' = ((U - 0Ah)/16)*U
	V' = ((V - 0Ah)/16)*V
$01Bh \le Y_L \le 0DAh$	U', V' = U, V
$0DBh \le Y_L \le 0EAh$	U' = (-1*(U - 0EBh)/16)*U
	V' = (-1*(V - 0EBh)/16)*V
$0EBh \le Y_L \le 0FFh$	U', V' = 0

Note that this functional block does not provide clamping or limiting of the luminance components. Final clamping operations on the luminance channel are deferred to the "DVCI-2 Encoder" block.

M. Output Encoder

The Output Encoder block, in conjunction with the "DVCI-2 Encoder" provides output analog signals compatible with NTSC (RS-170) or PAL formats. The output amplifier has adequate drive to handle the required level of 1.0 volts into 75 ohms. The Output Encoder takes digital values for Y, U, and V that have been clamped and corrected by the DVCI-2 Encoder and translates them to analog levels. In addition, the Output Encoder supplies a color burst of the

correct phase and amplitude, front porch and back porch timings, and provides proper line and field timing for NTSC or PAL video formats.

N. DVCI-2 Encoder

The "DVCI-2 Encoder" provides a digital output data format for the video data. The output data is a derivative of the DVCI specification, hence the name DVCI-2. This functional block provides final Y level clamping of the video signal, reflection of the U and V components about the digital value 128 and clamping as needed, and provides output of blanking values and ancillary data. For further information about the output data format and timings, see the document "**DVCI-2 Interface Specification**".

O. Input Timing

This functional block provides the timing reference for all internal circuitry (except for some portion of the Output Encoder) and the clocking waveforms for the video CCD imager. In addition to providing the clocking waveforms for the CCD, the input timing circuitry receives information from the AGC control for control of the CCD "electronic shutter".

The Input Timing block provides clocking and timing waveforms to support the following CCD imagers at the various field rates given:

Manufacturer	Format	CCD	Field Rate	Clock Rate
Sharp LZ2313H5 Panasonic MN3716MFE	512 x 492	NTSC	60 Hz., 30 Hz.*	9.66 MHz
Sharp LZ2323H4 Panasonic MN3726MFE	512 x 582	PAL	60 Hz., 30 Hz.* 50 Hz., 25 Hz.*	9.66 MHz 11.59 MHz
Sharp LZ2363 Sony ICX059AK	752 x 582	PAL	60 Hz., 30 Hz.,* 50 Hz., 25 Hz.*	17.025 MHz 14.188 MHz

*These field rates are unique to the VASIC design and may operate the CCD outside of manufacturer's recommended operating condition. Communication with the manufacturers listed above has provided the input that the CCDs will operate at other frame rates than the ones specified, but at some loss in sensitivity.

P. Output Timing

The Output Timing block takes the master clock from the Input Timing block and provides additional clock and timing for use by the DVCI-2 Encoder and the Output Encoder. In particular, the Output Timing block provides line and field timing for the operating format and proper blanking times for the selected output format.

Q. Power Up Logic

This functional block provides proper reset of the VASIC on power up and interacts with the I^2C Interface Logic to provide proper initialization of the VASIC to the current strap-selected mode. In addition to the initialization function, this block of logic handles ancillary functions such as shutter position detection, pushbutton detection and LED drive.

R. I^2C Interface

The I^2C Interface Block provides the external interface for the VASIC chip via the I^2C bus. The Interface Block provides the control path into all other functional blocks via an internal control bus. An external chip for handling pan, zoom, and dewarping functions (the so-called "zoom engine") in conjunction with an external frame buffer is resident on this same I^2C bus. The zoom engine chip shows up as additional registers in the I^2C bus and its presence can be detected by the availability of these additional registers. Currently, the register assignments for the VASIC and zoom engine are not known but all I^2C bus addresses are considered reserved.

No provision is made to have multiple cameras on the same I^2C bus--in particular, there is no device addressing capability on the I^2C bus and all addresses on the I^2C bus reserved.

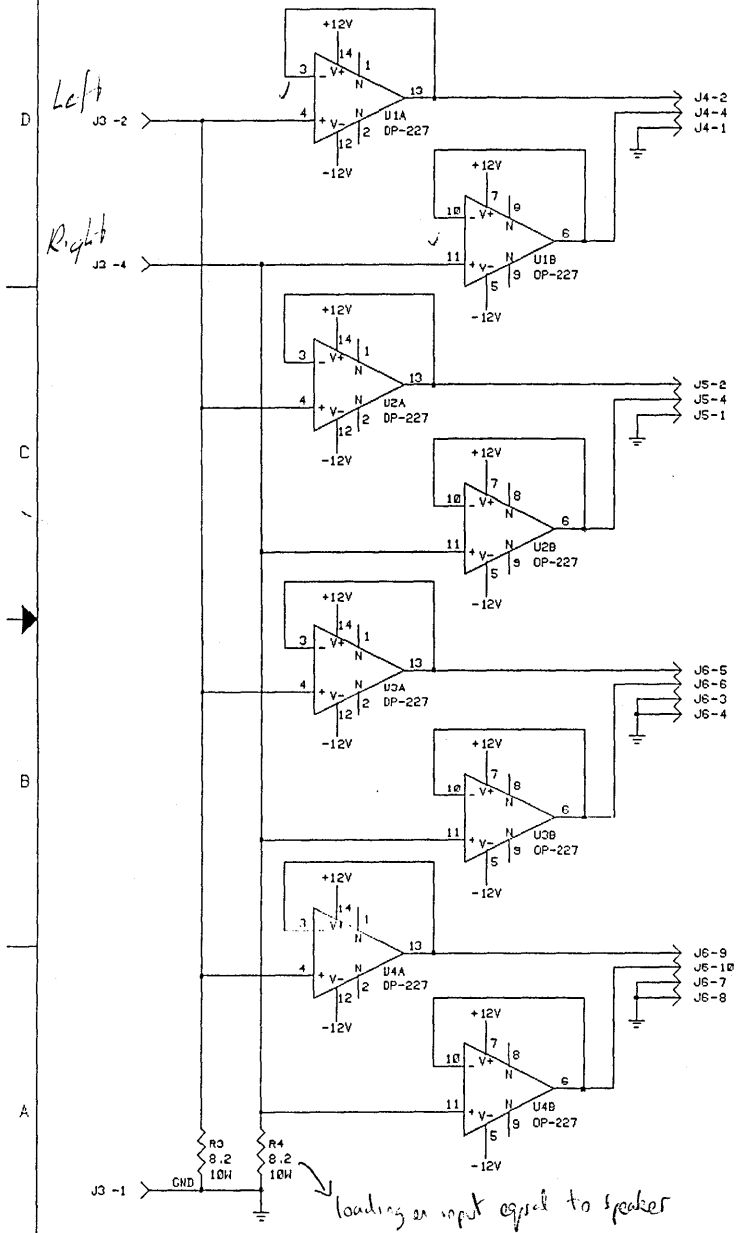
S. Microphone Preamp and AGC

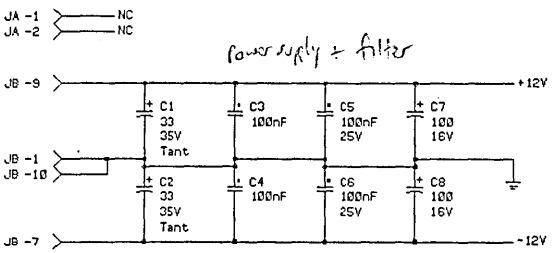
An internal preamplifier is provided for amplification of audio signals from an external microphone. The preamplifier provides initial amplification of the audio signal (20 dB) and an AGC amplifier provides an additional amplification stage. The AGC amplifier provides up to 40 dB of additional gain, depending on the amplification required. The loop filter for the AGC amplifier is provided external to the VASIC chip because of the large values of the time constants expected for audio purposes.

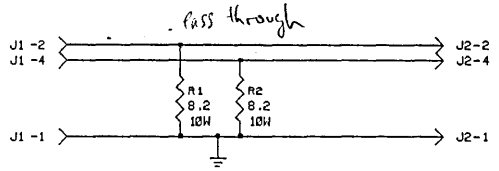
2 Osteres

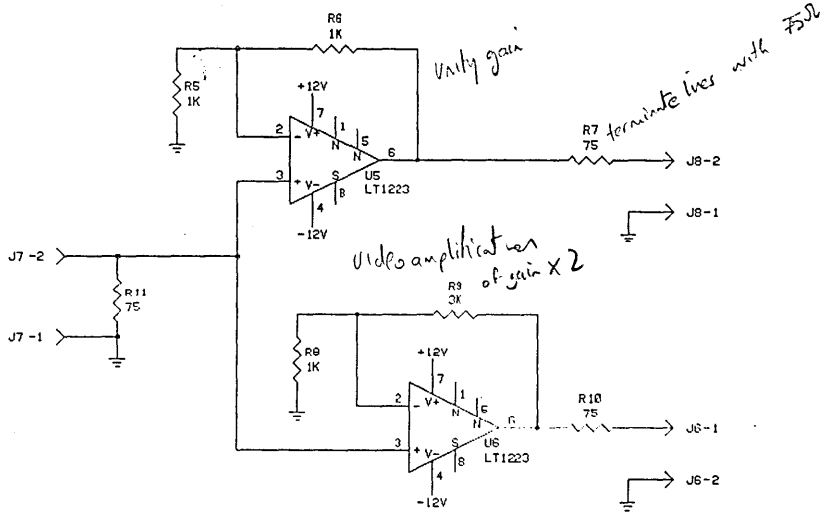
Analog Buffers

REVISIONS				
REV.	ECO NO.	DESCRIPTION	DATE	APPROVED







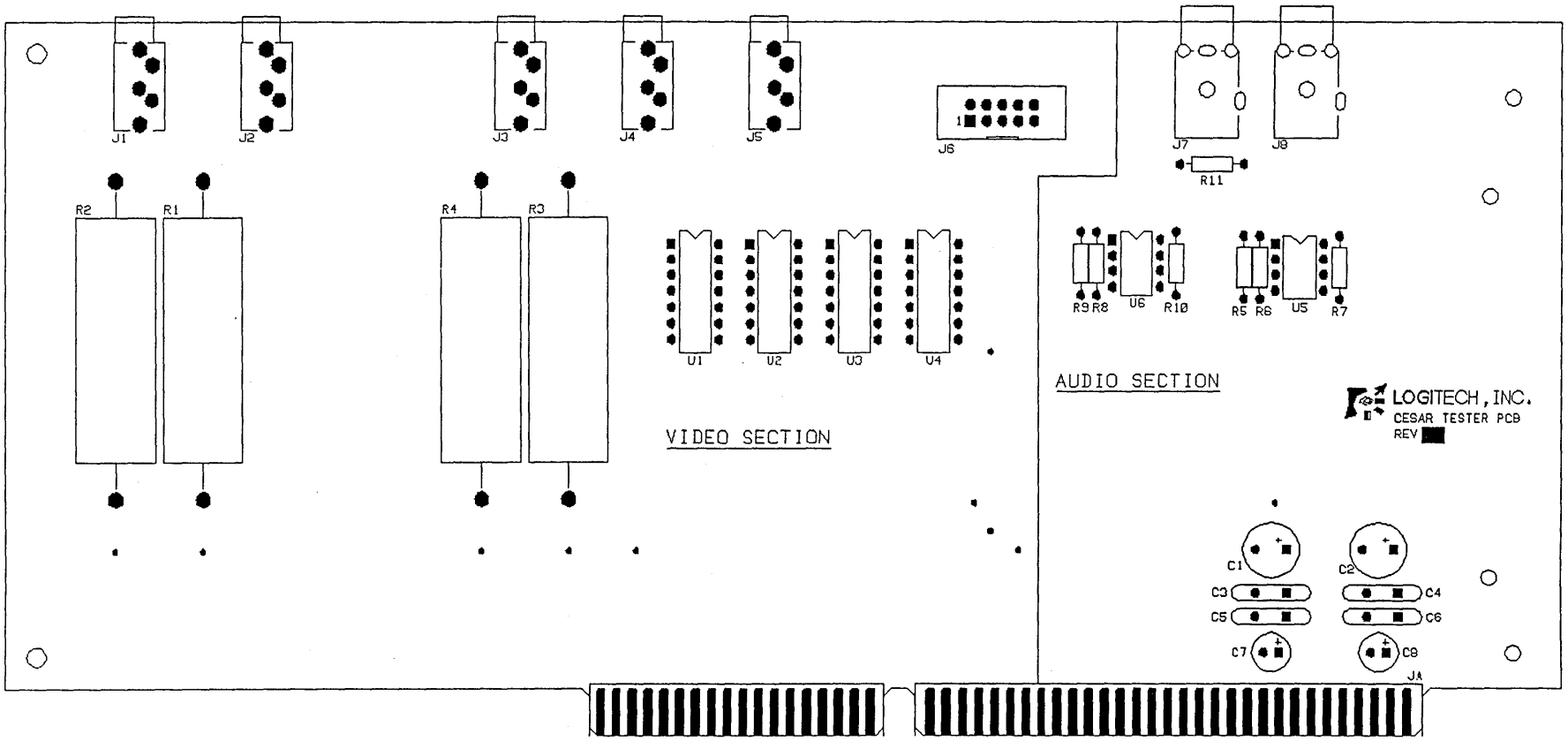


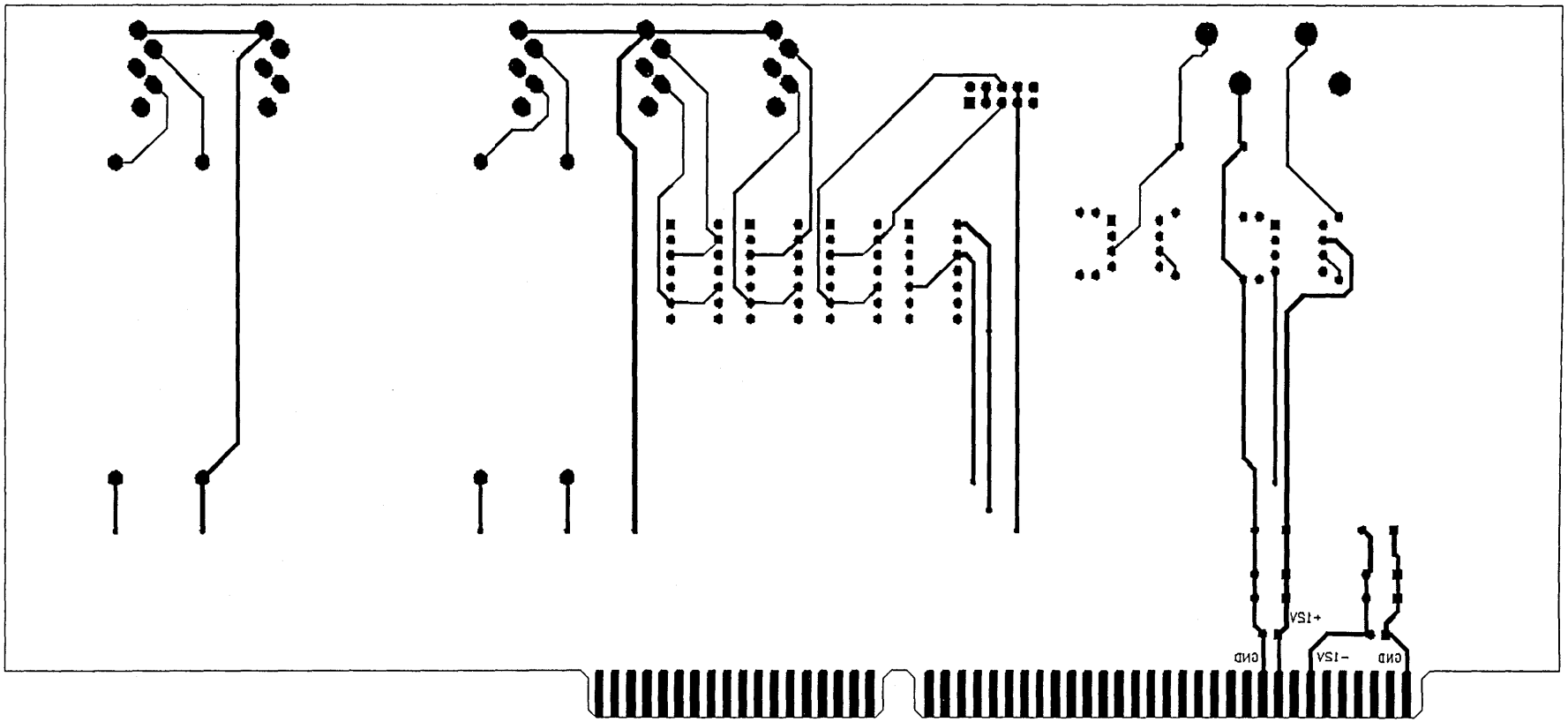
NOTES: (UNLESS OTHERWISE SPECIFIED)

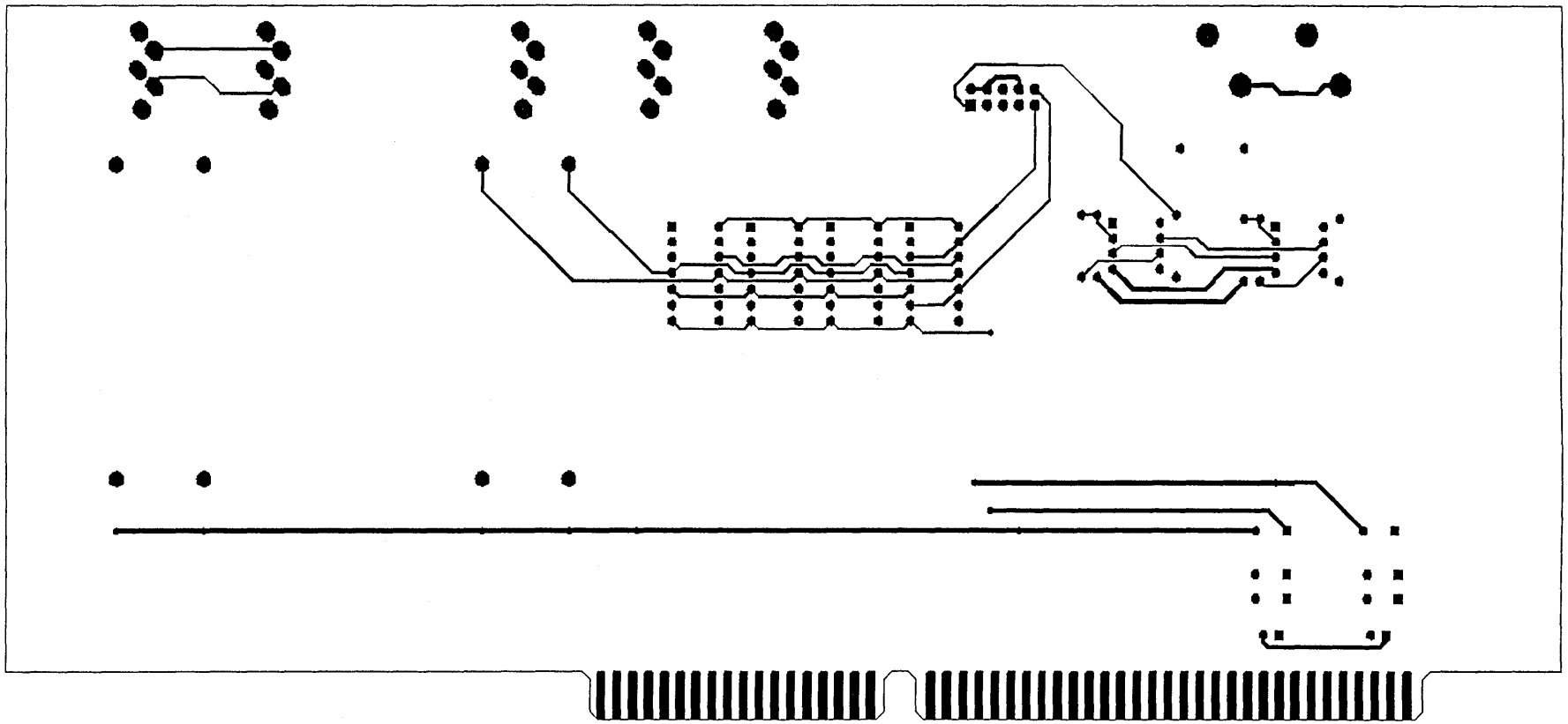
1. ALL CAPACITANCES IN MICRO-FARADS, 50V, 5%.
2. ALL RESISTANCES IN OHMS, 1/8W, 1%.

LOCKTECH

TITLE: CESAR TESTER PCB			
PART:	PWA-200URS-01-USA-PA1		REV PA1
SCHEMA:	SCH-260URS-01-USA-PA1		DESIGNER: URS BOESINGER
PCB:	200URS-01		ENGR. APPROVED:
A/W:	260URS-00		CHECK:
DATE:	23-Aug-94	14:58:01	DRAWN: NAME
FILE: C_TST_SC		CKT. SECT.:	SHEET 1 OF 1


J1
J2
J3
J4
J5
J6
J7
J8
R11
R2
R1
R4
R3
U1
U2
U3
U4
R9 R8
U6
R10
R5 R6
U5
R7
VIDEO SECTION
AUDIO SECTION
LOGITECH, INC.
CESAR TESTER PCB
REV
C1
C2
C3
C4
C5
C6
C7
C8
J9


+15A
GND
-15A
GND



+12V /M

C1	C-33	1
C3	C-100N	1
C5	C100N25V	1
C7	C-100	1
J0165	EDGECONB	9
U1	OP-227	14
U1	OP-227	7
U2	OP-227	14
U2	OP-227	7
U3	OP-227	14
U3	OP-227	7
U4	OP-227	14
U4	OP-227	7
U5	LT1223	7
U6	LT1223	7

-12V /M

C2	C-33	2
C4	C-100N	2
C6	C100N25V	2
C8	C-100	2
J0165	EDGECONB	7
U1	OP-227	12
U1	OP-227	5
U2	OP-227	12
U2	OP-227	5
U3	OP-227	12
U3	OP-227	5
U4	OP-227	12
U4	OP-227	5
U5	LT1223	4
U6	LT1223	4

ANON_44_ /M

J0137	3PSTEREO	4
R4	R-8.2	1
U1	OP-227	11
U2	OP-227	11
U3	OP-227	11
U4	OP-227	11

ANON_46_ /M

J0137	3PSTEREO	2
R3	R-8.2	1
U1	OP-227	4
U2	OP-227	4
U3	OP-227	4
U4	OP-227	4

ANON_48_ /M

J0141	3PSTEREO	4
U1	OP-227	10
U1	OP-227	6

ANON_49_ /M

J0154	10X2_M	10
U4	OP-227	10
U4	OP-227	6

```
ANON_50_ /M
          J0146 3PSTEREO      4
             U2   OP-227      10
             U2   OP-227       6

ANON_51_ /M
          J0141 3PSTEREO      2
             U1   OP-227      13
             U1   OP-227       3

ANON_52_ /M
          J0146 3PSTEREO      2
             U2   OP-227      13
             U2   OP-227       3

ANON_53_ /M
          J0154   10X2_M       5
             U3   OP-227      13
             U3   OP-227       3

ANON_54_ /M
          J0154   10X2_M       6
             U3   OP-227      10
             U3   OP-227       6

ANON_55_ /M
          J0154   10X2_M       9
             U4   OP-227      13
             U4   OP-227       3

ANON_56_ /M
          J0172 3PSTEREO      2
          J0175 3PSTEREO      2
             R1    R-8.2       1

ANON_57_ /M
          J0172 3PSTEREO      4
          J0175 3PSTEREO      4
             R2    R-8.2       1

ANON_59_ /M
          J0177 2PSTEREO      2
            R11     R-75       1
             U5   LT1223       3
             U6   LT1223       3

ANON_60_ /M
             R6     R-1K       2
             R7     R-75       1
             U5   LT1223       6

ANON_61_ /M
             R5     R-1K       1
             R6     R-1K       1
             U5   LT1223       2

ANON_62_ /M
          J0184 2PSTEREO      2
             R7     R-75       2
```

ANON_63_ /M

R10	R-75	1
R9	R-3K	2
U6	LT1223	6

ANON_64_ /M

R8	R-1K	1
R9	R-3K	1
U6	LT1223	2

ANON_65_ /M

J0154	10X2_M	1
R10	R-75	2

GND /M

C1	C-33	2
C2	C-33	1
C3	C-100N	2
C4	C-100N	1
C5	C100N25V	2
C6	C100N25V	1
C7	C-100	2
C8	C-100	1
J0137	3PSTEREO	1
J0141	3PSTEREO	1
J0146	3PSTEREO	1
J0154	10X2_M	2
J0154	10X2_M	3
J0154	10X2_M	4
J0154	10X2_M	7
J0154	10X2_M	8
J0165	EDGECONB	10
J0165	EDGECONB	1
J0172	3PSTEREO	1
J0175	3PSTEREO	1
J0177	2PSTEREO	1
J0184	2PSTEREO	1
R11	R-75	2
R1	R-8.2	2
R2	R-8.2	2
R3	R-8.2	2
R4	R-8.2	2
R5	R-1K	2
R8	R-1K	2

JC

J0193	EDGECONA	1
J0193	EDGECONA	2
U1	OP-227	1
U1	OP-227	2
U1	OP-227	8
U1	OP-227	9
U2	OP-227	1
U2	OP-227	2
U2	OP-227	8
U2	OP-227	9
U3	OP-227	1
U3	OP-227	2
U3	OP-227	8
U3	OP-227	9

U4	OP-227	1
U4	OP-227	2
U4	OP-227	8
U4	OP-227	9
U5	LT1223	1
U5	LT1223	5
U5	LT1223	8
U6	LT1223	1
U6	LT1223	5
U6	LT1223	8

```
/****************************************************************************
 *
 *   captest.h: Include file for the CapTest Sample Program
 *
 *   Microsoft Video for Windows Capture Class Sample Program
 *
 ****************************************************************************/
/****************************************************************************
 *
 *  THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
 *  KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
 *  IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
 *  PURPOSE.
 *
 *  Copyright (c) 1992 - 1995  Microsoft Corporation.  All Rights Reserved.
 *
 ****************************************************************************/


#include "dialogs.h"

//
// General purpose constants...
//
#define MAXVIDDRIVERS            10
#define MS_FOR_15FPS             66
#define DEF_PALNUMFRAMES         10
#define DEF_PALNUMCOLORS         236L
#define ONEMEG                   (1024L * 1024L)

//
// Menu Ids...
//
#define IDM_FILE                  100
#define IDM_F_SETCAPTUREFILE      101
#define IDM_F_SAVEVIDEOAS         102
#define IDM_F_ALLOCATESPACE       103
#define IDM_F_EXIT                104

#define IDM_EDIT                  110
#define IDM_E_COPY                111
#define IDM_E_PASTEPALETTE        112

#define IDM_CAPTURE               120
#define IDM_C_CAPTUREVIDEO        121
#define IDM_C_CAPTUREFRAME        122
#define IDM_C_CAPTURESETTINGS     123

#define IDM_OPTIONS               130
#define IDM_O_PREVIEW             131
#define IDM_O_OVERLAY             132
#define IDM_O_AUDIOFORMAT         133
#define IDM_O_VIDEOFORMAT         134
#define IDM_O_VIDEOSOURCE         135
#define IDM_O_VIDEODISPLAY        136
#define IDM_O_PALETTE             137
#define IDM_O_CHOOSECOMPRESSOR    140

// !!! Leave some room after this define for the drivers
#define IDM_O_DRIVERS             150
#define IDM_O_DRIVERS1            151
#define IDM_O_DRIVERS2            152
#define IDM_O_DRIVERS3            153
#define IDM_O_DRIVERS4            154
#define IDM_O_DRIVERS5            155
#define IDM_O_DRIVERS6            156
#define IDM_O_DRIVERS7            157
#define IDM_O_DRIVERS8            158
#define IDM_O_DRIVERS9            159
// The above are just reserved dummy IDs.

#define IDM_HELP                  190
#define IDM_H_ABOUT               191

#define IDS_ERR_REGISTER_CLASS    1
```



```
#define IDS_ERR_CREATE_WINDOW     2


//
// Macro Definitions...
//
#define IsDriverIndex(w) ( ((w) >= IDM_O_DRIVERS)  &&  \
                           ((w) - IDM_O_DRIVERS < MAXVIDDRIVERS) )

#define Width(X)  ((X).right - (X).left)
#define Height(X) ((X).bottom - (X).top)


//- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - ;
//
//  Win 16/32 portability stuff...
//
//
//
//- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - ;

#ifndef WIN32
    #ifndef FNLOCAL
        #define FNLOCAL     NEAR PASCAL
        #define FNCLOCAL    NEAR _cdecl
        #define FNGLOBAL    FAR PASCAL
        #define FNCGLOBAL   FAR _cdecl
    #ifdef _WINDLL
        #define FNWCALLBACK FAR PASCAL __loadds
        #define FNEXPORT    FAR PASCAL __export
    #else
        #define FNWCALLBACK FAR PASCAL __loadds
        #define FNEXPORT    FAR PASCAL __export
    #endif
    #endif

    //
    //
    //
    //
    #ifndef FIELD_OFFSET
    #define FIELD_OFFSET(type, field)    ((LONG)&(((type *)0)->field))
    #endif

    //
    //  based code makes since only in win 16 (to try and keep stuff out of
    //  our fixed data segment...
    //
    #define BCODE           _based(_segname("_CODE"))

    #define HUGE            _huge

    //
    //  stuff for Unicode in Win 32--make it a noop in Win 16
    //
    #ifndef _TCHAR_DEFINED
        #define _TCHAR_DEFINED
        typedef char            TCHAR, *PTCHAR;
        typedef unsigned char   TBYTE, *PTUCHAR;

        typedef PSTR            PTSTR, PTCH;
        typedef LPSTR           LPTSTR, LPTCH;
        typedef LPCSTR          LPCTSTR;
    #endif

    #define TEXT(a)         a
    #define SIZEOF(x)       sizeof(x)
    #define SIZEOFACMSTR(x) sizeof(x)
#else
    #ifndef FNLOCAL
        #define FNLOCAL     _stdcall
        #define FNCLOCAL    _stdcall
        #define FNGLOBAL    _stdcall
        #define FNCGLOBAL   _stdcall
        #define FNWCALLBACK CALLBACK
```

```
/****************************************************************************
 *
 *  THIS CODE AND INFORMATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY
 *  KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
 *  IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR
 *  PURPOSE.
 *
 *  Copyright (c) 1992 - 1995  Microsoft Corporation.  All Rights Reserved.
 *
 ****************************************************************************/
/****************************************************************************
 *
 *   captest.c: Source Code for the CapTest Sample Program
 *
 *   Microsoft Video for Windows Capture Class Sample Program
 *
 ****************************************************************************/


#define ENABLE_ERROR_CALLBACK           1
#define ENABLE_STATUS_CALLBACK          1
#define ENABLE_VIDEOFRAME_CALLBACKS     0

#define INC_OLE2
#include <windows.h>
#include <windowsx.h>
#include <commdlg.h>
#include <vfw.h>
#include <mmreg.h>
#include <io.h>
#include <fcntl.h>
#include <stdio.h>
#include <stdlib.h>
#include <memory.h>
#include <dos.h>

#include "captest.h"

//
// Global Variables
//
TCHAR           gachAppName[]  = TEXT("CapTestApp") ;
TCHAR           gachIconName[] = TEXT("CapTestIcon") ;
TCHAR           gachMenuName[] = TEXT("CapTestMenu") ;
TCHAR           gachMCIDeviceName[21] = TEXT("VideoDisc") ;  // default MCI device
TCHAR           gachString[128] ;
TCHAR           gachBuffer[200] ;

HINSTANCE       ghInstApp ;
HWND            ghWndMain ;
HWND            ghWndCap ;
HANDLE          ghAccel ;
WORD            gwDeviceIndex ;
WORD            gwPalFrames = DEF_PALNUMFRAMES ;
WORD            gwPalColors = DEF_PALNUMCOLORS ;
WORD            gwCapFileSize ;
DWORD           gdwFrameNum ;
DWORD           gdwVideoNum ;

CAPSTATUS       gCapStatus ;
CAPDRIVERCAPS   gCapDriverCaps ;
CAPTUREPARMS    gCapParms ;

LPWAVEFORMATEX glpwfex ;

// MakeProcInstance is only required for 16-bit apps
#ifndef WIN32
 FARPROC        fpErrorCallback;
 FARPROC        fpStatusCallback;
 FARPROC        fpFrameCallback;
 FARPROC        fpVideoCallback;
#endif

// Function prototypes
//
```



```
LONG FAR PASCAL MainWndProc(HWND, UINT, UINT, LONG) ;
LRESULT FNWCALLBACK ErrorCallbackProc(HWND, int, LPTSTR) ;
LRESULT FNWCALLBACK StatusCallbackProc(HWND, int, LPTSTR) ;
LRESULT FNWCALLBACK FrameCallbackProc(HWND, LPVIDEOHDR) ;
LRESULT FNWCALLBACK VideoCallbackProc(HWND, LPVIDEOHDR) ;

//
// WinMain: Application Entry Point Function
//
int PASCAL WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpszCmdLine, int nCmdShow)
{
////////////////////////////////////////////////////////////////////////
//  hInstance:      handle for this instance
//  hPrevInstance:  handle for possible previous instances
//  lpszCmdLine:    long pointer to exec command line
//  nCmdShow:       Show code for main window display
////////////////////////////////////////////////////////////////////////

    MSG         msg ;
    WNDCLASS    wc ;

    ghInstApp = hInstance ;
    if (! hPrevInstance) {
        // If it's the first instance, register the window class
        wc.lpszClassName = gachAppName ;
        wc.hInstance     = hInstance ;
        wc.lpfnWndProc   = MainWndProc ;
        wc.hCursor       = LoadCursor(NULL, IDC_ARROW) ;
        wc.hIcon         = LoadIcon(hInstance, gachIconName) ;
        wc.lpszMenuName  = gachMenuName ;
        wc.hbrBackground = GetStockObject(WHITE_BRUSH) ;
        wc.style         = CS_HREDRAW | CS_VREDRAW ;
        wc.cbClsExtra    = 0 ;
        wc.cbWndExtra    = 0 ;

        if (! RegisterClass(&wc)) {
            LoadString(ghInstApp, IDS_ERR_REGISTER_CLASS, gachString, sizeof(gachString)/sizeof(T
CHAR)) ;
            MessageBox(NULL, gachString, NULL,
#ifdef BIDI
                MB_RTL_READING |
#endif

            MB_ICONEXCLAMATION) ;
            return 0 ;
        }
    }

    // Create Application's Main window
    ghWndMain =
#ifdef BIDI

//
// unfortunately you can't just #ifdef the CreateWindow line and leave
// the parameters common: on Win32, CreateWindow is a macro and does not
// expand correctly if you ifdef only the 'CreateWindow(' line.
//
        CreateWindowEx(WS_EX_BIDI_SCROLL |  WS_EX_BIDI_MENU |WS_EX_BIDI_NOICON,
                               gachAppName,
                               TEXT("Capture Test App"),
                               WS_CAPTION      |
                               WS_SYSMENU      |
                               WS_MINIMIZEBOX  |
                               WS_MAXIMIZEBOX  |
                               WS_THICKFRAME   |
                               WS_CLIPCHILDREN |
                               WS_OVERLAPPED,
                               CW_USEDEFAULT, 0,
                               320, 240,
                               NULL,
                               NULL,
                               ghInstApp,
                               NULL) ;
#else
        CreateWindow (
```

```
        #define FNEXPORT    CALLBACK
    #endif

    #ifndef _CRTAPI1
    #define _CRTAPI1    __cdecl
    #endif
    #ifndef _CRTAPI2
    #define _CRTAPI2    __cdecl
    #endif
    #ifndef try
    #define try         __try
    #define leave       __leave
    #define except      __except
    #define finally     __finally
    #endif


    //
    //  there is no reason to have based stuff in win 32
    //
    #define BCODE

    #define HUGE
    #define HTASK                   HANDLE
    #define SELECTOROF(a)           (a)
    typedef LRESULT (CALLBACK* DRIVERPROC)(DWORD, HDRVR, UINT, LPARAM, LPARAM);

    //
    //  for compiling Unicode
    //
    #ifdef UNICODE
        #define SIZEOF(x)   (sizeof(x)/sizeof(WCHAR))
    #else
        #define SIZEOF(x)   sizeof(x)
    #endif
    #define SIZEOFACMSTR(x) (sizeof(x)/sizeof(WCHAR))
#endif


//
// Global Variables...
//
extern TCHAR           gachAppName[] ;
extern TCHAR           gachIconName[] ;
extern TCHAR           gachMenuName[] ;
extern TCHAR           gachString[] ;
extern TCHAR           gachMCIDeviceName[] ;

extern HINSTANCE       ghInstApp ;
extern HWND            ghWndMain ;
extern HWND            ghWndCap ;
extern HANDLE          ghAccel ;
extern WORD            gwDeviceIndex ;
extern WORD            gwPalFrames ;
extern WORD            gwPalColors ;
extern WORD            gwCapFileSize ;

extern CAPSTATUS       gCapStatus ;
extern CAPDRIVERCAPS   gCapDriverCaps ;
extern CAPTUREPARMS    gCapParms ;

extern LPWAVEFORMATEX glpwfex ;

//
// Dialog Box Procedures...
//
int FAR PASCAL AboutProc(HWND, UINT, UINT, LONG) ;
int FAR PASCAL AudioFormatProc(HWND, UINT, UINT, LONG) ;
int FAR PASCAL CapSetUpProc(HWND, UINT, UINT, LONG) ;
BOOL CALLBACK MakePaletteProc(HWND, UINT, UINT, LONG) ;
int FAR PASCAL AllocCapFileProc(HWND, UINT, UINT, LONG) ;
```

```
                            gachAppName,
                            TEXT("Capture Test App"),
                            WS_CAPTION      |
                            WS_SYSMENU      |
                            WS_MINIMIZEBOX  |
                            WS_MAXIMIZEBOX  |
                            WS_THICKFRAME   |
                            WS_CLIPCHILDREN |
                            WS_OVERLAPPED,
                            CW_USEDEFAULT, 0,
                            320, 240,
                            NULL,
                            NULL,
                            ghInstApp,
                            NULL) ;
#endif

    if (ghWndMain == NULL) {
        LoadString(ghInstApp, IDS_ERR_CREATE_WINDOW, gachString, sizeof(gachString)/sizeof(TCHAR)
) ;
        MessageBox(NULL, gachString, NULL,
#ifdef BIDI
                MB_RTL_READING |
#endif

        MB_ICONEXCLAMATION | MB_OK) ;
        return IDS_ERR_CREATE_WINDOW ;
    }

    ShowWindow(ghWndMain, nCmdShow) ;
    UpdateWindow(ghWndMain) ;
    ghAccel = LoadAccelerators(ghInstApp, gachAppName) ;

    // All set; get and process messages
    while (GetMessage(&msg, NULL, 0, 0)) {
        if (! TranslateAccelerator(ghWndMain, ghAccel, &msg)) {
            TranslateMessage(&msg) ;
            DispatchMessage(&msg) ;
        }
    }

    return msg.wParam ;
}  // End of WinMain


//
// ErrorCallbackProc: Error Callback Function
//
LRESULT FNWCALLBACK ErrorCallbackProc(HWND hWnd, int nErrID, LPTSTR lpErrorText)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:          Application main window handle
//  nErrID:        Error code for the encountered error
//  lpErrorText:   Error text string for the encountered error
////////////////////////////////////////////////////////////////////////

    if (!ghWndMain)
        return FALSE;

    if (nErrID == 0)            // Starting a new major function
        return TRUE;            // Clear out old errors...

    // Show the error ID and text
    wsprintf(gachBuffer, TEXT("Error# %d"), nErrID) ;

    MessageBox(hWnd, lpErrorText, gachBuffer,
#ifdef BIDI
                MB_RTL_READING |
#endif
                MB_OK | MB_ICONEXCLAMATION) ;

    return (LRESULT) TRUE ;
}
```



```
//
// StatusCallbackProc: Status Callback Function
//
LRESULT FNWCALLBACK StatusCallbackProc(HWND hWnd, int nID, LPTSTR lpStatusText)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:          Application main window handle
//  nID:           Status code for the current status
//  lpStatusText:  Status text string for the current status
////////////////////////////////////////////////////////////////////////

    if (!ghWndMain)
        return FALSE;

    if (nID == 0) {             // Zero means clear old status messages
        SetWindowText(ghWndMain, (LPTSTR) gachAppName) ;
        return (LRESULT) TRUE ;
    }

    // Show the status ID and status text...
    wsprintf(gachBuffer, TEXT("Status# %d: %s"), nID, lpStatusText) ;

        SetWindowText(ghWndMain, (LPTSTR)gachBuffer) ;

    return (LRESULT) TRUE ;
}


//
// FrameCallbackProc: Frame Callback Function
// Called whenever a new frame is captured but not streaming
//
LRESULT FNWCALLBACK FrameCallbackProc(HWND hWnd, LPVIDEOHDR lpVHdr)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:       Application main window handle
//  lpVHdr:     long pointer to VideoHdr struct containing captured
//              frame information
////////////////////////////////////////////////////////////////////////

    if (!ghWndMain)
        return FALSE;

    wsprintf(gachBuffer, TEXT("Preview frame# %ld "), gdwFrameNum++) ;

    SetWindowText(ghWndMain, (LPTSTR)gachBuffer) ;
    return (LRESULT) TRUE ;
}


//
// VideoCallbackProc: Video Stream Callback Function
// Called whenever a new frame is captured while streaming
//
LRESULT FNWCALLBACK VideoCallbackProc(HWND hWnd, LPVIDEOHDR lpVHdr)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:       Application main window handle
//  lpVHdr:     long pointer to VideoHdr struct containing captured
//              frame information
////////////////////////////////////////////////////////////////////////

    gdwVideoNum++;      // Testing:  just count the callbacks

    return (LRESULT) TRUE ;
}


//
// CenterCaptureWindow: Placess Capture Window at the Center of Main Window
//
static void CenterCaptureWindow(HWND hWndM, HWND hWndC)
{
////////////////////////////////////////////////////////////////////////
//  hWndM:      Application main window handle
```

```
//  hWndC:      Capture window handle
////////////////////////////////////////////////////////////////////////

    RECT       MainRect ;
    RECT       CapRect ;
    WORD       wCapXPos ;
    WORD       wCapYPos ;

    // Get the sizes of main and capture windows and
    // calculate the location for centering
    GetClientRect(hWndM, &MainRect) ;
    GetClientRect(hWndC, &CapRect) ;
    wCapXPos = max(0, (Width(MainRect) - Width(CapRect)) / 2) ;
    wCapYPos = max(0, (Height(MainRect) - Height(CapRect)) / 2) ;

    // Position the capture window at the required location
    MoveWindow(hWndC, wCapXPos, wCapYPos, Width(CapRect),
               Height(CapRect), TRUE) ;
}


//
// StartNewVideoChannel: Gets Selected Driver's Caps -- Updates menu,
//                       Checks Image Size -- Resizes display window,
//                       Enables Preview (at 15 FPS rate)
//
static void StartNewVideoChannel(HWND hWndM, HWND hWndC, WORD wIndex)
{
////////////////////////////////////////////////////////////////////////
//  hWndM:      Application main window handle
//  hWndC:      Capture window handle
//  wIndex:     Selected capture driver index
////////////////////////////////////////////////////////////////////////

    HMENU      hMenu = GetMenu(hWndM) ;

    // Get capture driver settings and update menu
    capDriverGetCaps(hWndC, &gCapDriverCaps, sizeof(CAPDRIVERCAPS)) ;
    EnableMenuItem(hMenu, IDM_O_OVERLAY, MF_BYCOMMAND |
               gCapDriverCaps.fHasOverlay ? MF_ENABLED : MF_GRAYED) ;
    EnableMenuItem(hMenu, IDM_O_VIDEOFORMAT, MF_BYCOMMAND |
               gCapDriverCaps.fHasDlgVideoFormat ? MF_ENABLED : MF_GRAYED) ;
    EnableMenuItem(hMenu, IDM_O_VIDEOSOURCE, MF_BYCOMMAND |
               gCapDriverCaps.fHasDlgVideoSource ? MF_ENABLED : MF_GRAYED) ;
    EnableMenuItem(hMenu, IDM_O_VIDEODISPLAY, MF_BYCOMMAND |
               gCapDriverCaps.fHasDlgVideoDisplay ? MF_ENABLED : MF_GRAYED) ;

    // Get video format and adjust capture window
    capGetStatus(hWndC, &gCapStatus, sizeof(CAPSTATUS)) ;
    SetWindowPos(hWndC, NULL, 0, 0, gCapStatus.uiImageWidth,
                 gCapStatus.uiImageHeight, SWP_NOZORDER | SWP_NOMOVE) ;

    // Start preview by default
    capPreviewRate(hWndC, MS_FOR_15FPS) ;
    capPreview(hWndC, TRUE) ;

    // Put check mark beside appropriate menu options
    CheckMenuItem(hMenu, wIndex + IDM_O_DRIVERS, MF_BYCOMMAND | MF_CHECKED) ;

    // Set Audio format menu appropriately
    EnableMenuItem(hMenu, IDM_O_AUDIOFORMAT,
                 gCapStatus.fAudioHardware ? MF_ENABLED : MF_GRAYED);
}


//
// MenuProc: Processes All Menu-based Operations
//
long FAR PASCAL MenuProc(HWND hWnd, UINT wParam, LONG lParam)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:       Application main window handle
//  hMenu:      Application menu handle
//  wParam:     Menu option
//  lParam:     Additional info for any menu option
```



```
////////////////////////////////////////////////////////////////////////

    OPENFILENAME ofn ;
    DWORD        dwError ;
    WORD         wIndex ;
    BOOL         fResult ;
    DWORD        dwSize ;
    TCHAR        achBuffer[_MAX_PATH] ;
    TCHAR        achFileName[_MAX_PATH] ;
    TCHAR        *szFileFilter = TEXT("Microsoft AVI\0")
                                 TEXT("*.avi\0")
                                 TEXT("All Files\0")
                                 TEXT("*.*\0") ;

    HMENU hMenu = GetMenu(hWnd) ;

    switch (wParam) {
        case IDM_F_SETCAPTUREFILE:
        {
            LPTSTR p;

            // Get current capture file name and
            // then try to get the new capture file name
            dwError = capFileGetCaptureFile(ghWndCap, achFileName,
                                            sizeof(achFileName)/sizeof(TCHAR));
            if (dwError) {

                // Get just the path info
                // Terminate the full path at the last backslash
                lstrcpy (achBuffer, achFileName);
                for (p = achBuffer + lstrlen(achBuffer); p > achBuffer; p--) {
                    if (*p == '\\') {
                        *(p+1) = '\0';
                        break;
                    }
                }

                _fmemset(&ofn, 0, sizeof(OPENFILENAME)) ;
                ofn.lStructSize = sizeof(OPENFILENAME) ;
                ofn.hwndOwner = hWnd ;
                ofn.lpstrFilter = szFileFilter ;
                ofn.nFilterIndex = 0 ;
                ofn.lpstrFile = achFileName ;
                ofn.nMaxFile = sizeof(achFileName)/sizeof(TCHAR) ;
                ofn.lpstrFileTitle = NULL;
                ofn.lpstrTitle = TEXT("Set Capture File") ;
                ofn.nMaxFileTitle = 0 ;
                ofn.lpstrInitialDir = achBuffer;
                ofn.Flags =
#ifdef BIDI
                OFN_BIDIDIALOG |
#endif
                OFN_HIDEREADONLY |
                OFN_NOREADONLYRETURN |
                OFN_PATHMUSTEXIST ;

                if (GetOpenFileName(&ofn))
                    // If the user has hit OK then set capture file name
                    capFileSetCaptureFile(ghWndCap, achFileName) ;
            }
        }
        break;

        case IDM_F_SAVEVIDEOAS:
            // Get the current capture file name and
            // then get the substitute file name to save video in
            dwError = capFileGetCaptureFile(ghWndCap, achFileName, sizeof(achFileName)/sizeof(TCH
AR));
            if (dwError) {

                _fmemset(&ofn, 0, sizeof(OPENFILENAME)) ;
                ofn.lStructSize = sizeof(OPENFILENAME) ;
                ofn.hwndOwner = hWnd ;
                ofn.lpstrFilter = szFileFilter ;
                ofn.nFilterIndex = 0 ;
```

```
                ofn.lpstrFile = achFileName ;
                ofn.nMaxFile = sizeof(achFileName)/sizeof(TCHAR) ;
                ofn.lpstrFileTitle = NULL ;
                ofn.lpstrTitle = TEXT("Save Video As...") ;
                ofn.nMaxFileTitle = 0 ;
                ofn.lpstrInitialDir = NULL ;
                ofn.Flags =
#ifdef BIDI
                OFN_BIDIDIALOG |
#endif
                OFN_PATHMUSTEXIST ;

                if (GetSaveFileName(&ofn))
                    // If the user has hit OK then set save file name
                    capFileSaveAs(ghWndCap, achFileName) ;
            }
            break;

        case IDM_F_ALLOCATESPACE:
            if (DialogBox(ghInstApp, MAKEINTRESOURCE(IDD_AllocCapFileSpace),
                          hWnd, AllocCapFileProc))
                // If user has hit OK then alloc requested capture file space
                if (! capFileAlloc(ghWndCap, (long) gwCapFileSize * ONEMEG))
                    MessageBox(NULL, TEXT("Can't pre-allocate capture file space"),
                               TEXT("Error"),
#ifdef BIDI
                               MB_RTL_READING |
#endif
                               MB_OK | MB_ICONEXCLAMATION) ;
            break ;

        case IDM_F_EXIT:
            DestroyWindow(hWnd) ;
            break;

        case IDM_E_COPY:
            capEditCopy(ghWndCap) ;
            break;

        case IDM_E_PASTEPALETTE:
            capPalettePaste(ghWndCap) ;
            break;

        case IDM_O_PREVIEW:
            // Toggle Preview
            capGetStatus(ghWndCap, &gCapStatus, sizeof(CAPSTATUS)) ;
            capPreview(ghWndCap, !gCapStatus.fLiveWindow) ;
            break;

        case IDM_O_OVERLAY:
            // Toggle Overlay
            capGetStatus(ghWndCap, &gCapStatus, sizeof(CAPSTATUS)) ;
            capOverlay(ghWndCap, !gCapStatus.fOverlayWindow) ;
            break ;

        case IDM_O_AUDIOFORMAT:
#ifdef  USE_ACM
            {
                ACMFORMATCHOOSE cfmt;

                // Ask the ACM what the largest wave format is.....
                acmMetrics(NULL,
                           ACM_METRIC_MAX_SIZE_FORMAT,
                           &dwSize);

                // Get the current audio format
                dwSize = max (dwSize, capGetAudioFormatSize (ghWndCap));
                glpwfex = (LPWAVEFORMATEX) GlobalAllocPtr(GHND, dwSize) ;
                capGetAudioFormat(ghWndCap, glpwfex, (UINT)dwSize) ;

                _fmemset (&cfmt, 0, sizeof (ACMFORMATCHOOSE));
                cfmt.cbStruct = sizeof (ACMFORMATCHOOSE);
                cfmt.fdwStyle =  ACMFORMATCHOOSE_STYLEF_INITTOWFXSTRUCT;
                cfmt.fdwEnum =   ACM_FORMATENUMF_HARDWARE |
                                 ACM_FORMATENUMF_INPUT;
```



```
                cfmt.hwndOwner = hWnd;
                cfmt.pwfx =     glpwfex;
                cfmt.cbwfx =    dwSize;
                if (!acmFormatChoose(&cfmt))
                    capSetAudioFormat(ghWndCap, glpwfex, (UINT)dwSize) ;

                GlobalFreePtr(glpwfex) ;
            }
#else
            // If not using ACM, remove the reference in the link line
            // of makefile.

            // Get current audio format and then find required format
            dwSize = capGetAudioFormatSize (ghWndCap);
            glpwfex = (LPWAVEFORMATEX) GlobalAllocPtr(GHND, dwSize) ;
            capGetAudioFormat(ghWndCap, glpwfex, (UINT)dwSize) ;

            if (DialogBox(ghInstApp, MAKEINTRESOURCE(IDD_AudioFormat), hWnd, AudioFormatProc))
                capSetAudioFormat(ghWndCap, glpwfex, (UINT)dwSize);  // If the user has hit OK, s
et the new audio format

            GlobalFreePtr(glpwfex) ;
#endif
            break ;

        case IDM_O_VIDEOFORMAT:
            if (gCapDriverCaps.fHasDlgVideoFormat) {
                // Only if the driver has a "Video Format" dialog box
                if (capDlgVideoFormat(ghWndCap)) {  // If successful,
                    // Get the new image dimension and center capture window
                    capGetStatus(ghWndCap, &gCapStatus, sizeof(CAPSTATUS)) ;
                    SetWindowPos(ghWndCap, NULL, 0, 0, gCapStatus.uiImageWidth,
                        gCapStatus.uiImageHeight, SWP_NOZORDER | SWP_NOMOVE) ;
                    CenterCaptureWindow(hWnd, ghWndCap) ;
                }
            }
            break;

        case IDM_O_VIDEOSOURCE:
            if (gCapDriverCaps.fHasDlgVideoSource) {
                // Only if the driver has a "Video Source" dialog box
                capDlgVideoSource(ghWndCap) ;
            }
            break ;

        case IDM_O_VIDEODISPLAY:
            if (gCapDriverCaps.fHasDlgVideoDisplay) {
                // Only if the driver has a "Video Display" dialog box
                capDlgVideoDisplay(ghWndCap) ;
            }
            break ;

        case IDM_O_PALETTE:
            if (DialogBox(ghInstApp, MAKEINTRESOURCE(IDD_MakePalette), hWnd, MakePaletteProc))
                // If the user has hit OK, capture palette with the
                // specified number of colors and frames
                capPaletteAuto(ghWndCap, gwPalFrames, gwPalColors) ;
            break;

        case IDM_C_CAPTUREVIDEO:
            gdwVideoNum = 0 ;  // Start counting video frames
            // Capture video sequence
            fResult = capCaptureSequence(ghWndCap) ;
            break;

        case IDM_C_CAPTUREFRAME:
            gdwFrameNum = 0 ;  // Start counting single frames
            // Turn off overlay / preview (gets turned off by frame capture)
        capPreview(ghWndCap, FALSE);
        capOverlay(ghWndCap, FALSE);

            // Grab a frame
            fResult = capGrabFrameNoStop(ghWndCap) ;
            break;
```

```
        case IDM_C_CAPTURESETTINGS:
            // Get the current setup for video capture
            capCaptureGetSetup(ghWndCap, &gCapParms, sizeof(CAPTUREPARMS)) ;

            // Invoke a Dlg box to setup all the params
            if (DialogBox(ghInstApp, MAKEINTRESOURCE(IDD_CapSetUp), hWnd, CapSetUpProc))
                // If the user has hit OK, set the new setup info
                capCaptureSetSetup(ghWndCap, &gCapParms, sizeof(CAPTUREPARMS)) ;
            break;

        case IDM_O_CHOOSECOMPRESSOR:
            capDlgVideoCompression(ghWndCap);
            break;

        case IDM_H_ABOUT:
            DialogBox(ghInstApp, MAKEINTRESOURCE(IDD_HelpAboutBox), hWnd, AboutProc) ;
            break ;

        default:
            // There is a chance, a driver change has been requested
            if ( IsDriverIndex(wParam) ) {
                // If it's a valid driver index...
                if (wParam - IDM_O_DRIVERS != gwDeviceIndex) {
                    // and a different one too then we need to do the rest

                    // Turn off preview/overlay, uncheck current driver option
                    capPreview(ghWndCap, FALSE) ;
                    capOverlay(ghWndCap, FALSE) ;
                    CheckMenuItem(GetMenu(hWnd), gwDeviceIndex + IDM_O_DRIVERS,
                                  MF_BYCOMMAND | MF_UNCHECKED) ;

                    // Connect to requested driver
                    if ( capDriverConnect(ghWndCap, (wIndex = (UINT) (wParam - IDM_O_DRIVERS))) )
{
                        // Connect worked fine -- update menu, start new driver...
                        CheckMenuItem(GetMenu(hWnd), wParam, MF_BYCOMMAND | MF_CHECKED) ;
                        gwDeviceIndex = (UINT) (wParam - IDM_O_DRIVERS) ;
                        StartNewVideoChannel(hWnd, ghWndCap, gwDeviceIndex) ;
                        CenterCaptureWindow(hWnd, ghWndCap) ;
                    }
                    else {
                        // if connect failed, re-connect back to previous driver
                        if (! capDriverConnect(ghWndCap, gwDeviceIndex)) {
                            MessageBox(hWnd, TEXT("Now can't connect back to previous driver !!")
,
                                       TEXT("Error"),
#ifdef BIDI
                            MB_RTL_READING |
#endif

                            MB_OK | MB_ICONSTOP) ;
                            return -1L ;
                        }
                        else
                            // Re-start previous driver as it was before
                            StartNewVideoChannel(hWnd, ghWndCap, gwDeviceIndex) ;
                            CenterCaptureWindow(hWnd, ghWndCap) ;
                    }
                }   // end of if ( != gwDeviceIndex)
            }   // end of if (IsDriverIndex())
            else {
                wsprintf(achBuffer, TEXT("How could you specify this (%u) Driver Index ?"),
                         wParam - IDM_O_DRIVERS) ;
                MessageBox(hWnd, achBuffer, TEXT("Oops!!"),
#ifdef BIDI
                MB_RTL_READING |
#endif
                MB_OK | MB_ICONEXCLAMATION) ;
            }

            break ;
    }

    return 0L ;
}
```



```
//
// MainWndProc: Application Main Window Procedure
//
LONG FAR PASCAL MainWndProc(HWND hWnd, UINT Message, UINT wParam, LONG lParam)
{
////////////////////////////////////////////////////////////////////////
//  hWnd:       Application main window handle
//  Message:    Next message to be processed
//  wParam:     WORD param for the message
//  lParam:     LONG param for the message
////////////////////////////////////////////////////////////////////////

    switch (Message) {
        case WM_COMMAND:
            MenuProc(hWnd, wParam, lParam) ;
            break ;

        case WM_CREATE:
        {
            TCHAR    achDeviceName[80] ;
            TCHAR    achDeviceVersion[100] ;
            TCHAR    achBuffer[100] ;
            WORD    wDriverCount = 0 ;
            WORD    wIndex ;
            DWORD   dwError ;
            HMENU   hMenu ;

            // First create the capture window
            ghWndCap = capCreateCaptureWindow((LPTSTR)TEXT("Capture Window"),
                                              WS_CHILD | WS_VISIBLE,
                                              0, 0, 160, 120,
                                              (HWND) hWnd, (int) 0) ;

            hMenu = GetSubMenu(GetMenu(hWnd), 2) ;  // 2 for "Option"

#if ENABLE_ERROR_CALLBACK
  #ifdef WIN32
            // Register the status and error callbacks before driver connect
            capSetCallbackOnError(ghWndCap, ErrorCallbackProc) ;
  #else
            fpErrorCallback = MakeProcInstance((FARPROC)ErrorCallbackProc, ghInstApp) ;
            capSetCallbackOnError(ghWndCap, fpErrorCallback) ;
  #endif
#endif

#if ENABLE_STATUS_CALLBACK
  #ifdef WIN32
            capSetCallbackOnStatus(ghWndCap, StatusCallbackProc) ;
  #else
            fpStatusCallback = MakeProcInstance((FARPROC)StatusCallbackProc, ghInstApp) ;
            capSetCallbackOnStatus(ghWndCap, fpStatusCallback) ;
  #endif
#endif

#if ENABLE_VIDEOFRAME_CALLBACKS
  #ifdef WIN32
            capSetCallbackOnVideoStream(ghWndCap, VideoCallbackProc) ;
            capSetCallbackOnFrame(ghWndCap, FrameCallbackProc) ;
  #else
            fpVideoCallback = MakeProcInstance((FARPROC)VideoCallbackProc, ghInstApp) ;
            capSetCallbackOnVideoStream(ghWndCap, fpVideoCallback) ;

            fpFrameCallback = MakeProcInstance((FARPROC)FrameCallbackProc, ghInstApp) ;
            capSetCallbackOnFrame(ghWndCap, fpFrameCallback) ;
  #endif
#endif
            // Try to connect one of the MSVIDEO drivers
            for (wIndex = 0 ; wIndex < MAXVIDDRIVERS ; wIndex++) {
                if (capGetDriverDescription(wIndex,
                            (LPTSTR)achDeviceName, sizeof(achDeviceName)/ sizeof(TCHAR),
                            (LPTSTR)achDeviceVersion, sizeof(achDeviceVersion)/sizeof(TCHAR))) {

                    // There is such a driver in the "system.ini" file.
```

```
                    // Append driver name to "Options" list in menu
                    wsprintf(achBuffer, TEXT("&%d %s"), wIndex, (LPTSTR)achDeviceName) ;
                    AppendMenu(hMenu, MF_ENABLED, IDM_O_DRIVERS+wIndex, achBuffer) ;

                    if (wDriverCount++ == 0) {
                        // Only if no other driver is already connected
                        dwError = capDriverConnect(ghWndCap, wIndex);
                        if (dwError) {
                            CheckMenuItem(GetMenu(hWnd), IDM_O_DRIVERS+wIndex, MF_BYCOMMAND | MF_
CHECKED) ;

                            gwDeviceIndex = wIndex ;
                        }
                    }
                } // end of if (capGetDriverDesc..())
            }

            // Now refresh menu, position capture window, start driver etc
            DrawMenuBar(hWnd) ;
            CenterCaptureWindow(hWnd, ghWndCap) ;
            StartNewVideoChannel(hWnd, ghWndCap, gwDeviceIndex) ;

            break ;
        }

        case WM_MOVE:
        case WM_SIZE:
            CenterCaptureWindow(hWnd, ghWndCap) ;
            break ;

        case WM_PALETTECHANGED:
        case WM_QUERYNEWPALETTE:
            // Pass the buck to Capture window proc
            PostMessage(ghWndCap, Message, wParam, lParam) ;
            break ;

        case WM_INITMENU:
        {
            BOOL          fResult ;

            // Initially check if "Options.PastePalette" should be enabled
            fResult = IsClipboardFormatAvailable(CF_PALETTE) ?
                      MF_ENABLED : MF_GRAYED ;
            EnableMenuItem((HMENU) wParam, IDM_E_PASTEPALETTE, fResult) ;

        // Check/Uncheck Preview and Overlay
            capGetStatus(ghWndCap, &gCapStatus, sizeof(CAPSTATUS)) ;
            CheckMenuItem((HMENU)wParam, IDM_O_PREVIEW, gCapStatus.fLiveWindow
                    ? MF_CHECKED : MF_UNCHECKED);
            CheckMenuItem((HMENU)wParam, IDM_O_OVERLAY,gCapStatus.fOverlayWindow
                    ? MF_CHECKED : MF_UNCHECKED);
        }

        case WM_PAINT:
        {
            HDC           hDC ;
            PAINTSTRUCT   ps ;

            hDC = BeginPaint(hWnd, &ps) ;

            // Included in case the background is not a pure color
            SetBkMode(hDC, TRANSPARENT) ;

            EndPaint(hWnd, &ps) ;
            break ;
        }

        case WM_CLOSE:
            // Disable and free all the callbacks
#if ENABLE_ERROR_CALLBACK
            capSetCallbackOnError(ghWndCap, NULL) ;
  #ifndef WIN32
            FreeProcInstance(fpErrorCallback) ;
  #endif
#endif
```



```
#if ENABLE_STATUS_CALLBACK
            capSetCallbackOnStatus(ghWndCap, NULL) ;
  #ifndef WIN32
            FreeProcInstance(fpStatusCallback) ;
  #endif
#endif

#if ENABLE_VIDEOFRAME_CALLBACKS
            capSetCallbackOnFrame(ghWndCap, NULL) ;
            capSetCallbackOnVideoStream(ghWndCap, NULL) ;
  #ifndef WIN32
            FreeProcInstance(fpFrameCallback) ;
            FreeProcInstance(fpVideoCallback) ;
  #endif
#endif
            // Destroy child windows, modeless dialogs, then this window...

            DestroyWindow(ghWndCap) ;
            DestroyWindow(hWnd) ;
            break ;

        case WM_DESTROY:
            PostQuitMessage(0) ;
            break ;

        default:
            return DefWindowProc(hWnd, Message, wParam, lParam) ;
    }

    return 0L;
}   // End of MainWndProc
```

To: Aidan Kehoe
cc:
From: Jay Feldis/CA/US/LOGITECH
Date: 04/16/96 04:42:38 PM
Subject: New Connector Pin Assignment

To: Stephane Broquere
cc: Jay Feldis, ahlgrim, atsushi, Steven_Manuel, bertk
From: bertk @ 4front.esd.sgi.com ("Bert Keely") @ SMTP
Date: 04/16/96 04:34:08 PM
Subject: New Connector Pin Assignment

MOOSECAM 68 POSITION CONNECTOR PINOUT

PIN#	SIGNAL		PAIR		PIN#	SIGNAL	PAIR
1	+5VOUT		10		35	n/c	
2	I2C_SCL		11		36	I2C_SDA	12
3	GPIIN_P		10		37	n/c	
4	n/c			38	n/c		
5	n/c			39	MIC_P	13	
6	n/c			40	MIC_N	13	
7	n/c			41	n/c		
8	n/c			42	n/c		
9	n/c			43	n/c		
10	n/c			44	n/c		
11	IN_GND[7]	9		45	n/c		
12	IN_P[7]	9		46	n/c		
13	IN_GND[6]	8		47	n/c		
14	IN_P[6]	8		48	n/c		
15	IN_GND[5]	7		49	n/c		
16	IN_P[5]	7		50	n/c		
17	IN_GND[4]	6		51	n/c		
18	IN_P[4]	6		52	n/c		
19	IN_GND[3]	5		53	n/c		
20	IN_P[3]	5		54	n/c		
21	IN_GND[2]	4		55	n/c		
22	IN_P[2] 4			56	n/c		
23	IN_GND[1]	3		57	n/c		
24	IN_P[1]	3		58	n/c		
25	IN_GND[0]	2		59	n/c		
26	IN_P[0]	2		60	n/c		
27	IN_P[CLK]	1		61	n/c		
28	IN_GND[CLK]	1		62	n/c		
29	n/c			63	n/c		
30	-12VOUT		11		64	+12VOUT	12
31	n/c			65	n/c		
32	n/c			66	n/c		
33	n/c			67	n/c		
34	n/c			68	n/c		

Yes, there are an incredible number of no-connects! They're for bidirectional

10-bit video and audio, which the system must support through this connector in professional video applications. So the camera cost raises... sure hope we sell lots in the professional video market!

Thx for your patience.

--

BertK

bertk@sgi.com (415)933-3024 fax(415)390-6159

I2C CONTROL PROTOCOL
FOR I/O EXPANDER

1.0 I/O Expander address

The I/O Expander address is 0x70 for writes and 0x71 for reads.

0	1	1	1	A2	A1	A0	R/!W

A[2..0] = 0 (Grounded)

1.1 Read/Writes Formats

Bit 0 (LSB) is used as the data direction bit.

- It is 0 for writes (address 0x70)
- It is 1 for reads (address 0x71)

1.2 Peripheral Map

P7	P6	P5	P4	*P3*	*P2*	*P1*	*P0*
Not Assigned	Position Switch	!LED	Push Button	*Not Assigned*	*Not Assigned*	*Not Assigned*	*Not Assigned*

- P6(Position Switch) and P4 (Push Button) are read datas.
- P5 (LED) is a read/write data.

Note :

- PO is the LSB.
- !LED means LED active low.
- For Not assigned datas value can be either 1 or 0 (don't care).

To: Aidan Kehoe
cc:
From: William Liu/TW/AP/LOGITECH
Date: 03/12/96 11:29:25 AM ZE8
Subject: Re:VM1.2 Failures

Hello, Aidan,
There are four test stations in Zeye operation line : ICT , Alightment , Function and Final test .
The yield rates and failure phenomena in March are as following:

	Alightment	Function	Final
3/2	Yield=92.1% no-image=>15ea	yield=97.2% unflat CCD=>2ea black&white image=> 1ea fuzzy image=>2ea	Yield=88% lens & mount dismatching=>26ea no-image=>1ea vibrant image=>5ea
3/5	Yield=96.3% no-image=>6ea	Yield=98% no-image=>1ea fuzzy image=>2ea	Yield=79.6% audio failed=>1ea lens & mount dismatching=>29ea vibrant image=>8ea
3/7	Yield=89% no-image=>11ea	Yield=89% out-of-criteria=>10 ea	Yield=81.7% lens & mount dismatching=>10ea vibrant image=>3ea
3/8	Yield=97.6% no-image=3ea	Yield=94.3% out-of-criteria=>7ea	Yield=74.1% lens & mount dismatching=> 39ea vibrant image=> 18ea solder problem of CCD =>2ea

Note :1 . Lens & mount dismatching : It means that lens and mount did not match , one of them is too small or too big to match the other one ,because both of them has tolerance .
2. Vibrant image : At final test , we adjust the focus from the farest to nearest distance , the color chart image on the monitor vibrate around the image center . Some of them are also caused by lens & mount dismatching .

We face two main problem , the one is lens & mount dismatching problem ,it cause yield rate always low . Could it be improved by desing?
The other one is metal case touch the parts or through hole , it happens white image. Althogh we used tafelon tapes to improve that , we can not guarantee the tape won't lose forever.

Regards,
William

HighVoltage, *MidVoltage*, *LowVoltage* - three levels of power setting. (e.g. 0.5V should be entered as 0.5)
Current - current setting. (e.g. 0.5Ampere should be entered as 500E-3)
CurrentLimit - current reading. (e.g. 0.5Ampere should be entered as 500E-3)

[Log]
Define log level for different output devices.
LogFile - define log file name.
Printer - (not in use)

[Time delay in Msec]
Time delay in milliseconds after setting certain registers or after performing certain tasks.

[Light Control]
Auto - { "*on*", "*off*" }. "*off*" switches to manual light control. "*on*" by default.
LPT - { *1*, *2*, *3* }. LPT=1 uses the address stored at 0040:0008, LPT=2 the address at 0040:000A, LPT3=3 the address at 0040:000C. While 0040:0008 is reserved for CAMKIT3.

[Miscellaneous]
CCDAdjustment - { "*on*", "*off*" }. on=enabled, off=disabled. To dedicate a system for CCD adjustment, you may disable all the subtests in the first section and set this switch on.
PPS - { "*on*", "*off*" }. *PPS=on* is to use the programmable power supply. *PPS=off* is to use the power adapter and no power test will be performed.

Functional Test Parameter Table:
(as a guideline of what to test and which variables are involved)

Test Item	GN	BB	RB	BR	SH	BS	RS	AG (Autogain)	AW (Autowhite)	VI	SG (Gammafactor)	GC (Gain Test)	LT (Light)	PW	0	1	2
Daylight Light Test	60	/	/	/	20	80	80	off	on	off	0	off	D	?	x	x	x
Incandescent Light Test	60	D	D	/	10	80	80	off	on	off	0	off	I	?	x	x	x
Incandescent Brightness	L	/	/	/	30	80	80	on	off	off	0	on	I	?	x	x	x
Gamma Factor 1.0 Test	18	/	/	/	40	80	80	off	off	off	0	off	D	?	/	/	x
Gamma Factor 3.0 Test	18	/	/	D	40	80	80	off	off	off	b3	off	D	?	/	/	x
Saturation maximum	18	/	/	/	40	ff	ff	off	off	on	0	off	D	?	x	x	x
Saturation minimum	18	/	/	/	40	0	0	off	off	on	0	off	D	?	x	x	x

AG = Auto Gain Control
AW = Auto White Balance
BB = Blue Balance [D=Delta, /=don'tcare]
BR = Brightness [D=Delta, /=don'tcare]
BS = Blue Saturation
GC = Gain Circuit Test
GN = Gain Value [L=Limit]
LT = Light [C=CoolWhite, D=Daylight, H=Horizon, I=Incandescent, O=Off, U=Ultraviolet]
PW = Power Test [?=undetermined]
RB = Red Balance [D=Delta, /=don'tcare]
RS = Red Saturation
SG = Gamma Factor
SH = Shutter Speed
VI = Visual Inspection

To: aidan kehoe
cc:
From: tonyr @ oaktech.com @ SMTP
Date: 02/22/96 12:25:00 PM
Subject: Re: Question on Oak I2C DLL

At this time my manager does not want to create special versions of our routines. But after looking at the code here is a work around that you can use.

Actually the read function will work the way you want if you make the sub-address 0(zero). Again the sub-address will not be sent if the sub-address is 0.

The write function does not work this way, it always sends the sub-address. But, there is no difference between the way the sub-addresses is handled from the data. So if you have one byte of data to send, put the data in the sub-address and set the byte count to 0. If you have n bytes of data, put the first byte in the sub-address and subsequent bytes in the data array, and set the byte count to n-1.

Be sure to comment your special usage of these functions so a week from now you remember what you did.

I hope this helps,
Tom Cahill

_____ Reply Separator _____
Subject: Question on Oak I2C DLL
Author: TonyR at Oak2po
Date: 2/22/96 6:46 AM

Hello Tony,
I have a question on the Oak I2C DLL. Could you please pass this mail on to Tom Cahill.
Thanks, Aidan

Hello Tom,

I have been using the Oak I2C DLL and everything is working well.

Now one of the OEM customers for our video camera has requested a modified version of the camera. The modification consists of some additional I/O that needs to be controlled via I2C from the host PC.

The additional I/O (for some leds and switches) will be added via an I2C I/O expander. The I/O is independant of the camera functions and will be at a different I2C address. This will not conform to the "station address" plus "register" addressing already used.

Is is possible to add a simplified I2C read/write bytes function that does not

Normal State → everything high

I2C CONTROL PROTOCOL FOR I/O EXPANDER

1.0 I/O Expander address

The I/O Expander address is 0x70 for writes and 0x71 for reads.

0	1	1	1	A2	A1	A0	R/!W

A[2..0] = 0 (Grounded)

1.1 Read/Writes Formats

Bit 0 (LSB) is used as the data direction bit.

- It is 0 for writes (address 0x70)
- It is 1 for reads (address 0x71)

1.2 Peripheral Map

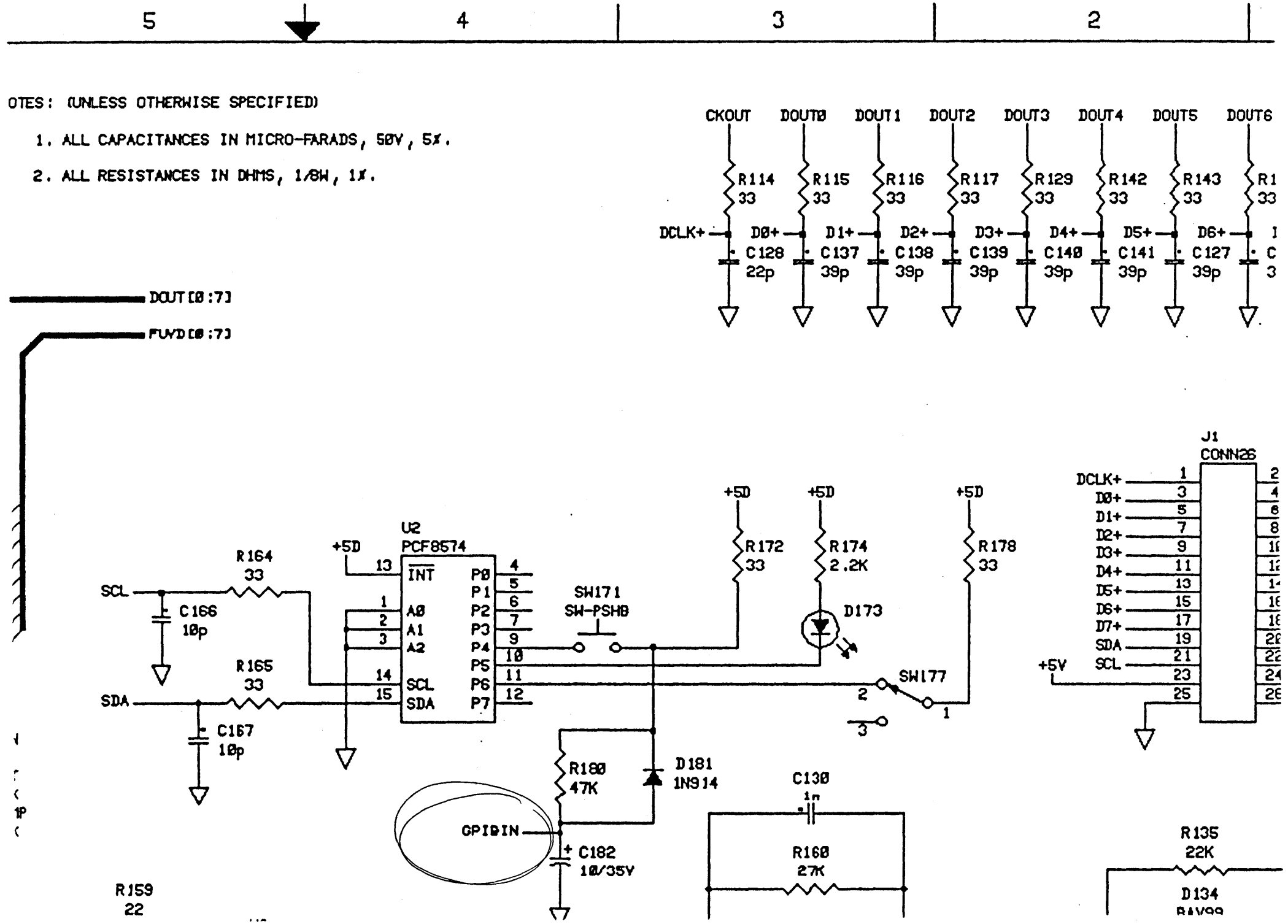
P7	P6	P5	P4	*P3*	*P2*	*P1*	*P0*
Not Assigned	Position Switch	!LED	Push Button	*Not Assigned*	*Not Assigned*	*Not Assigned*	*Not Assigned*

- P6(Position Switch) and P4 (Push Button) are read datas.
- P5 (LED) is a read/write data.

Note :

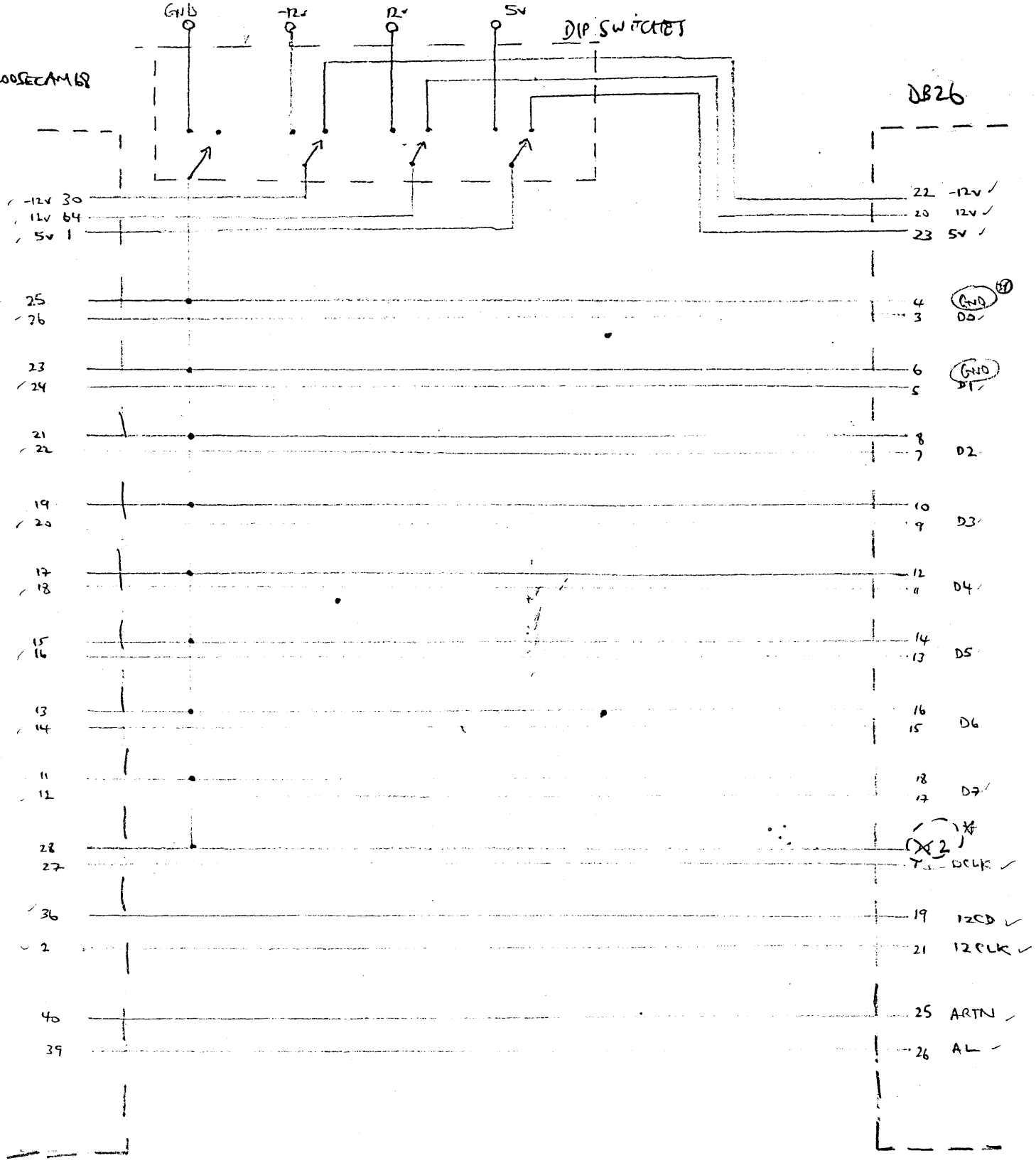
- PO is the LSB.
- !LED means LED active low.
- For Not assigned datas value can be either 1 or 0 (don't care).

PIN 24 is used for the


5
4
3
2
OTES: (UNLESS OTHERWISE SPECIFIED)
1. ALL CAPACITANCES IN MICRO-FARADS, 50V, 5%.
2. ALL RESISTANCES IN OHMS, 1/8W, 1%.
CKOUT
DOUT0
DOUT1
DOUT2
DOUT3
DOUT4
DOUT5
DOUT6
R114 33
R115 33
R116 33
R117 33
R129 33
R142 33
R143 33
DCLK+
D0+
D1+
D2+
D3+
D4+
D5+
D6+
C128 22p
C137 39p
C138 39p
C139 39p
C140 39p
C141 39p
C127 39p
DOUT[0:7]
FUYD[0:7]
U2
PCF8574
INT
A0
A1
A2
SCL
SDA
P0
P1
P2
P3
P4
P5
P6
P7
+5D
R164 33
R165 33
C166 10p
C167 10p
SW171
SW-PSHB
R172 33
R174 2.2K
R178 33
D173
SW177
J1
CONN26
+5V
R180 47K
D181 1N914
GPIBIN
C182 10/35V
C130 1n
R160 27K
R159 22
R135 22K
D134

MOOSECAM 68 POSITION CONNECTOR PINOUT

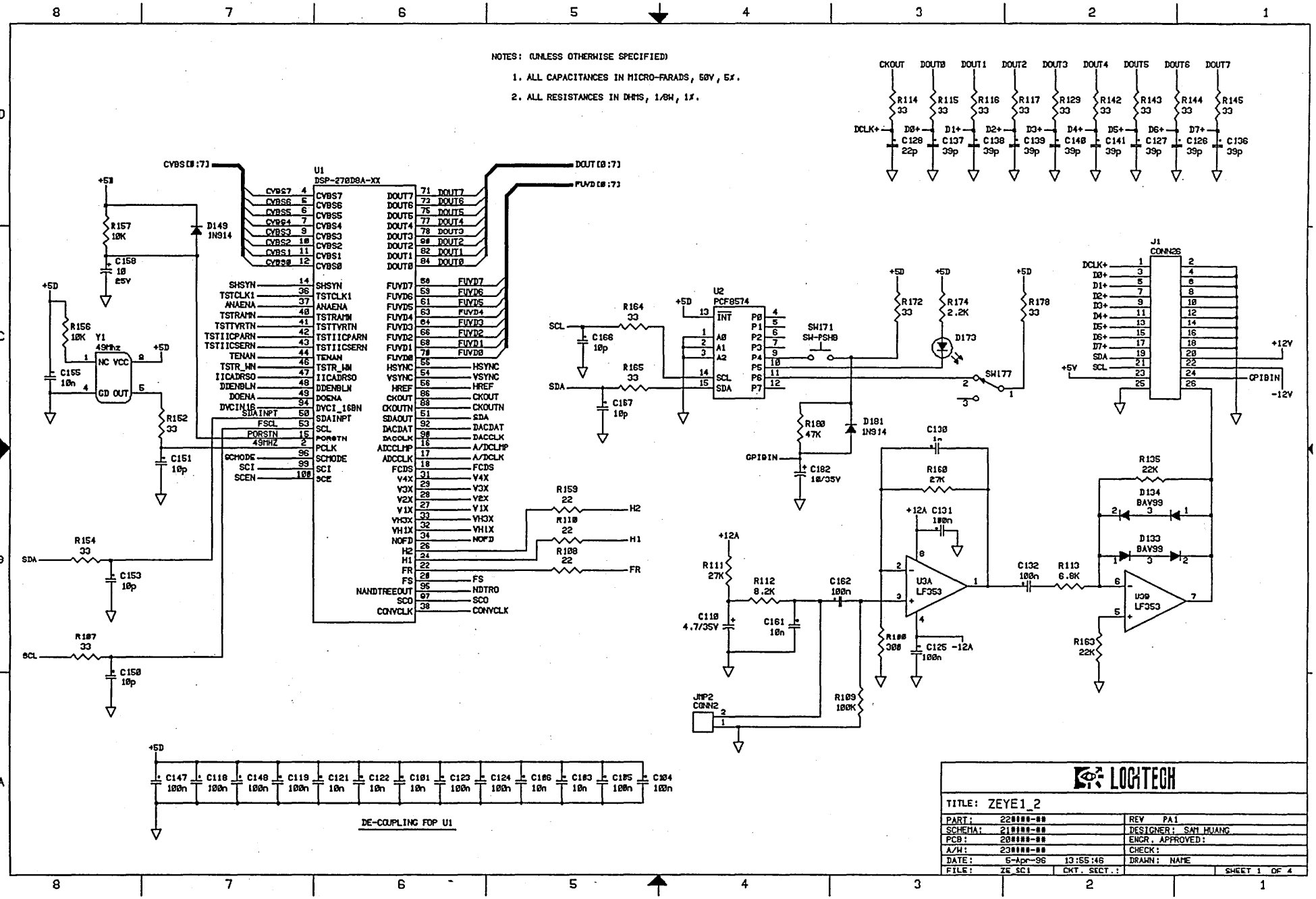
PIN#	SIGNAL	PAIR	PIN#	SIGNAL	PAIR
1	+5VOUT	10	35	n/c	
2	I2C_SCL	11	36	I2C_SDA	12
3	GPIIN_P	10	37	n/c	
4	n/c		38	n/c	
5	n/c		39	MIC_P	13
6	n/c		40	MIC_N	13
7	n/c		41	n/c	
8	n/c		42	n/c	
9	n/c		43	n/c	
10	n/c		44	n/c	
11	IN_GND[7]	9	45	n/c	
12	IN_P[7]	9	46	n/c	
13	IN_GND[6]	8	47	n/c	
14	IN_P[6]	8	48	n/c	
15	IN_GND[5]	7	49	n/c	
16	IN_P[5]	7	50	n/c	
17	IN_GND[4]	6	51	n/c	
18	IN_P[4]	6	52	n/c	
19	IN_GND[3]	5	53	n/c	
20	IN_P[3]	5	54	n/c	
21	IN_GND[2]	4	55	n/c	
22	IN_P[2]	4	56	n/c	
23	IN_GND[1]	3	57	n/c	
24	IN_P[1]	3	58	n/c	
25	IN_GND[0]	2	59	n/c	
26	IN_P[0]	2	60	n/c	
27	IN_P[CLK]	1	61	n/c	
28	IN_GND[CLK]	1	62	n/c	
29	n/c		63	n/c	
30	-12VOUT	11	64	+12VOUT	12
31	n/c		65	n/c	
32	n/c		66	n/c	
33	n/c		67	n/c	
34	n/c		68	n/c	

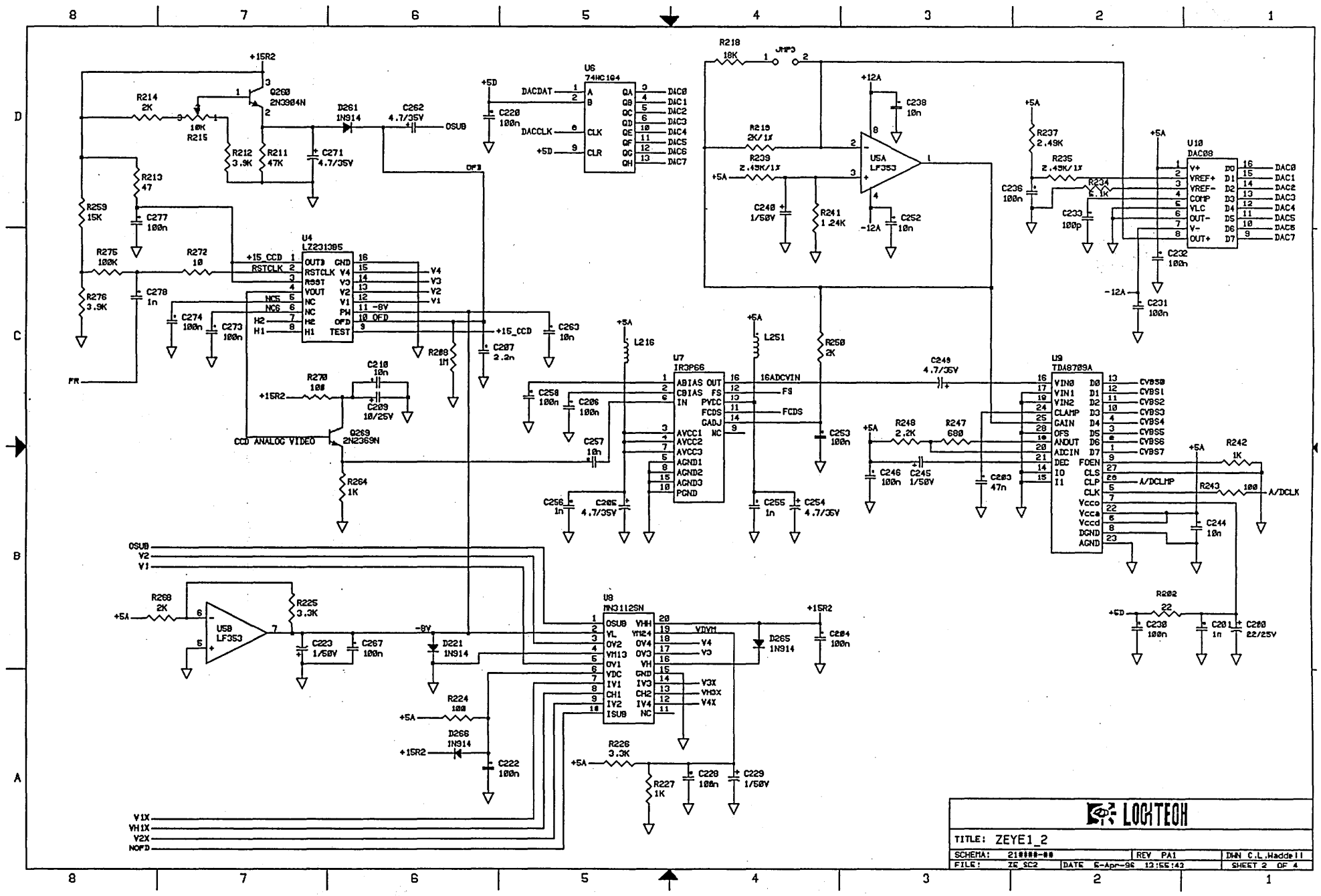


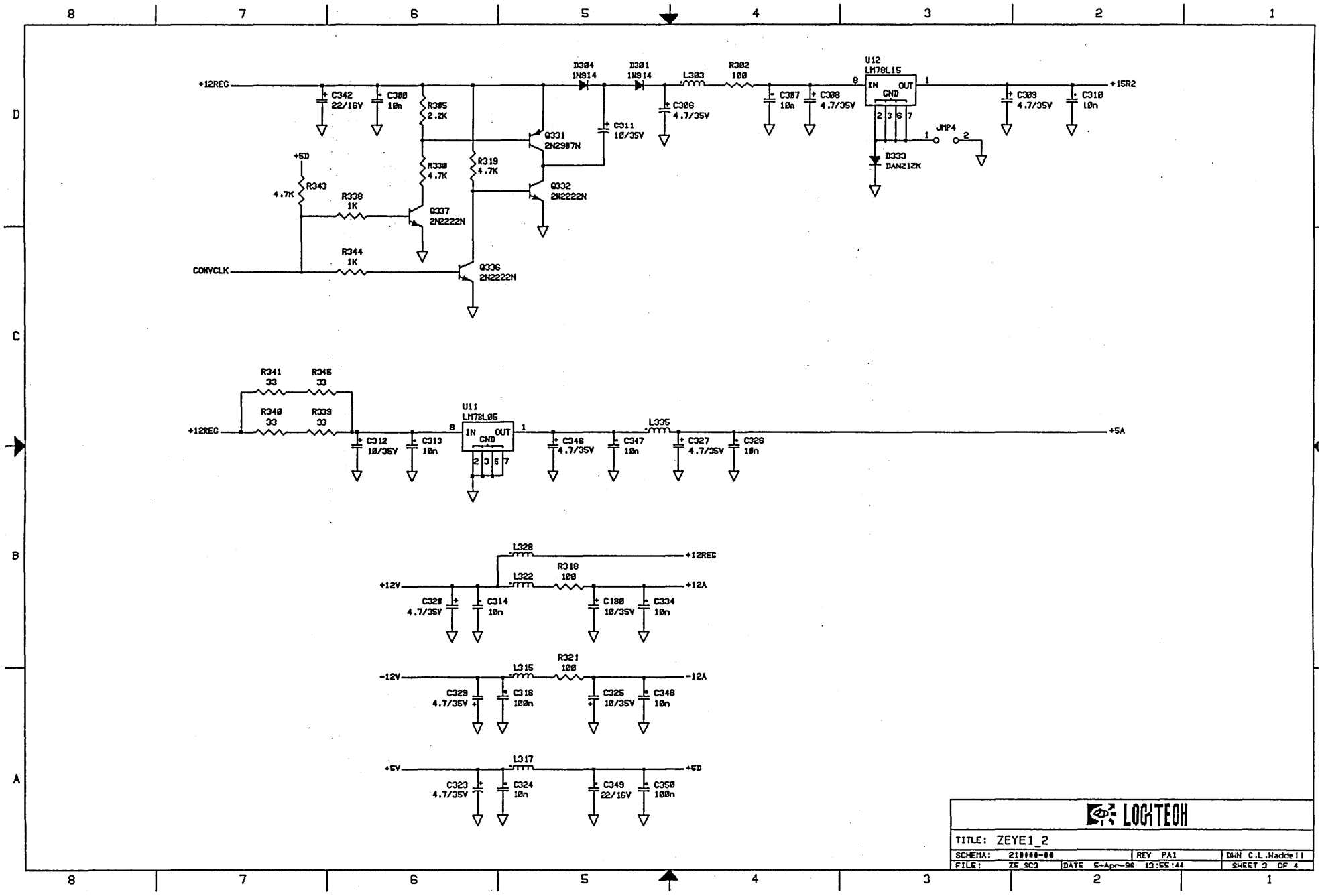
AMP
787254-1

AIDAN KEHOE
Aidan Kehoe
K4252

① ? fuse
② GNDs on one or all lines
③ IRQ option for resistor

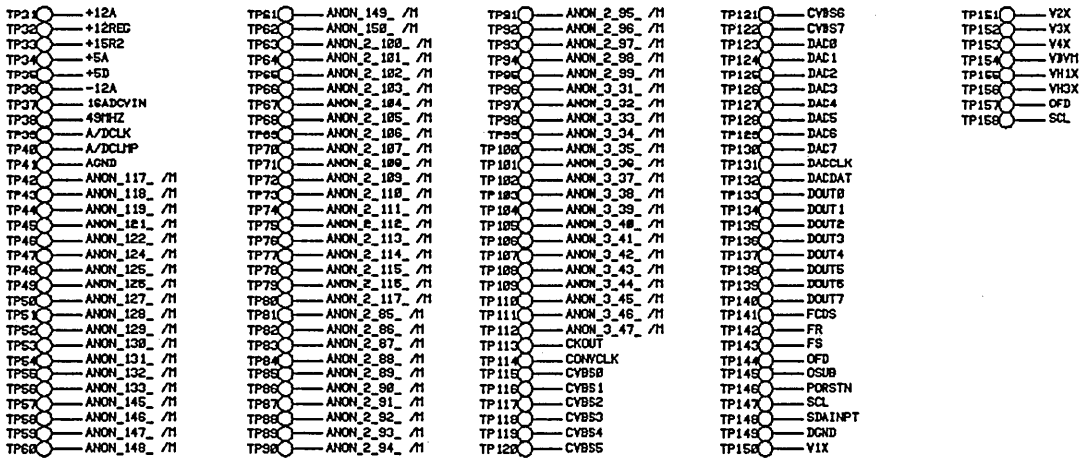

NOTES: (UNLESS OTHERWISE SPECIFIED)
1. ALL CAPACITANCES IN MICRO-FARADS, 50V, 5%.
2. ALL RESISTANCES IN OHMS, 1/8W, 1%.
U1
DSP-270D8A-XX
U2
PCF8574
J1
CONN26
U3A
LF353
U3B
LF353
JMP2
CONN2
SW171
SW-PSHB
SW177
D149
1N914
D173
D181
1N914
D134
BAV99
D133
BAV99
Y1
49Mhz
CKOUT
DOUT0
DOUT1
DOUT2
DOUT3
DOUT4
DOUT5
DOUT6
DOUT7
DCLK+
CVBS[0:7]
DOUT[0:7]
FUVD[0:7]
GPIBIN
+12V
-12V
+5V
+5D
+12A
-12A
SDA
SCL
H2
H1
FR
DE-COUPLING FOR U1
LOGITECH
TITLE: ZEYE1_2
PART: 22####-##
SCHEMA: 21####-##
PCB: 20####-##
A/W: 23####-##
DATE: 6-Apr-96 13:55:46
FILE: ZE_SC1
CKT. SECT.:
REV PA1
DESIGNER: SAM HUANG
ENGR. APPROVED:
CHECK:
DRAWN: NAME
SHEET 1 OF 4


+15R2
Q260 2N3904N
R214 2K
10K R215
R212 3.9K
R211 47K
D261 1N914
C262 4.7/35V
OSUB
C271 4.7/35V
R213 47
R259 15K
C277 100n
R275 100K
R272 10
R276 3.9K
C278 1n
FR
U4 LZ23108
+15_CCD
RSTCLK
C274 100n
C273 100n
C210 10n
R270 10Ø
C209 10/25V
Q269 2N2369N
CCD ANALOG VIDEO
R264 1K
R208 1M
C207 2.2n
C263 10n
OFD
+5D
C220 100n
U6 74HC164
DACDAT
DACCLK
DAC0
DAC1
DAC2
DAC3
DAC4
DAC5
DAC6
DAC7
R218 18K
JMP3
+12A
C238 10n
R219 2K/1%
R239 2.49K/1%
U5A LF353
C240 1/50V
R241 1.24K
-12A
C252 10n
+5A
R237 2.49K
R235 2.49K/1%
R234 5.1K
C236 100n
C233 100p
U10 DAC08
C232 100n
C231 100n
R250 2K
L216
L251
U7 IR3P66
16ADCVIN
FS
FCDS
C258 100n
C206 100n
C257 10n
C256 1n
C205 4.7/35V
C255 1n
C254 4.7/35V
C253 100n
C249 4.7/35V
U9 TDA8709A
CVBS0
CVBS1
CVBS2
CVBS3
CVBS4
CVBS5
CVBS6
CVBS7
R248 2.2K
R247 680
C246 100n
C245 1/50V
C203 47n
R242 1K
R243 100
A/DCLMP
A/DCLK
C244 10n
OSUB
V2
V1
R268 2K
U5B LF353
R225 3.3K
C223 1/50V
C267 100n
D221 1N914
U8 MN3112SN
VDVM
V4
V3
D265 1N914
C264 100n
R224 100
D266 1N914
C222 100n
R226 3.3K
R227 1K
C228 100n
C229 1/50V
R202 22
C238 100n
C201 1n
C200 22/25V
V1X
VH1X
V2X
NOFD
LOCKTECH
TITLE: ZEYE1_2
SCHEMA: 21####-##
REV PA1
DWN C.L.Haddell
FILE: ZE_SC2
DATE 5-Apr-96 13:55:43
SHEET 2 OF 4



LOCKTECH

TITLE: ZEYE1_2			
SCHEMA: 210000-00		REV PA1	DWN C.L.Haddell
FILE: ZE_SC2	DATE 5-Apr-96 13:55:44		SHEET 2 OF 4



LOCKTEOH

TITLE: ZEYE1_2			
SCHEMA: 21####-##		REV PA1	DWN C.L.Waddell
FILE: ZE_SC4	DATE 5-Apr-96	13:55:45	SHEET 4 OF 4

LOGICOOL

FAX Transmission

宛て先	Logi Inc. Mr. Kevin Scanlon 様	TEL : FAX :
送信元	株式会社ロジクール Tak Miyamoto	TEL : 03-3578-8851 FAX : 03-3578-8860 〒105 東京都港区虎ノ門 5-1-5 虎ノ門 45 森ビル
日付	1996 年 5月 22日	枚数 19 (但しこのページを含む)

件: Panasonic CCD Testing.

Hello! Kevin,

I send you Panasonic CCD Testing.
(This request is from Leslie.)

Please refer to the attached sheets.

1. Reliability Report - 1 page
2. Handling Precautions - 5 pages
3. CCD SOLID IMAGE PICKUP ELEMENT TEST METHOD - 12 pages.

Regards

Tak 5/22

May. 20. 1996
From:Tadahiro Sugino
Address:Matsushita Electronic Corporation
Quality Engineering Section
CCD Division
Semiconductor Group
Nagaokakyo, Kyoto, 617 Japan
Tel:075-956-9431 Fax:075-953-7180

Reliability report

1. CCD image sensor :MN3726MFE

2. Reliabilty tests specification

Test item a. Humid environmental test	Details and test conditions	Specification (r/n)
Unsaturated pressure cooker test.	Temperature: 121℃ Relative humidity: 85% Environmental pressure: 2.3kg/cm^2 Test time: 8h	0/15
b. Life test	Details and test conditions	Specification
Temperatare humidity storage life test.	Temperature: 85℃ Relative humidity: 85% Test time: 200h	0/15
High temperatare storage life test.	Temperature: 85℃ Test time: 1000h	0/15
Low temperatare storage life test.	Temperature: -40℃ Test time: 1000h	0/15
c. Thermal environmental test	Details and test conditions	Specification
Temperature cycle test.	-40℃ (30min)→5~35℃ (5min)→80℃ (30min)→5~35℃ (5min) ← 1 cycle → Test cycle: 20cycle	0/15
Thermal shock test.	-40℃ (5min)↔80℃ (5min)・・・Transportation time ← 1 cycle → is with in 10seconds Test cycle: 20cycle at the maximum.	0/15
Soldering heat resistance test.	Method:Dipping in the solden bath. Temperature: 310℃ Test time: 5s	0/15
d. Mechanical test	Details and test conditions	Specification
Mechanical shock test.	Acceleration of mechanical inpact: 14700m/s^2 Duration of inpact: 0.5ms Test cycle: 3cycle	0/15
Vibration test.	Acceleration of mechanical inpact: 196m/s^2 Frequency of vifration:Constant of sweep 100~2000Hz Test cycle: 4cycle	0/15
Terminal stength test.	Weight due to the terminal: 250g How to bend:Bend the terminal at right angle and back. Holding time: 3s	0/15
e. Special test.	Details and test conditions	Specification
Solderability test.	Soldering temperature: 270℃ Soldering time: 3s	0/15

Panasonic	Handling Precautions	WDP1016-G-500B WDP1020-G-0600A	
		Page	1/5

1. Static Electricity.

The performance of semiconductor devices such as LSIs is increasing every day due to advances in MOS technology and fine line semiconductor device manufacturing technology.
However, on the other hand, these devices are extremely weak against overvoltages such as static electricity and have the drawback that their function can be destroyed easily.
In order to ensure the reliability, it is necessary to clarify the phenomenon and mechanism of deterioration and destruction due to static electricity, and to establish standard processing methods and evaluation methods.

The semiconductor manufacturers are taking protective measures by incorporating low breakdown voltage junction diodes or diodes that operate in the forward bias region for external static electric voltages in parallel with the terminals, or by adding resistors using diffused layers.

Such protective circuits have external energy absorption capacities according to the dimensions, and have the operation of protecting the delicate gate oxide layer.

The static electricity breakdown countermeasures to be taken during device application are the measures to be taken by the users after understanding these limitations.
The electrostatic stresses on the devices after they are shipped from the factory and during their transportation, storage, passage through the set assembly line until they are finally incorporated into the end equipment can exceed the limits that can be solved by measures taken within the devices if they are carelessly abandoned, and hence the devices may lose their intended performance characteristics.

However, such electrostatic stresses can be suppressed to a low level easily using relatively simple measures.
While it is important to prevent the generation of static electricity and to remove any accumulated static charges as quickly as possible, there are countless opportunities in routine life fore environmental conditions such as the equipment material, packaging material, humidity level, the workers' clothes, etc.
The major precautions to be taken are listed below.

(1) Since static electricity is generated easily due to friction in a dry atmosphere, the humidity should be controlled by the airconditioner and a relative humidity or 50% or more should be maintained when using these devices. In particular, sufficient care should be taken when the relative humidity becomes less than 30%, electrostatic voltages ranging from several kilovolts to several tens of kilovolts are applied to the CCD image sensors thereby greatly increasing the probability of their being destroyed.

(2) In order to promote the natural discharging of static electricity accumulated on surrounding insulating materials, it is effective to use and AC driven ionizer.

Panasonic	Handling Precautions	WDP1016-G-500B WDP1020-G-0600A	
		Page	2 / 5

(3) Since the human body and chemical fiber garments can easily develop electrostatic voltages of several kilovolts due to friction, all operators should wear anti-static clothing.
Further, conductive shoes should also be worn.

(4) Use of nylon gloves should be avoided and cotton gloves should be used instead when handling CCD image sensors, and the body should always be grounded.
To do this, the customary method is to wear earth bands on the wrists when handling the devices.
During such operations, insert a resistor of about 1MΩ between the lead wires and ground for the sake of safety.
Further, conductive mats should be laid on the floor and on the workbench, and these mats should be grounded to remove any accumulated static charge.

(5) When staring CCD image sensors, make sure that they are not taken out of their conductive containers.
Also, avoid placing the devices on plastic or inserting them in styrene foam which can easily get charged.
If there is any need to return the devices to us, make sure that they are placed in the same conductive container in which they were delivered.

(6) At the time of soldering the devices, make sure the tip of the soldering iron is grounded in order to prevent high voltage leaks.
Also, ground the solder bath when the devices are soldered by dip soldering.

(7) Make sure that the measuring instruments such as the oscilloscope, digital voltmeter, etc., used during inspection are also properly grounded. Further, always make sure that the constant temperature bath, life test equipment, etc., used during reliability tests are properly grounded.

(8) When handing CCD image sensors, take care not to touch directly the leads and not to drop the devices.

(9) Connect all terminals that are not used to the same voltage as the ground terminal.

(10) When handling the printed circuit boards on which a CCD image sensor has been mounted, take the same precautions as when handling the individual CCD image sensors.

(11) Always use conductive or anti-static containers that are not affected by static electricity when transporting CCD image sensors or printed circuit boards on which CCD image sensors have been mounted.

Date	February. 26, '96				

Panasonic	Handling Precautions	WDP1016-G-500B WDP1020-G-0600A	
		Page	3 / 5

2. Precautions During Use.

(1) Do not use the devices at conditions exceeding the absolute maximum ratings given in the product specifications.

(2) Take care not to insert the devices in the wrong direction since inserting so can interchange the source and drain or operate the protective transistors in the forward bias thereby causing large currents to flow.
Such large currents can fuse open the aluminum interconnection patterns on the chip.

(3) Do not use power supplies whose transient voltages can exceed the absolute maximum ratings.

(4) To prevent the ocurrence of latch-up, use the recommended drive circuits that take into consideration the sequence of rising of the power supply voltages.
When a latch-up ocurrs, an abnormally large current flows between ϕV-PW(p-well) and the substrate generating excessive heat in the CCD image sensor and hence causing its defecate operation.

3. Soldering Precautions.

(1) Basically, these devices should be hand soldered using a 30W soldering iron at a temperature of 270℃(310℃ max) with the soldering time being less than 3 seconds per pin.

(2) When using dip soldering, the solder temperature should be 260±5℃ and the solder dipping time should be less than 10 seconds.

(3) Do not dip solder using a mounting oven as this will cause cracks or dirt in the glass window.

(4) When repairing the soldering or desoldering the device, sufficiently cool the soldering part.
Also, do not solder suction type tools for desoldering the devices.
When using an electric desoldering tool, use the type which uses the zero crossing ON/OFF type temperature control.
In addition, take countermeasures such as proper grounding against the surges from the AC power lines of vacuum pump motors, etc.

Date	February. 26, '96				

Panasonic	Handling Precautions	WDP1016-G-500B WDP1020-G-0600A	
		Page	4 / 5

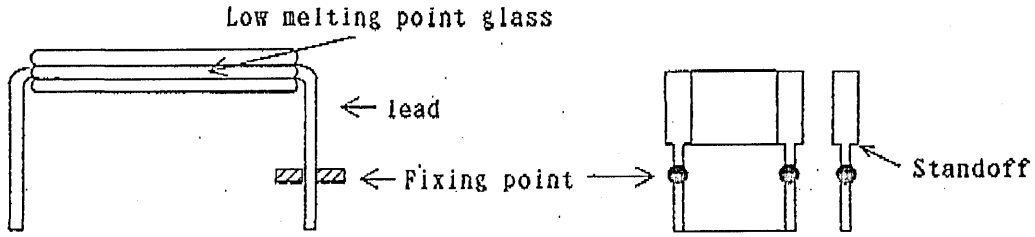
4.Measures to Prevent Damages to the Package.

Take the following precautions to prevent damages to the package.

(1) Since CCD image sensors use low melting point glass and ceramics,care should be taken not to drop them or subject them to mechanical shock.
In particular, when the device leads have been fixed in a socket or in a printed circuit board, the package can be damaged by a smaller shock than when the package is by itself.

(2) Make sure that the base of the leads of the CCD image sensor is not subject to stress when mounting the device.
Also,make sure that the point of applying the force is below the standoff of the leads. Since the leads of the package have been fixed using a low melting point glass,applying stress to the leads generates cracks in the low melting point glass at the base of the leads.

(3) When inserting the packages in the housing, make sure that any warping of the package is avoided.
Since the packages have some slight warp,pressing them between hard plates can break the packages.



Date	February.26,'96				

Panasonic	Handling Precautions	WDP1016-G-500B WDP1020-G-0600A	
		Page	5／5

5. Handling Precautions of Optical Devices.

While the surface glass has been cleaned sufficiently at the time of shipping the CCD image sensors, clean the surface glass again by wiping with a lens cleaning paper or a soft cloth just before using.
Also, the following precautions should be taken since scratches, dirt or dust on the surface glass can have detrimental effects on the obtained images.

(1) All work should be done in a clean environment such as a clean booth, etc.
A cleanliness class of about 1000 is appropriate for handling these devices.

(2) Take care not to touch the glass surface with fingers or other objects and not to cause adhesion of dirt or dust on the glass surface.
When any dirt or dust gets adhered to the glass surface, blow it off using an air blower in a static-free environment. Use of an ignited air blow is recommended for removing dust adhered due to static electricity.
However, in order to protect the device against static electricity, make sure that all the leads of the CCD image sensor have been grounded at the time of blowing away the dust.

(3) When the dirt or dust cannot be removed by an air blow or for removing oily dirt, carefully and lightly wipe the surface of the glass with a clean cotton bud or lens cleaning paper dipped in isopropyl alcohol, taking care not to scratch the glass surface.

(4) The imaging becomes defective when there is any scratch, crack, or breakage in the surface glass.
Do not strike the surface glass, or subject it to a shock big enough to deform it, or rub it strongly, or sctratch it otherwise. Also, even when wiping the surface glass with a soft cotton bud or cloth, if these are dry, dust is generated which in turn can scratch the surface glass.

(5) It is necessary to store the devices in specially designed cases in order to protect them from dirt and dust.

(6) As a measure against water condensation on the device when transporting it between rooms with wide differences in temperature, as required.

6. Precautions in Transportation and Storage.

(1) Since CCD color image sensors have or gamic fitters, make sure that they are not exposed to ultraviolet light or direct sunlight during storage, transportation, or working.
Leaving the devices exposed to strong night for long durations discolors the color filter.

(2) Since the CCD image sensors are precision optical components, avoid storing or using them at high teperature or high humidifies.

Date	February. 26, '96				

CODE No.								1 / 12

CCD SOLID IMAGE PICKUP ELEMENT TEST METHOD

PICTURE-COMPONENT DIVISION
MATSUSHITA ELECTRONICS CORPORATION
QUALITY ENGINEERING SECTION

APPROVED BY	CHECKED BY	CHECKED BY	MADE BY
	G. Horii		T. Sugino

May. 29, '95					

Panasonic	Test Method	2／12

1. GENERAL DESCRIPTION

1.1 Optical System

1.1.1 Lens

- FUJINON CF-25L, F0.85, f = 25 mm

1.1.2 Optical Filter

- Infrared Rays Cutting Filter CAW-500, t = 2.5 mm, Made by Hoya Glass

1.1.3 Light Box

- Color temperature: 3200K (Use a halogen lamp 2856K and attach a color temperature conversion filter LB- 40 (made by Hoya Glass))
- Illumination: Refer to Sheet No.3

1.1.4 Test Chart

- J-chart ----- EIAJ J-chart, Without glass, 3/10 window in the vertical direction
- 5 % cut chart ----- Periphery of the H and V image frame cut by 5 %
- U-chart ----- J-chart partly light shut off
- Flicker chart
- DNP color bar chart
- DNP grey scale chart

1.2 Check Unit

- MEC-possessed exclusive test unit: Refer to a block diagram (Sheet No.12)
- Frequency characteristic: Flat within a band
- Contour high-light circuit: None
- Gamma compensation: None

1.3 Test Environment

- Operating temperature: 25 ± 2 ℃
- Ambient illumination: 300 lux or less

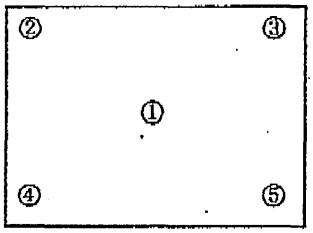
Panasonic	Test Method	3/12

2. LIGHT BOX

2.1 Light Source
- Halogen electric bulb light box

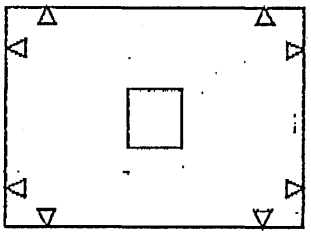
2.2 Pattern Surface Color Temperature
- 2856K± 20

2.3 Pattern Surface Illumination



① ・・・ 2900 ℓx ± 1 %
② ・・・ 2900 ℓx ± 5 %
③ ・・・ 〃
④ ・・・ 〃
⑤ ・・・ 〃

3. DEFINITION OF STANDARD CONDITION
- Lens: F:8 (f = 25 mm)
- Image angle: Picks up an image of the J-chart in a specified size.



J-chart

4. ADJUSTMENT OF ELEMENT
Adjust a measured sample as follows beforehand to make initial setting.

4.1 Initial Setting
- Items Set to Standard Values

V ϕ R	V Sub
V RD	V ϕ V1
V OD	V ϕ V2
V ϕ H1	V ϕ V3
V ϕ H2	V ϕ V4
V IS	V PW
V PT	

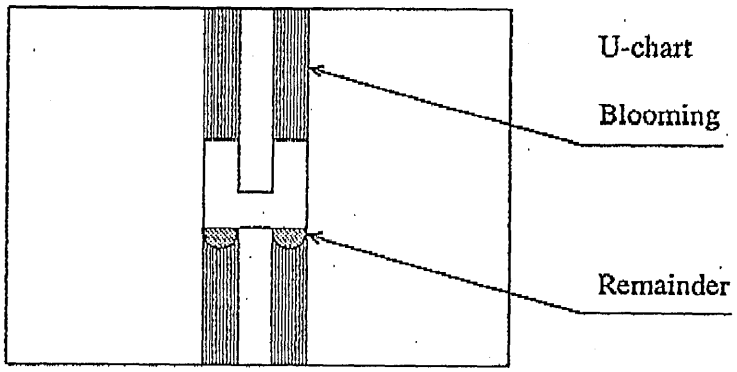
4.2 Items to Be Set to Optimum Values

4.2.1 Setting of VSub

(1) Initial setting is 8.0 V. Pick up an image of the U-chart at standard quantity of light × 100 and adjust VSub to a minimum voltage which does not generate blooming.

(2) After setting the above, when there is a remainder under the center of the U-chart on the monitor screen, increase a VSub voltage further to set to a minimum voltage where the remainder is removed.

(3) When roughness is generated under the minimum operating condition of VSub, set to a minimum voltage where no roughness is generated.



(4) When no blooming appears, set to 5.0 V.

5. ADJUSTMENT OF TEST UNIT

5.1 Initial Setting (Adjust once and no readjustment is required)

5.1.1 Adjustment of Burst Level

At TP4 (video output), adjust the Burst Level VR so that the burst level comes to a specifiied value after BPF.

5.1.2 Adjustment of Color Signal

In the standard condition(F:8), pick up an image of the DNP color bar chart with the standard element, and adjust the color signal system so that the specified condition can be obtained on the vector scope.

5.2 Adjustment to Optimum Values (Adjustment Required for Each Element)

5.2.1 Black Level Adjustment
Shut off a light and adjust the Black Level VR so that color balance at dark time can be obtained at TP4 (video output).

5.2.2 Top Gain Adjustment
Pick up an image of the J-chart in the standard condition (F:8) and adjust the Top Gain VR so that video output will come to a specified value at TP4 (video output).

5.2.3 White Balance Adjustment
Pick up an image without any charts in the standard condition (F:8) and adjust the YL (R-Y), YL (B-Y) Gain VR so that white will come to the center on the vector scope.

5.2.4 Chroma Level Adjustment
Pick up an image of the DNP color bar chart in the standard condition (F:8) and adjust the Chroma Gain VR so that degree of saturation for R will become 175 % with respect to an amplitude of color burst after BPF at TP4(video output).

6. CHARACTERISTIC CHECK

6.1 S/N

(1) Turn off a chroma signal.

(2) Shut off incident rays, connect TP4 (video output) to a noise meter, and measure S/N.
Noise meter conditions:
NTSC 100 kHz, 4.2 MHz LPF
PAL 100 kHz, 5.0 MHz LPF
S.C. trap ON

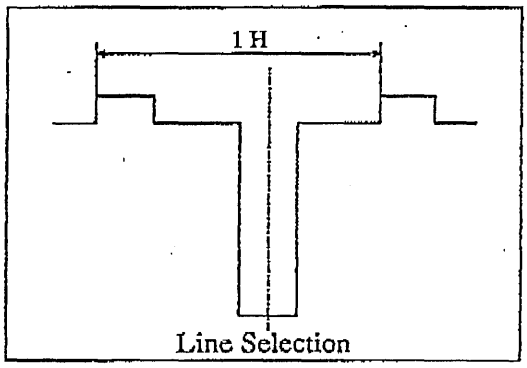
Panasonic	Test Method	6／12

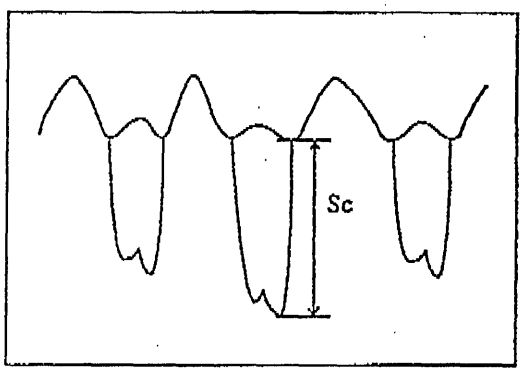
6.2 Carrier Saturation Output (Sc)

(1) Pick up an image of the J-chart at F:8 and observe TP3 (carrier) through an oscilloscope (H rate).

(2) Open a diaphragm from the standard condition of F:8 to search for a point where a carrier becomes maximum.

(3) Observe then TP1 (element output) through the oscilloscope (H rate) and line-select a higher signal level in the center of the screen and measure a larger amplitude.



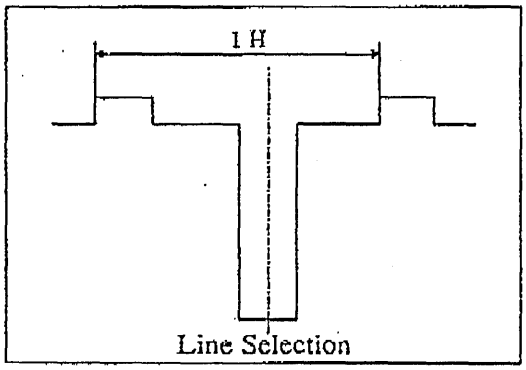


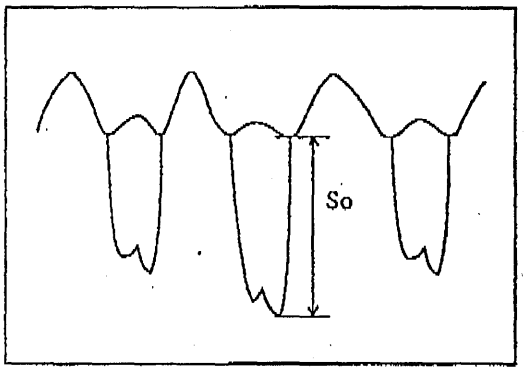
6.3 Sensitivity (So)

(1) Pick up an image of the J-chart in the standard condition (F:8).

(2) Observe TP1 (element output) through the oscilloscope (H rate) and line-select a higher signal level in the center of the screen.

(3) Measure a larger amplitude.





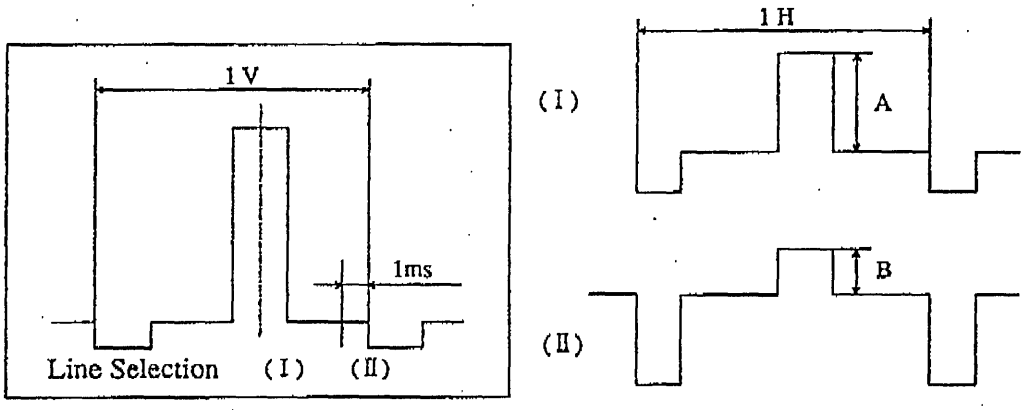
6.4 Vertical Smear (Sm)

(1) Pick up an image of the 1/10 chart in the standard condition of F:8.

(2) At TP2 (YH), Measure an output signal A in the center of the screen and a smear signal B in the 1ms section from the bottom of the screen through the oscilloscope (V rate).

(3) Obtain smear out of the following formula:

$$Sm = \frac{\text{Smear Signal (B)}}{\text{Output Signal (A)}} \times 100\ (\%)$$



Note 1) To measure the smear signal, use the average function of the oscilloscope to improve accuracy.

Note 2) When measuring the smear signal, turn off vertical read-out pulses.

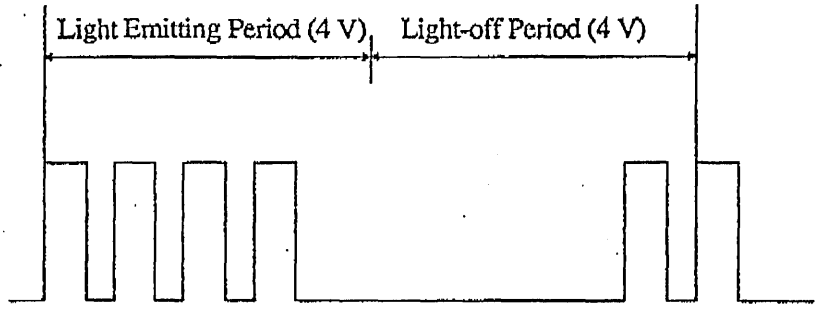
6.5 Image Lag (LAGd)

(1) Pick up an image of flickering light emitting diode (green). At TP2 (YH), adjust brightness of the light emitting diode so that the output level at LED ON time will become 1/8 of the signal level at J-chart standard image pickup time.

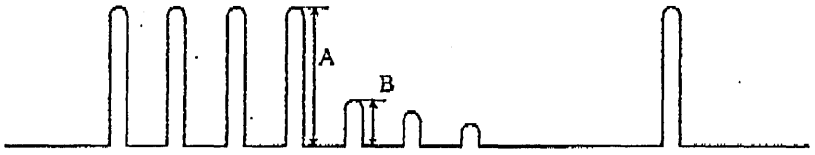
(2) Observe a TP2 signal through the oscilloscope (V rate), and measure a reference signal A, and a remaining signal B in the 1st field after the LED goes off.

(3) Obtain the residual image out of the following formula:

$$\text{Image Lag} = \frac{\text{Remaining signal (B)}}{\text{Reference signal (A)}} \times 100\ (\%)$$

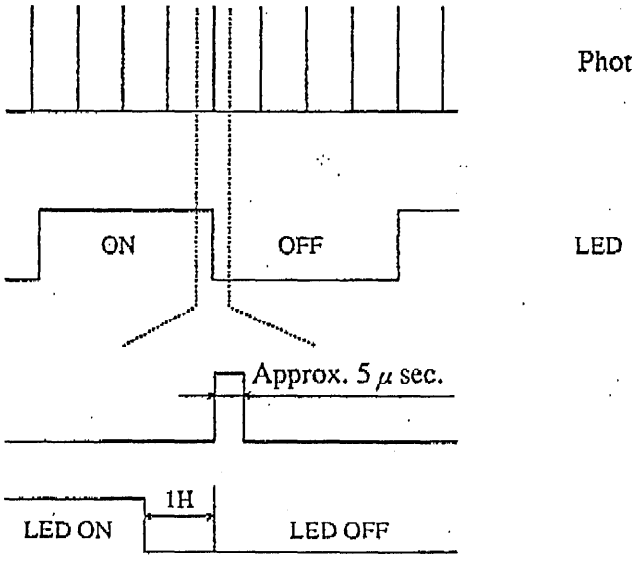


LED Applied Pulse



TP2 Output Signal

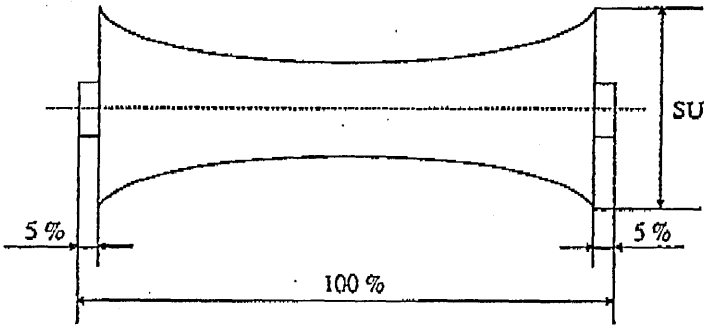
(4) LED ON and signal read-out pulse timing



Panasonic	Test Method	9／12

6.6 Color Shading (SU)

(1) Pick up an image of the 5 % cut chart in the standard condition (F:8).
(2) Adjust white balance to the vector center.
(3) At TP4(video output), measure P-P of a signal after BPF through the oscilloscope (H, V rate)



6.7 Color Dynamic Range (DR)

(1) Pick up an image of the J-chart and observe TP3 (carrier) through the oscilloscope.
(2) Open the diaphragm from the standard condition (F:8) and obtain a point where the carrier becomes maximum.
(3) Measure the then element output's (TP1) value, A (mVp-p).
(4) Obtain DR out of the following formula.

$$DR = \frac{A}{\text{Standard output (200 mV)}} \text{ (times)}$$

6.8 Picture Blemish (PB)

(1) Narrow down by one graduation (F:11) from the standard condition and pick up an image without any charts.
(2) Measure the scar signal level for TP2 (YH).
(3) Scar evaluation is expressed in terms of contrast ratio (%) to the standard quantity of light of the measured value under the above-mentioned condition.

Panasonic	Test Method	10/12

6.9 Color Flicker

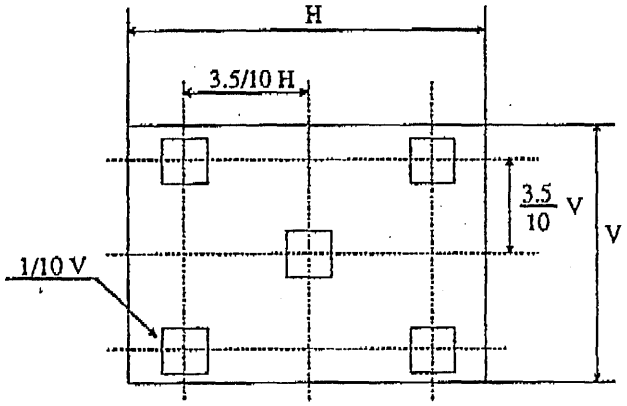
(1) Pick up an image of the flicker chart in the standard condition of F:8.

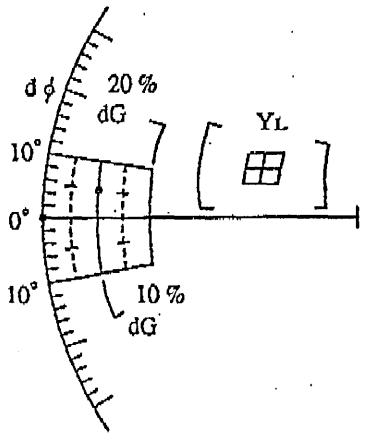
(2) Insert R, Ye, and Cy filters into 5 places in the screen, use the vector scope to observe partings of the amplitude and angle directions of the respective vectors. (Set the vector with larger amplitude to an outer circumferential position of 0 on the vector scope's scale to indicate a parted vector in terms of relative value.)

(3) Measure the worst point of the 5 points.

* 1) R, Ye, and Cy filters:
Use color glass filters.
Same characteristics as the DNP color bar chart.

* 2) Flicker chart





Example
[Amplitude difference: 10 %
Angle difference: 5 °]

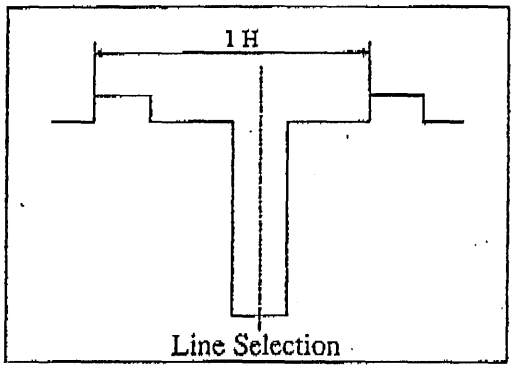
Panasonic	Test Method	11/12

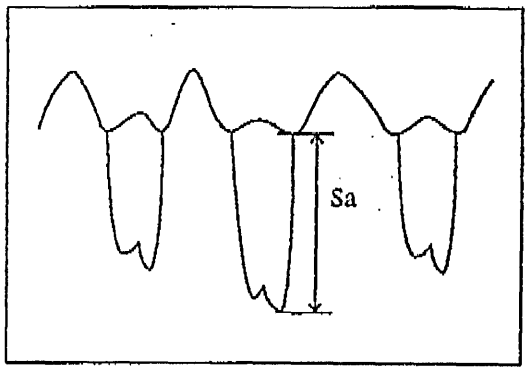
6.10 ϕR Bias Adjustment Method

(1) Pick up an image of the J-chart at F:1.4.

(2) Observe TP1(element output) through the oscilloscope (H rate) and line-select the signal level in the center of the screen

(3) Lower a ϕR bias voltage from around 5 V to adjust to a point where element output (Sa) becomes maximum.

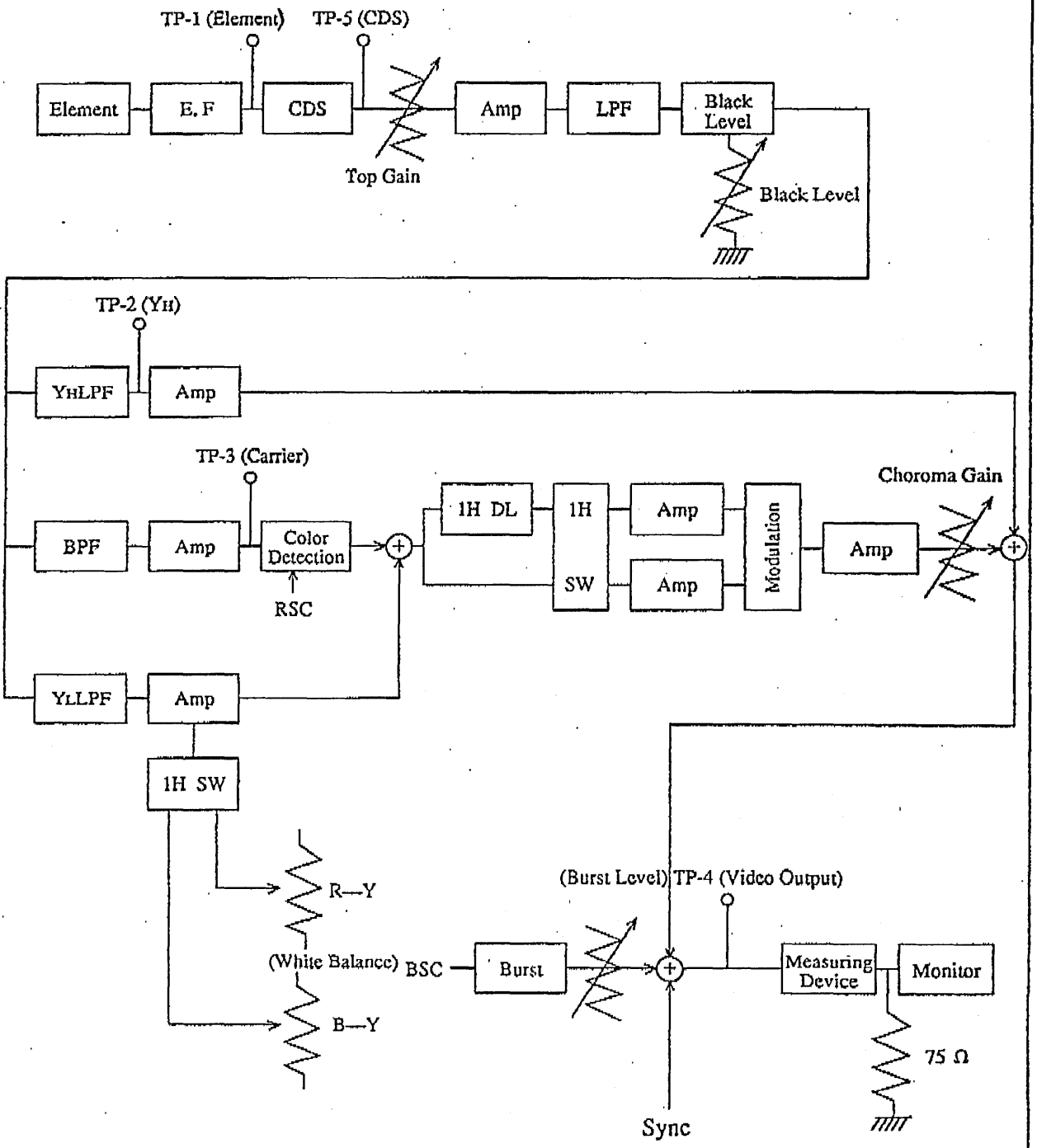




Panasonic	Test Method	12 / 12

Measurement Circuit Block Diagram

Note: No γ compensation,
No aperture compensation



SHARP

SPEC No. | CC057003B
ISSUE: Nov 18 1993

To ;

PRELIMINARY

SPECIFICATIONS

Product Type 1/3 type solid state color imaging device for NTSC system

Model No. LZ2313H5
(LZ2313H5)

※This specifications contains 18 pages including the cover and appendix.
If you have any objections, please contact us before issuing purchasing order.

CUSTOMERS ACCEPTANCE

DATE:

BY:

PRESENTED

BY: K. Misawa
K. MISAWA
Dept. General Manager

REVIEWED BY:

PREPARED BY: I. Baba

Development Dept. 4
VLSI Laboratories
Integrated Circuits Group
SHARP CORPORATION

CONTENTS

Note: The contents of this specification may be changed due to an improvement in characteristics or any other reason.
The circuit diagram and others included in this specification are intended for use to explain typical application examples. Therefore, we take no responsibility for any problem as may occur due to the use of the included circuit and for any problem with industrial proprietary rights or other rights.

1. GENERAL

LZ2313H5 is a 1/3 type (6.0 mm) solid state imaging device consisting of PN phote-diodes and CCDs(charge-coupled devices). Having about 270,000 pixels (horizontal 542 x vertical 492), it allows a stable color image to be obtained at high resolution.

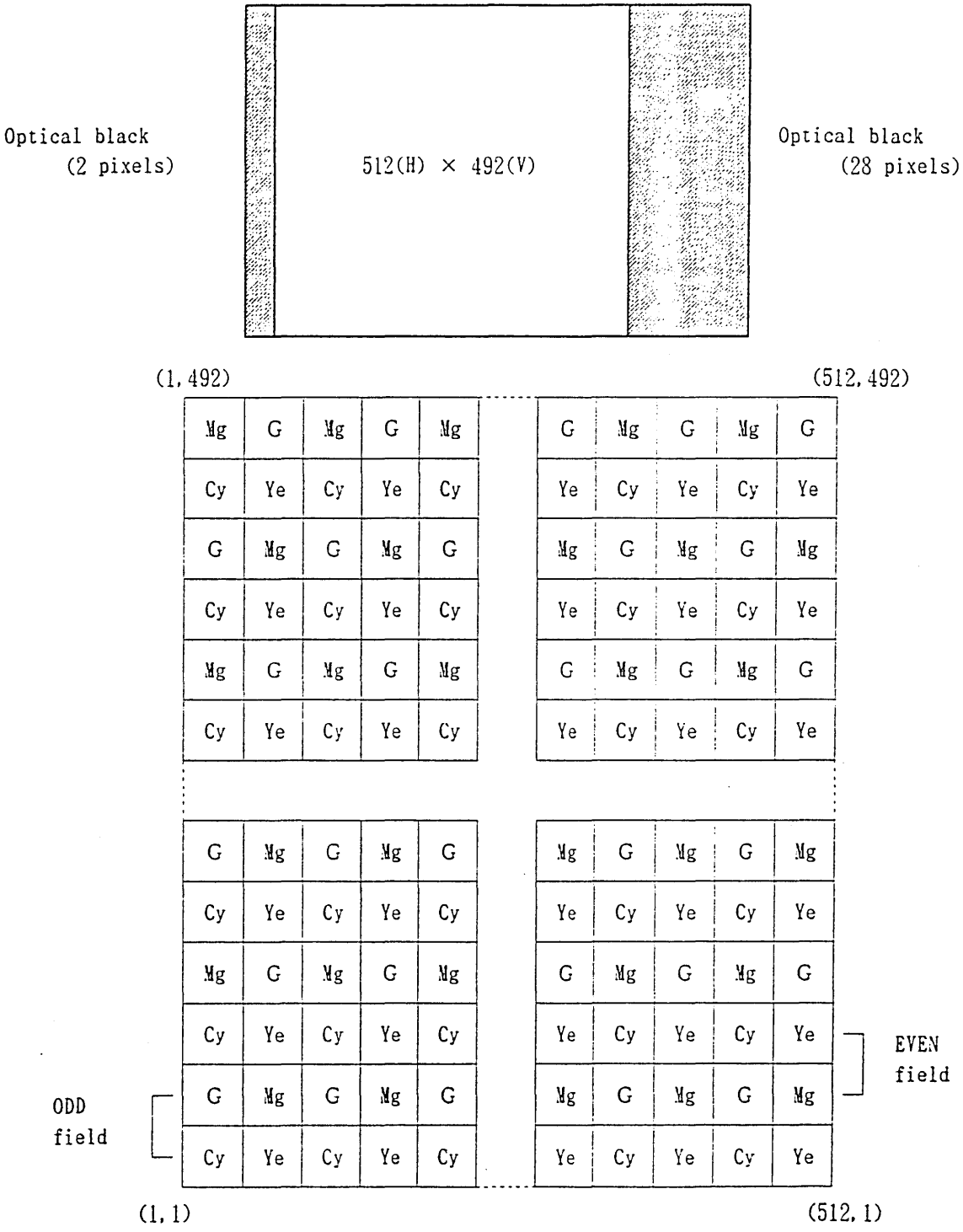
Features

1) Number of video picture elements : Horizontal 512 x vertical 492
   Pixel pitch : Horizontal 9.6 μm x vertial 7.5 μm
   Number of optically black pixel : Horizontal; front 2 and rear 28
2) Complementary color filters of Mg, G, Cy, and Ye
3) Reduced fixed pattern noise and lag
4) No sticking and no image distortion
5) Blooming suppression structure
6) Built-in output amplifier
7) 16-pin shrink DIP
   (Row space: 12.70 mm)
8) Variable electronic shutter(1/60 to 1/10000 s)
9) N-type silicon substrate N-MOS process
10) Not designed or rated as radiation hardened
11) Compatible with NTSC standard

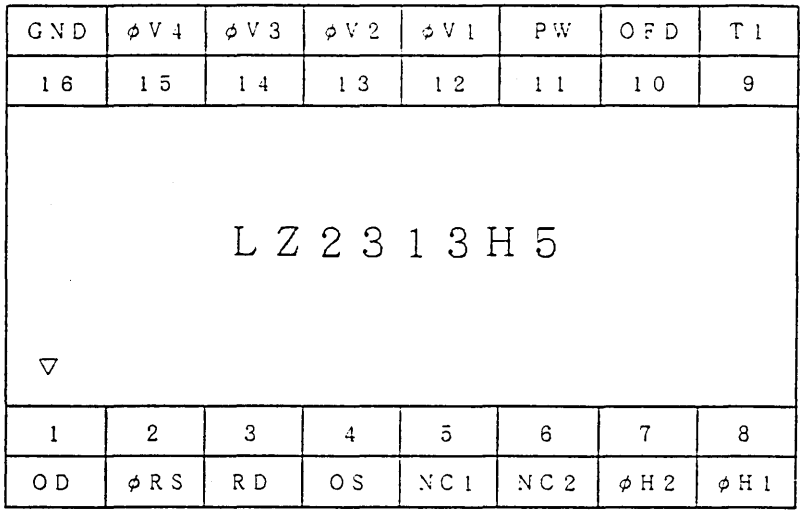
Applications

1) Cameras(Cam corders, industrial monitor cameras, etc.)
2) Pattern recognition

2. COMPOSITION OF PIXELS AND ARRANGEMENT OF COLOR FILTERS



3. PIN ASSIGNMENT AND PIN IDENTIFICATION



Symbol	Pin name
RD	Reset transistor drain
OD	Output transistor drain
OS	Video output
φRS	Reset transistor gate clock
φV1, φV2, φV3, φV4	Vertical shift register gate clock
φH1, φH2	Horizontal shift register gate clock
OFD	Overflow drain
PW	P type well
GND	Ground
T1	Test terminals
NC1, NC2	Non-connection (note 1)

(note 1)Connect each pin to GND directly or through a capacitor larger than 0.047μF.

4. ABSOLUTE MAXIMUM RATING

(Ta = 25℃)

Item	Symbol	Rating	Unit
Output transistor drain voltage	VOD	0 to +18	V
Reset transistor drain voltage	VRD	0 to +18	V
Overflow drain voltage	VOFD	0 to +55	V
Test terminal, T1	VT1	0 to +18	V
Reset gate clock voltage	VφRS	-0.3 to +18	V
Vertical shift register clock voltage	VφV	-9.0 to +18	V
Horizontal shift register clock voltage	VφH	-0.3 to +18	V
PW and vertical clock voltage difference	VPW-VφV	- 27 to 0	V
Storage temperature	Tstg	- 40 to +80	℃
Operating ambient temperature	Topr	- 20 to +70	℃

5. RECOMMENDED OPERATING CONDITIONS

Item		Symbol	Minimum	Typical	Maximum	Unit
Operating ambient tempetarure		Topr		25.0		℃
Output transistor drain voltage		VOD	14.5	15.0	16.0	V
Reset transistor drain voltage		VRD		VOD		V
Overflow drain voltage	When DC is applied(note1)	VOFD	5.0	(adj.)	19.0	V
	When pulse is applied p-p level (note2)	VφOFD	22.0			V
Ground		GND		0.0		V
P-well voltage		VPW	-9.0		VφVL	V
Test terminal, T1		VT1		VOD		V
Vertical shift register clock	LOW level	VφV1L, VφV2L VφV3L, VφV4L	-8.5	-8.0	-7.5	V
	INTERMEDIATE level	VφV1I, VφV2I VφV3I, VφV4I		0.0		V
	HIGH level	VφV1H, VφV3H	16.0	16.5	17.0	V
Horizontal shift register clock	LOW level	VφH1L, VφH2L	-0.05	0.0	0.05	V
	HIGH level	VφH1H, VφH2H	4.7	5.0	6.0	V
Reset gate clock	LOW level	VφRSL	0.0		VRD-13.0	V
	HIGH level	VφRSH	VRD-8.5		9.5	V
Frequency						
Vertical shift register clock		fφV1, fφV2 fφV3, fφV4		15.73		kHz
Horizontal shift register clock		fφH1, fφH2		9.53		MHz
Reset gate clock		fφRS		9.53		MHz

(note1) When DC voltage is applied, shutter speed is 1/60 seconds.
(note2) When pulse is applied, shutter speed is less than 1/60 seconds.

6. CHARACTERISTICS

No.	Item	Symbol	Note	Min.	Typ.	Max.	Unit
1	Photo response non-uniformity	PRNU	(a)			15	%
2	Carrier saturation	Vsat	(b)	450			mV
3	Dark output voltage	Vdark	(c)		0.3	3.0	mV
4	Dark signal non-uniformity	DSNU	(d)		0.6	2.0	mV
5	Sensitivity	R	(e)	440	600		mV
6	Smear ratio	SMR	(f)		0.009	0.016	%
7	Image lag	AI	(g)			1.0	%
8	Blooming suppression ratio	ABL	(h)	100			
9	Current dissipation	I_{DD}			4.0	8.0	mA
10	Output impedance	Ro			350		Ω
11	Vector breakup		(i)			5.0	°, %
12	Line crawling		(j)			3.0	%
13	Luminance flicker		(k)			2.0	%

【 Conditions 】

- Drive method : Field accumulation.
- DC and AC conditions : the typical values under the recommended operating conditions.
- Ambient temperature(Ta) : -25℃, but -60℃ for Item No. 3 and 4.
- Temperature of light source : 3200 K.
- Infrared absorbing filter (CM-500, 1 mm) is used.

【 Notes 】

- The standard output voltage is defined as 150 mV by the average output voltage under uniform illumination.
- The standard exposure level is defined when the average output voltage is 150 mV under uniform illumination.
- VOFD is adjusted to the minimum voltage with that ABL satisfy the specification.

(a) The image area is divided into 10 × 10 segments. The voltage of a segment is the average of output voltage from all the pixels within the segment. PRNU is defined by (Vmax - Vmin) / Vo, where Vmax and Vmin are the maximum and the minimum values of each segment's voltage respectively, when the average output voltage Vo is 150 mV.

(b) The output voltage measured at the carrier peak when carrier signal reaches maximum.

(c) The average output voltage under a non-exposure condition.

(d) The image area is divided into 10 × 10 segments. DSNU is defined by (Vdmax - Vdmin) under the non-exposure condition where Vdmax and Vdmin are the maximum and the minimum values of each segment's voltage, respectively, that is the average output voltage over all pixels in the segment.

(e) The average output voltage when a 1000 lux light source attached with a 90% refrector is imaged by a lens of F4, f50 mm.

(f) The sensor is adjusted to position a V/10 square at the center of image area where V is the vertical length of the image area. SMR is defined by the ratio of the output voltage detected during the vertical blanking period to the maximum of the pixel voltage in the V/10 square.

(g) The sensor is exposed at the exposure level corresponding to the standard condition preceding non-exposure condition. AI is defined by the ratio between the output voltage measured at the 1st field during the non-exposure period and the standard output voltage.

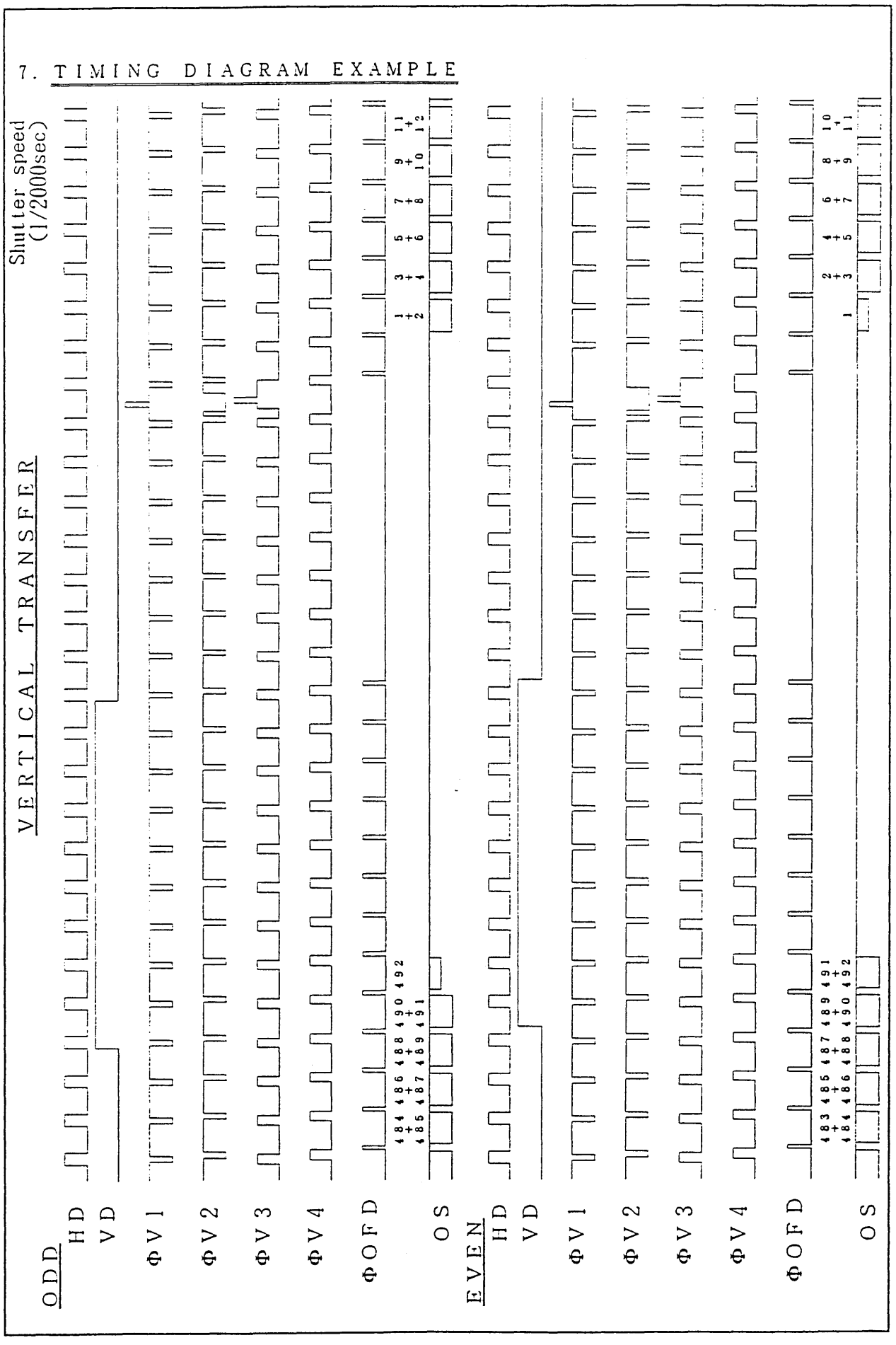
(h) The sensor is adjusted to position a V/10 square at the center of image area. ABL is the ratio between the exposure at the standard condition and the exposure at a point where a blooming is observed.

(i) Observed with a vector scope when the color bar chart is imaged under the standard exposure condition.

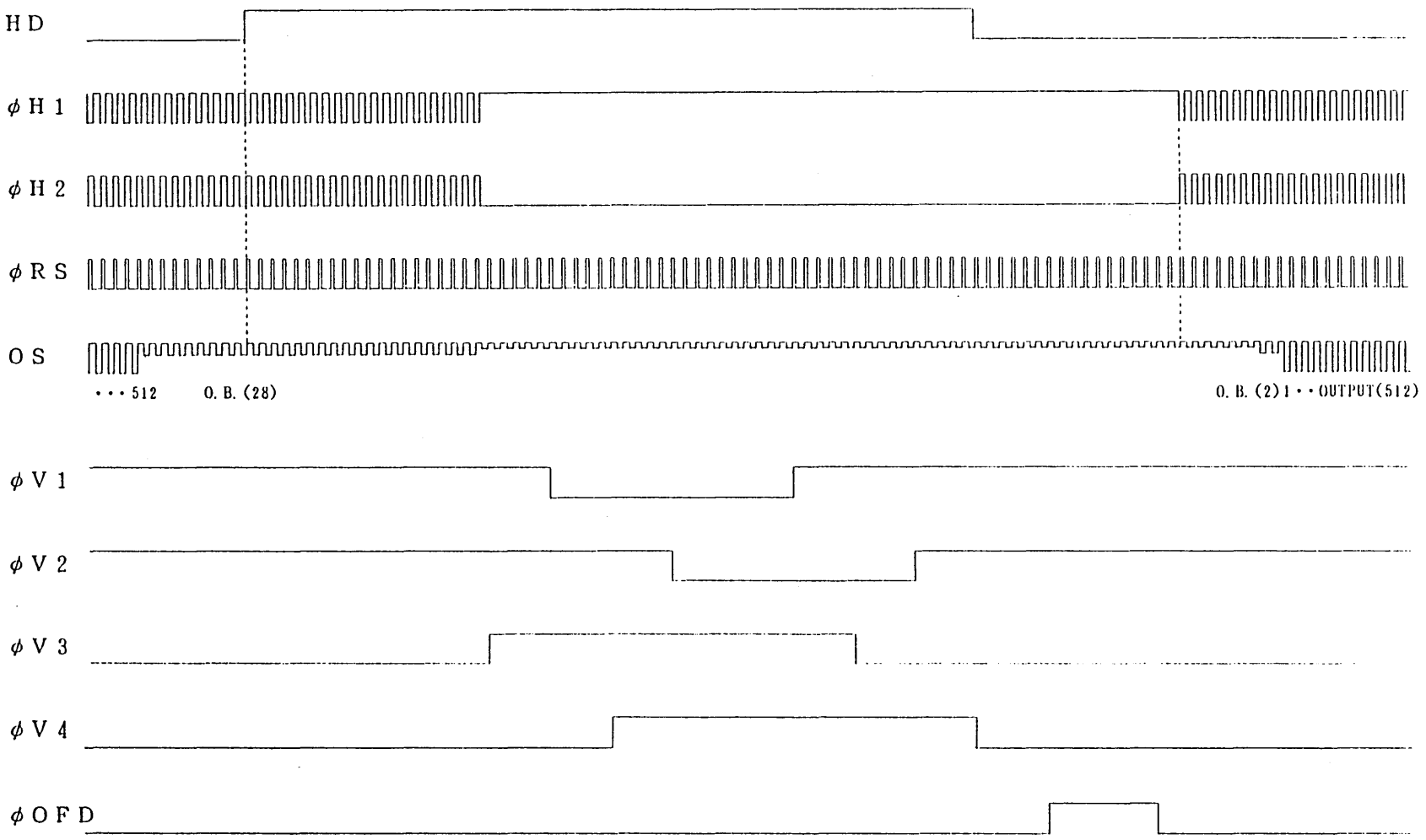
(j) The difference between the average output voltage of the (Mg+Ye), (G+Cy) line and the (Mg+Cy), (G+Ye) line under the standard exposure condition.

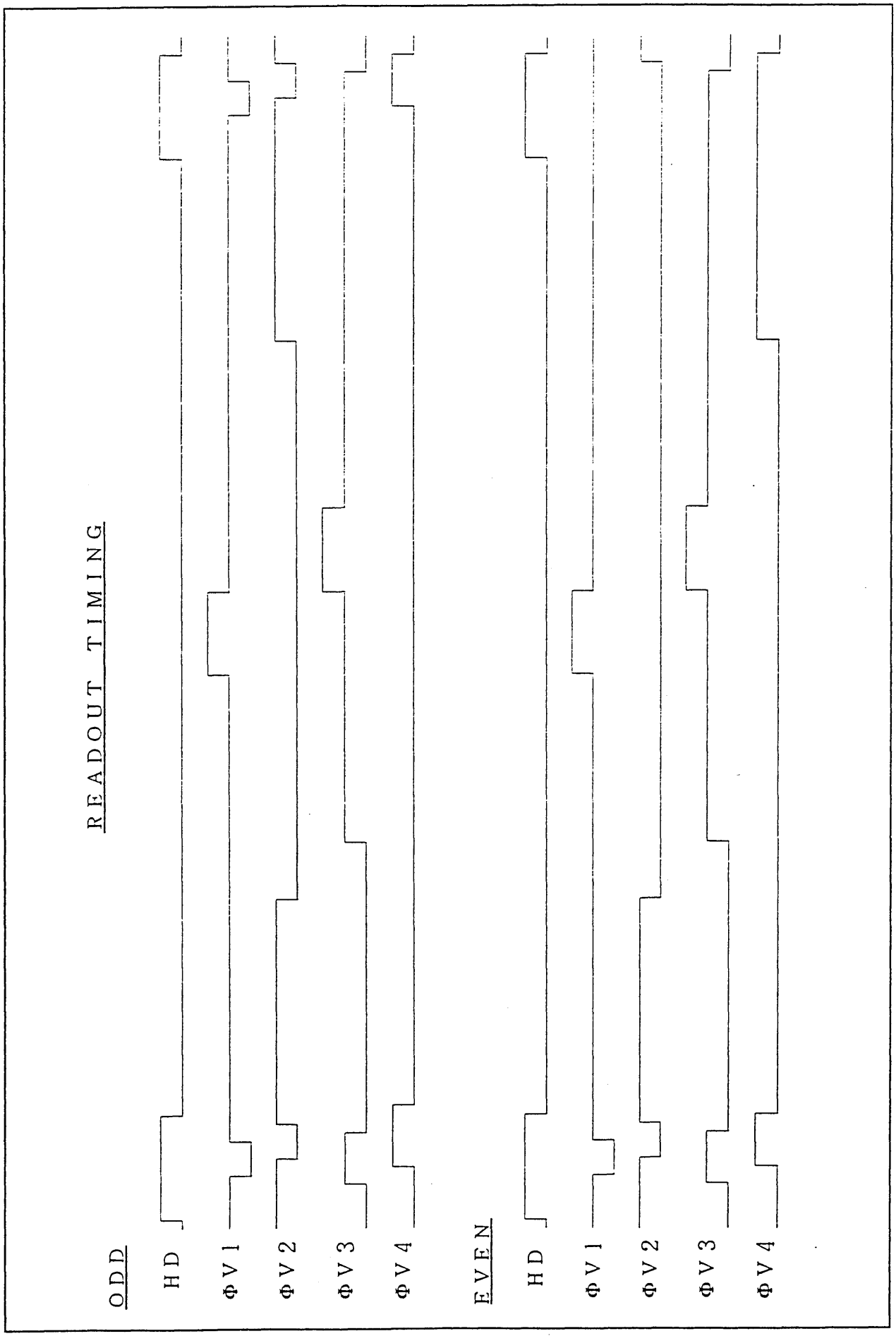
(k) The difference between the average output voltage of the odd field and the even field.

7. TIMING DIAGRAM EXAMPLE

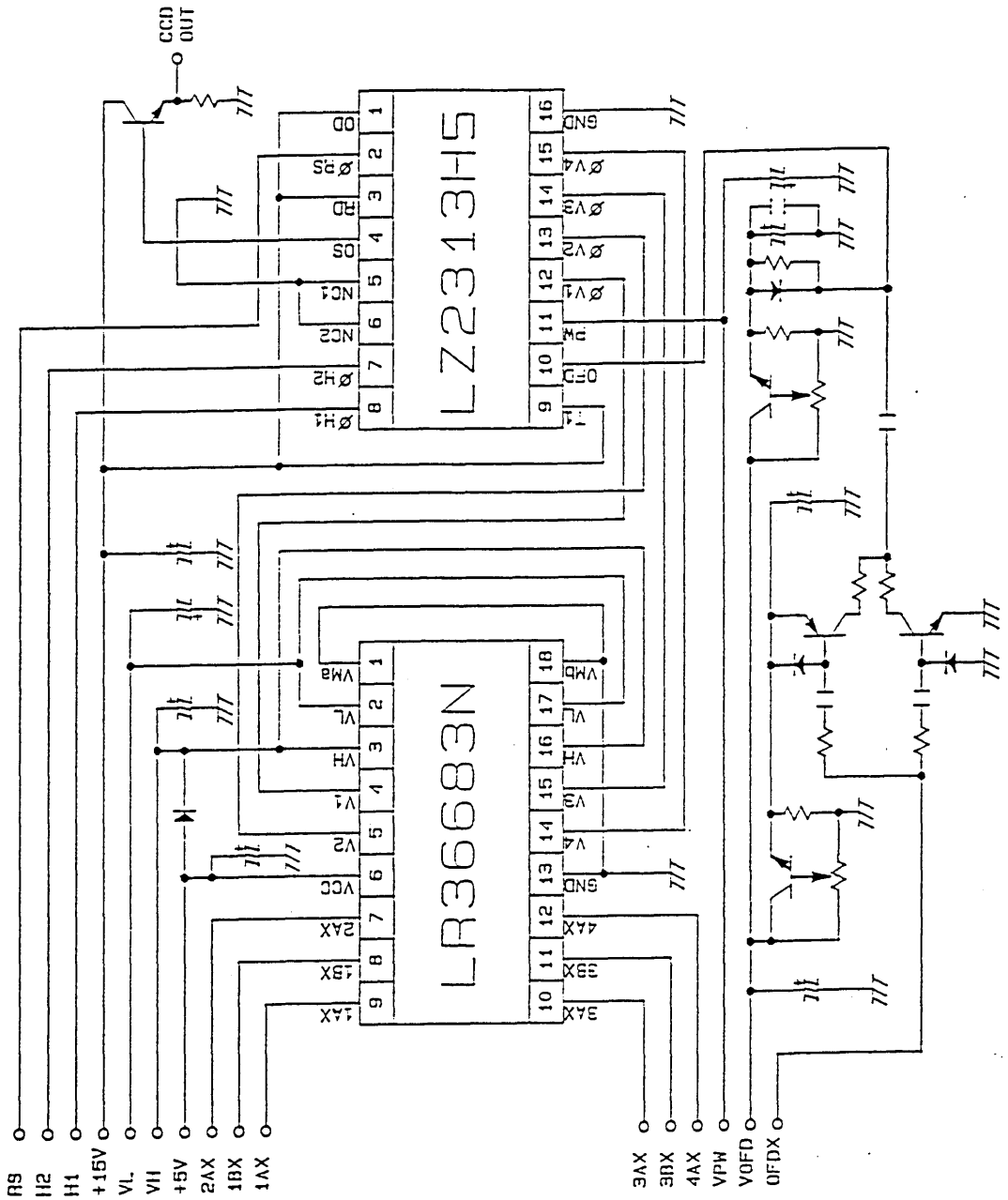


HORIZONTAL TRANSFER




READOUT TIMING
ODD
HD
ΦV1
ΦV2
ΦV3
ΦV4
EVEN
HD
ΦV1
ΦV2
ΦV3
ΦV4

8. STANDARD OPERATING CIRCUIT EXAMPLE



9. SPECIFICATION FOR BLEMISH

1) Definition of blemish

Blemish	Level of blemish (mV)	Permitted number of blemish		COMMENT
White blemish(I) (Exposed)	23 ≦ B	0		• B is defined in fig. 9-1(a). • Vout = Vstd. • M + N = 10, however, only 4 blemishes are permitted in area I shown in fig. 9-2.
	13 ≦ B < 23	M		
	B < 13	no count		
Black blemish(I) (Exposed)	23 ≦ B	0		
	13 ≦ B < 23	N		
	B < 13	no count		
White blemish(II) (Non-exposed)		AREA I	AREA II	• B is defined in fig. 9-1(b). • AREA I and II are defined in fig. 9-2. • Non-exposure condition. • Sum of the blemishes in AREA I and II are permitted 6.
	12 < B	0	0	
	9 < B ≦ 12	1	3	
	7 < B ≦ 9	2	4	
	6 < B ≦ 7	4	5	
	B ≦ 6	no count		
White blemish(III) (Shutter mode)	5.5 ≦ B	0		• B is defined in fig. 9-1(a). • Vout = Vstd/10. • The electronic shutter speed is set at 1/10000 s
	B < 5.5	no count		
Black blemish(III) (Shutter mode)	5.5 ≦ B	0		
	B < 5.5	no count		

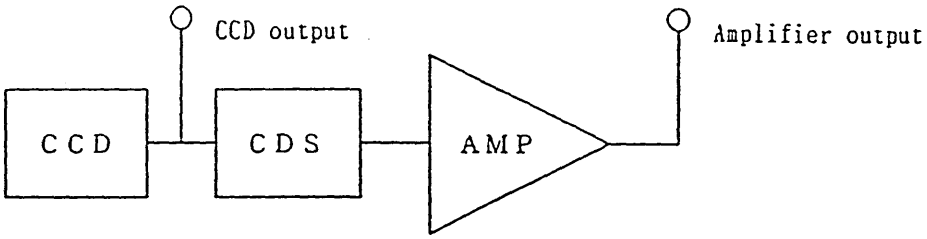
B :Blemish level defined in fig. 9-1.
Vout :Average output voltage
Vstd :150 mV. The standard output voltage defined in the specification of the characteristics.

2) Definition of stain

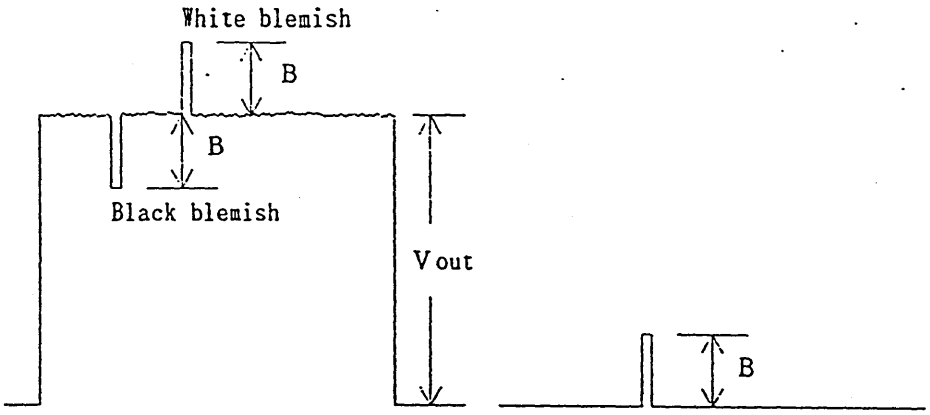
The measuring area is divided into segments which include 20 × 20 pixels, respectively. The difference between the average output voltage of neighboring segments is permitted below 1.5 mV, under the condition that the average output voltage of all imaging pixels is 75 mV(= Vstd/2).

【MEASURING CONDITION】

- Ta : 60 ℃
- Measuring block diagram



The output voltage is measured at the CCD output.
The gain of the amplifier is adjusted to the unity between the CCD output and the Amplifier output.



(a) Exposed (b) Non-exposed

fig. 9-1 Definition of the blemish level
(The wave form is the luminance signal measured at the Amplifier output)

【MEASURING AREA】

Measuring area includes all pixels image and optical black area excluding the outer 10 pixels of the left and right sides and the outer 9 lines of the upper and lower sides in the image area.

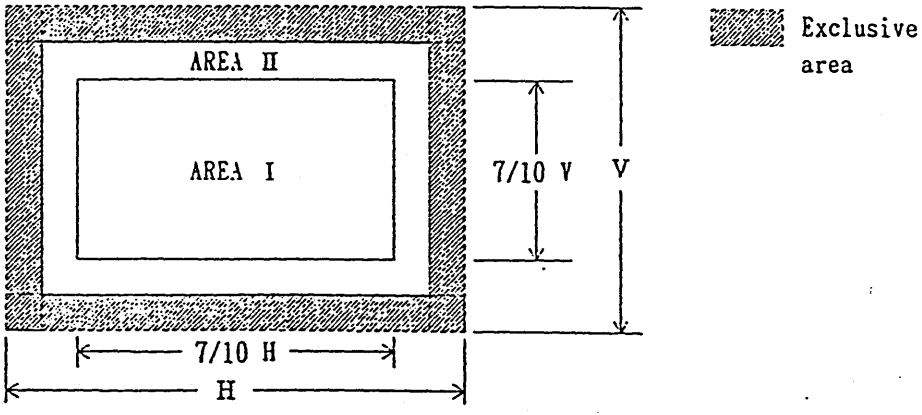


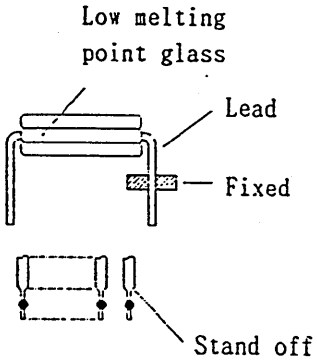
fig. 9-2 Definition of the measuring area

10. CAUTIONS FOR USE

10.1 Package Breakage

In order to prevent the package from being broken, observe the following instructions:

1) The CCD is a precision optical component and the package material is ceramic. Therefore,
   * Take care not to drop the device when mounting, handling, or transporting.
   * Avoid giving a shock to the package. Especially when leads are fixed to the socket and the circuit board, small shock could break the package more easily than when the package isn't fixed.
2) When applying force for mounting the device or any other purposes, fix the leads between a joint and a stand-off, so that no stress will be given to the jointed part of the lead. In addition, when applying force, do it at a point below the stand-off part.
   ••• The leads of the package are fixed with low melting point glass, so stress added to a lead could cause a crack in the low melting point glass in the jointed part of that lead.
3) When mounting the package on the housing, be sure that the package is not bent.
   ••• If a bent package is forced into place between a hard plate or the like, the package may be broken.
   Example for mounting
   * Place the buffers between the package and the housing.
   * Keep the bottom side of the package free.



4) If any damage or breakage occur on the surface of the glass cap, its characteristics could deteriorate. Therefore,
   * Do not hit the glass cap.
   * Do not give a shock large enough to cause distortion.
   * Do not scrub or scratch the glass surface.
     Even a soft cloth or applicator, if dry, could cause dust to scratch the glass.

10.2 Electrostatic damage

As compared with general MOS-LSI, CCD has lower ESD.
Therefore, please take the following anti-static measures when handling the CCD:

1) Always discharge static electricity by grounding the human body and the instrument to be used.
   To ground the human body, provide resistance of about 1 Meg ohm between the human body and the ground to be on the safe side.
2) When directly handling the device with fingers, hold the part without leads and do not touch any lead.
3) To avoid generating static electricity,
   a. do not scrub the glass surface with cloth or plastic
   b. do not attach any tape or labels
   c. do not clean the glass surface with dust-cleaning tape
4) When storing or transporting the device, put it in a container of conductive material.

10.3 Dust and contamination

Dust or contamination on the glass surface could deteriorate the output characteristic or cause a scar. In order to minimize dust or contamination on the glass surface, take the following precautions:

1) Handle CCD in a clean environment such as a cleaned booth.
(The cleanliness level should be, if possible, class 1000 at least.)
2) Do not touch the glass surface with fingers. If dust or contamination gets on the glass surface, the following cleaning method is recommended:
* Dust from static electricity should be blown off with an ionized air blower. For anti-electrostatic measures, however, ground all the leads on the device before blowing off the dust.
* The contamination on the glass surface should be wiped off with a clean applicator soaked in Isopropyl alcohol. Wipe slowly and gently in one direction only.
••• Frequently replace the applicater and do not use the same applicator to clean more than one device.

Note: In most cases, dust and contamination are unavoidable, even before the device is first used. It is, therefore, recommended that the above procedures should be taken to wipe out dust and contamination before using the device.

10.4 Cautions

1) Soldering should be manually performed within 5 seconds at 350℃ maximum at soldering iron.
2) Do not expose the device to strong light. For the color device, long exposure to strong light will fade the color of the color filter.
3) Avoid using or storing the CCD at high temperature or high humidity as it is a precision optical component. Do not give a mechanical shock to the CCD.
4) To apply power, first connect GND and then turn on OFD. After turning on OFD, turn on PW first and then turn on other powers and pulses.
Do not connect the device to or disconnect it from the plug socket while power is being applied.

CODE No.								1 / 12

CCD SOLID IMAGE PICKUP ELEMENT TEST METHOD

PICTURE-COMPONENT DIVISION
MATSUSHITA ELECTRONICS CORPORATION
QUALITY ENGINEERING SECTION

APPROVED BY	CHECKED BY	CHECKED BY	MADE BY
	G. Horii		T. Sugino

May. 29, '95					

Panasonic	Test Method	2/12

1. GENERAL DESCRIPTION

1.1 Optical System

1.1.1 Lens
- FUJINON CF-25L, F0.85, f = 25 mm

1.1.2 Optical Filter
- Infrared Rays Cutting Filter CAW-500, t = 2.5 mm, Made by Hoya Glass

1.1.3 Light Box
- Color temperature: 3200K (Use a halogen lamp 2856K and attach a color temperature conversion filter LB- 40 (made by Hoya Glass))
- Illumination: Refer to Sheet No.3

1.1.4 Test Chart
- J-chart ----- EIAJ J-chart, Without glass, 3/10 window in the vertical direction
- 5 % cut chart ----- Periphery of the H and V image frame cut by 5 %
- U-chart ----- J-chart partly light shut off
- Flicker chart
- DNP color bar chart
- DNP grey scale chart

1.2 Check Unit
- MEC-possessed exclusive test unit: Refer to a block diagram (Sheet No.12)
- Frequency characteristic: Flat within a band
- Contour high-light circuit: None
- Gamma compensation: None

1.3 Test Environment
- Operating temperature: 25 ± 2 ℃
- Ambient illumination: 300 lux or less

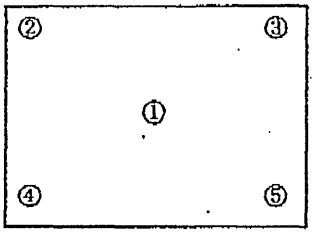
Panasonic	Test Method	3/12

2. LIGHT BOX

2.1 Light Source
- Halogen electric bulb light box

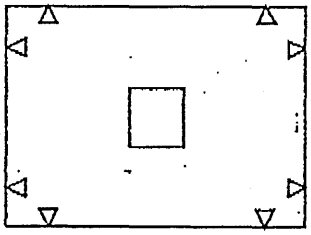
2.2 Pattern Surface Color Temperature
- 2856K± 20

2.3 Pattern Surface Illumination



① ・・・ 2900 ℓx ± 1 %
② ・・・ 2900 ℓx ± 5 %
③ ・・・ 〃
④ ・・・ 〃
⑤ ・・・ 〃

3. DEFINITION OF STANDARD CONDITION
- Lens: F:8 (f = 25 mm)
- Image angle: Picks up an image of the J-chart in a specified size.



J-chart

4. ADJUSTMENT OF ELEMENT
Adjust a measured sample as follows beforehand to make initial setting.

4.1 Initial Setting
- Items Set to Standard Values

V ϕ R	V Sub
V RD	V ϕ V1
V OD	V ϕ V2
V ϕ H1	V ϕ V3
V ϕ H2	V ϕ V4
V IS	V PW
V PT	

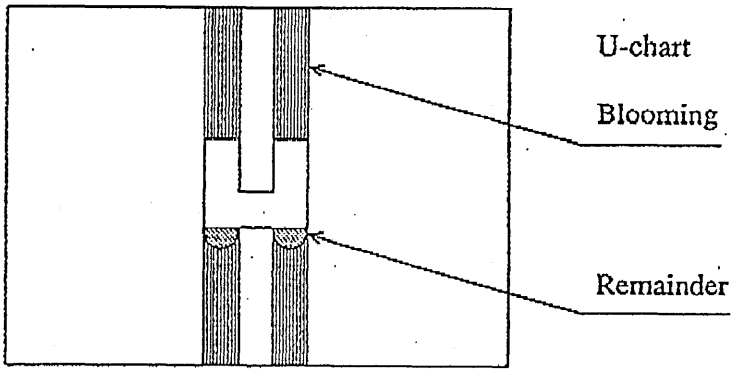
4.2 Items to Be Set to Optimum Values

4.2.1 Setting of VSub

(1) Initial setting is 8.0 V. Pick up an image of the U-chart at standard quantity of light × 100 and adjust VSub to a minimum voltage which does not generate blooming.

(2) After setting the above, when there is a remainder under the center of the U-chart on the monitor screen, increase a VSub voltage further to set to a minimum voltage where the remainder is removed.

(3) When roughness is generated under the minimum operating condition of VSub, set to a minimum voltage where no roughness is generated.



(4) When no blooming appears, set to 5.0 V.

5. ADJUSTMENT OF TEST UNIT

5.1 Initial Setting (Adjust once and no readjustment is required)

5.1.1 Adjustment of Burst Level

At TP4 (video output), adjust the Burst Level VR so that the burst level comes to a specified value after BPF.

5.1.2 Adjustment of Color Signal

In the standard condition(F:8), pick up an image of the DNP color bar chart with the standard element, and adjust the color signal system so that the specified condition can be obtained on the vector scope.

5.2 Adjustment to Optimum Values (Adjustment Required for Each Element)

5.2.1 Black Level Adjustment
Shut off a light and adjust the Black Level VR so that color balance at dark time can be obtained at TP4 (video output).

5.2.2 Top Gain Adjustment
Pick up an image of the J-chart in the standard condition (F:8) and adjust the Top Gain VR so that video output will come to a specified value at TP4 (video output).

5.2.3 White Balance Adjustment
Pick up an image without any charts in the standard condition (F:8) and adjust the YL (R-Y), YL (B-Y) Gain VR so that white will come to the center on the vector scope.

5.2.4 Chroma Level Adjustment
Pick up an image of the DNP color bar chart in the standard condition (F:8) and adjust the Chroma Gain VR so that degree of saturation for R will become 175 % with respect to an amplitude of color burst after BPF at TP4(video output).

6. CHARACTERISTIC CHECK

6.1 S/N

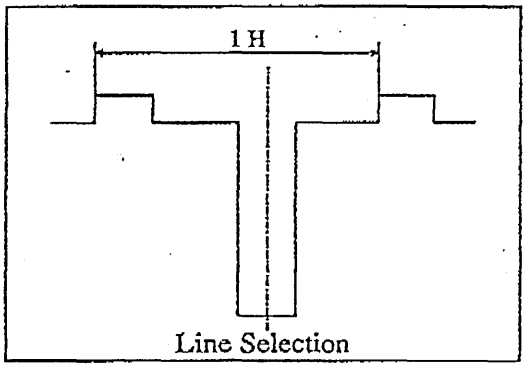
(1) Turn off a chroma signal.
(2) Shut off incident rays, connect TP4 (video output) to a noise meter, and measure S/N.
Noise meter conditions:
NTSC 100 kHz, 4.2 MHz LPF
PAL 100 kHz, 5.0 MHz LPF
S.C. trap ON

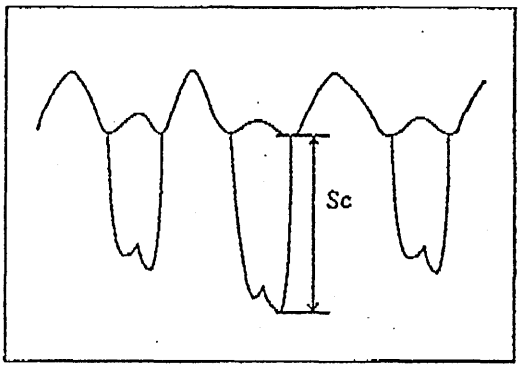
6.2 Carrier Saturation Output (Sc)

(1) Pick up an image of the J-chart at F:8 and observe TP3 (carrier) through an oscilloscope (H rate).

(2) Open a diaphragm from the standard condition of F:8 to search for a point where a carrier becomes maximum.

(3) Observe then TP1 (element output) through the oscilloscope (H rate) and line-select a higher signal level in the center of the screen and measure a larger amplitude.



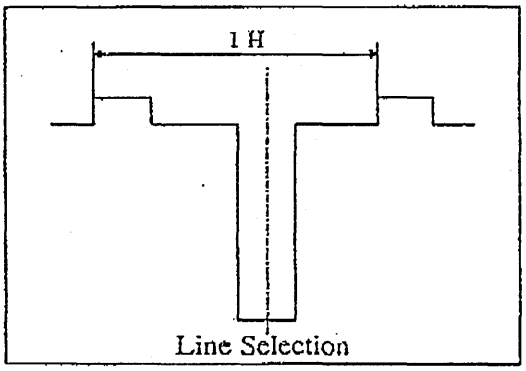


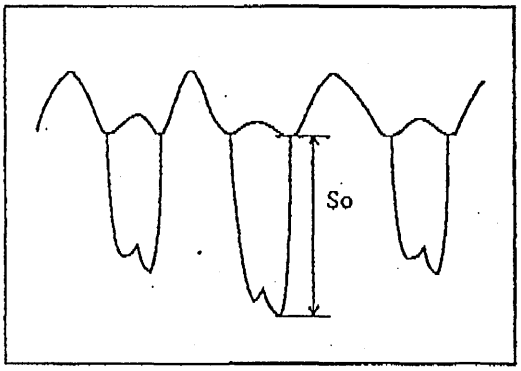
6.3 Sensitivity (So)

(1) Pick up an image of the J-chart in the standard condition (F:8).

(2) Observe TP1 (element output) through the oscilloscope (H rate) and line-select a higher signal level in the center of the screen.

(3) Measure a larger amplitude.





Panasonic	Test Method	7/12

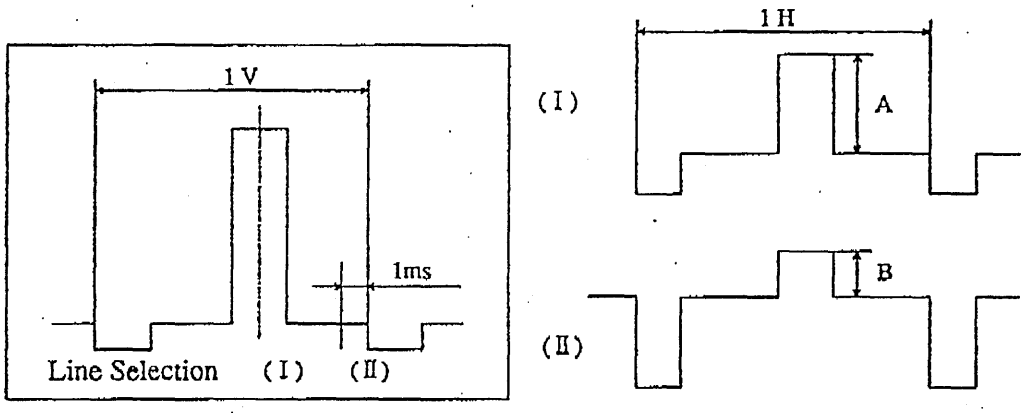
6.4 Vertical Smear (Sm)

(1) Pick up an image of the 1/10 chart in the standard condition of F:8.

(2) At TP2 (YH), Measure an output signal A in the center of the screen and a smear signal B in the 1ms section from the bottom of the screen through the oscilloscope (V rate).

(3) Obtain smear out of the following formula:

$$Sm = \frac{\text{Smear Signal (B)}}{\text{Output Signal (A)}} \times 100\ (\%)$$



Note 1) To measure the smear signal, use the average function of the oscilloscope to improve accuracy.

Note 2) When measuring the smear signal, turn off vertical read-out pulses.

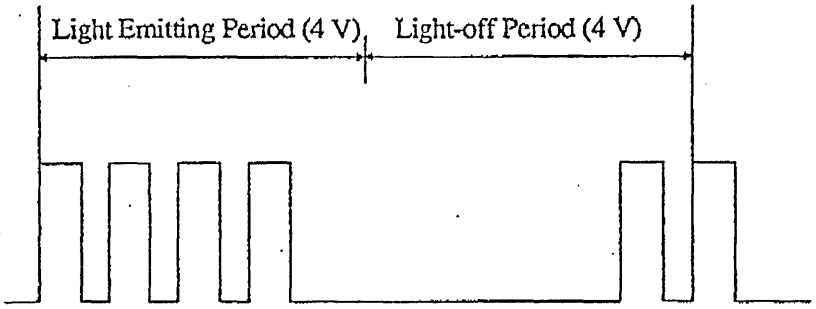
6.5 Image Lag (LAGd)

(1) Pick up an image of flickering light emitting diode (green). At TP2 (YH), adjust brightness of the light emitting diode so that the output level at LED ON time will become 1/8 of the signal level at J-chart standard image pickup time.

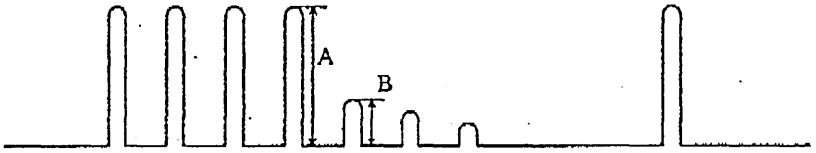
(2) Observe a TP2 signal through the oscilloscope (V rate), and measure a reference signal A, and a remaining signal B in the 1st field after the LED goes off.

(3) Obtain the residual image out of the following formula:

$$\text{Image Lag} = \frac{\text{Remaining signal (B)}}{\text{Reference signal (A)}} \times 100\ (\%)$$

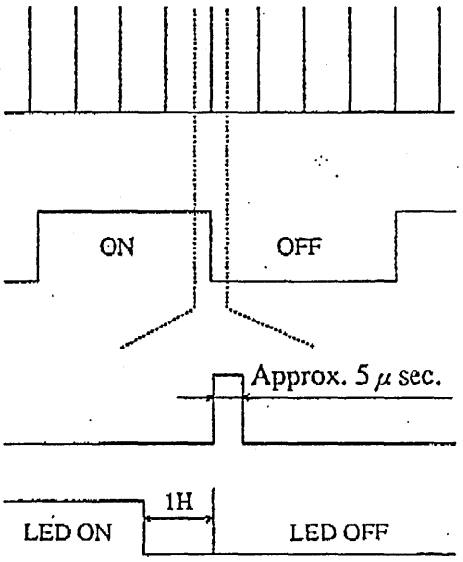


LED Applied Pulse



TP2 Output Signal

(4) LED ON and signal read-out pulse timing

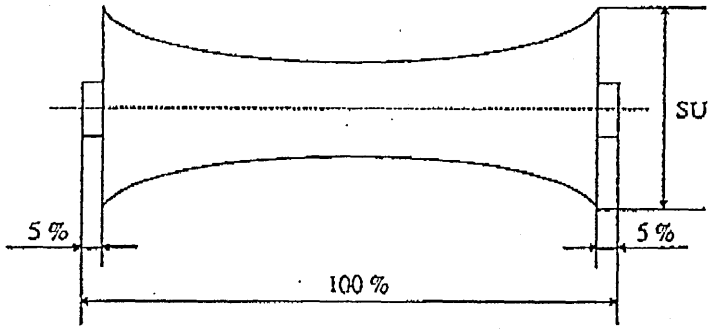


Photodiode read-out pulse

LED

6.6 Color Shading (SU)

(1) Pick up an image of the 5 % cut chart in the standard condition (F:8).
(2) Adjust white balance to the vector center.
(3) At TP4(video output), measure P-P of a signal after BPF through the oscilloscope (H, V rate)



6.7 Color Dynamic Range (DR)

(1) Pick up an image of the J-chart and observe TP3 (carrier) through the oscilloscope.
(2) Open the diaphragm from the standard condition (F:8) and obtain a point where the carrier becomes maximum.
(3) Measure the then element output's (TP1) value, A (mVp-p).
(4) Obtain DR out of the following formula.

$$DR = \frac{A}{\text{Standard output (200 mV)}} \text{ (times)}$$

6.8 Picture Blemish (PB)

(1) Narrow down by one graduation (F:11) from the standard condition and pick up an image without any charts.
(2) Measure the scar signal level for TP2 (YH).
(3) Scar evaluation is expressed in terms of contrast ratio (%) to the standard quantity of light of the measured value under the above-mentioned condition.

Panasonic	Test Method	10/12

6.9 Color Flicker

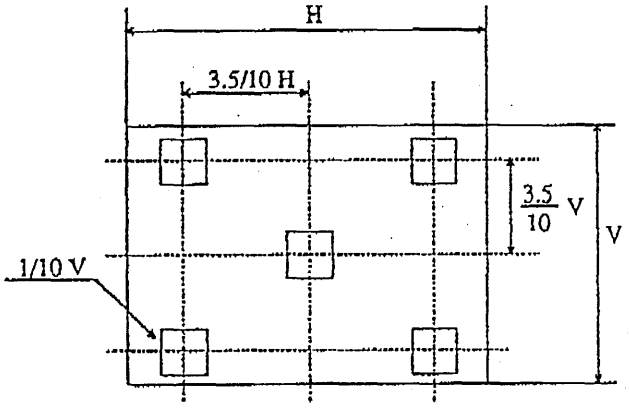
(1) Pick up an image of the flicker chart in the standard condition of F:8.

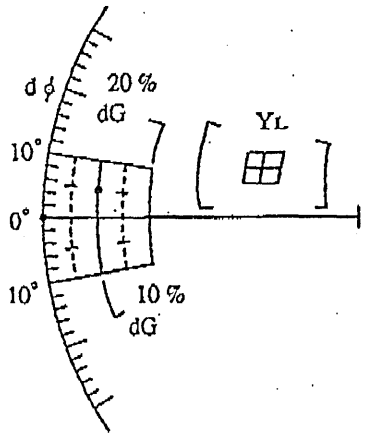
(2) Insert R, Ye, and Cy filters into 5 places in the screen, use the vector scope to observe partings of the amplitude and angle directions of the respective vectors. (Set the vector with larger amplitude to an outer circumferential position of 0 on the vector scope's scale to indicate a parted vector in terms of relative value.)

(3) Measure the worst point of the 5 points.

* 1) R, Ye, and Cy filters:
Use color glass filters.
Same characteristics as the DNP color bar chart.

* 2) Flicker chart





Example
[Amplitude difference: 10 %
Angle difference: 5 °]

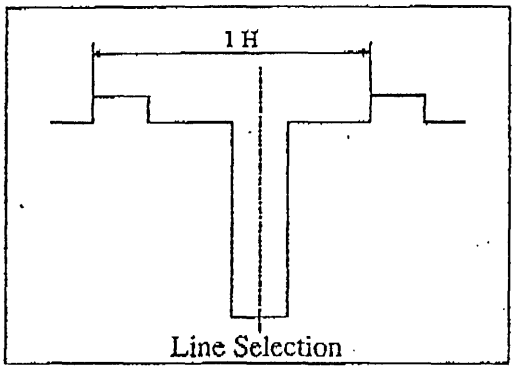
Panasonic	Test Method	11/12

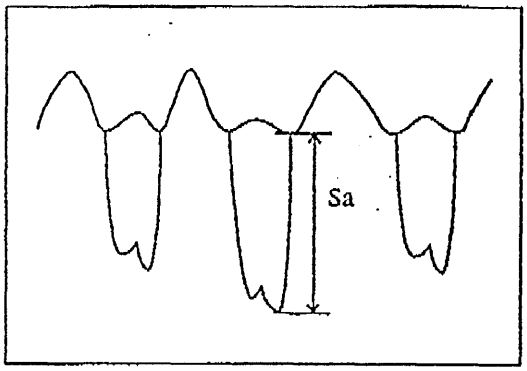
6.10 ϕR Bias Adjustment Method

(1) Pick up an image of the J-chart at F:1.4.

(2) Observe TP1(element output) through the oscilloscope (H rate) and line-select the signal level in the center of the screen

(3) Lower a ϕR bias voltage from around 5 V to adjust to a point where element output (Sa) becomes maximum.

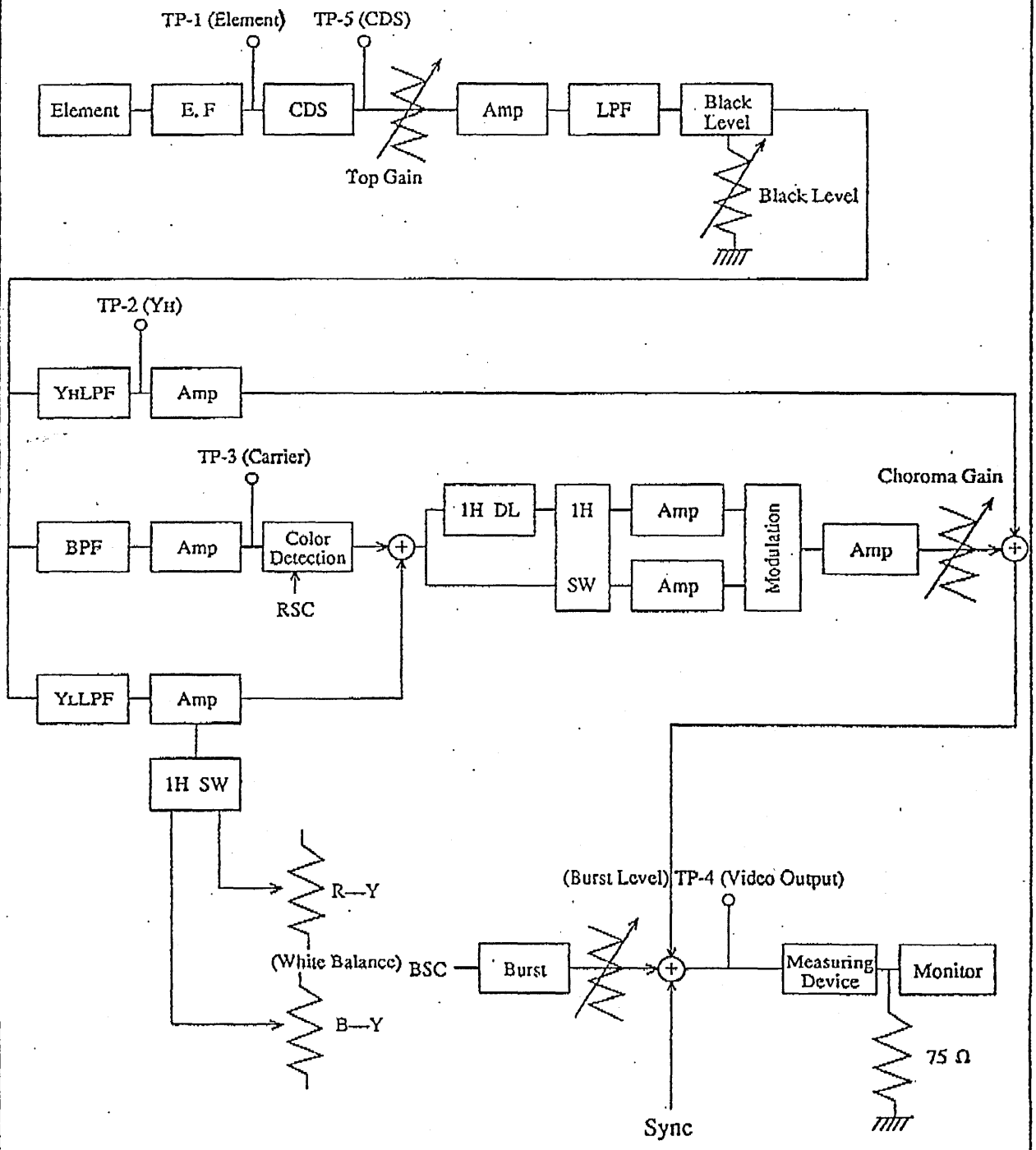




Panasonic	Test Method	12 / 12

Measurement Circuit Block Diagram

Note: No γ compensation,
No aperture compensation



APPLICATION NOTE

"CamKIT III" FOR OEM DEVELOPERS NOW AVAILABLE!

The new CamKIT III™ from Teleview Research is a complete development tool for interfacing the CDMC-270D8 color digital video camera to any PC-based host computer board or system. It provides the video engineer with a simple means to "patch" the supplied digital video camera into most existing video capture systems: RGB, Composite and "S" Video, or Philips 7191 digital format. The user may select from two camera sources, switchable on the board. Stereo audio is also buffered and routed to line outputs through common RCA jacks The kit includes all necessary camera hardware and control software to produce and control color video images on the developer's monitor.

INCLUDES

- CDMC-270D8 color digital video camera (board-level module)
- Interconnect cable from digital camera to prototyping board
- Camera interfacing/prototyping board with schematics
- Video output cables from prototyping board to RGB, Composite or "S" Video monitor ports.
- Parallel port cable from prototyping board to host computer
- I^2C Standard Interface Protocol Description
- Application notes, guidelines and installation diagram
- CamSOFT™ camera control software

FEATURES

- Dual connectors for digital camera 26-pin sub-D inputs
- RGB / Sync outputs for RGB monitor on standard VGA connector
- Y/ C outputs for S-Video monitor on mini-DIN connector
- Composite Video output available on RCA jack

APPLICATION NOTE

- Connectors for Phillips 7191 format (or equiv.) input and output
- I^2C camera control interface on DB-25 connector (to LPT port)
- User-friendly menu-driven camera control software (on 3 1/2 " disk)
- Programmable gamma correction (standard curve used as default)

COST * $1995 each plus applicable tax (includes 1 year warranty)
(*your CamKit purchase will be rebated with your first OEM order!*)

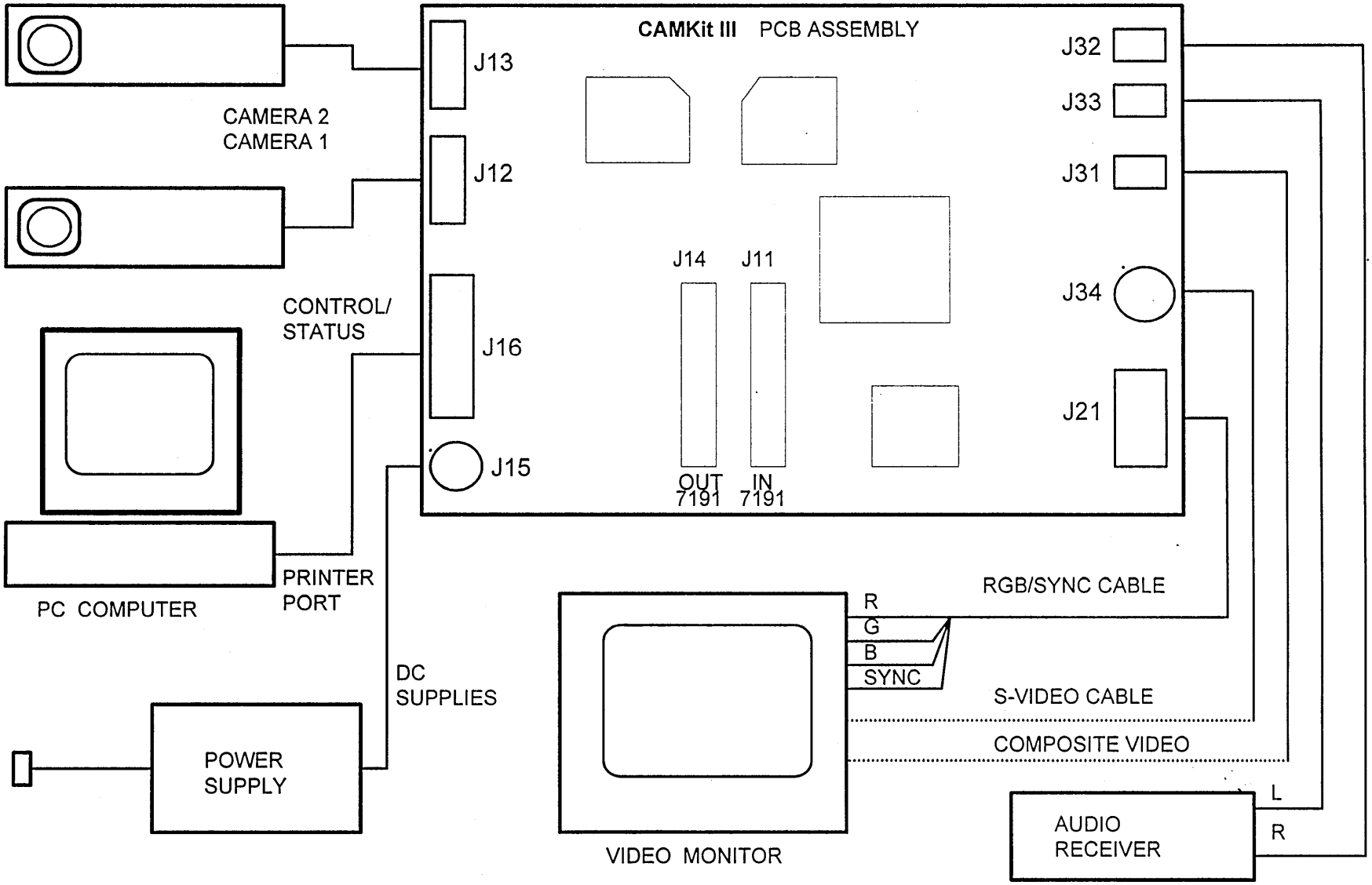
WHAT TO ORDER Ask for the **"CDMC CamKIT III"**

WHERE TO ORDER
Teleview Research, Inc.
4005 Miranda Ave., Suite 150
Palo Alto, CA. 94304
TEL 415-855-0900 FAX 415-855-9363

SHIPPING Within 2 weeks after receipt of your order (please specify carrier)

* Price is FOB Palo Alto, CA. and is subject to change without notice.

"CamKIT" and "CamSOFT" are trademarks of Teleview Research, Inc. of Palo Alto, CA.



CamKIT III Interconnection Block Diagram

Televiewe Research, Inc. 4005 Miranda Ave, Palo Alto, 94304 TEL 415.855-0900 FAX 415.855-9363

CamKIT-III APPLICATION NOTE

CamKIT-IIIOEM DEVELOPER "Camera kit"

For CDMC-270D8 Color Digital Video Camera

1.0 GENERAL

The CDMC CamKIT-III is designed to aid the OEM in the development of a digital camera system. This kit includes the CDMC-270D8, a board-level color digital video camera module, as well as a CamKIT-III interface PCB assembly. All necessary interconnect cables connecting the CamKIT-III to the camera, the monitor and the PC computer are included. Software to control the CamKIT-III via PC computer is supplied on a 3 1/2" floppy disk.

2.0 INTERFACE FEATURES

This kit features a variety of interfaces and provides the developer with the flexibility to experiment using the following interconnects:

- Camera and 5-Ft interconnect cable
- Dual Connectors for Digital Camera (26-pin D-sub)
- Connectors for Philips 7191 format input and output
- I^2C Camera control interface signals on output connector
- Connector and cable to PC computer's parallel printer port
- Connectors and cables to RGB monitor
- Connectors and cables to S-Video monitor
- Connectors and cables to Composite video monitor
- Connectors and cables to stereo audio RCA jack LINE inputs

3.0 THEORY OF OPERATION

The RP125A FPGA (U1) demultiplexes the 125M camera data from J12 and J13 into Y and UV data streams which go the the output connector J14. It also decodes the horizontal, vertical and field information and produces the 7191 timing signals HREFOUT (Horizontal Reference), HSYNCOUT (Horizontal Sync) and VSYNCOUT (Vertical Sync). The Y and UV outputs of the FPGA can be disabled so that the data from the input connector J11 drives the output connector J14.

The signal SELCHA is controlled by section 1 of the DIP switch SW11. This signal selects which of the two 125M camera data streams is decoded by the RP125A FPGA. When this signal is high the data from camera A (J12) is selected. When this signal is low the data from camera B (J13) is selected. This signal also controls the stereo audio source and which cameras I^2C clock line is enabled.

The signal SEL8N16 is controlled by section 2 of the DIP switch. This signal disables the decoded output of the RP125A FPGA and enables the output control INCDMC of the 7191 input connector J11. It also routes the timing signals from J11 to the timing signal outputs of the FPGA. When this signal is "0" the data from the input connector J11 is selected. When this signal is "1" the data decoded from the 125M camera is selected.

The signal OUTCDMC controls the output enable of the Y and UV bus drivers. When this signal is low the Y and UV outputs of the FPGA are disabled and the signal INCDMC is driven low.

The Y and UV data busses are connected to the SAA-7192A-WPA Color space converter (U2) This chip converts the luminance and chrominance information into red, green and blue video data. The data is clocked by the DCLK signal from the FPGA. The signal HREF controls the blanking of the red, green and blue outputs.

The red, green and blue digital data is converted to analog signals by the SAA-7169-WPA triple DAC, whose outputs are filtered to provide external NTSC monitor drive.

The FPGA outputs a composite sync signal called COMPSYNC. This signal is produced from the horizontal and vertical sync information of the selected video source. This signal controls the sync driver Q21 which drives a 1V p-p signal to the NTSC monitor. The RGB and SYNC signals are output to J21, which is a standard 15-pin D-sub VGA connector. (Use the VGA-to-BNC cable to connect to monitor's BNC input jacks).

The red, green and blue digital data is also input to U5 (SAA7199B), which converts this data to composite video and Y (Luma) as well as C (Chroma) signals. These signals inherently have high frequency components in them which are filtered out with low-pass anti-alias filters. The filtered composite video is output at J31 (RCA jack) and the filtered Y/C signals are output at J34 (S-Video mini DIN jack).

For applications utilizing mono or stereo audio, the selected source (from either J12 or J13) is made by the SELCHA selection of SW11-1. The selected (LINE OUT) audio outputs are through phono jacks J32 and J33 for use as inputs to any standard audio amplifier with LINE inputs.

DC power into this circuit board is provided from an external triple-output supply which connects to the board via a standard 5-pin DIN connector. The +5 volt and +/- 12 volt inputs are fused by F211, F213 and F212, respectively. Voltage present is indicated by LED1 (+5V), LED2 (-12V) and LED3 (+12V).

The SAA-7192A-WPA and the CDMC-170D8 camera are controlled by their IIC interfaces: The CamSOFT software running on a PC can control these devices when connected via parallel printer-port cable. (See CamSOFT Digital Video Camera Control Software Application Notes for software operational details)

4.0 INSTALLATION

Installation of the various components which comprise this digital camera development system may be accomplished by observing the Installation Block Diagram included as part of this documentation package. This Block Diagram depicts the connections necessary to build a basic color video system.

5.0 DOCUMENTATION

The following documentation is supplied as part of this kit:

- Schematic Diagrams of the Camera and the CamKIT-III PCB Assemblies
- Block Diagram of a basic digital color camera system

- I²C Protocol: Refer to attached **I²C Protocol for CDMC Revision 1.6** for camera control commands. Refer to the Phillips Desktop Video Handbook for controlling Phillips devices.
- **"CamSOFT" Rev 2.0** Digital Video Camera Control Software Application Notes

6.0 ELECTRICAL REFERENCE DATA

- **Power** : Power is supplied to the CamKIT-III via the 5 pin connector J15 or through pins 2, 4 and 6 of the 7191 format connector J11.
- **Inputs** : Two camera inputs (J12 & J13). One 7191 format digital video input (J11). CamKIT-III control on 25-pin D-sub connector J16
- **Outputs** : 7191 format digital video output connector (J14). Red, Green, Blue and Sync signals on 15-pin D-sub connector (J21). Composite video output at RCA jack (J31). S-video at 4-pin Mini-DIN connector (J34). Audio outputs at phono jacks (J32, J33). Control link status through 25-pin D-sub connector J25.
- **Pinout Data** : Refer to SCHEMATIC, CAMKIT III, Schematic Diagrams (Sheets 1-3) for further connector pinout data.

7.0 "CamSOFT" CONTROL SOFTWARE SUPPLIED

Televiev Research supplies a 3 1/2" floppy disk containing all executable, source and reference files needed to control the camera kit from a PC computer. Please refer to the **read.me** file on this disk for further information and/or instructions regarding these files.

8.0 REFERENCES

1. CCIR REC 601-2 standard : Encoding parameters of Digital Television for studios, 1990
2. I²C bus specification: Philips Desktop Video Handbook, Page 2-185,1993
3. Philips 7191 data sheet: page 3-191 of Philips Desktop Video Handbook
4. Philips 7199 datasheet: page 2-72 of Philips Desktop Video Handbook
5. ANSI/SMPTE 125M Specification, July 1992
6. Televiev CDMC-270D8 data sheet
7. Televiev IIC control protocol for CDMC-270D8, Rev 1.6, 8/31/93
8. Televiev CamSOFT Application Notes, Rev 2.0, 2/25/94

CamSOFT APPLICATION NOTE

"CamSOFT" CAMERA CONTROL SOFTWARE Rev 2.0

For CDMC-270D8 Color Digital Video Camera

1. Introduction

CamSOFT is a software package developed by Teleview Research for control of their Digital Video Camera. The software is hosted on a IBM AT compatible computer and requires a parallel printer port. The user has full control of the camera through a simple menu driven front-end User Interface, details of which are documented below.

2. System Requirements

Software (Teleview - provided):

- camsoft.exe
- gamma.vlt
- read.me
- Source Code File(s)

Hardware (User - provided):

- IBM AT compatible computer
- 25 pin male D connector type parallel interface (LPT)
- at least 384K RAM
- at least one 3.5" floppy disk drive
- MS-DOS rev 3.0 or higher
- 286/386/486 or higher processor - type

3. Installation

While CamSOFT can operate on a floppy disk based-system, a hard disk based system is preferred. Assuming a hard disk installation, perform the following steps:

- Power-on the computer.
- Upon loading DOS and receiving the DOS PROMPT, insert the 3.5" floppy disk into your 3.5" drive (e.g.- drive B:).
- Copy all the files from the floppy disk to the subdirectory of choice on the hard disk.
- Connect the supplied parallel cable to the 25-pin D connector of the computer's parallel (printer) port and to the 25-pin D connector on the CamKIT board.
- Connect the remaining interface equipment (monitor, power supply, etc.) as shown in the CamKIT Installation Block Diagram.
- Power up the remaining equipment and start-up the software by typing "camsoft" and pressing the ENTER key.

CamSOFT APPLICATION NOTE

- The software will first display a cover page, initialize the hardware, white balance the camera, and then put up the Command Menu to the CRT screen.
- The software is now ready to use.

4. Operation

The software automatically initializes the hardware before displaying the Command Menu. The monitor should display a color correct picture that has been White Balanced for existing light conditions. This presumes that the imager was properly focused onto a white surface. If this was not the case, refocus onto a white surface and perform another white balance ('AW'). This will be the known starting point from which the user may examine and/or change register contents to modify the camera's operational performance.

The commands to modify the camera's operation are as follows:

Command	Definition
AG [on|off]	**A**utomatic **G**ain Control register. Turning AGC off implies MGC mode of operation.
AW	**A**utomatic **W**hite Balance register. Invoking this command causes a White Balance to be performed by the camera. You should have a sheet of white paper for the camera to focus on BEFORE performing this.
IN	**IN**itialize all the camera registers. Default values will be loaded into the registers. While this will put the camera close to normal mode of operation, we recommend you also perform a White Balance.
LU [file]	Load video ram **L**ook-**U**p table ram. You should specify which **vlt** file to load. If you do not, the default file loaded shall be **gamma.vlt**
ST	Read and display all register **ST**atus. The contents of each of the registers shall be read out and displayed on a single line at the bottom of the screen. To stop the display, press any key except **<esc>**. To continue, press any key again. To exit out of **ST**atus read, press **<esc>**.
VE	Read the **VE**rsion number of the hardware. This number is hard-coded into the ASIC on the CamKIT board.

CamSOFT APPLICATION NOTE

Command	Definition
BB [xx]	Read or set the **B**lue **B**alance register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
BS [xx]	Read or set the **B**lue **S**aturation register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
GN [xx]	Read or set the manual **GaiN** register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
RB [xx]	Read or set the **R**ed **B**alance register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
RS [xx]	Read or set the **R**ed **S**aturation register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
SH [xx]	Read or set the **SH**utter speed register content. If no hex number is specified, a read register operation is performed. If a hex number is supplied, then that number is written into and becomes the new value of the register.
HE [on|off]	**H**elp **E**nable controls the automatic re-display of the command menu
EX	**EX**it the program. This command terminates the program and returns control of the computer back to DOS.

CAMSOFT Camera Control Software Rev 2.10L

For CDMC-270D8 Color Digital Video Camera

April 25, 1994

1. Introduction

CAMSOFT is a software package for control of the Digital Video Camera. The software is hosted on a IBM AT compatible computer and requires a parallel printer port. The user has full control of the camera through a simple DOS program or a control panel under Windows. The sources for the software are included.

2. System Requirements

- IBM AT compatible computer
- 25 pin male D connector type parallel interface (LPT)
- at least 384K RAM
- at least one 3.5" floppy disk drive
- MS-DOS rev 3.0 or higher
- 286/386/486 or higher processor - type

The sources are written in C. They have been compiled with Microsoft Visual C++ but it should be a simple matter to compile them with another C compiler for DOS or Windows. The Windows version is to be compiled into a .DLL. The Windows user interface is implemented in Visual Basic, with calls to the .DLL for the camera control.

3. Installation

While CAMSOFT can operate on a floppy disk based-system (DOS version), a hard disk installation is preferred. An installation program is included to facilitate the process.

Software Installation

- Insert the distribution disk in drive A: or B:
- Type A:INSTALL or B:INSTALL

  The installation defaults to C:\CAMKIT for the destination directory, making the following subdirectories:

DOS	for the DOS program and make file
WINDOWS	for the Windows programs and make files
SRC	for the C source common to DOS and Windows

Hardware Setup and Operation:

- Connect the supplied parallel cable to the 25-pin D connector of the computer's parallel (printer) port and to the 25-pin D connector on the CAMKIT board.

- Connect the remaining interface equipment (monitor, power supply, etc.) as shown in the CAMKIT Installation Block Diagram.
- Power up the remaining equipment and start-up the software by typing CAMSOFT from the C:\CAMKIT directory and pressing the ENTER key.
- The software will first display a cover page, initialize the hardware, white balance the camera, and then put up the Command Menu to the CRT screen.
- The software is now ready to use.

4. DOS Operation

To start the DOS version, change to the DOS subdirectory and call CAMSOFT:

C:

CD \CAMKIT\DOS

CAMSOFT

The software automatically initializes the hardware before displaying the Command Menu. The monitor should display a color correct picture that has been White Balanced for existing light conditions. This presumes that the imager was properly focused onto a white surface. If this was not the case, refocus onto a white surface and perform another white balance ('AW'). This will be the known starting point from which the user may examine and/or change register contents to modify the camera's operational performance.

The commands to modify the camera's operation are as follows:

Command	Definition
AG	Reads Automatic Gain Control status bit (AGCENA)
AG ON	Turns Automatic Gain Control ON
AG OFF	Turns Automatic Gain Control OFF The Automatic Gain Control status bit (AGCENA) is bit 0 of the Control/Status register (0). See the effect of AGC by turning it ON, then use the ST command to see how the Gain register value changes with various subjects (try dark ones vs. light ones). See how the image is affected by turning AGC OFF and changing the subjects from dark to light. The Gain register value remains constant after you turn AGC OFF. The Gain register value can be changed with the GN command.
AW	Performs Auto White Balance The Automatic White Balance Control bit (BCTL) is bit 1 of the Control/Status register (0).

Sets the BCTL bit to initiate the Automatic White Balance procedure, and waits for the same bit to clear (indicating the end of the procedure).

Put a white object in front of the camera and issue this command to perform a white balance for the ambient light. Notice the effect on the Blue Balance and Red Balance register values by issuing BB and RB commands.

BB Reads and displays the Blue Balance register (5)

BB xx Sets the Blue Balance register

xx is a hexadecimal value between 0 and FF.

BS Reads and displays the Blue Saturation register (7)

BS xx Sets the Blue Saturation register

xx is a hexadecimal value between 0 and FF.

EX EXit from CAMSOFT (back to DOS).

GN Reads and displays the Gain register (2)

GN xx Sets the Gain register (and turns AGC OFF)

xx is a hexadecimal value between 0 and FF.

HE Displays the help menu.

HE ON Turns the help menu display ON (menu is shown every time a command is issued).

HE OFF Turns the help menu display OFF (menu is not shown unless a HE command is issued).

IN INitialize all the camera registers. Default values will be loaded into the registers. While this will put the camera close to its normal mode of operation, it is better to then perform a White Balance operation (AW command).

LU [file] Load video Look-Up table RAM. You should specify which vlt file to load. When no file name is given, the file GAMMA.VLT is used. GAMMA.VLT is also automatically loaded when starting CAMSOFT. The file format is:

5 bytes: "VLUTV" (ignored)

256 bytes: Red data

256 bytes: Green data

256 bytes: Blue data

RB	Reads and displays the Red Balance register (4)
RB xx	Sets the Red Balance register xx is a hexadecimal value between 0 and FF.
RS	Reads and displays the Red Saturation register (6)
RS xx	Sets the Red Saturation register xx is a hexadecimal value between 0 and FF.
SH	Reads and displays the Shutter Speed register (1)
SH xx	Sets the Shutter Speed register xx is a hexadecimal value between 0 and FF.
ST	Repeatedly reads and displays all registers until the user hits the ESC key. Reads all registers and displays their values on the screen. Repeats itself until the ESC key is pressed. This is an easy way to see all the register values at once. When the Automatic Gain Control is ON, you can observe the changes on the Gain register as the subject changes in front of the camera. This is the only function available to read the value of the Brightness register.
VE	Reads and displays the VErsion number of the hardware. This number is hard-coded into the ASIC on the CAMKIT board.

5. Windows Operation

Operation under Windows is made possible through CAMSOFT.DLL and the control program CAMCTRL.EXE. These two files are copied to the hard disk and can be called from the Windows Program Manager.

From the Program Manager, you can start CAMCTRL by choosing the Run command in the File menu. Then type C:\CAMKIT\WINDOWS\CAMTRL in the Run dialog to start. You can also include CAMCTRL in one of the Program Manager groups by choosing the New command in the File menu. Make sure to specify C:\CAMKIT\WINDOWS as the working directory.

As long as the CAMSOFT.DLL file is in the same directory as CAMCTRL.EXE, there won't be any problem. When running Visual Basic in interpreted mode, you may need to copy CAMSOFT.DLL to the Windows SYSTEM directory.

6. Source Code: CAMSOFT.EXE and CAMSOFT.DLL

The same C source is used to produce the DOS version and the Windows DLL. They have been compiled with Microsoft's Visual C++, using the CAMSOFT.MAK files under the DOS and the WINDOWS sub directories. To compile the Windows version, define the WINDOWS symbol (-D WINDOWS) and link with a library for .DLL executables. It is usually simple to compile and link the DOS version from any C compiler and linker for DOS.

The source code consists of three files:

CAMSOFT.C contains the main entry point and command interpreter.

PHIL_IO.C contains the functions for the configuration of the CAMKIT chips, including downloading the gamma table.

CAM_IO.C contains the functions for the configuration of the digital video camera.

IIC_IO.C contains the low-level functions implementing the I2C interface through the printer port.

7. Source Code: CAMCTRL.EXE

CAMCTRL.EXE is produced from Visual Basic. You will find the necessary files under the WINDOWS sub directory. Open the CAMCTRL.MAK file to access the objects. When running in interpretive mode of Visual Basic, the CAMSOFT.DLL must be found in the Windows SYSTEM sub directory.

CAMSOFT.DLL offers two entry points of interest to the Visual Basic program. CamCmd is a function that accepts the literal commands in the style of the DOS program (2-letter codes and a parameter). CamInit is a function to be called at the beginning of the operation and whenever the Reset button is clicked on. It performs the same initialization as when starting the DOS program.

8. Files

The following files are copied to the CAMKIT subdirectories:

in CAMKIT\DOS:

CAMSOFT.MAK	make file to compile the C files from CAMKIT\SRC to CAMSOFT.EXE
CAMSOFT.EXE	DOS program
GAMMA.VLT	Gamma table (read and downloaded by CAMSOFT)

in CAMKIT\WINDOWS:

CAMCTRL.MAK	Make file for Visual Basic
CAMCTRL.EXE	Windows program
THREED.VBX	Visual Basic object file
CD.ICO	Icon file
CCABOUT.FRM	About box form (for Visual Basic)
CAMCTRL.FRM	Main form (for Visual Basic)
CAM.ICO	Icon file

CAMCTRL.VCW	Visual Basic file
CAMCTRL.WSP	Visual Basic file
VBRUN300.DLL	Visual Basic run-time file
GAMMA.VLT	Gamma table (needed at run-time)
CAMSOFT.DEF	.DEF file to build CAMSOFT.DLL
CAMSOFT.DLL	Control functions .DLL called from CAMCTRL
CAMSOFT.MAK	Visual C++ Make file to build CAMSOFT.DLL

in CAMKIT\SRC:

CAM_IO.C	Camera module
IIC_IO.C	I2C interface (through parallel port) module
PHIL_IO.C	Philips chip interface module
CAMSOFT.C	Main (user interface for DOS, control function for Windows)
CAM_IO.H	
IIC_IO.H	
PHIL_IO.H	
NOWINDOW.H	