General Description:

The 6567 Video Interface Chip (VIC II) is a multi-purpose video controller for use in both computer video terminals and video game applications. The color video display is 25 rows of 40 characters each with additional features including horizontal and vertical scroll, bit map graphics and movable image blocks (MIBS).

Features:

- Mask programmable sync. generation, NTSC = 6567, PAL = 6569, PAL-N=6572, PAL-M=6573.
- 47 Addressable control registers.
- Fully expandable system with up to 16K byte display memory.
- On chip color generation (16 colors).
- Up to eight independent movable image blocks (MIBS).
- Light gun/pen for target games.

Pin Configuration

<table>
<thead>
<tr>
<th>Pin</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB6</td>
<td>01</td>
</tr>
<tr>
<td>DB5</td>
<td>02</td>
</tr>
<tr>
<td>DB4</td>
<td>03</td>
</tr>
<tr>
<td>DB3</td>
<td>04</td>
</tr>
<tr>
<td>DB2</td>
<td>05</td>
</tr>
<tr>
<td>DB1</td>
<td>06</td>
</tr>
<tr>
<td>DB0</td>
<td>07</td>
</tr>
<tr>
<td>IRQ</td>
<td>08</td>
</tr>
<tr>
<td>LP In</td>
<td>09</td>
</tr>
<tr>
<td>CS/</td>
<td>10</td>
</tr>
<tr>
<td>R/W</td>
<td>11</td>
</tr>
<tr>
<td>EA</td>
<td>12</td>
</tr>
<tr>
<td>Vdd</td>
<td>13</td>
</tr>
<tr>
<td>Color out</td>
<td>14</td>
</tr>
<tr>
<td>Sync/Lum</td>
<td>15</td>
</tr>
<tr>
<td>AEC</td>
<td>16</td>
</tr>
<tr>
<td>PHG</td>
<td>17</td>
</tr>
<tr>
<td>NCT</td>
<td>18</td>
</tr>
<tr>
<td>CBS</td>
<td>19</td>
</tr>
<tr>
<td>Vss</td>
<td>20</td>
</tr>
<tr>
<td>DB7</td>
<td>21</td>
</tr>
<tr>
<td>DB8</td>
<td>22</td>
</tr>
<tr>
<td>DB9</td>
<td>23</td>
</tr>
<tr>
<td>DB10</td>
<td>24</td>
</tr>
<tr>
<td>DB11</td>
<td>25</td>
</tr>
<tr>
<td>DB12</td>
<td>26</td>
</tr>
<tr>
<td>DB13</td>
<td>27</td>
</tr>
<tr>
<td>DB14</td>
<td>28</td>
</tr>
<tr>
<td>DB15</td>
<td>29</td>
</tr>
<tr>
<td>DB16</td>
<td>30</td>
</tr>
<tr>
<td>DB17</td>
<td>31</td>
</tr>
<tr>
<td>DB18</td>
<td>32</td>
</tr>
<tr>
<td>DB19</td>
<td>33</td>
</tr>
<tr>
<td>DB20</td>
<td>34</td>
</tr>
<tr>
<td>DB21</td>
<td>35</td>
</tr>
<tr>
<td>DB22</td>
<td>36</td>
</tr>
<tr>
<td>DB23</td>
<td>37</td>
</tr>
<tr>
<td>DB24</td>
<td>38</td>
</tr>
<tr>
<td>DB25</td>
<td>39</td>
</tr>
<tr>
<td>DB26</td>
<td>40</td>
</tr>
<tr>
<td>Vcc</td>
<td></td>
</tr>
<tr>
<td>DB07</td>
<td></td>
</tr>
<tr>
<td>DB08</td>
<td></td>
</tr>
<tr>
<td>DB09</td>
<td></td>
</tr>
<tr>
<td>DB10</td>
<td></td>
</tr>
<tr>
<td>DB11</td>
<td></td>
</tr>
<tr>
<td>DB12</td>
<td></td>
</tr>
<tr>
<td>DB13</td>
<td></td>
</tr>
<tr>
<td>DB14</td>
<td></td>
</tr>
<tr>
<td>DB15</td>
<td></td>
</tr>
<tr>
<td>DB16</td>
<td></td>
</tr>
<tr>
<td>DB17</td>
<td></td>
</tr>
<tr>
<td>DB18</td>
<td></td>
</tr>
<tr>
<td>DB19</td>
<td></td>
</tr>
<tr>
<td>DB20</td>
<td></td>
</tr>
<tr>
<td>DB21</td>
<td></td>
</tr>
<tr>
<td>DB22</td>
<td></td>
</tr>
<tr>
<td>DB23</td>
<td></td>
</tr>
<tr>
<td>DB24</td>
<td></td>
</tr>
<tr>
<td>DB25</td>
<td></td>
</tr>
<tr>
<td>DB26</td>
<td></td>
</tr>
</tbody>
</table>

(Multiplexed addresses in parentheses)
CHARACTER DISPLAY MODE

In the character display mode, the 6567 fetches CHARACTER POINTERS from the VIDEO MATRIX area of memory and translates the pointers to character dot addresses in the 2,048 byte CHARACTER BASE area of memory. The video matrix is comprised of 1,000 consecutive locations in memory which each contain an eight bit character pointer. The location of the video matrix within memory is defined by bits VM13-VM10 in register 24 ($18) which are used as the 4 MSB of the video matrix address. The lower order 10 bits are provided by an internal counter (VC9-VC0) which steps through the 1000 character locations. Note that the 6567 provides only 14 address outputs so additional system hardware may be required for complete system memory decodes.

CHARACTER POINTER ADDRESS

A13 A12 A11 A10 A09 A08 A07 A06 A05 A04 A03 A02 A01 A00

CB13 CB12 CB11 VM10 VC9 VC8 VC7 VC6 VC5 VC4 VC3 VC2 VC1 VC0

The eight bit character pointer permits 256 different character definitions to be available simultaneously. Each character is an 8x8 matrix of dots stored in the character base as eight consecutive bytes. The location of the character base is defined by bits CB13-CB11 in register 24 ($18) which are used for the 3 most significant bits (MSB) of the character base address. The 11 lower order addresses are formed by the 8 bit character pointer from the video matrix (D7-D0) which selects a particular character, and a 3 bit raster counter (RC2-RC0) which selects one of the eight character bytes. The resulting characters are formatted as 25 rows of 40 characters each. In addition to the 8 bit character pointer, a 4-bit COLOR NYBBLE is associated with each video matrix location (the video matrix memory is 12 bits wide) which selects one of sixteen colors for each character.

CHARACTER DATA ADDRESS

A13 A12 A11 A10 A09 A08 A07 A06 A05 A04 A03 A02 A01 A00

CB13 CB12 CB11 D7 D6 D5 D4 D3 D2 D1 D0 RC2 RC1 RC0

IC, LSI, Video Controller
EXTENDED COLOR CHARACTER MODE

The extended color mode allows the selection of individual background colors for each character region while maintaining the normal 8 x 8 character resolution. This mode is selected by setting the ECM bit of register 17 (S11) to "1". The character dot data is displayed similar to the standard mode except the 2 MSB of the character pointer are used to select the background color for each character region.

**CHAR POINTER**

<table>
<thead>
<tr>
<th>MS BIT PAIR</th>
<th>BACKGROUND COLOR DISPLAYED</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Background #0 color (register 33 ($21))</td>
</tr>
<tr>
<td>01</td>
<td>Background #1 color (register 34 ($22))</td>
</tr>
<tr>
<td>10</td>
<td>Background #2 color (register 35 ($23))</td>
</tr>
<tr>
<td>11</td>
<td>Background #3 color (register 36 ($24))</td>
</tr>
</tbody>
</table>

Since the two MSB of the character pointers are used for color information, only 64 different character definitions are available. The 8562 forces CB10 and CB9 to "0" regardless of the original pointer values, so that only the first 64 character definitions are accessed. In extended color mode, each character can select one of sixteen individually defined foreground colors and one of the four available background colors. EXTENDED COLOR MODE AND MULTI-COLOR MODE CAN NOT BE ENABLED SIMULTANEOUSLY.

BIT MAP MODE

In bit map mode, the 6567 fetches data from memory to create a one-to-one correspondence between each displayed dot and memory bit. The bit map mode provides a screen resolution of 320H x 200V individually controllable display dots. Bit map mode is selected by setting the BMM bit in register 17 (S11) to a "1". The VIDEO MATRIX is accessed as in character mode, but the video matrix data is interpreted as color data rather than as character pointers. The VIDEO MATRIX COUNTER is also used to create the address to fetch the dot data for display from the 8,000 byte DISPLAY BASE. The display data address is formed as follows:

**BIT MAP DATA ADDRESS**

A13 A12 A11 A10 A09 A08 A07 A06 A05 A04 A03 A02 A01 A00

CB13 VM9 VM8 VM7 VM6 VM5 VM4 VM3 VM2 VM1 VM0 RC2 RC1 RC0

VMx denotes the video matrix counter outputs, RCx denotes the 3 bit raster line counter and CB13 is from register 24 ($18). The raster counter increments once each horizontal video line (raster line) and the video matrix counter increments once every eight lines. This address sequence causes each 8x8 dot block of the video display to reflect eight sequential memory locations.
STANDARD BIT MAP MODE

In the standard bit map mode, color information is derived only from the data stored in the video matrix (the color nybble is disregarded). These 8 bits are divided into two 4-bit nybbles, allowing two colors to be selected independently in each 8x8 dot block. When a bit in the display memory is a "0", the color of the output dot is set by the least significant (lower) nybble (LSN). Similarly, a display memory bit of "1" selects the output color specified by the MSN (upper nybble).

### DISPLAY BIT

<table>
<thead>
<tr>
<th>DISPLAY COLOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 Lower nybble of video matrix pointer</td>
</tr>
<tr>
<td>1 Upper nybble of video matrix pointer</td>
</tr>
</tbody>
</table>

MULTI-COLOR BIT MAP MODE

Multi-colored bit map mode is selected by setting the MCM bit in register 22 ($16) to a "1" in conjunction with the BMM bit. Multi-color mode utilizes the same memory addressing sequence as standard bit map mode, but interprets the display dot data differently.

### DISPLAY BIT PAIR

<table>
<thead>
<tr>
<th>DISPLAY COLOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>00 Background #0 color (register 33 ($21))</td>
</tr>
<tr>
<td>01 Upper nybble of video matrix pointer</td>
</tr>
<tr>
<td>10 Lower nybble of video matrix pointer</td>
</tr>
<tr>
<td>11 Video matrix color nybble</td>
</tr>
</tbody>
</table>

Note that the color nybble (DB11-DB8) IS used for the multi-color bit map mode. As in character multi-color mode, the horizontal dot size is doubled since two bits are required for color selection, resulting in a screen resolution of 160H x 200V. Utilizing multi-color bit map mode, three independently selected colors can be displayed in each 8 x 8 block in addition to the background color.
MOVABLE IMAGE BLOCKS

The movable image block (MIB) is a special type of display image which can be displayed at any screen position without the 8x8 dot block constraints inherent in character and bit map modes. Up to 8 unique MIBs can be displayed simultaneously, each defined by 63 bytes in memory displayed as a 24x21 dot array (shown below). A number of special features make MIBs especially suited for video graphics and game applications.

<table>
<thead>
<tr>
<th>BYTE</th>
<th>BYTE</th>
<th>BYTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>01</td>
<td>02</td>
</tr>
<tr>
<td>03</td>
<td>04</td>
<td>05</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>58</td>
<td>59</td>
</tr>
<tr>
<td>60</td>
<td>61</td>
<td>62</td>
</tr>
</tbody>
</table>

MIB DISPLAY BLOCK

MIB ENABLE

Each MIB can be selectively enabled for display by setting its corresponding enable bit (MnE) to "1" in register 21 ($15). If the MnE bit is "0", the MIB will not be displayed.

MIB POSITION

Each MIB is positioned via its X and Y position register (see register map 1.3) within 512 horizontal and 256 vertical positions. The position of a MIB is specified by the upper-left corner of the array. X locations 23 to 347 ($17-$5157) and Y locations 50 to 249 ($32-$F9) are visible. Since not all available MIB positions are entirely visible on the screen, MIBs may be moved smoothly on and off the display screen.
MIB COLOR

A separate 4-bit register is associated with each MIB to specify the MIB color. There are two MIB color modes.

STANDARD MIB COLOR

In the standard mode, a "0" bit of MIB data allows any background, character or bitmap data to show through (transparent) and a "1" bit is displayed as the MIB color determined by the corresponding MIB Color register.

MULTI-COLOR MIB

Each MIB can be individually specified to be multi-color via MnMC bits in the MIB multi-color register 28 ($1C). When the MnMC bit is "1", the corresponding MIB is displayed in the multi-color mode. In the multi-color mode, the MIB data is interpreted in pairs as in the other multi-color modes.

<table>
<thead>
<tr>
<th>MIB BIT PAIR</th>
<th>COLOR DISPLAYED</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Transparent (background data)</td>
</tr>
<tr>
<td>01</td>
<td>MIB Multi-color #0 (register 37 ($25))</td>
</tr>
<tr>
<td>10</td>
<td>MIB Color (registers 39-46 ($27-$2E))</td>
</tr>
<tr>
<td>11</td>
<td>MIB Multi-color #1 (register 38 ($26))</td>
</tr>
</tbody>
</table>

Since two bits of data are required for each color, the resolution of the MIB is reduced to 12 H x 21 V. Each horizontal dot is expanded to twice the standard size so that the size of the overall MIB does not change. Up to 3 colors can be displayed in each MIB (in addition to transparent), but the two colors specified by the MIB multi-color registers are shared among all 8 MIBs.

MIB MAGNIFICATION

Each MIB can be independently expanded (2X) in both the horizontal and vertical directions. Two registers contain the control bits (MnXE,MnYE) for the magnification control.

<table>
<thead>
<tr>
<th>REGISTER</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>29 ($1D)</td>
<td>Horizontal expand (MnXE) &quot;1&quot;=expand; &quot;0&quot;=normal</td>
</tr>
<tr>
<td>23 ($17)</td>
<td>Vertical expand (MnYE) &quot;1&quot;=expand; &quot;0&quot;=normal</td>
</tr>
</tbody>
</table>

No increase in resolution is realized by expanding the MIBs. The same 24x21 array (or 12x21 if multi-colored) is displayed except but the size of each dot is doubled in the desired direction (up to 4X standard dot dimension if a MIB is both multi-colored and expanded).
MIB PRIORITY

The priority of each MIB may be individually controlled with respect to the other displayed information (from character or bit map modes). The priority of each MIB is set by the corresponding bit (MnDP) of register 27 ($1B).

<table>
<thead>
<tr>
<th>BIT</th>
<th>PRIORITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MIB data displayed instead of any data (in front)</td>
</tr>
<tr>
<td>1</td>
<td>MIB data displayed only instead of Bkgd #0 or multi-color bit pair 01 (behind)</td>
</tr>
</tbody>
</table>

MIB data bits of "0" always permit any other information to be displayed (i.e. MIB transparent sections). The MIBs have a fixed priority with respect to each other, with MIB 0 having the highest priority (in front) and MIB 7 the lowest (behind). When MIB data (except transparent data) of two MIBs are co-incident, the data from the lower number MIB will be displayed.

COLLISION DETECTION

Two types of MIB collision (co-incident) are detected, MIB to MIB collision and MIB to display data collision.

MIB TO MIB COLLISION

A collision between two MIBs occurs when non-transparent output data of two MIBs are co-incident. Co-incident of MIB transparent areas will not generate a collision. When a collision occurs, the MIB bits (MnM) in the MIB-MIB COLLISION register 30 ($1E) will be set to "1" for each colliding MIB. The collision bits remain set until a read of the collision register, when they are automatically cleared. MIBs collisions are detected even if off screen, behind the border.

MIB TO DATA COLLISION

A second register, MIB-DATA COLLISION register 31 ($1F) also contains a bit (MiD) for each MIB which is set to "1" when both the MIB and display data are co-incident. Again, the co-incident of MIB transparent data and/or display background does not generate a collision. For special applications, the display data from the 0-1 multicolor bit pair also does not cause a collision. This feature permits their use as background display data without interfering with true MIB collisions. A MIB-DATA collision can occur off-screen in the horizontal direction if display data has been scrolled to an off-screen position (see scrolling). The MIB-DATA COLLISION register also automatically clears when read.
MIB COLLISION INTERRUPTS

The collision interrupt latches are set whenever the first bit of either register is set to "1". Once any collision bit within a register is set high, subsequent collisions will not set the interrupt latch until that collision register has been cleared to all "0"s by a read.

MIB MEMORY ACCESS

The data for each MIB is stored in 63 consecutive bytes of memory. Each block of MIB data is defined by a MIB pointer, located at the end of the VIDEO MATRIX. Since only 1,000 bytes of the video matrix are used in the normal display modes video matrix locations 1016-1023 (S3F8-S3FF) are used for MIB pointers 0-7 respectively. The eight bit MIB pointer from the video matrix together with the six bits from the MIB byte counter to count through the 63 bytes define the entire 14-bit address field.

MIB DATA ADDRESS

A13 A12 A11 A10 A09 A08 A07 A06 A05 A04 A03 A02 A01 A00

MP7 MP6 MP5 MP4 MP3 MP2 MP1 MP0 MC5 MC4 MC3 MC2 MC1 MC0

The MPx are the MIB pointer bits from the video matrix and MCx are the internally generated MIB counter bits. The MIB pointers are read from the video matrix at the end of every raster line. When the Y position register of a MIB matches the current raster line count, the actual fetches of MIB data begin. Internal counters automatically step through the 63 bytes of MIB data, displaying three bytes on each raster line.

OTHER FEATURES

SCREEN BLANKING

The display screen may be blanked by setting the BLNK bit in register 17 (S11) to a "1". When the screen is blanked, the entire screen displays the exterior color specified by register 32 (S20). When blanking is enabled, only transparent (Phase 1) memory accesses are required, permitting full processor utilization of the system bus. However, MOB data will be accessed if the MOBS are not also disabled.
DISPLAY ROW/COLUMN SELECT

The normal display space consists of 25 rows of 40 character regions per row. For special display purposes, the display window may be reduced to 24 rows or 38 characters. There is no change in the format of the displayed information, except that characters (bits) adjacent to the exterior border area are covered by the border.

<table>
<thead>
<tr>
<th>RSEL</th>
<th>Number of rows</th>
<th>CSEL</th>
<th>Number of columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>24 rows</td>
<td>0</td>
<td>38 columns</td>
</tr>
<tr>
<td>1</td>
<td>25 rows</td>
<td>1</td>
<td>40 columns</td>
</tr>
</tbody>
</table>

The RSEL bit is in register 17 ($11) and the CSEL bit is in register 22 ($16). For standard display the larger display window is normally used, while the smaller display window is normally used in conjunction with scrolling.

SCROLLING

The display data may be scrolled up to one character region in both the horizontal and vertical direction. When used in conjunction with the smaller display window (above), scrolling can be used to create a smooth panning motion of display data while updating the system memory only when a new character row (or column) is required. Scrolling is also used for centering a display within the screen.

<table>
<thead>
<tr>
<th>BITS</th>
<th>REGISTER</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>X2,X1,X0</td>
<td>22 ($16)</td>
<td>Horizontal Position</td>
</tr>
<tr>
<td>Y2,Y1,Y0</td>
<td>17 ($11)</td>
<td>Vertical Position</td>
</tr>
</tbody>
</table>

LIGHT PEN

The light pen input latches the current screen position into a pair of registers (LPX,LPY) on a low-going edge. The X position register 19 ($13) contains the 8 MSB of the X position at the time of transition. Since the X position is defined by a 9 bit counter, resolution to 2 horizontal dots is provided. Similarly, the Y position is latched in its register 20 ($14) with 8 bits provide unique raster resolution within the visible display. The light pen latch may be triggered only once per frame, and subsequent triggers within the same frame will have no effect.
RASTER REGISTER

The raster register is a dual function register. A read of the raster register 18 ($12) returns the lower 3 bits of the current raster position (the MSB-RC8 is located in register 17 ($11)). A write to the raster bits (including RC8) is latched for use in an internal raster compare. When the current raster matches the written value, the raster interrupt latch is set. The raster register should be interrogated to prevent display flicker by delaying display changes to occur outside the visible area. The visible area of the display is from raster 51 to raster 251 ($033-$0FB).

INTERRUPT REGISTER

The interrupt register indicates the status of the four sources of interrupt. An interrupt latch in register 25 ($19) is set to "1" when an interrupt source has generated an interrupt request.

<table>
<thead>
<tr>
<th>LATCH</th>
<th>ENABLE</th>
<th>WHEN SET</th>
</tr>
</thead>
<tbody>
<tr>
<td>IST</td>
<td>ERST</td>
<td>Actual raster count = stored raster count</td>
</tr>
<tr>
<td>IMDC</td>
<td>EMDC</td>
<td>MOB-DATA collision (first bit only)</td>
</tr>
<tr>
<td>IMMC</td>
<td>EMMC</td>
<td>MOB-MOB collision (first bit only)</td>
</tr>
<tr>
<td>ILP</td>
<td>ELP</td>
<td>First negative transition of LP per frame</td>
</tr>
<tr>
<td>IRQ</td>
<td></td>
<td>When IRQ/ output low</td>
</tr>
</tbody>
</table>

To enable an interrupt request to set the IRQ/ output to "0", the corresponding interrupt enable bit in register 26 ($1A) must be set to "1". Once an interrupt latch has been set, the latch may be cleared only by writing a "1" to the associated bit in the interrupt register. This feature allows selective handling of video interrupts without software storing of the active interrupts.

DYNAMIC RAM REFRESH

A dynamic ram refresh controller is built into the 6567 device. Five 8-bit row addresses are refreshed every raster line. This rate guarantees a maximum delay of 2.02 ms between the refresh of any single row address in a 128 refresh scheme. (The maximum delay is 3.66ms in a 256 address refresh scheme). This refresh is totally transparent to the system, since the refresh occurs during Phase 1 of the system clock. The 6567 generates both RAS/ and CAS/ which are normally connected directly to the dynamic rams. RAS/ and CAS/ are generated for every Phase 2 and every video data access (including refresh) so that external clock generation is not required.
## COLOR CODE TABLE

<table>
<thead>
<tr>
<th>DB11</th>
<th>DB10</th>
<th>DB09</th>
<th>DB08</th>
<th>HEX</th>
<th>COLOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$0</td>
<td>Black</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>$1</td>
<td>White</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>$2</td>
<td>Red</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>$3</td>
<td>Cyan</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>$4</td>
<td>Purple</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>$5</td>
<td>Green</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>$6</td>
<td>Blue</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>$7</td>
<td>Yellow</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$8</td>
<td>Orange</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>$9</td>
<td>Brown</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>$A</td>
<td>Light Red</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>$B</td>
<td>Dark Grey</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>$C</td>
<td>Medium Grey</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>$D</td>
<td>Light Green</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>$E</td>
<td>Light Blue</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>$F</td>
<td>Light Grey</td>
</tr>
</tbody>
</table>
### SCREEN POSITION DECODES

#### HORIZONTAL DECODES

<table>
<thead>
<tr>
<th>NAME</th>
<th>SET</th>
<th>CLEAR</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSYNC</td>
<td>416</td>
<td>452</td>
<td>Horizontal sync pulse</td>
</tr>
<tr>
<td>HEQ1</td>
<td>178</td>
<td>196</td>
<td>Horizontal equalization pulse 1</td>
</tr>
<tr>
<td>HEQ2</td>
<td>434</td>
<td>452</td>
<td>Horizontal equalization pulse 2</td>
</tr>
<tr>
<td>HBLANK</td>
<td>396</td>
<td>496</td>
<td>Blanks video during horiz retrace</td>
</tr>
<tr>
<td>BURST</td>
<td>456</td>
<td>492</td>
<td>Gates reference color burst</td>
</tr>
<tr>
<td>BKDE38</td>
<td>35</td>
<td>339</td>
<td>Enables 38 column background</td>
</tr>
<tr>
<td>BKDE40</td>
<td>28</td>
<td>348</td>
<td>Enables 40 column background</td>
</tr>
<tr>
<td>BOL</td>
<td>508</td>
<td>4</td>
<td>Begin line (internal clock)</td>
</tr>
<tr>
<td>EOL</td>
<td>340</td>
<td>346</td>
<td>End line (internal clock)</td>
</tr>
<tr>
<td>VINC</td>
<td>404</td>
<td>412</td>
<td>Increment vertical counter</td>
</tr>
<tr>
<td>CW</td>
<td>12</td>
<td>332</td>
<td>Enable character fetch</td>
</tr>
<tr>
<td>VMBA</td>
<td>496</td>
<td>332</td>
<td>Buss avail for character fetch</td>
</tr>
<tr>
<td>REFW</td>
<td>484</td>
<td>12</td>
<td>Enable dynamic ram refresh</td>
</tr>
<tr>
<td>SPBA</td>
<td>336</td>
<td>376</td>
<td>Buss avail for sprite #0 fetch</td>
</tr>
</tbody>
</table>

#### VERTICAL DECODES

<table>
<thead>
<tr>
<th>NAME</th>
<th>SET</th>
<th>CLEAR</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRESET</td>
<td>261</td>
<td>n/a</td>
<td>Resets vertical count to zero</td>
</tr>
<tr>
<td>VSYNC</td>
<td>17</td>
<td>20</td>
<td>Enables vertical sync</td>
</tr>
<tr>
<td>VEQ</td>
<td>14</td>
<td>23</td>
<td>Enables vertical equalization</td>
</tr>
<tr>
<td>VBLANK</td>
<td>13</td>
<td>24</td>
<td>Blanks video during vert retrace</td>
</tr>
<tr>
<td>VSW24</td>
<td>55</td>
<td>247</td>
<td>Enables 24 row screen window</td>
</tr>
<tr>
<td>VSW24</td>
<td>51</td>
<td>251</td>
<td>Enables 25 row screen window</td>
</tr>
<tr>
<td>EEVMF</td>
<td>48</td>
<td>248</td>
<td>Enables character fetch</td>
</tr>
</tbody>
</table>
## REGISTER MAP

<table>
<thead>
<tr>
<th>ADDRESS</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>00($00)</td>
<td>M0X7</td>
<td>M0X6</td>
<td>M0X5</td>
<td>M0X4</td>
<td>M0X3</td>
<td>M0X2</td>
<td>M0X1</td>
<td>M0X0</td>
<td>MIB 0 X-position</td>
</tr>
<tr>
<td>01($01)</td>
<td>M0Y7</td>
<td>M0Y6</td>
<td>M0Y5</td>
<td>M0Y4</td>
<td>M0Y3</td>
<td>M0Y2</td>
<td>M0Y1</td>
<td>M0Y0</td>
<td>MIB 0 Y-position</td>
</tr>
<tr>
<td>02($02)</td>
<td>M1X7</td>
<td>M1X6</td>
<td>M1X5</td>
<td>M1X4</td>
<td>M1X3</td>
<td>M1X2</td>
<td>M1X1</td>
<td>M1X0</td>
<td>MIB 1 X-position</td>
</tr>
<tr>
<td>03($03)</td>
<td>M1Y7</td>
<td>M1Y6</td>
<td>M1Y5</td>
<td>M1Y4</td>
<td>M1Y3</td>
<td>M1Y2</td>
<td>M1Y1</td>
<td>M1Y0</td>
<td>MIB 1 Y-position</td>
</tr>
<tr>
<td>04($04)</td>
<td>M2X7</td>
<td>M2X6</td>
<td>M2X5</td>
<td>M2X4</td>
<td>M2X3</td>
<td>M2X2</td>
<td>M2X1</td>
<td>M2X0</td>
<td>MIB 2 X-position</td>
</tr>
<tr>
<td>05($05)</td>
<td>M2Y7</td>
<td>M2Y6</td>
<td>M2Y5</td>
<td>M2Y4</td>
<td>M2Y3</td>
<td>M2Y2</td>
<td>M2Y1</td>
<td>M2Y0</td>
<td>MIB 2 Y-position</td>
</tr>
<tr>
<td>06($06)</td>
<td>M3X7</td>
<td>M3X6</td>
<td>M3X5</td>
<td>M3X4</td>
<td>M3X3</td>
<td>M3X2</td>
<td>M3X1</td>
<td>M3X0</td>
<td>MIB 3 X-position</td>
</tr>
<tr>
<td>07($07)</td>
<td>M3Y7</td>
<td>M3Y6</td>
<td>M3Y5</td>
<td>M3Y4</td>
<td>M3Y3</td>
<td>M3Y2</td>
<td>M3Y1</td>
<td>M3Y0</td>
<td>MIB 3 Y-position</td>
</tr>
<tr>
<td>08($08)</td>
<td>M4X7</td>
<td>M4X6</td>
<td>M4X5</td>
<td>M4X4</td>
<td>M4X3</td>
<td>M4X2</td>
<td>M4X1</td>
<td>M4X0</td>
<td>MIB 4 X-position</td>
</tr>
<tr>
<td>09($09)</td>
<td>M4Y7</td>
<td>M4Y6</td>
<td>M4Y5</td>
<td>M4Y4</td>
<td>M4Y3</td>
<td>M4Y2</td>
<td>M4Y1</td>
<td>M4Y0</td>
<td>MIB 4 Y-position</td>
</tr>
<tr>
<td>10($0A)</td>
<td>M5X7</td>
<td>M5X6</td>
<td>M5X5</td>
<td>M5X4</td>
<td>M5X3</td>
<td>M5X2</td>
<td>M5X1</td>
<td>M5X0</td>
<td>MIB 5 X-position</td>
</tr>
<tr>
<td>11($0B)</td>
<td>M5Y7</td>
<td>M5Y6</td>
<td>M5Y5</td>
<td>M5Y4</td>
<td>M5Y3</td>
<td>M5Y2</td>
<td>M5Y1</td>
<td>M5Y0</td>
<td>MIB 5 Y-position</td>
</tr>
<tr>
<td>12($0C)</td>
<td>M6X7</td>
<td>M6X6</td>
<td>M6X5</td>
<td>M6X4</td>
<td>M6X3</td>
<td>M6X2</td>
<td>M6X1</td>
<td>M6X0</td>
<td>MIB 6 X-position</td>
</tr>
<tr>
<td>13($0D)</td>
<td>M6Y7</td>
<td>M6Y6</td>
<td>M6Y5</td>
<td>M6Y4</td>
<td>M6Y3</td>
<td>M6Y2</td>
<td>M6Y1</td>
<td>M6Y0</td>
<td>MIB 6 Y-position</td>
</tr>
<tr>
<td>14($0E)</td>
<td>M7X7</td>
<td>M7X6</td>
<td>M7X5</td>
<td>M7X4</td>
<td>M7X3</td>
<td>M7X2</td>
<td>M7X1</td>
<td>M7X0</td>
<td>MIB 7 X-position</td>
</tr>
<tr>
<td>15($0F)</td>
<td>M7Y7</td>
<td>M7Y6</td>
<td>M7Y5</td>
<td>M7Y4</td>
<td>M7Y3</td>
<td>M7Y2</td>
<td>M7Y1</td>
<td>M7Y0</td>
<td>MIB 7 Y-position</td>
</tr>
<tr>
<td>16($10)</td>
<td>M7X8</td>
<td>M6X8</td>
<td>M5X8</td>
<td>M4X8</td>
<td>M3X8</td>
<td>M2X8</td>
<td>M1X8</td>
<td>M0X8</td>
<td>MIB of X-position</td>
</tr>
<tr>
<td>17($11)</td>
<td>RC8</td>
<td>BC8</td>
<td>BM8</td>
<td>BL8</td>
<td>RSEL</td>
<td>Y2</td>
<td>Y1</td>
<td>Y0</td>
<td>See text</td>
</tr>
<tr>
<td>18($12)</td>
<td>RC7</td>
<td>RC6</td>
<td>RC5</td>
<td>RC4</td>
<td>RC3</td>
<td>RC2</td>
<td>RC1</td>
<td>RC0</td>
<td>Raster register</td>
</tr>
<tr>
<td>19($13)</td>
<td>LPX8</td>
<td>LPX7</td>
<td>LPX6</td>
<td>LPX5</td>
<td>LPX4</td>
<td>LPX3</td>
<td>LPX2</td>
<td>LPX1</td>
<td>Light Pen X</td>
</tr>
<tr>
<td>20($14)</td>
<td>LPY8</td>
<td>LPY7</td>
<td>LPY6</td>
<td>LPY5</td>
<td>LPY4</td>
<td>LPY3</td>
<td>LPY2</td>
<td>LPY1</td>
<td>Light Pen Y</td>
</tr>
<tr>
<td>21($15)</td>
<td>M7E</td>
<td>M6E</td>
<td>M5E</td>
<td>M4E</td>
<td>M3E</td>
<td>M2E</td>
<td>M1E</td>
<td>M0E</td>
<td>MIB Enable</td>
</tr>
<tr>
<td>22($16)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>RES</td>
<td>MCM</td>
<td>CSEL</td>
<td>X2</td>
<td>X1</td>
<td>X0</td>
</tr>
<tr>
<td>23($17)</td>
<td>M7YE</td>
<td>M6YE</td>
<td>M5YE</td>
<td>M4YE</td>
<td>M3YE</td>
<td>M2YE</td>
<td>M1YE</td>
<td>M0YE</td>
<td>MIB Y-expand</td>
</tr>
<tr>
<td>24($18)</td>
<td>VM13</td>
<td>VM12</td>
<td>VM11</td>
<td>VM10</td>
<td>CB13</td>
<td>CB12</td>
<td>CB11</td>
<td>-</td>
<td>Memory Pointers</td>
</tr>
<tr>
<td>25($19)</td>
<td>IRQ/</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>ILP</td>
<td>IMMC</td>
<td>IMBC</td>
<td>IRST</td>
<td>Interrupt Register</td>
</tr>
<tr>
<td>26($1A)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>ELP</td>
<td>EMCM</td>
<td>EMBC</td>
<td>ERST</td>
<td>Enable Interrupt</td>
</tr>
<tr>
<td>27($1B)</td>
<td>M7DP</td>
<td>M6DP</td>
<td>M5DP</td>
<td>M4DP</td>
<td>M3DP</td>
<td>M2DP</td>
<td>M1DP</td>
<td>M0DP</td>
<td>MIB-DATA Priority</td>
</tr>
<tr>
<td>28($1C)</td>
<td>M7MC</td>
<td>M6MC</td>
<td>M5MC</td>
<td>M4MC</td>
<td>M3MC</td>
<td>M2MC</td>
<td>M1MC</td>
<td>M0MC</td>
<td>MIB Multicolor Sel</td>
</tr>
<tr>
<td>29($1D)</td>
<td>M7XE</td>
<td>M6XE</td>
<td>M5XE</td>
<td>M4XE</td>
<td>M3XE</td>
<td>M2XE</td>
<td>M1XE</td>
<td>M0XE</td>
<td>MIB X-expand</td>
</tr>
<tr>
<td>30($1E)</td>
<td>M7MC</td>
<td>M6MC</td>
<td>M5MC</td>
<td>M4MC</td>
<td>M3MC</td>
<td>M2MC</td>
<td>M1MC</td>
<td>M0MC</td>
<td>MIB-MIB Collision</td>
</tr>
<tr>
<td>31($1F)</td>
<td>M7MD</td>
<td>M6MD</td>
<td>M5MD</td>
<td>M4MD</td>
<td>M3MD</td>
<td>M2MD</td>
<td>M1MD</td>
<td>M0MD</td>
<td>MIB-DATA Collision</td>
</tr>
<tr>
<td>32($20)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>BC3</td>
<td>BC2</td>
<td>BC1</td>
<td>BC0</td>
<td>Exterior Color</td>
</tr>
<tr>
<td>33($21)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>BC03</td>
<td>BC02</td>
<td>BC01</td>
<td>BC00</td>
<td>Bkgd #0 Color</td>
</tr>
<tr>
<td>34($22)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>BC13</td>
<td>BC12</td>
<td>BC11</td>
<td>BC10</td>
<td>Bkgd #1 Color</td>
</tr>
<tr>
<td>35($23)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>BC23</td>
<td>BC22</td>
<td>BC21</td>
<td>BC20</td>
<td>Bkgd #2 Color</td>
</tr>
<tr>
<td>36($24)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>BC33</td>
<td>BC32</td>
<td>BC31</td>
<td>BC30</td>
<td>Bkgd #3 Color</td>
</tr>
<tr>
<td>37($25)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>MM03</td>
<td>MM02</td>
<td>MM01</td>
<td>MM00</td>
<td>MIB Multicolor #0</td>
</tr>
<tr>
<td>38($26)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>MM13</td>
<td>MM12</td>
<td>MM11</td>
<td>MM10</td>
<td>MIB Multicolor #1</td>
</tr>
<tr>
<td>39($27)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>MM03</td>
<td>MM02</td>
<td>MM01</td>
<td>MM00</td>
<td>MIB 0 Color</td>
</tr>
<tr>
<td>40($28)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M1C3</td>
<td>M1C2</td>
<td>M1C1</td>
<td>M1C0</td>
<td>MIB 1 Color</td>
</tr>
<tr>
<td>41($29)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M2C3</td>
<td>M2C2</td>
<td>M2C1</td>
<td>M2C0</td>
<td>MIB 2 Color</td>
</tr>
<tr>
<td>42($2A)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M3C3</td>
<td>M3C2</td>
<td>M3C1</td>
<td>M3C0</td>
<td>MIB 3 Color</td>
</tr>
<tr>
<td>43($2B)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M4C3</td>
<td>M4C2</td>
<td>M4C1</td>
<td>M4C0</td>
<td>MIB 4 Color</td>
</tr>
<tr>
<td>44($2C)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M5C3</td>
<td>M5C2</td>
<td>M5C1</td>
<td>M5C0</td>
<td>MIB 5 Color</td>
</tr>
<tr>
<td>45($2D)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M6C3</td>
<td>M6C2</td>
<td>M6C1</td>
<td>M6C0</td>
<td>MIB 6 Color</td>
</tr>
<tr>
<td>46($2E)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M7C3</td>
<td>M7C2</td>
<td>M7C1</td>
<td>M7C0</td>
<td>MIB 7 Color</td>
</tr>
</tbody>
</table>

**NOTE:** All dashes read as "1" regardless of written value.
SYSTEM INTERFACE

The 6567 video controller device interacts with the system data bus in a unique way. A 65XX system requires the system busses only during the Phase 2 (clock high) portion of the cycle. The 6567 device takes advantage of this characteristic by normally accessing system memory only during the Phase 1 (clock low) portion of the clock cycle. Therefore, operations such as character data fetches and memory refresh are totally transparent to the processor and do not reduce the processor through-put. The video chip provides the interface control signals required to maintain this bus sharing.

<table>
<thead>
<tr>
<th>AEC</th>
<th>PH0</th>
<th>CS/</th>
<th>R/W</th>
<th>ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>PHASE 1 - 8562 MEMORY FETCH</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>PHASE 2 - DMA (PROCESSOR OFF)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>WRITE TO 8562 REGISTER</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>READ FROM 8562 REGISTER</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>X</td>
<td>PROCESSOR MEMORY CYCLE</td>
</tr>
</tbody>
</table>

BUS MULTIPLEXING

The video circuit provides the AEC signal (address enable control) which is used to disable the processor address bus drivers, thereby allowing the video device to access the address bus. AEC is active low which permits direct connection to the AEC input of the 65XX family. The AEC signal is activated during every Phase 1 so that processor operation is not normally affected. Because of this bus "sharing", all memory accesses must be completed in 1/2 cycle. Since the video chip provides a 1MHz clock (which should be used as system Phase 2), a memory cycle is 500ns including address setup, data access and data setup to the reading device.

DMA OPERATION

Certain operations of the 6567 require data at a faster rate than available by reading only during the Phase 1 time; specifically, the access of character pointers from the video matrix and the fetch of MIB data. Therefore, the processor must be disabled and the data accessed during the Phase 2 clock. This is accomplished via the BA (bus available) signal. The BA line is normally high but is brought low during Phase 1 to indicate that the video chip will require a Phase 2 data access. Three Phase 2 times are allowed after BA low for the processor to complete any current memory accesses. On the fourth Phase 2 after BA low, the AEC signal will remain low during Phase 2 as the video chip fetches data. The BA line is normally connected to the RDY input of a 65XX processor. The character pointer fetches occur every eighth raster line during the display window and require 40 consecutive Phase 2 accesses to fetch the video matrix pointers.
The MIB pointers are fetched every other Phase 1 at the end of each raster line. As needed, additional cycles are used for MIB data fetches, with all necessary bus control provided by the 8562 device.

<table>
<thead>
<tr>
<th>PHASE</th>
<th>DATA</th>
<th>CONDITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MIB Pointer</td>
<td>Every raster</td>
</tr>
<tr>
<td>2</td>
<td>MIB Byte 1</td>
<td>Each raster while MIB is displayed</td>
</tr>
<tr>
<td>1</td>
<td>MIB Byte 2</td>
<td>Each raster while MIB is displayed</td>
</tr>
<tr>
<td>2</td>
<td>MIB Byte 3</td>
<td>Each raster while MIB is displayed</td>
</tr>
</tbody>
</table>

**INTERFACE SIGNAL DESCRIPTION**

2.8.3.1 DATA BUS (DB7-DB0)

The eight data bus pins are the bi-directional data port, controlled by CS/, RW, and Phase 0. The data bus can only be accessed while AEC and Phase 0 are high and CS/ is low.

**CHIP SELECT (CS/1)**

The chip select pin, CS/, is brought low to enable access to the circuit registers in conjunction with the address and RW pins. CS/ low is recognized only while AEC and Phase 0 are high.

**READ/WRITE (R/W)**

The read/write input, R/W, is used to determine the direction of data transfer on the data bus, in conjunction with CS/. When R/W is high ("1") data is transferred from the selected register to the data bus output. When R/W is low ("0") data presented on the data bus pins is loaded into the selected register.

**ADDRESS BUS (A10-A0)**

Address pins A06-A00 are multiplexed to allow direct connection to dynamic RAMs. During the row address time, A06-A00 are driven on pins A06-A00, while A13-A08 are driven on pins A05-A00 and pin A06 is "1" during the column time. A10-A07 is stable for the entire memory cycle. The lower six address pins, A05-A00, are also bi-directional. During a processor read or write of the video device, address pins A05-A00 are inputs which latch on the low edge of RAS/. The data on these address inputs selects the register for read or write as defined in the register map.

**CLOCK OUT**

The clock output, Phase 0, is the 1MHz clock used as the 65XX processor Phase 0 in. All system bus activity is referenced to this clock. The clock frequency is generated by dividing the 8MHz video input clock by eight.
RAS/CAS

The RAS/ and CAS/ signals provide the timing signals necessary for dynamic RAMs. These signals are provided twice each clock out cycle to support a memory cycle during both phase 1 and phase 2.

INTERRUPTS (IRQ/)

The interrupt output, IRQ/, is brought low when an enabled source of interrupt occurs with in the device. The IRQ/ output is open drain, requiring an external pull-up resistor.

VIDEO INTERFACE

The video output from the 6567 consists of two signals which must be externally mixed together to create NTSC compatible composite video. After appropriate mixing, the resulting signal can directly drive a video monitor or RF modulator for use with a standard television.

VIDEO CLOCK IN

This clock input is the fundamental dot shift rate clock. In addition, all other video sync and system interface signals are derived by dividing this clock. Normally, the desired frequency is 8 Mhz. The primary constraints are the system timing and the requirement that one horizontal raster line is 520 video clock cycles.

COLOR CLOCK IN

The color clock is used to derive the chrominance signal. NTSC video systems require this input to be 14.31818 Mhz. This base frequency is divided by 4 internally and then appropriately phase shifted to create the 3.5795 Mhz video color signal for NTSC applications.

SYNC/LUM OUTPUT

SYNC/LUM output from the 6567 contains all the video data, including horizontal and vertical syncs, as well as the luminance information of the video display is an open drain output requiring an external pullup of 500 ohms.

COLOR OUTPUT

The COLOR output contains all the chrominance information, including the color reference burst and the color of all display data. The COLOR output is open source and should be terminated with 1,000 ohms to ground.
## ELECTRICAL REQUIREMENTS

### ABSOLUTE MAXIMUM RATINGS

Stresses above those listed may cause permanent damage to the circuit. This is a stress rating only. Functional operation of the device at these or any conditions other than those indicated in the operational sections of this specification is not implied.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ambient temperature under bias</td>
<td>-25</td>
<td>+125</td>
<td>Deg. C.</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>-65</td>
<td>+150</td>
<td>Deg. C.</td>
</tr>
<tr>
<td>Applied Supply voltage Vcc</td>
<td>-0.5</td>
<td>+7.0</td>
<td>Volts</td>
</tr>
<tr>
<td>Applied Output voltage</td>
<td>-0.5</td>
<td>+5.5</td>
<td>Volts</td>
</tr>
<tr>
<td>Applied Input voltage</td>
<td>-0.5</td>
<td>+7.0</td>
<td>Volts</td>
</tr>
<tr>
<td>Applied Supply Volt. VDD</td>
<td>-0.5</td>
<td>+15.0</td>
<td>Volts</td>
</tr>
</tbody>
</table>

### OPERATING CONDITIONS

All electrical characteristics (A.C. and D.C.) are specified over the entire range of the operating conditions unless specifically noted.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage (Vcc)</td>
<td>4.75</td>
<td>5.25</td>
<td>Volts</td>
</tr>
<tr>
<td>Free Air Temperature</td>
<td>0</td>
<td>50</td>
<td>Deg. C.</td>
</tr>
<tr>
<td>Dot Clock cycle</td>
<td>122</td>
<td>123</td>
<td>ns</td>
</tr>
<tr>
<td>Dotclk lo pulse in</td>
<td>60</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>Dotclk hi pulse in</td>
<td>45</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>Color Clock frequency</td>
<td>14.317</td>
<td>14.319</td>
<td>MHz</td>
</tr>
<tr>
<td>Colorclk lo pulse in</td>
<td>25</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>Colorclk hi pulse in</td>
<td>25</td>
<td>-</td>
<td>ns</td>
</tr>
</tbody>
</table>

### D.C. CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Min</th>
<th>Max</th>
<th>Units</th>
<th>Test Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Input high level</td>
<td>Vih</td>
<td>2.0</td>
<td>Vcc</td>
<td>Volts</td>
<td></td>
</tr>
<tr>
<td>3.3.2 Input low level</td>
<td>Vil</td>
<td>-0.5</td>
<td>0.8</td>
<td>Volts</td>
<td></td>
</tr>
<tr>
<td>3.3.3 Output High level</td>
<td>Voh</td>
<td>2.4</td>
<td>-</td>
<td>Volts</td>
<td>Ioh = -200μA</td>
</tr>
<tr>
<td>3.3.4 Output Low level</td>
<td>Vol</td>
<td>-</td>
<td>0.4</td>
<td>Volts</td>
<td>Iol = 3.2mA</td>
</tr>
<tr>
<td>3.3.6 I/O Leakage</td>
<td>Ilkg</td>
<td>-10</td>
<td>10</td>
<td>μAmps</td>
<td>0.4V&lt;Vout&lt;2.4V (addr buss off)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(data buss off) Outputs open</td>
</tr>
<tr>
<td>3.3.7 Supply Current</td>
<td>Icc</td>
<td>-</td>
<td>200</td>
<td>mAmps</td>
<td></td>
</tr>
<tr>
<td>3.3.8 Pin capacitance</td>
<td>Cpin</td>
<td>-</td>
<td>50</td>
<td>μF</td>
<td></td>
</tr>
</tbody>
</table>
A.C. CHARACTERISTICS

A.C. characteristics are specified with input waveforms between 0.4V input low level and 2.4V input high level and Dot Clock in of 8.182 MHz. Outputs are loaded at the rated D.C. currents and voltage with 60pF total capacitive load (including fixtureing). All time measurements of active signals are referenced to 1.5V on inputs and outputs. Time measurements of high impedance signals are referenced to Vin and Voh levels. See Figure 1 Page 21 for timing relationships.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Min</th>
<th>Max</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PW C1kout hi</td>
<td>Tph</td>
<td>465</td>
<td>500</td>
<td>ns</td>
</tr>
<tr>
<td>PW C1kout lo</td>
<td>Tpl</td>
<td>475</td>
<td>510</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-RAS1o</td>
<td>Tthl</td>
<td>155</td>
<td>190</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-RASHi</td>
<td>Trhl</td>
<td>23</td>
<td>50</td>
<td>ns</td>
</tr>
<tr>
<td>PW RAS hi</td>
<td>Trp</td>
<td>135</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>Dly RAS1o-CAS1o</td>
<td>TRcd</td>
<td>25</td>
<td>65</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-CAS1o</td>
<td>Tch1</td>
<td>-</td>
<td>220</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-CASHi</td>
<td>Tch1</td>
<td>15</td>
<td>35</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-AECHi/lo</td>
<td>Taec</td>
<td>10</td>
<td>30</td>
<td>ns</td>
</tr>
<tr>
<td>Dly C1k-BAHi/lo</td>
<td>Tba</td>
<td>-</td>
<td>300</td>
<td>ns</td>
</tr>
<tr>
<td>Setup Addout-RAS1o</td>
<td>Tarsout</td>
<td>35</td>
<td>-</td>
<td>ns 6567 mem read</td>
</tr>
<tr>
<td>Hold Addout-RAS1o</td>
<td>Trnahout</td>
<td>30</td>
<td>45</td>
<td>ns 6567 mem read</td>
</tr>
<tr>
<td>Setup Addout-CAS1o</td>
<td>Tasc</td>
<td>0</td>
<td>-</td>
<td>ns 6567 mem read</td>
</tr>
<tr>
<td>Hold Addout-CASHi</td>
<td>Tcah</td>
<td>20</td>
<td>50</td>
<td>ns 6567 mem read</td>
</tr>
<tr>
<td>Setup Addin-RAS1o</td>
<td>Tasrin</td>
<td>25</td>
<td>-</td>
<td>ns uP read/write</td>
</tr>
<tr>
<td>Hold Addin-RAS1o</td>
<td>Trahin</td>
<td>0</td>
<td>-</td>
<td>ns uP read/write</td>
</tr>
<tr>
<td>Dly Data-CAS1o</td>
<td>Tcacout</td>
<td>-</td>
<td>220</td>
<td>ns uP read</td>
</tr>
<tr>
<td>Dly C1khi-dataoff</td>
<td>Toff</td>
<td>80</td>
<td>135</td>
<td>ns uP read</td>
</tr>
<tr>
<td>Setup Datain-C1k</td>
<td>Tds</td>
<td>50</td>
<td>-</td>
<td>ns mem/uP/color data</td>
</tr>
<tr>
<td>Hold Datain-C1k</td>
<td>Tdh</td>
<td>40</td>
<td>-</td>
<td>ns mem/uP/color data</td>
</tr>
</tbody>
</table>
FIGURE 1

6567 TIMING DIAGRAM

COMMODORE SEMICONDUCTOR

IC, LSI, Video Controller