Final Project
**FINAL PROJECT**

**THE PROBLEM**

A two-dimensional video image is transferred between processing modules by passing one pixel at a time qualified by horizontal and vertical blank pulses.

---

**Example**

*shown for 4x4 image*

<table>
<thead>
<tr>
<th></th>
<th>X-</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>13</td>
</tr>
</tbody>
</table>

HB

VB

---

**Clk**

---

**Pixel**

---

**HB**

---

**VB**

---

© COPYRIGHT 1996 DAVID MICHAEL CLARK. ALL RIGHTS RESERVED.
The task is to write a VHDL module that accepts the incoming video stream, stores it in a RAM, calculates a histogram from the input data, and locates the 50th percentile of the histogram. A functional block diagram and processing timeline are as follows:

---

**Digital System Design with VHDL Final Project Notes**

**ProjectNotes_97.doc**

**Slide 25**

**REVISION 001**

February 12, 1998

DAVID M. CLARK

© COPYRIGHT 1996 DAVID MICHAEL CLARK. ALL RIGHTS RESERVED.
Implement the following functions

• Input Decoder / Video RAM Controller
  - Accepts video stream, horizontal and vertical blanks, and generates RAM address, data, write enable, and chip select (data strobe) signals to store the incoming video into a RAM

• Histogram Calculation / Histogram RAM controller
  - At the beginning of the video stream, clears the histogram accumulation RAM
  - After the video stream has been stored in the video RAM, generates RAM address, data, write enable, and chip select (data strobe) signals to read the video into a RAM

• Readout Controller / Percentile Determination
  - Sequences out the histogram data, calculates the 50-percentile point, and presents the percentile point to output
  - 50-percentile point is defined as the point (value) in the histogram at which 50% or more of the pixel intensities fall at or below it
FINAL PROJECT

ILLUSTRATION

VIDEO PIXEL DATA STREAM = 0 0 0 1 2 4 6 5 7 4 2 0 0 0

HISTOGRAM RAM BIN COUNTS

INTENSITY

TIME (IN PIXEL INTERVALS)

PIXEL COUNTS
(Histogram Data)

50-percentile

PIXEL INTENSITY
(Histogram Address)
FINAL PROJECT

INTERFACE SPECIFICATIONS

- All external interfaces shall be std_logic or std_logic_vector
- The Module shall contain an external active-low reset (resetLowIn), and two external clocks, 10 MHz and 20MHz (clkIn and clk2xIn respectively)
- The Module shall contain a CPU programming interface consisting of:
  - a 24-bit address bus (cpuAddrIn)
  - a bidirectional 16-bit data bus (cpuDataBi)
  - a 1-bit active-low Write Enable (cpuWriteEnLowIn)
  - a 1-bit active-low Data Strobe (cpuDataStrobeLowIn)
- The Module shall contain a Video RAM interface consisting of:
  - an 8-bit address bus (vidAddrOut)
  - an 8-bit bidirectional data bus (vidDataBi)
  - a 1-bit active-low Write Enable (vidWriteEnLowOut)
  - a 1-bit active-low Chip Select (vidChipSelLowOut)
The Module shall contain a Histogram RAM interface consisting of:
- an 8-bit address bus (\textit{histAddrOut})
- an 8-bit bidirectional data bus (\textit{histDataBi})
- a 1-bit active-low Write Enable (\textit{histWriteEnLowOut})
- a 1-bit active-low Chip Select (\textit{histChipSelLowOut})

The Module shall contain an 8-bit data output (\textit{statDataOut}) and a data valid gate (\textit{statDataValidOut}) which shall be asserted during the output sequencing.

The Module shall accept 8-bit video data (\textit{pixelDataIn}), an active-high horizontal blank (\textit{horizBlankIn}), and an active-high vertical blank (\textit{vertBlankIn}).

\textbf{596B Students only:}
- The Module shall contain an active-low End-Of-Frame interrupt to the CPU (\textit{eofIntLowOut})
FINAL PROJECT

FUNCTIONAL SPECIFICATIONS: ALL STUDENTS

- Input image size shall be a maximum of 16 x 16 pixels
- The Module shall contain a CPU programmable threshold value.
- Every incoming pixel whose value exceeds the threshold value shall be replaced by the threshold value in the histogram determination.
- The Histogram RAM shall be cleared at the beginning of each video frame, prior to accumulation of the histogram.
The Module shall contain a CPU programmable Pause flag

Module function shall be suspended during each video frame in which the Pause flag is set at the beginning of the frame

Assertion of the Pause flag shall not interrupt processing of the current video frame

The Video RAM and the Histogram shall be accessible for Read and Write to the CPU during any frame in which the Pause flag is set

The Module shall supply an End-Of-Frame interrupt to the CPU

The interrupt shall be asserted when the last statistics value has been read out and shall remain asserted until an Interrupt Acknowledge is received. An Interrupt Acknowledge shall consist of a Write of 0 to the Interrupt Acknowledge address, followed by a Write of 1 to the same address
FINAL PROJECT
SAMPLE TIMING
## FINAL PROJECT

### CPU MEMORY MAP

<table>
<thead>
<tr>
<th>ADDRESS</th>
<th>MEMORY FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>00,0000 - 7F,FFFFh</td>
<td>reserved for internal CPU use</td>
</tr>
<tr>
<td>80,0000h</td>
<td>VIDEO THRESHOLD REGISTER</td>
</tr>
<tr>
<td>80,0001h</td>
<td>PAUSE FLAG</td>
</tr>
<tr>
<td>80,0002h</td>
<td>END-OF-FRAME INTERRUPT ACKNOWLEDGE REGISTER</td>
</tr>
<tr>
<td>80,0003 - 9F,FFFFh</td>
<td>reserved</td>
</tr>
<tr>
<td>A0,0000 - A0,00FFh</td>
<td>VIDEO RAM</td>
</tr>
<tr>
<td>A0,0100 - BF,FFFFh</td>
<td>reserved</td>
</tr>
<tr>
<td>C0,0000 - C0,00FFh</td>
<td>HISTOGRAM RAM</td>
</tr>
<tr>
<td>C0,0100 - FF,FFFFh</td>
<td>reserved</td>
</tr>
</tbody>
</table>