XAPP284 (v1.1) October 15, 2001

Matrix Math, Graphics, and Video

Author: Latha Pillai

Summary

Many pipelined functions in the computer graphics and video fields are expressed in matrix mathematics. This Matrix Multiplier application note describes a unique way to implement a 3 x 3-matrix multiplier using a Virtex™-II device. By running High-performance Virtex-II multipliers at multiples of the system clock rate in slower applications, silicon resources can be leveraged. The resulting efficient use of silicon resources relies on the Xilinx Digital Clock Manager (DCM) to generate a low-skew clock using multipliers running at a higher clock rate. The low-skew clock is a 3x multiple of the system clock.

The specific example in this application note is a color space conversion, viewed as a subset of matrix multiplication. However, the techniques used can be extended to other matrix math functions as well.

This application note describes the Logic Devices LF2272 Colorspace Converter/Corrector. The reference design implements the functions of an LF2272 in a Virtex-II FPGA for SDTV application speeds.

Matrix Math in 3D Graphics and Video

Many applications in video and 3D graphics can be expressed as a matrix multiply or a subset of a matrix multiply. Some of these applications include: image filtering/manipulation, video effects generation, video standards conversion, encoding/decoding, three-dimensional image manipulation, medical image processing, edge detection for object recognition, and FIR filtering for communications systems. One easily understood example (not covered here) is color-space conversion. Techniques for efficiently using high-performance math elements in FPGAs with a "time sharing" operation are discussed in this application note.

Triple Dot Product or Matrix Times a Vector: A Subset of Matrix Multiplication

Equation 1 shows a 3 x 3 matrix multiply consisting of 27 multiplies and 18 adds.

$$\begin{bmatrix} A11 & A12 & A13 \\ A21 & A22 & A23 \\ A31 & A32 & A33 \end{bmatrix} \bullet \begin{matrix} B11 & B12 & B13 \\ B21 & B22 & B23 \\ B31 & B32 & B33 \end{matrix} \Bigg] = \begin{matrix} C11 & C12 & C13 \\ C21 & C22 & C23 \\ C31 & C32 & C33 \end{matrix}$$

Or

$$(A11 \times B11) + (A12 \times B21) + (A13 \times B31) = C11$$
$$(A11 \times B12) + (A12 \times B22) + (A13 \times B32) = C12$$
$$(A11 \times B13) + (A12 \times B23) + (A13 \times B33) = C13$$

$$(A21 \times B11) + (A22 \times B21) + (A23 \times B31) = C21$$
$$(A21 \times B12) + (A22 \times B22) + (A23 \times B32) = C22$$
$$(A21 \times B13) + (A22 \times B23) + (A23 \times B33) = C23$$

$$(A31 \times B11) + (A32 \times B21) + (A33 \times B31) = C31$$
$$(A31 \times B12) + (A32 \times B22) + (A33 \times B32) = C32$$
$$(A31 \times B13) + (A32 \times B23) + (A33 \times B33) = C33$$

A subset of 3 x 3-matrix multiplication (Figure 1) is the matrix times a vector, with nine multiplies and six adds gives Equation 2:

$$\begin{pmatrix} A11 & A12 & A13 \\ A21 & A22 & A23 \\ A31 & A32 & A33 \end{pmatrix} \bullet \begin{matrix} B11 \\ B21 \\ B31 \end{matrix} = \begin{matrix} C11 \\ C21 \\ C31 \end{matrix}$$

Or

$$\begin{aligned} (A11 \times B11) + (A12 \times B21) + (A13 \times B31) &= C11 \\ (A21 \times B11) + (A22 \times B21) + (A23 \times B31) &= C21 \\ (A31 \times B11) + (A32 \times B21) + (A33 \times B31) &= C31 \end{aligned}$$

This is sometimes referred to as a vector rotation where a three-dimensional vector is rotated from one 3D coordinate space to another using a 3 x 3 transformation matrix. The math above can also be thought of as three vectors multiplied by a fourth or a triple-dot product. Finally, the video process of color conversion from one space to another can be described as a 3 x 3 matrix of constants multiplied by vector, resulting in a vector. By merely reloading the matrix a new conversion is specified.

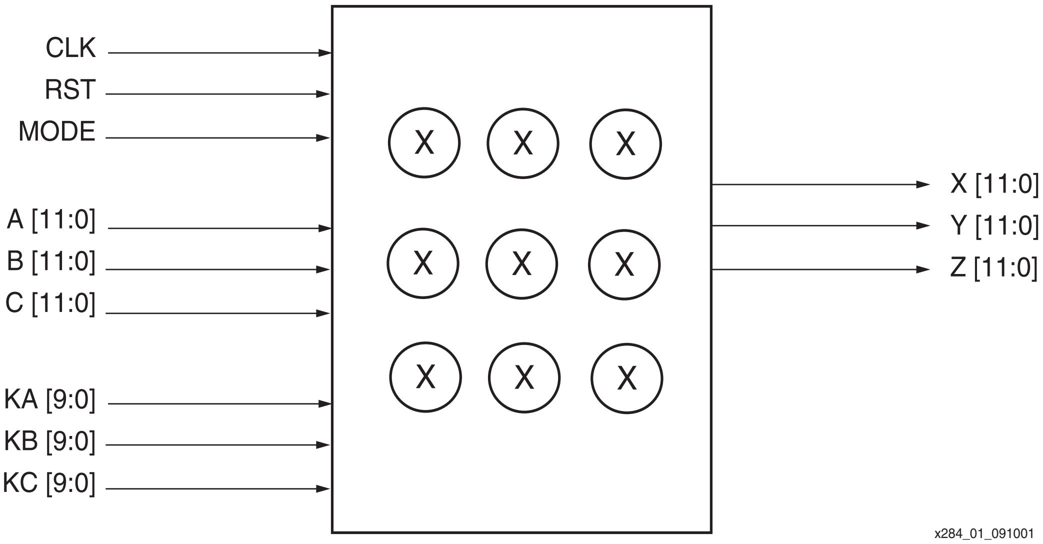

Figure 1: **Triple Dot Product**

Handling both HDTV (74.25 MHz) and SDTV (13.5 MHz) Rates:

A video method for working with both High Definition TV (HDTV) and Standard Definition TV (SDTV) rates is desirable. Trying to work with two rates vastly different rates (a factor of 5.5), requires designing the math differently for each. Because video content is a continuous stream, it can be easily pipelined. Video enters the FPGA a pixel at a time, in the same order, for every field. Hardware can be designed efficiently to satisfy the Standard Definition rates of 13.5 MHz and then to duplicate the hardware for HDTV. This required slightly adjusting the system clock. Because the HDTV rate is 5.5 times SDTV rate and the hardware is duplicated five times, (5.5/5 = 1.1), the system clock frequency should be increased by a factor of 1.1. A very efficient SDTV design is required because the HDTV version will be at least five times larger.

Xilinx Core Generator

The constant coefficient multiplier is a subset of the 3 x 3 matrix multiplier. The Xilinx Core Generator has drop-in modules for constant-coefficient multipliers to be used in the XC4000E, EX, XL, XV and SPARTAN families. These cores have guaranteed high performance and density. Both pipelined and non-pipelined version for 2's complement signed and unsigned numbers are available.

After installing the tool, download the latest libraries off of the Xilinx web site and look through the GUIs folder arrangement for possible solutions. FIR filters are found under the DSP folder. The online data sheets provide detailed implementation descriptions as well as expected size, shape, and speed in targeted parts. An RLOCed version of most cores can be output to guide the Xilinx map, place, and route software.

Triple Dot Product Using Virtex-II Device

The block diagram for the implementation of a subset of a 3 x 3 matrix multiplier, a 3 x 3 matrix multiplied by a vector (Equation 2), using a Virtex-II FPGA is shown in Figure 2. The 12-bit wide input signals are A, B, and C. The 10-bit coefficient inputs are KA, KB, and KC. The coefficient sets are stored in registers. The mode pin (MODE) decides the sets of registers being updated with the input coefficients. The outputs are determined by the following equations:

$$X = (A \times KA1) + (B \times KB1) + (C \times KC1)$$
$$Y = (A \times KA2) + (B \times KB2) + (C \times KC2)$$
$$Z = (A \times KA3) + (B \times KB3) + (C \times KC3)$$

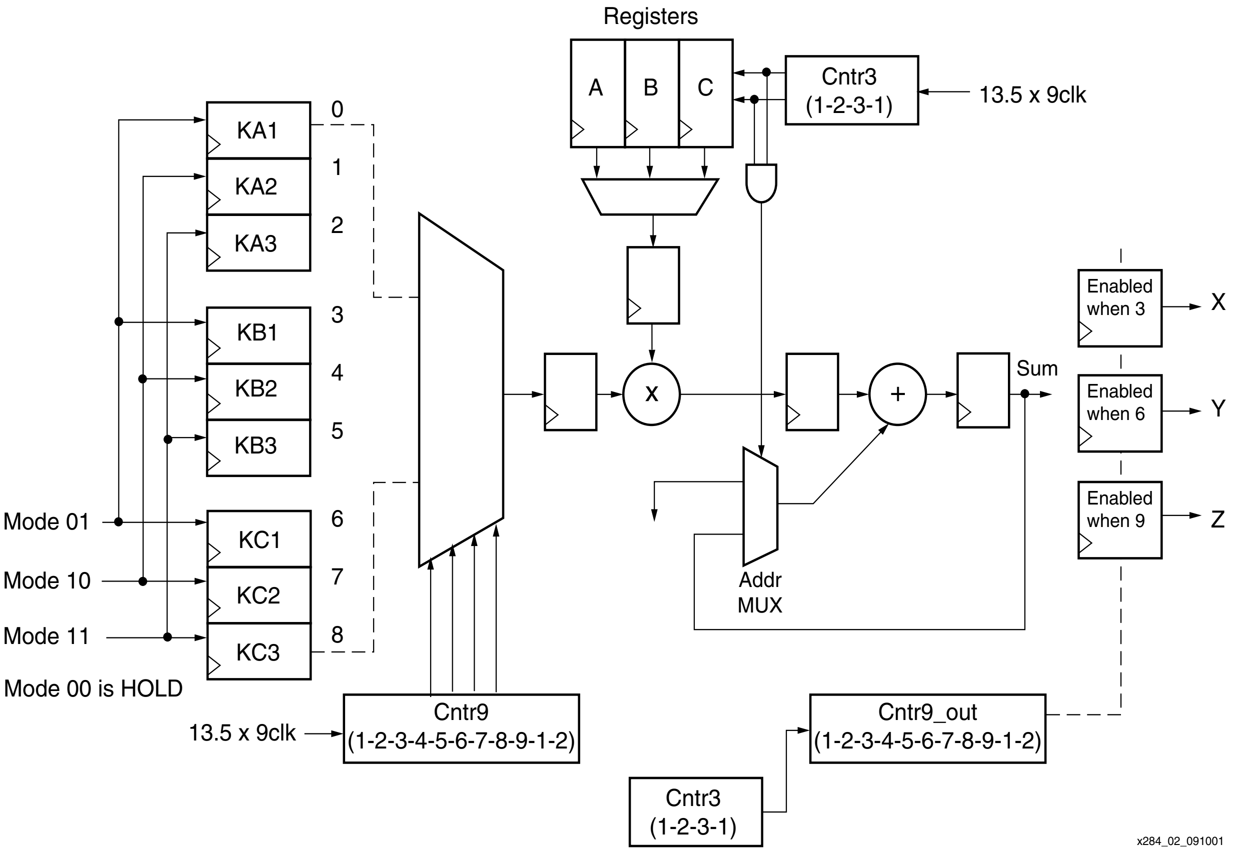

Figure 2: **3x3 Matrix Multiplier Block Diagram Using a Virtex-II Device**

The high-speed capability of a Virtex-II device allows the user to "multi pump" the multipliers. The term "multi pump" refers to feeding nine sets of inputs resulting in nine sets of results at nine times the clock rate of the system. This brings down the multiplier count to one from nine for the triple dot product. (See Figure 1.) The multiplier on the chip is run at nine times the speed. This would mean that for SDTV, the multiplier is run at 121.5 MHz (13.5 x 9). Each of the input data signals is fed into one multiplier at 9x speed.

Each of the input data signals is fed into one multiplier at nine times the speed. Each of the three coefficient sets, with a total of nine coefficient values are fed to the multiplier at 9x speed.

The outputs of the multiplier are adder at 9x speed. At every third clock, the adder output is stored in the X, Y, and Z output registers, respectively.

All the input signals and output signals are assumed to be in two's complement format. The 12-bit data input and the 10-bit coefficient inputs can give an adder output of up to 23-bits wide. Only the 12 most significant bits (MSB) bits of the adder output are used for X, Y, and Z outputs.

Virtex-II Implementation Results

The Verilog model of the above implementation with a target frequency of 150 MHz was synthesized using Synplicity. The resulting EDIF file was then implemented on the Virtex-II device using Design Manager. A timing constraint of 200 MHz was given and a "high-effort level" was chosen for the place and route tool.

Table 1: **Design Summary**

Number of slice flip flops	146 out of 6144	2%
Number used as LUTs	113	
Number of bonded IOBs	106 out of 172	61%
Number of MULT18X18s	1 out of 32	3%

Timing Summary

Minimum period	6.490 ns (maximum frequency: 154.083 MHz)
Minimum input arrival time before clock	3.656 ns
Minimum output required time after clock	4.346 ns

Reference Design

The VHDL and Verilog reference designs for this application note are available on the Xilinx web site in a .zip file:

ftp://ftp.xilinx.com/pub/applications/xapp/xapp284.zip

Conclusion

Matrix multiplication, which is widely used in the computer graphics and video area, can be implemented efficiently using a Virtex-II FPGA. The implementation uses only one embedded multiplier to make a 3 x 3-matrix multiplied by a vector. The high performance capability of the Virtex-II device enables the user to run the multiplier at 9x speed and time-share the multiplier, thus reducing the multiplier count to one. The design after place and route on a Virtex-II -5 FPGA was seen to run at 154 MHz, satisfying the SDTV rate of 13.5 x 9 = 121.5 MHz. When the hardware is duplicated five times for HDTV implementation, the required speed is 121.5 x 1.1 = 133.65 MHz (well within the achieved speed of 154 MHz.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
07/11/01	1.0	Initial Xilinx release.
10/15/01	1.1	Updated entire document. Corrected equation errors. Corrected Figure 1 and Figure 2. Corrected Table 1.