Reference DESIGN

Constant (k) Coefficient Multiplier Generator for Virtex

March 21, 1999 Application Note

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
FAX: +1 408-559-7114
Email: coregen@xilinx.com
URL: http://www.xilinx.com/ipcenter

Features

- DOS-based utility to generate fixed size 8x8, 12x12,16x16 and 20x20 Constant (K) Coefficient Multiplier macros
- Multiplies a variable X times a constant K
- Supports 2's complement Signed or Unsigned variable input
- Full precision output
- Supports pipelined or fully combinatorial construction
- Drop-in modules for the Virtex family
- High performance and density guaranteed through Relational Placed Macro (RPM) mapping and placement technology
- Interim solution until a more complete CORE Generator™ solution becomes available

Functional Description

The KCM_VGEN Virtex constant coefficient multiplier generator creates multipliers that multiply an N-bit wide variable by an N-bit fixed coefficient, and produces a 2N-bit result. The coefficient multiplication tables are stored in distributed ROM-based look-up tables (LUTs), taking advantage of the FPGA look-up table architecture. It is an efficient, high speed, parallel implementa-tion.

The variable input can be specified to be a 2's complement signed value, or as an unsigned value. The constant input is always signed. Note that the module will automatically handle a signed number times an unsigned number, or two signed numbers.

Pinout and Instantiation Template

Port names are described in Table 1. The instantiation templates for the Combinatorial and Pipelined multipliers are shown in Tables 2 and 3, respectively.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
X[N-1:0]	Input	Parallel data in – Operand
CLK (Not used in the Combinatorial version)	Input	Clock – Data inputs are captured and new output data formed on rising clock transitions.
Y[2N-1:0]	Output	Parallel data out – Product result.

Table 2: Combinatorial KCM Instantiation template

```
component <8-bit combinatorial> is
  port map (
     x: in  std_logic_vector (7 downto 0);
     y: out std_logic_vector (15 downto 0));
  end component;

component <12-bit combinatorial> is
  port map (
     x: in  std_logic_vector (11 downto 0);
     y: out std_logic_vector (23 downto 0));
  end component;

component <16-bit combinatorial> is
  port map (
     x: in  std_logic_vector (15 downto 0);
     y: out std_logic_vector (31 downto 0));
  end component;
```

Table 3: Registered KCM Instantiation template

```
component <8-bit registered> is
  Port map (
      x: in  std_logic_vector (7 downto 0);
      clk: in std_logic;
      y: out std_logic_vector (15 downto 0));
  end component;

component <12-bit registered> is
  port map (
      x: in  std_logic_vector (11 downto 0);
      clk: in std_logic;
      y: out std_logic_vector (23 downto 0));
  end component;

component <16-bit registered> is
  port map (
      x: in  std_logic_vector (15 downto 0);
      clk: in std_logic;
      y: out std_logic_vector (31 downto 0));
  end component;


component <20-bit registered> is
  port map (
      x: in  std_logic_vector (19 downto 0);
      clk: in std_logic;
      y: out std_logic_vector (39 downto 0));
  end component;
```

Foundation users can generate a Foundation schematic symbol for the generated multiplier by specifying 'Create macro symbol from netlist'. This function can be found under the 'Hierarchy' menu of the Foundation schematic editor, The EDIF netlist for the appropriate multiplier should be specified as the input.

Using KCM_VGEN

It is recommended that the user extract the files in the kcm_vgen.zip ZIP archive to a new directory. It is also recommended that the user generate all required macros in this directory, then copy them over to the project working directory.

The KCM_VGEN program is a DOS executable, therefore it must be run from a DOS box. To start it up, navigate to the KCM_VGEN directory from a DOS window and type, 'kcm_vgen'.

The program will prompt the user with a set of questions to determine how the macro should be generated. (When selecting macro names, please stay within the DOS 8-character file name limit.)

All macros are supplied as EDIF netlists (file extension '.edn') containing RPM (Relative Placement Macro) information. The RPM information constrains the macro to a specific shape and size, and ensures a certain level of performance.

Performance data

Those accustomed to using the XC4000X family of devices will find that even higher performance is possible with the Virtex parts. Performance data listed in the tables is based on the Virtex -5 speed grade. Results are reported in units of clock speed for the registered versions of the multiplier macros, and in units of nanoseconds of delay for the combinatorial macros.

In the case of combinatorial multipliers, the path includes the net connecting to the multiplier input in the test design, and ends with the final stage of logic in the macro (the Y outputs).

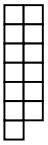
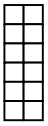
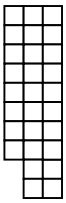
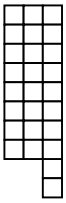
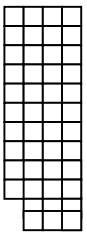
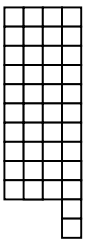
Higher clock speeds can be achieved by taking advantage of the on-chip DLL

Resource utilization

The size and shape of each macro is stated in terms of number of CLBs. In Virtex, a CLB is made up of 2 slices, and in some cases the CLB is not fully occupied. For this reason the slice count is also stated in Table 4.

For a given multiplier, you can use fewer resources and at the same time maintain a specific level of performance using the Virtex on-chip DLLs in conjunction with multi-cycle techniques. For example, you can use the same 8 bit KCM over 2 cycles plus an accumulator to implement a 16-bit function). The performance can be preserved by using a clock that has been doubled to run at twice the incoming speed using the DLL clock multiplier.

Table 4: Characterization Data

Core	CLB Count	Slice Count	Area Required for RPM (Rows, Columns)	Latency	Virtex-5 MHz/ns
8-bit pipelined	13	20	7,2 	2	165 MHz
8-bit combinatorial	12	18	6,2 	0	9 ns
12-bit pipelined	28	54	10,3 	3	148 MHz
12-bit combinatorial	26	42	10,3 	0	14 ns
16-bit pipelined	46	80	12,4 	3	142 MHz
16-bit combinatorial	42	72	12,4 	0	15 ns
20-bit pipelined	66	132	14,5	4	165MHz (-6)

Xilinx Reference Design License

By using the accompanying Xilinx, Inc. Reference Designs (the “Designs”), you agree to the following terms and conditions. You may use the Designs solely in support of your use in developing designs for Xilinx programmable logic devices or Xilinx HardWire™ devices. Access to the Designs is provided only to purchasers of Xilinx programmable logic devices or Xilinx HardWire™ devices for the purposes set forth herein.

The Designs are provided by Xilinx solely for your reference, for use as-is or as a template to make your own working designs. The Designs may be incomplete, and Xilinx does not warrant that the Designs are completed, tested, or will work on their own without revisions. The success of any designs you complete using the Designs as a starting point is wholly dependent on your design efforts. As provided, Xilinx does not warrant that the Designs will provide any given functionality, and all verification must be completed by the customer.

Xilinx specifically disclaims any obligations for technical support and bug fixes, as well as any liability with respect to the Designs, and no contractual obligations are formed either directly or indirectly by use of the Designs. XILINX SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION DIRECT, INDIRECT, INCIDENTAL, SPECIAL, RELIANCE OR CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF THE DESIGNS, EVEN IF XILINX HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Xilinx makes no representation that the Designs will provide the functionality you are looking for, or that they are appropriate for any given use. Xilinx does not warrant that the Designs are error-free, nor does Xilinx make any other representations or warranties, whether express or implied, including without limitation implied warranties of merchantability or fitness for a particular purpose. The Designs are not covered by any other license or agreement you may have with Xilinx.

The Designs are the copyrighted, confidential and proprietary information of Xilinx. You may not disclose, reproduce, transmit or otherwise copy the Designs by any means for any purpose not set forth in this license, without the prior written permission of Xilinx.

You agree that you will comply with all applicable governmental export rules and regulations, and that you will not export or reexport the Designs in any form without the appropriate government licenses.

XILINX, INC., 2100 Logic Drive, San Jose, California 95124