AccelChip Inc.

1900 McCarthy Blvd, Ste 204
Milpitas, CA 95035
Phone: (408) 943-0700
Fax: (408) 943-0661
E-mail: sales@accelchip.com
URL: www.accelchip.com

Features

- Inverts square matrices from 3x3 to 64x64 elements
- Fixed-point arithmetic for high-performance with up to 53 bits of precision
- Based on the triangular-orthogonal (QR) factorization of the input matrix
- Implementation based on Givens Rotations in conventional form using CORDIC rotations
- Includes reference MATLAB model and self-checking RTL testbench
- Core can be imported into Xilinx System Generator for DSP

Core Facts

Provided with Core	
Documentation	Users Guide
Design File Formats	VHDL, Verilog
Constraints Files	NA
Verification	Testbench generated from MATLAB, Test Vectors
Instantiation templates	VHDL, Verilog
Reference designs & application notes	NA
Additional Items	
Simulation Tools Used	
MATLAB (The MathWorks), ModelSim (Mentor Graphics)	
Support	
Provided by AccelChip Inc.	

Applications

- Kalman filtering
- Space-Time Adaptive Processing (STAP)
- Wireless signal processing
- Beamforming
- Software Defined Radio (SDR)
- Radar / Sonar
- Guidance, navigation and control
- Geo-/Astrophysical exploration
- Biomedical signal processing

Table 1: Core Implementation Statistics

Supported Family	Devices	Fmax (MHz)	Slices[1]	IOB[2]	Throughput	BRAM	MULT/ DSP48	Design Tools
4 x 4 Matrix Inverse, 12 bit input, 22 bit output								
Virtex-II Pro™	XC2VP30-7	46.3MHz	3758	550	92.14 KSPS	0	31	ISE 6.3.03i
Virtex-4™	XC4VSX55-11	46.5MHz	3514	550	92.55 KSPS	0	34	ISE 6.3.03i

Notes:

1) Actual slice count dependent on percentage of unrelated logic – see Mapping Report File for details

2) Assuming all core I/Os and clocks are routed off-chip

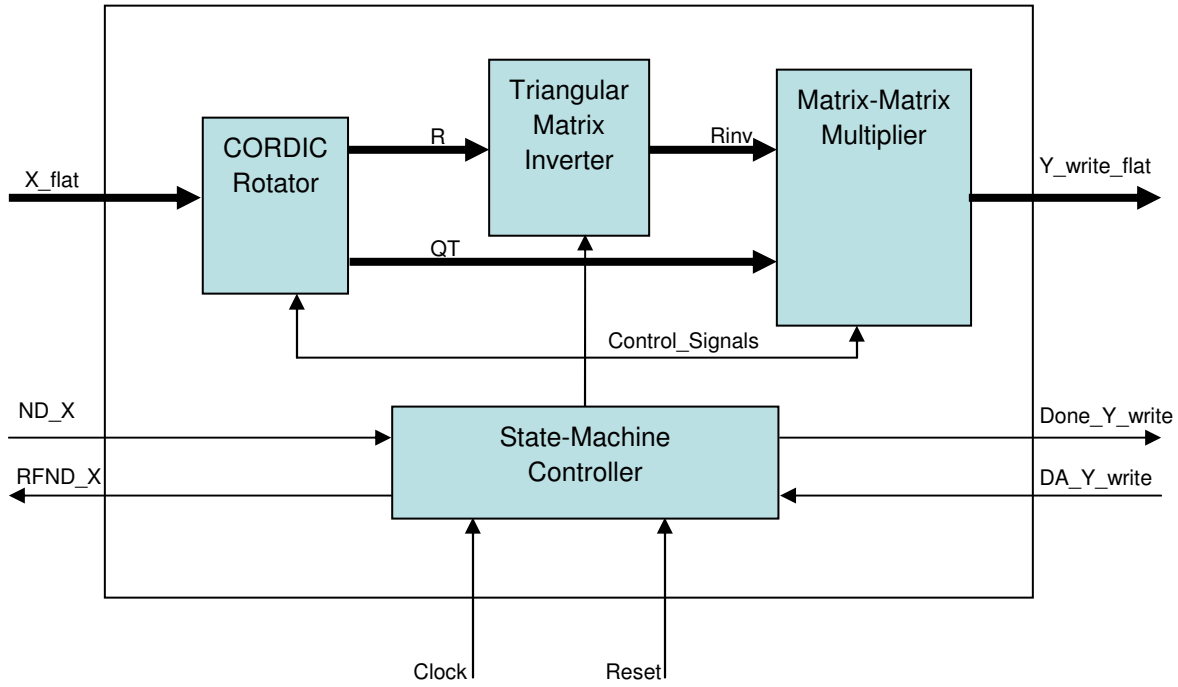

Figure 1: AccelCore QR Matrix Inverse Core Block Diagram

General Description

The AccelCore QR Matrix Inverse core computes the inverse of a real-valued, square input matrix. The implementation of the inversion is based on the triangular-orthogonal (QR) factorization of the input matrix followed by a product of the inverse of the matrix factors. (For applications where only QR factorization is required, contact AccelChip regarding availability of the AccelCore QR Matrix Factorization core.)

Functional Description

The AccelCore QR Matrix Inverse core uses a QR factorization algorithm based on Givens Rotations (GR) to produce the triangular (R) and orthogonal (Q) factors. The Givens Rotations are implemented in their conventional form using a COordinate Rotation DIgital Computer (CORDIC) for the vector rotations required to null elements below the diagonal and produce the upper triangular matrix [for further information, see ***Advanced Digital Signal Processing***, John G. Proakis et al., Macmillan Publishing Company, New York, New York, 1992]. The resulting R and Q factors are then each inverted and multiplied to produce the output inverse matrix Y.

Core Modifications

There are two levels of core modifications possible with AccelCore IP. First, there are implementation parameters specific to each type of core – these parameters are listed in the table below. Additionally, AccelChip can perform further optimizations based on customer speed and area requirements, such as rolling/unrolling of algorithmic loops or parallel implementations of matrix operations.

The following core-specific parameters are available for the AccelCore QR Matrix Inverse core.

Parameter Name	Parameter Description	Range
X quantizer	Input matrix quantization	2 to 24 bits
Implementation Algorithm	Algorithm used for factorization implementation	Conventional Givens Rotation
Input data type	Input matrix data type	Real
Matrix size	Input matrix size (specifies the number of rows and columns)	Integer between 3 and 64
Input/Output type	Input/output matrix dimension representation	1-D or 2-D
Output precision	Number of bits for output matrix	Up to 32 bits

Input Matrix Quantization

The number representation of the input is defined by the input matrix quantization. The AccelCore QR Matrix Inverse core accepts real-valued, fixed-point input data with quantization parameters defined by this quantization. The user can specify the total bitwidth and fractional bitwidth of values in the input matrix.

Implementation Algorithm

The matrix factorization implementation algorithm is based on conventional Givens Rotations with CORDIC.

Input Data Type

The input matrix for this core must be real-valued. If the input matrix has complex values, however, AccelChip can provide assistance in adapting the AccelCore QR Matrix Inverse core to complex matrices.

Matrix Size

The matrix size defines the number of rows and columns of the input and output matrices handled by the AccelCore QR Matrix Inverse core.

Input/Output Data Type

The AccelCore QR Matrix Inverse core can be generated to accept input and generate output matrices as 1-D or 2-D arrays.

Output Precision

The parameters that define the numerical precision of the inverse output matrix are automatically computed during the generation of the AccelCore QR Matrix Inverse core. The output precision can also be affected during synthesis of the core with the AccelChip DSP Synthesis tool by setting quantization directives. This allows great flexibility to the user to explore numerical precision and hardware implementation area/speed tradeoffs.

Core I/O Signals

The core I/O signals have not been fixed to specific device pins to provide flexibility for interfacing with user logic. Descriptions of all signal I/O are provided in Table 2.

Table 2: Core I/O Signals.

Signal	Signal Direction	Description
Clock	Input	Clock Input
Reset	Input	Reset Input
X_flat	Input	Input matrix data
RFND_X	Output	Ready for new data
ND_X	Input	New input data valid
Y_write_flat	Output	Output matrix data
Done_Y_write	Output	Done indication
DA_Y_write	Input	Data accepted indication

Core Assumptions

All implementations of matrix inversion can be sensitive to the condition number of the input matrix. This core is designed to operate on well-conditioned matrices, so the designer should take steps to ensure that that the input matrix is not ill-conditioned.

Verification Methods

The AccelCore QR Matrix Inverse core has a complete verification flow to ensure a bit-true implementation. The core is specfied using a fixed-point MATLAB model along with a set of stimulus and results. Based on the user's specifications, AccelChip produces an RTL model for the core, which is verified against the MATLAB model to ensure it is bit-true. AccelChip can also produce bit-true, cycle-accurate verification models for use with Simulink® from The MathWorks or Xilinx® System Generator for DSP.

Recommended Design Experience

The user should have some familiarity with linear algebra techniques and with HDL design methodologies.

Available Support Products

Customers may request that AccelChip provide versions of the AccelCore IP cores that can be imported into Xilinx System Generator for DSP. This allows the customer to verify the core using System Generator's system-level simulation facilities and libraries.

AccelChip Inc. uses proprietary algorithmic synthesis tools in the development of AccelCore IP cores. To obtain the AccelChip DSP Synthesis tool directly, contact your local AccelChip sales representative or send email to sales@accelchip.com.

Ordering Information

This product is available directly from AccelChip Inc. under the terms of the SignOnce IP License. The AccelCore QR Matrix Inverse core is available under the following AccelChip part numbers:

AccelChip Part Number	Description
QRI08	AccelCore QR Matrix Inverse Core – up to 8x8 matrix
QRI16	AccelCore QR Matrix Inverse Core – up to 16x16 matrix
QRIXX	AccelCore QR Matrix Inverse Core – larger than 16x16 matrix

Please contact your local AccelChip sales representative or send email to sales@accelchip.com.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com