CORDIC and SVD implementation on FPGA

Van Phu, Trong Huy, Thanh Huyen

Hanoi University of Science and Technology

Instructor: Dr. Ngo Vu Duc

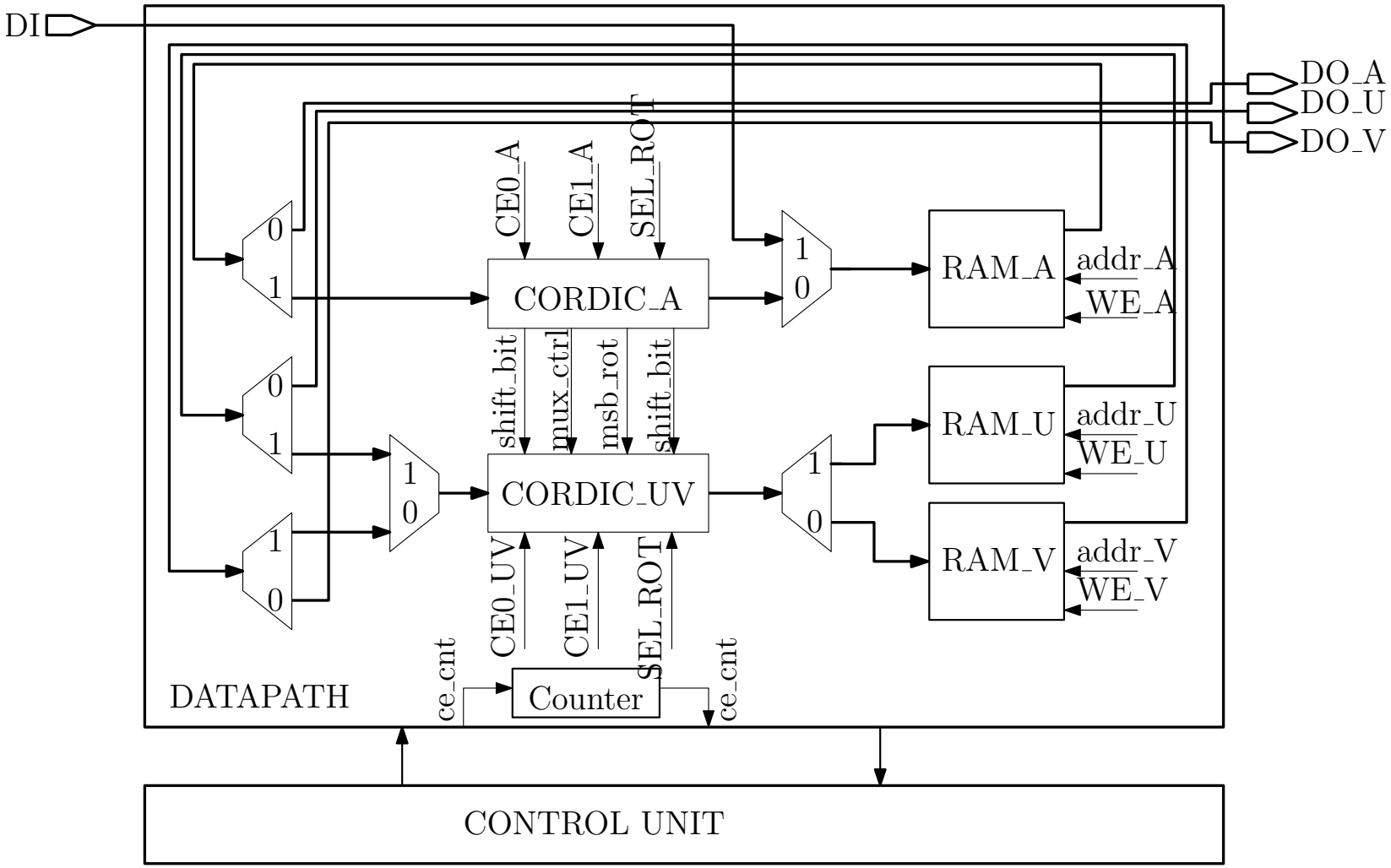
January 7, 2014

Overview

1. Specification and Block design
   - Block design
   - Block design
   - Block delay
   - Sub block design
2. Results and Demo

Specification and Block design

Block design



Specification and Block design

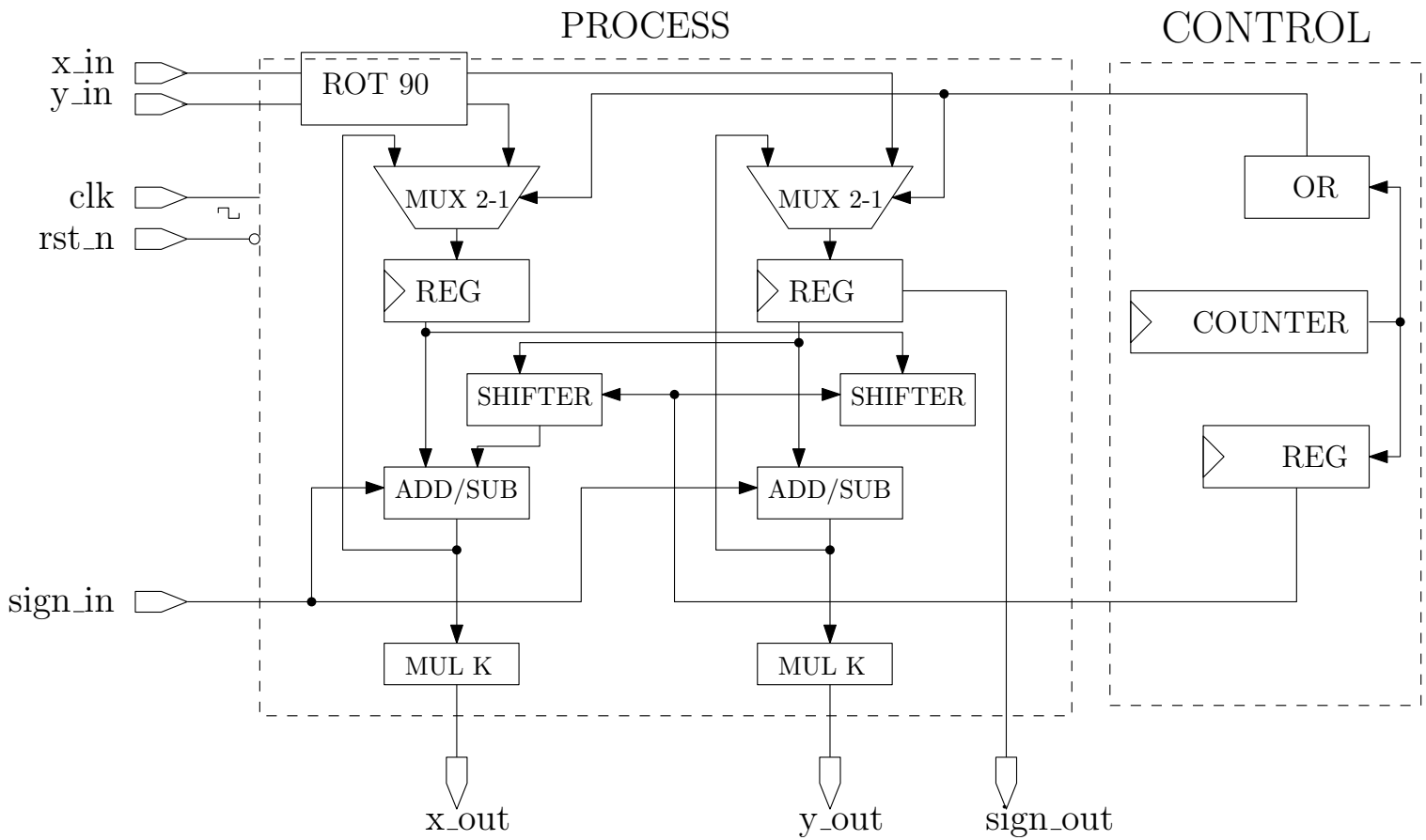
Block delay:

1. Load data in: 32 clks
2. Bidiag: 25 states
   each states: 17 clks (1 load out RAM, 15 CORDIC, 1 load in RAM)
3. Diag: 6 states
   each states:
   1. iteration: 16 times

   each times: 17 clks (1 load out RAM, 15 CORDIC, 1 load in RAM)
4. Load data out: 112 clks

Total delay: 2201 clks

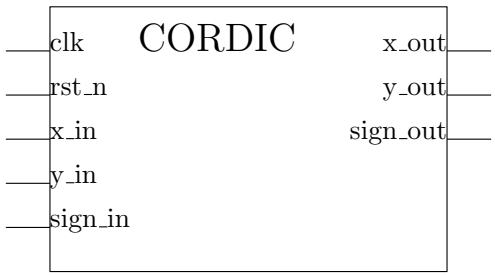
Diagram


PROCESS
CONTROL
x_in
y_in
clk
rst_n
sign_in
ROT 90
MUX 2-1
MUX 2-1
REG
REG
SHIFTER
SHIFTER
ADD/SUB
ADD/SUB
MUL K
MUL K
OR
COUNTER
REG
x_out
y_out
sign_out

Specification and Block design: Sub-Block

CORDIC

Block diagram

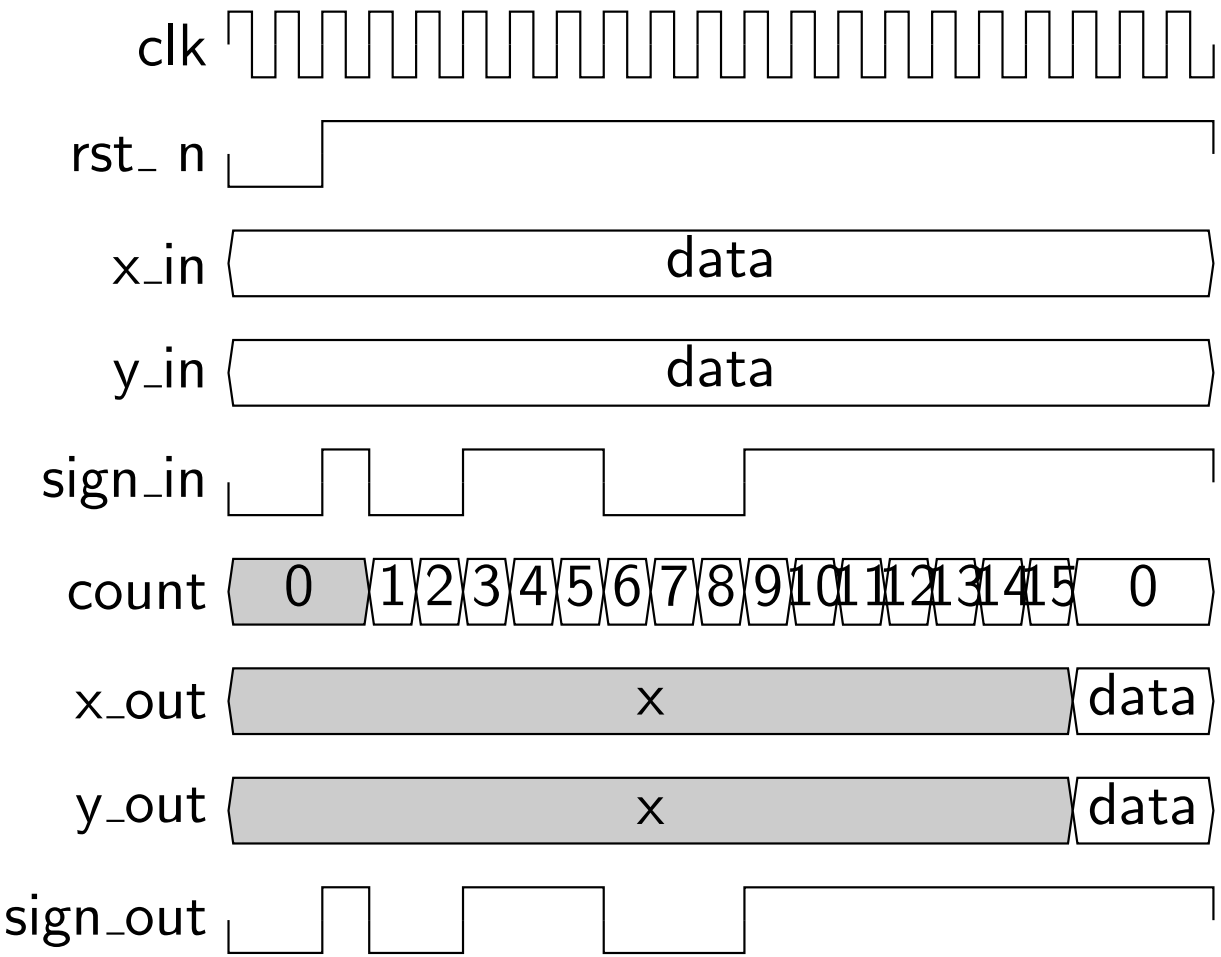


I/O port

Port	Data Width	Direction	Description
clk	1bit	Input	Clock
rst_n	1bit	Input	reset active low
x_in	24bits	Input	data in
y_in	24bits	Input	data in
sign_ in	1 bit	Input	sign bit input
x_ out	24 bits	Output	data out
y_ out	24 bits	Output	data out
sign_ out	1 bit	Output	sign bit output

Specification and Block design: Sub-Block

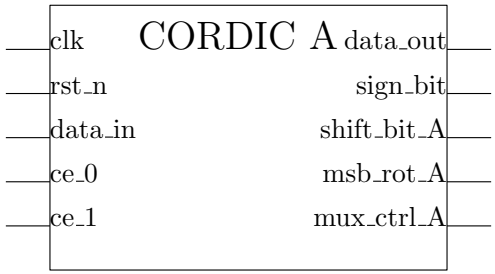
CORDIC



Specification and Block design: Sub-Block

CORDIC A

Block diagram

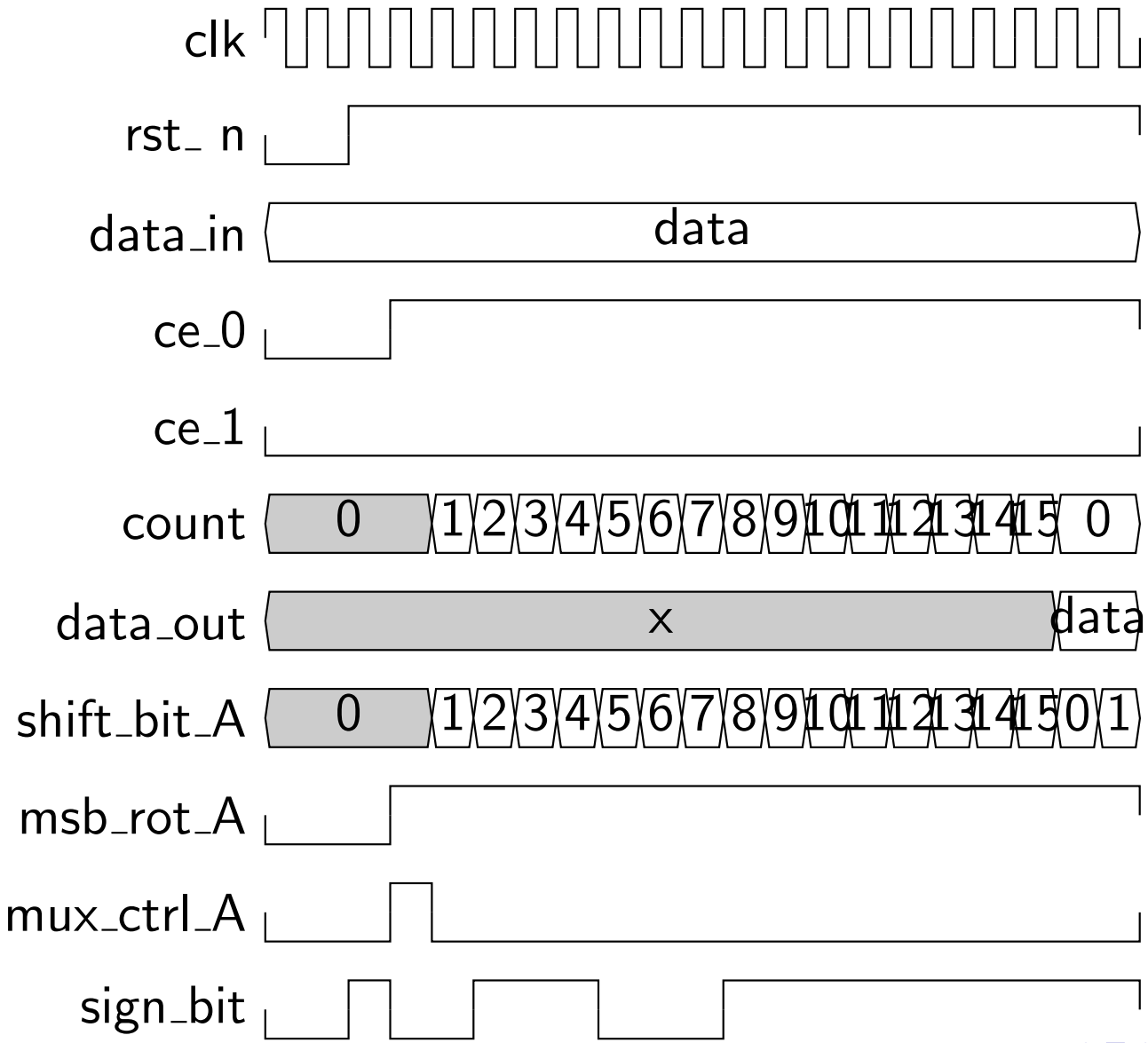


I/O port

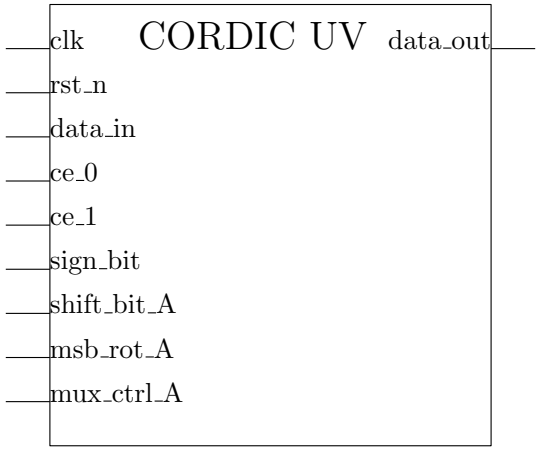
Port	Data Width	Direction	Description
clk	1bit	Input	Clock
rst_n	1bit	Input	reset active low
data_in	384bits	Input	data in
ce_0	1 bit	Input	enable/disable cordic slot 0
ce_1	1 bit	Input	enable/disable cordic slot 1
data_out	384bits	Output	data out
sign_bit	1 bit	Output	sign bit
shift_bit_A	4 bits	Output	shift bit signal is put out to CORDIC UV block
msb_rot_A	1 bit	Output	msb rotation signal is put out to CORDIC UV block
mux_ctrl_A	1 bit	Output	mux control signal is put out to CORDIC UV block

Specification and Block design: Sub-Block

CORDIC A



Block diagram

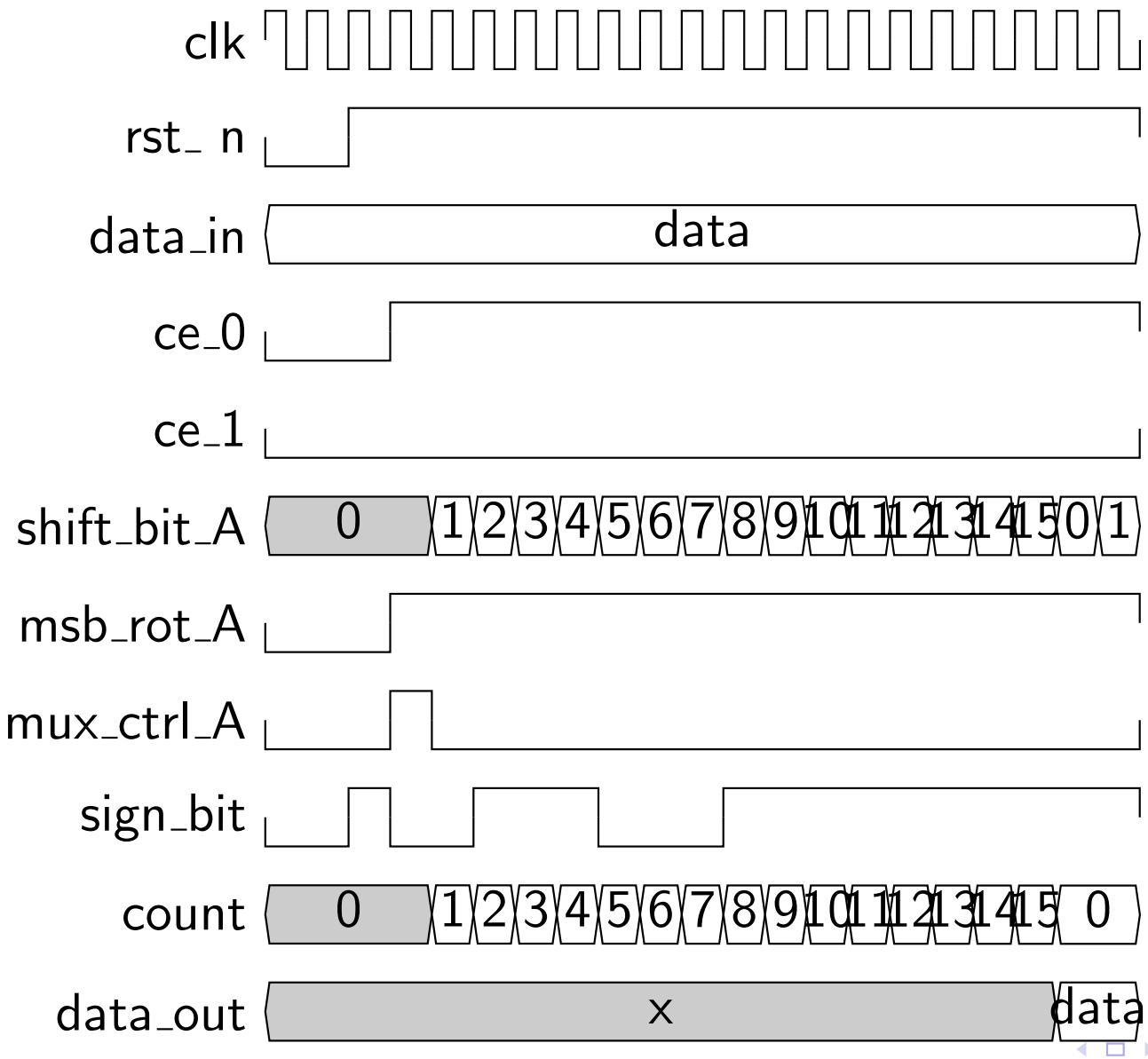


I/O port

Port	Data Width	Direction	Description
clk	1bit	Input	Clock
rst_n	1bit	Input	reset active low
data_in	384bits	Input	data in
ce_0	1 bit	Input	enable/disable cordic slot 0
ce_1	1 bit	Input	enable/disable cordic slot 1
sign_bit	1 bit	Input	sign bit
shift_bit_A	4 bits	Input	shift bit signal is put out from CORDIC UV block
msb_rot_A	1 bit	Input	msb rotation signal is put out from CORDIC UV block
mux_ctrl_A	1 bit	Input	mux control signal is put out from CORDIC UV block
data_out	384bits	Output	data out

Specification and Block design: Sub-Block

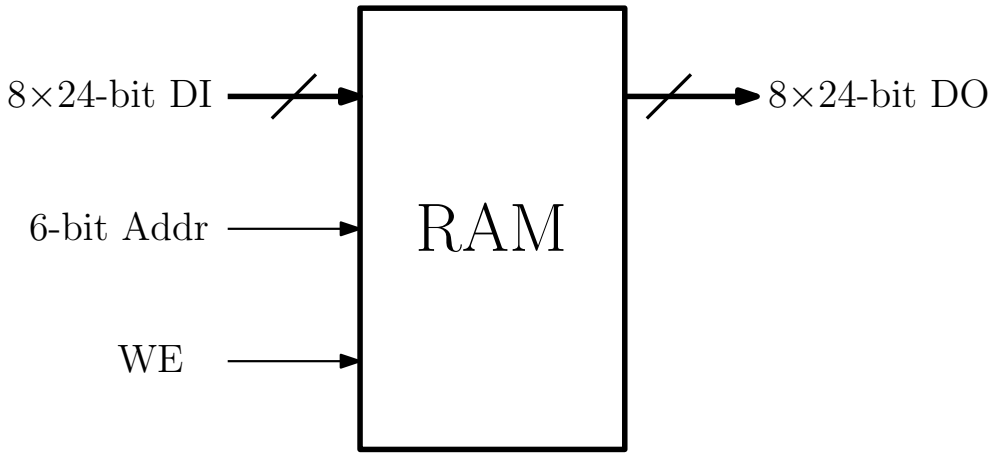
CORDIC UV



Specification and Block design: RAM

Diagram and I/O port

Diagram

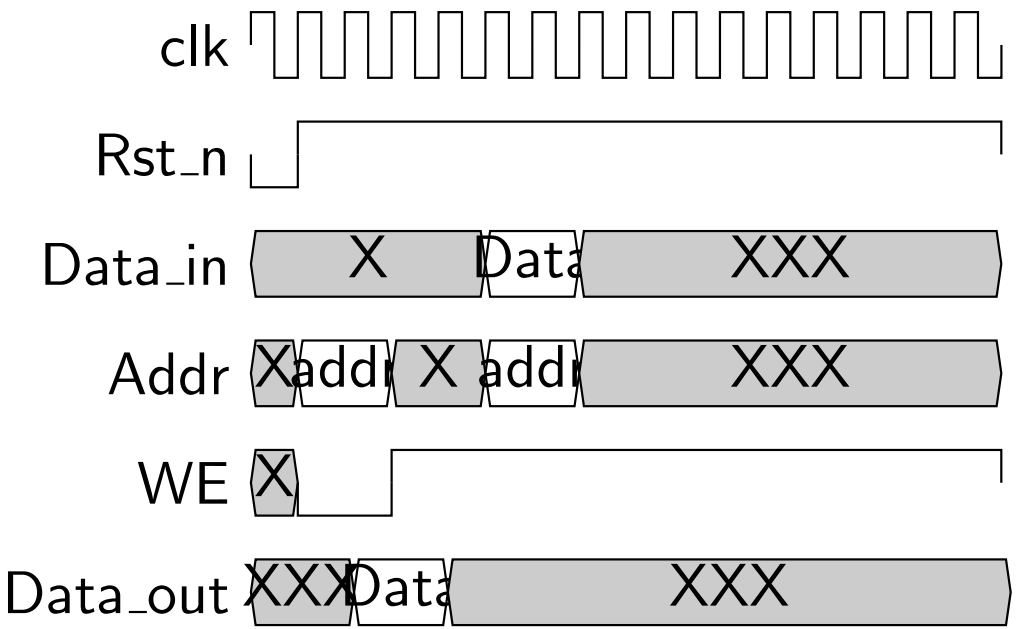


I/O port

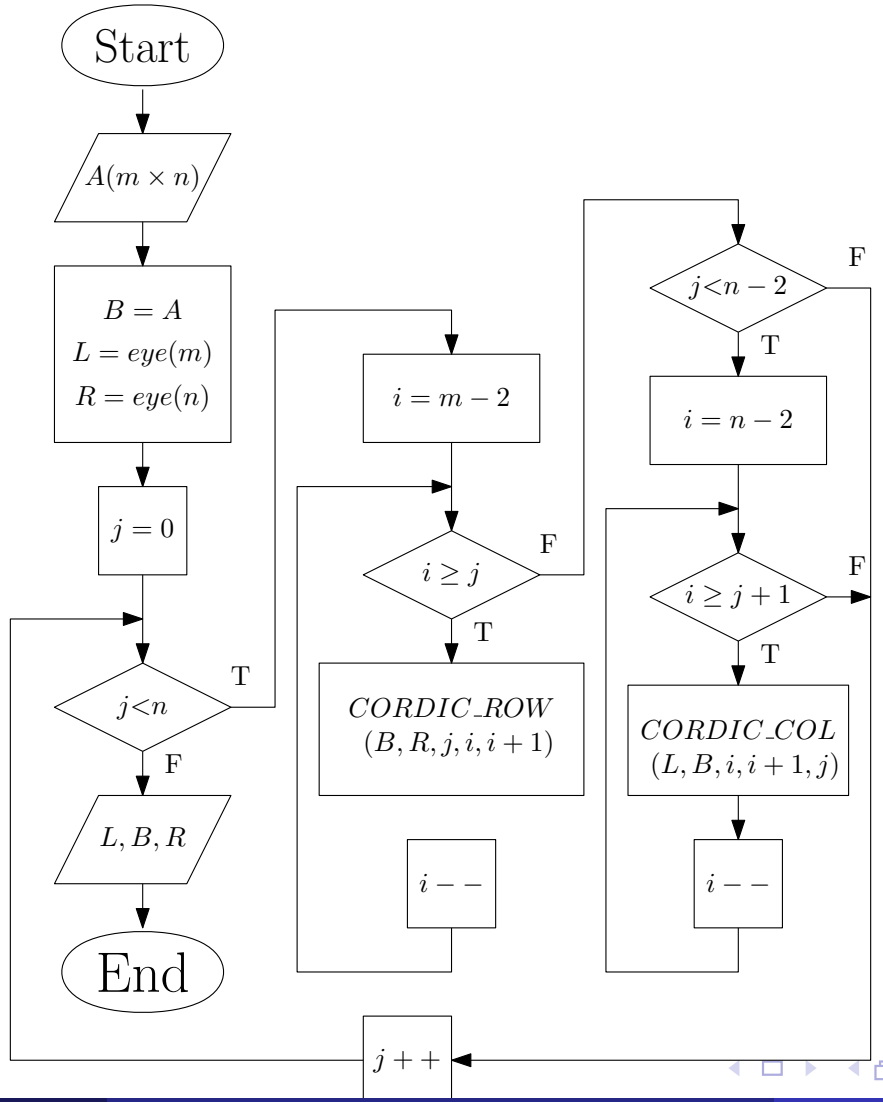
Port	Data Width	Direction	Description
clk	1bit	Input	Clock
rst_n	1bit	Input	reset active low
Data_in	8x24 bits	Input	data in
Addr	6bits	Input	data in
WE	1 bit	Input	sign bit input
DO	8x24 bits	Output	data out

Specification and Block design: RAM

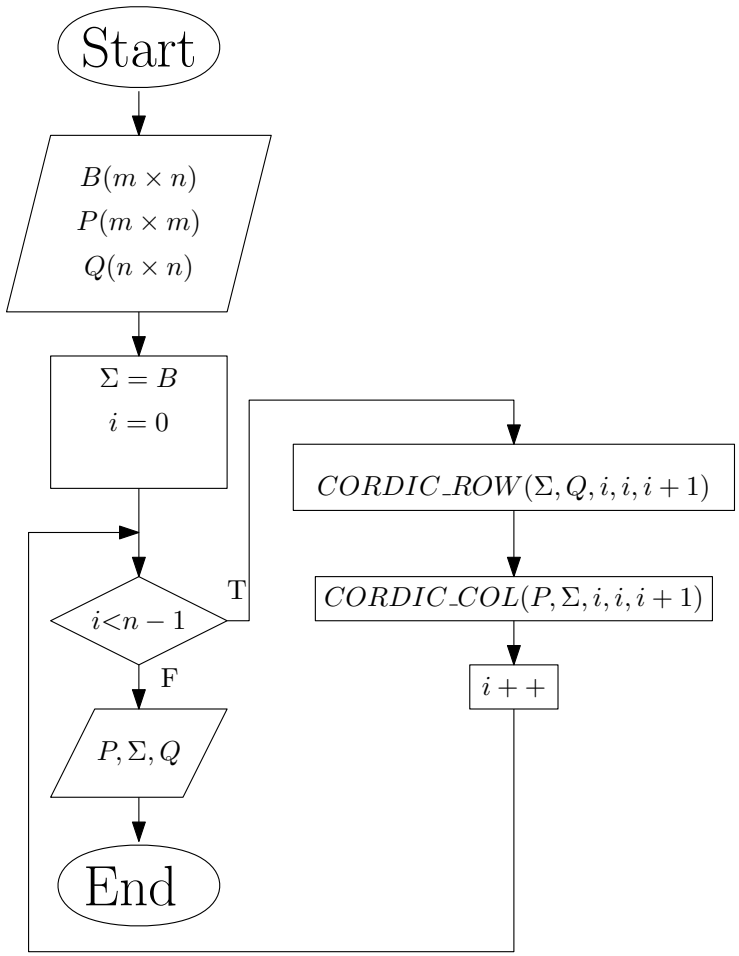
Timing



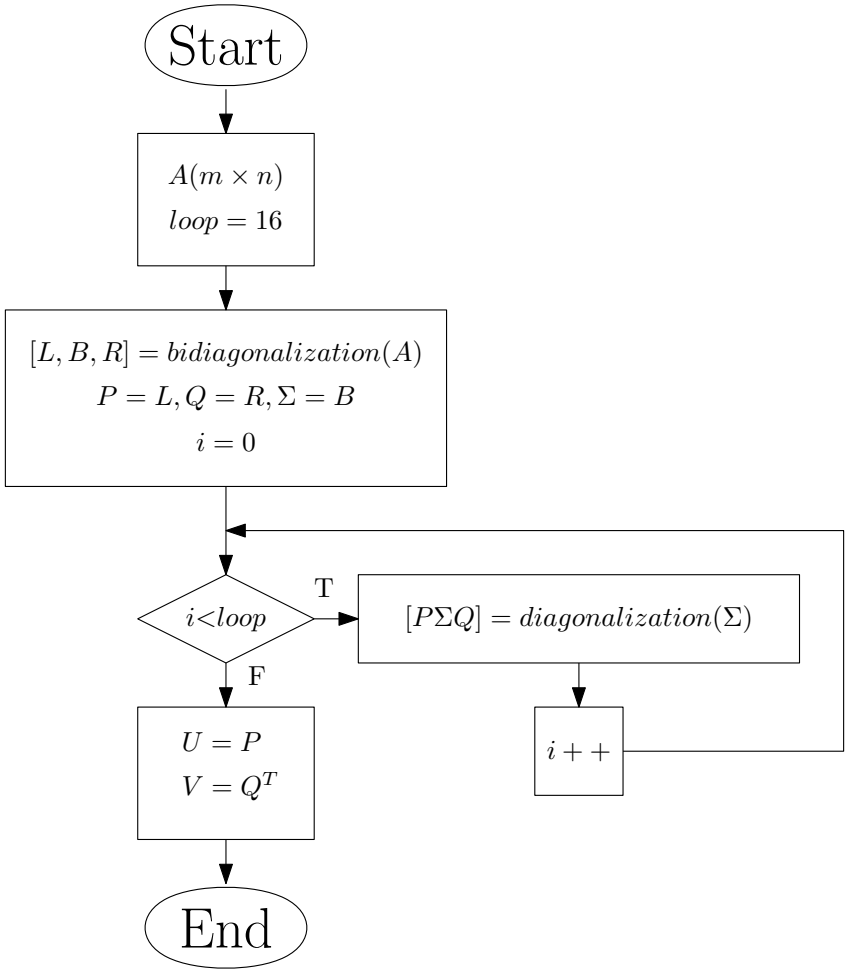
Bidiagonalization



Diagonalization



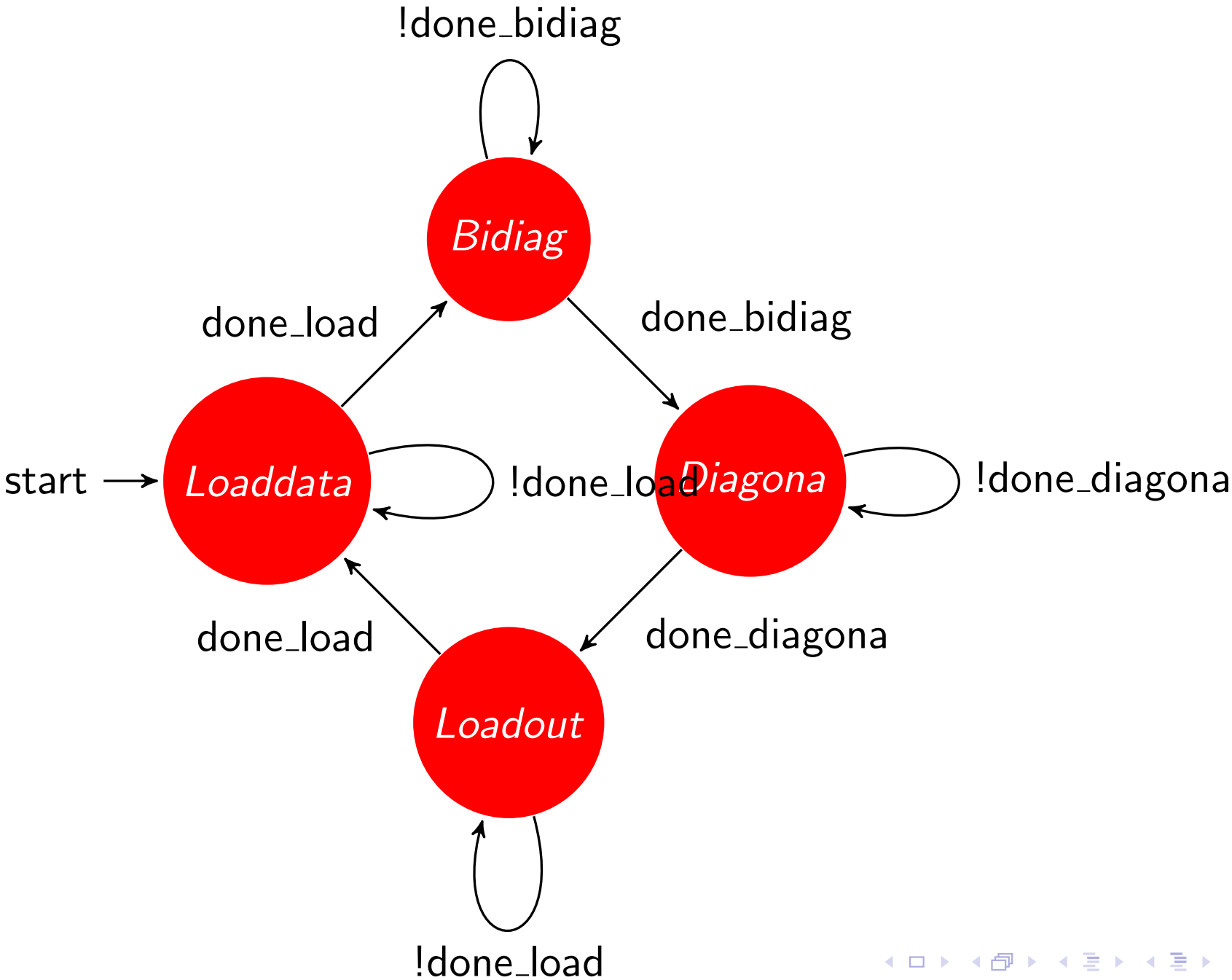
Bidiagonalization and Diagonalization



I/O port

Port	Data Width	Direction	Description
clk	1bit	Input	Clock
rst_n	1bit	Input	reset active low
cnt_svd	5bits	Input	count clock in each state
ce_svd	1 bit	Input	enable
ce_0_A	1 bit	Output	enable 2x4 cordic_0_A
ce_1_A	1 bit	Output	enable 2x4 cordic_1_A
ce_0_UV	1 bit	Output	enable 2x4 cordic_0_UV
ce_1_UV	1 bit	Output	enable 2x4 cordic_1_UV
mux_ctrl_0	1 bit	Output	control data in Ram A
demux_ctrl_0	1 bit	Output	control data out
mux_ctrl_1	1 bit	Output	control data in CORDIC UV
demux_ctrl_1	1 bit	Output	control data in Ram U,V
sel_cordic_rot	3 bit	Output	select element
ce_cnt	1 bit	Output	enable counter in Datapath
addr	6 bits	Output	address
we_A	1 bits	Output	write enable
we_U	1 bits	Output	write enable
we_V	1 bits	Output	write enable
reset	1 bits	Output	write enable
en_load_out	1 bits	Output	notice sign out

Diagram


!done_bidiag
Bidiag
done_load
done_bidiag
start
Loaddata
!done_load
Diagona
!done_diagona
done_load
done_diagona
Loadout
!done_load

Blocks of Highlighted Text

Bidiagonalization

Rotation

1. row : r6c0 $\div$ r0c0
2. column: r0c2 $\div$ r0c1
3. row: r6c1 $\div$ r1c1
4. column r1c2
5. row: r6c2 $\div$ r2c2
6. row: r6c3 $\div$ r3c3

Diagonalization

Rotation

1. column: c_r0c0
2. row: r_r0c0
3. column: c_r1c1
4. row: r_r1c1
5. column: c_r2c2
6. row: r_r2c2

Results

Hardware Implementation Results

The proposed SVD hardware computation unit was implemented in a low-cost FPGA device from Xilinx *XC6SLX45*. Table 1 summarizes the hardware implementation results for 8×4 matrix of the proposed SVD hardware computation unit as a percentage of the available resources, as well as the maximum operation frequency.

Resource Utilization	Xilinx *XC6SLX45*
Number of slice register	4512(8%)
Number of slice LUTs	19206(70%)
Multipliers	0
Max. Op. Freq.	47.690MHz

Table: RESOURCE UTILIZATION OF THE PROPOSED FPGA-BASED SVD COMPUTATION UNIT

Table 2 shows the corresponding singular values with the minimum and maximum estimation errors for the case of a 8×4 matrix. This table also shows the elapsed time for the software and hardware implementations

Singular Value	Matlab	FPGA	% error
σ_1	2555697	2555356	0.013
σ_2	1940169	1939863	0.016
σ_3	1210372	1210014	0.029
σ_4	960308	960143	0.017

Table: SVD COMPUTATION OF A 8×4 MATRIX

The End