Discrete Cosine Transform

Outline

- Algorithm
- Hardware behavior
- Performance

Algorithm

Discrete Cosine Transform

- 8*8 Discrete Cosine Transform(DCT) can be formulated as $A^T X A$
  - X is the 8*8 input signal
  - A is the 8*8 coefficient matrix, as shown below
    - a~g are constants

a	a	a	a	a	a	a	a
b	d	e	g	-g	-e	-d	-b
c	f	-f	-c	-c	-f	f	c
d	-g	-b	-e	e	b	g	-d
a	-a	-a	a	a	-a	-a	a
e	-b	g	d	-d	-g	b	-e
f	-c	c	-f	-f	c	-c	f
g	-e	d	-b	b	-d	e	-g

Decompose DCT(1)

- Define
  - $Y=AX$
  - $Z=AXA^T=YA^T$
- We first consider the computation of Y
- $Y=$

a	a	a	a	a	a	a	a
b	d	e	g	-g	-e	-d	-b
c	f	-f	-c	-c	-f	f	c
d	-g	-b	-e	e	b	g	-d
a	-a	-a	a	a	-a	-a	a
e	-b	g	d	-d	-g	b	-e
f	-c	c	-f	-f	c	-c	f
g	-e	d	-b	b	-d	e	-g

*

X_0
X_1
X_2
X_3
X_4
X_5
X_6
X_7

Decompose DCT(2)

- By symmetry of A, we can rewrite Y=AX as

$$\begin{bmatrix} Y_0 \\ Y_2 \\ Y_4 \\ Y_6 \end{bmatrix} = \begin{bmatrix} a & a & a & a \\ c & f & -f & -c \\ a & -a & -a & a \\ f & -c & c & -f \end{bmatrix} * \begin{bmatrix} X_0+X_7 \\ X_1+X_6 \\ X_2+X_5 \\ X_3+X_4 \end{bmatrix}$$

$$\begin{bmatrix} Y_1 \\ Y_3 \\ Y_5 \\ Y_7 \end{bmatrix} = \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} X_0-X_7 \\ X_1-X_6 \\ X_2-X_5 \\ X_3-X_4 \end{bmatrix}$$

Decompose DCT(3)

- $Z=YA^T=(AY^T)^T$ is similar to the previous step
  - We compute $Z^T=AY^T$ instead

$$\begin{bmatrix} Z^T_0 \\ Z^T_2 \\ Z^T_4 \\ Z^T_6 \end{bmatrix} = \begin{bmatrix} a & a & a & a \\ c & f & -f & -c \\ a & -a & -a & a \\ f & -c & c & -f \end{bmatrix} * \begin{bmatrix} Y^T_0+Y^T_7 \\ Y^T_1+Y^T_6 \\ Y^T_2+Y^T_5 \\ Y^T_3+Y^T_4 \end{bmatrix}$$

$$\begin{bmatrix} Z^T_1 \\ Z^T_3 \\ Z^T_5 \\ Z^T_7 \end{bmatrix} = \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} Y^T_0-Y^T_7 \\ Y^T_1-Y^T_6 \\ Y^T_2-Y^T_5 \\ Y^T_3-Y^T_4 \end{bmatrix}$$

Decompose IDCT (1)

- Define
  - $Y=A^TX$
  - $Z=A^TXA=YA$
- We first consider the computation of Y
- Y=

a	b	c	d	a	e	f	g
a	d	f	-g	-a	-b	-c	-e
a	e	-f	-b	-a	g	c	d
a	g	-c	-e	a	d	-f	-b
a	-g	-c	e	a	-d	-f	b
a	-e	-f	b	-a	-g	c	-d
a	-d	f	g	-a	b	-c	e
a	-b	c	-d	a	-e	f	-g

*

X_0
X_1
X_2
X_3
X_4
X_5
X_6
X_7

Decompose IDCT (2)

- We can rewrite $Y=A^TX$ as

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} a & c & a & f \\ a & f & -a & -c \\ a & -f & -a & c \\ a & -c & a & -f \end{bmatrix} * \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} + \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix}$$

$$\begin{bmatrix} Y_7 \\ Y_6 \\ Y_5 \\ Y_4 \end{bmatrix} = \begin{bmatrix} a & c & a & f \\ a & f & -a & -c \\ a & -f & -a & c \\ a & -c & a & -f \end{bmatrix} * \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} - \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix}$$

Decompose IDCT (3)

- $Z=YA=(A^TY^T)^T$ is similar to the previous step
  - We compute $Z^T=A^TY^T$ instead

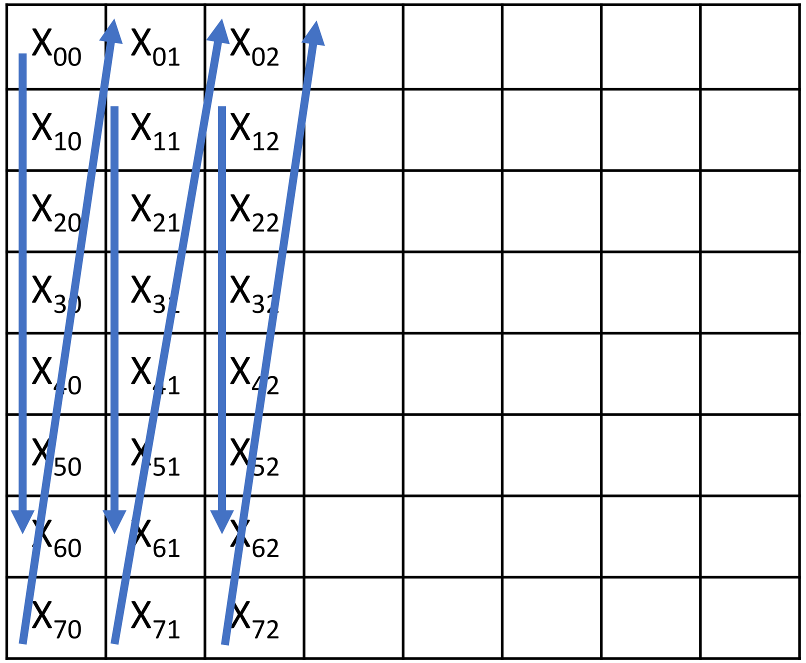
$$\begin{bmatrix} Z^T_0 \\ Z^T_1 \\ Z^T_2 \\ Z^T_3 \end{bmatrix} = \begin{bmatrix} a & c & a & f \\ a & f & -a & -c \\ a & -f & -a & c \\ a & -c & a & -f \end{bmatrix} * \begin{bmatrix} Y^T_0 \\ Y^T_2 \\ Y^T_4 \\ Y^T_6 \end{bmatrix} + \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} Y^T_1 \\ Y^T_3 \\ Y^T_5 \\ Y^T_7 \end{bmatrix}$$

$$\begin{bmatrix} Z^T_7 \\ Z^T_6 \\ Z^T_5 \\ Z^T_4 \end{bmatrix} = \begin{bmatrix} a & c & a & f \\ a & f & -a & -c \\ a & -f & -a & c \\ a & -c & a & -f \end{bmatrix} * \begin{bmatrix} Y^T_0 \\ Y^T_2 \\ Y^T_4 \\ Y^T_6 \end{bmatrix} - \begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} Y^T_1 \\ Y^T_3 \\ Y^T_5 \\ Y^T_7 \end{bmatrix}$$

Hardware Behavior

Data Input

- We assume the 8*8 input data follows the order below
- Input data will first go through Data Reorder Unit(DRU)



Data Reorder Unit (1)

- DRU either adds/subtracts the input or reorders them by parity
  - The behavior of DRU is shown below

	Clk_0	Clk_1	Clk_2	Clk_3	Clk_4	Clk_5	Clk_6	Clk_7	Clk_8	Clk_9	Clk_{10}	Clk_{11}	Clk_{12}	Clk_{13}	Clk_{14}	Clk_{15}	Clk_{16}	Clk_{17}
Input Data X	X_{00}	X_{10}	X_{20}	X_{30}	X_{40}	X_{50}	X_{60}	X_{70}	X_{01}	X_{11}	X_{21}	X_{31}	X_{41}	X_{51}	X_{61}	X_{71}	X_{02}	X_{12}
DRU Output Add						X_{30} + X_{40}	X_{20} + X_{50}	X_{10} + X_{60}	X_{00} + X_{70}					X_{31} + X_{41}	X_{21} + X_{51}	X_{11} + X_{61}	X_{01} + X_{71}	
DRU Output Sub						X_{30} - X_{40}	X_{20} - X_{50}	X_{10} - X_{60}	X_{00} - X_{70}					X_{31} - X_{41}	X_{21} - X_{51}	X_{11} - X_{61}	X_{01} - X_{71}	
DRU Output Even						X_{40}	X_{20}	X_{60}	X_{00}					X_{41}	X_{21}	X_{61}	X_{01}	
DRU Output Odd						X_{30}	X_{50}	X_{10}	X_{70}					X_{31}	X_{51}	X_{11}	X_{71}	

Data Reorder Unit (2)

- DRU saves the temporary values in a LIFO
  - Depth=4
  - X_{00} enters LIFO the first
  - X_{30} enters LIFO the last
  - X_{40} pairs with X_{30} (to compute $X_{40} \pm X_{30}$), so X_{30} leaves LIFO the first
  - Similarly, X_{50} pairs with X_{20}, X_{60} pairs with X_{10}, X_{70} pairs with X_{00}.

ACF/BDEG Matrix Multiplier (1)

- After decomposition
  - Coefficient matrixes are composed of either a, c, f or b, d, e, g
  - We separate them into two modules
    - ACF/BDEG matrix multipliers
    - They share the same behavior
    - They only differ by coefficients
- Each element of Y requires 4 pairs of data from DRU
  - Data pairs are first multiplied by all of the coefficients
  - Accumulator selects the right multiplication product and adds to the sum
    - 4 values are accumulated in parallel to keep up with the data rate

ACF/BDEG Matrix Multiplier (2)

- The behavior of ACF module when performing DCT is illustrated below

	Clk_5	Clk_6	Clk_7	Clk_8	Clk_9	Clk_{10}	Clk_{11}	Clk_{12}	Clk_{13}	Clk_{14}	Clk_{15}	Clk_{16}	Clk_{17}	Clk_{18}	Clk_{19}	Clk_{20}	Clk_{21}	Clk_{22}
DRU Output Add	$X_{30} + X_{40}$	$X_{20} + X_{50}$	$X_{10} + X_{60}$	$X_{00} + X_{70}$					$X_{31} + X_{41}$	$X_{21} + X_{51}$	$X_{11} + X_{61}$	$X_{01} + X_{71}$					$X_{32} + X_{42}$	$X_{22} + X_{52}$
ACF Multiply		$(X_{30} + X_{40})$ *acf	$(X_{20} + X_{50})$ *acf	$(X_{10} + X_{60})$ *acf	$(X_{00} + X_{70})$ *acf					$(X_{31} + X_{41})$ *acf	$(X_{21} + X_{51})$ *acf	$(X_{11} + X_{61})$ *acf	$(X_{01} + X_{71})$ *acf					$(X_{32} + X_{42})$ *acf
ACF Multiply Sum						Y_{00}	Y_{20}	Y_{40}	Y_{60}					Y_{01}	Y_{21}	Y_{41}	Y_{61}	

ACF/BDEG Matrix Multiplier (3)

- The behavior of BDEG module when performing IDCT is illustrated below

	Clk_5	Clk_6	Clk_7	Clk_8	Clk_9	Clk_{10}	Clk_{11}	Clk_{12}	Clk_{13}	Clk_{14}	Clk_{15}	Clk_{16}	Clk_{17}	Clk_{18}	Clk_{19}	Clk_{20}	Clk_{21}	Clk_{22}
DRU Output Odd	X_{30}	X_{50}	X_{10}	X_{70}					X_{31}	X_{51}	X_{11}	X_{71}					X_{32}	X_{52}
ACF Multiply		X_{30}* bdeg	X_{50}* bdeg	X_{10}* bdeg	X_{70}* bdeg					X_{31}* bdeg	X_{51}* bdeg	X_{11}* bdeg	X_{71}* bdeg					X_{32}* bdeg
ACF Multiply Sum						Y_{00} Y_{70} 2nd term	Y_{10} Y_{60} 2nd term	Y_{20} Y_{50} 2nd term	Y_{30} Y_{40} 2nd term					Y_{01} Y_{71}	Y_{11} Y_{71}	Y_{21} Y_{71}	Y_{31} Y_{71}	

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} a & c & a & f \\ a & f & -a & -c \\ a & -f & -a & c \\ a & -c & a & -f \end{bmatrix} * \begin{bmatrix} X_0 \\ X_2 \\ X_4 \\ X_6 \end{bmatrix} + \underbrace{\boxed{\begin{bmatrix} b & d & e & g \\ d & -g & -b & -e \\ e & -b & g & d \\ g & -e & d & -b \end{bmatrix} * \begin{bmatrix} X_1 \\ X_3 \\ X_5 \\ X_7 \end{bmatrix}}}_{2^{nd}\text{ term}}$$

Inverse Data Reorder Unit (1)

- IDRU either adds/subtracts the input or reorders them by the data order of Y
  - The behavior of IDRU when performing IDCT is shown below
  - IDRU saves the temporary values in a buffer

	Clk_{10}	Clk_{11}	Clk_{12}	Clk_{13}	Clk_{14}	Clk_{15}	Clk_{16}	Clk_{17}	Clk_{18}	Clk_{19}	Clk_{20}	Clk_{21}	Clk_{22}	Clk_{23}	Clk_{24}	Clk_{25}	Clk_{26}	Clk_{27}
BDEG Multiply Sum	Y_{00} Y_{70}	Y_{10} Y_{60}	Y_{20} Y_{50}	Y_{30} Y_{40}					Y_{01} Y_{71}	Y_{11} Y_{61}	Y_{21} Y_{51}	Y_{31} Y_{41}					Y_{02} Y_{72}	Y_{12} Y_{62}
ACF Multiply Sum	Y_{00} Y_{70}	Y_{10} Y_{60}	Y_{20} Y_{50}	Y_{30} Y_{40}					Y_{01} Y_{71}	Y_{11} Y_{61}	Y_{21} Y_{51}	Y_{31} Y_{41}					Y_{02} Y_{72}	Y_{12} Y_{62}
IDRU Output Y	$Y_{00} = Y_{00}^{1st} + Y_{00}^{2nd}$	$Y_{10} = Y_{10}^{1st} + Y_{10}^{2nd}$	$Y_{20} = Y_{20}^{1st} + Y_{20}^{2nd}$	$Y_{30} = Y_{30}^{1st} + Y_{30}^{2nd}$	$Y_{40} = Y_{40}^{1st} - Y_{40}^{2nd}$	$Y_{50} = Y_{50}^{1st} - Y_{50}^{2nd}$	$Y_{60} = Y_{60}^{1st} - Y_{60}^{2nd}$	$Y_{70} = Y_{70}^{1st} - Y_{70}^{2nd}$	$Y_{01} = Y_{01}^{1st} + Y_{01}^{2nd}$	$Y_{11} = Y_{11}^{1st} + Y_{11}^{2nd}$	$Y_{21} = Y_{21}^{1st} + Y_{21}^{2nd}$	$Y_{31} = Y_{31}^{1st} + Y_{31}^{2nd}$	$Y_{41} = Y_{41}^{1st} - Y_{41}^{2nd}$	$Y_{51} = Y_{51}^{1st} - Y_{51}^{2nd}$	$Y_{61} = Y_{61}^{1st} - Y_{61}^{2nd}$	$Y_{71} = Y_{71}^{1st} - Y_{71}^{2nd}$	$Y_{02} = Y_{02}^{1st} + Y_{02}^{2nd}$	$Y_{12} = Y_{12}^{1st} + Y_{12}^{2nd}$

Inverse Data Reorder Unit (2)

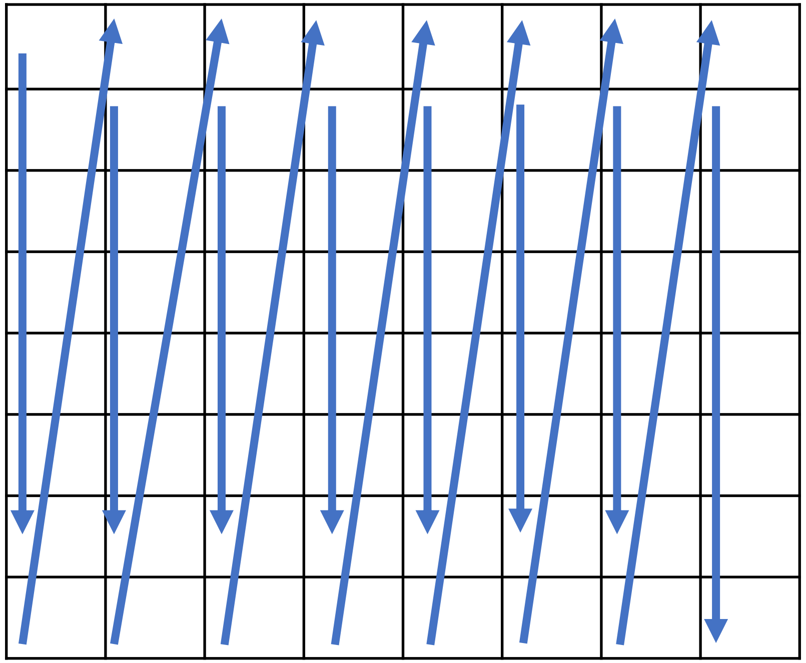
- The behavior of IDRU when performing DCT is shown below

	Clk_{10}	Clk_{11}	Clk_{12}	Clk_{13}	Clk_{14}	Clk_{15}	Clk_{16}	Clk_{17}	Clk_{18}	Clk_{19}	Clk_{20}	Clk_{21}	Clk_{22}	Clk_{23}	Clk_{24}	Clk_{25}	Clk_{26}	Clk_{27}
BDEG Multiply Sum	Y_{10}	Y_{30}	Y_{50}	Y_{70}					Y_{11}	Y_{31}	Y_{51}	Y_{71}					Y_{12}	Y_{32}
ACF Multiply Sum	Y_{00}	Y_{20}	Y_{40}	Y_{60}					Y_{01}	Y_{21}	Y_{41}	Y_{61}					Y_{02}	Y_{22}
IDRU Output Y	Y_{00}	Y_{10}	Y_{20}	Y_{30}	Y_{40}	Y_{50}	Y_{60}	Y_{70}	Y_{01}	Y_{11}	Y_{21}	Y_{31}	Y_{41}	Y_{51}	Y_{61}	Y_{71}	Y_{02}	Y_{12}

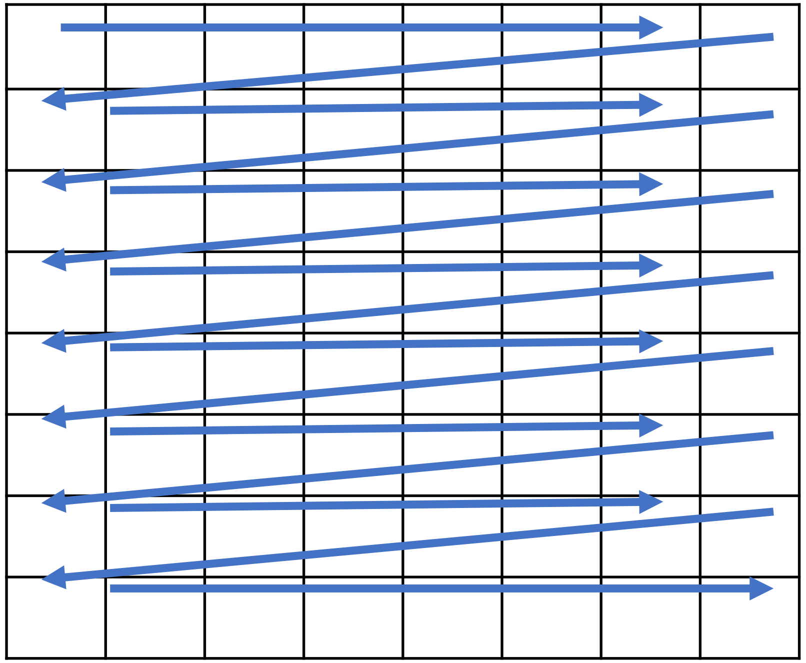
Transpose Matrix (1)

- Transpose Matrix is a 8*8 register array
  - Input Y from IDRU
  - Output Y^T to DRU
  - Reading order and writing order are different
    - As illustrated in the following page

Transpose Matrix (2)



- Input order of the first 8*8 data
- Output order of the second 8*8 data
- Input order of the third 8*8 data



- Output order of the first 8*8 data
- Input order of the second 8*8 data
- Output order of the third 8*8 data

Data Reorder Unit (3)

- To compute Z, we need Y^T
  - The output of Transpose Matrix is Y^T
- The rest of the Z computation is identical to the Y computation
  - We can reuse all of the existing components in a **4-cycle-interleaved** fashion
  - DRU receives Y^T from the transpose matrix as soon as Y_{00}~Y_{07} is ready
    - At this time, DRU could be processing the second 8*8 input X in parallel
- After going through DRU, ACF/BDEG, and IDRU the second time, IDRU outputs Z, the final result.

Data Reorder Unit (4)

- The complete DRU behavior

Input Data X	$X_{5,x}$	$X_{6,x}$	$X_{7,x}$	$X_{0,x+1}$	$X_{1,x+1}$	$X_{2,x+1}$	$X_{3,x+1}$	$X_{4,x+1}$	$X_{5,x+1}$	$X_{6,x+1}$	$X_{7,x+1}$	$X_{0,x+2}$	$X_{1,x+2}$
Input Data Y	Y_{01}	Y_{02}	Y_{03}	Y_{04}	Y_{05}	Y_{06}	Y_{07}	Y_{10}	Y_{11}	Y_{12}	Y_{13}	Y_{14}	Y_{15}
DRU Output Add	$X_{3,x}$ + $X_{4,x}$	$X_{2,x}$ + $X_{5,x}$	$X_{1,x}$ + $X_{6,x}$	$X_{0,x}$ + $X_{7,x}$	Y_{03} + Y_{04}	Y_{02} + Y_{05}	Y_{01} + Y_{06}	Y_{00} + Y_{07}	$X_{3,x+1}$ + $X_{4,x+1}$	$X_{2,x+1}$ + $X_{5,x+1}$	$X_{1,x+1}$ + $X_{6,x+1}$	$X_{0,x+1}$ + $X_{7,x+1}$	Y_{13} + Y_{14}
DRU Output Sub	$X_{3,x}$ - $X_{4,x}$	$X_{2,x}$ - $X_{5,x}$	$X_{1,x}$ - $X_{6,x}$	$X_{0,x}$ - $X_{7,x}$	Y_{03} - Y_{05}	Y_{02} - Y_{05}	Y_{01} - Y_{06}	Y_{00} - Y_{07}	$X_{3,x+1}$ - $X_{4,x+1}$	$X_{2,x+1}$ - $X_{5,x+1}$	$X_{1,x+1}$ - $X_{6,x+1}$	$X_{0,x+1}$ - $X_{7,x+1}$	Y_{13} - Y_{14}
DRU Output Even	$X_{4,x}$	$X_{2,x}$	$X_{6,x}$	$X_{0,x}$	Y_{04}	Y_{02}	Y_{06}	Y_{00}	$X_{4,x+1}$	$X_{2,x+1}$	$X_{6,x+1}$	$X_{0,x+1}$	Y_{14}
DRU Output Odd	$X_{3,x}$	$X_{5,x}$	$X_{1,x}$	$X_{7,x}$	Y_{03}	Y_{05}	Y_{01}	Y_{07}	$X_{3,x+1}$	$X_{5,x+1}$	$X_{1,x+1}$	$X_{7,x+1}$	Y_{13}

Performance

- Using TSMC 0.13um library
  - Area: 191006 um^2
  - Clock cycle for post-place-and-route simulation: 5.4ns
  - Throughput: 1 number(of 8*8 output data) per cycle
  - Latency: 80 cycles