IO 21000 Board Interface and a study of Reed-Solomon Code and circuit design

Speaker : Chi-Lin Huang , Hong-fu Chou

Advisor : Prof. Kwang-Cheng Chen , Prof. Mao-Chao Lin

Outline

- Interface of IO 21000 board
- Components of IO 21000 board
- Project Procedure
- A study of Reed-Solomon Code and circuit design
- Deliverable

Interface of IO21000 board (1/2)

- The Interfaces are consists of two parts of I/O:

The PCI interface:

(1)Input the data from demodulator (High rate Demodulator (HDR))

(2)Output the data from RS decoder

(3)Interface of GUI

- The SMA connector: Input data x 2, Output data x 2, and I/O clock x2 for two channels (i.e., total SMA x 8).

- The Input and output of SMA are designed for (a) testing purpose (b) external data I/O.

Interface of IO21000 board (2/2)

- Based on our observation from the IO21000 board in NSPO,
- (1) Input from demodulator: SMA data input and SMA clock input
- (2) Output from RS decoder: PCI interface

  where the Input of PCI interface is disable since input data from demodulator is ingested by SMA connector.
- (3) GUI interface: PCI interface

SMA connector (1/2)

- (1) Input data from demodulator: (serial, synchronized)
- Connector: SMA (female)
- Input: ECL Differential (default)
  LVDS, RS-422 or TTL (optional)
- Impedance: 50ohm terminated to -2volt(ECL)
- Data rate: 150M bps (maximum)

SMA connector (2/2)

- (2) Input clock: (serial , synchronized)
- Connector: SMA (female)
- Input: ECL Differential (default)
  LVDS, RS-422 or TTL (optional)
- Impedance: 50ohm terminated to -2volt(ECL)
- Duty cycle: 50% (error within 10 %)
- Clock phasing 0 or 180 degree

ECL (Emitter couple logic)

- ECL (Emitter couple logic)
  - a logic family in which current is steered through bipolar transistors to implement logic functions
  - Sometimes call current-mode logic or current-switch emitter-follower (CSEF) logic

LVDS (Low Voltage Differential Signal)

- LVDS (Low Voltage Differential Signal)
  - Low power consumption
  - Max data rate : 3.125 Gbps
  - Driven current : 350 ma
  - Input impedance: 100 Ω

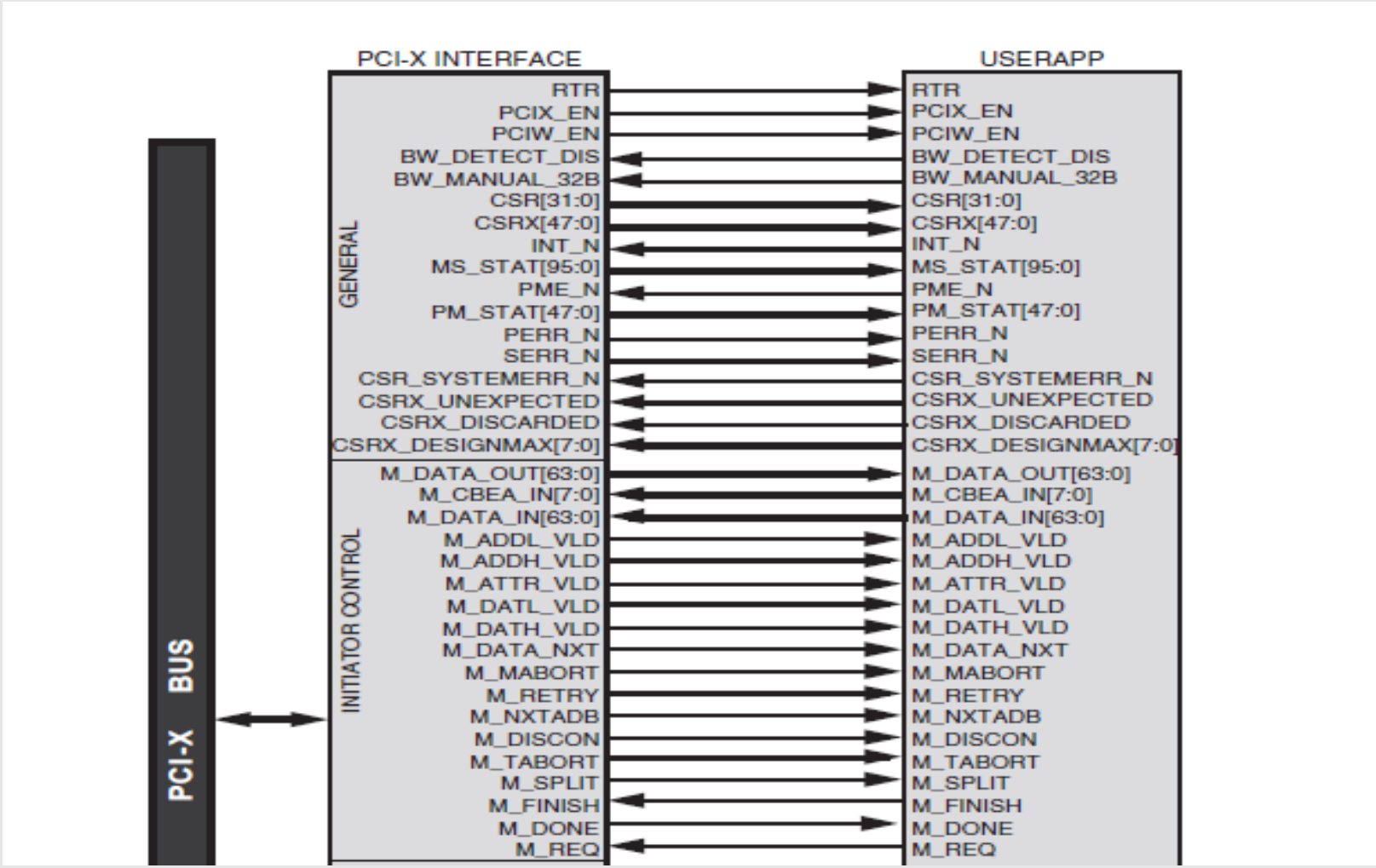
- Due to differential signaling such as excellent noise immunity and low device-generated switching noise.

PCI interface

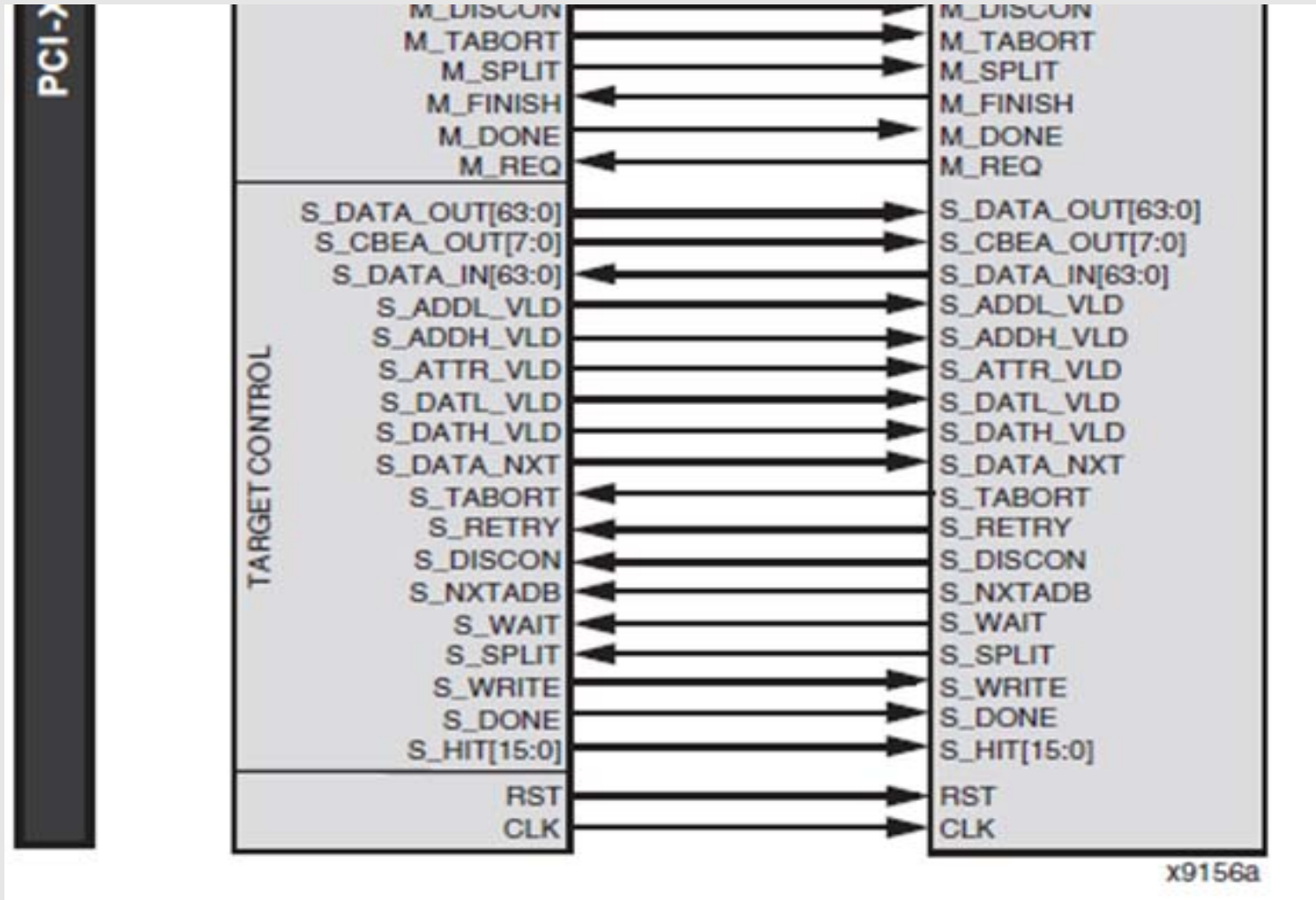
- PCI interface: PCI-X
- (1) Output from RS decoder (parallel, synchronized)
- (2) GUI interface(parallel, synchronized)

- All pins function can find in datasheet.

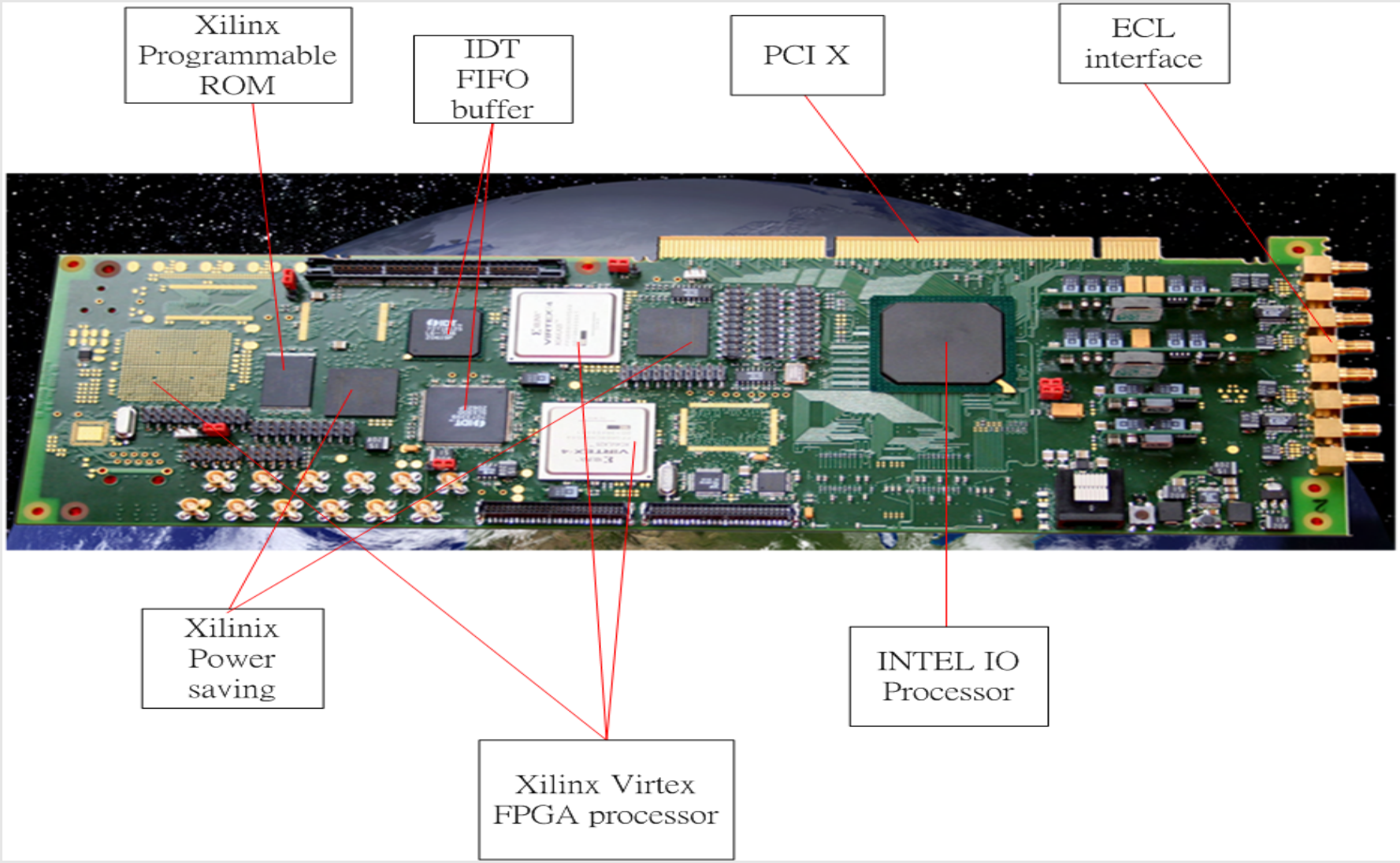
PCI interface pins configuration (1/2)



PCI interface pins configuration (2/2)



Components of IO 21000 board (1/2)



Components of IO 21000 board (2/2)

- 1. INTEL IO processor : FW80321M600
- 2. FPGA processor : Virtex-4 XC4VLX25-FF668-10C/I *3
- 3. Programmable ROM : Xilinx XCF32P VOG48
- 4. FIFO buffer : IDT 72V36104, IDT 72T18125
- 5. Power saving : XC2C256 CoolRunner-II CPLD

Intel IO processor (1/2)

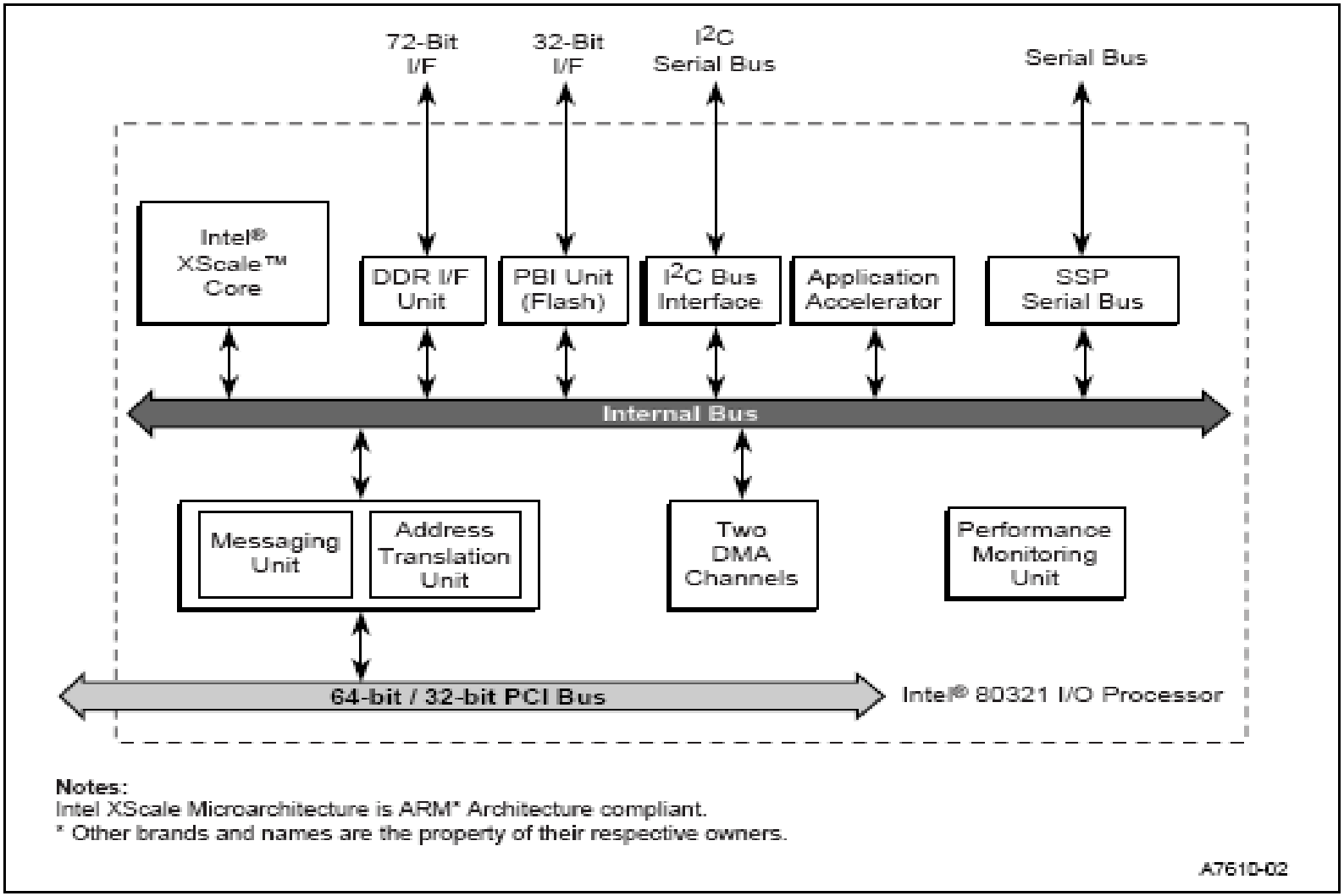
- Main function :
  - A single-function device that integrate the Intel XScale core with intelligent peripherals, including a PCI bus application bridge.
- Feature :

1. Core :
   - Intel XScale core with clock rate 600 MHz
   - ARM V5T instruction set
2. PCI bus interface: 64-bit/133 MHz Operation in PCI-X Mode
3. Memory Controller:
   - Up to 1 Gbyte of 64-bit DDR SDRAM
   - Up to 512 Mbytes of 32-bit DDR SDRAM
4. 554-Ball, Plastic Ball Grid Array
5. 8 general purpose I/O pins

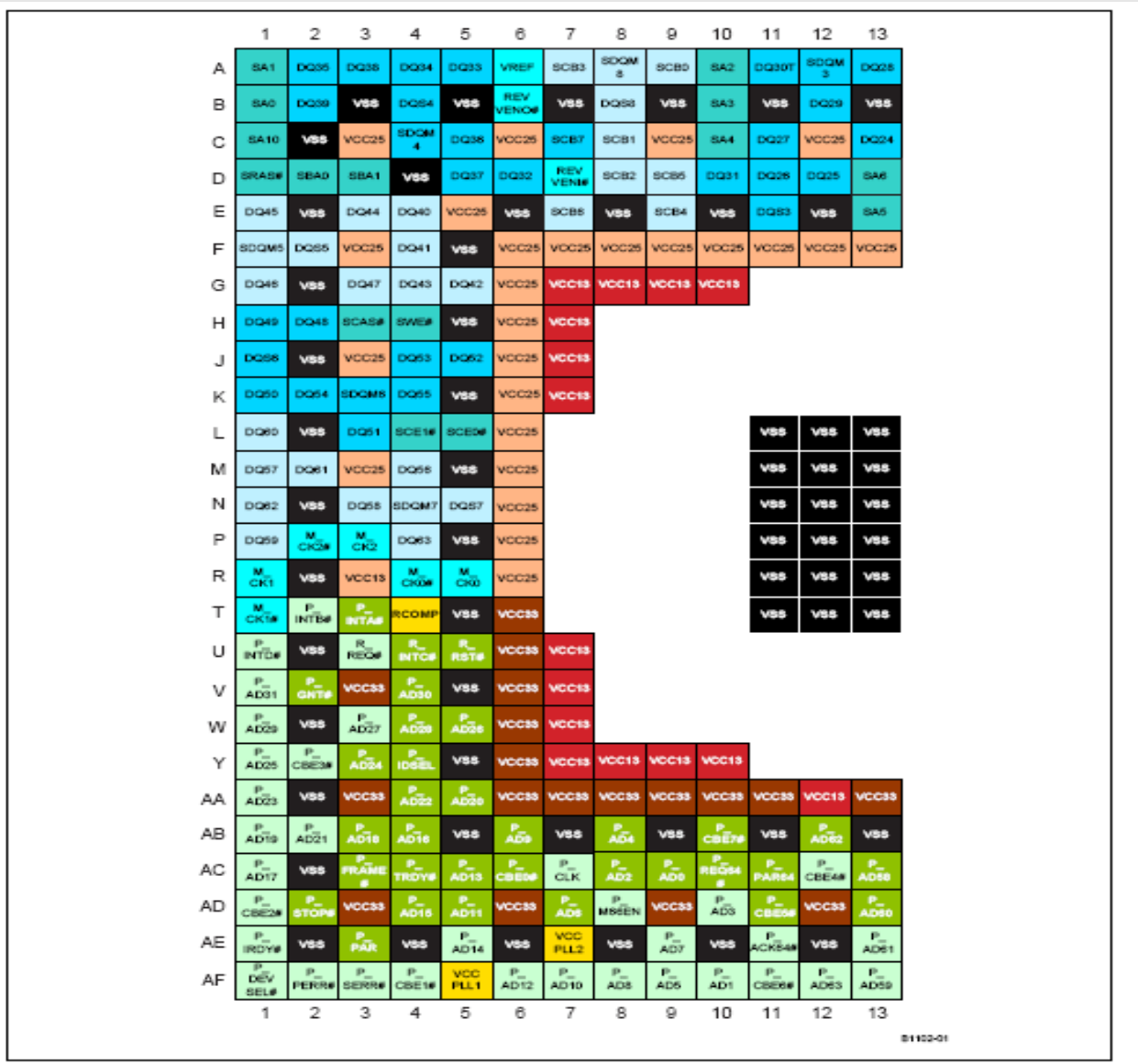
Intel IO processor (2/2)

- Major Interface :
  - Memory interface
    - DDR I/F unit output → 72-Bit I/F
    - PBI unit (Flash) → 32-Bit I/F
  - IIC serial bus interface → IIC serial bus
    - The IIC unit uses a serial bus developed by Philips Semiconductor* consisting of a two-pin interface
    - IIC DATA is used for data transfer and arbitration on the I^2C bus. This is one of two IIC buses that the user may enable
    - IIC CLOCK provides synchronous operation of the IIC bus. This is one of two I^2C buses that the user may enable
  - Synchronous serial port (SSP) unit → serial bus
    - a full-duplex synchronous serial interface
  - Message unit and Address translation unit → 64-bit / 32-bit PCI

Intel IO processor functional block diagram [3]

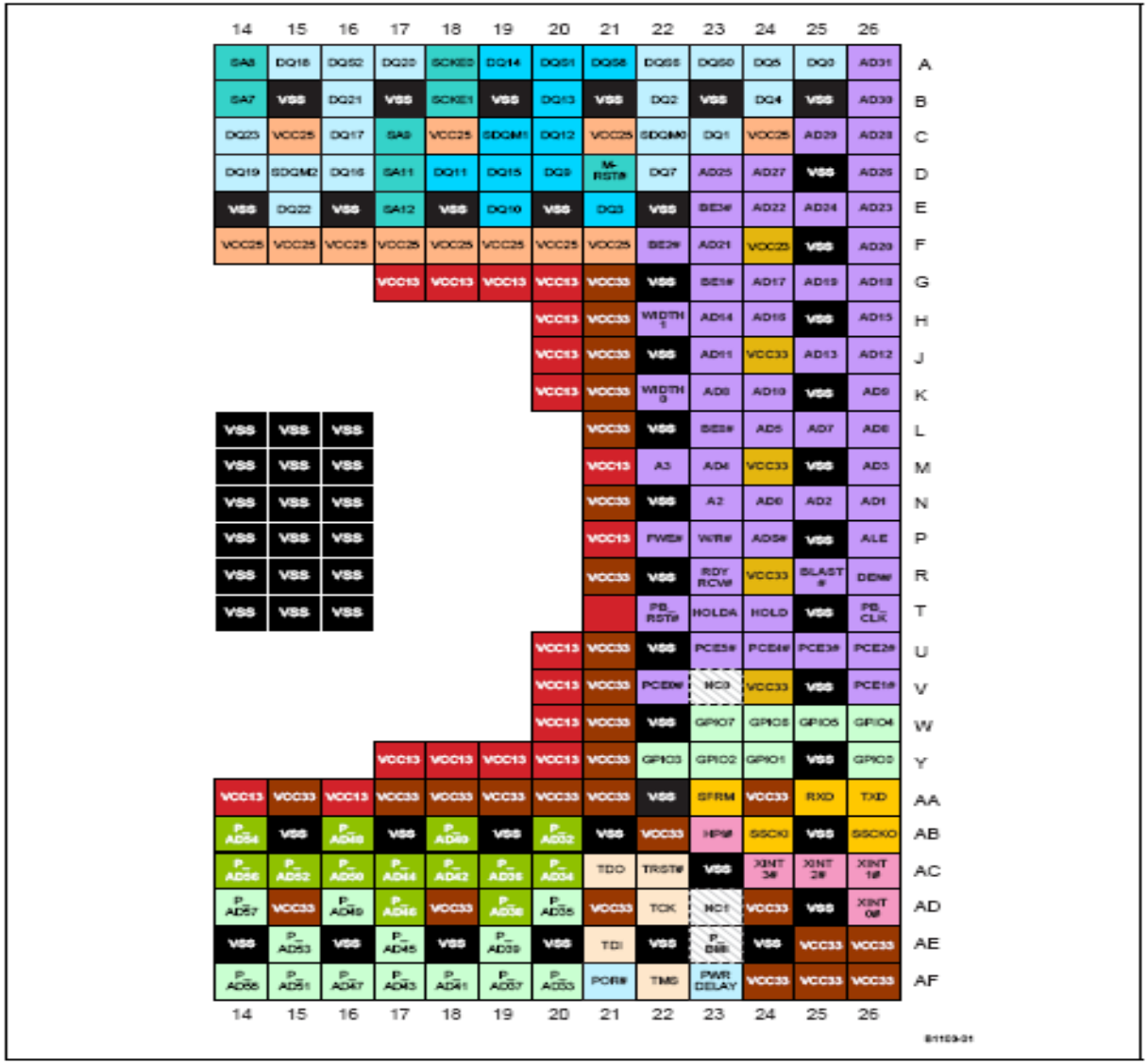


Intel IO processor Ball map (Left side) [3]



Intel IO processor Ball map (Right side) [3]

Figure 5. Ball Map - Right Side - Top View

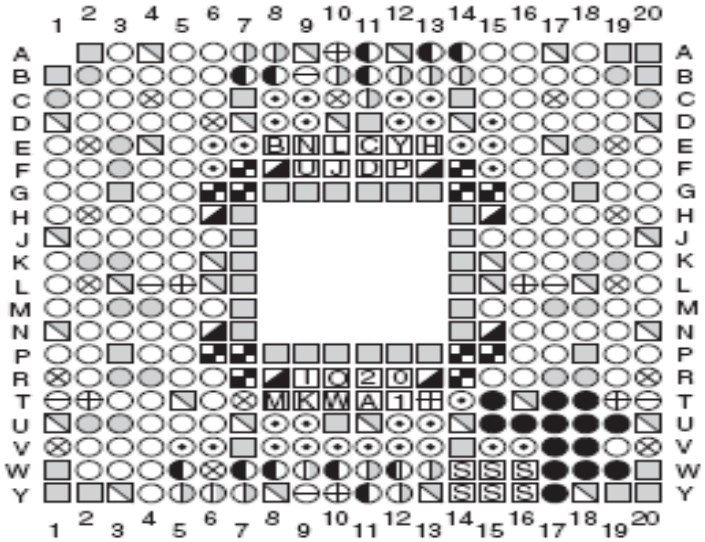


Xilinx Virtex 4 FPGA Processor

- Virtex-4 devices are produced on a state-of-the-art 90-nm copper process using 300-mm (12-inch) wafer technology.
- Feature:
  - 1.2V Core Voltage
  - Flexible Logic Resources
  - Max data rate: 500 MHz
  - I/O
    - Wide selections of I/O standards from 1.5V to 3.3V
    - Up to 960 user I/Os

Xilinx Virtex 4 FPGA Processor Pin out diagram [8]

SF363 (XC4VLX25) - Top View



User I/O Pins	Dedicated Pins		Other Pins
IO_LXXY_#	ADC	PROG_B	GND
Multi-Function Pins:	CCLK	PWRDWN_B	RSVD
ADC1 - ADC7	CS_B	RDWR_B	VBATT
D0 - D31	D_IN	SM	VCCAUX
CC	DONE	TCK	VCCINT
N_GC	DOUT_BUSY	TDI	VCCO
P_GC	HSWAPEN	TDO	NO CONNECT
LC	INIT	TMS	
SM1 - SM7	M2, M1, M0	TDP	
VREF		TDN	
VRN			
VRP			

Notes: 1. SM and ADC functionality in multi-function user I/O pins is reserved for future use.
2. Dedicated SM and ADC pins are reserved for future use.

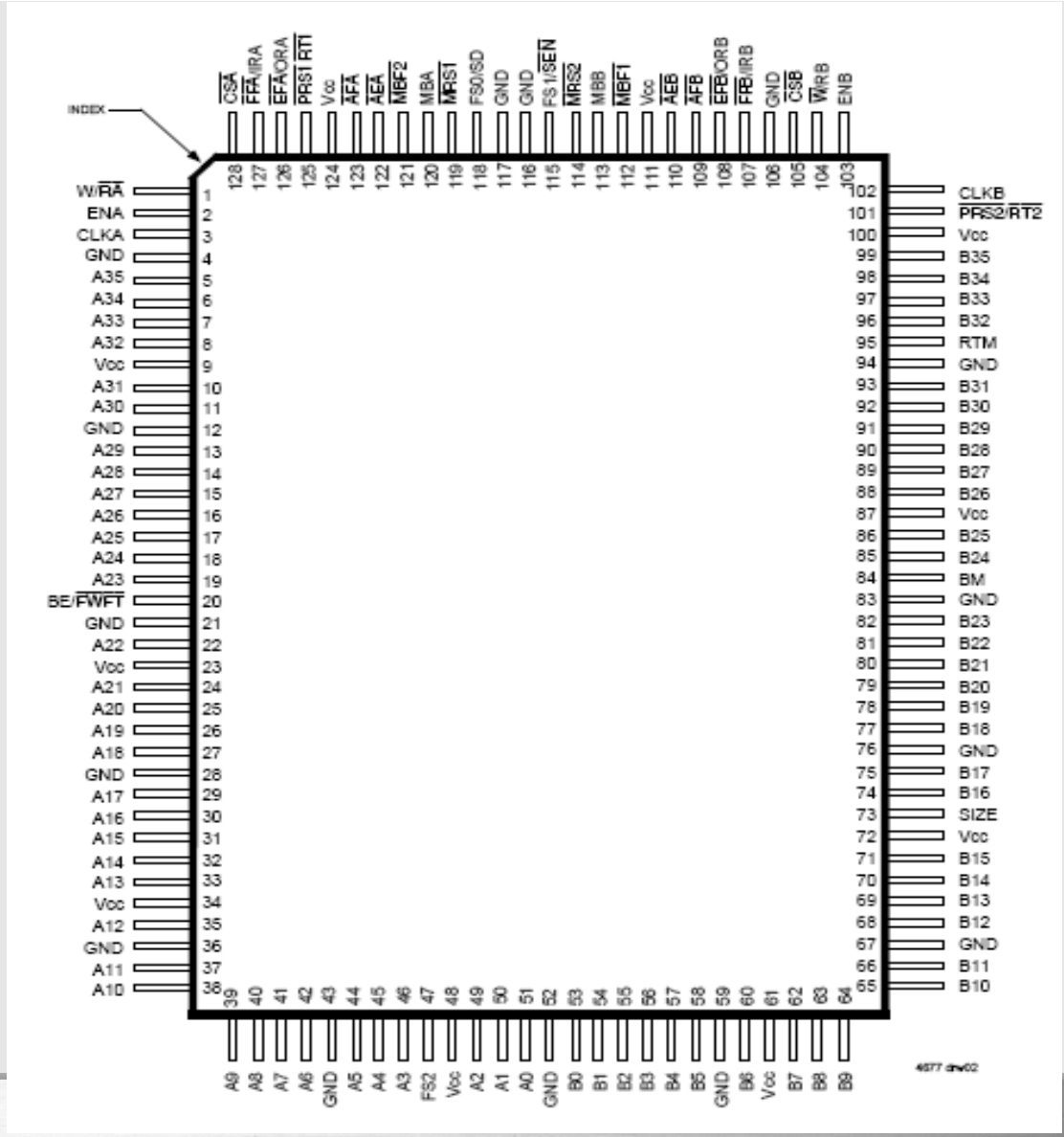
UG075_02_050108

IDT FIFO buffer (1/2)

- IDT_72V36104:
  - Memory storage capacity :
    - 65,536 x 36 x 2-bit
  - Supply voltage : 3.3 V
  - Clock frequencies up to 100 MHz
  - Free-running clock may be asynchronous or synchronous (simultaneous reading and writing of data on a single clock edge is permitted)
  - Two independent clocked FIFOs buffering data in opposite direction

- All pins function can find in datasheet

IDT_72V36104 FIFO buffer pin configuration [6]

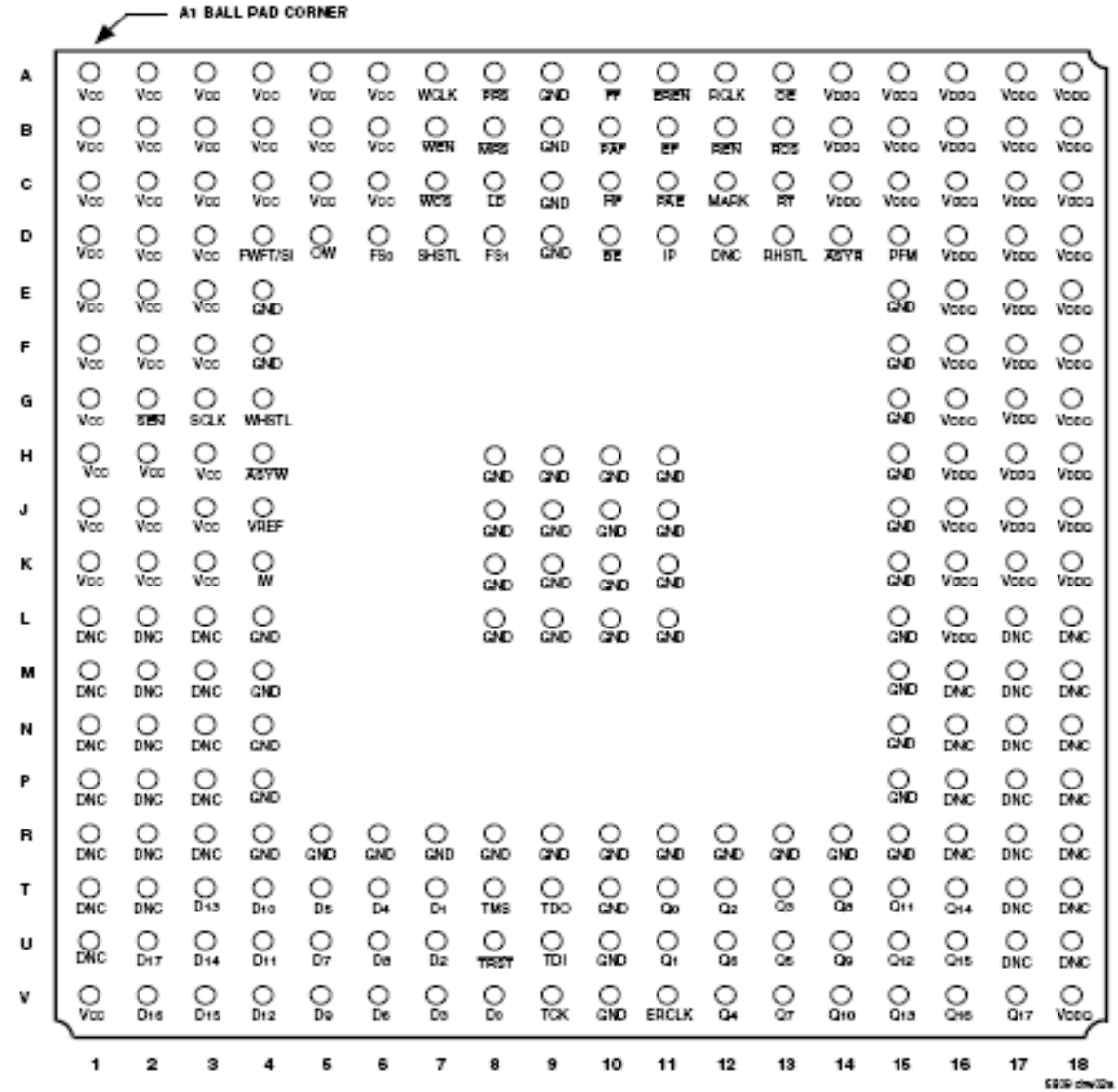


IDT FIFO buffer (2/2)

- IDT_72T18125
  - Memory storage capacity :
    - 524,288 x 18-bit/1,048,576 x 9-bit
  - Supply voltage : 3.3 V
  - Clock frequencies up to 225 MHz
  - Asynchronous/Synchronous translation on the read or write ports

- All pins function can find in datasheet.

IDT_72T18125 FIFO buffer pin configuration [7]



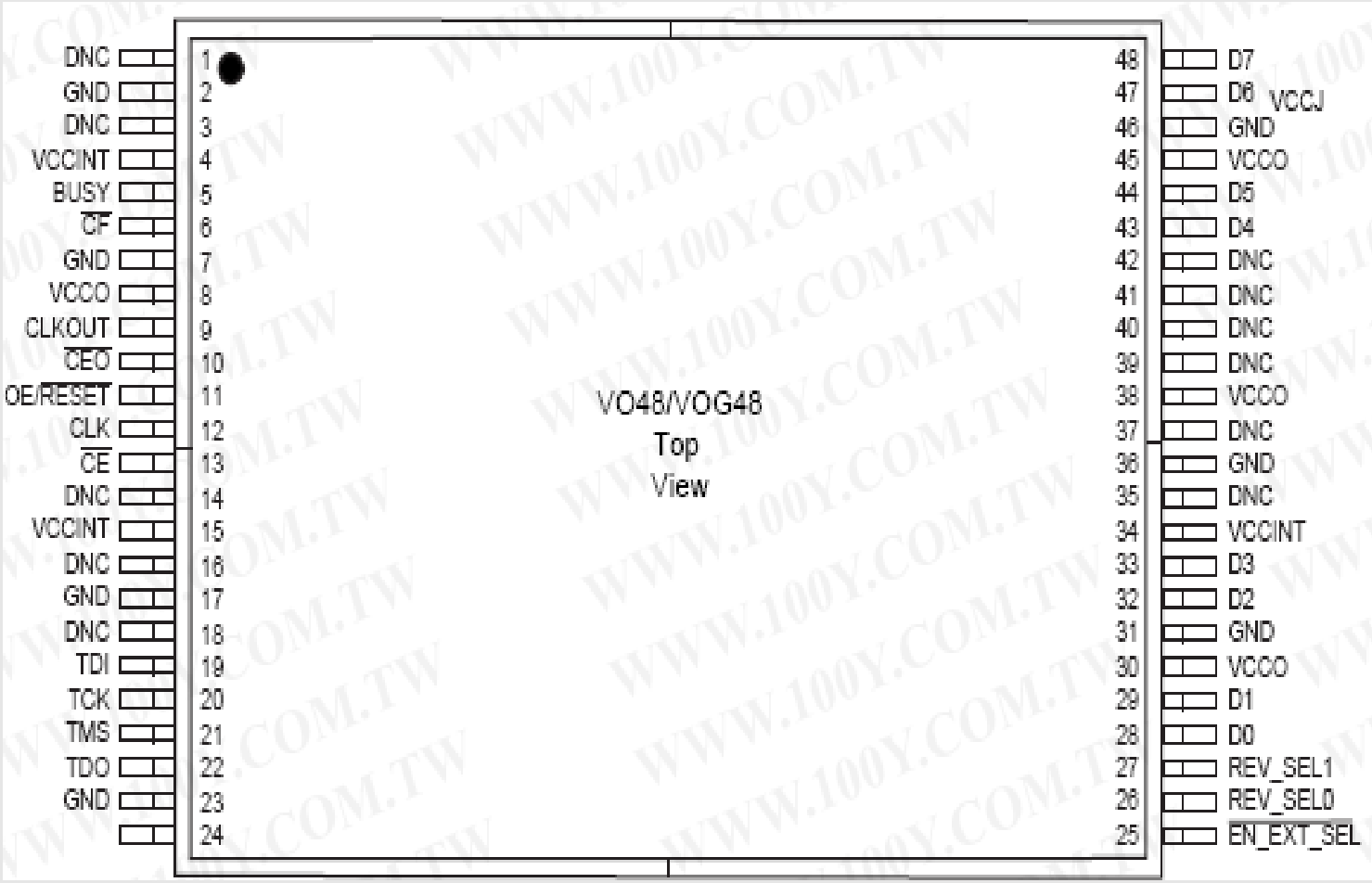
NOTE:
1. DNC - Do Not Connect.

Programmable ROM

- Xilinx XCF32P VOG48
  - Endurance of 20000 Program / Erase cycles
  - I/O
    - Dedicated Boundary-Scan (JTAG) I/O Power Supply
    - I/O pins compatible with voltage level ranging from 1.5v to 3.3 v
  - Supply voltage 1.8v
  - Serial or parallel FPGA configuration interface (up to 33 MHz)

- All pins function can find in datasheet

Programmable ROM pin configuration [4]



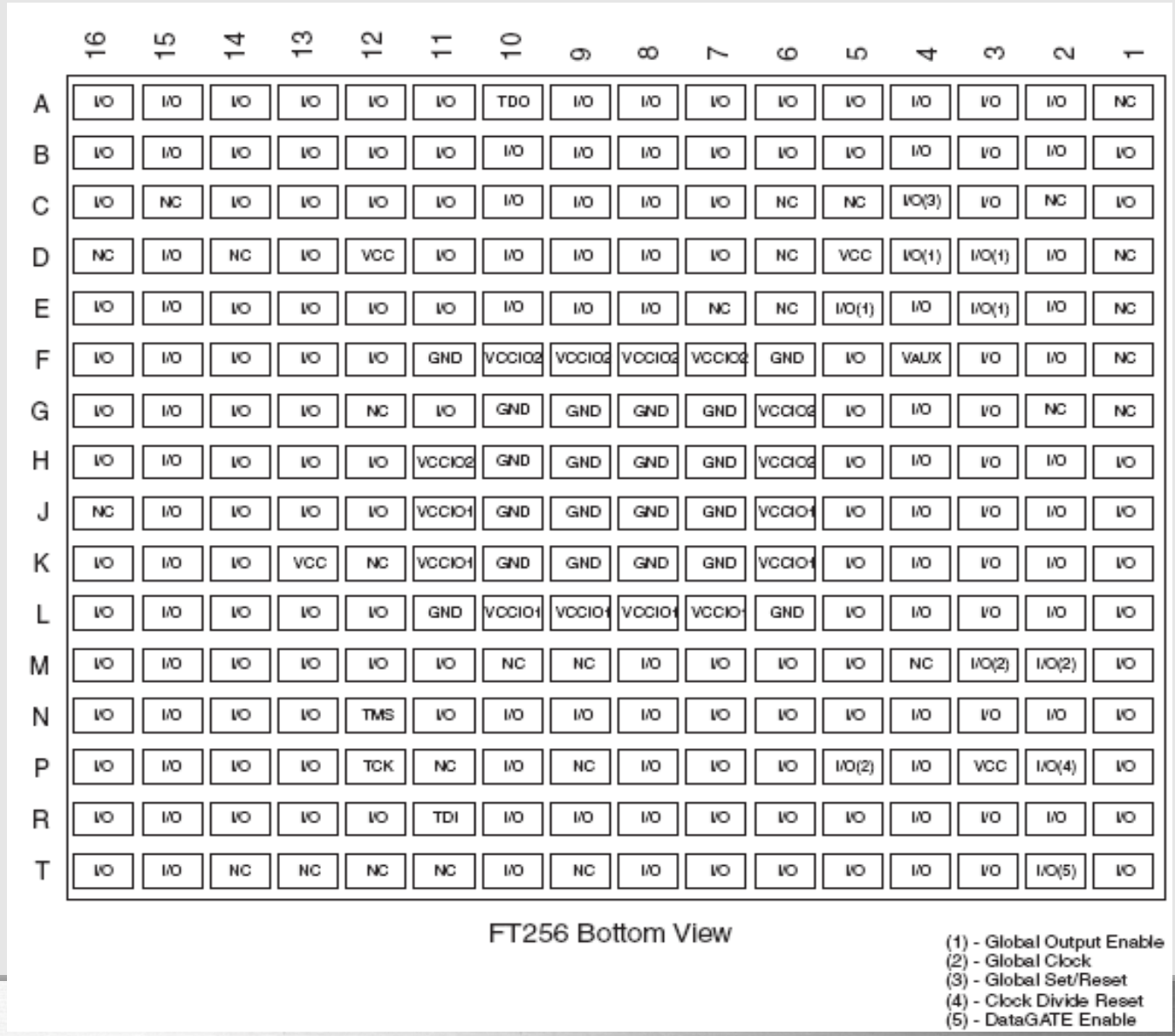
Power saving : XC2C256 CoolRunner-II CPLD

- Xilinx XC2C256 CoolRunner-II CPLD
- Function : Power savings to high-end communication equipment and high speed to battery operated devices. Due to the low power stand-by and dynamic operation, overall system reliability is improved.

  - Optimized for 1.8V systems
    - As fast as 5.7 ns pin-to-pin delays
    - As low as 13 μA quiescent current

  - All pins function can find in data sheet

Power saving : XC2C256 CoolRunner-II CPLD pin configuration [5]



	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
A	I/O	I/O	I/O	I/O	I/O	I/O	TDO	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	NC
B	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O
C	I/O	NC	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	NC	NC	I/O(3)	I/O	NC	I/O
D	NC	I/O	NC	I/O	VCC	I/O	I/O	I/O	I/O	I/O	NC	VCC	I/O(1)	I/O(1)	I/O	NC
E	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	NC	NC	I/O(1)	I/O	I/O(1)	I/O	NC
F	I/O	I/O	I/O	I/O	I/O	GND	VCCIO2	VCCIO2	VCCIO2	VCCIO2	GND	I/O	VAUX	I/O	I/O	NC
G	I/O	I/O	I/O	I/O	NC	I/O	GND	GND	GND	GND	VCCIO2	I/O	I/O	I/O	NC	NC
H	I/O	I/O	I/O	I/O	I/O	VCCIO2	GND	GND	GND	GND	VCCIO2	I/O	I/O	I/O	I/O	I/O
J	NC	I/O	I/O	I/O	I/O	VCCIO1	GND	GND	GND	GND	VCCIO1	I/O	I/O	I/O	I/O	I/O
K	I/O	I/O	I/O	VCC	NC	VCCIO1	GND	GND	GND	GND	VCCIO1	I/O	I/O	I/O	I/O	I/O
L	I/O	I/O	I/O	I/O	I/O	GND	VCCIO1	VCCIO1	VCCIO1	VCCIO1	GND	I/O	I/O	I/O	I/O	I/O
M	I/O	I/O	I/O	I/O	I/O	I/O	NC	NC	I/O	I/O	I/O	I/O	NC	I/O(2)	I/O(2)	I/O
N	I/O	I/O	I/O	I/O	TMS	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O
P	I/O	I/O	I/O	I/O	TCK	NC	I/O	NC	I/O	I/O	I/O	I/O(2)	I/O	VCC	I/O(4)	I/O
R	I/O	I/O	I/O	I/O	I/O	TDI	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O
T	I/O	I/O	NC	NC	NC	NC	I/O	NC	I/O	I/O	I/O	I/O	I/O	I/O	I/O(5)	I/O

FT256 Bottom View

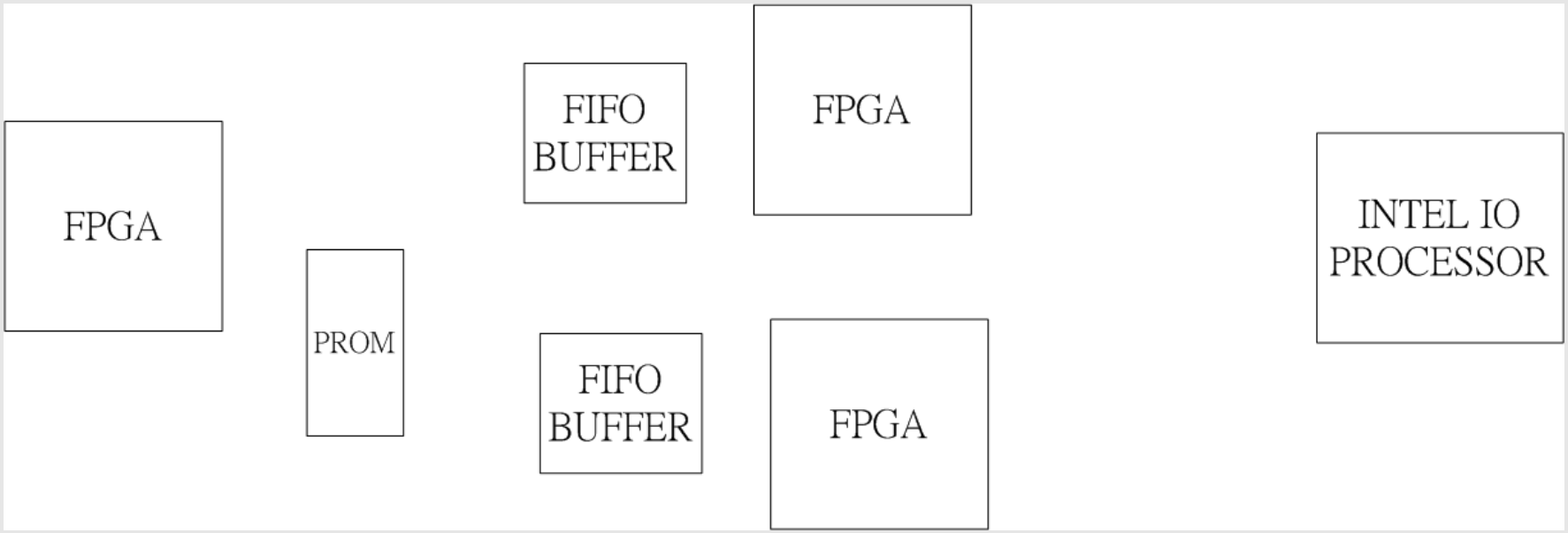
(1) - Global Output Enable
(2) - Global Clock
(3) - Global Set/Reset
(4) - Clock Divide Reset
(5) - DataGATE Enable

Project Procedure (1/4)

- 1. The first step is to identify the ECL interface and PCI-X interface as following,
  - The ECL interface is synchronous or asynchronous
  - The input data bit length , clock bit length, clock rate and data rate.
  - To identify what is the header bits to drive the ECL interface.
  - PCI-X is a typical standard, which can find in [10],[11]. All things above could probably do by logical analyzer and pattern generator.

Project Procedure (2/4)

- 2. Draw the total connection way among key components which we desire to know of the board.
  - By Digital MultiMeter (三用電表) in use of short circuit concept and datasheet
  - As following figure, try to complete the interface connection.



Project Procedure (3/4)

- 3. Confirm which processor is working alone or cooperation and where they get the instruction.
  - INTEL processor : load instruction from external ROM when system starts
  - FPGA processor : load instruction from internal ROM or external ROM when system starts.

  Both are need to be verified by logic analyzer to record the waveform,

- 4. Use software to disable the function block (such as Viterbi, R-S decoder ..) , we can distinguish if any FPGA processor is dedicated to some function block by logic analyzer and identify the location among processor of all function block.

Project Procedure (4/4)

- 5. If we know where the processor get the instruction to work, we can use logic analyzer to record the instruction. Then, by software to disable some function (such as R-S decoder), we hope to extract all instructions by observe the instruction waveform.

- 6. Use pattern generator to copy the observe instruction waveform to verify if the copied instruction can do the same action as disable function.

- Need help:
  - Logic analyzer
  - Pattern generator
  - Discuss when is suitable time for us to proceed measurement of board

A study of Reed-Solomon Code and circuit design : Basic computation

- Galois Field:
  - A Galois field of 2^m elements is defined by an irreducible binary polynomial of degree m
  - Ex: (285)=$2^0+2^2+2^3+2^4+2^8$ =(100011101)
    - $X^8+X^4+X^3+X^2+1$
  - The field element a^i is then defined as the remainder of the division of the i^{th} power of x by the defining polynomial
  - Ex: a^{12} = (11001101)

$$a^{12} \Rightarrow \frac{X^{12}}{X^8+X^4+X^3+X^2+1} = x^4+1+\frac{x^7+x^6+x^3+x^2+1}{x^8+x^4+x^3+x^2+1}$$

  - $a^i * a^j = a^{[(i+j)\bmod(2m-1]}$

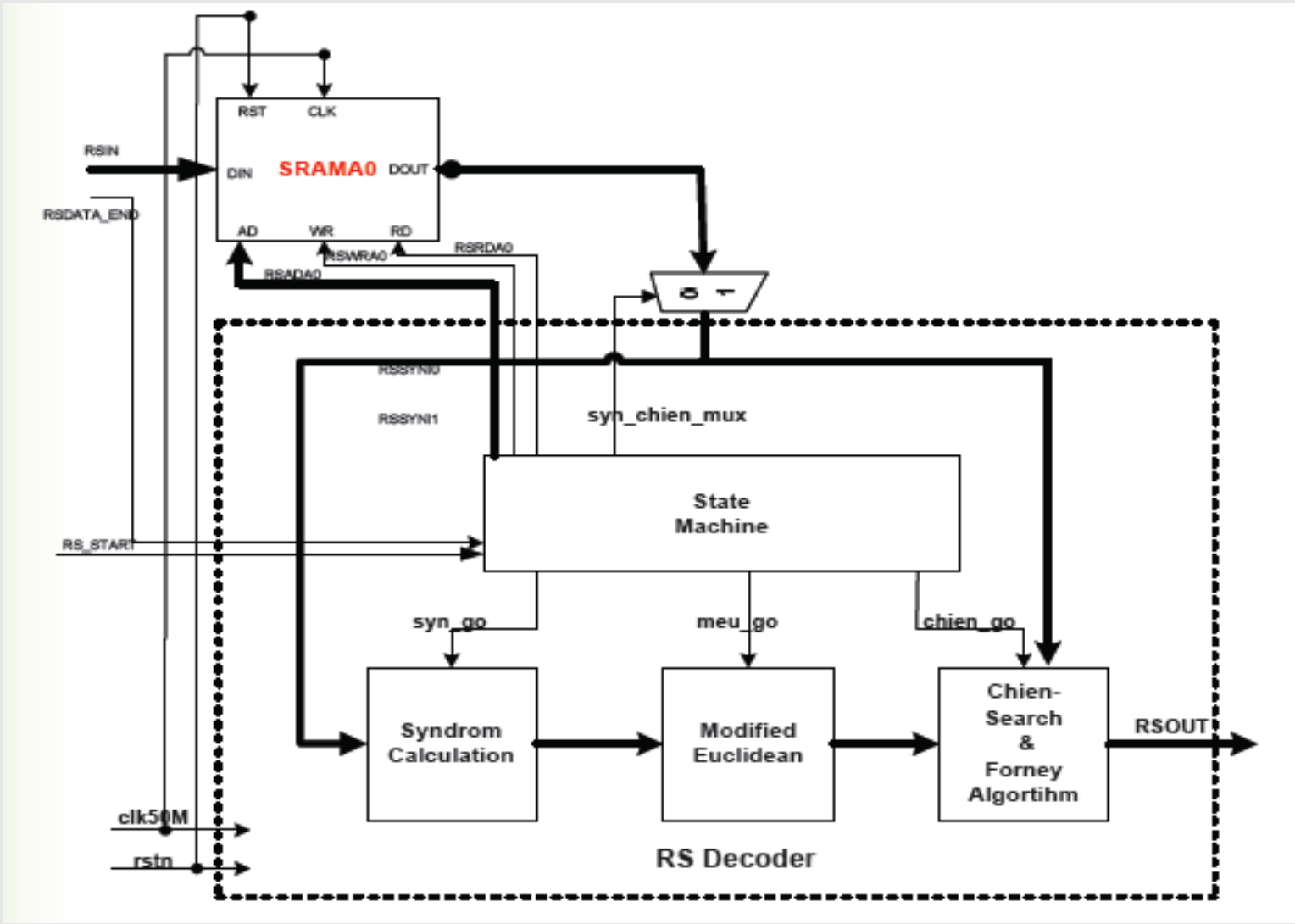
$$a^i / a^j = a^{[(i-j)\bmod(2m-1]}$$

$$a^i + a^j = xor(a^i + a^j)$$

RS decoding approach

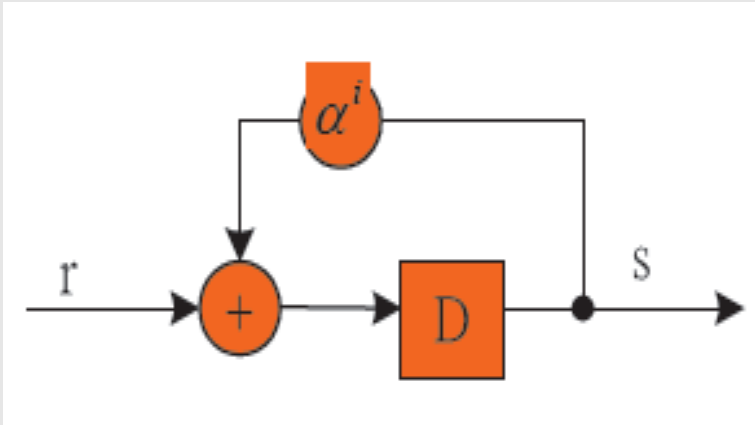
- RS codeword polynomial has consecutive 2t roots $\alpha, \alpha^2, \alpha^3, ..., \alpha^{2t}$
- A) Syndrome computation➔ $S_i = R(\alpha^i)$
- B) Key equation solving by Euclidean algorithm (輾轉相除法)
  ➔Find the error locator polynomial
- C) The roots of error locator polynomial imply error location
- D) The coefficient of error locator polynomial imply error magnitude
- E) Correct the received codeword

RS decoder

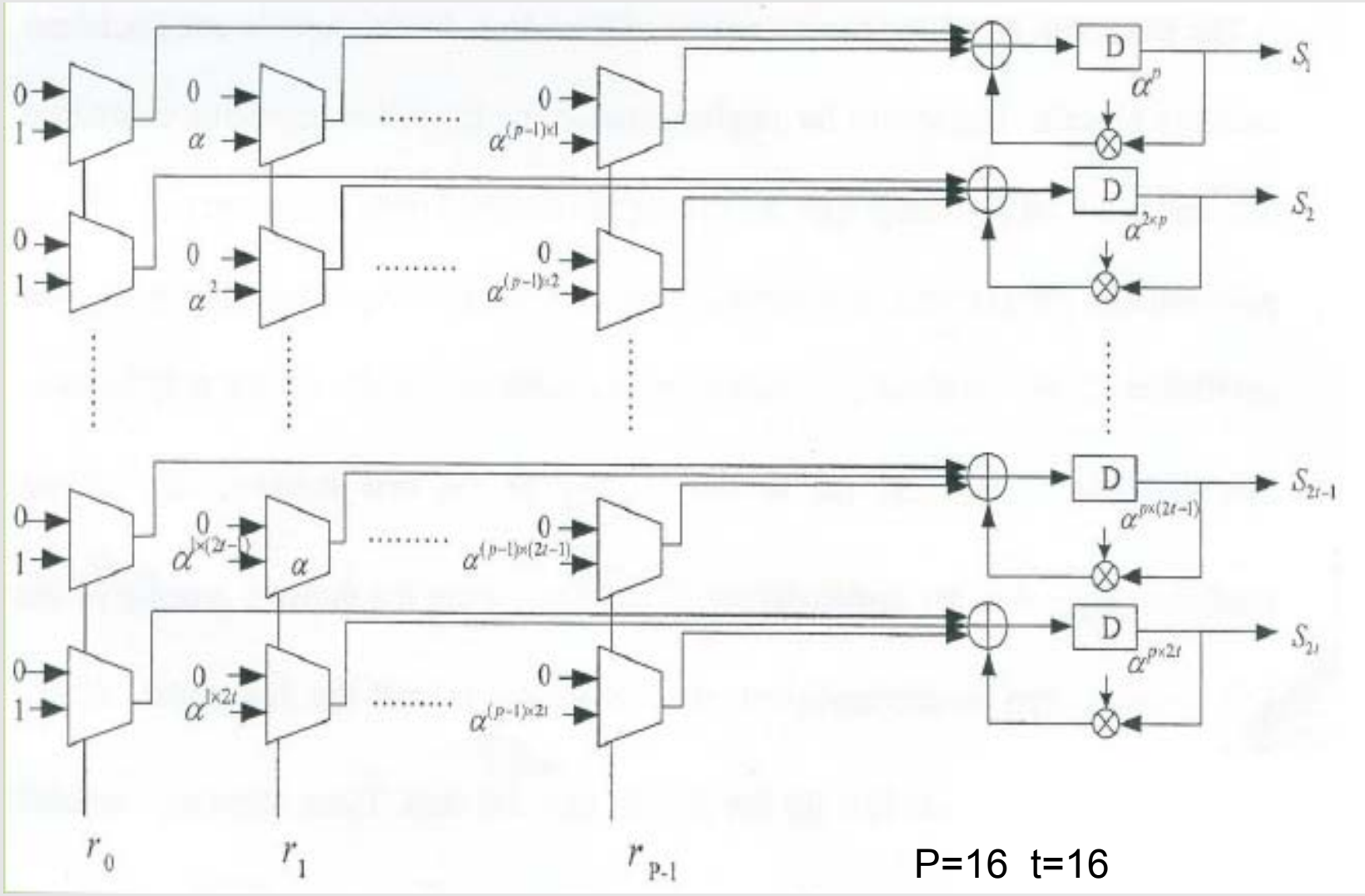

RST
CLK
RSIN
DIN
SRAMA0
DOUT
RSDATA_END
AD
WR
RD
RSWRA0
RSRDA0
RSADA0
0
1
RSSYNI0
RSSYNI1
syn_chien_mux
State Machine
RS_START
syn_go
meu_go
chien_go
Syndrom Calculation
Modified Euclidean
Chien-Search & Forney Algortihm
RSOUT
clk50M
rstn
RS Decoder

Syndrome Computation

- Syndrome Calculator t=16
  - R(X) = C(X) + E(X)
  - $R(\alpha^i) = C(\alpha^i) + E(\alpha^i)$,i=1,2,…….2t
  - $S_i = R(\alpha^i) = 0 + E(\alpha^i) = E(\alpha^i)$,i=1,2,…….2t
  - $S_i = r_{219}\alpha^{219i} + r_{218}\alpha^{218i} + ... + r_1\alpha^i + r_0$
  - $S_i = (r_{219}\alpha^{219i} + r_{218})\alpha^i + ..)\alpha^i + ... + r_1)\alpha^i + r_0$



Parallel syndrome generator



Key equation

- Key equation
  - $S_i = R(\alpha^i) = 0 + E(\alpha^i) = E(\alpha^i)$,i = 1,2,…….2t
  - $S_i = e_{j1}\alpha^{ij1} + e_{j2}\alpha^{ij2} + ... + e_{jv}\alpha^{ijv}$,jv = 1,2,……t
  - $S_i = Y_1 X_1^i + Y_2 X_2^i + ... + Y_v X_v^i$,v = 1,2,……t
  - Define $\Lambda(x)$ is the error locator polynomial

$$\Lambda(x) = (1+X_1x)(1+X_2x)(1+X_3x).....(1+X_vx)$$
$$= 1+\Lambda_1x+\Lambda_2x^2+\Lambda_3x^3+....+\Lambda_vx^v$$

$$0 = Y_l X_l^{j+v}\Lambda(X_l^{-1}) = S_{j+v} + \Lambda_1 S_{j+v-1} + ... + \Lambda_v S_j$$

- Euclid's Algorithm: fast to get value
  - Defined Error magnitude polynomial $\Omega(x)$

$$s(x)\Lambda(x) \bmod (x^{2t}) = \Omega(x)$$

$$\Theta(x)x^{2t} + \Omega(x) = S(x)\Lambda(x)$$

$$\Theta(x)x^{2t} + S(x)\Lambda(x) = \Omega(x)$$

  - Find GCD(x^{2t}, S(x)) 最大公因數

GCD transformation

- GCD-preserving transformation

$$A = a_i x^i + \ldots + a_1 x^1 + a_0$$

$$B = b_j x^j + \ldots + b_1 x^1 + b_0$$

- GCD(A,B)=GCD(A',B') , d=i-j , q= b_i / a_i

$$A' = A - q x^d B$$

$$B' = B$$

- Or GCD(A,B)=GCD(A',B'), d=j-I , q= b_i / a_i

$$A' = A$$

$$B' = B - q x^d A$$

Simple GCD example

- Ex: find GCD(256,108)
  - Find GCD(256,108) is the same as finding GCD(40,108)
  - 40 = 256 - 2*108
  - Find GCD(40,108) is the same as finding GCD(40,28)
  - 28 = 108 - 2*40
  - Find GCD(12,28) is the same as finding GCD(12,4)
  - 4 = 28 - 2*12
  - GCD(12,4) = 4 → GCD(256,108)=4

Modified Euclidean algorithm

- Avoid divider by Math skill
- GCD(x^{2t} , S(x))

$$\Theta(x)x^{2t} + S(x)\Lambda(x) = \Omega(x)$$

- Using ME : In last iteration , we can get $\Lambda(x) = L_i(x)\Omega(x) = R_i(x)$

$$R_0(x) = x^{2t}, Q_0(x) = S(x) = S_1 + S_2 x + \ldots + S_{2t} x^{2t}, L_0(x) = 0, U_0(x) = 1$$

$$R_i(x) = [\sigma_{i-1} b_{i-1} R_{i-1}(x) + \bar{\sigma}_{i-1} a_{i-1} Q_{i-1}(x)] + x^{|l_i - 1|}[\sigma_{i-1} a_{i-1} Q(x) + \bar{\sigma}_{i-1} b_{i-1} R_{i-1}(x)]$$

$$Q_i(x) = \sigma_{i-1} Q_{i-1}(x) + \bar{\sigma}_{i-1} R_{i-1}(x)$$

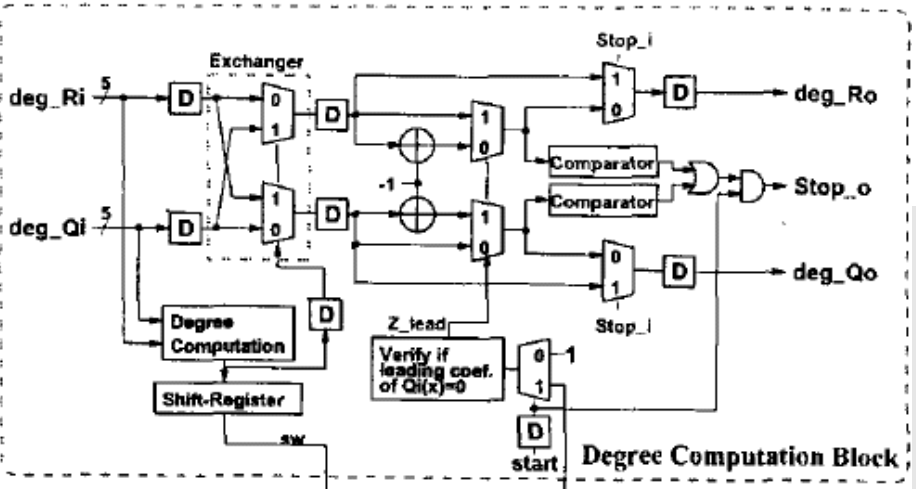
$$L_i(x) = [\sigma_{i-1} b_{i-1} L_{i-1}(x) + \bar{\sigma}_{i-1} a_{i-1} U_{i-1}(x)] + x^{|l_i - 1|}[\sigma_{i-1} a_{i-1} Q(x) + \bar{\sigma}_{i-1} b_{i-1} L_{i-1}(x)]$$

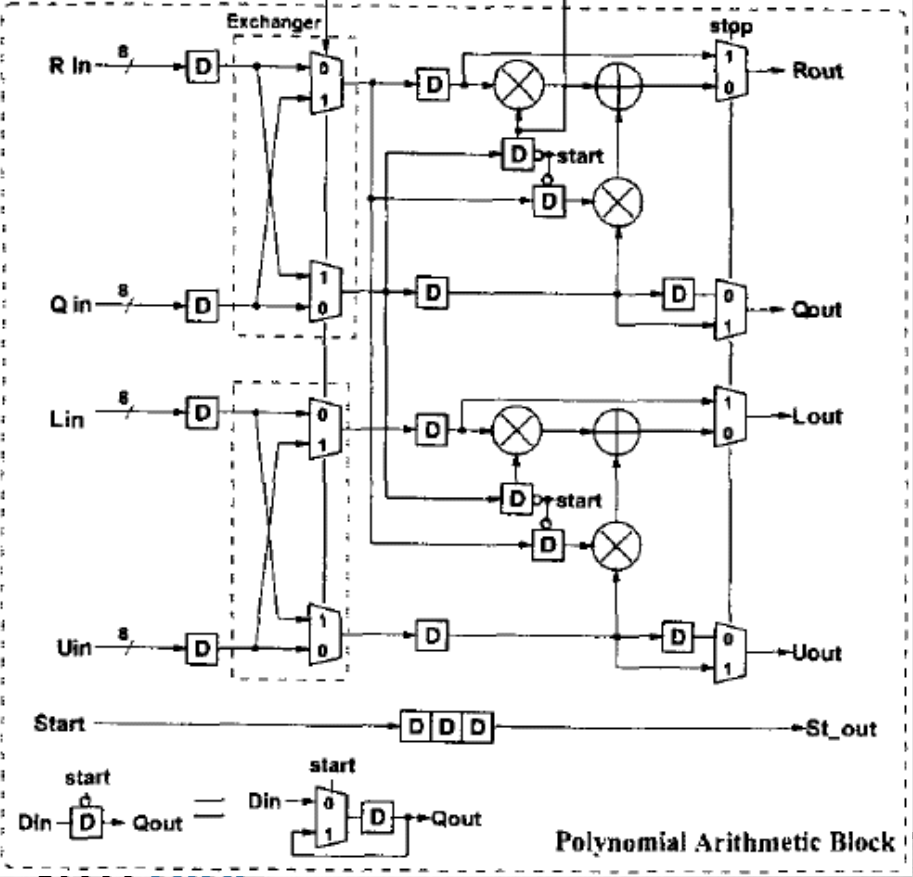
$$U_i(x) = \sigma_{i-1} U_{i-1}(x) + \bar{\sigma}_{i-1} L_{i-1}(x)$$

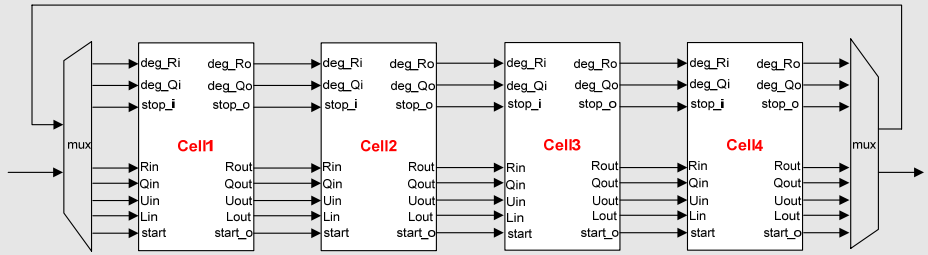
$$l_{i-1} = \deg(R_{i-1}(x)) - \deg(Q_{i-1}(x))$$

$$\sigma_{i-1} = 1, l_{i-1} \geq 0$$

$$\sigma_{i-1} = 0, l_{i-1} < 0$$


Exchanger
deg_Ri
deg_Qi
Stop_i
deg_Ro
Comparator
Comparator
Stop_o
deg_Qo
Stop_i
Degree Computation
Shift-Register
sw
Z_lead
Verify if leading coef. of Qi(x)=0
start
Degree Computation Block

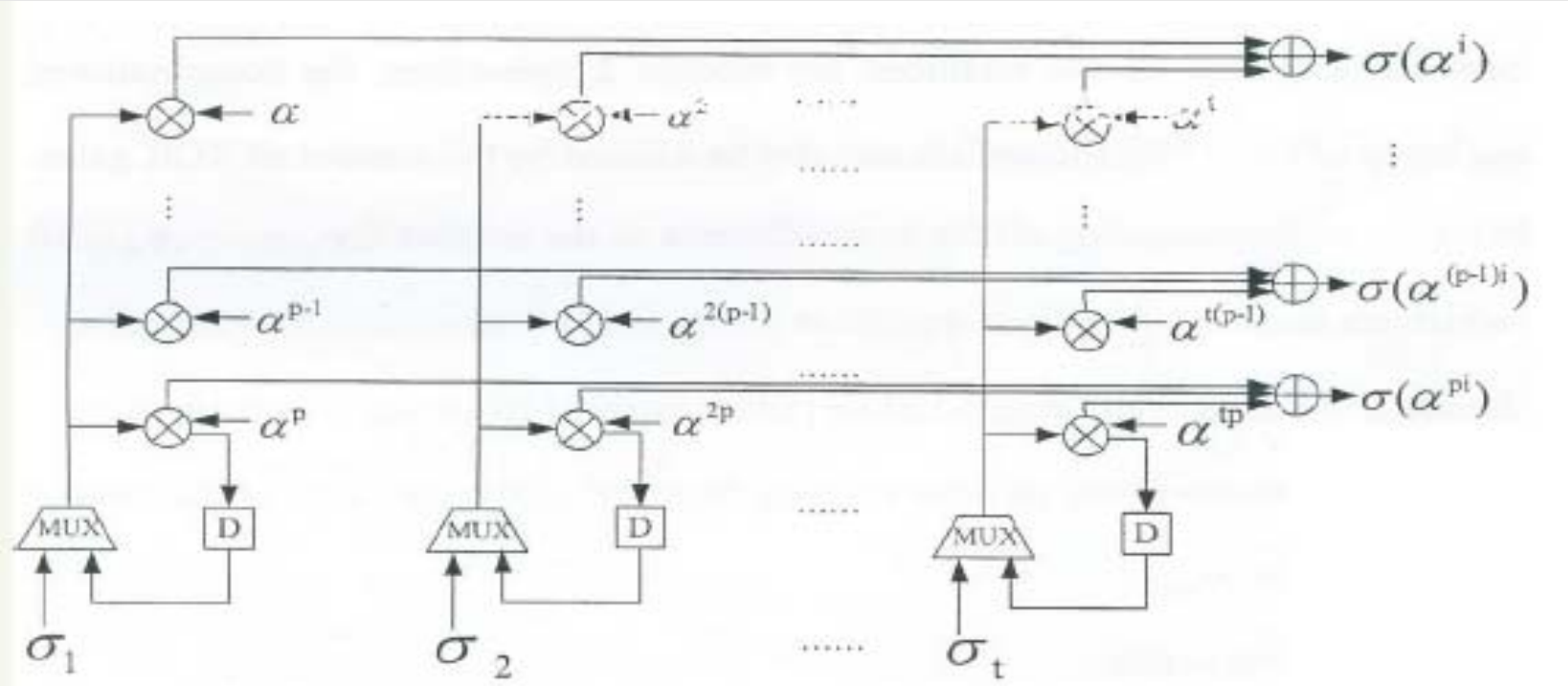

Exchanger
R In
Q in
Lin
Uin
stop
Rout
Qout
Lout
Uout
start
Start
St_out
Din
Qout
Polynomial Arithmetic Block


mux
Cell1
Cell2
Cell3
Cell4
deg_Ri
deg_Qi
stop_i
deg_Ro
deg_Qo
stop_o
Rin
Qin
Uin
Lin
start
Rout
Qout
Uout
Lout
start_o

Chien search

- Determine the roots (error location) of an error locator polynomial

$$\sigma(x) = \sigma_t x^t + \sigma_{t-1} x^{t-1} + \ldots + \sigma_0$$

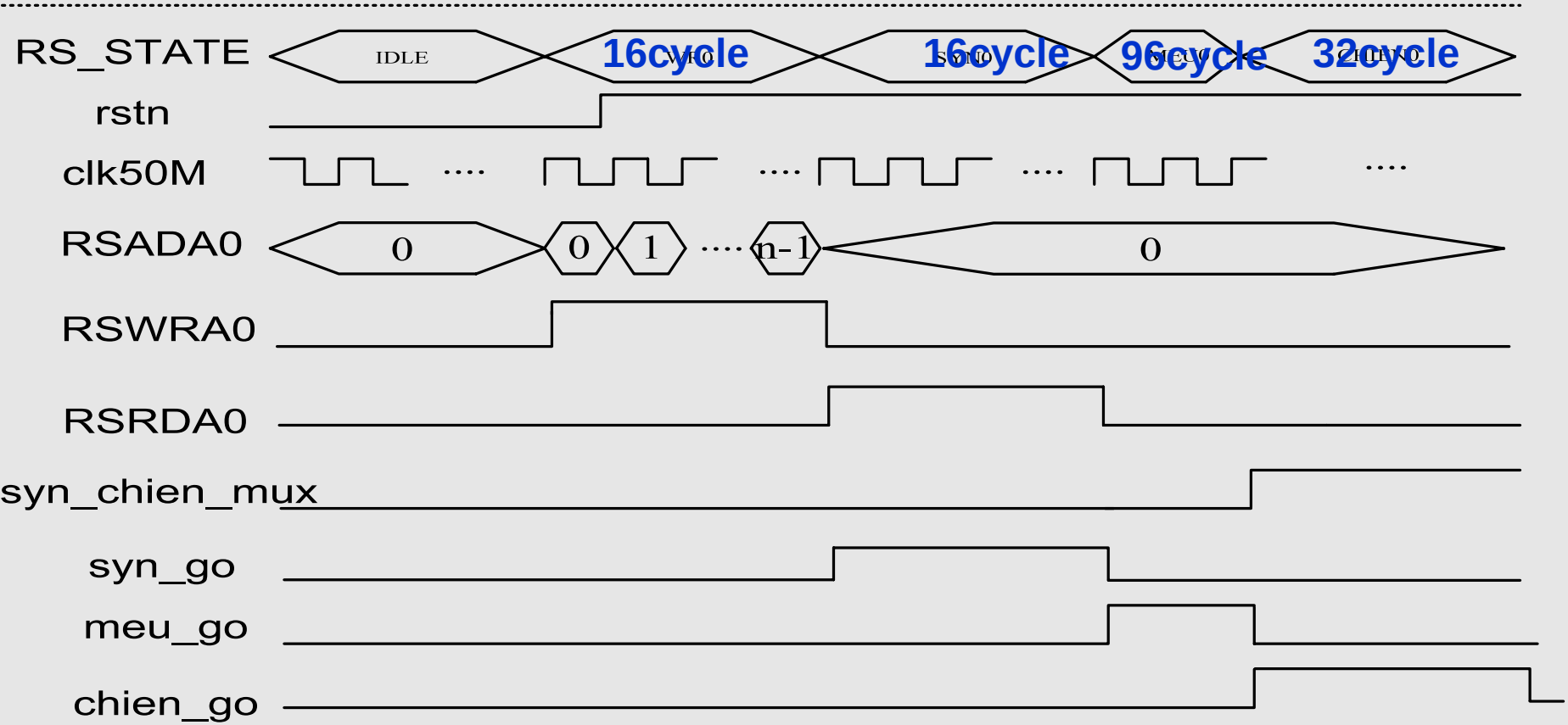


- Paraller Chien search with parallel factor p=16

- Forney Algorithm (Error value)

$$Y_j = \frac{\Omega(X_j^{-1})}{\Lambda'(X_j^{-1})}$$

- J=1,2,3……,n-1, it represents code to correct form $r^{n-1}, r^{n-2}, \ldots, r^1, r^0$



223*8bit/160cycle*50*10^8=557.5Mbps@50MHz.

IO pin of R-S decoder

.Signal	I / O	Description
CLK50Mhz	I	Input clock 50 MHz from Clock Gen
rstn	I	Reset (*active low*) for clock generation
RS_START	I	RS decoder starting
RSDATA_END	I	No data to RS decoder, So, RS decoder ending
RSSYNI0[N-1:0]	I	Input date from SRAMA0
RSADA0[M-1:0]	O	Address for SRAMA0
RSWRA0	O	Write enable for SRAMA0
RSRDA0	O	Read enable for SRAMA0
ERRADD1[7:0]	O	Error address for 1st Error
ERRADD2[7:0]	O	Error address for 2st Error
ERRADD3[7:0]	O	Error address for 3st Error
ERRADD4[7:0]	O	Error address for 4st Error
ERRADD5[7:0]	O	Error address for 5st Error
ERRADD6[7:0]	O	Error address for 6st Error
etc.		
ERRADD14[7:0]	O	Error address for 14st Error
ERRADD15[7:0]	O	Error address for 15st Error
ERRADD16[7:0]	O	Error address for 16st Error
SERR[3:0]	O	Corrected error amount

Implementation board

- Altera: Trex-S2-180-3

Memory spec

Table 2–1. Summary of TriMatrix Memory Features

Feature	M512 Blocks	M4K Blocks	M-RAM Blocks
Maximum performance	500 MHz	550 MHz	420 MHz
Total RAM bits (including parity bits)	576	4,608	589,824
Configurations	512 × 1 256 × 2 128 × 4 64 × 8 64 × 9 32 × 16 32 × 18	4K × 1 2K × 2 1K × 4 512 × 8 512 × 9 256 × 16 256 × 18 128 × 32 128 × 36	64K × 8 64K × 9 32K × 16 32K × 18 16K × 32 8K × 64 8K × 72 4K × 128 4K × 144
Parity bits	✓	✓	✓
Byte enable	✓	✓	✓
Pack mode		✓	✓
Address clock enable		✓	✓
Single-port memory	✓	✓	✓
Simple dual-port memory	✓	✓	✓
True dual-port memory		✓	✓
Embedded shift register	✓	✓	
ROM	✓	✓	
FIFO buffer	✓	✓	✓
Simple dual-port mixed width support	✓	✓	✓
True dual-port mixed width support		✓	✓

Deliverables

- From the deliverable of our proposal during this project, we will not complete the whole verification for replacing IO21000 board, but we will deliver:

- (i)Frame synchronization
- (ii)PRN descrambler
- (iii)RS decoder

where (i) and (ii) will be verified by software simulation (providing MATLAB code), and (iii) will be verified by hardware (FPGA).

Reference

- [1] Kongsberg Spacetec AS, online source, http://www.spacetec.no/
- [2] National Semiconductor, “LVDS Owner’s Manual Including High-Speed CML and Signal Conditioning, 4th edition, 2008
- [3] Intel, “FW80321 IO Processor datasheet”, January, 2005
- [4] Xilinx, ”XCF32P VOG48 datasheet”, March 14, 2005
- [5] Xilinx XC2C256 CoolRunner-II CPLD datasheet, March 8, 2007
- [6] IDT,”72V36104 datasheet”, March, 2001
- [7] IDT “72T18125 datasheet”, Feburary, 2009
- [8] Xilinx, “Virtex-4 FPGA packaging and pin out specification datasheet”, September, 2008
- [9] Xilinx, “Initiator/Target v6.5 for PCI-X User Guide”, 2007.
- [10] Xilinx, “Virtex-4 ML455 PCI/PCI-X Development Kit *User Guide*”, 2005.