

PaxDisk FroggyMatrix 프로세서

2013년 10월

주식회사 팩스디스크

PaxDisk 사업분야

- ☑ 팩스디스크는 산업계에서 필요로 하는 요소기술을 바탕으로 쓸모있는 기술을 제공합니다.
- ☑ 해당 기술은 IP 및 소프트웨어, 반도체, 제품, 서비스의 형태로 고객에게 제공됩니다.

Military



AirBorne Level SSD Technology →

- ☑ 군에서 필요로 하는 특수 저장장치를 연구 개발 생산합니다.

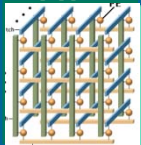
Data Centre



Peta Scale RAID Technology →

- ☑ 자체 반도체 설계 기술에 의해 단일 컨트롤러로 세계 최대용량의 기록을 보유하고 있습니다.

FroggyMatrix



FroggyMatrix Parallel Technology →

- ☑ 팩스디스크는 단일칩에서 256개의 프로세서가 연동되는 초소형 고속 멀티 프로세서 설계 기술을 보유하고 있습니다.

Automotive



WCDMA BlackBox Technology →

- ☑ 동영상기록용의 FAT32 상에서의 WearLeveling 기술로 SD/eMMC의 수명을 2배이상 연장합니다. WCDMA 관제형 블랙박스 제품또한 제공합니다.

보유 특허 및 인증 현황

☑ 팩스디스크는 SSD, BlackBox 관련 특허 21종을 보유하고 있습니다.

☑ ISO9001:2009 를 최근 취득하여 제품 및 서비스에 대한 품질 관리에 힘쓰고 있습니다.

보유 지적소유권 현황

등록특허 : 21건

- ☑ [특허등록] 에뮬레이터를 갖는 휴대용 반도체 메모리 저장장치
- ☑ [특허등록] 동적매핑테이블을 이용한 플래시메모리 관리방법
- ☑ [특허등록] 시스템에 장착된 낸드 플래시 소자의 수명 측정 방법*
- ☑ [특허등록] 다중에뮬레이션 기능을 갖는 SSD
- ☑ [특허등록] 다중스트로브를 이용한 플래시메모리시스템 및 제어방법
- ☑ [특허등록] 낸드플래시메모리의 데이터저장방법
- ☑ [특허등록] 다수의드라이브가 구현될 수 있는 SSD
- ☑ [특허등록] SSD의 캐쉬메모리 및 이를 이용한 캐시관리 시스템
- ☑ [특허등록] 보안기능을 갖는 SSD
- ☑ [특허등록] 애플램을 캐시메모리로 사용하는 플래시메모리장치
- ☑ [특허등록] 플래시메모리 저장장치 및 그에 따른 관리방법
- ☑ [특허등록] 레이드 기능을 지원하는 SSD
- ☑ [특허등록] 플래시 메모리 관리 방법
- ☑ [특허등록] 다중카메라를 이용하는 차량용 블랙박스장치
- ☑ [특허등록] 암호화 데이터 저장장치 및 그 저장방법
- ☑ [특허등록] 고속동작하는 반도체 스토리지 시스템 및 그 제어방법*
- ☑ [특허등록] 듀얼메모리 영역을 관리하는 반도체 스토리지 시스템의 제어방법*
- ☑ [특허등록] 데이터머지를 수행하는 반도체 스토리지시스템 및 그 제어방법*
- ☑ [특허등록] 반도체 스토리지 시스템 및 그 제어방법*
- ☑ [특허등록] 다채널 캐쉬를 이용한 SSD 및 캐시 데이터 저장방법
- ☑ [특허등록] 로고스키 코일을 이용한 전류센서

KS Q ISO 9001:2009 인증 취득

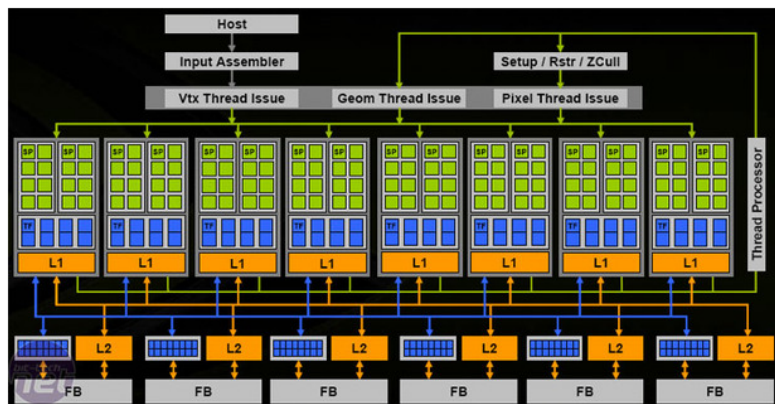


GPU 와 Multiprocessor

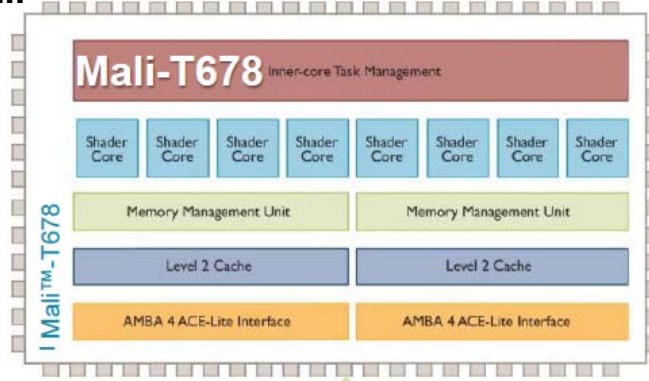
- ☑ inmos 는 1984년 세계 최초의 병렬처리 Transputer 를 발표하였다.
- ☑ nVIDIA 의 GPU, ARM 의 GPU 는 성공적으로 사용되고 있으나, 알고리즘 포팅이 용이하지 않다.

nVIDIA 나 ARM 과 같은 반도체 대기업에서 GPU 사업은 성공하고 있다.

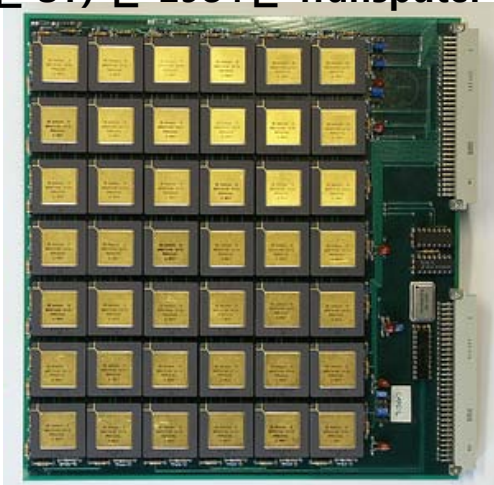
nVIDIA GPU



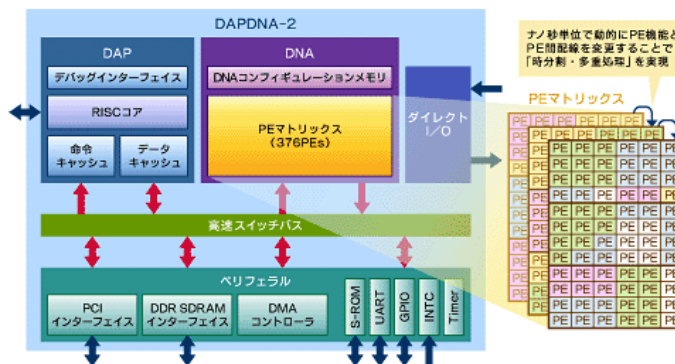
ARM-Mali



inmos (현 ST) 는 1984년 Transputer 와 Occam2 를 발표

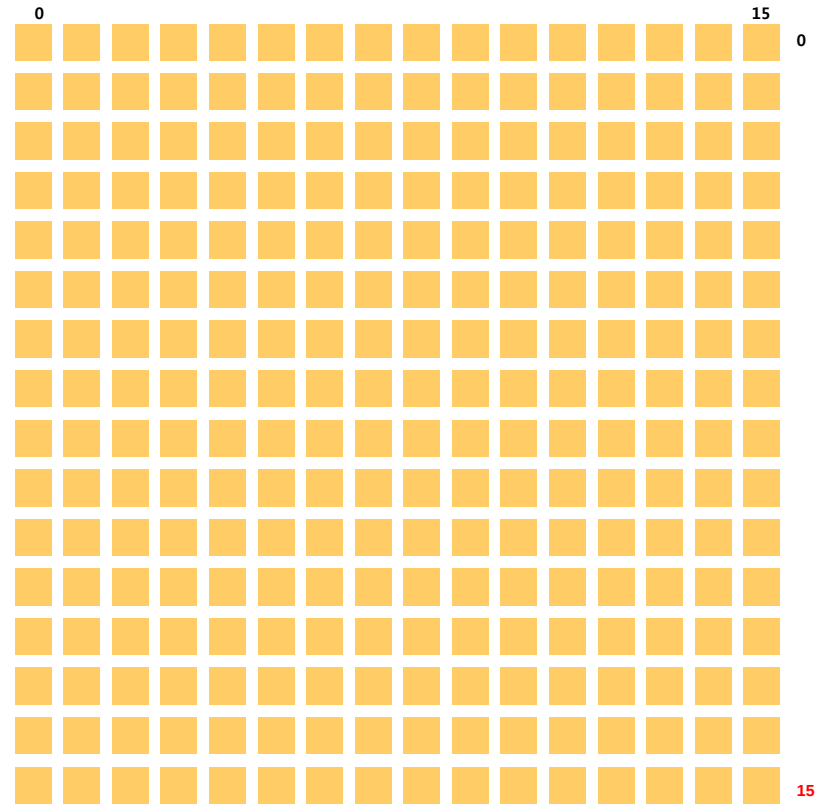


일본의 IP-Flex 는 CPU 구조의 실패가 파산의 큰 영향



- ☑ FroggyMatrix 는 팩스디스크에서 독자 개발한 micro 멀티프로세서 구조 이다.
- ☑ GPGPU 등과는 비교할 때 대단히 작은 프로세서 어레이로 구성되며 프로그래밍이 상대적으로 용이하다.

FroggyMatrix 프로세서



논두렁에 가로 16, 세로 16의 총 256마리의 개구리들이 모여 있습니다.
이 개구리들은 갓태어났지만 매우 빠르며 순발력이 뛰어난 개구리들
입니다. 이 개구리들은 가장 짧은 경로를 찾아서 서로 대화 할 수도
있습니다. 대화에 참여하지 않거나 일하지 않는 개구리들은 동면합니다.
이 개구리들에게 일을 시켜 보세요.
이 개구리는 한마리당 최대 1GHz(65nm)의 속도로 명령을 수행하기
때문에 동시에 초당 2천5백6십억 명령의 일을 합니다.
논두렁 하나당 256마리의 개구리들이 있고 논두렁 단위로 관리 됩니다.
다른 동물들과 달리 개구리는 비록 단순하고 작지만, 적게 먹고 빠릅니다.

- Opcode Type**

	7	6	5	4	3	2	1	0	Byte 1								Byte 2															
Type I	OPCODE								REG																							
Type II	OPCODE								REG								DATA8															
Type III	OPCODE								REG								DATA16L								DATA16H							



-
- The diagram illustrates the Froggy Core system architecture. At the center is the **Froggy Core**, a blue square. To its left are two orange rounded rectangles: **Instruction Memory** (top) and **Data Memory** (bottom), connected to the core by two horizontal blue lines. To the right of the core is a light green rounded rectangle containing four green rounded rectangles stacked vertically: **Timer**, **UART**, **SPI**, and **I2C**. A horizontal blue line connects the core to this peripheral block, with the label **pAPB** in red text above it. Below the core is a horizontal grey bar labeled **COMM** in red text. Four vertical blue lines connect the core to four green rounded rectangles below the bar, labeled **N**, **S**, **E**, and **W** from left to right.

8단 파이프라인 구조

-

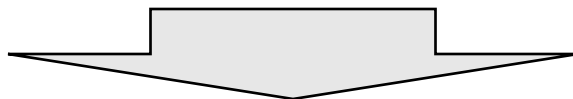
[illegible]

처음 구현한 FroggyMatrix (2x4) 구조

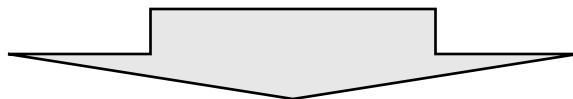
터미널 출력 테스트

- ☑ FroggyMatrix 4x4 부터는 현실적으로 HDL Generator 와 같은 TOOL 소프트웨어 없이는 곤란
- ☑ C 언어로 FroggyMatrix Generator 를 개발함 (P&R 을 대비한 Repeater 자동삽입)

FroggyMatrix (2x4)는 직접 HDL 타이핑하여 입력이
가능하지만 FroggyMatrix (4x4) 이상은 손으로 입력
하기에는 곤란함



C 언어로 FroggyMatrix Generator 프로그램을 개발하여
FroggyMatrix (16x16) 까지 대응되도록 개발함



P&R 을 대비한 Repeater 자동 삽입 기능을 추가함.
추후 GUI 를 도입하여 시각적 효과를 극대화할 필요 있음

FroggyMatrix Generator 의 출력파일 일부

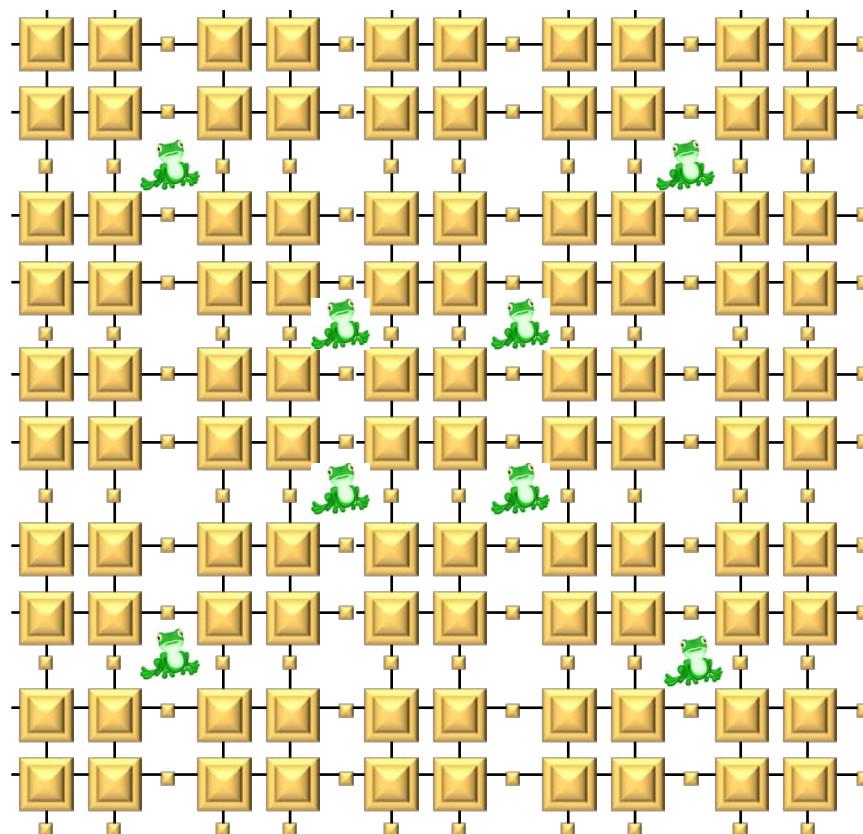
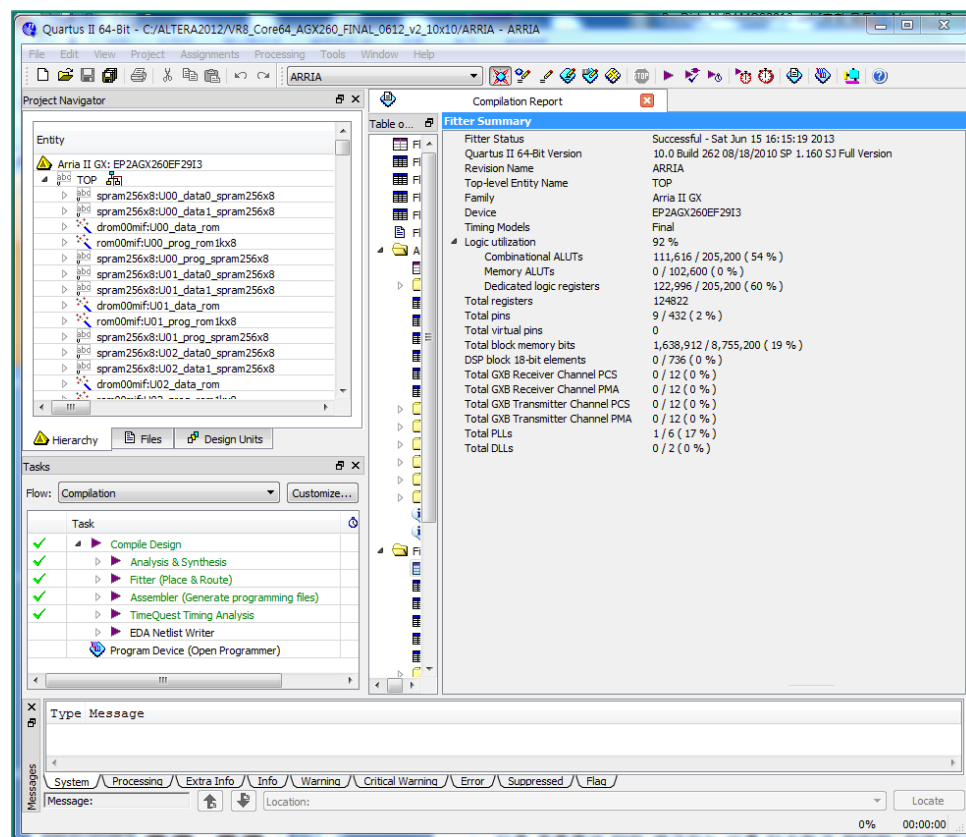
```
1 // FroggyMatrix
2 // Copyright(c) 2013 by PaxDisk
3
4 module FMextx_x_top # (
5     parameter      AW = 14,          //
6     parameter      DW = 8,           //
7     parameter      PN = 100)         //
8 (
9     input  wire      resethn,
10    input  wire      clk,
11    input  wire      clk_apb_slow,
12
13    output wire      prog00_rom1kx8_clk,
14    output wire      prog00_rom1kx8_cen,
15    output wire [ 9:0] prog00_rom1kx8_addr,
16    input  wire [DW - 1:0] prog00_rom1kx8_rd,
17
18    output wire      prog00_spram256x8_clk,
19    output wire      prog00_spram256x8_cen,
20    output wire      prog00_spram256x8_wen,
21    output wire [ 7:0] prog00_spram256x8_addr,
22    output wire [DW - 1:0] prog00_spram256x8_wd,
23    input  wire [DW - 1:0] prog00_spram256x8_rd,
24
25    output wire      data00_rom_clk,
26    output wire      data00_rom_cen,
27    output wire [ 7:0] data00_rom_addr,
28    input  wire [DW - 1:0] data00_rom_rd,
29
30    output wire      data00_ram_clk,
31    output wire      data00_ram_cen0,
32    output wire      data00_ram_cen1,
33    output wire      data00_ram_wen,
34    output wire [ 9:0] data00_ram_addr,
35    output wire [DW - 1:0] data00_ram_wd,
36    input  wire [DW - 1:0] data00_ram_rd,
```

FroggyMatrix(10x10) FPGA 합성 결과

Altera Arria II GX 260 에서 FroggyMatrix 에서 100 개의 Froggy Processor 를 연동시킴.

총 92% 의 Logic 사용, Froggy Clock : 모두 166MHz 로 동작 검증

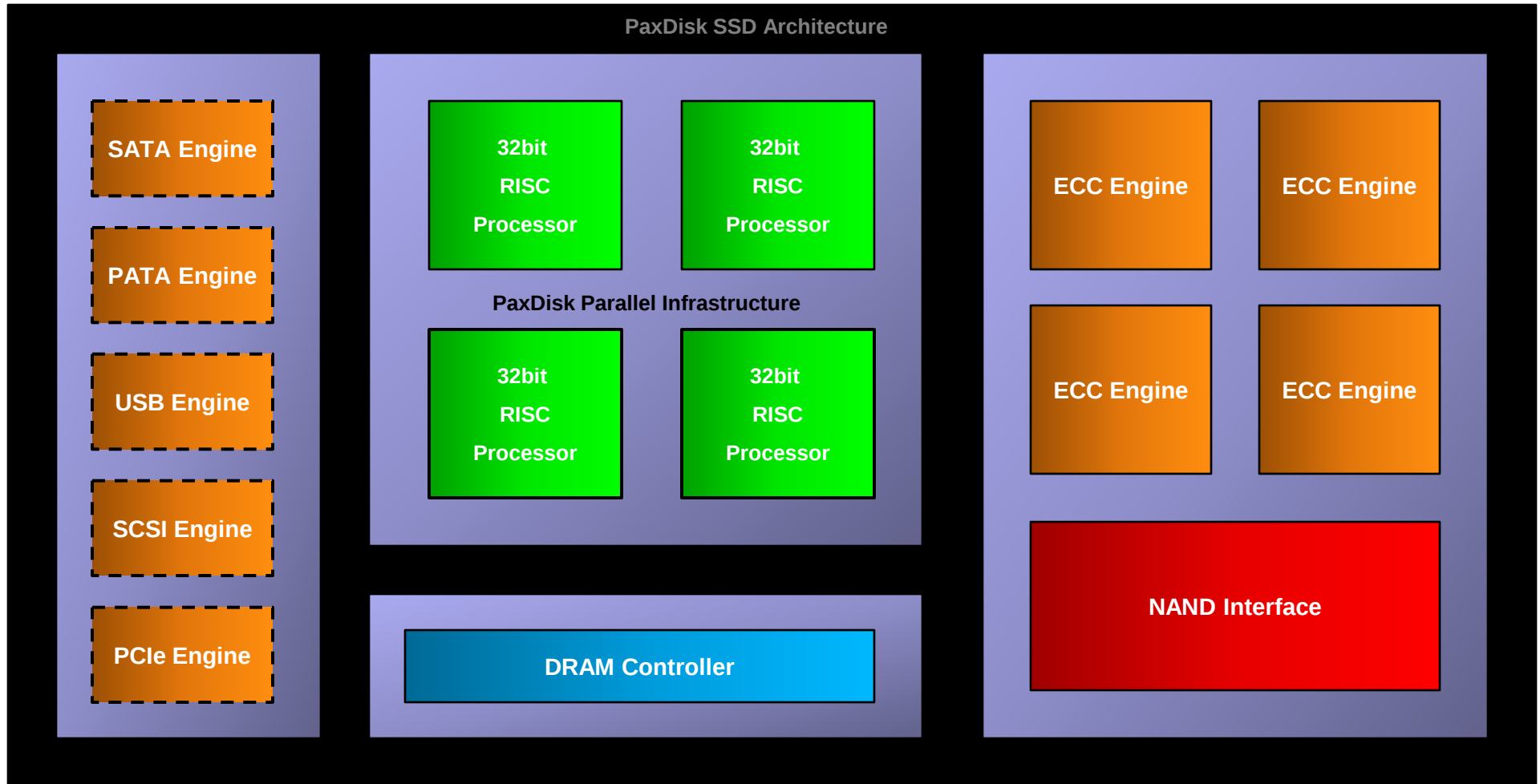
1개의 Froggy Processor 에서의 RAM 용량을 최소화(4KB)하여도 100개의 FroggyMatrix 에서는 400KB 가 됨



10x10 Torus 구조

SSD 용 Multi-Processor

SSD 에서 멀티 프로세서의 요구가 있어 FroggyMatrix 구조를 응용해 Core-A 에 적용해봄
Core-A 끼리는 통신 채널을 이용하기도 하고 DRAM Controller 는 공유 메모리로 적용



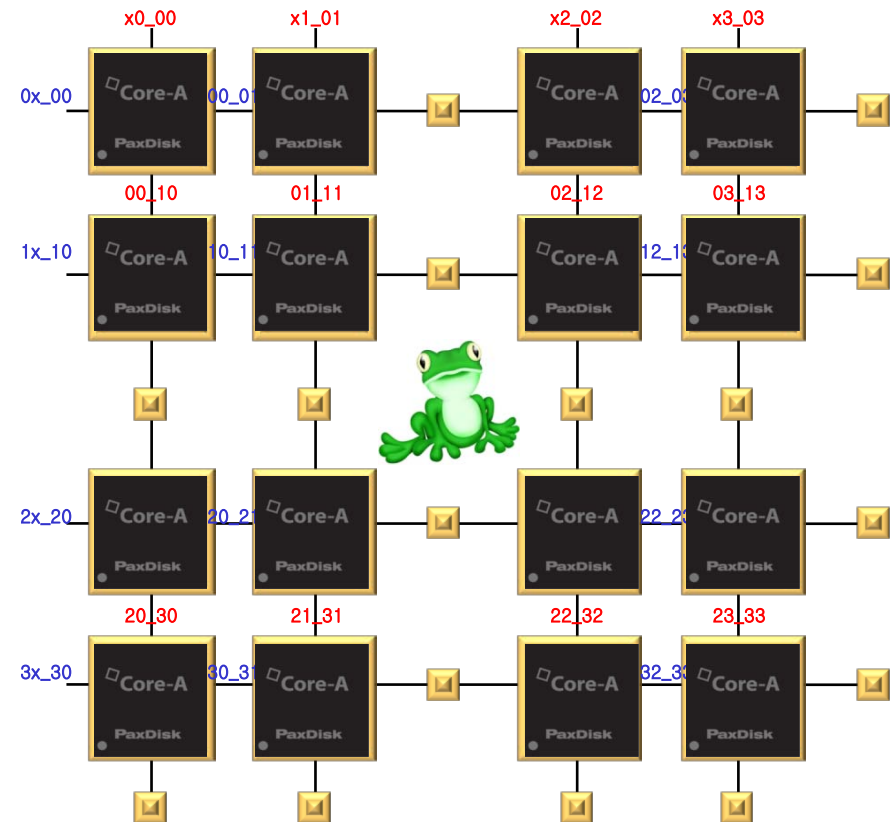
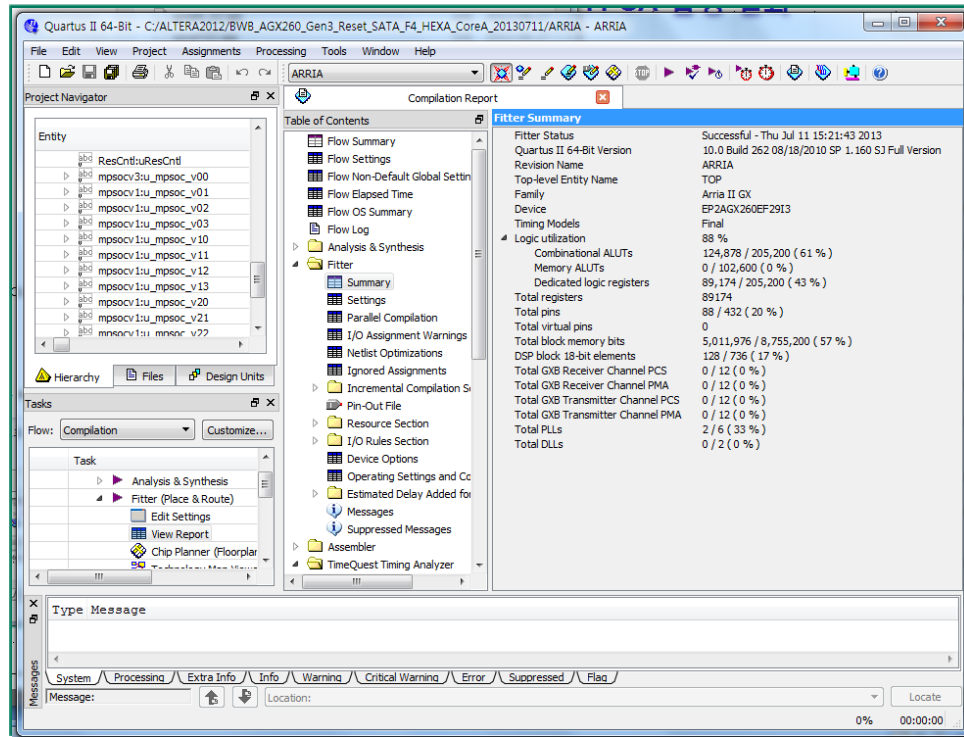
Core-A Matrix (4x4) FPGA 합성 결과

Altera Arria II GX 260 에서 FroggyMatrix 에서 16 개의 Core-A Processor 를 연동시킴.

총 88% 의 Logic 사용, Core-A Clock : 모두 75MHz 로 동작 검증 (ZeroWait TCM)

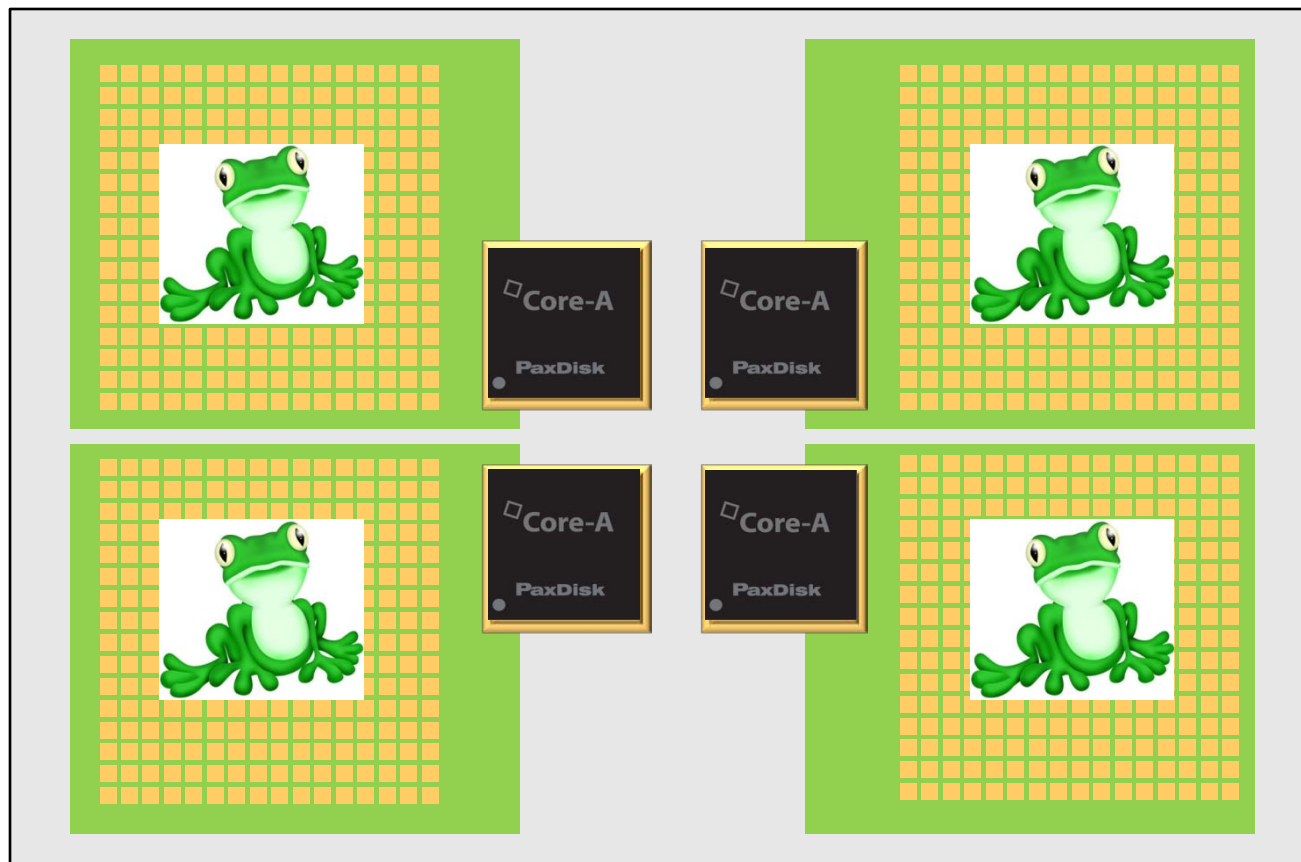
개발 Core-A 는 모두 DRAM 에 접근할 수 있으나 HDL Generator 를 아직 프로그램 작성하지 않았기 때문에

1개의 Core-A 만 DRAM 에 접근하도록 함



4x4 Torus 구조

Core-A Matrix 와 FroggyMatrix 를 묶어 프로세서의 역할을 분담시키는 구조가 가능함
Core-A 는 OS 및 UI 중심처리, FroggyMatrix 는 전용 H/W 를 추가한 고속 연산이 가능함
추후 SSD 에서의 LDPC 알고리즘 처리 및 3D-NAND 를 위한 Low Level 알고리즘등에 대응이 가능할 것으로 사료됨



Softwired BCH ECC
Softwired LDPC
Softwired Compression

감사합니다.

