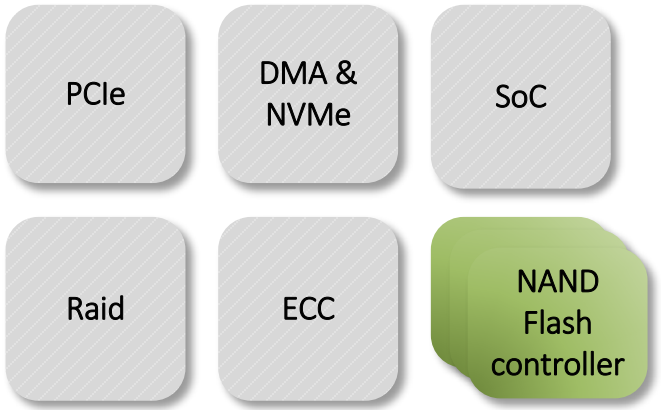
An Instruction-Based NAND Flash Processor Unit (NPU)

Yilei WANG,

Architect, Memblaze

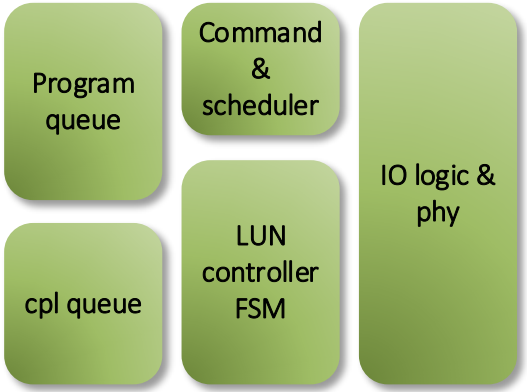
Controller capability review

- Flash Protocol standard
  - Unified ONFI 4.0, toggle 3.0
  - Vendor specified protocols
- Geometry organization
  - Block/page geometry
- Timing difference
- Complex of NAND management/ NAND validation requirement



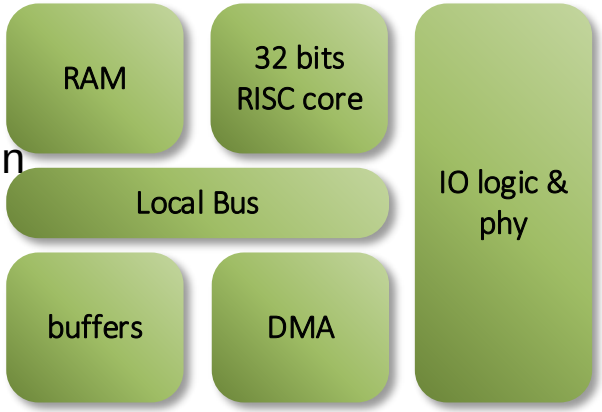
Hardware controller proposal

- Effective area cost
  - 2k FPGA LUTs in Memblaze solution
- Performance optimized
  - Most effective NAND scheduling
- Fixed pre-designed functions
  - Buffer geometry/commands/scheduling
- Poor capability for varity/future NAND device



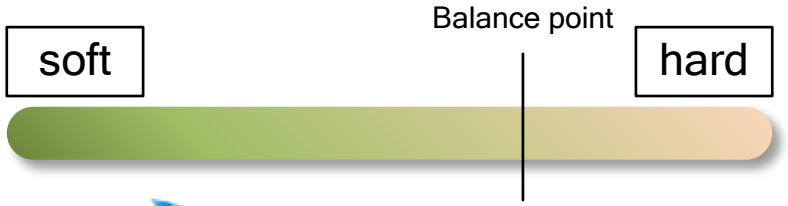
Software controller proposal

- Heave area cost
  - 7k FPGA LUTs+ 8KB memory in Memblaze solution
  - Unused CPU instruction/block
- Flexibility and control optimized
  - Soft scheduling/NAND sequence
  - Soft sequence/NAND command
- Strong capability for varity/future NAND device
  - Undocumented command/operation; variable NAND geometry



Balance point of controller design

- Feature demands
  - Configurable timing/sequence/NAND addressing/data frame size/ scheduling algorithm
  - HW scheduled and thread switch for LUN management
  - Other accelerate function as buffer/timer/power management
- Feature undemands
  - Competing performance/interrupt/stack management
  - Other CPU feature

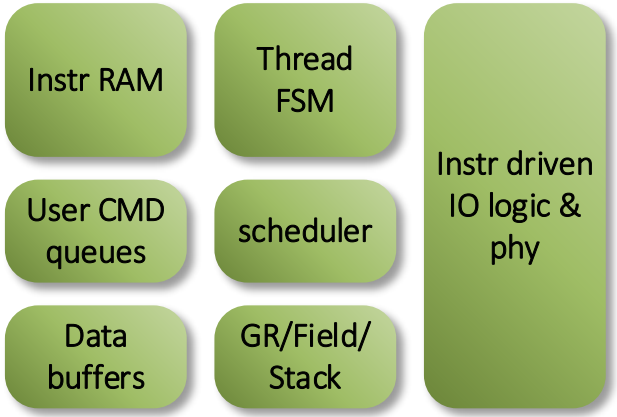


Defines a NPU- specs

- Instruction set define
  - General instruction as jump, bitwise, arithmetic operate, call, return…
  - NAND timing/sequence generation instruction
  - LUN thread/scheduling instruction
  - User command parser/queue and data Buffer management instruction
- Hardware define
  - NAND IO generation, atomic operation for NAND command/address/data
  - Buffers/scheduler/user command Queues
  - Instruction ram/general registers/status register/thread stack

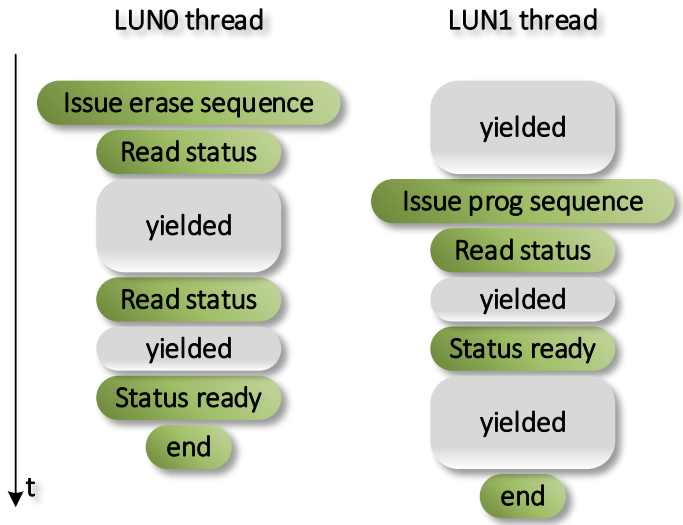
Defines a NPU- arch

- Arch define
  - IO logic & PHY, atom operation for NAND command/address/data
  - Thread FSM, running the instruction sequence as normal CPU as step of fetch/decode/execute from instruction RAM
  - Scheduler, a programmable Thread switcher, and Scheduler.
  - User command queue and data buffer, the user interface for the controller
  - GR/Field/Stack, the local storage and thread stack

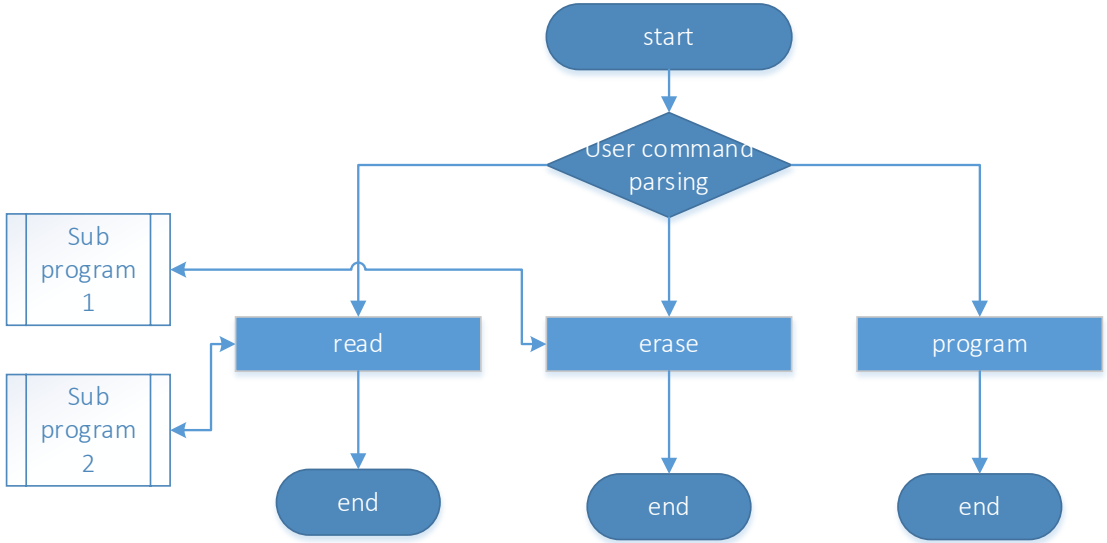


Defines a NPU- thread switches

- Thread switches
  - A thread is assigned to the LUN operation
  - Thread could be yielded at any time depending on the instruction programing. E.g., an erase thread could be yielded while status read
  - The yielded thread releases the thread FSM. thus the thread FSM could serve another LUN operations



NPU Programing model

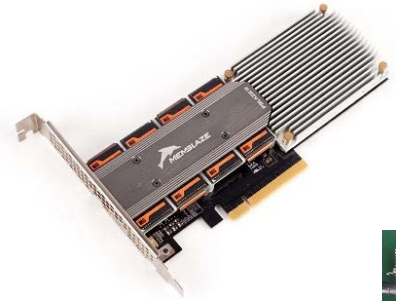


Memblaze NPU design

- Memblaze NPU features
  - Micro-code programmable interface behavior & timing controller
  - Micro-code programmable NAND operation
  - Programmable NAND PHY ADDR define
  - Programmable data frame length
  - Programmable CE configuration
  - FSM-> 32 instructions,
  - FSM-> full Turing machine
  - Multiple command queue for each LUN
  - Scheduler-> 2D programmable, mixed HPQ/WRR
  - Scheduler-> target enable/disable
  - Timer for each LUN
  - Hard Macro for LUN supports
  - Hard Macro for buffer size/length configuration
  - Hard Macro for bus size configuration
  - Hard Macro for instruction RAM size configuration
  - Hard Macro for each register address configuration
  - Hard Macro for each register default value configuration

Memblaze NPU design

	memblaze Hardware NAND Flash controller	memblaze software NAND Flash controller	Memblaze NAND flash Processor Unit
processor	-	FPGA soft CPU	Thread FSM
pipeline	-	Yes	No
interrupt	-	Yes	No
general register	-	16*32	1*32
instruction	-	Standard	full Customized
instruction Memory	-	32KByte	8KByte
Data memory	-	64KByte(share for	No
Logic	2k LUTs	7k LUTs	4k LUTs
Coding	fix function	C/C++	Assembly
Scheduler	fix	pure software	Hardware
Scheduler Algorithm	WRR	any	2D muxed WRR+HP
Hardware thread Support	Yes	No	Yes
number of threads support	16	16	Hard parameter
Timer	No	1 timer	Yes, per-thread
Hardware Buffer manage	Yes	No	Yes
Power Mangement Support	No	No	Yes
Hardware Power loss manage	Yes	No	Yes
Soft bit read	No	Yes	Yes
Channel Lock	Yes	Yes	Yes
Target lock	Yes	Yes	Yes
PHY control	fixed	Fixed	Microcode
Test debug	No	Yes	Yes
user command	fixed	full flexibility	full flexibility



Memblaze Pblaze3 with Hardware NAND controller

Memblaze flash validation platform with software NAND controller (only 1 channel valid)

Memblaze flash validation platform with NPU NAND controller (full 4 channel valid)

Summary

- NPU is the just right controller for NAND
- Right for any NAND in feature and any undisclosed NAND Vendor operations
- Right for both performance and FPGA/ASIC area cost to hit the balance point.
- Right for offload the task of main SSD processors
- Right for the precision NAND control, specially, for fine tuning in different strategy of enterprise applications

NAND Flash Processor Unit (NPU)

Thank you!!

yilei.wang@Memblaze.com