Reusability and Flexibility in SSD Controller Design - A Platform Approach

A.Vityaev, A.Saxena,

Mobiveil Inc.

Challenges Facing SSD Controller Designers

- Wide range of controller IC specs
  - Significant effort to design each IC → Design reuse is critical
- Can Software-Development methodologies be applied?
  - Provides Flexibility and Reuse, often at expense of Reliability
  - Very different development environment
- Mobiveil is Silicon IP and Platform Provider
  - Wide range of customer requirements →

  → Requires **both** Reliability **and** Flexibility/Reuse

Flash Memory Summit

Mobiveil Storage Portfolio

Silicon IP Blocks:

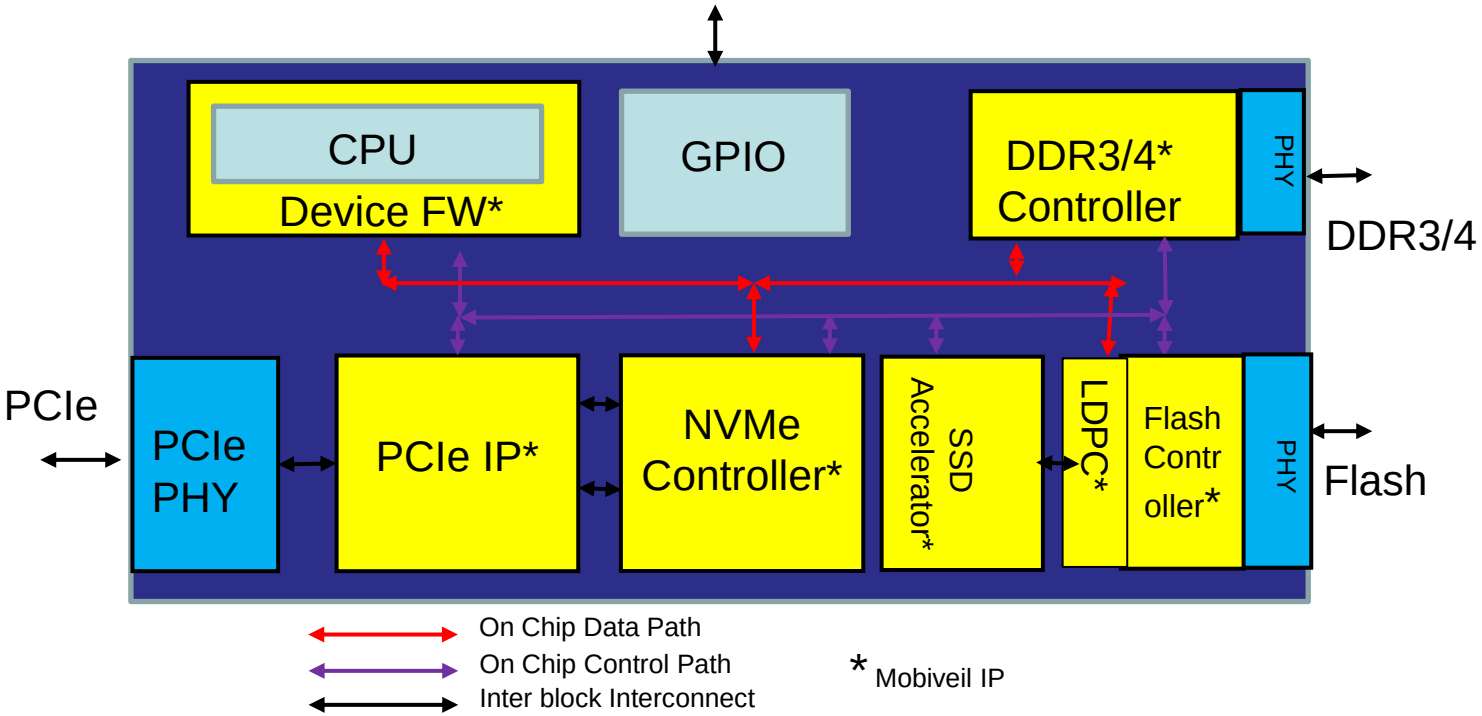
PCIe Gen4/3/2/1	HBM	Quad-SPI	LDPC
NVMe 1.3	ONFI/Toggle	eMMC	FTL
DDR/LPDDR	Hyper Bus	U-NFC	

Platforms:

SSD Platform

Flash Characterization

NVMe SSD Controller Platform



Software vs. Chip Development

- More product revisions
- Complex code hierarchy
- Extensive use of external components
- Complex (API) interface between components

Higher Flexibility & Reuse

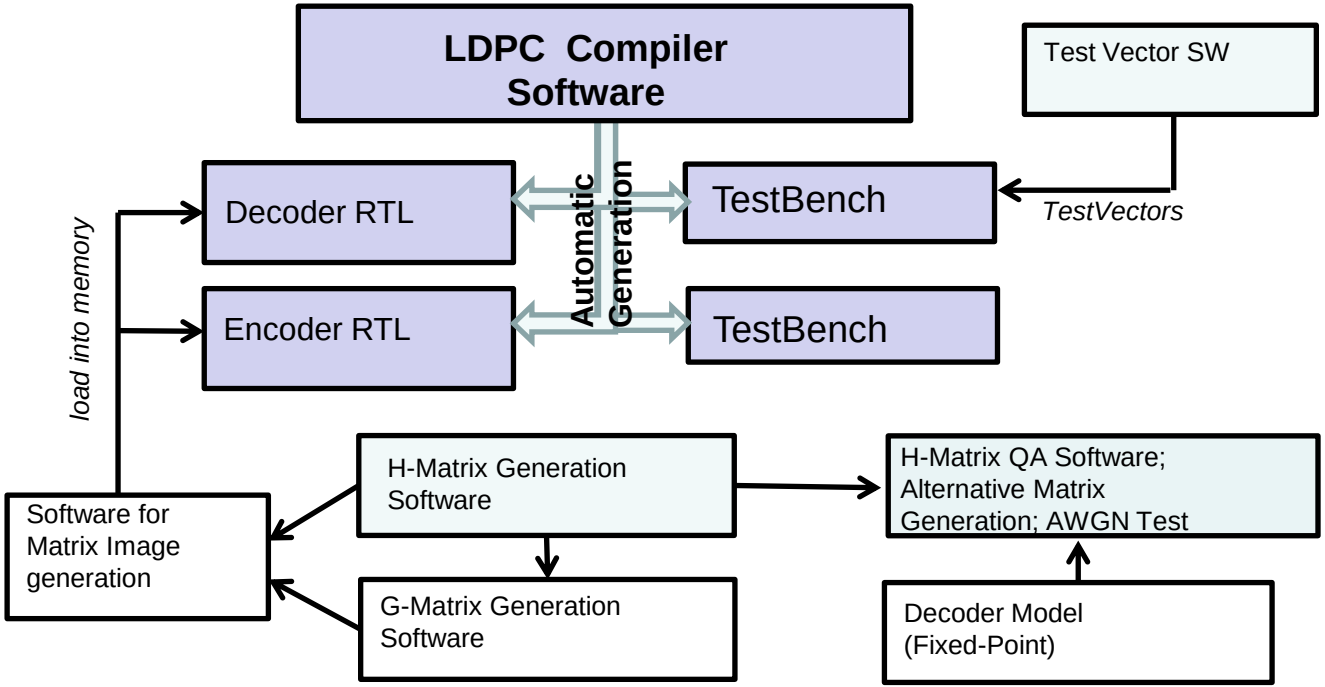
Higher Reliability

- Few product revisions
- Flatter code hierarchy
- Tightly controlled use of external components
- Transparent interface between components

Design Compilers

- Best solution for blocks with highest spec variability
- Software components generating final Verilog files
- Automatically generated Test-Benches for given parameters
- Software for generating Test Vectors
- Control via programmable state machines
  - Micro-Processor cores are not used

Example: LDPC Compiler



Benefits of Design Compilers

- Wide range of specifications for Silicon IP Cores that can be generated
- No increase in complexity of the generated IP Cores
  - Flexibility adds complexity to the Design Compiler, not to the resulting IP Core
- No added complexity for Real-Time Firmware
  - Configuration files are loaded at Start-Up or in Deep Recovery Modes

LDPC Compiler Spec Range

Range of key spec's that can be achieved in Compiler-generated IP Cores

LDPC IP Core Spec	Range
Max code word size*	0 to >16KB
Max parity percentage*	0% to over 50%
Data rate (End-of-Life per core)	50MB/s to > 4GB/s
Max simultaneous LDPC codes* (On-the-Fly-switching)	1-8
Max number of LDPC codes (Off-line reprogramming)	Unlimited
Interface Buses Widths	Wide range

**Maxima are set during compile time, after IP Core is generated, any value smaller than Max can be used*

Functional Programmability

- Extensive use of programmable state machines and sequencers
  - Micro-processor cores are not used inside blocks
  - Provides design flexibility without affecting real-time FW
  - Simple / “Flat Hierarchy” solution → more reliable

Flash Reliability Subsystem / LDPC
Program user LDPC matrices
Programmable puncturing & padding
Programmable LUT select algo
Configurable soft-read procedure

Flash Interface Controller
Program an arbitrary command
Flexible timing control
Configurable address scheme
Configurable R/B scheme

Component Interfaces

- Software approach (API) can not be used in hardware
  - In high-reliability software applications, “simple” interfaces are frequently used

- Standard interface buses used in Mobiveil Platform
- Configurable Interface and DataPath Widths

Flash Controller DataPath →	Configurable Width
LDPC DataPath →	Configurable Width
NVMe Controller DataPath→	Configurable Width

Configurability of Controller Components

SSD Controller
FTL in Hardware
Configurable Data Striping
Configurable Number of Data Pools

DDR3/4 Controller
Fully Configurable

NVMe Controller
Multi-Port or Single-Port
Configurable Number of IO Queues
Configurable IO Queue Depth
Configurable # of DMA Engines
Configurable Buffer Size
Configurable # of NV Memory Channels

For More Information

- "An FPGA-Based NVMe SSD Subsystem" Presentation at FMS Theater, Wed. 6pm
- Visit us at Mobiveil Booth #610
- Email to Info@Mobiveil.com