

HW/SW CO-VALIDATION

Michael Bair, Principal Engineer, Intel Corporation

AGENDA

- Background
 - What I want
 - The direction we have taken
 - Looking for feedback
-

MY HW-CENTRIC VIEW IS CHANGING...

- HW Design still needs to do the following:
 - Increase performance / lower power
 - Add new features
 - Integrate *everything*
 - Etc...
-

MY HW-CENTRIC VIEW IS CHANGING...

- Additionally, HW Design now needs to do the following
 - Help ***anybody*** create platforms (simplicity, ease of use)
 - Help the platforms be low cost
 - Help the systems verify themselves
 - And otherwise... get the hell out of the way of SW
-

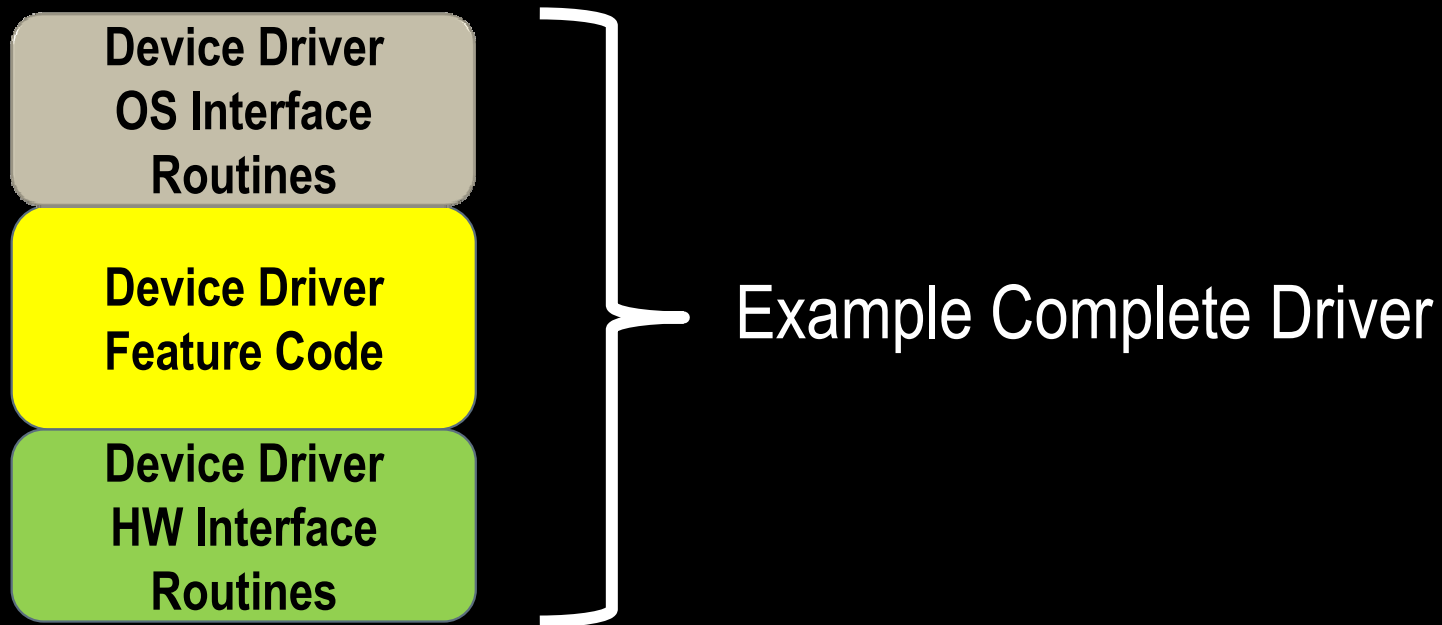
REQUIREMENTS FOR HW/SW CO-DESIGN

I need a system for HW/SW design, and in that system...

- I want Isolation of HW/SW for development
 - I want it to work regardless of the amount of change in HW and SW
 - I want it to minimize unneeded change and/or thrash
 - I want it to allow seamless integration
 - I want it to work regardless of how close the teams are or how well they get along
 - I want it to work with all systems (OS's, Code standards, Development methods, validation techniques, etc)
 - I do NOT want to over-validate my HW or SW
 - And I want it to be cheap, fast, flexible
-

MY WISH STATE: A COMPOSITE OF WHAT WE'VE DONE RIGHT

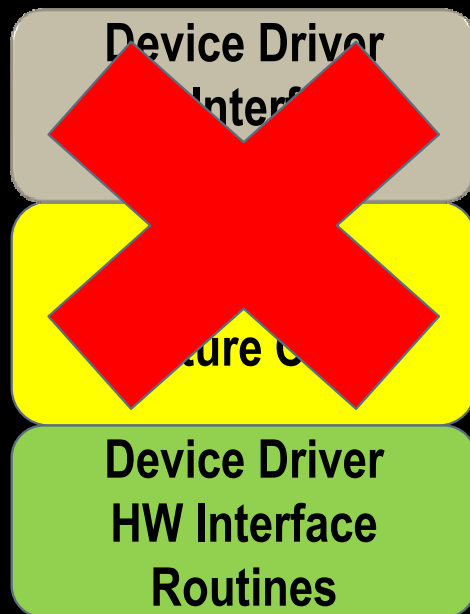
- It all starts with modular SW...



...During Isolation...

MY WISH STATE: A COMPOSITE OF WHAT WE'VE DONE RIGHT

- The HW Interface routines get used by HW team too!



These routines are like a contract between HW and SW...
Use them to develop SW, and to validate HW!

...During Isolation...

MY WISH STATE:

A COMPOSITE OF WHAT WE'VE DONE RIGHT

- A Platform Simulator to run my entire SW stack!
 - Speed and HW-correctness => Well Validated SW

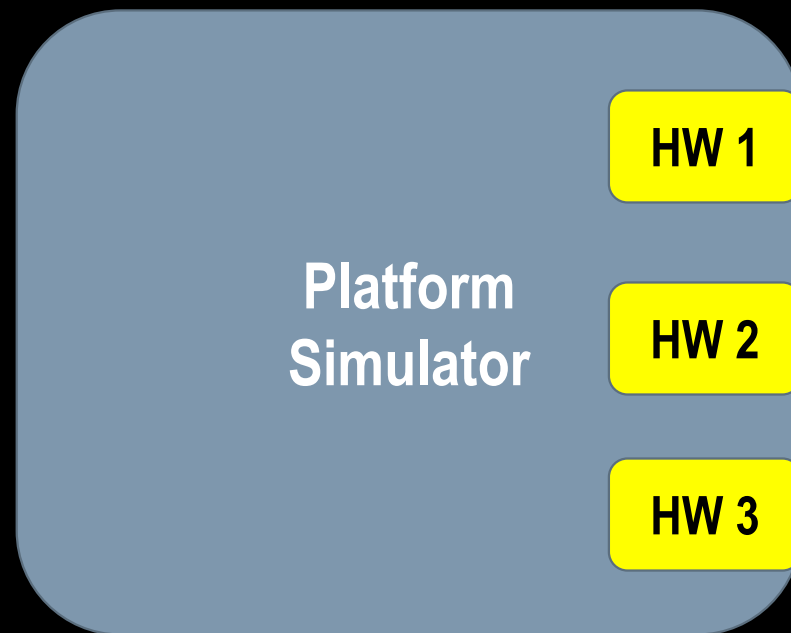


...During Isolation...

MY WISH STATE:

A COMPOSITE OF WHAT WE'VE DONE RIGHT

- A Platform Simulator that can pull in pieces of HW
 - RTL or FPGA

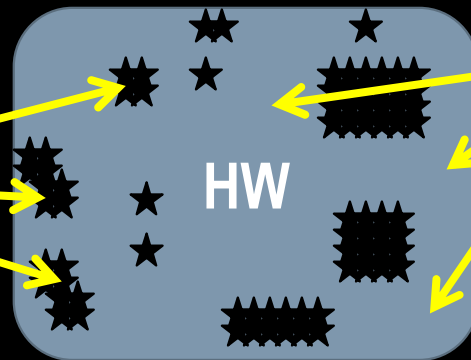


...Transition from Isolation to Integration...

MY WISH STATE: A COMPOSITE OF WHAT WE'VE DONE RIGHT

- Validate HW with real SW
 - Limit synthetic validation to critical areas

Validate the HW with real SW – automatically targets important areas



Do not validate the HW where it is not used!!
Exceptions apply: Think security, IO stress, etc

...During Integration...

IS IT THAT SIMPLE?

- Modular SW
- HW Interface Routines used as building block for SW... and Validation tool for HW
- Fast platform simulator that can be replaced piecemeal with HW
- After Integration, use OS-based validation as much as possible

Or... am I making a fundamental mistake in my requirements?

- Isolation, Etc
-

Q & A