IBIS Hierarchical Overrides and BIRD88

Bob Ross

DesignCon East IBIS Summit Meeting

Boxborough, Massachusetts

April 5, 2004

TERASPEED CONSULTING GROUP

What is Hierarchical Override?

- Add Override rather than "Conditional" Replacement
  - Letting higher order detail/functionality win
  - Practical needs drove expansion
- History
  - IBIS grew from 8 pages in 1993 to 140+ in 2004 with layers of capability blended into EBD (later), [Component], [Package], [Model] – Cornerstone skeleton for surrounding development
  - Least disruptive with downward/upward compatible changes
  - Presentation motivation: BIRD88 [Driver Schedule] initialization
- Many cases from the early days, for example:
  - L_pin overrides default L_pkg,
  - [Rising/Falling Waveform] overrides [Ramp]
- Plus EDA interface overrides (e.g., model and differential pin assignments)

Why this Approach?

- Least disruptive
  - Keeps SIMPLE rules rather than "conditional"
  - Keeps IBIS sections similar – easier visual inspection
  - Keeps parser simple – lower cost, fewer "conditional" upgrades to specify, track, check, repair, …
  - Keeps original required "defaults", even if not fully meaningful for syntactical robustness (Typ column always required)
- Penalties
  - Not always technically robust or correct ([Ramp] for LVDS)
  - Redundant (or wrong) information confusing and wasteful
  - Inheritance of capabilities differences (e.g., [Driver Schedule] versus [Submodel] with inheritance of voltage rails)
  - Burden falls on EDA vendors to sort out and implement properly

TERASPEED CONSULTING GROUP

Package Example Overrides (H to L)

- EBD's a separate issue (calls EBDs or components)
- ICM under discussion possibly using [Alternate Package Model Mechanism]
- [Define Package], [Package Model] (V. 2.1, V. 3.2)
  - Coupled package Matrices (V. 2.1), or Len L, R, C/ (V. 3.2) overrides *_pin and *_pkg below
    - [Number of Sections]/[Model Data] "flags" instead of overrides
  - [Package Model] in .ibs file overrides same model in .pkg file
- [Model] L_pin, R_pin, C_pin (V. 1.1)
  - Optional specific pin overrides default package
- [Package] L_pkg, R_pkg, C_pkg (V. 1.1)
  - Required Typ column – lowest order default

Voltage Rail Overrides

- [External Model] under [Model] (V. 4.1) (external power)
- [Pin Mapping] assignments and EDA or multi-lingual rail voltage assignments – some EDA vendors may override voltages (V. 3.2, V. 4.1))
- [Pullup Reference], [Pulldown Reference], [Gnd Clamp Reference], [Power Clamp Reference], (V. 2.1), [External Reference] (V. 4.1)
  - SIMPLE RULE: [Voltage Range] REQUIRED (unless first FOUR rails are given) – no conditional mix and match
  - Gives added detail not available from [Voltage Range] – such as ECL and PECL voltage, hot swap clamps, etc.
- [Voltage Range] and implied default 0 V ground (V. 1.1)
  - Typ column required

TERASPEED CONSULTING GROUP

Threshold Overrides

- [External Model] under [Model] thresholds (V. 4.1)
  - D_receive
- [Diff Pin] threshold vdiff (V. 2.1)
  - Overrides single ended thresholds below
- [Receiver Thresholds] under [Model] (V. 4.1)
  - Adds detail and new thresholds including some differential checking conditions
- [Model Spec] thresholds under [Model] (V. 3.2)
  - Adds typ/min/max detail and more choices
- [Model] subparameters (V. 1.1, V. 2.1)
  - Some still required for certain Model_types
  - Historical Vinh, Vinl requirements retained even if overridden by [Model Specification] (with legacy 2.0 V and 0.8 V defaults)

C_comp – an Exception

- [External Model] under [Model] (V. 4.1)
  - C_comp, C_comp_* optional
- [Model] C_comp_pullup, C_comp_pulldown, C_comp_gnd_clamp, C_comp_power_clamp (V. 4.1)
  - Different rule, C_comp NOT required if any of above exist
  - So rule is at least one of C_comp or C_comp_* must exist
- [Model] C_comp (V. 1.1)
  - Required Typ value
- Scheduled [Model] under [Model]
  - C_comp values should be "0", should be ignored since the overall mode control (Model_type) is at the higher level [Model]
  - Still required entry in scheduled [Model] for simplicity of documentation

[Diff Pin] and [Model] Overrides

- [Diff Pin] partial overrides of [Model] (V. 2.1)
  - Polarity set by [Diff Pin] and inv_pin columns
    - Needed since same [Model] can be used for any pin
  - vdiff threshold used instead of model thresholds
  - Skewing of inverting operation with respect to non-inverting (tdelay_*)
- [Model] (V. 1.1)
  - Polarity and other subparameters override those below
  - [Model] STATE is central
- [Driver Schedule] references to models (V. 3.2)
  - Initiation and polarity of scheduled models made relative to the [Model] STATE and [Driver Schedule] delays
  - Specified in BIRD88

TERASPEED CONSULTING GROUP

My (stated/unstated) [Driver Schedule] Assumptions (in BIRD35.3)

- Under [Model], the model state assumptions must prevail
  - Works for single transition and one cycle (Rising and Falling)
  - FULL CYCLE for [Driver Schedule]
    - Rise_on_dly Rise_off_dly Fall_on_dly Fall_off_dly
    - r1 or NA r2 or NA f3 or NA f4 or NA
    - Full cycle => down-up (d-u), (u-d), (d-u-d-u), (u-d-u-d) sequencing for "slow" clock (assumes no over-clocking)
    - 16 possibilities, but 5 [r/f || NA] table variations with 8 total variations considering numerical ordering, finite durations
  - Follows top-level [Model] state (Low or High) and input stimulus in transparent manner including Polarity
  - Everything is relative to Rise or Fall edges of "internal simulator pulse"
- Therefore known initial state (High or Low) allows single switch and one cycle simulation

TERASPEED CONSULTING GROUP

Scheduled Model Initial States (should be implemented for full capability)

R_on	R_off	F_on	F_off	Model Low	Model High
r	NA	f	NA	Low	High
NA	r	NA	f	High	Low
r1	r2	NA	NA	Low	Low
r2	r1	NA	NA	High	High
NA	NA	f1	f2	High	High
NA	NA	f2	f1	Low	Low
r1	r2	f2	f1	Low	Low
r2	r1	f1	f2	High	High

Illegal Cases

R_on	R_off	F_on	F_off	Reason
NA	NA	NA	NA	unknown initial state
r	NA	NA	f	u-u sequencing
NA	r	f	NA	d-d sequencing
r1	r2	f1	f2	u-d-d-u sequencing
r2	r1	f2	f1	d-u-u-d sequencing

And just 1 or 3 numerical entries (sequencing)

Summary

- Expansion of IBIS based mostly on evolutionary hierarchical overrides
  - Simple rules (easier specification and parser maintenance)
  - Some exceptions – and complicated rules
- [Driver Schedule] insertion consistent with [Model] states
  - Initial scheduled model states defined in BIRD88, as originally intended

TERASPEED CONSULTING GROUP