Portmap Syntax Proposal

IBIS Interconnect Task Group Meeting

September 20, 2023

Arpad Muranyi, Senior Product Architect

SIEMENS

Outline

Requirements for the planned port mapping syntax
Original HL_portmap syntax
Using CONNECT for Extended Net definitions
Extensions to support DIFFPORTS, DESCRIPTION, SIDE
Illustration for a Touchstone file used in an EMD model
Differences between IBIS package models and EMD models
Expanded HL_portmap syntax to support port connections in IBIS package and EMD models

General requirements for the planned port mapping syntax

Support “flexible usage” of port connections

- Needed when the Touchstone model is used in an undefined, variable environment
  - e.g., connectors, active devices (transistors, IC-s), etc., on PCB footprint
- Ports should have meaningful descriptions which describe their **functionality** without knowing what is connected to them on the outside (e.g., base, emitter, collector, connector’s pin name, IBIS [Pin] name, top level [EMD Pin List] name)
- We might consider creating a list of reserved names for common, known functionalities

Support “fixed usage” of port connections

- Needed when the Touchstone model is used in a well-defined, unchanging environment
  - e.g., package model, on-die interconnect, multi-chip module, memory module, etc.
- Ports should have complete connectivity information about **what is connected to them** on the outside (connecting to IBIS RefDes.PinName or nested EMD RefDes.PinName)

Support a combination of the above within the same Touchstone file

- e.g., “flexible usage” [EMD Pin List] to PCB footprint connection on one side and “fixed usage” [Designator Pin List] to IBIS [Pin] connection on the other side

Requirement details for the planned port mapping syntax

Support extended net definitions

- List(s) of port numbers which are “electrically connected”, i.e., have low insertion loss at the Nyquist frequency

Support differential port pair definitions

- List(s) of port number pairs which are used for differential signaling
- In the future this could be extended to lists of port number **triplets**, **quadruplets**, etc., which are used in chord signaling

Support Side (End) definitions

- These are port attributes to group them by general physical location

Support port description definitions

- These are port attributes to give them meaningful, functional descriptions

Requirement details (cont'd)

Support signal only, PDN only, or combined signal and PDN models

Support pin grouping (for either the positive or the negative terminals of the ports)

Support IBIS package model port connections to **IBIS pin, pad** and **buffer terminal** with **pin_name**, **signal_name** and **bus_label**

Support definitions for making connections to Designator ground pins

Support reference terminal connections to EMD/IBIS pins, Designator pins and "universal ground" (A_gnd)

The existing HL_portmap syntax definition

```
! PORT <port_number> [+](TermDescription) [-(TermDescription)] [NAME:<port_name>]
```

```
TermDescription :: =PinTerm | PinGroupTerm
PinTerm ::= PIN <RefDes>.<PinName> [,[<NetName>] [,S|P ]]
PinGroupTerm ::= GROUP[:<group_name>] [,[<NetName>] [,S|P ]]
```

```
! GROUP PORT <port_number> [NAME:<group_name>,] PinList
PinList ::= <RefDes>.<PinName> [,PinList]
```

```
! CONNECT ConnList
```

The existing “CONNECT” statement would work for Extended Net definitions

```
! CONNECT  1,  2,  3,  4,  5,  6;
! CONNECT  7,  8,  9, 10, 11, 12;
```

or

```
! CONNECT  1,  2,  3,  4,  5,  6;  7,  8,  9, 10, 11, 12;
```

- CONNECT statements should be optional
- They are ***NOT*** the same as the HSPICE .connect statement!
- They should be thought of as signal flow or extended net definitions
- The port list(s) define electrically associated ports
- Each port list is terminated by a semicolon
- Port numbers within a port list are separated by a comma (white space?)
- A port list may span multiple lines, but each line should start with “CONNECT” and end with a semicolon (;) (?)
- Multiple port lists may be placed on the same line

We could add “DIFFPORTS” statements to support Differential Port definitions

```
! DIFFPORTS  1,  2;  3,  4;  5,  6;
! DIFFPORTS  7,  8;  9, 10; 11, 12;
```

- DIFFPORTS statements should be optional
- Port duplets define which ports are differential pairs
- Non-inverting port number is first, followed by the inverting port number
- Each port pair is terminated by a semicolon
- Port numbers within a port pair are separated by a comma (white space?)
- Multiple port duplets may be placed on the same line

We could add “DESCRIPTION” to each PORT line to support Port Description definitions

```
! PORT <PortNumber> [+](TermDescription) [-(TermDescription)] [NAME:<PortName>] [DESCRIPTION:<PortDescription>]
```

- DESCRIPTION statements should be optional
- Could be used to define “pre-layout” functionality for the ports or they could contain an arbitrary string (I am not sure how useful that would be)
  - We may consider making a list of reserved functionality names, such as drain, gate, source, base, emitter, collector, etc.

We could add “SIDE” to each PORT line to support Port Side definitions

```
! PORT <PortNumber> [+](TermDescription) [-(TermDescription)] [NAME:<PortName>] [SIDE:<PortSideName>]
```

- SIDE statements should be optional
- Could be used to define the general physical location of groups of ports
  - We may consider making a list of reserved side names
- All ports with the same SIDE shall be on the same RefDes

HL_portmap illustration

including extensions for DIFFPORTS and SIDE

for a Touchstone file used in an EMD model

```
! HL_PORTMAP 1.0
! PORT 1   +(PIN U1.94,  A0, S)     -(GROUP:GND.U1,  GND, P)  NAME:A0.U1.94.GND      SIDE:Die1
! PORT 2   +(PIN U2.94,  A0, S)     -(GROUP:GND.U2,  GND, P)  NAME:A0.U2.94.GND      SIDE:Die2
! PORT 3   +(PIN U3.94,  A0, S)     -(GROUP:GND.U3,  GND, P)  NAME:A0.U3.94.GND      SIDE:Die3
! PORT 6   +(PIN BGA.K2, A0, S)     -(GROUP:GND.BGA, GND, P)  NAME:A0.BGA.K2.GND     SIDE:EMDpin
! ...
! PORT 7   +(PIN U1.95,  A1, S)     -(GROUP:GND.U1,  GND, P)  NAME:A1.U1.95.GND      SIDE:Die1
! PORT 8   +(PIN U2.95,  A1, S)     -(GROUP:GND.U2,  GND, P)  NAME:A1.U2.95.GND      SIDE:Die2
! PORT 9   +(PIN U3.95,  A1, S)     -(GROUP:GND.U3,  GND, P)  NAME:A1.U3.95.GND      SIDE:Die3
! PORT 12  +(PIN BGA.J2, A1, S)     -(GROUP:GND.BGA, GND, P)  NAME:A1.BGA.J2.GND     SIDE:EMDpin
! ...
! PORT 121 +(PIN U1.71,  CLK_c, S)  -(GROUP:GND.U1,  GND, P)  NAME:CLK_c.U1.71.GND   SIDE:Die1
! PORT 126 +(PIN BGA.E2, CLK_c, S)  -(GROUP:GND.BGA, GND, P)  NAME:CLK_c.BGA.E2.GND  SIDE:EMDpin
! PORT 127 +(PIN U1.70,  CLK_t, S)  -(GROUP:GND.U1,  GND, P)  NAME:CLK_t.U1.70.GND   SIDE:Die1
! PORT 132 +(PIN BGA.D2, CLK_t, S)  -(GROUP:GND.BGA, GND, P)  NAME:CLK_t.BGA.D2.GND  SIDE:EMDpin
! ...
! PORT 163 +(GROUP VDD.U1,  VDD, P) -(GROUP GND.U1,  GND, P)  NAME:VDD.VDD.U1.GND    SIDE:Die1
! PORT 164 +(GROUP VDD.U2,  VDD, P) -(GROUP GND.U2,  GND, P)  NAME:VDD.VDD.U2.GND    SIDE:Die2
! PORT 165 +(GROUP VDD.U3,  VDD, P) -(GROUP GND.U3,  GND, P)  NAME:VDD.VDD.U3.GND    SIDE:Die3
! PORT 168 +(GROUP VDD.BGA, VDD, P) -(GROUP GND.BGA, GND, P)  NAME:VDD.VDD.BGA.GND   SIDE:EMDpin
! ...
! GROUP PORT   1 NAME:GND.U1,   U1.2,   U1.11,  U1.15,  U1.23,  U1.38,  U1.42,  U1.43,  U1.47,  U1.50,  U1.54,  U1.55,  U1.59,  U1.60,  U1.64...
! GROUP PORT   2 NAME:GND.U2,   U2.2,   U2.11,  U2.15,  U2.23,  U2.38,  U2.42,  U2.43,  U2.47,  U2.50,  U2.54,  U2.55,  U2.59,  U2.60,  U2.64...
! GROUP PORT   3 NAME:GND.U3,   U3.2,   U3.11,  U3.15,  U3.23,  U3.38,  U3.42,  U3.43,  U3.47,  U3.50,  U3.54,  U3.55,  U3.59,  U3.60,  U3.64...
! GROUP PORT   6 NAME:GND.BGA,  BGA.A5, BGA.A9, BGA.A11, BGA.A14, BGA.AA5, BGA.AA13, BGA.AC5, BGA.AC9, BGA.AC11, BGA.AC14, BGA.B2, BGA.C1, BGA.C4, BGA.C5...
! ...
! GROUP PORT   7 NAME:GND.U1,   U1.2,   U1.11,  U1.15,  U1.23,  U1.38,  U1.42,  U1.43,  U1.47,  U1.50,  U1.54,  U1.55,  U1.59,  U1.60,  U1.64...
! GROUP PORT   8 NAME:GND.U2,   U2.2,   U2.11,  U2.15,  U2.23,  U2.38,  U2.42,  U2.43,  U2.47,  U2.50,  U2.54,  U2.55,  U2.59,  U2.60,  U2.64...
! GROUP PORT   9 NAME:GND.U3,   U3.2,   U3.11,  U3.15,  U3.23,  U3.38,  U3.42,  U3.43,  U3.47,  U3.50,  U3.54,  U3.55,  U3.59,  U3.60,  U3.64...
! GROUP PORT  12 NAME:GND.BGA,  BGA.A5, BGA.A9, BGA.A11, BGA.A14, BGA.AA5, BGA.AA13, BGA.AC5, BGA.AC9, BGA.AC11, BGA.AC14, BGA.B2, BGA.C1, BGA.C4, BGA.C5...
! ...
! GROUP PORT 163 NAME:VDD.U1,   U1.4,   U1.8,   U1.17,  U1.20,  U1.33,  U1.36,  U1.40,  U1.45,  U1.49,  U1.52,  U1.57,  U1.62,  U1.65,  U1.67...
! GROUP PORT 163 NAME:GND.U1,   U1.2,   U1.11,  U1.15,  U1.23,  U1.38,  U1.42,  U1.43,  U1.47,  U1.50,  U1.54,  U1.55,  U1.59,  U1.60,  U1.64...
! GROUP PORT 164 NAME:VDD.U2,   U2.4,   U2.8,   U2.17,  U2.20,  U2.33,  U2.36,  U2.40,  U2.45,  U2.49,  U2.52,  U2.57,  U2.62,  U2.65,  U2.67...
! GROUP PORT 164 NAME:GND.U2,   U2.2,   U2.11,  U2.15,  U2.23,  U2.38,  U2.42,  U2.43,  U2.47,  U2.50,  U2.54,  U2.55,  U2.59,  U2.60,  U2.64...
! GROUP PORT 165 NAME:VDD.U3,   U3.4,   U3.8,   U3.17,  U3.20,  U3.33,  U3.36,  U3.40,  U3.45,  U3.49,  U3.52,  U3.57,  U3.62,  U3.65,  U3.67...
! GROUP PORT 165 NAME:GND.U3,   U3.2,   U3.11,  U3.15,  U3.23,  U3.38,  U3.42,  U3.43,  U3.47,  U3.50,  U3.54,  U3.55,  U3.59,  U3.60,  U3.64...
! GROUP PORT 168 NAME:VDD.BGA, BGA.A3, BGA.A4, BGA.A7,  BGA.AB5, BGA.AB9, BGA.AB11, BGA.AB13, BGA.AC7, BGA.B1,  BGA.B5,  BGA.B9, BGA.B11, BGA.B13, BGA.C3...
! GROUP PORT 168 NAME:GND.BGA, BGA.A5, BGA.A9, BGA.A11, BGA.A14, BGA.AA5, BGA.AA13, BGA.AC5, BGA.AC9, BGA.AC11, BGA.AC14, BGA.B2, BGA.C1, BGA.C4, BGA.C5...
! ...
! CONNECT 1, 2, 3, 6; 7, 8, 9, 12; 121, 126; 127, 132;
! DIFFPORTS  127, 121; 132, 126;
! END_HL_PORTMAP
```


SIEMENS

Differences between IBIS package models and EMD models

IBIS has three “interfaces”

- Pin, Pad, Buffer
- Package models go between pin and pad
- On-die interconnect models go between pad and buffer
- Combined models go between pin and buffer

There is no RefDes inside an IBIS model (U1, U2, etc.)

Connections in IBIS are made by combinations of:

- pin names, (pad names [Die Supply Pads]), signal names, bus label names
- Pin_I/O, Pad_I/O, Buffer_I/O, Pin_rail, Pad_rail, Buffer_rail, *_ref terminals, A_gnd

As opposed to that, EMD has only two interfaces

- The interconnect model goes between [EMD Pin List] and [Designator Pin List]

Connections in EMD are made by combinations of:

- pin names, signal names, bus label names
- Pin_I/O, Pin_rail, A_gnd
- Designator connections use the **<RefDes>.<PinName | SignalName | BusLabelName>** syntax

Summary of IBIS package modeling syntax

```
Signals:
   all connections (pin, pad, buffer) by I/O pin_name

Rails:
   at pin:
      rail pin_name, rail signal_name, bus_label
   at pad:
      rail signal_name, bus_label, [Die Supply Pads] pad_name
   at buffer:
      rail signal_name, bus_label, I/O pin_name
```

```
Signals:
   Terminal Type:       Pin_I/O, or
                        Pad_I/O, or
                        Buffer_I/O
   Qualifier + Entry:   pin_name <SignalPinName>

Rails at pin:
   Terminal Type:       Pin_Rail
   Qualifier + Entry:   pin_name <RailPinName>, or
                        signal_name <RailSignalName>, or
                        bus_label <BusLabelName>

Rails at pad:
   Terminal Type:       Pad_Rail
   Qualifier + Entry:   signal_name <RailSignalName>, or
                        bus_label <BusLabelName>, or
                        pad_name <PadName from [Die Supply Pads]>

Rails at buffer:
   Terminal Type:       Buffer_Rail
   Qualifier + Entry:   signal_name <RailSignalName>, or
                        bus_label <BusLabelName>

   Terminal Type:       Pullup_ref, or
                        Pulldown_ref, or
                        Power_clamp_ref, or
                        Gnd_clamp_ref, or
                        Ext_ref
   Qualifier + Entry:   pin_name <SignalPinName>

Any interface:
   Terminal Type:       A_gnd
   Qualifier + Entry:   None
```

Summary of EMD modeling syntax

```
Signals:
   all connections by I/O pin_name

Rails (at pin only):
   rail pin_name, rail signal_name, bus_label
```

```
Signals:
   Terminal Type:       Pin_I/O
   Qualifier + Entry:   pin_name <EMD_SignalPinName>, or
                        pin_name <Designator_RefDes.SignalPinName>

Rails (at pin):
   Terminal Type:       Pin_Rail
   Qualifier + Entry:   pin_name <EMD_RailPinName>, or
                        pin_name <Designator_RefDes.RailPinName>

                        signal_name <EMD_RailSignalName>, or
                        signal_name <Designator_RefDes.RailSignalName>, or
                        signal_name <*.Designator_RailSignalName>

                        bus_label <EMD_BusLabelName>, or
                        bus_label <Designator_RefDes.RailSignalName>, or
                        bus_label <*.Designator_BusLabelName>

Any interface:
   Terminal Type:       A_gnd
   Qualifier + Entry:   None
```

Hierarchy of terminal types, (qualifiers) and <QualifierEntries>

Terminal types and associated (qualifiers):

```
Pin_I/O           (pin_name)
Pad_I/O           (pin_name)
Buffer_I/O        (pin_name)

Pin_Rail          (pin_name)  (signal_name)  (bus_label)
Pad_Rail          (pad_name)  (signal_name)  (bus_label)
Buffer_Rail       ---         (signal_name)  (bus_label)

Pullup_ref        (pin_name)
Pulldown_ref      (pin_name)
Power_clamp_ref   (pin_name)
Gnd_clamp_ref     (pin_name)
Ext_ref           (pin_name)

A_gnd             ---
```

Qualifiers and associated <QualifierEntries> with comments:

```
pin_name
   <SignalPinName>          from IBIS [Pin] (pkg model) or “top level” [EMD Pin List]
   <RefDes.SignalPinName>   from [Designator Pin List]
   <RailPinName>            from IBIS [Pin] (pkg model) or “top level” [EMD Pin List]
   <RefDes.RailPinName>     from [Designator Pin List]

pad_name
   <PadName>                from [Die Supply Pads] (pkg model)

signal_name
   <RailSignalName>         from IBIS [Pin] (pkg model) or “top level” [EMD Pin List]
   <RefDes.RailSignalName>  from [Designator Pin List]
   <*.RailSignalName>       from [Designator Pin List]

bus_label
   <BusLabelName>           from IBIS keywords (pkg model) or “top level” [EMD Pin List]
   <RefDes.BusLabelName>    from [Designator Pin List]
   <*.BusLabelName>         from [Designator Pin List]
```

Expanded HL_portmap syntax to support port connections in IBIS package and EMD models

```
! PORT <PortNumber> [+](TermDescription) [-(TermDescription)] [NAME:<PortName>] [DESCRIPTION:<PortDescription>]
                                                                                 [SIDE:<PortSideName>]

  TermDescription ::= PinTerm | PinGroupTerm  | SigPinTerm  | SigPadTerm  | SigBufTerm  |
                                              RailPinTerm | RailPadTerm | RailBufTerm |
                                              PUrefTerm | PDrefTerm | PCrefTerm | GCrefTerm | EXTrefTerm | AgndTerm

  PinTerm      ::= PIN <RefDes>.<PinName> [, [<NetName>] [, S|P]]
  PinGroupTerm ::= GROUP [:<group_name>]  [, [<NetName>] [, S|P]]

  SigPinTerm   ::= Pin_I/O  : pin_name, <QualifierEntry>
  SigPadTerm   ::= Pad_I/O  : pin_name, <QualifierEntry>
  SigBufTerm   ::= Buf_I/O  : pin_name, <QualifierEntry>

  RailPinTerm  ::= Pin_rail : pin_name, | signal_name, | bus_label, <QualifierEntry>
  RailPadTerm  ::= Pad_rail : pad_name, | signal_name, | bus_label, <QualifierEntry>
  RailBufTerm  ::= Buf_rail :           signal_name, | bus_label, <QualifierEntry>

  PUrefTerm    ::= PU_ref   : pin_name, <QualifierEntry>
  PDrefTerm    ::= PD_ref   : pin_name, <QualifierEntry>
  PCrefTerm    ::= PC_ref   : pin_name, <QualifierEntry>
  GCrefTerm    ::= GC_ref   : pin_name, <QualifierEntry>
  EXTrefTerm   ::= Ext_ref  : pin_name, <QualifierEntry>

  AgndTerm     ::= A_gnd
```

Where <QualifierEntries> are:

```
pin_name,    <SignalPinName>  | <RefDes.SignalPinName>  | <RailPinName> | <RefDes.RailPinName>
pad_name,    <PadName>
signal_name, <RailSignalName> | <RefDes.RailSignalName> | <*.RailSignalName>
bus_label,   <BusLabelName>   | <RefDes.BusLabelName>   | <*.BusLabelName>
```

Thank you!

SIEMENS