

Designing High Speed Printed Circuit Boards Using DxDesigner and Expedition

Robert Navarro

Jet Propulsion Laboratory, California Institute of Technology.

User2User

The 2007 Mentor Graphics International User Conference

Abstract: Mentor's DxDesigner and Expedition schematic capture and printed circuit board tools were chosen to implement a custom high speed signal processing board containing many high pin count Field Programmable Gate Arrays and many high speed serial connections with data rates over 2 Gigasamples/sec. The methodology used to place the parts and route the board involved the interaction of both the DxDesigner and Expedition tools. The basic design philosophy was to specify as much as possible through design constraints at the schematic level. This paper will explore implementing that philosophy in both tools to facilitate part placement and trace routing.

Introduction

The process of designing state of the art printed circuit boards (PCB's) is very time consuming and detail oriented. Modern schematic capture and board layout systems contain numerous features to aid in successful implementation. Many aspects of PCB design are common to all computer aided design (CAD) systems, but the particular system used has a strong influence on the approach taken. Using Mentor Graphics DxDesigner and Expedition, the steps taken to implement a working high speed, high density PCB are described.

These steps are described in the context of an actual PCB designed and implemented at NASA's Jet Propulsion Laboratory. The Array Processing Element (APE) board is a custom PCB for a prototype version of a proposed Deep Space Array Based Network [1]. The APE board performs various signal processing operations on digitized signals from antennas in the Deep Space Array Based Network. It contains 12 BGA components with 10 having between 500 to 1152 pins. Most signals were clocked at a rate of 160 MHz and some signals ran at 2.56 GHz and 1280 MHz rates. The chip types included Field Programmable Gate Arrays (FPGAs), a 3 GHz 72x72 crossbar switch and a PowerPC processor. The board used the Advanced Telecommunications Architecture (ATCA) form factor (280 x 322.25 mm) to fit all the required circuitry. The connectivity of the board provided over 40 Gigabit/s data throughput and contained about 3000 nets. The majority of these signals were impedance controlled and over 1200 were differential pairs. A block diagram of the APE board is shown in Figure 1.

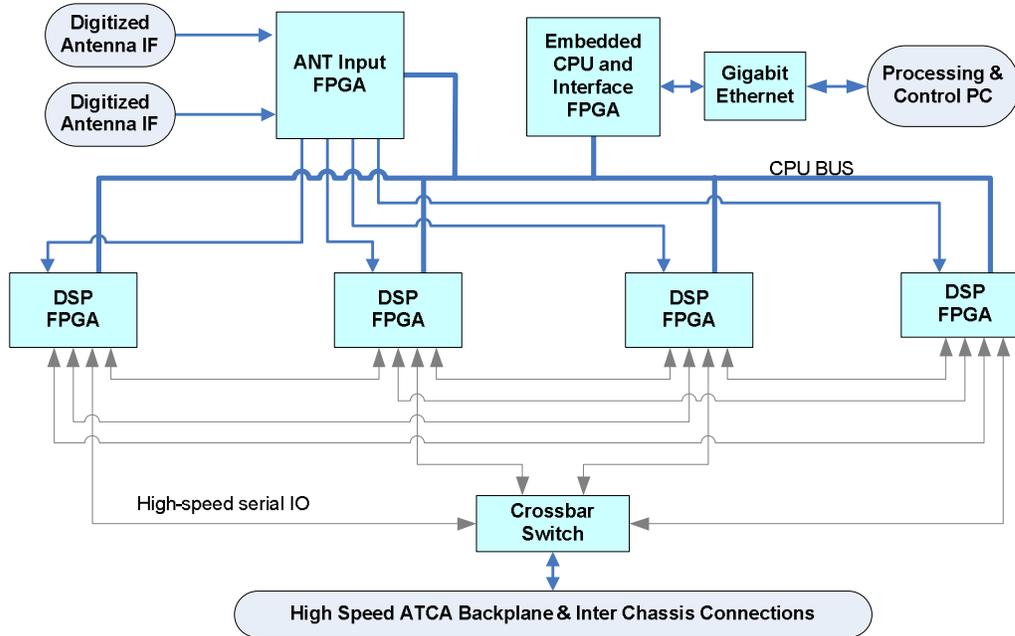


Figure 1: Block Diagram of APE Board.

Design Tools and Design Flow

Mentor Graphics DxDesigner was used for schematic capture and Expedition was used for layout and routing. The first revision of the design was done in ePD 3.1 and WG2002.1. The latest revision of the board was completed in ePD 2005.1. DxDesigner was successfully able to handle a very large design (over 81 pages) and allowed routing constraints such as marking of high speed nets and grouping together pairs of nets as differential pairs. Also, Expedition's ability to handle differential pairs with the high speed design module for both manual routing and autorouting was essential to successful implementation of the APE board. The Constraint Entry System (CES) was not available when the design was started in ePD 3.1 and it was not used later when the design was brought into later versions of DxDesigner and Expedition.

The design flow for using DxDesigner and Expedition was very integrated. Both tools were used throughout the design process. Below, a generalized flow for designing PCB's geared towards DxDesigner and Expedition is offered.

- Start with Block Diagram of Design and major specifications.
- Make necessary library parts. Make schematic and PCB parts simultaneously. Use similar parts as templates.
- Enter preliminary schematic. May leave out some passives and small support components.
- Create PCB netlist. Pass to Expedition.
- Place major parts. Verify that there is enough board space and a suitable placement is possible.
- Define some net constraints and back annotate to DxDesigner.
- Finalize schematic. Modify Library parts if necessary. Do thorough DxDesigner design rule checks.
- Forward annotate to Expedition
- Re-place major components based on final schematic. Use netlines and autoroutes to explore different topologies.
- Decide on board layers, via sizes, trace sizes, clearances. Refine constraints.
- Back annotate to schematic.
- Finalize part placement. Use Cross Probe with schematic to finish placing parts especially passives like bypass capacitors.
- Fanout Power and Ground nets. Modify bypass caps capacitor routes or placement as needed. Fix these nets.

- Hand route critical analog signals, high current signals. Fix these nets.
- Set net order constraints for multinode transmission line nets and determine settings for matched length nets.
- Try simple autorouting strategy. Route all nets at once.
- Perform any special fanouts for BGA parts.
- Refine autorouting strategy. Find best, fastest route. Save for later revisions.
- Iterate Process: may need to unfix nets, move parts, change constraints or clearances.
- Final back annotate to schematic.
- Perform Expedition Design Rule checks. Make Drill files, Assembly & Fabrication drawings and Gerber files. Send Files to design house.

Experience with APE Board

Most of the FPGA, DC-DC converters, high speed connectors, processor and processor support chips for the APE board were not in the part or pattern libraries that came with DxDesigner or Expedition. After making the schematic library parts and PCB library patterns, a few simple style guidelines were established for schematic entry. The schematic sheets were all standardized to size C sheets because this was the largest size that was readable on 8.5 x 11 inch paper. In addition, a flat schematic style was used. Although the schematic was 80 pages, the flat style allowed an easy one to one correspondence with physical layout. All active low signals were marked with a leading “-“ character and all differential signals used a trailing “_p” and “_n” to denote the two polarities. Using the trailing characters for differential signals allowed easier sorting of nets later in Expedition.

An initial schematic containing the most of the major active components and their interconnections was entered in DxDesigner. After a design rules check in DxDesigner, the design was forward annotated to Expedition and a preliminary part placement for these components was performed. Different colors for differential and high speed nets versus normal netlines helped in parts placement. The first priority in part placement was to make sure the parts fit on the board since the board size was fixed. Secondly, using the netlines as a guide congestion and netline length were minimized by eye. Figure 2 shows the parts placement and netlines for larger components on the top side of the PCB.

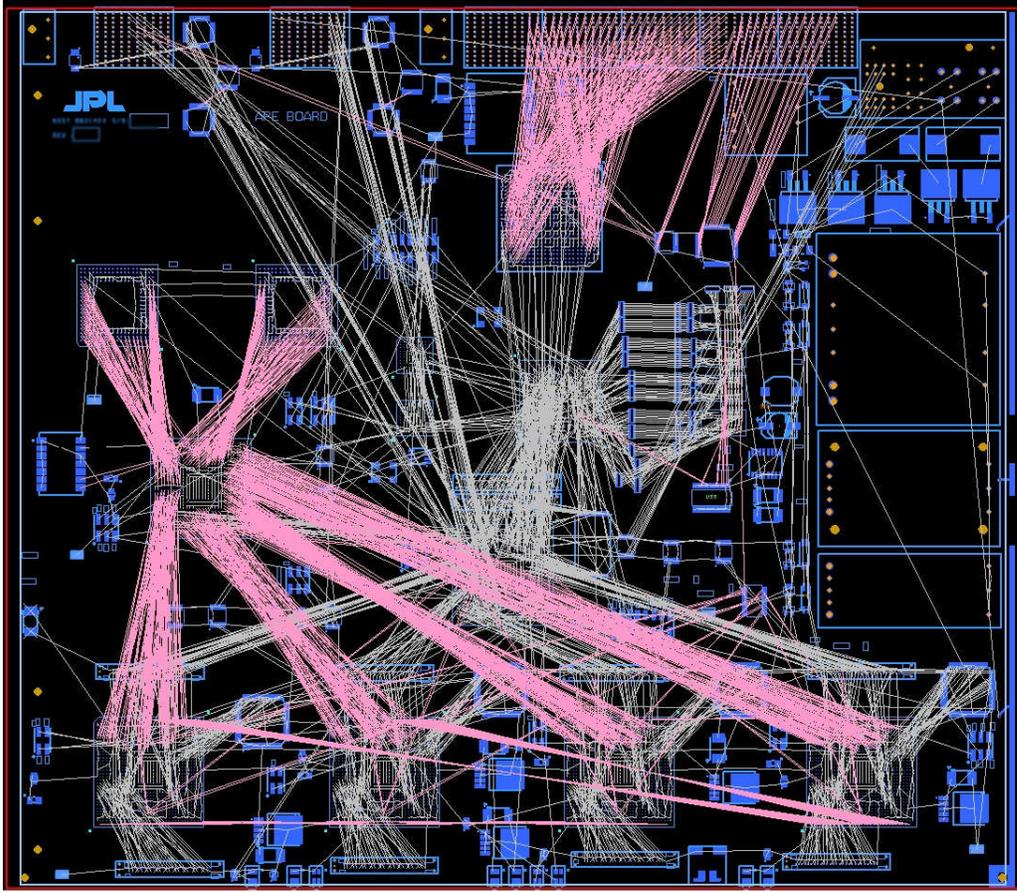


Figure 2: Preliminary Parts Placement with netlines showing.

With these parts placed on the board, a preliminary estimate on board layers, via sizes, trace sizes and net clearances was made. Later in the design process these parameters were refined and improved on an iterative basis. Net classes were also defined in Expedition at this point. Later, when these net classes were back annotated to DxDesigner, individual nets could be assigned to the defined classes. The list below gives some of the net class names that were defined and their purpose:

- Normal – Default net class
- High1280 – High Speed 1280 MHz speed differential signals with matched net length.
- Single50 – Single ended 50 ohm matched impedance lines.
- Diff_pair – 100 ohm differential matched impedance lines 128 to 160 MHz clock rates.
- Rocket – 100 ohm differential matched impedance lines, 2.5 GHz clock rates.
- ChipPwr – Chip Power (+5, +3.3, +1.5V) nets.
- Gnd – Ground nets
- PPCmem – Normal nets grouped together for better routing to processor memory.
- Vregpwr – Voltage regulator power lines.

The board stackup, shown in Figure 3, used 15 layers (8 signal layers, 7 plane layers). Each signal layer, except top and bottom, was sandwiched between a plane layer. This allowed all internal layer nets to be used as stripline transmission lines. Also, it minimized crosstalk between layers. Impedance was controlled by working with the board manufacturer to control dielectric thickness to within 1 mil. Because the BGA patterns were as large as 34x34 arrays with 1.0mm pitch, routing was a challenge in these areas. To increase routing channels on the board, blind vias were used. Top blind vias were defined to connect the top 4 signal layers and bottom blind vias were defined to connect the bottom four routing layers. Together with regular through vias, this via stackup was used for BGA fanouts and differential line nets and is shown in Figure 4.

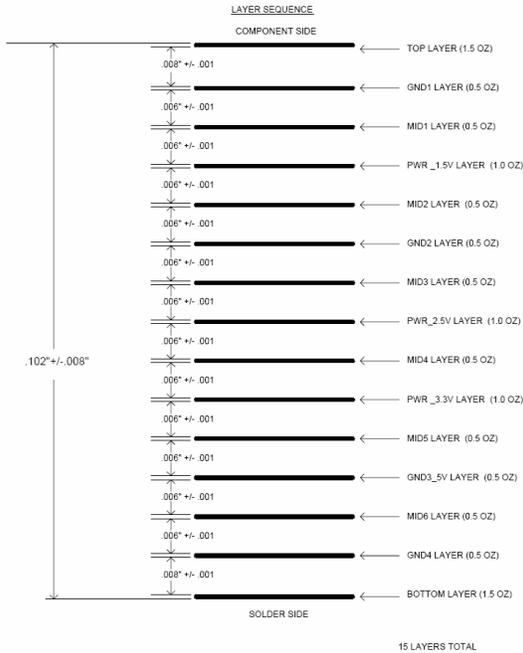


Figure 3: Board Layer Stackup

High Speed Signal Via Stackup			
Via Span	Top Vias	Bottom Vias	Through Vias
Laminate 1-2	Green	Red	Red
Laminate 2-3	Green	Red	Red
Laminate 3-4	Green	Red	Red
Laminate 4-5	Green	Red	Red
Laminate 5-6	Green	Red	Red
Laminate 6-7	Green	Red	Red
Laminate 7-8	Green	Red	Red
Laminate 8-9	Green	Red	Red
Laminate 9-10	Green	Red	Red
Laminate 10-11	Green	Red	Red
Laminate 11-12	Green	Red	Red
Laminate 12-13	Green	Red	Red
Laminate 13-14	Green	Red	Red
Laminate 14-15	Green	Red	Red

Figure 4: Via Stackup for BGA and high speed signal vias.

Using the preliminary board stackup, via sizes, and net classes and clearances, a very rudimentary autoroute was performed using only a fanout and routing pass on all nets. The goal of this autoroute was not 100% completion, but ease of routing and seeing how the autorouter picked routing channels. With this information, the number of layers needed, via sizes, trace sizes and net clearances were further refined. Finally, this information was backannotated to the DxDesigner schematic and placement of the bypass capacitors and other passives commenced.

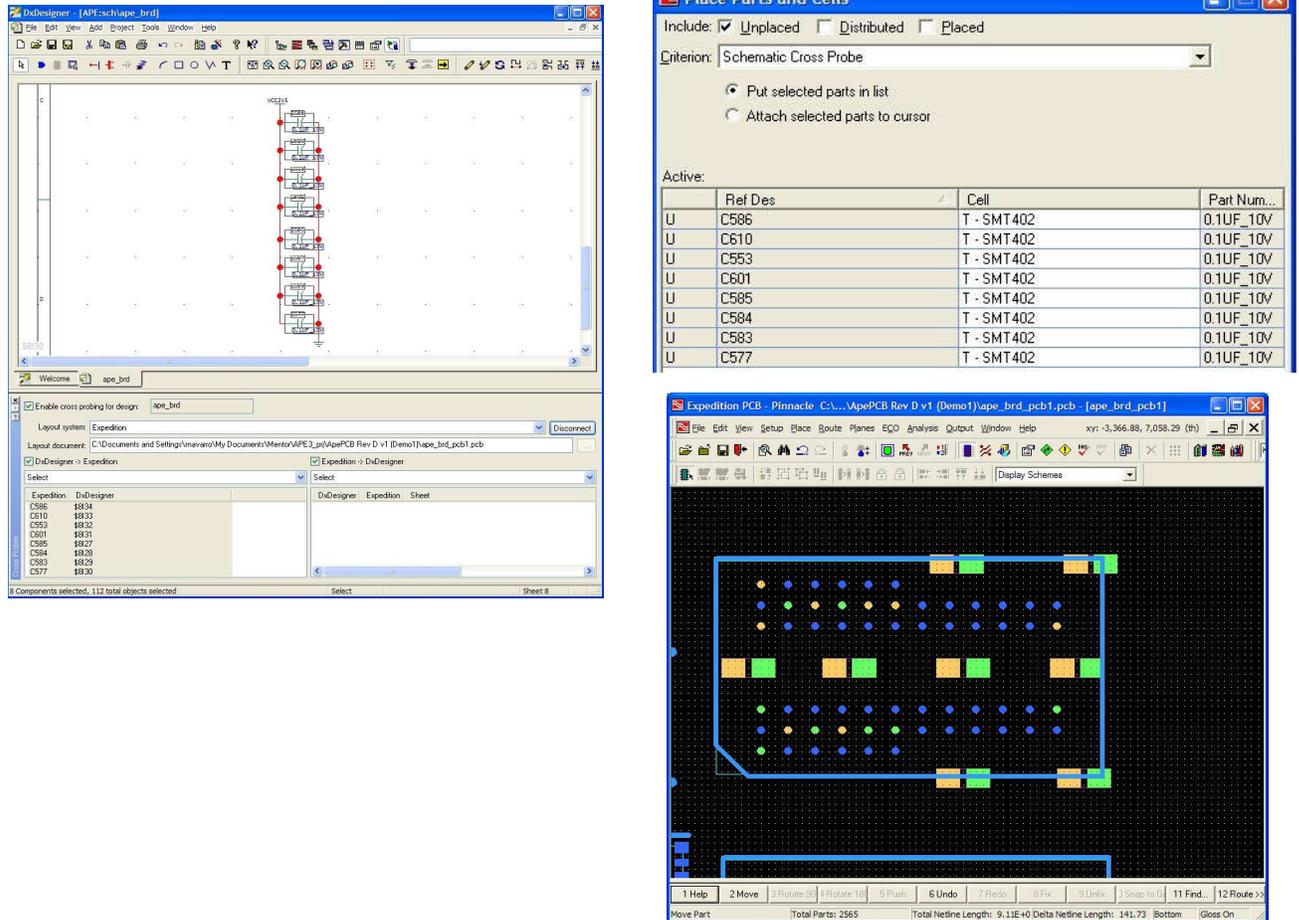


Figure 5: Windows and Dialog Boxes used for Schematic Cross Probe Part Placement

For high speed digital design, bypass capacitors are essential. These passive components must be as close as possible to chips driving high edge rate signals. On the APE board, bypass capacitors for active components were placed on the same page in the schematic for small parts and on adjacent pages for multi-page schematic parts. If it was unclear which part a capacitor went with, a “group” attribute was added to clearly mark which part the component went with. The schematic cross-probe part placement criterion was used to select these parts for placement next to the correct chip. In addition plane nets such as GND and VCC15V, VCC25V and VCC33V were assigned colors to their Netlines, Pads, Vias, Traces and Planes as a visual aid in part placement. Figure 5 shows the various windows used parts placement using cross-probing.

After all the parts for the APE board were placed, the power and ground nets were routed by hand using the fanout and plow tools. Some minor movement of passive parts was done during this time to help get a more direct fanout. Also, the fanout was done in sections depending on whether or not the section would have more direct fanouts with vertical or horizontal routing bias set. After all power and ground nets were fanned out, critical analog signals and high current signals were hand routed with the plow tool. Finally, these nets were fixed so that the autorouter would not change them.

Before autorouting, a few more manual routing steps needed to be taken. First, some of the transmission line nets were multimode and needed their routing order specified in the Net Properties dialog box. Secondly, some of the matched length nets were routed with the manual Route tool followed by the Tune tool in order to experiment with Tune settings. After satisfactory settings were found these nets were fixed. Lastly, updated constraints for the route order and match length nets were back annotated to the schematic.

Another step before autorouting was BGA fanout. On the APE board, the BGA chips were set up to use either through vias or top blind vias for fanout. It was desired for the autorouter to pick the best choice depending on whether the via needed to extend into the bottom 4 routing layers or not. During autorouting, the via constraints were set to use either through, top blind or bottom blind vias. However, if the BGA chips are fanned out with this setting, all the BGA pads are fanned out with top blind vias. So, the solution is to temporarily set the via constraints for the BGA pad nets to use only through vias and fanout those nets. These BGA fanouts must not be fixed before autorouting. During the autorouting process, vias using only the top routing layers will change to top blind vias leaving more routing channels below. This procedure greatly increased the routeability of the board.

Before starting the autorouter, all manually routed connections were checked both manually and with Expedition design rule checks. Furthermore, all fabrication and assembly layers were completed as much as possible. Some fabrication layers, like the drill drawings needed slight modifications after autorouting. Gerber and NC drill settings were defined and silkscreen text was moved into its final position pending autorouting. This version of the PCB was saved as a “gold” version before autorouting. Many autorouting passes were performed and this provided a fixed starting point for each pass.

After each autorouting pass, the result was compared with previous passes and pass definitions were modified in order to converge on the best route. During this process, part placement was modified, some fixed nets were moved, and the schematic was even slightly modified in order to make the board routable. Finally, the planes were processed, fabrication and assembly layers were finalized and Gerber, NC Drill and IPC356 files were generated.

A diagram of the final autorouter passes defined for the APE board is shown below in Figure 6. The use of Net Classes allowed the most critical nets to be routed first. Also, they allowed nets in a certain area to be grouped together. Higher effort routing passes were not performed until all the nets had at least a low effort route in order to avoid cutting off routing channels.

Pass	Pass Type	Items to Route	Order	Start	End	Now	Layers	Via Grid	Rte. Grid	Fix	Pause
<input type="checkbox"/>	Fanout	All Nets	Auto	1	3		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	Route	Nets...	Auto	1	2		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	1	2		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	1	1		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	1	2		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	1	2		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	All Nets	Auto	1	1		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	2	4		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	Net Classes...	Auto	2	4		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	All Nets	Auto	1	4		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Via Min	All Nets	Auto	1	1		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Route	All Nets	Auto	5	5		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	Via Min	All Nets	Auto	2	2		All Enabled	(Default)	(Default)	<input type="checkbox"/>	<input type="checkbox"/>

Figure 6: Autorouting Passes

Conclusion

A general design flow for high speed, high density PCB boards tailored for Mentor Graphics DxDesigner and Expedition was presented. The APE board was presented as an case study of this design flow. The careful planning and organization of this design flow resulted in a successful implementation of this board. Specifying the routing constraints in DxDesigner was useful for both design and documentation of the APE board. The ability to specify more part placement constraints in DxDesigner and pass this information on to Expedition would have been useful. This feature would have been especially helpful in placing the large number of bypass capacitors for the FPGA components.

Acknowledgements

The work reported in this paper was conducted at the Jet Propulsion Laboratory, California Institute of Technology under contract from the National Aeronautics and Space Administration (NASA).

References

- [1] R. Navarro, J. Bunton. "Signal Processing in the Deep Space Array Network," *The Interplanetary Network Progress Report*, 42-157N. May 15 2004. http://tmo.jpl.nasa.gov/progress_report/42-157/157N.pdf